Autonomous AI Security Auditing of Webmin 2.670: 507 Newly Discovered Vulnerabilities

Armin Šupuk (ORCID 0009-0009-7135-1071)

19 November 2026

Contents

Abstract **12**

Introduction **12**

Study design and evidence **12**
AI-assisted research context . . . 12
Finding identity and verification . . . 13
Source revisions . . . 13
CVE cross-references . . . 13
Classification and scoring . . . 13
Host-profile sample and exposure gate . . . 13

Results and interpretation **14**
CVSS assessment coverage . . . 14

Individual paths relevant to one-click RCE **14**
XSS paths without recorded special conditions (73) . . . 15
Privilege escalation paths without recorded special conditions (51) . . . 16

Discussion **17**
A rapid change in security auditing . . . 17
How complete was this audit? . . . 17

Research ethics and automated assistance **17**

Related work **17**

Data and evidence availability **18**

Conclusion **18**

Findings by primary type **18**

Access control and ACL bypass (106) **18**
Core · ACL (8) . . . 18
Core · Integration (3) . . . 21
Core · Webmin (4) . . . 21
Module · Apache (3) . . . 23
Module · At (1) . . . 23
Module · Bind8 (4) . . . 24
Module · Change User (1) . . . 25
Module · Cluster Useradmin (1) . . . 25
Module · Cron (1) . . . 26
Module · Custom (1) . . . 26
Module · DHCP (2) . . . 26
Module · Fdisk (1) . . . 27

Module · Filemin (3) . . . 27
Module · Filter (1) . . . 28
Module · Firewall (4) . . . 28
Module · Firewall6 (4) . . . 30
Module · Fsdump (2) . . . 31
Module · Init (4) . . . 31
Module · Iscsi Client (1) . . . 32
Module · LDAP Client (1) . . . 33
Module · LDAP Users and Groups (1) . . . 33
Module · Logviewer (3) . . . 33
Module · Mailboxes (1) . . . 34
Module · Minecraft (1) . . . 35
Module · Mount (4) . . . 35
Module · MySQL (3) . . . 36
Module · Net (1) . . . 37
Module · PHP Configuration (2) . . . 38
Module · Postfix (6) . . . 38
Module · PostgreSQL (1) . . . 40
Module · Project Meta (1) . . . 40
Module · Quota (1) . . . 41
Module · Samba (1) . . . 41
Module · Servers (1) . . . 41
Module · Software (1) . . . 42
Module · Spam (2) . . . 42
Module · Squid (2) . . . 43
Module · Status (1) . . . 43
Module · Tcpwrappers (1) . . . 43
Module · Time (2) . . . 44
Module · Translations (1) . . . 44
Module · Tunnel (2) . . . 45
Module · Ui Demo (1) . . . 45
Module · Updown (2) . . . 46
Module · Useradmin (5) . . . 46
Module · Usermin (7) . . . 48
Module · Webminlog (1) . . . 50
Module · Xterm (1) . . . 50

Cross-site scripting and HTML injection (140) **51**
Core · Runtime (4) . . . 51
Core · Cross-cutting (1) . . . 52
Core · Integration (1) . . . 52
Core · MiniServ (1) . . . 53
Core · Webmin (3) . . . 53
Module · Apache (4) . . . 54
Module · Backup Config (2) . . . 55
Module · Bacula Backup (4) . . . 56
Module · Bind8 (1) . . . 57
Module · Bsdexports (1) . . . 57
Module · Bsdfdisk (2) . . . 57
Module · Cluster Passwd (1) . . . 58
Module · Cluster Software (3) . . . 58
Module · Cluster Useradmin (5) . . . 59
Module · Cluster Usermin (2) . . . 61
Module · Cluster Webmin (2) . . . 61
Module · Cpan (1) . . . 62
Module · Custom (1) . . . 62
Module · DHCP (5) . . . 63
Module · Documentation (1) . . . 64
Module · Dovecot (2) . . . 65
Module · Exports (1) . . . 65
Module · Fail2Ban (1) . . . 66

Module · Fetchmail (1) . . . 66
Module · Filemin (2) . . . 66
Module · Filter (1) . . . 67
Module · Firewall (4) . . . 67
Module · Firewall6 (1) . . . 68
Module · Htpasswd File (1) . . . 69
Module · Idmapd (1) . . . 69
Module · Inetd (1) . . . 69
Module · Init (2) . . . 70
Module · Ipfw (1) . . . 70
Module · Ipsec (1) . . . 70
Module · Iscsi Target (2) . . . 71
Module · Iscsi Tgtd (1) . . . 71
Module · LDAP Client (2) . . . 72
Module · LDAP Server (3) . . . 72
Module · LDAP Users and Groups (2) . . . 73
Module · Logrotate (1) . . . 74
Module · Logviewer (1) . . . 74
Module · Lpadmin (2) . . . 74
Module · Lvm (2) . . . 75
Module · Mailboxes (4) . . . 76
Module · Minecraft (3) . . . 77
Module · Mount (1) . . . 78
Module · MySQL (4) . . . 78
Module · Net (1) . . . 79
Module · Openslp (1) . . . 80
Module · Package Updates (3) . . . 80
Module · Pam (2) . . . 81
Module · PHP Configuration (1) . . . 81
Module · Postfix (1) . . . 82
Module · Pptp Client (1) . . . 82
Module · Proc (2) . . . 82
Module · Qmailadmin (1) . . . 83
Module · Quota (1) . . . 83
Module · Raid (1) . . . 84
Module · Samba (1) . . . 84
Module · Sarg (1) . . . 84
Module · Servers (6) . . . 85
Module · Shorewall (4) . . . 87
Module · Shorewall6 (1) . . . 88
Module · Smart Status (1) . . . 88
Module · Software (4) . . . 89
Module · Spam (1) . . . 90
Module · Status (3) . . . 90
Module · Stunnel (1) . . . 91
Module · Updown (2) . . . 91
Module · Useradmin (2) . . . 92
Module · Webalizer (1) . . . 93
Module · Webminlog (4) . . . 93

Command and data injection (116) **94**
Core · ACL (1) . . . 94
Core · Command-line utilities (3) . . . 95
Core · Cross-cutting (3) . . . 95
Core · Integration (1) . . . 96
Core · Packaging (1) . . . 97
Module · Apache (3) . . . 97
Module · Backup Config (2) . . . 98
Module · Bandwidth (1) . . . 99
Module · Bind8 (1) . . . 99
Module · Cluster Cron (1) . . . 99

Module · Cluster Useradmin (2) . . . 100
Module · Cluster Webmin (1) . . . 100
Module · Cpan (2) . . . 100
Module · Cron (1) . . . 101
Module · DHCP (8) . . . 101
Module · Filemin (2) . . . 104
Module · Filter (2) . . . 105
Module · Firewall (1) . . . 105
Module · Firewalld (2) . . . 106
Module · Fsdump (1) . . . 106
Module · Heartbeat (1) . . . 107
Module · Inetd (1) . . . 107
Module · Init (1) . . . 107
Module · Ipfw (2) . . . 108
Module · LDAP Client (1) . . . 108
Module · Lpadmin (1) . . . 109
Module · Lvm (1) . . . 109
Module · Mailboxes (2) . . . 109
Module · Minecraft (4) . . . 110
Module · Mount (3) . . . 111
Module · MySQL (7) . . . 112
Module · Net (4) . . . 114
Module · Openslp (1) . . . 115
Module · Package Updates (1) . . . 116
Module · PHP Configuration (4) . . . 116
Module · Postfix (3) . . . 117
Module · PostgreSQL (7) . . . 118
Module · Proftpd (1) . . . 120
Module · Raid (5) . . . 121
Module · Samba (4) . . . 122
Module · Sarg (1) . . . 123
Module · Sendmail (2) . . . 124
Module · Shorewall (1) . . . 124
Module · Shorewall6 (1) . . . 125
Module · Smart Status (1) . . . 125
Module · Smf (1) . . . 125
Module · Software (2) . . . 126
Module · Spam (3) . . . 126
Module · Squid (4) . . . 127
Module · T (1) . . . 128
Module · Time (1) . . . 129
Module · Useradmin (1) . . . 129
Module · Usermin (1) . . . 129
Module · Webalizer (1) . . . 130
Module · Xterm (1) . . . 130
Module · Zones (1) . . . 130

File and path handling (44) **131**
Core · Runtime (3) . . . 131
Core · Webmin (1) . . . 132
Module · Backup Config (2) . . . 132
Module · Filemin (1) . . . 133
Module · Grub2 (1) . . . 133
Module · Help (1) . . . 133
Module · Htaccess Htpasswd (1) . . . 134
Module · Kea Dhcp (1) . . . 134
Module · LDAP Client (1) . . . 134
Module · LDAP Server (1) . . . 135
Module · LDAP Users and Groups (1) . . . 135
Module · Logrotate (2) . . . 135
Module · Mailboxes (2) . . . 136

Module · Minecraft (3) . . . 136
Module · Net (1) . . . 137
Module · Pam (1) . . . 138
Module · PHP Configuration (2) . . . 138
Module · Postfix (2) . . . 139
Module · PostgreSQL (1) . . . 139
Module · Proc (1) . . . 140
Module · Proftpd (1) . . . 140
Module · Samba (3) . . . 140
Module · Sendmail (2) . . . 141
Module · Shell (1) . . . 142
Module · Shorewall (1) . . . 142
Module · Software (1) . . . 143
Module · SSH Server (2) . . . 143
Module · Status (1) . . . 143
Module · Updown (1) . . . 144
Module · Useradmin (1) . . . 144
Module · Vendor Perl (1) . . . 144

Information exposure and secrets (41) **145**
Core · ACL (2) . . . 145
Core · Command-line utilities (1) . . . 145
Core · Runtime (2) . . . 146
Core · Cross-cutting (1) . . . 146
Core · Integration (1) . . . 147
Core · Packaging (3) . . . 147
Module · Cluster Webmin (1) . . . 148
Module · Fsdump (2) . . . 148
Module · Init (1) . . . 149
Module · Iscsi Client (2) . . . 149
Module · Kea Dhcp (1) . . . 149
Module · LDAP Server (4) . . . 150
Module · LDAP Users and Groups (3) . . . 151
Module · Mailboxes (1) . . . 152
Module · MySQL (3) . . . 152
Module · PHP Configuration (1) . . . 153
Module · PostgreSQL (1) . . . 153
Module · Project Meta (1) . . . 154
Module · Samba (1) . . . 154
Module · Servers (2) . . . 154
Module · Squid (1) . . . 155
Module · Status (2) . . . 155
Module · Useradmin (1) . . . 156
Module · Usermin (3) . . . 156

Authentication and session handling (14) **157**
Core · ACL (2) . . . 157
Core · Webmin (2) . . . 158
Module · Exports (1) . . . 158
Module · Fail2Ban (1) . . . 159
Module · Fastrpc (1) . . . 159
Module · File (1) . . . 159
Module · Iscsi Server (1) . . . 160
Module · Lvm (1) . . . 160
Module · Samba (1) . . . 160
Module · Useradmin (1) . . . 161
Module · Usermin (2) . . . 161

Transport and cryptography (19) **162**
Core · Packaging (1) . . . 162
Core · Webmin (2) . . . 162

Module · Cluster Usermin (1) . . . 162
Module · Cluster Webmin (1) . . . 163
Module · Cpan (1) . . . 163
Module · Fsdump (1) . . . 163
Module · LDAP Server (1) . . . 164
Module · LDAP Users and Groups (2) . . . 164
Module · Mailboxes (2) . . . 165
Module · Pptp Client (1) . . . 165
Module · Servers (2) . . . 165
Module · Status (2) . . . 166
Module · Useradmin (1) . . . 167
Module · Usermin (1) . . . 167

Request forgery and origin checks (4) **167**
Core · Cross-cutting (1) . . . 167
Module · Grub2 (1) . . . 168
Module · Mailboxes (1) . . . 168
Module · Minecraft (1) . . . 168

Availability and race conditions (7) **169**
Module · Bind8 (1) . . . 169
Module · Cron (2) . . . 169
Module · DHCP (1) . . . 170
Module · Filemin (1) . . . 170
Module · Package Updates (1) . . . 170
Module · Passwd (1) . . . 171

Other security defects (16) **171**
Core · Webmin (1) . . . 171
Module · Cluster Usermin (1) . . . 171
Module · Cluster Webmin (1) . . . 171
Module · LDAP Client (1) . . . 172
Module · LDAP Server (1) . . . 172
Module · Net (1) . . . 172
Module · Package Updates (1) . . . 173
Module · Pam (2) . . . 173
Module · PostgreSQL (1) . . . 174
Module · Proc (1) . . . 174
Module · Proftpd (1) . . . 174
Module · Project Meta (1) . . . 175
Module · Software (1) . . . 175
Module · SSH Server (1) . . . 175
Module · Status (1) . . . 176

Confirmed historical security issues (567) **176**
ACL · Core · ACL (5) . . . 176
ACL · Core · Runtime (10) . . . 176
ACL · Core · Cross-cutting (1) . . . 177
ACL · Core · Integration (4) . . . 177
ACL · Module · Ajaxterm (1) . . . 178
ACL · Module · Apache (4) . . . 178
ACL · Module · At (2) . . . 178
ACL · Module · Bind8 (1) . . . 178
ACL · Module · Certmgr (1) . . . 178
ACL · Module · Change User (1) . . . 179
ACL · Module · Cluster Passwd (2) . . . 179
ACL · Module · Cluster Webmin (1) . . . 179
ACL · Module · Cron (3) . . . 179
ACL · Module · DHCP (7) . . . 179
ACL · Module · Dovecot (1) . . . 180
ACL · Module · Fail2Ban (2) . . . 180

ACL · Module · Fdisk (1) . . . 180
ACL · Module · File (5) . . . 180
ACL · Module · Filemin (9) . . . 181
ACL · Module · Firewall (3) . . . 181
ACL · Module · Fsdump (1) . . . 182
ACL · Module · Htaccess Htpasswd (2) . . . 182
ACL · Module · Init (1) . . . 182
ACL · Module · Itsecur Firewall (2) . . . 182
ACL · Module · LDAP Server (1) . . . 182
ACL · Module · Lpadmin (1) . . . 183
ACL · Module · Mailboxes (1) . . . 183
ACL · Module · MySQL (3) . . . 183
ACL · Module · Net (3) . . . 183
ACL · Module · Nginx (1) . . . 183
ACL · Module · PHP Configuration (3) . . . 183
ACL · Module · Postfix (1) . . . 184
ACL · Module · PostgreSQL (3) . . . 184
ACL · Module · Proc (2) . . . 184
ACL · Module · Proftpd (1) . . . 184
ACL · Module · Project Meta (1) . . . 184
ACL · Module · Samba (1) . . . 185
ACL · Module · Sendmail (2) . . . 185
ACL · Module · Software (1) . . . 185
ACL · Module · Spam (2) . . . 185
ACL · Module · Squid (2) . . . 185
ACL · Module · SSH Server (2) . . . 185
ACL · Module · Syslog (2) . . . 186
ACL · Module · Updown (1) . . . 186
ACL · Module · Useradmin (3) . . . 186
ACL · Module · Usermin (1) . . . 186
ACL · Module · Vnc (1) . . . 186
ACL · Module · Webminlog (1) . . . 186
ACL · Module · Xterm (2) . . . 187
XSS · Core · ACL (2) . . . 187
XSS · Core · Runtime (9) . . . 187
XSS · Core · Cross-cutting (2) . . . 188
XSS · Core · Integration (5) . . . 188
XSS · Module · Apache (3) . . . 188
XSS · Module · Bacula Backup (2) . . . 189
XSS · Module · Bind8 (3) . . . 189
XSS · Module · Blue Theme (1) . . . 189
XSS · Module · Burner (1) . . . 189
XSS · Module · Certmgr (2) . . . 189
XSS · Module · Cfengine (1) . . . 189
XSS · Module · Cluster Cron (2) . . . 190
XSS · Module · Cluster Shell (4) . . . 190
XSS · Module · Cluster Shutdown (1) . . . 190
XSS · Module · Cluster Software (3) . . . 190
XSS · Module · Cluster Usermin (1) . . . 191
XSS · Module · Cluster Webmin (1) . . . 191
XSS · Module · Cpan (5) . . . 191
XSS · Module · Cron (5) . . . 191
XSS · Module · Custom (1) . . . 192
XSS · Module · Dhcp Dns (1) . . . 192
XSS · Module · Disk Usage (1) . . . 192
XSS · Module · Documentation (4) . . . 192
XSS · Module · Exports Nfs4 (1) . . . 192
XSS · Module · Fail2Ban (1) . . . 192
XSS · Module · Fetchmail (1) . . . 193
XSS · Module · File (2) . . . 193
XSS · Module · Filemin (4) . . . 193

XSS · Module · Filter (4) . . . 193
XSS · Module · Firewall (1) . . . 194
XSS · Module · Fsdump (1) . . . 194
XSS · Module · Gray Theme (2) . . . 194
XSS · Module · Htaccess Htpasswd (1) . . . 194
XSS · Module · Init (2) . . . 194
XSS · Module · Inittab (1) . . . 194
XSS · Module · Itsecur Firewall (1) . . . 195
XSS · Module · LDAP Users and Groups (2) . . . 195
XSS · Module · Link (1) . . . 195
XSS · Module · Mailboxes (10) . . . 195
XSS · Module · Mailcap (1) . . . 196
XSS · Module · Majordomo (2) . . . 196
XSS · Module · Mon (1) . . . 196
XSS · Module · Mount (1) . . . 196
XSS · Module · MySQL (1) . . . 196
XSS · Module · Old Blue Theme (1) . . . 197
XSS · Module · Package Updates (2) . . . 197
XSS · Module · Pam (1) . . . 197
XSS · Module · Passwd (2) . . . 197
XSS · Module · Postfix (1) . . . 197
XSS · Module · PostgreSQL (2) . . . 197
XSS · Module · Proc (2) . . . 197
XSS · Module · Proftpd (1) . . . 198
XSS · Module · Quota (1) . . . 198
XSS · Module · Raid (1) . . . 198
XSS · Module · Servers (3) . . . 198
XSS · Module · Shell (2) . . . 198
XSS · Module · Software (2) . . . 199
XSS · Module · Squid (1) . . . 199
XSS · Module · Ssh (1) . . . 199
XSS · Module · Ssh2 (2) . . . 199
XSS · Module · Status (1) . . . 199
XSS · Module · Syslog (1) . . . 199
XSS · Module · Time (2) . . . 199
XSS · Module · Tunnel (1) . . . 200
XSS · Module · Updown (5) . . . 200
XSS · Module · Useradmin (4) . . . 200
XSS · Module · Usermin (4) . . . 201
XSS · Module · Webminlog (4) . . . 201
XSS · Module · Wuftpd (1) . . . 201
XSS · Module · Xterm (1) . . . 201
Injection · Core · Command-line utilities (1) . . . 202
Injection · Core · Runtime (4) . . . 202
Injection · Core · Cross-cutting (1) . . . 202
Injection · Core · Integration (5) . . . 202
Injection · Core · Webmin (1) . . . 203
Injection · Module · Apache (1) . . . 203
Injection · Module · At (2) . . . 203
Injection · Module · Bacula Backup (1) . . . 203
Injection · Module · Bind8 (1) . . . 203
Injection · Module · Certmgr (3) . . . 203
Injection · Module · Cluster Shutdown (1) . . . 204
Injection · Module · Cluster Software (2) . . . 204
Injection · Module · Cpan (1) . . . 204
Injection · Module · Cron (1) . . . 204
Injection · Module · Custom (4) . . . 204
Injection · Module · Fdisk (10) . . . 204
Injection · Module · Fetchmail (1) . . . 205
Injection · Module · File (5) . . . 205
Injection · Module · Filemin (2) . . . 206

Injection · Module · Filter (1) . . . 206
Injection · Module · Firewalld (1) . . . 206
Injection · Module · Format (1) . . . 206
Injection · Module · Fsdump (4) . . . 206
Injection · Module · Gray Theme (1) . . . 207
Injection · Module · Init (6) . . . 207
Injection · Module · Ipfw (1) . . . 207
Injection · Module · Ipsec (1) . . . 207
Injection · Module · Itsecur Firewall (2) . . . 208
Injection · Module · Lpadmin (2) . . . 208
Injection · Module · Lvm (2) . . . 208
Injection · Module · Mailboxes (1) . . . 208
Injection · Module · Mailcap (1) . . . 208
Injection · Module · Majordomo (2) . . . 208
Injection · Module · Mon (1) . . . 208
Injection · Module · Mount (4) . . . 209
Injection · Module · MySQL (7) . . . 209
Injection · Module · Net (6) . . . 210
Injection · Module · Nis (2) . . . 210
Injection · Module · Package Updates (2) . . . 210
Injection · Module · Postfix (2) . . . 211
Injection · Module · PostgreSQL (4) . . . 211
Injection · Module · Proc (5) . . . 211
Injection · Module · Qmailadmin (1) . . . 212
Injection · Module · Quota (8) . . . 212
Injection · Module · Samba (1) . . . 212
Injection · Module · Sendmail (1) . . . 212
Injection · Module · Software (12) . . . 213
Injection · Module · Squid (2) . . . 214
Injection · Module · SSH Server (3) . . . 214
Injection · Module · Status (2) . . . 214
Injection · Module · Syslog (1) . . . 214
Injection · Module · Updown (2) . . . 214
Injection · Module · Useradmin (3) . . . 215
Injection · Module · Usermin (2) . . . 215
Injection · Module · Vgetty (1) . . . 215
Files and paths · Core · Runtime (3) . . . 215
Files and paths · Core · Cross-cutting (1) . . . 216
Files and paths · Core · Integration (4) . . . 216
Files and paths · Core · Packaging (1) . . . 216
Files and paths · Core · Webmin (1) . . . 216
Files and paths · Module · Ajaxterm (1) . . . 216
Files and paths · Module · Cron (1) . . . 216
Files and paths · Module · File (1) . . . 216
Files and paths · Module · Filemin (5) . . . 217
Files and paths · Module · LDAP Client (1) . . . 217
Files and paths · Module · Mailboxes (3) . . . 217
Files and paths · Module · Nginx (2) . . . 217
Files and paths · Module · PostgreSQL (1) . . . 218
Files and paths · Module · Qmailadmin (1) . . . 218
Files and paths · Module · Syslog (1) . . . 218
Files and paths · Module · Updown (2) . . . 218
Files and paths · Module · Useradmin (2) . . . 218
Files and paths · Module · Webalizer (1) . . . 218
Information exposure · Core · ACL (1) . . . 219
Information exposure · Core · Runtime (7) . . . 219
Information exposure · Core · Cross-cutting (1) . . . 219
Information exposure · Core · Integration (5) . . . 219
Information exposure · Core · Packaging (1) . . . 220
Information exposure · Core · Webmin (2) . . . 220
Information exposure · Module · Ajaxterm (3) . . . 220

Information exposure · Module · Apache (1) . . . 220
Information exposure · Module · Certmgr (3) . . . 220
Information exposure · Module · Cron (1) . . . 221
Information exposure · Module · Documentation (1) . . . 221
Information exposure · Module · Fsdump (1) . . . 221
Information exposure · Module · Itsecur Firewall (1) . . . 221
Information exposure · Module · LDAP Client (1) . . . 221
Information exposure · Module · LDAP Server (1) . . . 221
Information exposure · Module · MySQL (2) . . . 222
Information exposure · Module · Pptp Client (1) . . . 222
Information exposure · Module · Project Meta (2) . . . 222
Information exposure · Module · Status (1) . . . 222
Information exposure · Module · Vnc (2) . . . 222
Information exposure · Module · Xterm (2) . . . 222
Authentication · Core · ACL (1) . . . 223
Authentication · Core · Runtime (11) . . . 223
Authentication · Core · Cross-cutting (3) . . . 224
Authentication · Core · Integration (1) . . . 224
Authentication · Module · Ajaxterm (1) . . . 224
Authentication · Module · Images (1) . . . 224
Authentication · Module · Minecraft (1) . . . 224
Authentication · Module · Passwd (1) . . . 224
Authentication · Module · SSH Server (1) . . . 224
Authentication · Module · Useradmin (1) . . . 225
Authentication · Module · Xterm (5) . . . 225
Transport · Core · Runtime (2) . . . 225
Transport · Core · Integration (2) . . . 225
Transport · Core · Packaging (1) . . . 226
Transport · Core · Webmin (1) . . . 226
Transport · Module · Certmgr (1) . . . 226
Transport · Module · Documentation (1) . . . 226
Transport · Module · Dovecot (2) . . . 226
Transport · Module · Htpasswd File (1) . . . 226
Transport · Module · Project Meta (1) . . . 226
Transport · Module · SSH Server (1) . . . 226
Transport · Module · Translations (1) . . . 227
Request forgery · Module · Ajaxterm (1) . . . 227
Request forgery · Module · Flashterm (1) . . . 227
Request forgery · Module · Ssh2 (1) . . . 227
Availability · Core · Runtime (1) . . . 227
Availability · Module · Dhcp Dns (1) . . . 227
Availability · Module · Itsecur Firewall (1) . . . 227
Availability · Module · Procmail (1) . . . 227
Availability · Module · Proftpd (1) . . . 228
Availability · Module · Servers (1) . . . 228
Availability · Module · Shell (1) . . . 228
Availability · Module · Updown (1) . . . 228
Other · Core · ACL (2) . . . 228
Other · Core · Runtime (2) . . . 228
Other · Core · Webmin (1) . . . 228
Other · Module · Apache (1) . . . 229
Other · Module · Fail2Ban (3) . . . 229
Other · Module · Filemin (1) . . . 229
Other · Module · Filter (1) . . . 229
Other · Module · Firewall (10) . . . 229
Other · Module · Firewall6 (1) . . . 230
Other · Module · Fsdump (1) . . . 230
Other · Module · Ipfilter (1) . . . 230
Other · Module · Itsecur Firewall (2) . . . 230
Other · Module · Nftables (1) . . . 230
Other · Module · Passwd (1) . . . 231

Other · Module · Samba (1) . . . 231
Other · Module · Sentry (1) . . . 231
Other · Module · Servers (1) . . . 231
Other · Module · Software (1) . . . 231
Other · Module · SSH Server (1) . . . 231
Other · Module · Translations (1) . . . 231
Other · Module · Usermin (1) . . . 231
Other · Module · Webmincron (1) . . . 232

Security hardening observations without demonstrated impact (95) 232
Core · Command-line utilities (2) . . . 232
Core · Runtime (8) . . . 232
Core · Cross-cutting (2) . . . 232
Core · Integration (2) . . . 233
Core · Packaging (1) . . . 233
Core · Webmin (1) . . . 233
Module · Apache (4) . . . 233
Module · At (2) . . . 233
Module · Backup Config (3) . . . 234
Module · Bacula Backup (3) . . . 234
Module · Bind8 (1) . . . 234
Module · Bsdfdisk (1) . . . 234
Module · Cluster Cron (1) . . . 234
Module · Cluster Software (1) . . . 234
Module · Cpan (1) . . . 235
Module · Cron (4) . . . 235
Module · Custom (1) . . . 235
Module · DHCP (1) . . . 235
Module · Fail2Ban (1) . . . 235
Module · Fetchmail (1) . . . 235
Module · File (1) . . . 235
Module · Filemin (3) . . . 236
Module · Firewall (1) . . . 236
Module · Grub2 (1) . . . 236
Module · Init (3) . . . 236
Module · Kea Dhcp (1) . . . 236
Module · LDAP Client (1) . . . 236
Module · Logrotate (1) . . . 237
Module · Logviewer (1) . . . 237
Module · Mailboxes (3) . . . 237
Module · MySQL (1) . . . 237
Module · Net (1) . . . 237
Module · Nftables (2) . . . 237
Module · Nis (2) . . . 237
Module · Passwd (1) . . . 238
Module · PHP Configuration (1) . . . 238
Module · Postfix (2) . . . 238
Module · PostgreSQL (1) . . . 238
Module · Proc (1) . . . 238
Module · Proftpd (1) . . . 238
Module · Project Meta (1) . . . 238
Module · Quota (1) . . . 239
Module · Samba (1) . . . 239
Module · Sendmail (1) . . . 239
Module · Servers (2) . . . 239
Module · Smart Status (1) . . . 239
Module · Software (1) . . . 239
Module · Status (1) . . . 239
Module · Syslog Ng (2) . . . 240
Module · System Status (1) . . . 240
Module · Systemd (2) . . . 240

Module · Useradmin (7) . . . 240
Module · Usermin (1) . . . 241
Module · Webminlog (1) . . . 241
Module · Xterm (1) . . . 241
Module · Zones (1) . . . 241

Prior CVE reference set (109 identifiers) **241**

Mapped upstream fixes (135) **243**

References **245**

Research snapshot: 19 November 2026. Generated from the canonical research catalogue and structured applicability profiles.

Abstract

An experienced security researcher translated techniques developed before modern AI into a fully autonomous Codex campaign against Webmin. The campaign delivered 507 canonical, reproduced findings at a pinned Webmin 2.670 revision with 44,297 tracked files, 476,216 Perl/CGI lines overall, and 386,178 production code lines measured by cloc. Its 524 applicability paths have CVSS 4.0 Base assessments and explicit conditions. Execution consumed 8,347,303,160 model tokens; the same token mix at the current Standard API rates would cost about US$4,578. Among paths without recorded special conditions, 48 of 73 XSS and 38 of 51 privilege-escalation paths match module gates present in at least 80% of 6,756 sampled host profiles. The catalogue also records 567 confirmed historical issues and 95 hardening observations. In the author's judgment, the audit captured over 90% of classic defects at the pinned revision, possibly every XSS. The author has observed a sharp rise in AI-produced findings within the past two months, especially in recent weeks. This case study records what a fully autonomous, researcher-designed campaign can achieve at this rapidly changing point.

Introduction

Webmin exposes a broad administrative interface through core services and optional modules. Security impact depends on the route into a module, the permissions held by the acting principal, and deployment-specific services and options. This case study documents the output of a fully autonomous Codex campaign designed by an experienced researcher. Its version-bound catalogue records independently fixable findings, verified applicability paths, explicit prerequisites, CVSS vectors, source links, and a revision-bound inclusion rule. Later independent discoveries at the same revision can test the author's assessment of near-saturation.

Study design and evidence

Item	Audited scope
Audited target	Webmin 2.670 source at `3a2b2b8671c0`
Units	507 canonical findings; 526 reproduced catalogue records
Host-profile sample	6,756 usable module enumerations of 7,684 profile files
Inclusion rule	A `reproduced` verification at the pinned revision
Exclusions	Historical-only candidates and separate duplicate entries

Full audited source revision: `3a2b2b8671c0669c129f5275b3dd1853106ae02b`.

At that revision, the Git tree contains 44,297 tracked files (91.0 MB of blob content). Among them are 3,806 `.pl`, `.pm`, and `.cgi` files containing 476,216 lines, and 126 `module.info` manifests. Separately, cloc 2.04 counts 386,178 nonblank, noncomment code lines in 3,731 production files. That count includes Perl, JavaScript, CSS, shell, HTML, C, Python, and batch files, and excludes test, vendored Perl, demo, localization, documentation, and repository-metadata directories.

AI-assisted research context

The author adapted security-auditing techniques developed before modern AI into a role-based, multi-worker Codex system. After the objective was set, the system autonomously searched, analyzed, tested, reconciled, and

recorded findings. The author was presented with the resulting catalogue. The campaign used the Daybreak Blue routing alias `gpt-daybreak-blue-latest` at `xhigh` reasoning effort [7,10]. The exact task design, orchestration, and lab configuration remain unpublished as the author's long-developed research methods.

The execution records contain 8,312,322,763 input tokens, including 8,158,630,784 cached tokens, and 34,980,397 output tokens: 8,347,303,160 tokens in total. At the published 19 November 2026 Standard API prices for `gpt-5.6-sol`, the current target of the Daybreak Blue alias, the recorded token mix would cost US4, 577.83($US4.00 per million uncached input, US0.40 $cached input, and US$20.00 output) [10,11]. This is an API-equivalent model-token estimate.

We found no earlier publicly documented project-wide AI-assisted Webmin audit in its official security notices or GitHub advisories [8,9].

Finding identity and verification

The target is a pinned source checkout. We selected 526 records with a `reproduced` verification at that revision and excluded later-only and historical-only observations. Identity reconciliation consolidated 19 duplicate records into 507 canonical findings. One finding represents an independently fixable defect; one applicability scenario represents a distinct route or configuration-dependent manifestation. A scenario enters the scored population only when it links to a selected verification. Each entry states the observed behavior and any recorded limits; raw verification records remain private.

Of the 507 selected canonical records, 475 have an audited verification described as runtime, HTTP, or focused execution, while 32 use historical-observation review with latest-stable source and evidence reconciliation. These evidence classes are distinct; a `reproduced` status does not imply that every path was exercised in a live browser or installed package. The audit did not establish that every finding affects every official 2.670 package: a separate release source-divergence review found differences between the tag and published artifacts. That review is retained privately.

Source revisions

The audited source is `3a2b2b8671c0669c129f5275b3dd1853106ae02b`. The checked public upstream source is `0dd66e3d5b5c1fa61212c38d7c8df0d283bb88b8`. Evidence-backed public fix mappings appear in the appendix.

CVE cross-references

Published CVEs were known before the campaign and served as background, not as audit discoveries. The Known Bugs Webmin project currently links 109 distinct CVE identifiers: 107 published and two rejected. The full reference set appears in the appendix. None of the 507 audited-revision findings has a recorded CVE identifier.

Classification and scoring

The recorded `weakness:*`, `boundary:*`, and `impact:*` tags determine the primary display section and path families. The primary section is a deterministic index; a finding may span several weaknesses or impacts. Reviewed applicability profiles record exposure modules, actor routes, verification links, and prerequisites. We score each reviewed path linked to a reproduced finding with the CVSS 4.0 Base metric set [1], retaining the vector and metric rationales in the public data. Evidence references remain in the private scorecard. CVSS measures impact and exploit conditions for the documented path; the host-profile signal is kept separate.

Host-profile sample and exposure gate

The saved inventory was profiled on 2026-11-17 using anonymous version and public-asset module probes. Its inputs came from the project's Censys, Netlas, and other known-host inventory streams. The unit is a profile file for a target endpoint; the corpus was not sampled at random and may contain several endpoints associated with one installation. No exploit requests were sent to these hosts as part of the exposure calculation.

Profile-selection step	Count
Saved target profiles	7,684
Unavailable module enumerations	−13
Completed enumerations	7,671
Only automatically included modules detected	−915

Profile-selection step	Count
Usable module enumerations	**6,756**

Of the usable files, 2,892 have a detected 2.670 version signal, 962 have a different detected version, and 2,902 carried a legacy 2.670 fallback label when the version probe returned no evidence. Those fallback labels were excluded from detected-version counts and vulnerability inference. The accompanying `webmin-fingerprint` v0.1.0 emits `null` for an unobserved Webmin version.

Results and interpretation

The primary index places 140 findings under XSS and HTML injection, 116 under command and data injection, and 106 under access control and ACL bypass. These are display classes, not exclusive weakness counts or a comparison of deployment risk. The catalogue below keeps the component and path-specific conditions visible for each entry.

All listed findings have a recorded reproduction against the audited revision. The imported severity labels are replaced by reviewed CVSS 4.0 Base assessments for 524 of 524 reviewed paths across 507 of 507 canonical findings. Each displayed score includes its vector and is this research project's assessment, not a vendor or NVD rating. CVSS Base excludes the host-profile footprint. The published scorecard is tied to the audited revision.

The per-entry percentage is the share of usable profiles where the exposure module's unauthenticated asset probe was positive; core paths are assigned all usable profiles by definition of the exposure gate. Alternative path modules are combined as a union, counting each profile once. Only paths linked to a finding verification at the audited revision are counted; 11 unlinked alternative scenarios are omitted. Specific ACLs, options, external services, attacker control, and victim action are generally not observable in these profiles, so the displayed module signal is a screening estimate, not a measured vulnerability rate or a statistical upper bound for all Webmin installations. The sample is assembled from known hosts and is not representative of the global installed base; version applicability is not inferred from the anonymous version signal.

Applicability profiles are present and reviewed for all 507 entries. The condition lines name prerequisites supported by the canonical finding evidence; the host corpus does not record whether a given installation meets them. N/A (13 entries) marks installer, development, or unprobed-module paths that the running-host module corpus cannot measure.

CVSS assessment coverage

The 507 findings have 524 reviewed applicability paths. 524 paths have reviewed CVSS 4.0 Base assessments. Path ratings: 10 Critical, 253 High, 248 Medium, 10 Low, 3 None. CVSS Base scores reflect the documented exploit path in the affected configuration; they do not include the host-profile footprint. Vectors and metric rationales appear in public data; source reports remain private. Multiple paths on one finding can have different scores. A score of 0.0 marks a reproduced behavior without a demonstrated CVSS security impact, including the blocked reset-token redemption in WEBMIN-0916. 14 operator-workflow paths have no distinct attacker-route tag; their CVSS privilege metric models the authorized configuration action. 250 paths have no `impact:*` taxonomy tag; their impact metrics were assessed from the finding narratives and linked evidence. CVSS is owned by FIRST and used by permission. See the FIRST CVSS v4.0 specification for the scoring definitions.

Individual paths relevant to one-click RCE

The listed XSS browser-entry paths and privilege-escalation actions were proven end to end individually at the audited Webmin 2.670 revision. A compatible pair on the same host and session can form a one-click RCE route. The tables include reviewed paths with no recorded additional prerequisites. Rank is a dense rank of exposure-gate counts; CVSS orders ties for display. Find each full entry below by its ID.

The exposure gate covers at least 80% of usable profiles for **48/73 XSS paths** (48 findings) and **38/51 privilege-escalation paths** (38 findings). Module gates use positive asset probes; the one core path in each family is assigned to all usable profiles. If all completed enumerations are kept in the denominator (7,671), the same 80% threshold is met by 24 XSS and 18 privilege-escalation paths.

XSS paths without recorded special conditions (73)

Rank	Finding ID	Exposure gate	CVSS-B	Gate share
1	WEBMIN-0518	Core	5.3 Medium	100.00% (6,756/6,756)
2	WEBMIN-0048	MySQL	5.3 Medium	97.10% (6,560/6,756)
2	WEBMIN-0628	MySQL	5.1 Medium	97.10% (6,560/6,756)
3	WEBMIN-2443	Net	4.8 Medium	95.80% (6,472/6,756)
4	WEBMIN-0090	Servers	5.3 Medium	95.66% (6,463/6,756)
4	WEBMIN-0436	Servers	5.1 Medium	95.66% (6,463/6,756)
4	WEBMIN-0773	Servers	5.1 Medium	95.66% (6,463/6,756)
5	WEBMIN-0435	Useradmin	5.3 Medium	95.63% (6,461/6,756)
5	WEBMIN-0236	Useradmin	5.1 Medium	95.63% (6,461/6,756)
6	WEBMIN-0273	Postfix	5.3 Medium	95.40% (6,445/6,756)
6	WEBMIN-0590	PHP Configuration	5.1 Medium	95.40% (6,445/6,756)
7	WEBMIN-0598	Package Updates	5.1 Medium	95.00% (6,418/6,756)
8	WEBMIN-0557	Filemin	5.1 Medium	94.83% (6,407/6,756)
8	WEBMIN-0785	Filemin	4.8 Medium	94.83% (6,407/6,756)
9	WEBMIN-0430	Bind8	5.1 Medium	94.63% (6,393/6,756)
10	WEBMIN-0917	Dovecot	5.1 Medium	94.57% (6,389/6,756)
11	WEBMIN-0034	Mailboxes	5.3 Medium	94.32% (6,372/6,756)
11	WEBMIN-0870	Mailboxes	5.3 Medium	94.32% (6,372/6,756)
11	WEBMIN-0336	Mailboxes	5.1 Medium	94.32% (6,372/6,756)
12	WEBMIN-0855	Fail2Ban	5.1 Medium	93.75% (6,334/6,756)
13	WEBMIN-0007	Apache	5.3 Medium	92.07% (6,220/6,756)
13	WEBMIN-0363	Apache	5.1 Medium	92.07% (6,220/6,756)
13	WEBMIN-0532	Apache	5.1 Medium	92.07% (6,220/6,756)
13	WEBMIN-0605	Apache	5.1 Medium	92.07% (6,220/6,756)
14	WEBMIN-0172	Fetchmail	5.1 Medium	87.30% (5,898/6,756)
15	WEBMIN-0218	Firewall	5.1 Medium	86.68% (5,856/6,756)
15	WEBMIN-0264	Firewall	5.1 Medium	86.68% (5,856/6,756)
15	WEBMIN-0542	Firewall	5.1 Medium	86.68% (5,856/6,756)
16	WEBMIN-0386	Pam	5.1 Medium	86.04% (5,813/6,756)
17	WEBMIN-0159	Raid	7.0 High	85.98% (5,809/6,756)
18	WEBMIN-0043	Webalizer	5.1 Medium	85.60% (5,783/6,756)
19	WEBMIN-0770	Logviewer	5.1 Medium	85.24% (5,759/6,756)
20	WEBMIN-0237	LDAP Server	5.1 Medium	85.01% (5,743/6,756)
21	WEBMIN-0361	Iscsi Target	5.1 Medium	84.93% (5,738/6,756)
22	WEBMIN-0464	DHCP	5.1 Medium	84.87% (5,734/6,756)
22	WEBMIN-0553	DHCP	5.1 Medium	84.87% (5,734/6,756)
22	WEBMIN-0635	DHCP	5.1 Medium	84.87% (5,734/6,756)
22	WEBMIN-0878	DHCP	5.1 Medium	84.87% (5,734/6,756)
23	WEBMIN-0309	Samba	5.1 Medium	84.81% (5,730/6,756)
24	WEBMIN-0548	Shorewall6	5.1 Medium	84.72% (5,724/6,756)
25	WEBMIN-0073	Shorewall	5.1 Medium	84.68% (5,721/6,756)
25	WEBMIN-0132	LDAP Client	5.1 Medium	84.68% (5,721/6,756)
25	WEBMIN-0240	Shorewall	5.1 Medium	84.68% (5,721/6,756)
25	WEBMIN-0344	Shorewall	5.1 Medium	84.68% (5,721/6,756)
25	WEBMIN-0524	Shorewall	5.1 Medium	84.68% (5,721/6,756)
26	WEBMIN-0476	Iscsi Tgtd	6.8 Medium	83.29% (5,627/6,756)
27	WEBMIN-0452	Bacula Backup	5.3 Medium	81.88% (5,532/6,756)
27	WEBMIN-0721	Bacula Backup	5.1 Medium	81.88% (5,532/6,756)
28	WEBMIN-0359	Proc	2.4 Low	20.43% (1,380/6,756)
29	WEBMIN-0118	Updown	5.3 Medium	5.60% (378/6,756)
29	WEBMIN-0230	Updown	5.1 Medium	5.60% (378/6,756)
30	WEBMIN-0751	Custom	5.1 Medium	5.52% (373/6,756)
31	WEBMIN-0149	Init	5.1 Medium	5.45% (368/6,756)
32	WEBMIN-0293	Webminlog	5.1 Medium	5.43% (367/6,756)
32	WEBMIN-0427	Webminlog	5.1 Medium	5.43% (367/6,756)
32	WEBMIN-0734	Webminlog	5.1 Medium	5.43% (367/6,756)
32	WEBMIN-0954	Webminlog	5.1 Medium	5.43% (367/6,756)

Rank	Finding ID	Exposure gate	CVSS-B	Gate share
33	WEBMIN-0031	Webmin	5.3 Medium	5.39% (364/6,756)
34	WEBMIN-0566	Backup Config	5.3 Medium	5.31% (359/6,756)
34	WEBMIN-0424	Backup Config	5.1 Medium	5.31% (359/6,756)
35	WEBMIN-0275	Cpan	5.1 Medium	4.25% (287/6,756)
36	WEBMIN-0015	LDAP Users and Groups	5.3 Medium	4.17% (282/6,756)
37	WEBMIN-0098	Lpadmin	5.3 Medium	4.14% (280/6,756)
37	WEBMIN-2405	Lpadmin	4.8 Medium	4.14% (280/6,756)
38	WEBMIN-0216	Lvm	5.1 Medium	1.12% (76/6,756)
38	WEBMIN-0298	Lvm	5.1 Medium	1.12% (76/6,756)
38	WEBMIN-0617	Software	5.1 Medium	1.12% (76/6,756)
38	WEBMIN-0651	Software	4.8 Medium	1.12% (76/6,756)
39	WEBMIN-0178	Zones	5.1 Medium	0.64% (43/6,756)
40	WEBMIN-0367	Minecraft	5.1 Medium	0.04% (3/6,756)
40	WEBMIN-0740	Minecraft	5.1 Medium	0.04% (3/6,756)
41	WEBMIN-0177	Htpasswd File	5.1 Medium	0.01% (1/6,756)
42	WEBMIN-0618	Pptp Client	5.1 Medium	0.00% (0/6,756)

Privilege escalation paths without recorded special conditions (51)

Rank	Finding ID	Exposure gate	CVSS-B	Gate share
1	WEBMIN-0714	MySQL	8.7 High	97.10% (6,560/6,756)
1	WEBMIN-0880	MySQL	8.7 High	97.10% (6,560/6,756)
1	WEBMIN-0668	MySQL	7.1 High	97.10% (6,560/6,756)
2	WEBMIN-0029	Spam	7.1 High	96.64% (6,529/6,756)
3	WEBMIN-0437	Servers	7.1 High	95.66% (6,463/6,756)
4	WEBMIN-0398	Postfix	8.7 High	95.40% (6,445/6,756)
4	WEBMIN-0456	PHP Configuration	8.7 High	95.40% (6,445/6,756)
4	WEBMIN-0515	Postfix	8.7 High	95.40% (6,445/6,756)
4	WEBMIN-0686	Postfix	8.7 High	95.40% (6,445/6,756)
4	WEBMIN-0801	Postfix	7.1 High	95.40% (6,445/6,756)
5	WEBMIN-0313	Mount	8.7 High	95.17% (6,430/6,756)
5	WEBMIN-0536	Mount	7.1 High	95.17% (6,430/6,756)
6	WEBMIN-0967	Usermin	8.7 High	94.75% (6,401/6,756)
7	WEBMIN-0595	Bind8	7.1 High	94.63% (6,393/6,756)
8	WEBMIN-0153	Mailboxes	7.1 High	94.32% (6,372/6,756)
8	WEBMIN-0401	Mailboxes	5.8 Medium	94.32% (6,372/6,756)
9	WEBMIN-0591	Firewalld	8.7 High	93.04% (6,286/6,756)
9	WEBMIN-0907	Firewalld	8.7 High	93.04% (6,286/6,756)
10	WEBMIN-0045	Raid	8.7 High	85.98% (5,809/6,756)
10	WEBMIN-0158	Raid	8.7 High	85.98% (5,809/6,756)
10	WEBMIN-0455	Raid	8.7 High	85.98% (5,809/6,756)
10	WEBMIN-0473	Raid	8.7 High	85.98% (5,809/6,756)
10	WEBMIN-0484	Raid	8.7 High	85.98% (5,809/6,756)
10	WEBMIN-0159	Raid	7.1 High	85.98% (5,809/6,756)
11	WEBMIN-0482	Squid	8.7 High	85.82% (5,798/6,756)
11	WEBMIN-0645	Squid	7.1 High	85.82% (5,798/6,756)
12	WEBMIN-0832	Logviewer	8.7 High	85.24% (5,759/6,756)
12	WEBMIN-2453	Logviewer	7.1 High	85.24% (5,759/6,756)
13	WEBMIN-0500	Sarg	8.7 High	85.20% (5,756/6,756)
14	WEBMIN-0351	Openslp	8.7 High	85.08% (5,748/6,756)
15	WEBMIN-0671	Smf	8.7 High	84.92% (5,737/6,756)
16	WEBMIN-0128	DHCP	7.1 High	84.87% (5,734/6,756)
16	WEBMIN-0227	DHCP	7.1 High	84.87% (5,734/6,756)
16	WEBMIN-0311	DHCP	7.1 High	84.87% (5,734/6,756)
16	WEBMIN-0529	DHCP	7.1 High	84.87% (5,734/6,756)
17	WEBMIN-0814	Samba	8.7 High	84.81% (5,730/6,756)
18	WEBMIN-0343	Shorewall6	8.7 High	84.72% (5,724/6,756)

Rank	Finding ID	Exposure gate	CVSS-B	Gate share
19	WEBMIN-0072	Shorewall	8.7 High	84.68% (5,721/6,756)
20	WEBMIN-0117	Init	7.1 High	5.45% (368/6,756)
21	WEBMIN-0256	Webmin	7.1 High	5.39% (364/6,756)
22	WEBMIN-0535	Time	8.7 High	5.37% (363/6,756)
23	WEBMIN-0680	Proftpd	8.7 High	5.28% (357/6,756)
23	WEBMIN-0027	Proftpd	7.1 High	5.28% (357/6,756)
24	WEBMIN-0499	Cpan	7.7 High	4.25% (287/6,756)
25	WEBMIN-0052	Smart Status	8.7 High	4.23% (286/6,756)
26	WEBMIN-0242	Heartbeat	8.7 High	4.10% (277/6,756)
27	WEBMIN-0379	Lvm	8.7 High	1.12% (76/6,756)
28	WEBMIN-0481	Minecraft	8.7 High	0.04% (3/6,756)
28	WEBMIN-0555	Minecraft	8.7 High	0.04% (3/6,756)
28	WEBMIN-0596	Minecraft	8.7 High	0.04% (3/6,756)
28	WEBMIN-0520	Minecraft	7.1 High	0.04% (3/6,756)

Discussion

A rapid change in security auditing

In the author's personal observation, the rate at which AI systems produce security findings has risen massively within the past two months, with the steepest visible increase in the past few weeks. The author cannot identify a single date for the change. The present case study captures a tipping point: an experienced researcher could encode earlier audit techniques into a fully autonomous Codex campaign and receive a large, source-linked, reproduced catalogue from a mature administrative application. The same methods and model capabilities are changing quickly, so this result serves as a dated record of what was possible at the time of the campaign.

How complete was this audit?

Drawing on years of security-research experience and the breadth of the campaign, the author believes it found more than 90% of the classic security defects in the pinned Webmin source, particularly XSS and authorization or privilege escalation flaws. The author considers it plausible that the campaign found every XSS present there. The project's linked CVE reference set contains 107 published identifiers accumulated across Webmin releases; this campaign documented 507 independently fixable findings at the single audited revision. In the author's judgment, the catalogue is close to complete for these classic classes. Some further issues may emerge, but the number of new findings against this exact revision should decline dramatically. Later independent reports can test that assessment by comparing each root cause and manifestation with the catalogue.

The author plans to push a pull request to Webmin with fixes for all catalogue findings that still require changes on upstream master.

Research ethics and automated assistance

The host-profile analysis used previously saved anonymous asset and version probes; the paper reports aggregate counts and omits target addresses and raw profiles. The autonomous Codex campaign performed discovery, triage, validation, evidence organization, and classification. Inclusion is tied to recorded verification at the pinned revision. The author designed the campaign and assumes responsibility for the published claims.

Related work

The OWASP Web Security Testing Guide [2] organizes web-application tests around explicit controls and evidence of failure. Mu et al. [3] show why reproducibility details matter when interpreting reported vulnerabilities. Li et al. [5] find that project-scale LLM vulnerability detectors and agents can uncover distinct flaws while still suffering high false-discovery rates, incomplete context, and high costs. Sheng et al. [6] demonstrate automated discovery and reproduction. This case study keeps finding identity, applicability paths, and source observations separate; CWE [4] supplies a broader weakness vocabulary, and FIRST's CVSS v4.0 [1] supplies scoring definitions.

Data and evidence availability

This paper contains the included finding list, selection rule, reviewed path conditions, and links to the pinned public Webmin source where a direct source location was recorded. The accompanying public data archive `public-data.zip` contains structured data for 507 audited-revision findings, 524 included paths, 567 confirmed historical issues, and 95 hardening observations: prerequisites, exposure gates, CVSS vectors and rationales, ranking rows, repository-size measurements, anonymous module aggregates, campaign token totals, and mapped public fix commits. The companion MIT-licensed `webmin-fingerprint` release 10.5281/zenodo.24367586 provides its source archive, an npx-ready npm package, and a Linux x86-64 binary, with SHA-256 checksums.

Conclusion

A researcher-designed autonomous AI campaign documented 507 canonical findings and 524 reviewed applicability paths at one pinned Webmin 2.670 source revision using 8,347,303,160 model tokens. It demonstrates the present capacity of a fully autonomous Codex audit designed from an experienced researcher's methods. The author believes the catalogue approaches saturation for classic issues at this revision. Later independent findings will test that judgment. The complete cohort follows for finding-level review.

Findings by primary type

Type	Canonical findings
Access control and ACL bypass	106
Cross-site scripting and HTML injection	140
Command and data injection	116
File and path handling	44
Information exposure and secrets	41
Authentication and session handling	14
Transport and cryptography	19
Request forgery and origin checks	4
Availability and race conditions	7
Other security defects	16
Total	**507**

The primary type is selected deterministically from the existing `weakness:*`, `boundary:*`, and `impact:*` catalogue tags. A finding may have additional weaknesses or impacts; the section heading is an index, not a complete taxonomy. The final section collects uncommon or unclassified cases.

Access control and ACL bypass (106)

Core · ACL (8)

WEBMIN-0143 — Global ACL save paths do not enforce ACL-edit permission

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The ACL module's user and group editors show global ACL controls only when the acting operator has ACL-edit permission, but save_user.cgi and save_group.cgi accept the posted acl_security_form marker and persist global ACL settings without rechecking that permission. These settings govern cross-module restrictions including filesystem root, dot-file visibility, read-only mode, and download-address policy. A direct request therefore crosses the delegation boundary even though the ordinary UI hides the controls.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: ACL delegate may save users or groups but is denied global ACL editing. [nondefault]

Audited source: `acl/edit_user.cgi`, `acl/save_user.cgi`.

WEBMIN-0144 — Group save endpoint bypasses the groups permission

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

edit_group.cgi rejects an operator whose ACL module permissions set groups=0, but save_group.cgi has no equivalent server-side gate before it updates or creates groups and propagates their module lists. A crafted direct POST can therefore mutate group privileges despite the explicit denial, including adding a privileged module to all members of a chosen group.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: ACL delegate is denied group administration but retains access to the direct group-save endpoint. [nondefault]

Audited source: `acl/edit_group.cgi`, `acl/save_group.cgi`.

WEBMIN-0145 — Client-certificate request handlers ignore the cert permission

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The ACL module defines cert as the permission controlling whether a Webmin user can request a client certificate, but cert_form.cgi and cert_issue.cgi do not test it. An authenticated operator with cert=0 can still open the request form and reach issuance-handler validation. When Webmin's client-certificate authority is configured and the request contains a valid browser-generated key, this exposes an alternative-login credential workflow that delegation policy explicitly denied.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: ACL delegate is denied certificate administration while a CA is configured. [nondefault]

Audited source: `acl/defaultacl`, `acl/acl_security.pl`.

WEBMIN-0224 — Password-policy delegates can administer the external identity store

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The ACL module displays external SQL/LDAP identity-store controls only to principals with the sql permission, but its configuration-save, SQL-table creation, and LDAP-base provisioning handlers authorize on the separate pass permission. A constrained authenticated administrator with pass=1 and sql=0 can therefore cross the intended delegation boundary and modify the server-wide authentication backend. The latest-stable reproduction confirmed that this principal receives a normal redirect after posting to save_sql.cgi.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: ACL delegate has password-policy permission but is denied external identity-store administration. [nondefault]

Audited source: `acl/edit_sql.cgi`, `acl/save_sql.cgi`.

WEBMIN-0396 — Group ACL saving bypasses ACL/module restrictions and permits path traversal

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

save_acl.cgi treats a non-empty _acl_group as sufficient to enter the group branch and checks only access.groups. It does not require access.acl, does not validate _acl_mod against the caller's permitted modules, does not verify that the submitted group exists before the write, and constructs the ACL path from both request values. The local helper likewise writes or unlinks config_directory/module/group.gacl without containment checks. This permits unauthorized group ACL changes and path traversal to an attacker-selected .gacl target.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: The escalation depends on configured ACL/module restrictions. [nondefault]

Audited source: `acl/save_acl.cgi`, `web-lib-funcs.pl`.

WEBMIN-0522 — Group deletion can orphan anonymous access onto permissive module defaults

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The direct deletion endpoints reject a Webmin user referenced by miniserv's anonymous mapping, but delete_group.cgi and delete_groups.cgi require only group-management permission and pass each member to delete_user() without the same check. delete_user() removes the account, module assignment, and per-user ACL files while leaving the anonymous mapping untouched. A request admitted by the surviving mapping still receives the deleted username, and module ACL loading falls back to module defaults when its per-user ACL is absent. In the isolated latest-stable lab, a synthetic operator with groups=1, delete=0, and users=operator deleted a group containing the anonymous backing identity; the identity and group disappeared, the mapping remained, and an unauthenticated request to the mapped ACL index returned HTTP 200 with relevant defaults create=1, delete=1, groups=1, and users=*. The path requires the stated anonymous-mapping configuration and delegated group-management permission.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: Anonymous Webmin access maps to a user who belongs to a group a delegate can delete. [nondefault]

Audited source: `acl/delete_user.cgi`, `acl/delete_group.cgi`.

WEBMIN-0916 — Delegated ACL administrator can mint a reset token for an out-of-scope Unix account

CVSS 4.0 Base (CVSS-B): 0.0 (None) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:N/SC:N/SI:N/SA:N`.

The administrator-side reset handler authorizes only the Webmin account supplied as user or user_acc, while a separate request value selects the Unix account placed in the token as uuser. Existence and sudo-capability checks establish that the Unix identity is valid, but do not establish that the delegated caller is allowed to reset it. The token consumer uses uuser in its Unix password-update branch. Consequently, an authenticated ACL administrator permitted to edit a low-privilege Webmin account can directly request a token for a different sudo-capable Unix account, crossing the configured target boundary. Frozen stable source retains this separation. The latest-stable lab confirmed issuance of a token with uuser=root for a caller authorized only over target, although the final password change was not reproduced because an independent hostname-mismatch check rejected that admin-generated token.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: A delegated ACL administrator may edit a low-privilege Webmin account but not the selected Unix account. [nondefault]; The selected Unix account exists and satisfies the sudo-capability check; the demonstrated target was root. [environmental]

Audited source: `acl/forgot_send.cgi`, `forgot.cgi`.

WEBMIN-0988 — Safe users can inherit newly added permissive ACL capabilities

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

New safe users do not persist the negative global ACL marker intended to prevent future permission inheritance. The edit page sets negative=1 only in a local object, suppresses the global ACL form for safe users, and the safe save branch instead persists the root safe profile, which contains negative=0. Consequently, when an accessible module later adds an allow-by-default ACL key with a restrictive negative ACL value, the safe account inherits the permissive default. The frozen latest-stable container reproduction demonstrates this behavior and the negative=1 deny control.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: A safe user is granted a module that later gains a permissive allow-by-default ACL key. [nondefault]

Audited source: `acl/edit_user.cgi`, `acl/save_user.cgi`.

Core · Integration (3)

WEBMIN-0333 — Configured HTTP tunnel can connect outside its allowed target and forward credentials

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The HTTP tunnel parses the request-supplied URL into connection host, port, and path separately from its configured target validation. A target that passes or bypasses the string check can direct the server to another authority. Fixed or variable upstream Basic credentials are then attached to that request, and upstream HTML is relayed through the Webmin endpoint. This creates authenticated SSRF with credential disclosure and same-origin active-content exposure.

Profile-based footprint: 4.13% observed module signal (279/6,756 usable profiles; observed gate: tunnel).

Recorded conditions: Tunnel is configured with a target restriction and upstream Basic credentials. [nondefault]; An authenticated Tunnel user chooses another host reachable by the Webmin server. [environmental]

Audited source: `tunnel/link.cgi`, `tunnel/tunnel-lib.pl`.

WEBMIN-0668 — Restricted MySQL module users can force-reset an arbitrary database account

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The MySQL login endpoint accepts a per-user database ACL as sufficient authorization, but it takes login, password, and force from the request and calls the root-executed force-reset routine before updating that ACL. A restricted module user can therefore request a different database account such as root; the sink stops the database, starts it with grant checks disabled, and applies a password-change query. This crosses the intended per-user and per-database authorization boundary.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

WEBMIN-0722 — MySQL custom backup parameters bypass the authorized destination and backup user

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The MySQL backup form accepts arbitrary mysqldump options. Although tokens are shell-quoted, options such as --result-file retain their program semantics and make the root-context mysqldump process write outside the selected backup path; the configured backup user applies only to the pipeline writer.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL backup delegate has a destination restriction or demoted writer but can add custom mysqldump options. [nondefault]

Audited source: `mysql/backup_form.cgi`, `mysql/mysql-lib.pl`.

Core · Webmin (4)

WEBMIN-0032 — Mandatory update signatures fail open when GnuPG setup fails

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Both update-list and module-file verification are entered only when gnupg_setup reports success. When setup fails, even documented mode 2 (must check) bypasses signature download and verification, then returns parsed metadata or a successful module check. An attacker able to influence the update channel can therefore defeat the requested authenticity policy.

Profile-based footprint: 5.39% observed module signal (364/6,756 usable profiles; observed gate: webmin).

Recorded conditions: Mandatory update-signature mode 2 is configured. [nondefault]; GnuPG setup fails when an update is installed. [environmental]

Audited source: webmin/webmin-lib.pl.

WEBMIN-0366 — IP-only proxy trust is not restricted to configured proxy sources

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

Miniserv restricts forwarded client-address headers to trusted_proxies only when that list is non-empty. The access form requires a proxy entry for combined IP-and-certificate trust but permits IP-only trust with an empty list. In that configuration, any directly connected client can set the address subsequently used by per-user IP checks and logging. The Docker reproduction showed a loopback request denied by a per-user allow rule becoming accepted when it supplied the allowed address in X-Forwarded-For.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: IP-only proxy trust is enabled without a trusted-proxy source list. [nondefault]; A client can reach MiniServ directly and supply a forged forwarded-address header. [environmental]

Audited source: miniserv-lib.pl, webmin/change_access.cgi.

WEBMIN-0744 — Granular Webmin Configuration ACL does not protect lock termination endpoint

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N.

The Webmin Configuration index applies per-page allow/disallow ACLs only to visible links. kill_lock.cgi does not enforce the lock-page ACL before enumerating active locks and signaling a matching PID. An authenticated module user restricted to another page can therefore directly terminate a root Webmin process that holds a valid lock, causing denial of service and risking interruption of a privileged configuration operation.

Profile-based footprint: 5.39% observed module signal (364/6,756 usable profiles; observed gate: webmin).

Recorded conditions: Webmin Configuration delegate is denied lock-page access but can reach kill_lock.cgi. [nondefault]

Audited source: webmin/index.cgi, webmin/kill_lock.cgi.

WEBMIN-2399 — TLS cipher-list rejection is not detected

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N.

MiniServ wraps Net::SSLeay::set_cipher_list in eval and reports only a non-empty Perl exception. Net::SSLeay reports an invalid cipher expression by returning false without throwing, so an intended cipher restriction can fail without the configured error being logged or acted upon, leaving the active TLS policy different from the administrator's expectation.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: MiniServ is configured with a TLS cipher-list expression that the SSL library rejects without throwing. [nondefault]

Audited source: miniserv-lib.pl.

Module · Apache (3)

WEBMIN-0008 — Client-controlled prior target bypasses alias directory restriction

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Alias save logic allows a disallowed target when an old_to request field is present and itself outside the allowed alias root. Because old_to is a hidden client field rather than server-derived state, a restricted editor can forge it and save a new Alias or ScriptAlias pointing outside the permitted directory. The frozen newest-stable Docker harness reproduced acceptance of /etc/shadow while the simulated allowed root was /srv/allowed.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: Apache delegate has an alias target directory restriction and permission to save aliases. [nondefault]

Audited source: `apache/mod_alias.pl`.

WEBMIN-0305 — Direct virtual-server save bypasses Virtualmin ownership restriction

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Apache module code identifies a virtual host managed by Virtualmin and withholds its address/root/name editor, while the generic directive saver also filters Virtualmin-owned fields on the server. However, apache/save_vserv.cgi—the direct target of the withheld form—accepts an authorized direct request without evaluating is_virtualmin_domain. It then changes or deletes the virtual-host structure and writes DocumentRoot and ServerName. An authenticated Apache-module user who is allowed to edit that virtual host and has vaddr can therefore cross the intended Virtualmin ownership boundary. The path requires authenticated, target-specific module privileges and can cause service redirection, configuration desynchronization, or outage.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: Apache delegate has virtual-host editing and vaddr permission for a Virtualmin-owned host. [nondefault]

Audited source: `apache/virt_index.cgi`, `apache/save_virt.cgi`.

WEBMIN-2467 — Create-only Apache user can add a global wildcard Listen port

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The Apache virtual-host creation endpoint authorizes the request with the create ACL, but its wildcard custom-port path can append a global numeric Listen directive without checking the separate global-options ACL or the directive allow/deny policy. A delegated create-only principal can therefore persist an additional all-interface listener that may become active on a later Apache reload or restart. The caller explicitly selects listening, the port is restricted to digits, and configuration activation occurs on a later Apache reload or restart.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: Apache delegate may create virtual hosts but is denied global Listen configuration. [nondefault]; The creation request asks for a wildcard custom port and Apache is later reloaded for active exposure. [environmental]

Audited source: `apache/create_virt.cgi`, `apache/acl_security.pl`.

Module · At (1)

WEBMIN-0545 — Restricted Webmin user can run another user's At job

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The single-job endpoint resolves an arbitrary request-supplied job ID from the complete At spool and then handles run and delete operations differently. Deletion obtains the caller's module ACL and requires can_edit_user for the job owner, but the run branch immediately writes the stored job to a temporary executable and calls proc::safe_process_exec with the job owner's UID and GID. Other reviewed paths filter or reject jobs through can_edit_user. Consequently, an authenticated At-module user restricted to a different Unix identity can bypass that restriction by directly posting id and run=1, trigger another user's stored command early or repeatedly, and receive its output. A frozen latest-stable reproduction used a harmless root-owned synthetic job: the mode=3 session was denied by edit_job.cgi, yet delete_job.cgi executed the same job as UID 0.

Profile-based footprint: 4.35% observed module signal (294/6,756 usable profiles; observed gate: at).

Recorded conditions: At delegate can access the module but is denied control of the selected job owner's account. [nondefault]

Audited source: `at/delete_job.cgi`, `at/edit_job.cgi`.

Consolidated reproduction records: `webmin-0176` (Restricted At-module user can force execution of another user's queued job).

Module · Bind8 (4)

WEBMIN-0017 — Find Free IPs bypasses per-zone authorization

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

The Find Free IPs CGI resolves an arbitrary requested zone and reads its records when access.findfree is enabled, but does not apply can_edit_zone to the resolved object. Per-zone, per-view, and directory restrictions are therefore not enforced on this read workflow. A restricted authenticated module user can query record-derived address occupancy and host information for another configured zone.

Profile-based footprint: 94.63% observed module signal (6,393/6,756 usable profiles; observed gate: bind8).

Recorded conditions: BIND delegate has Find Free IPs permission but is denied the requested zone by per-zone, view, or directory scope. [nondefault]; The target BIND zone is configured and contains records from which address occupancy can be inferred. [environmental]

Audited source: `bind8/find_free.cgi`, `bind8/bind8-lib.pl`.

WEBMIN-0019 — BIND lifecycle handlers collapse distinct apply scopes

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N`.

The BIND ACL models apply=2 as zone-only and apply=3 as whole-configuration-only. start.cgi and stop.cgi nevertheless authorize both modes with a truthiness test and immediately invoke start_bind or stop_bind. Hiding lifecycle links does not protect direct requests, so a scoped module user can control DNS service availability outside the granted operation class.

Profile-based footprint: 94.63% observed module signal (6,393/6,756 usable profiles; observed gate: bind8).

Recorded conditions: BIND delegate has apply=2 (zone-only) or apply=3 (configuration-only), without whole-server lifecycle permission. [nondefault]; The BIND service is configured so a start or stop request affects DNS availability. [environmental]

Audited source: `bind8/acl_security.pl`, `bind8/start.cgi`.

WEBMIN-0191 — DNSSEC permission bypass on direct zone signing

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The BIND module has a dedicated dnssec permission, and normal DNSSEC key-management routes enforce it. The direct sign_zone.cgi action checks only zone edit access before calling the signing routine. In the frozen 2.670 lab, dnssec=0 blocked the control page while the direct action reached signing and changed the zone file before failing on the intentionally unavailable signer.

Profile-based footprint: 94.63% observed module signal (6,393/6,756 usable profiles; observed gate: bind8).

Recorded conditions: BIND delegate can edit a zone but is denied the separate DNSSEC permission. [nondefault]

Audited source: `bind8/sign_zone.cgi`.

WEBMIN-0429 — Zone conversion bypasses per-zone and read-only ACLs

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

convert_master.cgi and convert_slave.cgi accept a requested zone and mutate its type/directives after checking only the broad destination-type capability. Neither handler enforces can_edit_zone or the read-only ACL. Frozen-stable execution changed an excluded zone from master to slave while the caller's module ACL allowed only allowed.test and xss.test and set ro=1.

Profile-based footprint: 94.63% observed module signal (6,393/6,756 usable profiles; observed gate: bind8).

Recorded conditions: BIND delegate has destination-type permission but is denied the selected zone or is read-only. [nondefault]

Module · Change User (1)

WEBMIN-2458 — Change Language and Theme locale ACL deny is ineffective

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The Change Language and Theme ACL form offers an administrator a locale yes/no control, and the module's mutation handler consults that permission. However, the ACL save callback copies only lang, theme, and pass. Since defaultacl supplies locale=1, submitting locale=0 produces no stored override and effective ACL loading restores allow. The affected user can therefore continue changing only that user's locale or legacy date format contrary to the administrator's selected policy.

Profile-based footprint: 5.34% observed module signal (361/6,756 usable profiles; observed gate: change-user).

Recorded conditions: Change Language and Theme delegate is explicitly denied locale changes. [nondefault]

Module · Cluster Useradmin (1)

WEBMIN-0567 — Cluster-useradmin feature switches hide controls without enforcing authorization

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Cluster Users and Groups exposes configuration switches labeled as allowing user/group creation, user/group search, refresh, and synchronization, but the implementation consults them only when rendering index-page controls. The corresponding CGI handlers do not enforce the switches. The related conf_host_links setting likewise changes rendered links without protecting edit_host.cgi. Consequently, an authenticated user with module access can directly request disabled functionality. The frozen 2.670 lab reproduced direct access to the actual create forms, search handlers, synchronization form, and refresh handler after all seven options were set to zero; no mutating form was submitted during the defensive test.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: Cluster Users and Groups feature switch denies an operation while the delegate retains module access. [nondefault]

Audited source: `cluster-useradmin/index.cgi`.

Module · Cron (1)

WEBMIN-0065 — Hourly scheduling policy is bypassable by a crafted save request

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The hourly-only module and per-user ACL settings constrain only the rendered minute controls. The server-side parse_times_input routine accepts all_mins or arbitrary minute arrays without checking hourly_only. An authenticated Cron-module user who may create or edit a permitted target user's job can therefore schedule an authorized command every minute or at multiple minutes per hour, bypassing the administrator's frequency policy and potentially causing resource exhaustion or more persistent execution.

Profile-based footprint: 96.86% observed module signal (6,544/6,756 usable profiles; observed gate: cron).

Recorded conditions: Cron delegate is subject to the hourly-only scheduling restriction. [nondefault]

Audited source: `cron/cron-lib.pl`.

Module · Custom (1)

WEBMIN-2477 — Safe-mode ACL authorizes every configured custom command

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Custom Commands safe-mode ACL disables editing but sets cmds to a wildcard. When a user's module ACL carries the safe marker, Webmin loads this template, and can_run_command treats the wildcard as authorization for every configured command. Assigning the module therefore exposes current and future custom commands—including commands configured for privileged accounts—rather than a least-privilege subset. Administrator assignment of the module remains a required precondition.

Profile-based footprint: 5.52% observed module signal (373/6,756 usable profiles; observed gate: custom).

Recorded conditions: Custom Commands is assigned to a user through the safe-mode ACL profile. [nondefault]; At least one configured command runs with privileges beyond that user. [environmental]

Audited source: `custom/safeacl`, `web-lib-funcs.pl`.

Module · DHCP (2)

WEBMIN-0129 — Alternate MAC encoding bypasses DHCP host uniqueness policy

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

save_host.cgi checks a submitted hardware address for a duplicate by comparing its raw text with stored values, and only afterward normalizes dash-delimited or unformatted addresses into colon notation. Because multiple accepted spellings identify the same MAC address, a delegated host creator can submit an alternate spelling that passes uniq_hst and is then persisted identically to an existing reservation. The newest-stable run with host-create permission and uniqueness enforcement enabled saved two 00:11:22:33:44:55 entries. The demonstrated impact is DHCP reservation integrity and potential availability loss, not privilege escalation or command execution.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: DHCP host-MAC uniqueness enforcement is enabled. [nondefault]

Audited source: `dhcpd/save_host.cgi`.

WEBMIN-0529 — Delegated DHCP array-option editing permits root-owned configuration directive injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

In save_options.cgi, a custom option whose definition begins with array follows a branch that performs no validation or quoting of cval_N. The value is handed to directive_lines, which concatenates unquoted values verbatim. An authenticated DHCP-module user who is meant to edit only an allowed subnet, host, group, or shared network can include a semicolon and additional syntax so Webmin writes extra statements into the root-owned DHCP configuration. The frozen 2.670 runtime reproduced this with a subnet-only account lacking global and apply privileges: cval_0 set to `1; authoritative` produced `option lab-array 1; authoritative;` in a UID-0-owned dhcpd.conf. This establishes persistent arbitrary DHCP directive injection and a delegated authorization-boundary bypass. The test did not execute a DHCP daemon command or establish direct OS-root command execution.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `dhcpd/save_options.cgi`, `dhcpd/dhcpd-lib.pl`.

Module · Fdisk (1)

WEBMIN-0779 — fdisk mutation endpoint does not enforce partition safety and backend policy

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The fdisk edit pages suppress mutation controls when a partition is mounted or when a GPT partition is represented as read-only because parted is unavailable or module mode is fdisk. The authoritative save_part.cgi endpoint rechecks only the module user's disk selection. It does not re-evaluate device_status, require the editable marker value, or enforce the configured GPT backend before invoking delete_partition or change_type. An authenticated user who already has fdisk edit access to the selected disk can bypass these intended state/backend constraints with a crafted direct request, creating a credible integrity and availability risk. The caller must already have module edit permission for the selected disk.

Profile-based footprint: 95.60% observed module signal (6,459/6,756 usable profiles; observed gate: fdisk).

Recorded conditions: Fdisk editor targets a mounted partition or a GPT partition whose normal UI disables mutation. [environmental]; The delegate still has fdisk edit permission for the selected disk. [nondefault]

Audited source: `fdisk/edit_part.cgi`, `fdisk/edit_disk.cgi`.

Module · Filemin (3)

WEBMIN-0526 — Direct file-edit requests bypass the per-user MIME edit ACL

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Filemin exposes allowed_for_edit as a per-user ACL describing MIME types permitted for editing, but the value is used only to decide whether the directory listing shows an edit link. The edit and save endpoints enforce path containment and filesystem permissions but do not compare the target MIME type with allowed_for_edit. A user can directly invoke those endpoints to read or modify an excluded file inside an otherwise allowed directory. Frozen 2.670 source retains this split, and an isolated authenticated latest-stable run reproduced it: the UI omitted the edit action for blocked.bin under an application-json-only ACL, while a direct save_file.cgi POST changed the binary file from BEFORE-B01009 to AFTER-B01009.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: Filemin delegate has a MIME edit allowlist that excludes an otherwise accessible file. [nondefault]

Audited source: `filemin/acl_security.pl`, `filemin/filemin-lib.pl`.

WEBMIN-0876 — Direct editor endpoints bypass the MIME editing allowlist

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The File Manager ACL labels allowed_for_edit as the MIME types permitted for the built-in editor. The listing code consults that value when deciding whether to render an edit action, but edit_file.cgi and save_file.cgi enforce only the allowed path and underlying file permissions. A user can request those handlers directly for a disallowed MIME type and read or replace the file. The controlled 2.670 reproduction configured text/plain as the sole allowed edit type, confirmed that an image/png target had no edit action, then read its marker and overwrote it through the direct handlers.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: Filemin MIME edit allowlist excludes a file within the delegate's allowed path. [non-default]

Audited source: `filemin/acl_security.pl`, `filemin/filemin-lib.pl`.

WEBMIN-0886 — Strict ownership ACL permits writable files owned by other users

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The strict File Manager ACL is described as allowing deletion and movement only for files owned by the selected Unix user. Its gate uses Perl's -w predicate instead of comparing file ownership, so group-writable or ACL-writable files owned by another UID pass. The frozen 2.670 container demonstrated UID 1001 passing the gate and renaming a file owned by UID 1002.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: Filemin delegate uses strict ownership mode for a selected Unix account. [nondefault]; A different-owned file is writable by that account through group or filesystem ACL permissions. [environmental]

Audited source: `filemin/filemin-lib.pl`, `filemin/lang/en`.

Module · Filter (1)

WEBMIN-0619 — Usermin mail-filter condition injection enables same-account command execution

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:N/SI:N/SA:N`.

The simple header-condition save path does not reject carriage returns or line feeds. When regex mode is disabled it only applies quotemeta if a fixed regular-expression metacharacter set is present; line breaks, backticks, and equals signs are outside that set. The downstream Procmail recipe serializer concatenates the condition directly and preserves embedded line breaks. An authenticated Usermin Filter module user can therefore inject a new Procmail action or variable assignment with command substitution, then trigger it with matching mail. Commands run as the corresponding Unix account because the module switches to the remote user before loading and writing filter data. This bypasses module-only restrictions but is not direct root execution.

Profile-based footprint: 87.39% observed module signal (5,904/6,756 usable profiles; observed gate: filter).

Recorded conditions: A Usermin Filter user can save a line-break-bearing condition and a matching message is delivered. [environmental]

Audited source: `filter/save.cgi`, `procmail/procmail-lib.pl`.

Module · Firewall (4)

WEBMIN-0217 — Restricted firewall user can move rules with disallowed actions

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The move-selected dispatcher performs table-level authorization but does not call can_jump for the indexed rules it relocates. A restricted firewall user can submit a rule index that the UI withholds and move a rule carrying a disallowed target action to another chain. Saved-file mode silently changes the policy an administrator may later apply; direct mode writes the changed table through iptables-restore immediately. The latest-stable lab

safely used an inert save file and confirmed that a user limited to jumps=DROP moved an ACCEPT rule into INPUT.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall delegate may move rules but has a jump-target restriction excluding the selected rule. [nondefault]

Audited source: `firewall/save_policy.cgi`, `firewall/firewall-lib.pl`.

WEBMIN-0573 — Filtered-chain protection is bypassable through direct mutation endpoints

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The firewall index treats filter_chain matches as externally managed: it hides matching chains and suppresses controls for rules that jump to them. The mutation handlers do not repeat that policy. A delegated firewall user who has table and chain-operation ACLs can call save_policy.cgi or a rule handler directly with the hidden chain name and change or remove its rules. This can defeat an administrator's separation between Webmin-managed rules and protections managed by fail2ban or another firewall tool.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall filter_chain protects a chain while the delegate retains table and chain mutation rights. [nondefault]

Audited source: `firewall/index.cgi`, `firewall/save_policy.cgi`.

WEBMIN-0574 — Caller-controlled IP family selection crosses firewall module ACL boundaries

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Privileged firewall handlers derive the target family from the version request parameter but derive permissions from the CGI's current module name. A request to the firewall module with version=inet6 therefore loads IPv6 operations and data while retaining the firewall module ACL. A user assigned only firewall, with firewall6 unavailable or more restrictive, can use allowed firewall actions against IPv6 rules.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall delegate can edit IPv4 rules but lacks comparable IPv6 firewall access. [nondefault]

Audited source: `firewall/firewall-lib.pl`, `firewall/save_policy.cgi`.

WEBMIN-0850 — IPv6 firewall address validation fails open

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The IPv6 firewall library's check_ipmask routine calculates whether each comma-separated item is acceptable, but its invalid-item branch returns 1 and its completion path also returns 1. save_rule6.cgi uses that return value as the validation gate for source and destination fields, then stores the normalized value. An authenticated Firewall-module user with rule-edit permission can therefore submit malformed address text that is accepted into rule state. A malformed saved rule can prevent the intended IPv6 firewall configuration from applying under an already privileged module account.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall delegate can save an IPv6 rule with an invalid source or destination address. [environmental]

Audited source: `firewall/firewall6-lib.pl`, `firewall/save_rule6.cgi`.

Module · Firewall6 (4)

WEBMIN-0540 — Jump-target ACL bypass through custom rule arguments

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

A delegated firewall editor with a restricted jumps ACL can submit no parsed jump and place a long-form --jump target in the raw custom-arguments field. can_jump() authorizes the empty parsed target, and rule serialization appends the unchecked custom argument after known options. This permits firewall actions outside the administrator-defined target allowlist.

Profile-based footprint: 83.53% observed module signal (5,643/6,756 usable profiles; observed gate: firewall6).

Recorded conditions: The boundary requires a configured jump-target restriction. [nondefault]

Audited source: `firewall/save_rule6.cgi`, `firewall/firewall-lib.pl`.

WEBMIN-0541 — Bulk rule delete and move bypass per-rule jump authorization

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The bulk delete and move branches trust submitted rule indexes and do not call can_jump() for each affected object. Although the rule list hides checkboxes for disallowed targets, a crafted request can mutate those rules. This violates the module's delegated jump-target restriction and can affect firewall integrity and availability.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall delegate can bulk-delete or move rules but is denied their jump target. [nondefault]

Audited source: `firewall/save_policy.cgi`.

WEBMIN-0593 — Request parameter selects IPv6 operations under IPv4 module authorization

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Shared firewall CGI scripts authorize the module named by the request URL during init_config, then get_ipvx_version accepts a version field from the request to select IPv4 or IPv6 implementation code. A user allowed to access firewall but denied firewall6 can therefore request /firewall/index.cgi?version=inet6 and enter IPv6 behavior while get_module_acl still resolves firewall permissions. If the firewall ACL grants rule-management capabilities, that user can act on an IP-family policy outside the denied firewall6 module assignment and can weaken or replace IPv6 filtering.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall delegate is allowed IPv4 management but denied firewall6 management. [nondefault]

Audited source: `firewall/firewall-lib.pl`, `firewall/index.cgi`.

WEBMIN-0603 — IPv6 firewall operations can use the IPv4 firewall module authorization context

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Webmin authorizes access using the module name derived from the CGI path, and feature ACL loading defaults to that same active module. The shared firewall endpoints then accept a request-controlled version value; a value ending in 6 selects firewall6-lib.pl even when the request remains under /firewall/. Consequently, an authenticated user globally allowed firewall but denied firewall6 can invoke IPv6 firewall behavior while retaining the firewall module's table and action permissions. This crosses the global module-access boundary and can permit unauthorized IPv6 rule, policy, apply, or related operations when the corresponding firewall feature ACL is enabled.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall delegate is allowed IPv4 management but denied firewall6 management. [nondefault]

Audited source: `firewall/firewall-lib.pl`, `web-lib-funcs.pl`.

Module · Fsdump (2)

WEBMIN-0494 — Restore-only fsdump users can select an arbitrary program for remote execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The filesystem-backup restore endpoint authorizes access with the restore ACL, while the module separately exposes cmds and extra ACL controls. For tar restores, the rmt input is neither hidden nor rejected when those command-oriented privileges are disabled. Webmin validates it as a single non-whitespace value and escapes it for the local shell, then passes it to GNU tar as --rmt-command. GNU tar supplies that value to the remote-shell transport as the program to start on the remote host. Consequently, a restore-only Webmin user who can authenticate to a remote target can select an arbitrary existing remote executable, including a shell, under the authenticated remote account. The local shell escaping is effective against local metacharacter injection but does not enforce the intended module authorization boundary for remote program selection.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: fsdump).

Recorded conditions: FSdump delegate can restore tar archives but is denied command and extra-option controls. [nondefault]; Remote tar transport is used and the request supplies a custom rmt program. [environmental]

Audited source: `fsdump/restore.cgi`, `fsdump/linux-lib.pl`.

WEBMIN-0851 — Restricted filesystem-backup editors can execute a custom remote-shell program as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Filesystem Backup module exposes create/edit permission separately from its before/after-command and extra-argument permissions, but its remote-shell parser accepts a crafted custom selector and an arbitrary executable path without consulting either command-related ACL. The value is persisted as the backup's remote-shell command and supplied to GNU tar for a remote backup. Enabled backup jobs are created for root, so a restricted fsdump editor who can place or reference an executable can cause that executable to run with root privileges when the backup runs.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: fsdump).

Recorded conditions: An enabled remote backup job uses GNU tar and a custom remote-shell executable. [nondefault]; The restricted editor can place or reference the selected executable. [environmental]; The scheduled or manual root-run backup job executes afterward. [environmental]

Audited source: `fsdump/save_dump.cgi`, `fsdump/fsdump-lib.pl`.

Module · Init (4)

WEBMIN-0277 — Run-only Bootup ACL can write Upstart and systemd service definitions

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Bootup ACL value 2 is the module's start-only role, but both Upstart and systemd save handlers authorize any nonzero value before allowing create, replace, and delete operations on root service definitions. A limited Webmin user can place attacker-chosen commands into a service definition and start it through the permissions they already hold, converting run-only module access into root-level command execution.

Profile-based footprint: 5.45% observed module signal (368/6,756 usable profiles; observed gate: init).

Recorded conditions: The finding is specific to a non-default run-only Bootup ACL. [nondefault]

Audited source: `init/acl_security.pl`, `init/save_upstart.cgi`.

WEBMIN-0346 — Run-only init-module users can create arbitrary root systemd services

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The run-only ACL tier is intended to control existing boot actions without edit rights, as shown by save_action.cgi requiring bootup==1 for modifications. The systemd save endpoint instead accepts any nonzero bootup value and allows new services with request-controlled atstart content. create_systemd_service writes that content to ExecStart in a privileged unit. A bootup=2 user can therefore configure a service command that systemd runs as root, exceeding the run-only authorization boundary.

Profile-based footprint: 5.45% observed module signal (368/6,756 usable profiles; observed gate: init).

Recorded conditions: The finding is specific to a non-default run-only Bootup ACL. [nondefault]

Audited source: `init/acl_security.pl`, `init/save_systemd.cgi`.

WEBMIN-0495 — Run-only init ACL permits privileged launchd creation and configuration

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The init module defines bootup ACL value 2 as run-only, but edit_launchd.cgi and save_launchd.cgi authorize with a non-zero truth test rather than requiring value 1. A run-only delegated user can therefore open the launchd creation/configuration form and reach creation, raw plist overwrite, reload, and deletion branches. On a macOS launchd host these operations manage root-owned system daemons; choosing the program for a new daemon or rewriting a plist can execute attacker-selected commands as root.

Profile-based footprint: 5.45% observed module signal (368/6,756 usable profiles; observed gate: init).

Recorded conditions: The finding is specific to a non-default run-only Bootup ACL. [nondefault]

WEBMIN-0759 — Run-only init ACL can mutate root systemd units

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The init ACL defines bootup value 2 as run only, while full mutation is value 1. In frozen stable, edit_systemd.cgi and save_systemd.cgi authorize with a nonzero test, so a run-only account can open the unit editor and submit creation or replacement data. The save path passes a supplied start command to create_systemd_service, which writes a root-owned unit ExecStart and reloads systemd. This crosses the documented ACL boundary and can convert delegated service-control access into persistent root command execution.

Profile-based footprint: 5.45% observed module signal (368/6,756 usable profiles; observed gate: init).

Recorded conditions: The finding is specific to a non-default run-only Bootup ACL. [nondefault]

Audited source: `init/mass_systemd.cgi`.

Module · Iscsi Client (1)

WEBMIN-0356 — Prefix-only numeric validation permits Open-iSCSI configuration directive injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The save handler for login retry count, maximum queued commands, and device queue depth validates only a leading positive integer with /^[1-9]\d*/. It then sends the complete request value to save_directive, which concatenates it into an iscsid.conf line. A crafted value containing a valid numeric prefix and newline therefore creates an additional Open-iSCSI directive, allowing a module user to bypass the form's intended field restrictions and alter authentication, startup, or other client behavior.

Profile-based footprint: 4.11% observed module signal (278/6,756 usable profiles; observed gate: iscsi-client).

Recorded conditions: iSCSI Client delegate can save the affected numeric settings with a numeric prefix and newline. [environmental]

Audited source: iscsi-client/save_iscsi.cgi, iscsi-client/iscsi-client-lib.pl.

Module · LDAP Client (1)

WEBMIN-0493 — NSS LDAP filter persistence fails open

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N.

When saving a per-database filter for the legacy NSS LDAP client, save_base.cgi appends the filter text to $file instead of $base. The script subsequently persists only $base as nss_base_<database>, so the selected LDAP restriction is discarded while the operation reports success. On hosts where NSS/PAM or another local service relies on that filter to exclude directory identities, the resulting lookup scope can fail open and expose accounts outside the intended set.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: ldap-client).

Recorded conditions: The legacy NSS LDAP client uses a per-database filter relied on by local identity lookups. [environmental]

Audited source: ldap-client/save_base.cgi.

Module · LDAP Users and Groups (1)

WEBMIN-0784 — LDAP user creation can create groups despite denied gcreate permission

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

In the LDAP Useradmin new-user workflow, save_user.cgi requires ucreate but does not require gcreate when a nonzero gidmode asks it to create a primary group. The same request validates or derives a group name, allocates a GID, and invokes the LDAP add operation. By contrast, the dedicated save_group.cgi endpoint explicitly requires gcreate. An authenticated operator delegated user creation but denied group creation can therefore cross that ACL boundary and create a directory group as part of a new-user request.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: LDAP Useradmin delegate has ucreate but not gcreate permission. [nondefault]; The new-user request selects primary-group creation. [environmental]

Audited source: ldap-useradmin/save_user.cgi, ldap-useradmin/save_group.cgi.

Module · Logviewer (3)

WEBMIN-0832 — Delegated Log Viewer ACL editor can configure root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

Log Viewer treats any extra-log value ending in a pipe as a shell command. The parser combines administrator-owned module extras with the current user's per-user ACL extras, while the Log Viewer ACL save routine stores the delegated ACL editor's extras field without command validation or a distinct command-execution permission. A delegated ACL administrator who may edit the Log Viewer ACL for a target can therefore place an arbitrary command in that ACL. When the target opens the resulting extra-log entry, view_log.cgi selects the command and passes its shell pipeline to proc::safe_process_exec with UID and GID set to zero. This converts limited ACL-management authority into root command execution; exploitation requires an editable account the attacker controls or interaction by the target account.

Profile-based footprint: 85.24% observed module signal (5,759/6,756 usable profiles; observed gate: logviewer).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: logviewer/acl_security.pl, logviewer/logviewer-lib.pl.

WEBMIN-0859 — Direct progressive endpoint bypasses the syslog viewing ACL

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:H/VA:N/SC:N/SI:N/SA:N.

The progressive journal CGI validates the selected generated journal command with can_edit_log but does not enforce the separate syslog module ACL used by the ordinary viewer. Because can_edit_log defaults to true when no granular logs list is configured, a user granted the module but denied syslog access can call the progressive endpoint directly and receive journal output. The frozen latest-stable Docker reproduction confirmed that the ordinary endpoint denied the restricted account while the progressive endpoint returned the controlled journal marker.

Profile-based footprint: 85.24% observed module signal (5,759/6,756 usable profiles; observed gate: logviewer).

Recorded conditions: Log Viewer delegate is denied syslog viewing while permitted to use the module. [nondefault]; A generated journal command is available to the progressive endpoint. [environmental]

Audited source: logviewer/view_log_progress.cgi, logviewer/view_log.cgi.

WEBMIN-2453 — Default module delegation exposes arbitrary root-readable files

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

The ordinary logviewer ACL defaults enable the explicit arbitrary-file route with any=1 and omit a logs allowlist. can_edit_log treats an absent allowlist as unrestricted, and Webmin executes the CGI with host-level file access. A non-master Webmin account granted only logviewer under these defaults can therefore disclose arbitrary root-readable files. This is recorded as a security observation about an unsafe delegation default, not as an ACL check bypass.

Profile-based footprint: 85.24% observed module signal (5,759/6,756 usable profiles; observed gate: logviewer).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: logviewer/defaultacl, logviewer/logviewer-lib.pl.

Module · Mailboxes (1)

WEBMIN-0823 — HTML mail remote-resource controls can be bypassed

CVSS 4.0 Base (CVSS-B): 6.9 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N.

The HTML-mail remote-resource control has four independently observed bypass variants. Historical versions used a greedy prefix and case-sensitive IMG matcher, allowing earlier images in a multi-image body and uppercase or mixed-case tags to remain active; those two parser variants are fixed in Webmin 2.670. Webmin 2.670 still stops scanning when the first matched image is a data URI and does not inspect CSS url() or other non-img resource declarations. Those current variants can cause a viewer's browser to contact a sender-controlled endpoint, disclosing message-open timing and client network metadata despite the selected privacy mode.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: An authenticated user views HTML mail with remote images after a data URI or in an unhandled CSS context. [environmental]

Audited source: mailboxes/folders-lib.pl.

Module · Minecraft (1)

WEBMIN-0767 — Minecraft-only Webmin user can upload and launch an arbitrary JAR as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Minecraft module accepts an authenticated multipart JAR upload, checks only whether the file utility labels it ZIP/JAR, copies it into minecraft_dir, and makes it mode 0755. The same module lets that Webmin principal select the uploaded filename and start it using java -jar. No module-specific ACL separates version upload or server start from general Minecraft-module access. The shipped configuration sets unix_user=root, and the command builder applies a user switch only when unix_user is not root. Therefore a user delegated only this module can supply Java bytecode and have Webmin execute it as UID 0. The frozen 2.670 Docker run reproduced the chain with a Minecraft-only account: the uploaded JAR was root-owned mode 0755, the Java process ran as root, and payload code recorded uid=0.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: The host has a Java runtime and a usable Minecraft server start path for the uploaded JAR. [environmental]

Module · Mount (4)

WEBMIN-0222 — Delegated fsck-order editing injects arbitrary records into root-owned fstab

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The save path accepts the HTTP `order` value without verifying that it is a single integer. The order-only comparison added by fb56744e invokes `change_mount`, and the Linux serializer writes that value verbatim as the final fstab field. A newline therefore terminates the legitimate record and appends an attacker-chosen record to root-owned fstab. This bypasses delegated Mount ACL intent and can yield attacker-selected mounts when privileged tooling consumes fstab, as well as boot or service denial of service. The exact 2.670 `change_mount` implementation reproduced the additional record in the serialized latest-stable-only lab.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: The host uses the Linux fstab serializer for mount records. [environmental]; Privileged mount or boot tooling later consumes the injected fstab entry. [environmental]

Audited source: `mount/save_mount.cgi`, `mount/linux-lib.pl`.

WEBMIN-0483 — Delegated NFS option editing permits root fstab record injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

On Linux, check_options validates NFS timeo but copies retrans, port, wsize, and rsize directly from the request into the options map. The map is joined without removing control characters and create_mount writes it into the root-owned fstab. A newline therefore terminates the authorized mount's options field and begins a second attacker-chosen fstab record, bypassing Mount-module filesystem-path restrictions. Processing that record as root can mount an attacker-controlled NFS export over a privileged directory; a target such as /etc/cron.d can turn the primitive into conditional root command execution when the mounted content is consumed.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: The host uses the Linux NFS and fstab mount path. [environmental]; The attacker can provide the referenced NFS export for the escalation example. [environmental]; A later privileged mount action processes the injected entry. [environmental]

Audited source: `mount/linux-lib.pl`.

WEBMIN-0536 — Delegated NFS option editing permits arbitrary root-owned fstab entry injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The NFS option handler treats nfs_auth and nfs_sec as though POST requests must contain one of the radio-button values shown by the UI, but it never validates those values server-side. A delegated mount-module user can place a newline and a complete fstab record in nfs_auth. check_options stores that text in the sec option and create_mount writes it verbatim to the root-owned fstab. This bypasses delegated filesystem type and mount-point restrictions by adding an independent attacker-chosen mount entry. A later privileged fstab consumer can apply attacker-selected filesystem sources and options, including an attacker-controlled NFS source with suid and exec semantics. The Docker reproduction establishes arbitrary root-owned fstab injection, not immediate OS command execution.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mount/linux-lib.pl`.

WEBMIN-0803 — Delegated NFS mount input can inject unauthorized fstab entries

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The NFS check_location routine performs hostname and export-directory validation only inside the show-mount/nfs_check branch, then returns the request-derived host and directory as a device string. Webmin's form parser decodes percent-encoded newlines and tabs without removing them. The permanent-mount writer replaces spaces with \040 but writes tabs and newlines unchanged into the configured fstab file. Although save_mount.cgi applies the delegated user's directory and filesystem-type ACL to the declared mount, injected lines are not covered by that check. A permitted permanent NFS mount can therefore append another fstab entry targeting a path and filesystem type outside the delegated authorization boundary.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: Mount delegate can save a permanent NFS entry with newline or tab characters in host or export text. [environmental]

Audited source: `mount/linux-lib.pl`, `web-lib-funcs.pl`.

Module · MySQL (3)

WEBMIN-0498 — Restricted MySQL module users can access manual configuration read and write endpoints

CVSS 4.0 Base (CVSS-B): 8.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`.

The manual editor endpoints require only general MySQL module access. They do not enforce the unrestricted perms flag before enumerating parsed MySQL configuration files, reading the selected file, or replacing it with attacker-supplied content through Webmin's privileged tempfile writer. A restricted module user can call the CGI endpoints directly, bypassing UI visibility and gaining root-mediated read/write access to database service configuration, with potential database-service compromise after configuration activation.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: A Webmin user has MySQL module access with perms=0 and is denied unrestricted MySQL administration. [nondefault]; A parsed MySQL configuration file is available to the manual editor and Webmin can write it through its privileged tempfile path. [environmental]

Audited source: `mysql/edit_manual.cgi`, `mysql/save_manual.cgi`.

WEBMIN-0714 — Delegated MySQL module account can force-reset a database administrator password

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Webmin's MySQL login handler permits a request when access.user is set, even when access.noconfig denies database-login configuration. The same handler accepts an editable account name and, when force=1 is submitted, calls force_set_mysql_admin_pass. That helper exercises host-level service control: it stops the local database, starts a safe wrapper with --skip-grant-tables, changes the submitted account password, shuts down the temporary server, and starts the service normally. A Webmin user delegated a limited database identity can therefore cross that boundary, select an administrator account such as root, replace its password, and obtain database-administrator control while also causing a service interruption.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mysql/login.cgi`, `mysql/acl_security.pl`.

WEBMIN-0737 — Custom mysqldump options bypass backup destination and writer controls

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The MySQL backup CGI validates the normal destination against the delegated user's bpath and sends output through a writer that may be demoted with command_as_user, but it separately accepts arbitrary whitespace-delimited mysqldump options. backup_database shell-escapes each token yet executes mysqldump itself on the root side. Output-affecting client options such as --result-file therefore operate outside the validated destination and outside the lower-privileged writer. The isolated latest-stable harness used the real backup_database construction and a command-compatible deterministic dump helper: the helper ran as UID 0, received --result-file=/tmp/b01469-outside-bpath.sql, created that file, and simultaneously piped the ordinary destination through UID 65534.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL backup delegate has a destination restriction or demoted writer but can add custom mysqldump options. [nondefault]

Module · Net (1)

WEBMIN-0442 — Delete-disabled Networking user can remove interface configuration

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Networking module defines a separate delete ACL and hides deletion controls when it is false, but delete_bifcs.cgi never checks that ACL. It only checks whether the caller may edit the selected interface, then deletes its root-owned boot configuration; Delete-and-Apply can additionally deactivate the live interface. Direct POST submission therefore bypasses an administrator's explicit no-delete policy and can cause a persistent host network outage.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: Networking delegate may edit an interface but has deletion disabled. [nondefault]

Audited source: `net/acl_security.pl`, `net/delete_bifcs.cgi`.

Module · PHP Configuration (2)

WEBMIN-0173 — Delegated PHP Configuration access can bypass its Unix-user restriction and execute as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

An authenticated Webmin user delegated the PHP Configuration module can bypass its configured non-root Unix-user file-I/O restriction when anyfile and manual editing are enabled. The stable manual endpoint accepts the attacker-selected path, and should_switch_user deliberately returns false for files under /etc, causing write_file_contents_as_user to write with Webmin's root effective UID. In the isolated 2.670 image, an ACL user of nobody authorized /etc/ld.so.preload, created it as root, and caused a nobody-owned shared object to run with effective UID 0 in the next root process. The default ACL enables anyfile and manual editing, so an administrator who changes only user from root to an unprivileged account retains the vulnerable combination. The path requires authenticated delegated access.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: The impact depends on a configured PHP Unix-user restriction. [nondefault]

Audited source: `phpini/save_manual.cgi`, `phpini/phpini-lib.pl`.

WEBMIN-0809 — PHP configuration Unix-user restriction is bypassed for /etc targets

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The PHP Configuration module has a distinct ACL setting for the Unix user under which file reads and writes are meant to occur. For a target under /etc, should_switch_user returns false, causing whole-file writes and cached-line flushes to run in Webmin's root process instead. An authenticated module user must still pass the PHP-file authorization check, but can modify an allowed root-protected PHP configuration even when the configured non-root Unix user is denied by the filesystem. This is a privilege-boundary violation with scope limited to PHP files authorized by the module ACL.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: The finding depends on a configured PHP Unix-user restriction. [nondefault]

Audited source: `phpini/phpini-lib.pl`.

Module · Postfix (6)

WEBMIN-0013 — Postfix map chooser bypasses the allowed-directory ACL for backend configuration writes

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Postfix map chooser save action does not enforce a page-specific ACL or the module's allowed map directory before writing MySQL or LDAP backend configuration. It parses the request-supplied map value, accepts an absolute path for an existing backend map, passes that path to save_backend_config, and flushes it as the root-running Webmin process. A Postfix-module user can therefore create or modify an exact path outside the account's configured map directory. Several accepted backend values permit embedded newlines, giving control over additional file lines. The action requires authenticated Postfix-module access, but it crosses the intended per-user filesystem boundary and can lead to privileged configuration corruption or root-level execution when a security-sensitive writable target is selected.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: An existing Postfix MySQL or LDAP backend map is selected for the save action. [environmental]; The caller can select an accepted absolute path outside the configured map allowlist. [environmental]

Audited source: `postfix/map_chooser_save.cgi`, `postfix/postfix-lib.pl`.

WEBMIN-0205 — Restricted Postfix user can add mappings outside the delegated directory

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

For a multi-map parameter, save_map.cgi authorizes an addition by checking only the last configured map path. create_mapping independently prefers the last map whose type is not regexp or pcre. If an earlier hash map is outside the user's allowed directory and the final regexp map is inside it, a direct authenticated request passes the directory check but writes the new entry to the outside map as the Webmin root process. This crosses the module's delegated filesystem boundary and can alter mail routing or access policy.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: Postfix delegate has a directory restriction and can edit a multi-map parameter with an out-of-scope earlier map. [nondefault]

Audited source: `postfix/save_map.cgi`, `postfix/postfix-lib.pl`.

WEBMIN-0314 — Postfix general-settings ACL bypass permits arbitrary root-owned main.cf changes

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Postfix general-options form rejects users whose general ACL is false, but its save_opts_misc.cgi target omits the same check. The endpoint passes the entire parsed request to save_options, which treats any key ending in _def as a Postfix setting name and invokes set_current_value. Consequently, an authenticated user delegated other Postfix access can directly submit an arbitrary parameter/value pair and make the root Webmin process alter main.cf and attempt a reload despite general=0. The demonstrated impact affects configuration integrity and service availability; direct root shell access or general command execution was not shown.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: The boundary requires a Postfix role denied the general-settings page. [nondefault]

Audited source: `postfix/general.cgi`, `postfix/save_opts_misc.cgi`.

WEBMIN-0330 — Sender-dependent save endpoint bypasses Postfix page ACL boundaries

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The sender-dependent transport-map save endpoint authorizes only the separately delegable dependent capability, then passes the full request map to save_options. That helper treats every key ending in _def as a Postfix parameter selector and writes the derived parameter without restricting it to sender_dependent_default_transport_maps. Consequently, an authenticated user delegated only this page can modify arbitrary recognized Postfix main.cf settings governed by other page permissions. A harmless latest-stable reproduction used smtpd_banner to demonstrate the boundary violation; more security-sensitive routing, relay, and copy parameters are exposed through the same generic write primitive.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: Postfix delegate has sender-dependent-map permission but is denied the page governing the injected setting. [nondefault]

Audited source: `postfix/save_opts_dependent.cgi`, `postfix/postfix-lib.pl`.

WEBMIN-0653 — Substituted Postfix map target bypasses the delegated directory ACL

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

save_map.cgi builds the mapping cache through helpers that resolve Postfix parameter substitutions, but its add-path directory check parses get_current_value(map_name), the unresolved configuration string, and authorizes only the last path visible there. create_mapping later calls get_real_value(map_name), selects the last non-regexp resolved map, and flushes that file. For a value such as an allowed literal map followed by $outside_maps,

a direct authenticated save request passes the delegated-directory check and appends to the out-of-scope map with Webmin's privileges. The frozen 2.670 lab reproduced the mismatch with harmless files while running as UID 0.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: The finding depends on a configured Postfix map-directory restriction. [nondefault]

Audited source: `postfix/save_map.cgi`, `postfix/postfix-lib.pl`.

WEBMIN-0687 — Generic map endpoints omit SNI feature authorization

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The SNI page enforces access{sni}, but its map-content workflow uses generic edit, save, and delete handlers that accept map_name and do not enforce the corresponding page capability. Directory checks constrain which existing map file can be touched but do not restore the missing feature authorization. A Postfix module user explicitly denied SNI access can therefore send a direct shared-handler request and alter an existing TLS certificate map within the allowed directory.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: Postfix delegate is denied SNI page access but can edit an existing map in an allowed directory. [nondefault]

Audited source: `postfix/save_map.cgi`.

Module · PostgreSQL (1)

WEBMIN-0983 — CREATE DATABASE optional attributes permit authenticated SQL injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

The frozen 2.670 newdb.cgi quotes the database name but concatenates optional path, owner, encoding, and template request values into CREATE DATABASE. CREATE DATABASE bypasses the DBI execution branch and is delivered as a complete string to psql -c. A Webmin principal delegated database creation can terminate the encoding literal and append another statement that runs with the configured PostgreSQL administration identity.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: PostgreSQL delegate can create databases and controls an optional path, owner, encoding, or template field. [environmental]

Audited source: `postgresql/newdb.cgi`, `postgresql/postgresql-lib.pl`.

Module · Project Meta (1)

WEBMIN-0995 — Privileged GitHub Actions dependencies use mutable references

CVSS 4.0 Base (CVSS-B): 9.2 (Critical) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

GitHub Actions configuration in this episode delegates execution to mutable branch or major-version references. The package caller supplies API, SMTP, SSH, and signing-related secrets to webmin/webmin-ci-cd's master-workflow.yml@main; the inactive-cleanup caller gives issue and pull-request write permissions to close-inactive.yml@main; and the test job executes actions/checkout@v4. A compromise or malicious change in one of those referenced supply chains can change the code executed by a later workflow run without a corresponding immutable-reference change in this repository. The dedicated pull-request review caller examined in the episode is absent from frozen 2.670, but the package, cleanup, and test instances remain.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: A privileged GitHub Actions workflow invokes mutable dependencies while repository or signing secrets are available. [environmental]

Audited source: `.github/workflows/webmin.dev+webmin.yml`, `.github/workflows/close-inactive.yml`.

Module · Quota (1)

WEBMIN-0915 — Restricted quota roles can retarget hash-prefixed XFS identities

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The quota save handlers authorize a request-provided user or group name before the Linux XFS setter removes a leading hash. Deny-list ACL modes compare exact raw strings, so an excluded identity such as root is denied while #root is allowed. The XFS setter then removes the hash and constructs a privileged xfs_quota limit operation for root. The same mismatch applies to users and groups. This is not shell-command injection: the command arguments remain escaped; the security consequence is unauthorized quota modification for an identity outside the delegated operator's intended scope.

Profile-based footprint: 5.43% observed module signal (367/6,756 usable profiles; observed gate: quota).

Recorded conditions: Quota delegate uses an exact deny-list for a user or group on XFS. [nondefault]; The request prefixes that denied name with #. [environmental]

Audited source: `quota/quota-lib.pl`, `quota/linux-lib.pl`.

Module · Samba (1)

WEBMIN-2455 — Share-level Unix-extensions setting is ineffective because Samba treats it as global-only

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:P/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

Webmin's Samba file-permissions page renders and saves the Unix-extensions boolean for both the global section and ordinary shares. Samba classifies this setting as global-only. When an operator selects a different value for an ordinary share, Webmin writes the line under that share, but Samba warns about the global parameter in a service section, ignores the share-level line, and keeps the global value effective. This can mislead an authorized operator into believing a share-specific restriction on SMB1/CIFS Unix features is active.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: An operator saves Unix extensions in an ordinary Samba share rather than the global section. [environmental]

Module · Servers (1)

WEBMIN-0437 — Delegated Servers editor can delete unauthorized root-owned .serv files

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The confirmed multi-server deletion endpoint checks only the module-wide edit bit, then sends every request-supplied ID to delete_server. It never verifies that each ID is in the delegate's allowed server set, and delete_server interpolates traversal components before appending .serv and unlinking as root. Webmin 2.670 reproduced both deletion of a forbidden root-owned server record and deletion of a root-owned file outside the module directory via traversal. The forced .serv suffix constrains the external-file target set.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `servers/delete_servs.cgi`, `servers/servers-lib.pl`.

Consolidated reproduction records: `webmin-0306` (Servers-module edit users can traverse the server-record deletion path).

Module · Software (1)

WEBMIN-0920 — RPM package information lookup allows authenticated root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The RPM package-information function concatenates request-controlled package and optional version values into $n and inserts $n unquoted into shell command strings passed to open_execute_command. The Software Packages detail endpoint supplies the request fields directly. open_execute_command implements output mode with a shell-form pipe, so shell substitutions or separators in either field execute as the Webmin service account. The RPM calls pass safe=1, which treats them as non-mutating and bypasses Webmin's global read-only command veto. A delegated authenticated module user can therefore cross the intended read-only/module boundary and run arbitrary commands, commonly with root privileges.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: The Software Packages detail path uses the RPM package backend. [environmental]

Audited source: `software/edit_pack.cgi`, `software/rpm-lib.pl`.

Module · Spam (2)

WEBMIN-0029 — Delegated SpamAssassin module users can overwrite arbitrary root-accessible files

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The Webmin-side set_config_file authorization check constructs an allowed-path map from optional ACL file/files values but performs rejection only when that map contains an entry. The module default ACL contains no file allowlist, so a delegated account can submit any file path. Write-capable handlers then treat that path as SpamAssassin configuration; in newest-stable, save_manual.cgi derives its own selectable-file list from the attacker-selected path and writes submitted content to it. Because the Webmin process has administrative filesystem privileges, this crosses the delegated module boundary and can overwrite root-accessible files, causing system integrity loss and potential privilege escalation.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: spam).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `spam/spam-lib.pl`, `spam/defaultacl`.

WEBMIN-0798 — AWL group chooser can silently clear the effective group restriction

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The auto-whitelist group-mode control displays an awl_groups textbox but constructs its group chooser with awl_users as the destination. A group chosen through the popup therefore populates the user field. When group mode is saved, acl_security_save clears both restriction keys and retains only the empty awl_groups request value. can_edit_awl treats empty user and group scopes as unrestricted, so a user with the spam module's AWL feature can pass authorization checks for arbitrary target users instead of only members of the intended group. The latest-stable reproduction confirmed a denied control request became authorized through the guard after the modeled chooser submission.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: spam).

Recorded conditions: SpamAssassin AWL delegate is restricted to a group and uses the group chooser while saving group mode. [nondefault]

Audited source: `spam/acl_security.pl`, `web-lib-funcs.pl`.

Module · Squid (2)

WEBMIN-0103 — Delegated header-rule administration can overwrite out-of-scope Squid access policy

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Squid header-access save and move handlers enforce the separately delegable headeracc permission but accept the directive family from request data without an allowlist. A headeracc-only user can therefore select an unrelated directive family and cause save_directive to persist changes outside the delegated section. The concrete demonstration selects http_access and changes a deny-all rule to allow-all while actrl is withheld.

Profile-based footprint: 85.82% observed module signal (5,798/6,756 usable profiles; observed gate: squid).

Recorded conditions: Squid delegate can edit header rules but is denied access-control rules. [nondefault]

Audited source: `squid/save_headeracc.cgi`, `squid/acl_security.pl`.

WEBMIN-0445 — Transparent-proxy firewall endpoint omits its dedicated ACL check

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Squid module defines a dedicated iptables access option, but the transparent-proxy edit and save endpoints do not test it. A module user denied that feature can directly invoke the save endpoint to mutate and optionally apply privileged firewall and Squid configuration state.

Profile-based footprint: 85.82% observed module signal (5,798/6,756 usable profiles; observed gate: squid).

Recorded conditions: Squid delegate can use the module but is denied its transparent-proxy iptables feature. [nondefault]

Audited source: `squid/acl_security.pl`, `squid/save_iptables.cgi`.

Module · Status (1)

WEBMIN-0646 — Read-only Status ACL permits forced privileged monitor runs

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The status module's safe ACL sets edit=0 and sched=0, presenting the module as read-only to restricted users. Its refresh.cgi endpoint reads that ACL through status-lib.pl but performs no permission check before creating the monitor wrapper and invoking monitor.pl with --force. monitor.pl treats this as a web-forced pass, updates monitoring state, and can execute administrator-configured transition commands and notification mechanisms. An authenticated safe-profile user can therefore trigger privileged operational side effects despite lacking both advertised state-changing permissions.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: Status delegate is read-only but can reach forced monitor execution. [nondefault]

Audited source: `status/safeacl`, `status/refresh.cgi`.

Module · Tcpwrappers (1)

WEBMIN-0140 — TCP Wrappers module delegation enables conditional root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

An authenticated Webmin user delegated only the TCP Wrappers module can select the documented spawn or twist directive and persist an attacker-chosen shell command into the configured hosts.allow or hosts.deny file, which Webmin writes with root privilege. When a matching libwrap-enabled service accepts a connection,

libwrap passes that command to /bin/sh in the service process context. The latest-stable reproduction used a tcpwrappers-only account to store the command and a matching root socat libwrap listener to trigger it, producing a root-owned proof file. Root impact is conditional on a reachable privileged libwrap consumer; TCP Wrappers is a legacy integration and is not a typical dependency of many current daemon defaults, but it remains realistic on older or explicitly libwrap-enabled hosts. The path requires authentication, module delegation, and a reachable privileged libwrap consumer.

Profile-based footprint: 85.11% observed module signal (5,750/6,756 usable profiles; observed gate: tcpwrappers).

Recorded conditions: A matching root-running libwrap service must be configured and reachable. [nondefault]

Audited source: `tcpwrappers/save_rule.cgi`, `tcpwrappers/edit_rule.cgi`.

Module · Time (2)

WEBMIN-0232 — Authenticated zero-step cron schedule persistently denies Webmin workers

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N`.

The shared Webmin-cron parser accepts an arbitrary schedule token supplied by the authenticated request and stores it unchanged when it is not a consecutive numeric selection. The schedule renderer recognizes */N and advances its loop by N without requiring N to be positive. A stored */0 minutes value therefore leaves the renderer in a zero-step loop. In latest stable, the scheduler also performs modulo by the same unchecked step, adding another availability failure path. The Time module exposes the parser to users granted its NTP permission while fixing the scheduled target to time::sync_time_cron, so the demonstrated impact is authenticated persistent denial of Webmin service, not arbitrary command execution.

Profile-based footprint: 5.37% observed module signal (363/6,756 usable profiles; observed gate: time).

Recorded conditions: A Time-module user with NTP scheduling permission submits or renders a */0 schedule. [environmental]

Audited source: `time/apply.cgi`, `webmincron/webmincron-lib.pl`.

WEBMIN-1006 — Hardware-clock synchronization bypasses the hardware-time module ACL

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The Time module stores permission to change system time separately from permission to change hardware time, with a true hwdate ACL value meaning hardware-time edits are denied. The action_sync_s handler nevertheless rejects a request only when both hwdate and sysdate are true. An authenticated delegated user who may change system time but may not change hardware time can submit action_sync_s directly; the handler then calls set_hardware_time_to_system_time, which runs hwclock --systohc in the privileged Webmin process. The UI hides the button from that user, but the endpoint does not enforce the independent hardware-time denial. The latest-stable container reproduced the unauthorized control-flow reachability without modifying a real clock.

Profile-based footprint: 5.37% observed module signal (363/6,756 usable profiles; observed gate: time).

Recorded conditions: The boundary requires a role denied hardware-clock synchronization. [nondefault]

Audited source: `time/apply.cgi`, `time/acl_security.pl`.

Module · Translations (1)

WEBMIN-0691 — Croatian Cron ACL labels invert permission questions

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The Croatian Cron module ACL form asks whether the delegate cannot control access, view/edit commands, create, delete, move or terminate jobs, but Yes maps to the positive permission value 1. An administrator who answers the negative question literally can grant a capability intended to be withheld. Exploitation depends on an administrator configuring a delegated Cron user in this locale; the text itself does not bypass authorization.

The reviewed changes transcode or repair these labels; the finding does not assert that they originated the wording.

Profile-based footprint: 96.86% observed module signal (6,544/6,756 usable profiles; observed gate: cron).

Recorded conditions: Administrator configures delegated Cron access through the Croatian ACL form. [non-default]

Audited source: `cron/lang/hr`, `cron/lang/en`.

Module · Tunnel (2)

WEBMIN-0203 — Configured tunnel target bypass enables authenticated SSRF and conditional credential disclosure

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The tunnel module parses the protocol, host, port, and path from authenticated request PATH_INFO. Its configured URL handling does not enforce that those parsed connection fields match the configured target: at the reviewed revision a raw string-prefix check accepts hostname extensions, and newest stable merely replaces the display URL without changing the parsed connection destination. A Webmin user authorized for the tunnel module can consequently make the Webmin server issue HTTP(S) requests to arbitrary reachable hosts and ports despite the configured target, including internal services. If the module is configured with fixed downstream credentials, the same request path sends the configured Basic Authorization value to the attacker-selected peer. The issue is not unauthenticated and does not by itself execute operating-system commands on the Webmin host; its high impact depends on delegated tunnel access and reachable sensitive services or fixed downstream credentials.

Profile-based footprint: 4.13% observed module signal (279/6,756 usable profiles; observed gate: tunnel).

Recorded conditions: An authenticated Tunnel user supplies a reachable URL that passes or bypasses the configured-target check. [environmental]; Credential disclosure additionally requires fixed or variable upstream credentials in Tunnel configuration. [nondefault]

Audited source: `tunnel/link.cgi`.

WEBMIN-0410 — HTTP Tunnel fixed-target bypass forwards configured credentials to an attacker-selected host

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The HTTP Tunnel module parses host, port, path, and protocol from request PATH_INFO before consulting the configured URL. When a fixed URL exists, the code replaces only the display-oriented openurl variable; the parsed connection tuple remains unchanged. If fixed login is enabled, Webmin then sends the configured Basic username and password to the request-selected server. An authenticated principal delegated only the Tunnel module can therefore bypass the administrator's fixed target, perform server-side requests to HTTP/HTTPS endpoints reachable from the Webmin host, and exfiltrate the configured remote credential. HTML returned by that endpoint is also served without script sanitization from the authenticated Webmin origin, increasing the impact. The finding does not claim unauthenticated access or direct OS command execution.

Profile-based footprint: 4.13% observed module signal (279/6,756 usable profiles; observed gate: tunnel).

Recorded conditions: Tunnel has a fixed target and fixed upstream login credentials. [nondefault]; An authenticated Tunnel user chooses a different reachable HTTP or HTTPS endpoint. [environmental]

Audited source: `tunnel/link.cgi`, `tunnel/tunnel-lib.pl`.

Module · Ui Demo (1)

WEBMIN-2478 — ACL-disabled read-only UI demo permits restricted shared configuration changes

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The UI Demo is presented as read-only and its own apply/save handlers do not mutate system state, but it sets $main::no_acl_check and declares noacl=1 while exposing a config.info-backed default_tab preference. Webmin's root config_save.cgi accepts noacl modules even when foreign_available is false and writes their parsed module configuration unless noconfig is separately set. Consequently, an authenticated account with only an unrelated module can change the shared UI Demo default tab. The option schema restricts the accepted value to the shared UI Demo default-tab preference.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: A restricted account can reach UI Demo through its noacl configuration but is denied ordinary shared configuration edits. [nondefault]

Audited source: `ui-demo/ui-demo-lib.pl`, `ui-demo/module.info`.

Module · Updown (2)

WEBMIN-0523 — Unauthenticated Updown TLS substitution enables conditional root RCE through root-consumed download targets

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Updown accepts a self-signed, hostname-mismatched TLS response. A network attacker can substitute content without a Webmin account; when an authorized workflow selects a root-consumed destination, the substituted bytes can execute as root.

Profile-based footprint: 5.60% observed module signal (378/6,756 usable profiles; observed gate: updown).

Recorded conditions: Updown downloads from a TLS endpoint an attacker can impersonate. [environmental]; An authorized download selects a root-consumed destination for code-execution impact. [nondefault]

Audited source: `updown/updown-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0943 — Fixed-user upload fails open to root when the Unix account is missing

CVSS 4.0 Base (CVSS-B): 6.9 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

ACL mode 3 assumes that the authenticated Webmin username resolves to a Unix account, but the passwd lookup is not checked. When no account exists, UID and GID remain undefined. switch_uid_to then skips the identity transition, leaving the root-started CGI at root effective identity for the destination write. The frozen latest-stable container reproduced this with a real configured Webmin identity named labadmin that had no Unix account: a non-direct upload to an ACL-permitted directory produced a UID 0, GID 0 file containing the submitted bytes.

Profile-based footprint: 5.60% observed module signal (378/6,756 usable profiles; observed gate: updown).

Recorded conditions: Updown upload uses fixed-user ACL mode 3. [nondefault]; The Webmin username has no matching Unix account at upload time. [environmental]

Audited source: `updown/upload.cgi`, `updown/updown-lib.pl`.

Module · Useradmin (5)

WEBMIN-0053 — Batch endpoints bypass delegated user and group controls

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

A delegated Webmin principal with batch access can reach destructive or identity-changing operations that the corresponding action ACLs deny: user and group delete branches omit udelete/gdelete, and denied user/group renames continue after printing an error. Group batch validation additionally tests user-ID fields for its upper bound, and group change hooks receive user-shaped variables. Together these defects let a batch-capable delegated administrator exceed the operation, identifier-range, and configured-hook policy intended for that account.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: User and Groups delegate has batch access but lacks one of the per-action user or group permissions. [nondefault]

Audited source: useradmin/batch_exec.cgi, useradmin/gbatch_exec.cgi.

WEBMIN-0534 — Group cloning bypasses the group-edit ACL by invoking the user checker

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

The group-clone branch loads the requested group but authorizes it with can_edit_user, whose decision is based on user-edit ACL settings and user-specific fields. Normal group editing uses can_edit_group, which is based on the independent group-edit ACL. Consequently, a delegated administrator who may create groups can obtain a clone form populated from a group that their group-edit ACL denies. The form includes the restricted group's member list and stored encrypted group password. A frozen latest-stable container reproduced the divergent decision with uedit_mode=0 and gedit_mode=1: the invoked clone guard allowed while the group guard denied.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: User and Groups delegate may create groups but may not edit the source group selected for cloning. [nondefault]

Audited source: useradmin/edit_group.cgi, useradmin/user-lib.pl.

WEBMIN-0683 — Group-batch endpoint bypasses the group-rename ACL

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

The authenticated group-batch modification endpoint checks whether the Webmin account may edit the target group. When the submitted row contains a replacement group name and the separate grename ACL is false, the endpoint prints its rename-denied message but does not stop processing. It assigns the requested name, passes through a later check that covers GID constraints rather than grename, and calls modify_group. A Webmin account with batch access and edit access to a group can therefore rename that group even though its group-rename permission is explicitly disabled. The group name is a system identity used by service and authorization configuration, so unauthorized renaming can cause integrity and availability impact.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: User and Groups delegate has group batch/edit access but is denied group renaming. [nondefault]

WEBMIN-0769 — Empty stored hash bypasses old-password verification

CVSS 4.0 Base (CVSS-B): 9.3 (Critical) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The shared password verifier first calls unix_crypt and accepts when its returned string equals the stored password hash. unix_crypt returns an empty string for a salt/hash shorter than its permitted prefix, so an empty stored hash causes any supplied password to be accepted. The anonymous password-change endpoint requires a non-empty supplied old-password parameter but delegates correctness to this verifier, creating an old-password bypass for empty-hash accounts when that endpoint is enabled. The locked frozen-stable lab reproduced the verifier acceptance and a locked-hash rejection control.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: An account has an empty stored password hash and anonymous password change is enabled. [nondefault]

Audited source: useradmin/md5-lib.pl, web-lib-funcs.pl.

WEBMIN-0815 — Direct requests can bypass the form's passwordless-account suppression

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The user editor suppresses the blank-password radio choice for a new account when empty_mode is false, but save_user.cgi accepts a direct passmode=0 submission. Its only rejection path is check_password_restrictions on an empty string; when passwd_min is zero or unset and no other password rule rejects the empty value, the handler assigns an empty password and creates the account. An authenticated Webmin principal with user-creation permission can therefore provision a passwordless local account despite the form restriction. Actual remote login exposure depends on whether a local service and PAM policy accept empty passwords.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: User and Groups configuration disallows blank passwords in the form while password restrictions still permit empty values. [nondefault]; An authenticated user-creation delegate submits passmode=0 directly. [environmental]

Audited source: `useradmin/edit_user.cgi`, `useradmin/save_user.cgi`.

Module · Usermin (7)

WEBMIN-0058 — Usermin blocked and mobile endpoints bypass per-feature ACL restrictions

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Usermin configuration module defines and consumes per-feature ACL flags to hide disallowed navigation, and neighboring endpoints enforce their flags explicitly. The blocked-user view and clear action do not check blocked, while the mobile view and save action do not check mobile. Consequently, an authenticated Webmin account with module-level Usermin access can directly invoke denied sub-features. The frozen latest-stable runtime rejected edit_session.cgi with session=0, yet returned the blocked-user page with blocked=0, delivered the clear action's restart signal, and persisted mobile_nosession=1 with mobile=0.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Configuration delegate is denied the blocked or mobile feature permission. [nondefault]

Audited source: `usermin/index.cgi`, `usermin/edit_blocked.cgi`.

WEBMIN-0296 — Usermin TLS administration bypasses the module's SSL capability restriction

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Usermin's module ACL exposes a distinct ssl capability, and guarded newkey.cgi enforces it, but the refreshed edit_ssl.cgi, change_ssl.cgi, savekey.cgi, and per-IP SSL handlers do not. An authenticated Webmin user allowed into the Usermin Configuration module but assigned ssl=0 can directly invoke these handlers. Sensitive operations include disabling TLS, changing certificate/key and cipher configuration, overwriting TLS key material, and changing per-IP certificate-chain settings. The frozen 2.670 lab reproduced a persisted ssl=1 to ssl=0 change by that restricted account while newkey.cgi returned the expected authorization error.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Configuration delegate has module access but ssl=0. [nondefault]

Audited source: `usermin/change_ssl.cgi`, `usermin/newkey.cgi`.

WEBMIN-0323 — Delegated Usermin user can bypass SSL ACL and overwrite privileged TLS key files

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Usermin module exposes SSL configuration as a distinct delegated ACL, but copycert.cgi does not check that permission before copying Webmin's private key and certificate to the paths in Usermin's miniserv configuration. The related change_ssl.cgi action likewise accepts an authenticated module user's key path without checking the SSL ACL. Consequently, a Webmin account granted only the Usermin Configuration module with ssl=0 can directly invoke these action CGIs and cause the root Webmin process to replace an existing valid PEM key/certificate path with Webmin's TLS material. A web-only attacker must select an existing readable valid key/certificate target. The resulting privileged integrity and availability impact crosses the delegated security boundary.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Configuration delegate can reach the module but has ssl=0. [nondefault]

Audited source: usermin/copycert.cgi, usermin/change_ssl.cgi.

WEBMIN-0479 — Detailed ACL bypass permits unauthorized Usermin TLS configuration changes

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Usermin Configuration module exposes a detailed ssl permission, but change_ssl.cgi does not enforce it before rewriting Usermin's MiniServ configuration. An authenticated Webmin user who is allowed into the module but explicitly denied ssl administration can directly submit the handler and change TLS enablement, key and certificate paths, protocol-disable flags, compression and cipher choices, then trigger the Usermin restart path. The frozen 2.670 source retains this mismatch, and a network-disabled latest-stable lab account configured with ssl=0 successfully changed the test Usermin configuration from ssl=0 to ssl=1.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Configuration delegate has module access but is denied the ssl capability. [nondefault]

Audited source: usermin/change_ssl.cgi, usermin/newkey.cgi.

WEBMIN-0496 — Usermin SSL handlers bypass the delegated ssl sub-ACL

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

An authenticated Webmin principal granted the Usermin Configuration module but denied its ssl subpage can access edit_ssl.cgi and submit change_ssl.cgi directly. The module index hides the SSL link when $access{'ssl'} is false, but the SSL handlers do not enforce that permission; change_ssl.cgi writes SSL enablement, certificate/key paths, cipher policy, and protocol-disable flags, then restarts Usermin. This breaks the intended delegated-administration boundary and lets the principal disable or weaken Usermin transport security, disrupt the service, or create conditions for credential and session exposure to a network observer.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Configuration delegate can use the module but is denied its ssl subpage. [nondefault]

Audited source: usermin/acl_security.pl, usermin/index.cgi.

WEBMIN-0724 — Usermin two-factor and webserver sub-ACLs are not enforced by their handlers

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

Usermin defines dedicated twofactor and web ACL values and filters the corresponding index links when those values are false, but the read and mutation handlers do not check them. A principal already admitted to the Usermin module can directly call the handlers despite denial of the narrower subfeature. The two-factor handler changes the global authentication provider; the web handler changes global server behavior such as

stderr exposure, redirect settings, caching, and directory listing. Frozen latest-stable runtime testing with both ACL values set to 0 confirmed that the index hid both links while both direct mutation requests succeeded.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Configuration delegate is denied twofactor or web capability while retaining module access. [nondefault]

Audited source: `usermin/index.cgi`, `usermin/change_twofactor.cgi`.

WEBMIN-0810 — Usermin TLS key upload bypasses delegated ssl access control

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Usermin's savekey.cgi does not enforce the module's ssl feature ACL before accepting uploaded key/certificate data, writing it to the configured Usermin TLS files, and requesting service activation. A Webmin account delegated the Usermin module but explicitly assigned ssl=0 is therefore still able to alter Usermin TLS identity or write unusable TLS material. The neighboring newkey.cgi endpoint enforces the same ACL and served as a live control. In the frozen 2.670 lab, the restricted account received 'You are not allowed to use this page' from newkey.cgi, while savekey.cgi reported that the supplied key and certificate were saved and the target file hash changed to the submitted marker-bearing content.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Configuration delegate has module access but ssl=0. [nondefault]

Audited source: `usermin/savekey.cgi`, `usermin/newkey.cgi`.

Module · Webminlog (1)

WEBMIN-0681 — Webmin Log annotations can be changed outside the caller's ACL scope

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

webminlog/rollback.cgi resolves the requested action and handles annosave by calling save_annotation, redirecting, and exiting before the rollback permission and can_user/can_mod checks. An authenticated Webmin Log principal who is restricted to their own logs or to selected modules can therefore submit an out-of-scope action ID and create, overwrite, or delete that action's annotation. The action record itself is not changed, but annotation metadata attached to the audit record loses its intended user/module integrity boundary.

Profile-based footprint: 5.43% observed module signal (367/6,756 usable profiles; observed gate: webminlog).

Recorded conditions: Webmin Log delegate is restricted to selected users or modules but can submit another action ID. [nondefault]

Audited source: `webminlog/rollback.cgi`, `webminlog/webminlog-lib.pl`.

Consolidated reproduction records: `webmin-0076` (Webmin-log annotations bypass record-level authorization).

Module · Xterm (1)

WEBMIN-0959 — Xterm user parameter bypasses sudoenforce and can select a root shell

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

When xterm ACLs specify user=root and sudoenforce=1 for an authenticated Webmin user who maps to a local non-root account, resolve_shell_user normally selects that authenticated Unix account. Supplying an explicit user parameter makes the sudoenforce condition false, leaves the intermediate choice as root, and then applies the requested account. Requesting user=root therefore produces a root terminal despite the displayed sudo-only enforcement setting. The shipped default ACL uses user=root and sudoenforce=1, increasing the importance of correct role configuration.

Profile-based footprint: 76.87% observed module signal (5,193/6,756 usable profiles; observed gate: xterm).

Recorded conditions: The Webmin xterm ACL selects user=root with sudoenforce=1 while the authenticated identity maps to a local non-root account. [environmental]

Audited source: `xterm/xterm-lib.pl`, `xterm/lang/en`.

Cross-site scripting and HTML injection (140)

Core · Runtime (4)

WEBMIN-0518 — User and group chooser identities lacked contextual output encoding

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The reviewed chooser changes escaped backslashes but interpolated NSS user and group names directly into JavaScript string arguments and visible HTML. A quote-bearing identity could alter generated authenticated-origin markup when a Webmin user opened the chooser. Frozen 2.670 HTML-escapes displayed text but still protects only the outer single-quoted attribute delimiter; a double quote remains able to close the nested JavaScript string. The latest-stable image reproduced the incomplete encoding with a benign marker payload.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `user_chooser.cgi`, `group_chooser.cgi`.

WEBMIN-0829 — Forwarded host injection in authenticated WebSocket JavaScript

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

With proxy-header trust enabled and no explicit WebSocket host, Webmin uses X-Forwarded-Host as the browser-visible WebSocket authority. The authority is not constrained to host syntax or JavaScript-escaped before xterm inserts the URL into a single-quoted WebSocket constructor. A quote-bearing forwarded host can therefore inject same-origin JavaScript into an authenticated terminal page, subject to the stated proxy-header and page-access preconditions.

Profile-based footprint: 76.87% observed module signal (5,193/6,756 usable profiles; observed gate: xterm).

Recorded conditions: Proxy-header trust must be explicitly enabled. [nondefault]

Audited source: `web-lib-funcs.pl`, `xterm/index.cgi`.

WEBMIN-0941 — Pre-authentication login banner attribute injection

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

When a login banner contains the LOGINURL placeholder, pam_login.cgi and session_login.cgi filter the requester-controlled page value before substituting it into administrator-authored HTML. The filter is not aware of the later attribute context and does not encode quotes, so a crafted page value can terminate the intended attribute and add an autofocus event handler that executes before authentication.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: An administrator-configured login banner must contain the LOGINURL placeholder. [nondefault]

Audited source: `session_login.cgi`.

WEBMIN-0999 — Mailbox HTML sanitizer permits executable nested iframe srcdoc

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The mailbox attachment viewer serves HTML/XHTML after regex-based `filter_javascript` and `safe_urls` transformations that do not parse nested `iframe srcdoc` content. Entity-encoded script survives the filter and becomes active when an authenticated mailbox reader opens the attacker-supplied attachment. The frozen 2.670 filter-chain reproduction preserved a benign marker payload.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: The attacker can deliver a crafted HTML or XHTML attachment to the victim's mailbox. [environmental]

Audited source: `web-lib-funcs.pl`, `mailboxes/detach.cgi`.

Consolidated reproduction records: `webmin-0519` (Regex HTML sanitizer permits executable iframe srcdoc content).

Core · Cross-cutting (1)

WEBMIN-0243 — Stored XSS through mail-queue MIME attachment filenames

CVSS 4.0 Base (CVSS-B) · path `xss:module:postfix:queue-attachment-filename`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:sendmail:queue-attachment-filename`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The mail-queue detail views place a parser-derived MIME attachment filename into a single-quoted detach URL and into anchor text without HTML encoding. A remote sender can queue a crafted filename containing a single quote and markup; when a privileged operator views the queued message, that markup is interpreted in the Webmin origin. The frozen 2.670 source retains this parser-to-sink flow.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: postfix, sendmail).

Recorded conditions: An attacker-controlled MIME attachment filename remains in a Postfix queued message that an authenticated operator opens. [environmental]; An attacker-controlled MIME attachment filename remains in a Sendmail queued message that an authenticated operator opens. [environmental]

Audited source: `postfix/view_mailq.cgi`, `sendmail/view_mailq.cgi`.

Core · Integration (1)

WEBMIN-0236 — Single-group chooser fails to encode group names in JavaScript context

CVSS 4.0 Base (CVSS-B) · path `xss:module:useradmin:webmin-created-group`: 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:useradmin:nss-group`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The single-group chooser HTML-encodes the visible label but inserts the same NSS-visible group name into a double-quoted JavaScript string in an inline `onClick` handler. A quote-bearing group name can terminate the string and execute an expression when an authorized Webmin user opens or uses the chooser. The frozen CGI reproduction retained a harmless assignment expression in the handler.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: The attacker controls a group record returned through the host's configured NSS source. [environmental]

Audited source: `useradmin/my_group_chooser.cgi`, `useradmin/save_group.cgi`.

Consolidated reproduction records: `webmin-0300` (NSS group name causes stored XSS in the single-group chooser).

Core · MiniServ (1)

WEBMIN-0059 — Unauthenticated failed-login usernames become stored XSS in Webmin and Usermin blocked-user pages

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

With user blocking enabled, an unauthenticated client can store a markup-bearing username by reaching the failed-login threshold. Webmin and Usermin blocked-user administration pages render the stored token without encoding when an administrator explicitly opens the page.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Failed-login user blocking is enabled. [nondefault]; An administrator opens the blocked-user administration page. [environmental]

Audited source: `usermin/edit_blocked.cgi`, `webmin/webmin-lib.pl`.

Core · Webmin (3)

WEBMIN-0031 — External module metadata is rendered as trusted administrator HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Module names, download URLs, and descriptions obtained from an external catalog are inserted into link text, an event-handler attribute, and table cells without contextual encoding. If the catalog or its delivery path is controlled, opening the chooser can execute script in the Webmin administrative origin; the historical main-page notification sink was merged because it crosses the same metadata boundary.

Profile-based footprint: 5.39% observed module signal (364/6,756 usable profiles; observed gate: webmin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `webmin/third_chooser.cgi`, `ui-lib.pl`.

WEBMIN-0178 — Authenticated reflected XSS through request-controlled form errors

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Webmin::Form accepts request parameters named ui_error_<field> as field error messages, and Webmin::Section inserts those messages into generated HTML without encoding. The Zones create-form endpoint passes the parsed CGI hash into this framework and renders a name field, so a crafted ui_error_name value can place attacker-chosen script markup in a response viewed by an authenticated Webmin administrator. Frozen newest-stable source retains the same data flow under zones/lib/WebminUI, and the latest-stable container rendered the proof script element verbatim.

Profile-based footprint: 0.64% observed module signal (43/6,756 usable profiles; observed gate: zones).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `zones/lib/WebminUI/Form.pm`, `zones/lib/WebminUI/Section.pm`.

WEBMIN-2406 — Hostname title option reflects raw Host markup into HTML title output

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

With request-derived hostname display and hostname-prefixed titles enabled, Webmin concatenates HTTP_HOST into the page title without HTML encoding. A crafted raw Host value can close the title element and inject inert or active markup into login and module responses. Cross-browser delivery requires a deployment path that forwards an attacker-selected non-hostname Host value.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: The hostname display and hostname-prefixed title options must both be enabled. [nondefault]

Audited source: `web-lib-funcs.pl`, `session_login.cgi`.

Module · Apache (4)

WEBMIN-0007 — Raw Apache diagnostics reach Webmin HTML errors

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Apache configuration-test, lifecycle-command, and error-log text crosses into Webmin HTML error responses. The reviewed hardening encoded several such paths, but the frozen newest-stable source still has raw diagnostic interpolation in other paths, including after-change rollback errors and failed-start output. The Docker harness invoked the newest-stable after-change path and captured a raw HTML payload unchanged.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `apache/apache-lib.pl`, `apache/restart.cgi`.

WEBMIN-0363 — Stored active-markup injection in Apache authentication listings

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Apache authentication user names, group names, and group members are persisted with only minimal syntax validation and later rendered without HTML encoding. User and group names are sent to ui_link, whose fallback implementation directly interpolates both href and display text, while members are joined directly into a table cell. An actor able to influence an allowed authentication file can persist active markup that is interpreted in the Webmin origin when an authorized operator views the list. The condition remains in frozen newest stable and was reproduced with an inert active-markup marker in the isolated latest-stable image.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `apache/save_authuser.cgi`, `apache/list_authusers.cgi`.

WEBMIN-0532 — Stored attribute injection in the PHP 7 directive editor

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The PHP 7 directive editor parses each php_value/php_admin_value into a name and value, then places both strings directly in single-quoted HTML value attributes. The save routine accepts any non-empty value, so a quote can terminate the attribute and add event-handler attributes. A lower-privileged Apache-module operator or another configuration writer can persist the payload; viewing the directive form then renders it in the authenticated Webmin origin.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `apache/mod_php7.pl`.

WEBMIN-0605 — Delegated ErrorDocument URL causes stored XSS in the Apache directive editor

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

ErrorDocument URL mode validates only that the submitted value contains no whitespace. Quotes therefore survive saving into Apache configuration. When an operator later opens the ErrorDocument editor, edit_ErrorDocument concatenates the stored URL into a double-quoted input value without HTML attribute encoding. A whitespace-free URL can close the value and add autofocus and onfocus attributes, executing script in the Webmin origin under the viewer's session. Virtual-host and directive ACLs limit where the value can be stored but do not protect a more privileged operator who later views that configuration, and the default Webmin CSP allows inline handlers.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `apache/core.pl`, `apache/apache-lib.pl`.

Module · Backup Config (2)

WEBMIN-0424 — Stored XSS through scheduled backup destination rendering

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Backup Configuration persists local or remote scheduled-backup destinations containing HTML metacharacters, then renders parsed path components through `nice_dest` and `ui_link` without encoding. A module user can store active markup that executes for a later viewer with a more capable Webmin session. The frozen runtime returned an SVG event-handler marker in the destination anchor.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: backup-config).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `backup-config/backup-config-lib.pl`, `backup-config/index.cgi`.

Consolidated reproduction records: `webmin-0813` (Stored cross-site scripting through scheduled backup destinations).

WEBMIN-0566 — Backup restore error reflects verbose tar member names as active HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

During an actual configuration restore, verbose tar extraction writes selected member names into the combined output buffer. When tar returns a nonzero status, execute_restore wraps that buffer in a pre element without HTML escaping, and restore.cgi prints the resulting error into the authenticated page. An attacker-influenced backup containing a markup-bearing member name can therefore inject active content into a privileged operator's Webmin-origin response when extraction is made to fail.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: backup-config).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `backup-config/backup-config-lib.pl`, `backup-config/restore.cgi`.

Module · Bacula Backup (4)

WEBMIN-0362 — Bacula console output is rendered as active Webmin-origin HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Bacula restore CGI reads interactive output from the configured console process into the global wait buffer and prints it directly inside the authenticated Webmin response. A malicious or compromised Bacula Director (or substituted configured console process) can include active HTML in the first file-selection prompt; because Webmin performs no HTML escaping, the markup executes in the victim administrator's Webmin origin and can issue same-origin administrative requests. The path is an XSS/session-impact primitive, not direct root shell execution.

Profile-based footprint: 81.88% observed module signal (5,532/6,756 usable profiles; observed gate: bacula-backup).

Recorded conditions: The attacker controls output from the configured Bacula Director or console process. [environmental]; A privileged Webmin user opens the affected restore view. [environmental]

Audited source: `bacula-backup/restore.cgi`, `web-lib-funcs.pl`.

WEBMIN-0452 — Stored XSS through unescaped Bacula resource names

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Bacula list pages read resource Name values from configuration and supply them directly as the visible text argument to ui_link. The default helper inserts that argument into HTML without html_escape, while module save paths accept HTML metacharacters in resource names. A crafted persisted name can therefore execute script-capable markup when an authenticated operator loads an affected list page. The latest-stable Docker reproduction confirmed unchanged emission of an img element carrying an onerror handler.

Profile-based footprint: 81.88% observed module signal (5,532/6,756 usable profiles; observed gate: bacula-backup).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `ui-lib.pl`, `bacula-backup/list_clients.cgi`.

WEBMIN-0547 — Stored XSS through a scheduled Bacula Copy-job name without a destination volume

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

A Bacula Backup module user can save a job name containing whitespace-free active HTML because save_job.cgi only checks that the name contains a non-whitespace character. For a padded scheduled Copy-job status row with no destination volume, get_director_status retains that name. dirstatus_form.cgi passes it through joblink into ui_columns_row; ui_link and ui_columns_row treat their text arguments as HTML and do not escape them. When a privileged Webmin user views Director Status, the stored payload executes in the Webmin origin. The path requires delegated Bacula configuration access or director-output control and a later victim page view. Successful script execution inherits the victim's Webmin session and can cross module boundaries.

Profile-based footprint: 81.88% observed module signal (5,532/6,756 usable profiles; observed gate: bacula-backup).

Recorded conditions: A padded scheduled Copy-job status row has no destination volume and retains the crafted job name. [environmental]

Audited source: `bacula-backup/bacula-backup-lib.pl`, `bacula-backup/save_job.cgi`.

WEBMIN-0721 — Bacula treechooser parameters permit inline-script injection

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Bacula `treechooser.cgi` interpolates decoded `job` and `volume` request values into single-quoted inline JavaScript arguments without encoding. A crafted accepted request can terminate the string and execute script in the authenticated Webmin origin. Default referrer checking requires an accepted navigation context, such as a same-origin request.

Profile-based footprint: 81.88% observed module signal (5,532/6,756 usable profiles; observed gate: bacula-backup).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `bacula-backup/treechooser.cgi`, `web-lib-funcs.pl`.

Consolidated reproduction records: `webmin-0729` (Bacula treechooser parameters permit inline-script injection).

Module · Bind8 (1)

WEBMIN-0430 — BIND ALL-records renderer emits zone data as active HTML

CVSS 4.0 Base (CVSS-B) · path `xss:module:bind8:all-records-webmin-editor`: 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:bind8:all-records-external-zone-data`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The BIND ALL/custom-record editor escapes ordinary values but joins and emits ALL-record data without HTML encoding. A TXT record containing an SVG event handler appeared verbatim in the frozen 2.670 table response and can execute in an authenticated Webmin session.

Profile-based footprint: 94.63% observed module signal (6,393/6,756 usable profiles; observed gate: bind8).

Recorded conditions: The attacker controls DNS data consumed by the managed zone. [environmental]

Audited source: `bind8/edit_recs.cgi`.

Consolidated reproduction records: `webmin-0190` (Stored XSS in BIND ALL-records value rendering).

Module · Bsdexports (1)

WEBMIN-0357 — Authenticated reflected XSS in Dutch BSD Exports directory validation

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

BSD Exports save_export.cgi splits the request-controlled dirs value and passes a nonexistent directory token to the Dutch save_edir message, which interpolates the token into error-page HTML without encoding. On supported BSD-family installations, a crafted request viewed by an authorized Webmin user therefore executes attacker-controlled markup in the Webmin origin. A root Webmin victim makes this root-equivalent session compromise, but the defect is not itself direct OS command execution.

Profile-based footprint: 0.56% observed module signal (38/6,756 usable profiles; observed gate: bsdexports).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `bsdexports/lang/nl`, `bsdexports/save_export.cgi`.

Module · Bsdfdisk (2)

WEBMIN-0964 — Stored HTML injection in BSD disk usage status rendering

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The shared fdisk::device_status_link helper builds HTML containing the raw directory/status string returned for a mounted or configured device. BSD disk pages pass this HTML through localized usage messages and table or confirmation rendering without escaping the underlying string. A mount directory containing markup

therefore becomes active same-origin HTML when an authenticated user views the affected disk page. Exploitation requires control of a relevant mount entry and a separate page view, but Webmin module permissions can make the mount-entry controller less privileged than the victim session.

Profile-based footprint: 3.27% observed module signal (221/6,756 usable profiles; observed gate: bsdfdisk).

Recorded conditions: The host runs an affected BSD disk usage page. [environmental]; The attacker can control a mounted or configured device's directory/status string. [environmental]; An authenticated Webmin user opens the affected BSD disk page. [environmental]

Audited source: `fdisk/fdisk-lib.pl`, `bsdfdisk/edit_part.cgi`.

WEBMIN-2421 — Device-reported disk model can become active HTML in the Webmin disk list

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The FreeBSD disk module obtains model strings from device and system-tool output and uses the value as an HTML table cell without html_escape on the disk-list page. ui_columns_row concatenates cell values as HTML. A crafted model string that reaches these sources can therefore create markup in the authenticated Webmin origin when an administrator views the list. The disk-detail page's separate encoded rendering does not protect the list-page sink.

Profile-based footprint: 3.27% observed module signal (221/6,756 usable profiles; observed gate: bsdfdisk).

Recorded conditions: The host runs the FreeBSD disk module page. [environmental]; The attacker can influence a device or tool model string returned to the disk list. [environmental]; An authenticated administrator opens the affected FreeBSD disk list. [environmental]

Audited source: `bsdfdisk/bsdfdisk-lib.pl`, `bsdfdisk/index.cgi`.

Module · Cluster Passwd (1)

WEBMIN-0761 — Stored username attribute injection in the cluster password user list

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

cluster-passwd/index.cgi HTML-escapes each cached username when using it as link text, but inserts the same value directly into the href passed to ui_link. ui_link encloses that href in single quotes without encoding it. A username containing a single quote can therefore terminate the href value and add an event-handler attribute. The value is persistent cluster-host data, and the affected page is an authenticated administrative surface. Exploitation requires influence over cached user data from a configured host and a privileged viewer, which limits reach, but successful script execution can issue same-origin administrative requests with the viewer's authority.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: cluster-passwd).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: `cluster-passwd/index.cgi`, `ui-lib.pl`.

Module · Cluster Software (3)

WEBMIN-0113 — Stored XSS through managed-server metadata in Cluster Software table mode

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

When Cluster Software is configured for table mode, its managed-systems table renders the stored server description as a raw table cell and the stored hostname as raw link text. The Servers save path accepts the description without HTML validation or output encoding. A Webmin account permitted to edit a Servers entry can therefore persist active markup that runs when another authenticated account views the Cluster Software

index. Because Webmin modules can be delegated separately, this can execute same-origin actions with the viewer's broader authority. The condition requires table mode, a managed server, and a victim page view.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: cluster-software).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: `cluster-software/index.cgi`, `ui-lib.pl`.

WEBMIN-0621 — Stored XSS through Cluster Software server metadata

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Server descriptions and group names persisted by the Servers module can contain HTML metacharacters. Cluster Software renders those fields as raw option or icon-title HTML. This creates a stored same-origin script-injection path across Webmin module permissions: a user able to edit server metadata can target a user who views Cluster Software. The frozen 2.670 container returned both crafted description and group payloads verbatim in an HTTP 200 index response whose policy permits inline script handlers.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: cluster-software).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: `cluster-software/index.cgi`, `servers/save_serv.cgi`.

WEBMIN-0844 — Unescaped package metadata remains executable in Cluster Software views

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Cluster Software caches package metadata received from managed hosts and renders package name, class, and description directly in search result columns. The installation-completion page likewise passes several package_info fields, including free-text vendor or class data on supported package systems, directly to ui_table_row. Both UI helpers treat these arguments as HTML rather than text. A malicious package or compromised managed host can therefore persist crafted markup that executes when an authenticated administrator views a matching search result or installation detail. Exact newest-stable source inspection and an isolated Docker test confirmed the raw flows and payload preservation.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: cluster-software).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: `cluster-software/search.cgi`, `cluster-software/do_install.cgi`.

Module · Cluster Useradmin (5)

WEBMIN-0114 — Stored HTML/script injection in managed-server table view

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Cluster Users and Groups table view takes persisted managed-server host/description metadata and supplies it to ui_columns_row as HTML without contextual escaping. The Servers module stores the description from a submitted form without output-oriented encoding, and access to the Servers and Cluster Users modules can be assigned separately. A principal able to modify a server description can therefore persist active markup that executes when another Webmin operator opens the table-mode cluster page.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: cluster-useradmin/index.cgi, ui-lib.pl.

WEBMIN-0423 — Stored cross-site scripting in the Unix user chooser

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The Cluster Users and Groups membership button opens /useradmin/my_user_chooser.cgi. That chooser reads Unix/NSS account records and renders the account name and GECOS/display-name values into an inline onClick handler; it also writes the display name directly into an HTML table cell. Only apostrophes in the display name are replaced, which does not encode HTML metacharacters or make the value safe for JavaScript. An attacker who controls visible account metadata can therefore supply active markup that runs when a permitted Webmin administrator loads the chooser. Webmin's useradmin module-access check and the chooser's ACL filtering remain in force, so this is not an unauthenticated or cross-module authorization bypass. The privileged Webmin origin makes successful execution capable of issuing same-origin administrative requests with the viewer's authority.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

WEBMIN-0718 — Managed-host account names render without HTML escaping in the central Webmin origin

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

cluster-useradmin refreshes user and group records from managed Webmin hosts and stores them locally. On the managed-host detail page, each stored username and group name is passed directly as the display-text argument to ui_link. The default helper inserts that argument verbatim between anchor tags. A compromised managed host can therefore return an account name containing active HTML; when a central administrator views the host, the browser interprets the markup under the central Webmin origin. This creates a management-plane pivot capable of administrator-context actions or session compromise. The path requires a compromised managed host and a later administrator page view.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: cluster-useradmin/refresh.cgi, cluster-useradmin/edit_host.cgi.

WEBMIN-0762 — Stored server-label markup executes in Cluster Users views

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

server_name concatenates persisted server host and description values as raw strings. Cluster Users index.cgi sends that value to the managed-server icon title, and generate_icon embeds the title directly inside an anchor without HTML encoding. The Servers editor accepts any non-whitespace host in mode 0 and persists it verbatim, while description is also stored without HTML normalization. A delegated Servers editor can therefore save markup that is interpreted in the same Webmin origin when a principal with Cluster Users access views the managed server. That script can act with the viewer's broader Webmin permissions. The path requires authenticated server-configuration access and a later viewer with Cluster Users access.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: cluster-useradmin/cluster-useradmin-lib.pl, cluster-useradmin/index.cgi.

WEBMIN-0891 — Stored XSS through unencoded server descriptions in cluster search results

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The multi-result Cluster Users user and group searches collect each matched account's managed-server label from the shared Servers configuration and pass the joined labels directly to ui_columns_row. That helper treats every column as HTML. A Webmin identity allowed to edit Servers-module entries can store active markup in a server description; when a separate Cluster Users identity renders a matching multi-result search, the markup is placed into the privileged Webmin page without HTML encoding. This creates a cross-user stored-XSS path across module roles.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: `cluster-useradmin/search_user.cgi`, `cluster-useradmin/search_group.cgi`.

Module · Cluster Usermin (2)

WEBMIN-0112 — Managed Usermin version is emitted as unescaped table-cell HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Cluster Usermin obtains a managed host's Usermin version through a remote management call and stores it in the host record. In table mode, the index passes that value directly to ui_columns_row, whose contract treats cell values as HTML and concatenates them into a td without encoding. A managed server that returns crafted markup can therefore place active markup in the central Webmin page viewed by an authenticated administrator. The Docker test used an inert onerror marker and confirmed raw preservation at the rendering boundary; it did not execute script in a browser.

Profile-based footprint: 85.07% observed module signal (5,747/6,756 usable profiles; observed gate: cluster-usermin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: `cluster-usermin/index.cgi`, `ui-lib.pl`.

WEBMIN-0627 — Stored XSS in Cluster Usermin module and theme selectors

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Cluster Usermin obtains module and theme hashes from managed Usermin hosts, saves them in its per-host cache, and later places each dir and desc value directly into HTML option markup. A party controlling metadata on a managed host can close the option element and inject executable HTML into the authenticated central Webmin page. The latest-stable lab reproduced a raw img element carrying an onerror handler in the HTTP response. This is a trust-boundary pivot from a managed host into an administrator's central Webmin origin.

Profile-based footprint: 85.07% observed module signal (5,747/6,756 usable profiles; observed gate: cluster-usermin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Module · Cluster Webmin (2)

WEBMIN-0109 — Table mode exposes stored and remote host metadata to raw HTML rendering

CVSS 4.0 Base (CVSS-B) · path `xss:module:cluster-webmin:stored-server-description`: 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:cluster-webmin:remote-host-version`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The managed-host table passes the locally stored server description and remote-reported Webmin version to ui_columns_row without HTML encoding. That helper deliberately accepts HTML and inserts each string directly into a table cell. With table mode enabled, crafted markup saved by a Servers-module editor or supplied by a managed host is therefore rendered in the Cluster Webmin origin when another user opens the index. The default-off configuration and required metadata influence bound the exposed path.

Profile-based footprint: 4.19% observed module signal (283/6,756 usable profiles; observed gate: cluster-webmin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]; The attacker controls metadata returned by a linked or managed Webmin host. [environmental]

Audited source: `cluster-webmin/index.cgi`, `ui-lib.pl`.

WEBMIN-0530 — Stored server labels allow cross-user script execution in Cluster Webmin

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

A Webmin account granted the Servers module's edit/add capabilities can save a credentialed server whose description contains markup. Cluster Webmin's server_name concatenates that stored description without encoding, and its index writes the result directly into an HTML option. When a Cluster Webmin user renders the page, the injected inline script executes in that user's authenticated Webmin origin. The reproduced response's Content-Security-Policy permits inline script, so the payload can act with the viewer's same-origin authority.

Profile-based footprint: 4.19% observed module signal (283/6,756 usable profiles; observed gate: cluster-webmin).

Recorded conditions: Cluster modules require linked/managed hosts and are not functional from module installation alone. [nondefault]

Audited source: `cluster-webmin/cluster-webmin-lib.pl`, `cluster-webmin/index.cgi`.

Module · Cpan (1)

WEBMIN-0275 — Authenticated reflected XSS in CPAN return-link footer

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CPAN download and error paths pass request-controlled return and returndesc parameters to footer rendering. The footer interpolates the destination into href and the description into anchor content without contextual escaping. An authenticated crafted request can therefore inject active HTML in the Webmin origin. A root Webmin victim gives root-equivalent administrative impact, but the primitive is browser-side and is not direct remote shell execution.

Profile-based footprint: 4.25% observed module signal (287/6,756 usable profiles; observed gate: cpan).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `cpan/download.cgi`, `web-lib-funcs.pl`.

Module · Custom (1)

WEBMIN-0751 — Stored active markup is rendered in Custom Commands parameter-form headings

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Custom Commands persists an editor-supplied HTML description and a plain description, and the command and SQL parameter forms pass the selected value directly to ui_table_start. The table helper emits its heading as raw HTML. Webmin's Custom Commands ACL separates the ability to edit commands from the list of commands another user can execute, so a command editor can place active markup in a form viewed under an executor's authenticated Webmin origin. The same module demonstrates the intended defensive control by

using filter_javascript for an HTML-description rendering path on its main page, but the two parameter-form sinks omit it.

Profile-based footprint: 5.52% observed module signal (373/6,756 usable profiles; observed gate: custom).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `custom/form.cgi`, `custom/sqlform.cgi`.

Module · DHCP (5)

WEBMIN-0464 — Reflected XSS in DHCP lease network subheader

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The authenticated DHCP lease viewer uses request-controlled network and netmask values as HTML-bearing substitutions for the page subheader and passes the result to ui_print_header without encoding. A crafted URL can therefore inject active script into a response viewed by a Webmin user who has DHCP lease-read access. Because the script runs in Webmin's origin, it can act with the victim's administrative session privileges. The flaw is present in frozen stable 2.670 and the latest-stable-only Docker lab preserved a harmless script marker verbatim in the generated header HTML.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `dhcpd/list_leases.cgi`, `ui-lib.pl`.

WEBMIN-0553 — DHCP host lookup reflects active markup in no-match errors

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The DHCP host lookup normalizes a request as a MAC address only when it contains a colon, hyphen, or dot. A crafted no-delimiter value therefore remains unchanged even when its hexadecimal characters would otherwise trigger canonicalization. If no host matches, lookup_host.cgi substitutes that value into lookup_ehost and Webmin's error renderer emits the original message into an HTML heading without encoding. A crafted link can consequently execute JavaScript in the origin of an authenticated victim with DHCP-module access. Because Webmin commonly operates as a root administration interface, script running in a root administrator's session can perform privileged same-origin actions.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `dhcpd/lookup_host.cgi`, `dhcpd/lang/en`.

WEBMIN-0635 — Delegated DHCP editor can store JavaScript in shared-network names

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The DHCP shared-network save handler accepts any non-whitespace name and persists it without JavaScript-string encoding. The host editor later inserts that name into a double-quoted JavaScript literal used to construct assignment options, then calls the generated setparent function with the selected network. A user with scoped DHCP edit access can therefore store a name that closes the string and evaluates attacker-selected JavaScript when another authenticated user opens the affected editor. Because the script runs in Webmin's authenticated origin, a victim with broader Webmin privileges can be acted as, including a root administrator.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: dhcpd/save_shared.cgi, dhcpd/edit_host.cgi.

WEBMIN-0755 — Stored XSS through a DHCP shared-network name

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The by-network DHCP lease summary renders a configured shared-network name as a raw ui_columns_row cell. The shared-network save path accepts any non-whitespace name, so a value such as an SVG element with an event handler survives parsing and is inserted into the authenticated page as markup. A configuration-capable actor can therefore persist script-bearing content that executes when a lease-reading Webmin user visits the summary. The page requires r_leases access, while the attacker also needs control of DHCP configuration.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: The victim has r_leases access to the affected DHCP lease table. [environmental]; A configuration-capable actor can store a markup-bearing shared-network name. [environmental]

Audited source: dhcpd/list_leases.cgi, dhcpd/save_shared.cgi.

WEBMIN-0878 — Stored script-capable HTML injection through DHCP object descriptions

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

DHCP object descriptions are accepted as configuration comments and two tabular index paths place those comments into ui_checked_columns_row without HTML output encoding. That helper emits supplied cells verbatim. In the frozen latest-stable container, a subnet description containing an img element with an inline event handler appeared as a literal element in the authenticated DHCP response, while the separately constructed label copy was encoded. A user who can write an object description can therefore store script-capable markup that runs in the Webmin origin when another authorized user views the affected table. The observed Content-Security-Policy explicitly permits unsafe-inline script.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: dhcpd/index.cgi, ui-lib.pl.

Module · Documentation (1)

WEBMIN-0250 — Plain-HTTP documentation search results execute in the authenticated Webmin origin

CVSS 4.0 Base (CVSS-B) · path xss:module:man:remote-result-description: 2.3 (Low) · CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

CVSS 4.0 Base (CVSS-B) · path xss:module:man:remote-result-href: 2.3 (Low) · CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The Documentation module fetches Google search results over plaintext HTTP and emits parsed anchor descriptions and href values into Webmin HTML without context-appropriate encoding. An on-path attacker who controls the response can plant script that runs when an authenticated user performs the search. The script inherits that user's Webmin origin authority.

Profile-based footprint: 4.57% observed module signal (309/6,756 usable profiles; observed gate: man).

Recorded conditions: The attacker can intercept or impersonate the plaintext remote search response. [environmental]

Audited source: man/man-lib.pl, man/search.cgi.

Consolidated reproduction records: `webmin-0685` (Plaintext Google search response permits authenticated-origin XSS).

Module · Dovecot (2)

WEBMIN-0771 — Unescaped no-init Dovecot startup output reaches Webmin error HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

When Dovecot has no init_script configured, start_dovecot executes the configured Dovecot command, captures its combined output, and on failure returns that output inside raw <pre> markup. start.cgi passes the value to error(), whose <pre>-aware HTML path emits it as markup. A diagnostic containing attacker-controlled HTML can therefore run script in the Webmin origin when another authenticated user triggers startup. The action-backed branch entity-encodes its output, but the fallback does not.

Profile-based footprint: 94.57% observed module signal (6,389/6,756 usable profiles; observed gate: dovecot).

Recorded conditions: Dovecot has no init_script configured, selecting the fallback command-output path. [nondefault]; The failed Dovecot start emits diagnostic text containing attacker-controlled HTML. [environmental]; An authenticated Webmin user triggers the failed start. [environmental]

Audited source: `dovecot/dovecot-lib.pl`.

WEBMIN-0917 — Dovecot listen validation reflects script-capable markup

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Dovecot networking save handler tokenizes the custom listen value and sends any invalid token directly to text("net_ealisten", $ip). The English message embeds that substitution in an HTML tt element, while Webmin's text helper and HTML error path do not encode the value. A crafted invalid token can therefore become active markup in an authenticated Webmin error response. The isolated frozen-2.670 reproduction observed a literal SVG onload element in the returned HTML. The path requires an authenticated victim with Dovecot-module access to render the crafted response.

Profile-based footprint: 94.57% observed module signal (6,389/6,756 usable profiles; observed gate: dovecot).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `dovecot/save_net.cgi`, `dovecot/lang/en`.

Module · Exports (1)

WEBMIN-0341 — Authenticated reflected XSS in NFS Exports invalid-host error

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

NFS Exports save_export.cgi substitutes the request-controlled host parameter into the localized save_ehost message and passes that message to Webmin's fatal error renderer. The renderer keeps the original message for HTML output even though it strips tags only into a separate logging string. An authenticated request using the Dutch locale therefore reflects active HTML in the Webmin origin. A crafted request shown to a privileged Webmin victim can execute script with that victim's Webmin authority; the issue does not by itself provide unauthenticated access or direct OS command execution.

Profile-based footprint: 4.11% observed module signal (278/6,756 usable profiles; observed gate: exports).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `exports/lang/nl`, `exports/save_export.cgi`.

Module · Fail2Ban (1)

WEBMIN-0855 — Authenticated reflected HTML injection in Fail2Ban purge-age validation

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Fail2Ban global configuration save endpoint selects dbpurgeagecus in custom mode and passes it to time_to_seconds_error(). For an input consisting of digits followed by an invalid non-whitespace unit, the helper substitutes that unit directly into the HTML-formatted config_ewrongunit string. Webmin's text() helper does not escape substitutions, and error() renders the argument as HTML. An authenticated latest-stable test using the inert unit marker batch-proof returned that marker as a literal element rather than escaped text. This establishes reflected HTML injection; no JavaScript payload was executed.

Profile-based footprint: 93.75% observed module signal (6,334/6,756 usable profiles; observed gate: fail2ban).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `fail2ban/save_config.cgi`, `fail2ban/fail2ban-lib.pl`.

Module · Fetchmail (1)

WEBMIN-0172 — Reflected XSS in Fetchmail user context headers

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

When the module uses its shipped per-user configuration mode (config_file is empty), the Fetchmail poll editor, global editor, and check page place the decoded user request parameter inside literal HTML markup and give it to a subheader renderer that treats input as HTML. The shipped mode-0 ACL accepts arbitrary target strings and the code does not require getpwnam() to return a user before rendering. A crafted link followed by an authenticated Webmin victim therefore executes script in the victim's Webmin origin. The user-interaction requirement limits reliability, but a root administrator victim makes the resulting session capability root-equivalent.

Profile-based footprint: 87.30% observed module signal (5,898/6,756 usable profiles; observed gate: fetchmail).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `fetchmail/edit_poll.cgi`, `fetchmail/fetchmail-lib.pl`.

Module · Filemin (2)

WEBMIN-0557 — Authenticated File Manager HTML injection across path and error sinks

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Request-controlled paths and filenames reached HTML, attribute, URL, inline-JavaScript, and error-list contexts without consistent contextual encoding. A crafted authenticated request can execute active markup in the Webmin origin. The frozen newest stable still renders a rename destination name directly inside the File Manager error list; the isolated latest-stable reproduction returned the raw img/onerror marker.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `filemin/rename.cgi`, `filemin/filemin-lib.pl`.

WEBMIN-0785 — Filemin error helper renders attacker-influenced filenames as active HTML

CVSS 4.0 Base (CVSS-B) · path `xss:module:filemin:request-controlled-error-filename`: 4.8 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:filemin:filesystem-controlled-error-filename`: 2.4 (Low) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Filemin's shared `print_errors` helper emits request-derived filename errors as HTML. The chown and delete failure paths can therefore return active markup from a crafted nonexistent name; the frozen runtime rendered an `img/onerror` marker from both endpoints. Exploitation requires an authenticated Filemin action or an induced action in an authenticated session.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: The attacker controls a filename in a directory accessible to the Filemin operator. [environmental]

Audited source: `filemin/filemin-lib.pl`, `filemin/rename.cgi`.

Consolidated reproduction records: `webmin-0607` (Filemin failure messages render crafted filenames as active HTML).

Module · Filter (1)

WEBMIN-0327 — Filter move error reflects executable markup in an authenticated page

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

move.cgi uses the request-controlled movefrom folder identifier in an error message when folder lookup fails. The text substitution does not encode HTML, and the core fallback error renderer inserts the resulting message directly into an authenticated HTML response. A crafted request can therefore execute script in the victim's Webmin/Usermin origin. Exploitation requires a logged-in authorized victim and an existing filter index; the resulting script inherits that victim session rather than directly executing on the server.

Profile-based footprint: 87.39% observed module signal (5,904/6,756 usable profiles; observed gate: filter).

Recorded conditions: The authenticated victim has an existing filter index for the move error path. [environmental]

Audited source: `filter/move.cgi`, `web-lib-funcs.pl`.

Module · Firewall (4)

WEBMIN-0218 — Stored chain names permit cross-user script injection

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Chain creation and rename accept any non-whitespace name, while the firewall index embeds each name directly inside an HTML heading. A restricted user with chain-creation access can persist a compact SVG event-handler payload that a higher-privileged administrator receives as live markup. The default stable response policy permits unsafe-inline script. The lab used a harmless assignment marker, did not execute browser code, and confirmed the exact raw element in the labadmin response.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `firewall/newchain.cgi`, `firewall/index.cgi`.

WEBMIN-0219 — Stored firewall comments permit cross-user script injection

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The rule editor stores arbitrary comment text and the index passes the selected comment representation into an HTML column without encoding. When comment display is enabled, a restricted user who may edit an allowed rule can persist an event-handler payload that a higher-privileged administrator receives as live markup under

an unsafe-inline response policy. The lab used a harmless assignment marker and confirmed the raw element in the labadmin response without executing browser code.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: Firewall comment display must be enabled before the stored comment is rendered. [nondefault]

Audited source: `firewall/save_rule.cgi`, `firewall/index.cgi`.

WEBMIN-0264 — Firewall-only user can plant stored XSS in administrator chain views

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Webmin 2.670's firewall newchain.cgi checks the delegated newchain permission and table ACL but validates the supplied chain name only with /^\S+$/. It then persists the value in the root-owned firewall rules file. When the firewall index is opened, index.cgi places each stored chain name inside an HTML-bearing text substitution without HTML escaping. A firewall-only delegated account can therefore plant active markup that executes when a higher-privileged administrator views the firewall module, enabling authenticated same-origin actions with the victim's Webmin session. The latest-stable lab reproduced this with separate delegated and administrator accounts.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `firewall/newchain.cgi`, `firewall/index.cgi`.

WEBMIN-0600 — Stored XSS in the firewall cluster add-server selector

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Webmin server descriptions are persisted from the Servers Index module without HTML-output validation. The Linux Firewall cluster page escapes descriptions in its existing-server table but emits the same persisted description directly as option content in the add-server selector. A crafted value can close the option and select elements and inject event-capable markup in the Webmin origin. Separate module ACL checks make this relevant when a server-entry editor and a firewall-cluster operator are different users; even for one user, stored markup is rendered on a later privileged page view.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: The sink requires configured firewall-cluster server state. [nondefault]

Audited source: `servers/save_serv.cgi`, `firewall/cluster.cgi`.

Module · Firewall6 (1)

WEBMIN-0542 — Firewall rule comments reach a raw HTML rendering sink

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Rule comments are stored without HTML encoding, passed directly into the firewall rule-list column array, and concatenated by the UI row renderer. An authenticated rule editor can persist Webmin-origin markup viewed by another account. Raw HTML injection is confirmed; event-handler execution remains dependent on the victim browser and response policy.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: firewall/index.cgi, ui-lib.pl.

Module · Htpasswd File (1)

WEBMIN-0177 — Unescaped username and request values permit stored and reflected XSS

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The added HTPasswd File module has two related missing-output-encoding paths. First, save.cgi accepts a created or renamed username after checking only for a colon, list_users preserves that text, and index.cgi inserts it into link content without HTML encoding. Second, edit.cgi places the idx and new request values directly inside single-quoted hidden-input attributes. An authenticated module user can therefore persist active markup in a managed username when create or rename rights are available, while a crafted edit URL can reflect active markup to another authorized user. Script executing in the Webmin origin can issue requests with the viewer's Webmin session and privileges. Both vectors require authenticated module interaction or victim navigation; cross-user script execution can still act with the viewer's privileges.

Profile-based footprint: 0.01% observed module signal (1/6,756 usable profiles; observed gate: htpasswd-file).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: htpasswd-file/save.cgi, htpasswd-file/index.cgi.

Module · Idmapd (1)

WEBMIN-0371 — Stored XSS through idmapd daemon-path configuration

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

A delegated user who can edit idmapd module configuration can store HTML in the free-text idmapd_path value. When that configured path does not exist, idmapd/index.cgi inserts it without HTML encoding into the index_eidmapd message, including the Norwegian template added by the reviewed change. Opening the module therefore renders attacker-controlled active markup in the authenticated Webmin origin. The path requires module-configuration permission and a later viewer of the error page.

Profile-based footprint: 4.13% observed module signal (279/6,756 usable profiles; observed gate: idmapd).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: idmapd/index.cgi, idmapd/lang/no.

Module · Inetd (1)

WEBMIN-0334 — Authenticated reflected XSS in Norwegian Inetd service-name validation

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

When Norwegian is selected, save_serv.cgi inserts an invalid attacker-controlled service name into error_invalidservice, and Webmin's shared fatal-error renderer emits the resulting string as HTML without encoding. An authenticated request accepted by Webmin's referer policy can therefore execute active markup in the Webmin origin. The path requires authentication, module access, Norwegian locale selection, and an accepted request under the referer check.

Profile-based footprint: 4.23% observed module signal (286/6,756 usable profiles; observed gate: inetd).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: inetd/lang/no, inetd/save_serv.cgi.

Module · Init (2)

WEBMIN-0149 — Stored action-log XSS lets a run-only Bootup user target Webmin administrators

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

init/start_stop.cgi logs the request-controlled name parameter after running a valid service action. init/log_parser.pl interpolates the log object into HTML, and the Webmin Actions Log applies filter_javascript rather than HTML encoding it. Character-reference-encoded script markup inside an iframe srcdoc attribute survives that filter. A bootup=2 user can store this markup while performing an allowed action; it is then emitted when an administrator views the log. Because the response permits same-origin child frames and unsafe-inline script, the payload executes in the Webmin origin and can perform authenticated administrative requests as the victim, providing a realistic privilege-escalation path to root-level Webmin capabilities.

Profile-based footprint: 5.45% observed module signal (368/6,756 usable profiles; observed gate: init).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `init/start_stop.cgi`, `init/log_parser.pl`.

WEBMIN-0517 — System journal messages produce stored XSS in Init service-start output

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

On a failed systemd service start, the Init module appends global journalctl -xe output to the command result and mass_systemd.cgi emits that result as raw HTML inside a pre element. A local unprivileged process can normally submit messages to the system journal; a message containing a closing pre tag and script markup therefore becomes active content when a bootup-authorized Webmin user triggers the failure. Script execution occurs in the authenticated Webmin origin and can act with the victim session's administrative privileges, making an Init-only or full administrator session a root-equivalent escalation target.

Profile-based footprint: 5.45% observed module signal (368/6,756 usable profiles; observed gate: init).

Recorded conditions: A systemd service start fails and the Init page includes journal output. [environmental]; A local unprivileged process can submit attacker-controlled journal text. [environmental]; An authorized Webmin user triggers the failed start and views the resulting page. [environmental]

Audited source: `init/init-lib.pl`, `init/mass_systemd.cgi`.

Module · Ipfw (1)

WEBMIN-0546 — Authenticated stored XSS through the IPFW version override

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

An authenticated Webmin user who can configure the IPFW module can submit an arbitrary version value even though the UI advertises only 0, 1, and 2. The stored value is later copied into ipfw_version, substituted by text() without escaping, and emitted verbatim by the stock theme post-header. A privileged operator opening the module therefore executes attacker-controlled markup in the Webmin origin, making the impact root-equivalent through the victim session; this is not direct unauthenticated RCE.

Profile-based footprint: 84.99% observed module signal (5,742/6,756 usable profiles; observed gate: ipfw).

Recorded conditions: The host runs the IPFW BSD Firewall backend. [environmental]

Audited source: `ipfw/config.info`, `config-lib.pl`.

Module · Ipsec (1)

WEBMIN-0302 — Authenticated reflected XSS in German IPsec duplicate-interface error

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The German config_eri string interpolates $1, and ipsec/save_config.cgi supplies an unvalidated duplicate ri_N POST value before passing the resulting string to error(). The HTML error renderer emits that original string without HTML encoding. In latest stable 2.670, an authenticated request returned an active svg/onload element verbatim under a CSP that permits inline script. Exploitation requires German locale and a victim session authorized for IPsec; cross-user delivery on the standard setup must also pass Webmin's default same-origin/trusted-referrer defense. If a privileged administrator is reached, script in the Webmin origin can issue root-equivalent administrative requests; delivery remains subject to the stated preconditions. This is XSS, not direct unauthenticated OS command execution.

Profile-based footprint: 85.49% observed module signal (5,776/6,756 usable profiles; observed gate: ipsec).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `ipsec/save_config.cgi`, `ipsec/lang/de`.

Module · Iscsi Target (2)

WEBMIN-0361 — Delegated iSCSI module user can plant stored JavaScript in administrative lists

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

An authenticated Webmin user who has iSCSI Target module access can store attacker-controlled markup either as an IncomingUser username or as an allow-list name. The module later concatenates those persisted values into target and allow-list administrative pages without HTML encoding. When a more privileged administrator views the affected page, the payload executes as same-origin Webmin script under the victim's session, crossing the intended delegated-module-user boundary. No direct server-side remote-code-execution path was found in this batch.

Profile-based footprint: 84.93% observed module signal (5,738/6,756 usable profiles; observed gate: iscsi-target).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `iscsi-target/save_target.cgi`, `iscsi-target/index.cgi`.

WEBMIN-0391 — Authenticated reflected XSS in iSCSI invalid-address error

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

In the iSCSI Target module, save_allow.cgi splits the request-controlled addrs field, passes each invalid token to text('allow_eaddr', $addr), and sends the resulting localized string to Webmin's fatal-error renderer. The Norwegian template places $1 into the sentence without HTML encoding, while the error renderer emits the string as HTML. An authenticated crafted request therefore reflects attacker-selected active markup in the Webmin origin. This is root-equivalent when the victim holds a root Webmin session, but it is not direct operating-system command execution.

Profile-based footprint: 84.93% observed module signal (5,738/6,756 usable profiles; observed gate: iscsi-target).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `iscsi-target/save_allow.cgi`, `iscsi-target/lang/no`.

Module · Iscsi Tgtd (1)

WEBMIN-0476 — iSCSI manual editor discloses arbitrary local files and reflects raw path markup

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:A/VC:H/VI:N/VA:N/SC:L/SI:L/SA:N`.

An authenticated Webmin principal granted the iscsi-tgtd module can supply an arbitrary absolute file path to edit_manual.cgi. The handler uses that request value without checking it against the primary and included TGTd configuration files, reads the selected path as the root-running Webmin process, and returns its contents

in the editor textarea. The same value is also interpolated into the page description without HTML escaping, allowing reflected same-origin markup and potentially script execution in a victim's authenticated Webmin session. The save endpoint's file-list membership check prevents this from becoming an arbitrary-file write, but it does not protect the view endpoint. The primary impact is disclosure of root-readable host data outside the authority intended by delegation of this storage-specific module.

Profile-based footprint: 83.29% observed module signal (5,627/6,756 usable profiles; observed gate: iscsi-tgtd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `iscsi-tgtd/edit_manual.cgi`, `iscsi-tgtd/save_manual.cgi`.

Module · LDAP Client (2)

WEBMIN-0132 — LDAP popup browser permits reflected and stored authenticated XSS

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

popup_browser.cgi places the request-derived LDAP base in a double-quoted JavaScript string inside a single-quoted onClick attribute. quote_escape converts a literal double quote to ", but the HTML parser converts that entity back to a quote before compiling the event handler, so an expression-bearing base can execute script. The same page also joins LDAP attribute values directly into HTML without html_escape. A crafted navigation can therefore execute reflected script for an authenticated LDAP Client administrator, while a directory principal able to control an attribute value can plant a stored trigger. Script runs in Webmin's origin with the victim's authenticated capabilities.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: ldap-client).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `ldap-client/popup_browser.cgi`.

WEBMIN-0350 — LDAP Client Norwegian invalid-host error permits authenticated reflected XSS

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Norwegian LDAP Client catalog supplies server_ehost with a raw $1 placeholder. save_server.cgi splits the request-controlled host value, and on failed address resolution passes the token directly through text('server_ehost', $h) to error(). The web error renderer emits the resulting string as HTML without escaping it. An authenticated request containing an SVG/onload payload therefore returns active markup in the Webmin origin. Script execution can exercise the viewing session's Webmin privileges; a root Webmin victim gives root-equivalent control-plane impact. The reproduced path required an authenticated, referer-accepted request, and direct OS command execution was not demonstrated.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: ldap-client).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `ldap-client/save_server.cgi`, `ldap-client/lang/no`.

Module · LDAP Server (3)

WEBMIN-0102 — LDAP-controlled content can execute as Webmin-origin markup

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The LDAP browser treats table cells and status-message arguments as HTML but supplies LDAP attribute names/values and search text without output encoding. An actor able to influence directory content can place active markup in an attribute that executes when an authorized Webmin user browses the entry; crafted search

input provides a reflected variant. The latest-stable container test emitted an event-handler payload unchanged inside a rendered table cell, and the frozen default CSP permits inline script execution.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: The attacker controls an LDAP attribute visible to the Webmin LDAP browser. [environmental]

Audited source: `ldap-server/edit_browser.cgi`, `ui-lib.pl`.

WEBMIN-0237 — Stored XSS in LDAP ACL selector can compromise a privileged Webmin session

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

An authenticated delegated LDAP ACL editor can save a DN selector containing HTML/SVG markup because acl_save.cgi only applies a non-whitespace DN-shaped regex. parse_ldap_access embeds the saved selector in literal tt markup, and edit_acl.cgi passes that value through ui_link and the columns renderer without HTML escaping. When another administrator visits the LDAP ACL list, the markup executes as stored JavaScript in Webmin's authenticated origin. If the viewer is a root Webmin administrator, the script can exercise that root session's same-origin privileges. The path requires LDAP ACL-edit permission and a later view by a more-privileged Webmin user.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `ldap-server/acl_save.cgi`, `ldap-server/ldap-server-lib.pl`.

WEBMIN-0376 — LDAP Server browser search reflected XSS in Norwegian response

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

ldap-server/edit_browser.cgi reads the search request parameter and, when a directory search has no matching child objects, passes it to the localized browser_nomatch message as literal ... content without html_escape. The Norwegian catalog added in the reviewed commit retains that $1 placeholder. A crafted search value can therefore break into active HTML in the authenticated Webmin origin. A victim with an authenticated Webmin session and LDAP Server browser access can have same-origin script run with that session's administrative reach.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `ldap-server/edit_browser.cgi`, `ldap-server/lang/no`.

Module · LDAP Users and Groups (2)

WEBMIN-0015 — Raw LDAP attribute values render as active HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The raw LDAP attributes page inserts attribute names and values into ui_columns_table without HTML encoding. The table renderer treats ordinary string cells as markup. An LDAP-controlled value can therefore supply an element with an event handler that executes in the authenticated Webmin origin when an operator views raw attributes. The latest-stable image emitted the synthetic img/onerror payload unchanged.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: ldap-useradmin/raw.cgi, ui-lib.pl.

WEBMIN-0503 — Stored XSS through LDAP sambaAcctFlags in the user editor

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The LDAP user editor obtains sambaAcctFlags from the directory and passes it directly to ui_table_row. That helper accepts HTML and places the value verbatim in a table cell. A principal able to modify this attribute can persist active markup that executes when a suitably authorized Webmin operator opens the LDAP user's editor. The path requires LDAP attribute-write influence and a subsequent operator visit to the user editor.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: The attacker can change the LDAP user's sambaAcctFlags attribute. [environmental]

Audited source: ldap-useradmin/edit_user.cgi, ui-lib.pl.

Module · Logrotate (1)

WEBMIN-1023 — Global Force Log Rotation renders logrotate diagnostics as active HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The global Force Log Rotation action inserts logrotate's combined output directly inside an HTML `pre` element. A local actor able to influence a wildcard-rotation filename can cause an error diagnostic to include active markup; when a privileged operator runs the force action, script executes in that Webmin session. The per-entry Rotate Now response encodes equivalent output, while the global response does not.

Profile-based footprint: 5.34% observed module signal (361/6,756 usable profiles; observed gate: logrotate).

Recorded conditions: The attacker controls a filename under a directory covered by a privileged wildcard rotation rule. [environmental]

Audited source: logrotate/force.cgi, logrotate/save_log.cgi.

Consolidated reproduction records: `webmin-0154` (Logrotate error output permits privileged-session XSS through attacker-controlled filenames).

Module · Logviewer (1)

WEBMIN-0770 — Reflected XSS through the logviewer linktitle parameter

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

Log Viewer passes request-controlled `linktitle` directly to the page header, which emits it as markup under a policy permitting inline script. A crafted authorized request returned an SVG event-handler marker in both header contexts in the frozen runtime. Externally initiated delivery must satisfy Webmin's referer check.

Profile-based footprint: 85.24% observed module signal (5,759/6,756 usable profiles; observed gate: logviewer).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: logviewer/view_log.cgi, web-lib-funcs.pl.

Consolidated reproduction records: `webmin-0763` (Reflected script injection through the log-view linktitle parameter).

Module · Lpadmin (2)

WEBMIN-0098 — Stored XSS through an unescaped print-job filename

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The Print Jobs page places the queue-provided filename field into a table cell without HTML encoding. A print submitter can supply an active HTML element such as an SVG onload handler. When a privileged Webmin operator views the queue, the handler executes under Webmin's origin; Webmin 2.670's response policy explicitly allows inline script, so the browser-side effect can use the operator session. This requires a privileged viewer and queue metadata influence and is not unauthenticated direct command execution.

Profile-based footprint: 4.14% observed module signal (280/6,756 usable profiles; observed gate: lpadmin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `lpadmin/list_jobs.cgi`, `lpadmin/cups-lib.pl`.

WEBMIN-2405 — Referrer-gated active-HTML injection in lpadmin print-jobs page

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

In frozen Webmin 2.670, lpadmin/list_jobs.cgi inserts the request-supplied printer name into the page title and empty-queue message as HTML without escaping. An authenticated same-origin request returned the supplied SVG onload element twice as active markup, and the response CSP permits inline script. MiniServ's default referrer defense blocked the external/no-referrer control, so exploitation requires a request that passes that gate or a relaxed referrer configuration.

Profile-based footprint: 4.14% observed module signal (280/6,756 usable profiles; observed gate: lpadmin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Module · Lvm (2)

WEBMIN-0216 — LVM size-unit value injects active HTML into resize confirmation

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

save_lv.cgi numerically consumes size_units but retains the original request string, then a resize-confirmation branch serializes every request parameter into a single-quoted hidden input without HTML encoding. A value beginning with a valid number can therefore close the attribute and add an event-bearing SVG element. The endpoint requires an authenticated Webmin context with LVM access and a valid resize-confirmation path; inducing a root Webmin administrator to issue the crafted request gives script execution in that privileged origin and root-equivalent session impact.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: lvm).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `lvm/save_lv.cgi`.

WEBMIN-0298 — LVM size-mode error reflects active HTML in authenticated Webmin origin

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

save_lv.cgi selects size modes with numeric comparisons but, for an unrecognized numeric prefix, concatenates the original size_mode request value into error() without HTML encoding. A value such as 9 followed by an event-bearing SVG reaches the fallback and is emitted as live markup. The endpoint requires an authenticated Webmin context and a valid volume group; inducing a root Webmin administrator to issue the crafted request yields script execution in that privileged origin and therefore root-equivalent Webmin session impact.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: lvm).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `lvm/save_lv.cgi`, `web-lib-funcs.pl`.

Module · Mailboxes (4)

WEBMIN-0034 — HTML-mail sanitizer preserves entity-obfuscated JavaScript iframe URLs

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The HTML-mail sanitizer filters literal javascript text before applying a URL policy that returns every unmatched non-CGI scheme unchanged. An attacker can interrupt the scheme with an HTML character reference, such as javascript:, so the literal filter misses it and safe_url preserves it in an iframe src. The browser's HTML parser decodes the reference in the attribute, leaving an executable JavaScript URI in attacker-controlled mail rendered under the authenticated application origin. Frozen stable 2.670 retains the source behavior, and the isolated lab reproduced the sanitizer bypass.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mailboxes/folders-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0336 — Unescaped dom parameter permits authenticated reflected XSS in mailbox return links

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Mailbox navigation accepts dom from the request and user_list_link() concatenates it into a return URL without URL or HTML-attribute encoding. The core footer places that URL directly inside double-quoted href attributes. A quote-bearing dom value therefore terminates the original attribute and adds attacker-controlled event-handler attributes. The frozen stable default CSP includes 'unsafe-inline', and the isolated latest-stable reproduction emitted an autofocus/onfocus handler from a harmless proof payload. Exploitation requires an authenticated mailbox-capable victim to open a crafted URL, but resulting script runs in the Webmin origin and can act with that session's privileges.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mailboxes/mailboxes-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0656 — Authenticated Mailboxes response XSS persists through unescaped footer URL rendering

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

mailboxes/delete_mail.cgi accepts the request-controlled folder value, uses its numeric prefix to select a folder, and later concatenates the unchanged string into the URL passed to ui_print_footer. The core and bundled Gray/Blue footer implementations write that URL directly inside a double-quoted href. A value beginning with a valid numeric index can close the attribute and inject executable markup. The issue requires an authenticated Mailboxes-capable victim and a selectable message, but JavaScript then runs with the Webmin origin and the victim's administrative authority. The frozen 2.670 container reproduced the injection using a harmless SVG onload marker.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: The authenticated mailbox user has a selectable message for the delete action. [environmental]

Audited source: `mailboxes/delete_mail.cgi`, `gray-theme/theme.pl`.

WEBMIN-0870 — Unescaped iCalendar metadata bypasses plain-mail markup and image handling

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Calendar MIME fields are parsed from message-controlled data and placed into get_calendar_data output without HTML encoding. In the plain-message display branch, the raw calendar text is inserted after an opening pre tag, while only the ordinary message body is escaped; the image-disabling rewrite runs solely in the HTML-message branch. A calendar SUMMARY can therefore close the pre block and add elements such as images, forms, or links to iframe srcdoc. The iframe does not allow scripts, which limits code execution, but it permits popups and still loads ordinary image URLs. The latest-stable reproduction confirmed a safe loopback marker with image and form markup survives this path.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mailboxes/folders-lib.pl`, `mailboxes/view_mail.cgi`.

Module · Minecraft (3)

WEBMIN-0367 — Action-message parameters render as active HTML

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The player and world-command pages place the request-supplied msg value directly into an HTML status fragment. A logged-in Minecraft module user can be made to render active markup when the request satisfies Webmin's referrer protection. The frozen lab returned the SVG test payload verbatim; a direct cross-origin request received Webmin's security warning.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `minecraft/view_conn.cgi`, `minecraft/edit_cmds.cgi`.

WEBMIN-0702 — Stored cross-site scripting through Minecraft world names in manual backup output

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The immediate-backup handler writes every backup status line directly into an HTML response. Backup status text includes world names taken from directories beneath the configured Minecraft server path. A less-privileged account able to create a world directory with a markup-bearing name can therefore cause script-capable HTML to be returned when a Webmin Minecraft-module operator performs Save and Backup Now. The Docker reproduction exercised the authenticated endpoint and observed the raw payload twice with no escaped form. The path requires attacker filesystem access and a victim performing the backup action; successful execution occurs in the privileged Webmin origin.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: The attacker can create a markup-bearing world directory beneath the configured Minecraft server path. [environmental]; An authorized operator performs Save and Backup Now. [environmental]

Audited source: `minecraft/save_backup.cgi`, `minecraft/minecraft-lib.pl`.

WEBMIN-0740 — Persistent JAR filename markup enables stored XSS in the versions table

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The installed-version list derives file from directory entries and passes it directly to ui_radio_columns_row. The renderer inserts column content without HTML encoding, while the upload handler permits arbitrary basename characters before a numeric .jar suffix. Consequently a persistent JAR filename can inject elements and event-handler attributes into the authenticated versions page. The stable lab used a harmless void(0) handler and observed the marker literally in row, input, label, and cell contexts without executing browser code.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `minecraft/add_version.cgi`, `minecraft/list_versions.cgi`.

Module · Mount (1)

WEBMIN-0388 — Attacker-controlled NFS export name executes in the mount chooser

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Filesystem Mounts chooser accepts non-whitespace export names from a remote `showmount` response and inserts them as link text and into inline JavaScript without encoding. A malicious or impersonated NFS server can return a quote-bearing export name that executes script when an authenticated user opens or selects the entry. The frozen 2.670 reproduction retained the executable handler shape.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: The attacker controls or impersonates the NFS server response. [environmental]

Audited source: `mount/nfs_export.cgi`, `mount/linux-lib.pl`.

Consolidated reproduction records: `webmin-0038` (Attacker-controlled NFS export names execute in the Webmin origin).

Module · MySQL (4)

WEBMIN-0048 — Stored XSS through unencoded database column names in table views

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Frozen Webmin 2.670 derives table column names from database metadata and passes them to column_sort_link, which inserts each name unencoded into both a single-quoted href attribute and an HTML element. A database principal able to create a crafted column can store executable markup that runs when a higher-privilege Webmin operator views table rows. The isolated latest-stable lab reproduced the exact response with an SVG onload handler crossing the sortfield attribute boundary.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mysql/mysql-lib.pl`, `mysql/view_table.cgi`.

WEBMIN-0497 — MySQL connection diagnostics can inject markup into the Webmin origin

CVSS 4.0 Base (CVSS-B): 2.3 (Low) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The MySQL status routine returns DBI exception text or mysqladmin output as a diagnostic. The module index wraps that value in a tt element and supplies it to ui_details with HTML rendering enabled, without encoding the diagnostic itself. A party able to influence a failed MySQL connection response or the configured client output can inject markup or script into a page viewed by an authenticated Webmin user, allowing actions in the Webmin origin with the viewer's session authority.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: The configured MySQL server or client diagnostic output is attacker-influenced during a failed connection. [nondefault]; An authenticated Webmin operator opens the MySQL index while that diagnostic is displayed. [environmental]

Audited source: `mysql/mysql-lib.pl`, `mysql/index.cgi`.

WEBMIN-0628 — Persistent MySQL configuration path is emitted as active HTML

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The MySQL module's missing-my.cnf branch inserts the persistent config.my_cnf value into an HTML-bearing translation argument without encoding. The setting is a free-text module configuration value, and Webmin's text() helper performs literal placeholder substitution. An authenticated user with delegated module-configuration access can therefore store markup in an unreadable local path; when another Webmin user opens the module, the markup is emitted in the same-origin administrative response and can execute script with the visitor's privileges.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mysql/index.cgi`, `mysql/config.info`.

WEBMIN-0665 — Remote MySQL metadata is rendered as active Webmin HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The MySQL module treats metadata returned by a configured database endpoint as display content. The remote version value is passed through translation substitution and into ui_print_header's HTML-capable right-side argument without HTML encoding. A controlled endpoint can return active markup that runs in the Webmin origin when an administrator opens the module. The same trust-boundary class was also observed historically in character-set option labels and is merged into this finding.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: The attacker controls metadata returned by the configured MySQL endpoint. [environmental]

Audited source: `mysql/mysql-lib.pl`, `mysql/index.cgi`.

Module · Net (1)

WEBMIN-2443 — Netplan route validation reflects raw field content into HTML errors

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Netplan static-route save path validates each submitted network, mask, and gateway but interpolates an invalid value into the fatal HTML error without output encoding. Webmin's error renderer accepts caller-supplied HTML, so the field becomes active markup in the authenticated administrative origin. The caller needs route-write permission and an accepted request path. The Docker test showed literal HTML injection with a non-executing SVG marker.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `net/netplan-lib.pl`, `web-lib-funcs.pl`.

Module · Openslp (1)

WEBMIN-0352 — OpenSLP configuration paths produce stored XSS in Norwegian error views

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

OpenSLP index.cgi inserts stored slpd and slpd_conf configuration values into the Norwegian index_eslpd and index_econf HTML messages without output encoding. A module configuration editor can persist active markup that is returned to authenticated viewers when the selected executable or configuration file is missing. This audit makes no claim about the exact introduction or fix commit.

Profile-based footprint: 85.08% observed module signal (5,748/6,756 usable profiles; observed gate: openslp).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `openslp/index.cgi`, `openslp/lang/no`.

Module · Package Updates (3)

WEBMIN-0199 — Repository package metadata renders as active HTML across Package Updates views

CVSS 4.0 Base (CVSS-B) · path `xss:module:package-updates:repository-metadata-list-results`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:package-updates:repository-description-detail`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Package Updates passes repository-derived package names, descriptions, versions, and sources to HTML-capable helpers in lists, details, and dependency previews. A malicious configured repository can place script in the package cache; an authenticated administrator viewing the package can execute it in the Webmin origin before installing anything.

Profile-based footprint: 95.00% observed module signal (6,418/6,756 usable profiles; observed gate: package-updates).

Recorded conditions: The attacker controls or compromises a configured package repository or its metadata feed. [environmental]

Audited source: `package-updates/index.cgi`, `package-updates/update.cgi`.

Consolidated reproduction records: `webmin-0489` (Package Updates renders repository descriptions as unescaped HTML).

WEBMIN-0598 — Authenticated reflected XSS through package-update redir

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

package-updates/update.cgi accepts redir from an authenticated request and later passes it unchanged to ui_print_footer. The default and Gray-theme footer implementations place that value directly inside double-quoted href attributes. A redir beginning with a quote can close the attribute and inject active HTML; the 2.670 runtime emitted an SVG onload payload twice in the footer. Webmin's response CSP permits inline script execution. Successful exploitation requires an authenticated, package-updates-authorized victim request that passes Webmin's referrer/origin protection, but execution occurs in the sensitive Webmin origin with the victim session's permissions.

Profile-based footprint: 95.00% observed module signal (6,418/6,756 usable profiles; observed gate: package-updates).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `package-updates/update.cgi`, `web-lib-funcs.pl`.

WEBMIN-0649 — Package repository URLs render as stored active HTML

CVSS 4.0 Base (CVSS-B) · path `xss:module:package-updates:apt-repository-url`: 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:package-updates:yum-repository-url`: 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Package Updates accepts HTML-bearing APT or YUM repository URLs and later passes the stored URL to a table cell without encoding, while separately escaping the repository name. A user able to manage repositories can plant script that runs when an authenticated operator opens the repository tab. The APT path returned an SVG event-handler marker in the frozen runtime.

Profile-based footprint: 95.00% observed module signal (6,418/6,756 usable profiles; observed gate: package-updates).

Recorded conditions: The repository-URL path uses the APT package backend. [environmental]; A module user can save the crafted repository URL. [environmental]; A later authenticated Webmin user opens the repository list. [environmental]; The repository-URL path uses the YUM package backend. [environmental]

Audited source: `software/yum-lib.pl`, `software/apt-lib.pl`.

Consolidated reproduction records: `webmin-0884` (Stored cross-site scripting through an APT repository URL).

Module · Pam (2)

WEBMIN-0386 — PAM module name is rendered as active HTML

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

In the new-module path, edit_mod.cgi takes module from the request and prints it inside the module-name table row without html_escape. An authenticated PAM-module page can therefore reflect attacker-supplied markup into the Webmin origin. A benign SVG element remained unencoded in the frozen 2.670 response, confirming browser-active HTML injection. Exploitation requires an authenticated, PAM-authorized victim to open a crafted URL.

Profile-based footprint: 86.04% observed module signal (5,813/6,756 usable profiles; observed gate: pam).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `pam/edit_mod.cgi`.

WEBMIN-0889 — Stored HTML injection remains on PAM edit pages

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

At frozen newest-stable 2.670, the PAM parser still reads service descriptions from configuration-file comments, and edit_pam.cgi, edit_inc.cgi, and edit_mod.cgi insert those descriptions into HTML table values without html_escape. The UI helper treats its value argument as HTML. An isolated latest-stable container preserved a crafted img element and event handler in the rendered row.

Profile-based footprint: 86.04% observed module signal (5,813/6,756 usable profiles; observed gate: pam).

Recorded conditions: The attacker controls a PAM service-file description on the managed host. [environmental]

Module · PHP Configuration (1)

WEBMIN-0590 — Authenticated reflected XSS in PHP Configuration edit-page file header

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

PHP Configuration edit pages interpolate the request-controlled file parameter directly into an HTML header. In an authenticated request accepted by Webmin's referer protection and module ACL, active markup executes in Webmin's privileged origin. The latest stable 2.670 source retains the sink, and an isolated latest-stable container returned a crafted SVG/onload element verbatim. Exploitation requires user interaction with a crafted edit URL and an accepted authenticated navigation/referer context; successful script execution can perform Webmin operations with the victim's session authority.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `phpini/edit_misc.cgi`, `phpini/phpini-lib.pl`.

Module · Postfix (1)

WEBMIN-0273 — Queued-message attachment filename yields stored Webmin-origin XSS

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

A remote sender can place a single quote and HTML event attributes in a MIME attachment filename. When an authorized Webmin user opens the retained message in the Postfix mail queue, Webmin inserts the filename into both the attachment URL and link text without attribute or text escaping. Script then runs in Webmin's authenticated origin. The path requires queue retention and victim navigation; an affected administrator session can have root-equivalent authority.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mailboxes/boxes-lib.pl`, `postfix/view_mailq.cgi`.

Module · Pptp Client (1)

WEBMIN-0618 — Stored XSS through PPTP tunnel names in the Webmin module index

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

PPTP Client accepts any nonblank tunnel name and persists it as a peer filename. The module index recovers that filename and supplies it to icons_table as both a link fragment and title. The default icon renderer inserts the title as raw link-body HTML and leaves the link unencoded, allowing a module-authorized account to persist active markup that executes for a later Webmin viewer. Script in Webmin's authenticated origin can perform actions with the victim's Webmin privileges; the finding is browser-side rather than direct shell execution.

Profile-based footprint: 0.00% observed (0/6,756 usable profiles; gate: pptp-client). No numeric deployment bound follows from a negative asset probe.

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Module · Proc (2)

WEBMIN-0359 — Local process arguments render as active HTML in the authenticated process UI

CVSS 4.0 Base (CVSS-B): 2.4 (Low) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Webmin 2.670 obtains process command arguments from ps without HTML encoding. On the Process Manager detail page, a parent process's argument text is used directly as ui_link text and each subprocess's argument text is used directly as a ui_columns_row cell; both helpers concatenate caller-provided content into HTML.

An unprivileged local user can therefore place active markup in the argv of a running process and trigger same-origin script execution when a more privileged authenticated Webmin user views a related process detail page. Browser code running in the victim's Webmin origin can act with that victim's authority.

Profile-based footprint: 20.43% observed module signal (1,380/6,756 usable profiles; observed gate: proc).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `proc/linux-lib.pl`, `proc/edit_proc.cgi`.

WEBMIN-0861 — Process trace renders process-controlled command and syscall text as active HTML

CVSS 4.0 Base (CVSS-B) · path `xss:module:proc:syscall-arguments`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:proc:process-command-text`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Process Manager inserts process command text and `strace` string arguments into the trace page without HTML encoding. A local user can place markup in a command or syscall argument; a later authorized trace view interprets it in the operator's Webmin origin. The frozen parser and renderer retained an SVG marker verbatim.

Profile-based footprint: 20.43% observed module signal (1,380/6,756 usable profiles; observed gate: proc).

Recorded conditions: The attacker controls text emitted by a process selected for tracing. [environmental]

Audited source: `proc/trace.cgi`, `proc/linux-lib.pl`.

Consolidated reproduction records: `webmin-0512` (Process trace syscall arguments execute as HTML in the Webmin origin).

Module · Qmailadmin (1)

WEBMIN-0301 — Qmail queue refresh reflects raw request data into a form action

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

qmailadmin/list_queue.cgi appends scalar $in, which ReadParse retains as the raw POST body/query string, to the queue refresh target. ui_buttons_row places that target inside a single-quoted form action without HTML attribute escaping. A raw request body containing an apostrophe followed by autofocus/onfocus attributes therefore breaks out of the action value and executes script when a non-empty queue causes the refresh form to render. The endpoint requires Qmail Administration access and victim interaction; script running in a root Webmin administrator's origin is root-equivalent, though this is not direct command execution.

Profile-based footprint: 85.04% observed module signal (5,745/6,756 usable profiles; observed gate: qmailadmin).

Recorded conditions: The Qmail queue is non-empty so the refresh form renders. [environmental]

Audited source: `qmailadmin/list_queue.cgi`, `web-lib-funcs.pl`.

Module · Quota (1)

WEBMIN-0488 — NSS-controlled XFS quota owner names are rendered as active HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Numeric XFS quota owner tokens are expanded with getpwuid/getgrgid to full NSS-controlled user or group names. The quota list passes those names as raw table-cell or link text, and ui_link does not HTML-encode its text argument. A directory-backed or otherwise attacker-influenced NSS name containing markup can therefore become stored same-origin script when a privileged quota operator views the affected XFS quota page. Script in a Webmin administrative origin can perform actions with the victim's privileges, which may include root-level command execution. Exploitation requires control of a matching NSS identity and quota record plus a privileged

page view; possession of an LDAP directory credential alone does not directly execute server commands through this code path.

Profile-based footprint: 5.43% observed module signal (367/6,756 usable profiles; observed gate: quota).

Recorded conditions: The attacker controls a matching NSS identity name and a corresponding XFS quota record. [environmental]

Audited source: quota/linux-lib.pl.

Module · Raid (1)

WEBMIN-0159 — Spare-group validation bypass permits root configuration injection and stored XSS

CVSS 4.0 Base (CVSS-B) · path `xss:module:raid:default`: 7.0 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:H/VA:L/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `privilege-escalation:module:raid:default`: 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

Spare-group creation validates only the new-group branch. A direct group_mode=1 request can supply an arbitrary group value that is concatenated into the root-owned mdadm.conf ARRAY line. Embedded newlines can add directives such as PROGRAM, creating a root configuration-integrity violation and potential event-triggered execution when the named executable and monitor conditions exist. The same persisted group is parsed and rendered raw in view_raid.cgi, allowing stored active HTML under the Webmin origin. Both flows remain in 2.670; isolated reproduction showed exact stable serialization of an injected PROGRAM line and raw rendering of an img/onerror payload.

Profile-based footprint: 85.98% observed module signal (5,809/6,756 usable profiles; observed gate: raid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: raid/create_raid.cgi, raid/raid-lib.pl.

Module · Samba (1)

WEBMIN-0309 — Raw query string permits attribute injection in the Samba connections refresh link

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

view_users.cgi constructs its refresh URL from scalar $in, which ReadParse retains as the original query string. That URL is inserted into a single-quoted href without HTML-attribute escaping. A crafted request containing a literal single quote can terminate href and add an event attribute, permitting reflected script execution in the authenticated Webmin origin when a permitted user visits the URL.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: samba/view_users.cgi, ui-lib.pl.

Module · Sarg (1)

WEBMIN-0198 — SARG report directory names execute as stored markup in Webmin

CVSS 4.0 Base (CVSS-B): 2.4 (Low) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The SARG report viewer's directory-listing branch inserts each filesystem entry name into a single-quoted href and into the HTML body without attribute or text escaping. A crafted entry beneath the configured SARG output directory is therefore served as active markup when an authenticated SARG user browses a directory without index.html. In Webmin 2.670, a harmless SVG onload marker was returned verbatim under the Webmin

origin. The path requires attacker influence over a report entry name and an authenticated viewer who opens the report.

Profile-based footprint: 85.20% observed module signal (5,756/6,756 usable profiles; observed gate: sarg).

Recorded conditions: The attacker can create or rename an entry beneath the configured SARG output directory. [environmental]; The viewed directory has no index.html, selecting the directory-listing branch. [environmental]

Audited source: `sarg/view.cgi`.

Module · Servers (6)

WEBMIN-0088 — Remote linked server can execute script in the local Webmin origin

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

link.cgi authenticates to a registered remote server, then emits the remote response headers and HTML as a response from the local Webmin host. Link rewriting does not remove scripts or otherwise isolate active content. A compromised registered server can therefore return JavaScript that runs with the local Webmin origin when a local administrator follows the server link. The script can read local-origin responses and issue authenticated administrative requests with the victim's Webmin permissions. This is a high-impact lateral administrative compromise path, although the batch did not demonstrate a direct operating-system command execution primitive or classify it as critical.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: The exploit path requires an already configured linked remote server. [nondefault]

Audited source: `servers/link.cgi`.

WEBMIN-0090 — Unauthenticated discovery response can plant stored script in server editor

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The server-discovery code accepts colon-separated UDP responses and stores the fourth field as realhost without validation. edit_serv.cgi later interpolates that stored realhost directly into an HTML <tt> element. A network-adjacent attacker who answers a discovery probe can plant script-capable markup that executes when an administrator opens the discovered server record. The resulting same-origin script can act with the administrator's Webmin privileges. The frozen stable source and marker-only Docker reproduction confirm the data sink remains present; no direct command execution was attempted or demonstrated.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `servers/servers-lib.pl`, `servers/edit_serv.cgi`.

WEBMIN-0436 — Servers delegate can store active HTML in privileged server-list links

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Servers save path accepts a host containing quotes because it rejects only whitespace. The server-list URL builder then places that stored host into an href without attribute-context escaping. Webmin 2.670 reproduced the issue for a Servers-only delegate: a root viewer's index contained an onmouseover attribute supplied by the delegate. This is root-equivalent through a root Webmin victim, but it is not direct unauthenticated operating-system command execution.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: servers/save_serv.cgi, servers/index.cgi.

WEBMIN-0474 — Delegated linked-server hostname becomes stored XSS in duplicate discovery output

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The hostname-based duplicate branch in find_servers compares an untrusted UDP response hostname with configured linked-server hostnames, constructs a URL from that value, and interpolates the URL into the Find Servers HTML inside a tt element without escaping. The linked-server save path accepts any non-whitespace hostname when no remote login mode is selected. A delegated Servers user can therefore store an active-tag hostname and arrange for a matching discovery response; when a privileged user runs discovery, the tag executes in the authenticated Webmin origin. With a root Webmin victim the browser impact is root-equivalent, but the sink does not itself execute an operating-system command.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: The attacker can supply a matching server-discovery UDP response. [environmental]; A privileged Webmin user later runs linked-server discovery. [environmental]; A linked-server record contains the attacker-selected hostname that matches the discovery response. [environmental]

Audited source: servers/save_serv.cgi, servers/servers-lib.pl.

WEBMIN-0577 — Linked-server TLS peer authentication failure exposes administrative credentials and same-origin active content

CVSS 4.0 Base (CVSS-B): 6.0 (Medium) · CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:N/VA:N/SC:L/SI:L/SA:N.

The linked-server proxy treats a false or missing checkssl value as nocheckhost, then transmits fixed, current-user, or prompted HTTP Basic credentials and relays the peer's HTML through the local Webmin route. New server records omit checkssl, so the control is initially unchecked. An attacker able to answer the outbound connection can therefore receive linked-server credentials and return active HTML in the local Webmin origin. In stable 2.670, enabling the control rejects a certificate-name mismatch but still supplies no CA trust or self-signed rejection requirement, so a self-signed certificate with the expected name is accepted. Intercepted root/admin credentials or same-origin actions in a privileged local session can be root-equivalent. The path requires linked-server use and network, DNS, routing, or endpoint substitution; direct unauthenticated server-side execution was not demonstrated.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: The attacker can substitute the linked-server endpoint or intercept its outbound connection. [environmental]; A Webmin user invokes the linked-server proxy route. [environmental]

Audited source: servers/edit_serv.cgi, servers/save_serv.cgi.

WEBMIN-0773 — Unvalidated server fields permit stored attribute injection in Servers Index

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The Servers module persists type verbatim and validates host only as a non-whitespace token. Icon mode interpolates type into an image URL that generate_icon writes directly to a double-quoted src attribute; ordinary link mode interpolates host into a URL that ui_link or generate_icon writes without attribute encoding. An edit/add-capable module user can therefore persist an event-handler or other injected attribute that executes when another authorized user views the record. The frozen stable Docker reproduction exercised the real save_serv.cgi and Servers Index flow and observed the stored type rendered as an onerror attribute.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: servers/save_serv.cgi, servers/index.cgi.

Module · Shorewall (4)

WEBMIN-0073 — Shorewall zone comments permit stored cross-site scripting

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The Shorewall zones editor accepts a comment, stores it after a hash marker, and the zones parser later recovers it. zones_row returns that value as a displayed cell; list.cgi passes non-leading cells directly to ui_checked_columns_row, which treats cell values as HTML. An authenticated user delegated zones-file access can therefore store browser markup that executes when another Webmin user views the zones list. Frozen Webmin 2.670 retains the complete flow, and the latest-stable-only lab rendered an inert SVG event marker as literal markup.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: shorewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: shorewall/shorewall-lib.pl, shorewall/list.cgi.

WEBMIN-0240 — Stored cross-site scripting through the Shorewall TOS mark column

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

An authenticated user who can edit Shorewall TOS entries can submit a whitespace-free HTML or SVG value in mark. tos_validate returns the value unchanged, save.cgi persists it, tos_row returns it unchanged, and list.cgi passes it to ui_checked_columns_row, which concatenates cell values as HTML. A more privileged user who opens the TOS list therefore renders attacker markup in the Webmin origin. Inline event-handler payloads are viable under the emitted CSP, permitting same-origin actions with the viewer's Webmin privileges. The reproduction used inert SVG markup and did not execute JavaScript or operating-system commands.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: shorewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: shorewall/shorewall-lib.pl, shorewall/save.cgi.

WEBMIN-0344 — Stored XSS in Shorewall routing-rule source and destination fields

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The Shorewall routing-rule validator treats any non-whitespace source or destination value as valid and the row formatter returns those values unchanged. The generic Shorewall list page embeds the first value as raw link content and later values as raw cell content. A Webmin user delegated access to the route_rules file can therefore persist an inline script in a root-owned Shorewall configuration file. When another Webmin user, including a full administrator, opens the routing-rule list, the script runs in Webmin's origin; the latest-stable response policy permits inline script. The path requires an authenticated delegated module user and a separate privileged viewer of the poisoned list.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: shorewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: shorewall/shorewall-lib.pl, shorewall/list.cgi.

WEBMIN-0524 — Authenticated stored XSS in Shorewall table comments

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Shorewall's comment-save handler accepts arbitrary non-empty comment text and persists it in the selected table. The list handler recognizes COMMENT and ?COMMENT records, then joins the stored tokens into an italicized link body without HTML escaping. A delegated Shorewall user can therefore persist markup with an event handler that executes for a later viewer in the Webmin origin. The latest-stable Docker run reproduced raw SVG event-handler markup in the authenticated response. This is not unauthenticated or direct operating-system command execution, but a privileged Webmin victim can make the browser-mediated impact root-equivalent.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: shorewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `shorewall/savecmt.cgi`, `shorewall/list.cgi`.

Module · Shorewall6 (1)

WEBMIN-0548 — Authenticated stored XSS in Shorewall6 comments

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

A Webmin user authorized to edit Shorewall6 rules or traffic-control rules can store HTML in a comment. Webmin 2.670 persists the submitted value in a ?COMMENT record and later inserts it into the table page without HTML encoding. JavaScript therefore executes when a privileged Webmin user views the affected table. With a root-equivalent Webmin victim this is root-equivalent browser-context compromise, but it is not direct unauthenticated operating-system command execution.

Profile-based footprint: 84.72% observed module signal (5,724/6,756 usable profiles; observed gate: shorewall6).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `shorewall6/savecmt.cgi`, `shorewall6/list.cgi`.

Module · Smart Status (1)

WEBMIN-0169 — SMART device output renders as active HTML in privileged Webmin pages

CVSS 4.0 Base (CVSS-B) · path `xss:module:smart-status:device-identity`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:smart-status:attribute-rows`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:smart-status:self-test-output`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

SMART Drive Status passes smartctl device identity, parsed attributes, and self-test output into HTML-capable table or preformatted renderers without encoding. An actor able to influence attached, passthrough, or network-backed device output can plant markup that executes when an authorized Webmin user views status or starts a test. The frozen lab retained an SVG event-handler marker from the device-model field.

Profile-based footprint: 4.23% observed module signal (286/6,756 usable profiles; observed gate: smart-status).

Recorded conditions: The attacker controls an attached, passthrough, virtual, or network-backed device's smartctl output. [environmental]

Audited source: `smart-status/smart-status-lib.pl`, `smart-status/index.cgi`.

Consolidated reproduction records: `webmin-0342` (SMART device identity metadata is rendered as active HTML in the privileged status page); `webmin-0962` (Device-supplied SMART attributes render as unescaped HTML).

Module · Software (4)

WEBMIN-0097 — Plaintext rpmfind feed can execute script in a privileged Webmin session

CVSS 4.0 Base (CVSS-B): 2.3 (Low) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The software module downloads rpmfind.net search results over unauthenticated HTTP, parses a URL from the remote HTML, and embeds that value directly inside an onClick JavaScript string. A network-positioned attacker or compromised upstream can therefore return a crafted RPM URL whose quote characters become JavaScript syntax. The user must search and select the crafted result, after which the script runs in Webmin's authenticated origin and can exercise that user's administrative capabilities. This is not a direct unauthenticated or interaction-free operating-system RCE, but on the common deployment where Webmin runs as root, same-origin administrative actions can plausibly lead to root-level host compromise.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: The attacker can replace the unauthenticated HTTP rpmfind response or control that upstream. [environmental]; A Webmin user searches for and selects the crafted package result. [environmental]

WEBMIN-0202 — Stored cross-site scripting via repository package descriptions

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Webmin imports YUM Summary and APT package-description text from enabled repository metadata into the package `desc` field and renders it through HTML-capable helpers in availability, update, detail, and selection views. A malicious or compromised enabled repository can thereby run script in an administrator's Webmin session when that administrator opens a package view, before any package installation.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: The attacker controls metadata returned by an enabled APT or YUM repository. [environmental]; An enabled APT or YUM repository supplies the package description metadata. [environmental]; A privileged Webmin user opens the package-management view. [environmental]

Audited source: `software/yum-lib.pl`, `software/apt-lib.pl`.

WEBMIN-0617 — Unencoded IPKG filter and repository metadata execute as Webmin markup

CVSS 4.0 Base (CVSS-B) · path `xss:module:software:reflected-filter`: 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:software:repository-section-metadata`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Two IPKG values reach HTML without context-appropriate encoding. A request filter is interpolated into the package editor's footer href, and unmatched repository Section metadata becomes a category label printed directly in the tree. The first supports reflected script execution; the second supports repository-metadata script execution. Both execute in the Webmin origin and are represented as one output-encoding finding.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: The attacker controls metadata returned by a configured IPKG repository. [environmental]

Audited source: `software/ipkg-edit_pack.cgi`, `web-lib-funcs.pl`.

WEBMIN-0651 — Package return link permits reflected markup injection in the footer

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Software Packages handoff forwards caller-controlled return-link data to package-updates. The receiving page preserves the value and renders it through ui_print_footer, whose bundled footer implementation places the URL directly inside a double-quoted href. A return value containing a quote and markup can escape the attribute and create active same-origin content. The required victim interaction and module access reduce reach, but successful script execution can act with the authenticated victim's Webmin browser authority.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `software/install_pack.cgi`, `package-updates/update.cgi`.

Module · Spam (1)

WEBMIN-0373 — Norwegian SpamAssassin validation error permits authenticated reflected XSS

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The Norwegian SpamAssassin header_ename message interpolates an invalid message-test name without HTML encoding. save_simple.cgi passes the request-controlled name_0 value into that message, and Webmin's fatal-error renderer emits it as HTML. An authenticated crafted request can therefore execute script in the Webmin origin; with a root Webmin victim this is root-equivalent session compromise, but it is not direct server-side command execution.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: spam).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `spam/lang/no`, `spam/save_simple.cgi`.

Module · Status (3)

WEBMIN-0157 — Status monitor descriptions render externally controlled text as active HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Remote HTTP status lines, TLS FTP diagnostics, and wildcard-monitor filenames enter the shared monitor description field, which the status editor renders as HTML. An actor controlling a monitored endpoint response or a filename in a monitored writable directory can place active markup in a later authenticated editor view. The HTTP variant returned an unescaped `img/onerror` marker in the frozen Webmin 2.670 lab.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: A configured HTTP monitor reaches an attacker-controlled or compromised endpoint. [environmental]

Audited source: `status/http-monitor.pl`, `status/ftp-monitor.pl`.

Consolidated reproduction records: `webmin-0749` (HTTP monitor status-line XSS in authenticated edit view).

WEBMIN-0340 — Unescaped monitor-history inputs allow reflected and stored cross-site scripting

CVSS 4.0 Base (CVSS-B) · path `xss:module:status:crafted-history-url`: 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `xss:module:status:remote-monitor-history`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The status edit page requires monitor edit access, but it treats dynamic history content as trusted HTML. The page URL-decodes all and changes query values and inserts them directly into single-quoted history-link href attributes; an unauthenticated attacker can lure an authenticated status editor to a crafted URL that injects an event handler. The same renderer inserts persisted nice_value/value strings without HTML encoding. Because service_status accepts result hashes from configured remote Webmin monitors and monitor.pl stores those values unchanged, a compromised monitored endpoint can also persist active markup for a central operator's later view. Webmin's default response policy permits unsafe-inline script, and Webmin commonly runs with root-equivalent administrative authority, so origin script can perform privileged management actions with the victim's session. A valid monitor history entry is required for either rendering path.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: A valid monitor history entry exists for the rendered edit page. [environmental]; The attacker controls the result data returned by a configured remote monitor. [environmental]

Audited source: `status/edit_mon.cgi`, `web-lib-funcs.pl`.

WEBMIN-0510 — Status monitor group name stored XSS reaches the System Information dashboard

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

A user with delegated Status edit permission can submit a whitespace-free HTML payload in the groups parameter. save_mon.cgi persists that token without validation, nice_remotes substitutes it into the 'Members of $1' label without HTML escaping, and the System Information block inserts the resulting string as HTML. When the block is enabled and a root/admin viewer opens the dashboard, attacker-controlled active content runs in the Webmin origin and can perform same-origin privileged actions.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: The System Information members block is enabled on the dashboard. [nondefault]

Audited source: `status/save_mon.cgi`, `status/status-lib.pl`.

Module · Stunnel (1)

WEBMIN-0372 — Delegated Stunnel configuration permits stored XSS in Norwegian error page

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

An authenticated user allowed to edit Stunnel module configuration can persist HTML in stunnel_path. When the non-executable path is viewed under the Norwegian locale, index.cgi supplies that value as $1 and the localized index_estunnel string inserts $1 into both text and href without escaping. This produces stored cross-site scripting in the authenticated Webmin origin; exploitation requires delegated Stunnel access with configuration editing allowed and a Norwegian-locale view.

Profile-based footprint: 4.06% observed module signal (274/6,756 usable profiles; observed gate: stunnel).

Recorded conditions: The reproduced sink requires a non-default locale/language selection. [nondefault]

Audited source: `stunnel/lang/no`, `stunnel/index.cgi`.

Module · Updown (2)

WEBMIN-0118 — Stored XSS through background-download HTTP errors

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

An attacker-controlled HTTP server can return an invalid status line containing markup. complete_http_download returns that entire line as an error, download.pl saves it in the background-download

record, and index.cgi places the stored error into the progress cell without output encoding. A delegated module user can therefore plant script that executes when a privileged operator views the module, inheriting the operator's authenticated Webmin origin and authority.

Profile-based footprint: 5.60% observed module signal (378/6,756 usable profiles; observed gate: updown).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `web-lib-funcs.pl`, `updown/download.pl`.

WEBMIN-0230 — Authenticated Up/down archive filename XSS in extraction progress

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The upload handler derives a destination path from multipart filename metadata. When archive extraction is selected, it inserts that path between literal tt tags in the upload_unzipping progress message without HTML encoding. A basename containing active markup is therefore interpreted in the authenticated Webmin response. Exploitation requires an uploader with module permission to submit, or be induced to submit, the attacker-named archive with extraction selected; it is not an unauthenticated automatic compromise. Script execution inherits the Webmin origin and can operate through the victim session, making a privileged or root administrator the high-impact case.

Profile-based footprint: 5.60% observed module signal (378/6,756 usable profiles; observed gate: updown).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `updown/upload.cgi`.

Module · Useradmin (2)

WEBMIN-0435 — Stored injection through unescaped passwd GECOS data in the user chooser

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The user chooser enumerates account records with my_getpwent and uses passwd field 6 as the displayed real-name value. In the multi-select path it replaces only apostrophes, interpolates the value into a double-quoted JavaScript argument inside an onClick attribute, and emits the same value directly into table HTML. ui_link concatenates the caller-supplied tag fragment without contextual encoding. A crafted GECOS value can therefore break the JavaScript string or inject active markup when an authorized Webmin operator views the chooser. The path requires control of account metadata and a later operator view.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `useradmin/my_user_chooser.cgi`, `ui-lib.pl`.

WEBMIN-2409 — Automatic-home display renders stored home paths as raw HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

The user editor's automatic-home branch concatenates the stored home path directly between HTML tt tags. Manual home-path validation requires an absolute ACL-permitted path but does not reject HTML metacharacters, and external identity sources can also supply the field. In the frozen 2.670 lab, the actual edit_user.cgi response emitted a harmless img/onerror sentinel as a live element. Exploitation requires both influence over the stored path and a viewer whose useradmin ACL selects automatic-home display.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: The attacker controls the account home attribute returned by an external identity source. [environmental]; The viewer ACL selects automatic-home display for the affected account. [environmental]

Audited source: useradmin/edit_user.cgi, useradmin/save_user.cgi.

Module · Webalizer (1)

WEBMIN-0043 — Authenticated reflected XSS through the Webalizer log path

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

In Webmin 2.670, webalizer/edit_log.cgi accepts the file query parameter, applies can_edit_log as a directory-containment authorization check, and then concatenates the value into <tt>$in{'file'}</tt> for ui_table_row without HTML encoding. An authenticated Webalizer user—or a victim induced to follow an eligible crafted link while authenticated—can therefore cause attacker-chosen markup and inline event code to execute in the Webmin origin. The session cookie is HttpOnly, but same-origin script can read page data and submit authenticated Webmin requests. The latest-stable lab reproduced the raw executable SVG attribute using a harmless DOM assignment marker.

Profile-based footprint: 85.60% observed module signal (5,783/6,756 usable profiles; observed gate: webalizer).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: webalizer/edit_log.cgi, webalizer/webalizer-lib.pl.

Module · Webminlog (4)

WEBMIN-0293 — Stored XSS in Webminlog rollback confirmation

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The rollback confirmation page embeds original logged file data and live diff output directly in HTML without output encoding. A lower-privilege Webmin user who can place markup into configuration data and then create a subsequent logged change can seed the rollback record. When a privileged Webminlog user reviews that action for rollback, attacker markup executes in the Webmin origin. This can cross a privilege boundary and perform administrative actions with the viewer's session. No remote-code-execution sink was identified in the reviewed commits.

Profile-based footprint: 5.43% observed module signal (367/6,756 usable profiles; observed gate: webminlog).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: webminlog/rollback.cgi, webminlog/webminlog-lib.pl.

WEBMIN-0427 — Webminlog search reflects the return URL into footer attributes in stable 2.670

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

Frozen stable 2.670 decodes Webminlog search parameters and passes a truthy return value directly to ui_print_footer. Both the generic footer and bundled blue/gray theme path insert the URL into double-quoted href attributes without HTML-attribute escaping. A crafted search.cgi URL can therefore inject event-handler attributes; no matching log result is required for the footer to render. Exploitation requires an authenticated victim authorized for Webminlog, and successful script executes in the Webmin origin with access to the victim's privileged session. Stable's default Content-Security-Policy permits unsafe-inline script. A latest-stable-only container reproduced the issue using the actual stable ReadParse and footer routines.

Profile-based footprint: 5.43% observed module signal (367/6,756 usable profiles; observed gate: webminlog).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: webminlog/search.cgi, web-lib-funcs.pl.

WEBMIN-0734 — Unescaped external-search subtitle enables authenticated reflected XSS

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The Webmin Actions Log accepts search_sub_title from request data and uses it as ui_print_header subtext without HTML escaping. The shared default and gray-theme post-header implementations concatenate that subtext directly inside an HTML element. An authenticated request that passes Webmin's referer check can therefore render attacker-supplied tags and inline event handlers in the Webmin origin. The default unknown-referrer warning reduces straightforward cross-site link delivery; same-origin launch paths and configurations that permit unknown referrers remain exposed. The latest-stable lab reproduced raw SVG/event-handler markup in the response.

Profile-based footprint: 5.43% observed module signal (367/6,756 usable profiles; observed gate: webminlog).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: webminlog/search.cgi, webminlog/view.cgi.

WEBMIN-0954 — Stored XSS in single-line Webmin action-diff objects

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N.

The action-detail page obtains stored file-change records with list_diffs and, for single-line records, interpolates the record's object directly inside tt markup before passing it through ui_tag and ui_table_row. The diff body is HTML-escaped, but the object is not. A principal able to create a logged object containing HTML can therefore store active markup that is emitted when another authorized user views the action. The frozen 2.670 container returned a seeded img element with an onerror handler verbatim in the authenticated response, whose Content Security Policy permits inline script execution.

Profile-based footprint: 5.43% observed module signal (367/6,756 usable profiles; observed gate: webminlog).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: webminlog/view.cgi, webminlog/webminlog-lib.pl.

Command and data injection (116)

Core · ACL (1)

WEBMIN-0225 — External identity lookups interpolate untrusted names into backend queries

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N.

Targeted external user and group lookup helpers build SQL IN lists by adding literal quote characters around supplied names and build LDAP filters by inserting the same names into cn assertions. They neither use SQL parameters nor apply LDAP filter escaping. When an authenticated ACL workflow can pass a crafted target name and the matching external backend is configured, the value can change query semantics, select unintended records, fail the query, or impose additional backend work. The latest-stable harness executed the exact SQL-building helper and observed the controlled tainted name in prepared SQL with no placeholder.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: An external SQL or LDAP identity backend is configured and an ACL workflow accepts an attacker-crafted name. [environmental]

Audited source: acl/acl-lib.pl.

Core · Command-line utilities (3)

WEBMIN-0690 — Developer helpers execute shell metacharacters from repository-controlled paths

CVSS 4.0 Base (CVSS-B): 7.0 (High) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The language-manager Git auto-commit path and update-devel build one-argument Perl system strings by concatenating checkout paths, generated file paths, module identifiers, and language metadata without shell quoting. A crafted checkout/module name or checkout location can therefore break out of the intended Git/update command and execute arbitrary shell commands as the developer. These are local developer tools rather than web request handlers, so exploitation requires an operator to run the affected workflow; the impact is nevertheless command execution in that account.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: An operator runs language-manager or update-devel on a checkout with attacker-controlled path or module metadata. [environmental]

Audited source: `bin/language-manager`, `bin/update-devel`.

WEBMIN-0764 — Service-control config directory can cross into shell execution

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:N/SI:N/SA:N`.

The service-control command treats --config as a literal directory while validating miniserv.conf, but later interpolates that same value into one-argument Perl system calls. Shell metacharacters embedded in an otherwise valid directory name are therefore evaluated when start, stop, restart, reload, kill, or force-restart is invoked. A lower-trust value reaching privileged administrative automation can execute shell commands as that caller. The path requires control of the --config directory value in a privileged service-control invocation.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: Privileged service-control automation supplies a valid --config directory containing shell metacharacters. [environmental]

Audited source: `bin/server`, `bin/webmin`.

WEBMIN-0822 — Crafted Webmin root path is executed through string eval

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

bin/webmin reads root= from the selected miniserv.conf, verifies only that the resulting string names a directory, appends /vendor_perl, and inserts that value inside eval "use lib '$libroot'". An existing root directory whose name contains a closing quote and Perl statements breaks out of the intended string and executes those statements. Frozen 2.670 requires bin/webmin to run as root, so attacker influence over an alternate configuration/root path used by privileged automation can become root code execution. The default /etc/webmin configuration is normally root-controlled, and this audit found no remote path to the value, which materially limits exposure.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: Root runs bin/webmin with a configured root directory whose name contains Perl string-breakout syntax. [environmental]

Audited source: `bin/webmin`.

Core · Cross-cutting (3)

WEBMIN-0572 — Download filenames can split HTTP response headers

CVSS 4.0 Base (CVSS-B) · path `publication:module:file:primary`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `publication:module:filemin:primary`: 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:N/VA:N/SC:L/SI:L/SA:N`.

Download handlers construct Content-Disposition by interpolating a filename without first removing line breaks or quotes. A permitted existing filename containing a newline can create an additional HTTP response header in the Webmin origin. The condition remains in the frozen stable Filemin download handler and was reproduced using only the latest-stable Docker target.

Profile-based footprint: 94.92% observed module signal (6,413/6,756 usable profiles; observed gate: file, filemin).

Recorded conditions: An allowed existing filename contains a newline and an authorized user downloads it. [environmental]

Audited source: `filemin/download.cgi`, `mailboxes/detach.cgi`.

WEBMIN-0671 — Authenticated root command injection in the SMF logfile viewer

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The SMF logfile viewer extracts text between single quotes from the logfile request parameter but does not quote or validate it before embedding it in wc and tail shell commands. An authenticated SMF module user can inject shell separators and run arbitrary commands in Webmin's root context.

Profile-based footprint: 84.92% observed module signal (5,737/6,756 usable profiles; observed gate: smf).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `smf/log_viewer.cgi`.

WEBMIN-0854 — Authenticated update requests can inject root-run YUM/DNF options

CVSS 4.0 Base (CVSS-B) · path `publication:module:package-updates:primary`: 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `publication:module:software:primary`: 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The package-update endpoint accepts client-controlled option text and forwards it without an option allowlist to the selected package-system installer. In the YUM/DNF path, the text is shell-quoted token by token and appended to the command, so shell control characters are neutralized while package-manager options remain operative. An authenticated module user can therefore supply options such as an additional repository location together with `--nogpgcheck`, expanding a root-run package transaction beyond the administrator's configured repository trust boundary.

Profile-based footprint: 95.01% observed module signal (6,419/6,756 usable profiles; observed gate: package-updates, software).

Recorded conditions: Package Updates uses YUM or DNF and a module user supplies an option-shaped update parameter. [environmental]

Audited source: `package-updates/update.cgi`, `package-updates/package-updates-lib.pl`.

Core · Integration (1)

WEBMIN-0622 — Authenticated Amavis configuration injection yields stored Perl code execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The optional SpamAssassin/Amavis editor accepts arbitrary request text for fields presented as numeric because check_amavis_value only validates when its mode is the literal "*", while callers pass 1. save_amavis_directive then embeds those values unquoted in the Perl-format Amavis configuration. A user explicitly granted this optional page can persist Perl statements that execute when the configuration is loaded. The shipped default module ACL omits the page, and the execution identity depends on the Amavis service startup model.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: spam).

Recorded conditions: Amavis editing is enabled and delegated to a Webmin user. [nondefault]; The injected Perl configuration is later loaded. [environmental]

Audited source: spam/save_amavisd.cgi, spam/spam-amavis-lib.pl.

Core · Packaging (1)

WEBMIN-0783 — Privileged setup.pl evaluates shell syntax from system hostname

CVSS 4.0 Base (CVSS-B): 8.5 (High) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The Perl installer obtains a hostname/FQDN from local host and resolver configuration, inserts it without shell quoting into an OpenSSL subjectAltName fragment, and invokes the resulting text with Perl's shell-interpreted pipe open. When setup.pl runs as root, command-substitution syntax carried by the hostname executes as root. The frozen 2.670 source retains the dataflow, and the latest-stable Docker lab reproduced it with the real installer.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: Root runs setup.pl with an attacker-influenced hostname containing shell substitution syntax. [environmental]

Audited source: setup.pl, web-lib-funcs.pl.

Module · Apache (3)

WEBMIN-0006 — Unsafe executable storage of SSL passphrases

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

Fixed SSL private-key passphrases are embedded verbatim into a generated /bin/sh helper as an unquoted echo argument, and the helper is set to mode 0755. This makes the secret readable to local users and gives shell metacharacters in passphrase text executable meaning when the helper is invoked. The frozen newest-stable lab reproduced both the 0755 mode and harmless command substitution.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: Apache fixed-passphrase helper mode must be selected. [nondefault]

Audited source: apache/mod_ssl.pl.

WEBMIN-0458 — Apache Listen protocol field permits directive injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

For Apache 2.4, Webmin renders the Listen protocol as a select limited to an empty value, http, or https, but save_BindAddress_Listen_Port reads the raw Name_i request field without enforcing that set. It appends the value to the Listen directive, and the common Apache serializer writes embedded newlines verbatim. An authenticated account delegated Apache global networking configuration can therefore add an unrelated directive to root-managed Apache configuration, crossing the intended editor boundary. The exact downstream effect depends on the injected directive and a subsequent configuration reload; the controlled reproduction used only the inert ServerTokens Prod directive.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: Apache delegate has global networking configuration permission and submits a newline in Listen protocol text. [environmental]

Audited source: apache/core.pl, apache/apache-lib.pl.

WEBMIN-0604 — Delegated ErrorDocument editing permits Apache configuration injection and root RCE on reload

CVSS 4.0 Base (CVSS-B): 8.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The ErrorDocument message save path wraps attacker input in quotes but does not reject embedded quotes or line breaks. A Webmin user who is intentionally limited to ErrorDocument on an authorized virtual host can submit a message that closes the directive, inserts a new line such as a piped ErrorLog command, and comments out the trailing quote. parse_inputs and save_directive persist that scalar verbatim, and flush_file_lines writes the embedded newline to Apache's configuration. When the configuration is next loaded, a piped logging command is launched by Apache's privileged parent, normally yielding root command execution. The attacker need not hold the separate global-configuration or apply ACL; a later authorized or routine restart is sufficient.

Profile-based footprint: 92.07% observed module signal (6,220/6,756 usable profiles; observed gate: apache).

Recorded conditions: A later Apache configuration reload or restart loads the injected directive. [environmental]

Audited source: `apache/core.pl`, `apache/apache-lib.pl`.

Module · Backup Config (2)

WEBMIN-0204 — Tar file-list option injection expands root backup contents

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

execute_backup accepts recursively expanded filesystem names, removes only a leading slash, writes each raw name followed by a line feed, and passes the resulting file to GNU tar with T mode. Unix filenames may themselves contain line feeds, and GNU tar's default file-list parser treats resulting lines beginning with '-' as options. A less-privileged writer in a recursively backed-up directory can therefore embed a '-C' directive and another filename, causing the root-run backup to include a root-readable file outside the selected directory. The frozen latest-stable function-level reproduction used a harmless root-only synthetic file and confirmed it appeared in the archive. The reproduction supports unintended file inclusion; it does not support an arbitrary-command-execution claim.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: backup-config).

Recorded conditions: An attacker can create a leading-dash filename in a directory included in a privileged tar backup. [environmental]

Audited source: `backup-config/backup-config-lib.pl`.

WEBMIN-0486 — Leading-dash restore entries permit tar option injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

restore.cgi accepts newline-delimited Other files values and execute_restore appends them to the selected archive members. The routine applies Perl quotemeta and constructs a shell command of the form tar ... $qfiles, but it does not place '--' before the supplied member list. Quoting preserves each shell argument yet does not change a leading hyphen, so GNU tar can parse a purported file name as an option. A delegated restore operator can therefore reach tar option actions under Webmin's service privileges. The path requires access to arbitrary configuration restore, which already grants substantial administrative authority.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: backup-config).

Recorded conditions: Backup Configuration delegate can restore archives and supplies a leading-dash Other files member. [environmental]

Audited source: `backup-config/restore.cgi`, `backup-config/backup-config-lib.pl`.

Module · Bandwidth (1)

WEBMIN-0095 — Bandwidth interface validation permits root command injection through Shorewall

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Bandwidth setup accepts a manually supplied interface when it contains any sequence of non-whitespace characters. With Shorewall selected, stable setup_shorewall_rules interpolates the value without quoting into four commands in Shorewall's executable start file and immediately performs a restart. An authenticated Webmin principal restricted to the Bandwidth module's setup capability can therefore inject shell syntax and execute arbitrary commands as the root-running Webmin service. The 2.670 reproduction used only a harmless id command and obtained uid=0(root).

Profile-based footprint: 82.40% observed module signal (5,567/6,756 usable profiles; observed gate: bandwidth).

Recorded conditions: Bandwidth Monitoring must be configured to use its Shorewall backend. [nondefault]

Audited source: `bandwidth/setup.cgi`, `bandwidth/bandwidth-lib.pl`.

Module · Bind8 (1)

WEBMIN-0595 — Delegated BIND record editor can append injected lines to a root-selected filesystem target

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

In Webmin 2.670, bind8/save_record.cgi checks the submitted zone and record type against the delegated user's ACL, but it separately trusts the submitted file form field for the eventual create_record call. The NAPTR replacement field accepts a percent-decoded newline, and make_record preserves that newline when flushing the selected file. Consequently, a user restricted to example.test and NAPTR records can make the root Webmin process append an injected line to a path outside the configured zone file. The controlled latest-stable run verified that a disallowed A request did not change the target, whereas the allowed NAPTR request appended exactly one marker to a root-owned outside target.

Profile-based footprint: 94.63% observed module signal (6,393/6,756 usable profiles; observed gate: bind8).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `bind8/save_record.cgi`.

Module · Cluster Cron (1)

WEBMIN-2416 — Unescaped Cluster Cron values permit remote Perl statement injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Cluster Cron builds a Perl program for remote_eval by interpolating cluster_user and cluster_command directly inside single-quoted Perl string literals. A stored command containing closing quotes and parentheses can terminate the intended command_as_user/backquote_command expression and append Perl statements that execute in the remote Webmin evaluation context. The frozen latest-stable implementation retains the construction, and an inert container probe set an in-memory marker outside the intended nobody command wrapper. This is a security observation rather than a standard-ACL privilege-escalation finding: ordinary access to the Cluster Cron module already permits arbitrary job commands and selection of root as the target user.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: cluster-cron).

Recorded conditions: A cluster Cron command or username contains crafted Perl syntax and is sent to a managed host. [environmental]

Module · Cluster Useradmin (2)

WEBMIN-0060 — Managed cluster host can inject Perl into privileged cross-host user creation

CVSS 4.0 Base (CVSS-B): 8.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Cluster Users and Groups accepts group records returned by a managed Webmin host and caches the gid field unchanged. During user creation, the first selected host's cached gid is copied without a digits-only check and interpolated between single quotes in Perl code passed to remote_eval for each selected target. A compromised or malicious managed host can return a quote-bearing gid; a later authenticated cross-host user-creation operation then executes attacker-chosen Perl in the target useradmin RPC context. Where the linked-server credential is root, this provides root command execution and lateral movement. The attacker must control a managed host and wait for a subsequent operator action.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: A compromised managed cluster host supplies user data that an operator later synchronizes to another host. [environmental]

Audited source: `cluster-useradmin/refresh.cgi`, `cluster-useradmin/create_user.cgi`.

WEBMIN-0115 — Remote Perl code injection through synchronized home path

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Cluster user synchronization caches account records returned by managed systems and later embeds each home field inside single-quoted Perl snippets passed to remote_eval on a selected target. A quote-bearing home value can terminate the intended string and add Perl expressions. When an operator synchronizes the account with home creation enabled, those expressions execute under the target Webmin RPC service identity. This creates a propagation path from a compromised managed account source to another managed server.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: cluster-useradmin).

Recorded conditions: A managed host controls a synchronized home path and an operator enables cross-host home creation. [environmental]

Audited source: `cluster-useradmin/sync.cgi`, `web-lib-funcs.pl`.

Module · Cluster Webmin (1)

WEBMIN-0208 — Cluster upgrade local package path permits root command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Cluster Webmin upgrade handler accepts a request-provided local package path for source=0, verifies that the exact literal path is readable, and then interpolates it without shell quoting into rpm, dpkg, and tar commands. An authorized cluster-upgrade operator who can identify or place a readable filename containing shell metacharacters can execute an appended shell command with the Webmin service privileges. The frozen latest-stable lab exercised the actual CGI with such a readable path and captured uid=0 command output.

Profile-based footprint: 4.19% observed module signal (283/6,756 usable profiles; observed gate: cluster-webmin).

Recorded conditions: The path requires configured Cluster Webmin hosts. [nondefault]

Audited source: `cluster-webmin/upgrade.cgi`.

Module · Cpan (2)

WEBMIN-0382 — Failed RPM fallback evaluates a CPAN module token as Perl code

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The CPAN RPM fallback accepts a request-controlled, whitespace-delimited module token without a module-name grammar. In frozen Webmin 2.670, an unmatched token is wrapped as perl(<token>), and an empty package-installer result leads to eval of a Perl use string containing that token. A token beginning with an installed core module and followed by a BEGIN block compiles and executes before fallback processing. The isolated latest-stable image reproduced a harmless marker with real and effective UID 0.

Profile-based footprint: 4.25% observed module signal (287/6,756 usable profiles; observed gate: cpan).

Recorded conditions: CPAN RPM fallback sees an unmatched crafted module token and the package installer returns no result. [environmental]

Audited source: `cpan/download.cgi`, `cpan/config`.

WEBMIN-0425 — Authenticated root RCE in CPAN RPM fallback module check

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

When an authenticated user authorized for the CPAN module requests a CPAN-source install, the RPM path creates a `perl(<submitted token>)` package candidate. If the system package installer returns no packages, Webmin tests whether the module is already installed by interpolating that untrusted token into Perl string eval. Webmin CGI code runs as root, so Perl syntax embedded in the module token can execute arbitrary commands as uid 0. The path requires an RPM package backend, a package-manager attempt that returns no package, and an authorized CPAN-module user.

Profile-based footprint: 4.25% observed module signal (287/6,756 usable profiles; observed gate: cpan).

Recorded conditions: The host uses the RPM package backend for CPAN-source installation. [environmental]; The package-manager attempt returns no matching package, reaching the Perl module fallback check. [environmental]

Audited source: `cpan/download.cgi`, `cpan/config`.

Module · Cron (1)

WEBMIN-2413 — Authenticated Cron save-and-run response-header injection

CVSS 4.0 Base (CVSS-B): 0.0 (None) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:N/SC:N/SI:N/SA:N`.

For an existing Cron job, save_cron.cgi uses the request idx value both as a numeric Perl array index and, when saverun is set, as raw text inside a redirect URL. ReadParse percent-decodes a line feed in idx, while redirect emits the constructed value as a Location header. An authenticated Webmin user who is permitted to edit the selected Cron job can therefore add headers to the resulting 302 response. The save operation remains protected by can_edit_user checks, exec_cron.cgi rechecks the resolved job user and changes to that Unix identity, and miniserv ignores the remaining CGI body for Location responses. Consequently the demonstrated impact is limited response manipulation, not cross-account privilege escalation, XSS, or command execution.

Profile-based footprint: 96.86% observed module signal (6,544/6,756 usable profiles; observed gate: cron).

Recorded conditions: Cron delegate can edit an existing job and invokes save-and-run with a CR/LF-bearing idx. [environmental]

Audited source: `cron/save_cron.cgi`, `web-lib-funcs.pl`.

Module · DHCP (8)

WEBMIN-0128 — Delegated DHCP editor can inject directives into privileged configuration

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The authenticated save_options.cgi endpoint enforces object-level write access, then processes custom DHCP option values. For string/text options it places the submitted value inside quotes without escaping embedded quotes or newlines, while directive_lines serializes that value verbatim and appends a semicolon only to the final physical line. A user allowed to edit one host can therefore terminate the intended value and insert an additional DHCP directive into the root-owned configuration. Latest-stable reproduction with global=0 and host write

permission produced a separate marker directive in a uid-0, mode-0600 file. This demonstrates privileged DHCP configuration injection, not arbitrary operating-system command execution or a root shell.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: dhcpd/save_options.cgi, dhcpd/dhcpd-lib.pl.

WEBMIN-0161 — Delegated DHCP option edits can inject directives and expose local file lines

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

An authenticated Webmin user with write access to a delegated DHCP object can submit a custom option value containing an embedded quote and newline. save_options.cgi places literal surrounding quotes around the raw value but does not neutralize those embedded delimiters, and directive_lines preserves them as new configuration lines, allowing the value to escape the intended object scope. If that user also has the independently configurable apply permission, an injected include or another controlled failure can make the privileged DHCP command emit a '<path> line <number>:' diagnostic. parse_error_out trusts that path, reads up to eleven surrounding lines with Webmin's privileges, and returns them in the error page. The latest-stable container reproduced both exact stable-code primitives; it did not emulate a full browser session or a production DHCP daemon.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: DHCP delegate has object-write permission and can submit a quote and newline in a custom option. [nondefault]; Local-file disclosure additionally requires the separately delegated apply permission. [nondefault]

Audited source: dhcpd/save_options.cgi, dhcpd/dhcpd-lib.pl.

WEBMIN-0227 — Delegated DHCP bracketed option type injection writes additional service directives

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

In the DHCP module's version-3 option-definition path, a dtype enclosed by outer braces is accepted with unrestricted single-line interior characters, split on whitespace, and serialized without quoting. An authenticated user authorized to edit options for an otherwise limited DHCP object can insert semicolons, braces, quotes, and a comment marker to terminate the intended option definition and write additional directives into the root-owned DHCP configuration. Depending on object placement and the injected DHCP syntax, this can escape delegated scope, change address-allocation or network policy, or prevent the daemon from loading configuration. The verified primitive is DHCP configuration injection; direct shell execution or root command execution was not demonstrated.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: dhcpd/save_options.cgi, dhcpd/dhcpd-lib.pl.

WEBMIN-0279 — DHCP interface configuration enables deferred root command execution through shell-sourced environment files

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

An authenticated Webmin account assigned the DHCP module and allowed to edit module configuration can POST shell syntax as iface to dhcpd/save_iface.cgi. The handler checks noconfig but not the separate apply permission or any DHCP object-write permission, accepts arbitrary nonempty tokens, and persists them through write_env_file as Debian INTERFACESv4/INTERFACES or RHEL DHCPDARGS. write_env_file

places whitespace values inside double quotes without escaping shell substitutions, so $(...) and backticks remain active. If a root service start, restart, or boot path shell-sources that file, the planted command runs as UID 0. The trigger and consumer matter: this finding applies to shell-sourcing Debian or legacy sysconfig service paths and does not claim that modern systemd EnvironmentFile parsing by itself evaluates shell syntax.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: The host uses a shell-sourcing Debian or legacy sysconfig DHCP service path. [environmental]; A later privileged service start, restart, or boot sources the generated file. [environmental]

Audited source: `dhcpd/save_iface.cgi`, `web-lib-funcs.pl`.

WEBMIN-0311 — Delegated domain-search editing injects directives into root-owned DHCP configuration

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The DHCP options endpoint authorizes the selected configuration scope, then passes the domain-search request value to type 6. That branch splits the value only on whitespace and surrounds each token with double quotes; it does not escape an embedded quote, semicolon, or comment marker. The generic renderer writes the constructed value unchanged and appends its own semicolon. A DHCP-scoped authenticated writer can therefore terminate the domain-search value and persist an additional directive in the root-owned DHCP configuration. Frozen Webmin 2.670 rendered an inert crafted value as `option domain-search "safe.example";authoritative;#";`. This proves service-configuration injection; it does not by itself demonstrate root command execution.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `dhcpd/save_options.cgi`, `dhcpd/dhcpd-lib.pl`.

WEBMIN-0380 — Delegated FreeBSD DHCP configuration can persist root command execution in rc.conf

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The DHCP interface-saving CGI accepts an authenticated user's iface value, performs only whitespace split-and-join normalization, and on FreeBSD passes it to init::save_rc_conf. That helper writes dhcpd_ifaces="<value>" into root-owned rc.conf without escaping dollar-command substitution or backticks. Shell substitutions remain active inside double quotes, so when FreeBSD boot or service tooling later sources rc.conf, attacker-chosen commands execute as root. The path requires authenticated delegated DHCP configuration access, a FreeBSD host, and a later privileged source operation.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: The host uses the FreeBSD DHCP rc.conf backend. [environmental]; A later privileged boot or service action sources rc.conf. [environmental]

Audited source: `dhcpd/save_iface.cgi`, `init/init-lib.pl`.

WEBMIN-0701 — Array-valued custom DHCP options allow configuration-statement injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The custom-option save loop copies cval_N into $cv. When the parsed option type begins with array, the value is deliberately left unquoted but is not checked for semicolons, comments, braces, or newlines. directive_lines then concatenates that scalar directly into dhcpd.conf and appends a terminator. An authenticated DHCP module user with object write permission can therefore turn one array value into more than one ISC DHCP statement, crossing the intended field-level constraint. The frozen 2.670 image reproduced this using a harmless controlled on-commit assignment; actual effect requires the altered configuration to be applied or otherwise consumed by the DHCP service.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: DHCP delegate can save an array-typed custom option containing statement delimiters. [environmental]

Audited source: `dhcpd/save_options.cgi`, `dhcpd/dhcpd-lib.pl`.

WEBMIN-0766 — Authenticated DHCP interface input can execute commands when Debian defaults are sourced

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

In the DHCP module, save_iface.cgi accepts the iface POST value, splits it only on whitespace or NUL, rejoins it, and writes it to the Debian INTERFACESv4 setting when that key exists. The shared write_env_file routine serializes the value as shell assignment text without escaping shell metacharacters; a value containing spaces is enclosed in double quotes, which still permit command substitution. An authenticated Webmin user with DHCP configuration rights can therefore persist a command in /etc/default/isc-dhcp-server that executes when a privileged service or administrator sources the file. The Docker proof used a harmless marker inside an isolated container and confirmed the marker appeared only after sourcing the generated file.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: The Debian DHCP configuration has the INTERFACESv4 key used by this save path. [environmental]; A privileged service or administrator later sources the generated file. [environmental]

Audited source: `dhcpd/save_iface.cgi`, `web-lib-funcs.pl`.

Module · Filemin (2)

WEBMIN-0847 — Authenticated tar option injection can execute commands as the Filemin identity

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:N/SI:N/SA:N`.

File Manager builds tar command strings by appending each selected relative name after the archive path and -C directory. Shell escaping prevents shell metacharacter splitting but leaves a leading hyphen meaningful to GNU tar. An authenticated requester can therefore submit checkpoint options and an action, causing command execution as the configured Filemin Unix identity. This is especially consequential when a Webmin account is restricted to selected paths while Filemin operates as root or another more privileged identity.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: The Filemin archive path uses GNU tar with checkpoint-option semantics. [environmental]; Filemin runs as a Unix identity with greater host authority than the restricted Webmin account. [environmental]

Audited source: `filemin/compress.cgi`, `filemin/filemin-lib.pl`.

WEBMIN-2463 — Browser-only chmod mode limit permits GNU option injection

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The chmod dialogs truncate the mode field in browser JavaScript, but chmod.cgi accepts an unrestricted perms request value and places it in chmod's first argument position. Shell metacharacters are escaped, but GNU chmod still parses a leading option. In the controlled 2.670 reproduction, --reference named a synthetic file outside allowed_paths; chmod copied that file's 0711 mode onto an allowed target that began at 0644. This demonstrates option injection and a narrow cross-boundary metadata reference, while modification remains confined to a target the user could already chmod.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: Filemin delegate is confined by allowed_paths. [nondefault]; The submitted chmod mode begins with a GNU option such as --reference. [environmental]

Audited source: `filemin/chmod.cgi`.

Module · Filter (2)

WEBMIN-0066 — Authenticated Usermin identity can inject a root shell command

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

During root-started Usermin Filter module initialization, the configured Virtualmin lookup path and raw authenticated remote_user are concatenated into a single backquote_command string. WebminCore executes that string through shell backticks before switch_to_remote_user. An accepted account identity containing shell metacharacters can therefore run arbitrary commands as root when the module loads. The precondition is control of a syntactically accepted authenticated identity name; conventional Unix username policies may reduce exposure, but Webmin/Usermin can use non-POSIX authentication sources.

Profile-based footprint: 87.39% observed module signal (5,904/6,756 usable profiles; observed gate: filter).

Recorded conditions: The authentication source accepts an authenticated account identity containing shell metacharacters. [nondefault]

Audited source: `filter/filter-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0246 — Filter rule newline injection executes commands as the mail account

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:N/SI:N/SA:N`.

The header-condition save path accepts an embedded newline in condvalue when regular-expression mode is requested. Filter rule generation concatenates that value into a Procmail condition, and the Procmail serializer preserves the newline, allowing a following `* ? shell-command` condition to be inserted. When a matching message is delivered, Procmail evaluates the injected shell condition. This turns Filter-rule authority into command execution as the associated Unix delivery account; it does not by itself grant root unless that account or delivery context is root.

Profile-based footprint: 87.39% observed module signal (5,904/6,756 usable profiles; observed gate: filter).

Recorded conditions: A Filter user can save a header condition and a matching message is later delivered through Procmail. [environmental]

Audited source: `filter/save.cgi`, `filter/filter-lib.pl`.

Module · Firewall (1)

WEBMIN-0295 — Mixed-quote firewall comments escape the serialized comment argument

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The firewall rule serializer chooses double quotes whenever an argument contains an apostrophe, but it does not escape double quotes that are also present. For a --comment value containing both quote types, an internal double quote can close the wrapper and turn later comment text into standalone iptables-restore option tokens. The controlled stable probe rendered the benign test comment x' " --dport 22 " so --dport and 22 were separate tokens. This can alter rule semantics or make restoration fail, threatening firewall integrity or availability. The precondition is an authenticated operator already permitted to save firewall rules with --comment storage; this finding does not claim authority beyond that module permission.

Profile-based footprint: 86.68% observed module signal (5,856/6,756 usable profiles; observed gate: firewall).

Recorded conditions: An authorized firewall-rule editor saves a comment containing both quote types. [environmental]

Audited source: `firewall/firewall-lib.pl`.

Module · Firewalld (2)

WEBMIN-0591 — FirewallD port-forward requests permit root shell command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

FirewallD port-forward create, update, individual-delete, and bulk-delete paths pass request-derived protocol or encoded rule fields to helpers that concatenate those values into shell command strings for backquote_logged. Ports and destination addresses receive partial validation, but the protocol and old-rule components do not. An authenticated Webmin principal with access to this module can inject shell syntax and run arbitrary commands with the Webmin process privileges, normally root. The frozen latest-stable source retains the request paths and shell sinks, and the isolated Docker test produced harmless create-path and delete-path marker files.

Profile-based footprint: 93.04% observed module signal (6,286/6,756 usable profiles; observed gate: firewalld).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `firewalld/save_forward.cgi`, `firewalld/firewalld-lib.pl`.

WEBMIN-0907 — FirewallD rule deletion passes client-controlled rich and direct rules to a shell

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The FirewallD rule-removal endpoint accepts rule strings from request data and dispatches them to rich or direct removal helpers. The rich path encloses the string in shell single quotes without escaping embedded quotes; the direct path appends the string unquoted. Both command strings reach backquote_logged, which executes a scalar through the shell. An authenticated user who is authorized for the FirewallD module can therefore exceed that module's intended firewall-management scope and run arbitrary commands with the Webmin service process privileges. Frozen stable source inspection confirmed both paths, and the network-disabled latest-stable container produced harmless sentinel files through each path.

Profile-based footprint: 93.04% observed module signal (6,286/6,756 usable profiles; observed gate: firewalld).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `firewalld/save_rules.cgi`, `firewalld/firewalld-lib.pl`.

Module · Fsdump (1)

WEBMIN-0971 — Non-Linux FSdump restore arguments permit privileged shell command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

In the frozen stable source, FreeBSD restore construction appends extra command-line text and selected-file text directly to a string later supplied to a shell; IRIX and Solaris retain related raw fragments. The common restore CGI checks only the restore ACL, while the backend omits the Linux implementation's extra-option ACL check and word-wise quotemeta handling. A crafted restore request can therefore append a shell command in the privileged FSdump execution context on an affected non-Linux deployment. The latest-stable container test loaded the unmodified stable FreeBSD builder, set restore access to true and extra access to false, supplied a semicolon payload, and observed the marker command execute.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: fsdump).

Recorded conditions: The host uses an affected non-Linux restore backend such as FreeBSD, IRIX, or Solaris. [environmental]

Audited source: `fsdump/freebsd-lib.pl`, `fsdump/irix-lib.pl`.

Module · Heartbeat (1)

WEBMIN-0242 — Delegated Heartbeat configuration permits root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Heartbeat's module configuration exposes the heartbeat executable path as a generic editable text field. The module index interpolates that value unquoted into the shell command used for the -V version probe. A delegated Heartbeat user whose module ACL does not explicitly set noconfig can store shell syntax and cause arbitrary commands to run as the root Webmin process simply by loading the module index. The 2.670 runtime proof used only id and observed uid=0(root).

Profile-based footprint: 4.10% observed module signal (277/6,756 usable profiles; observed gate: heartbeat).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `heartbeat/index.cgi`, `heartbeat/config.info`.

Module · Inetd (1)

WEBMIN-0385 — Authenticated inetd delete redirects permit response-header injection

CVSS 4.0 Base (CVSS-B): 0.0 (None) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:N/SC:N/SI:N/SA:N`.

The service and RPC save endpoints percent-decode request fields and, when the delete button is selected, interpolate the record-position values into a redirect URL without validating them as numeric. A module-authorized user can place CR/LF in these values and make redirect emit an extra response header. The demonstrated effect is confined to the authenticated caller's response; the audit found no direct OS command execution, root escalation, or established cross-user delivery path.

Profile-based footprint: 4.23% observed module signal (286/6,756 usable profiles; observed gate: inetd).

Recorded conditions: Inetd delegate submits a CR/LF-bearing record position with a delete action. [environmental]

Audited source: `inetd/save_serv.cgi`, `inetd/save_rpc.cgi`.

Module · Init (1)

WEBMIN-0472 — Delegated Status monitor editing enables root command execution through a crafted init action

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

An authenticated user allowed to edit System and Server Status monitors can submit an arbitrary non-empty action for an init-service monitor even though the UI displays a service dropdown. On systems using Webmin's SysV init backend, status_action preserves an absolute action string and action_running interpolates it unquoted into a shell command. A crafted absolute shell command is therefore executed when the monitor is evaluated, in Webmin's privileged context. The condition is platform-dependent because the modern systemd branch quotes the service name, but SysV init remains a supported newest-stable backend.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: The host uses Webmin's SysV init backend for the monitor action. [environmental]; The monitor is evaluated after the action value is stored. [environmental]

Audited source: `status/init-monitor.pl`, `init/init-lib.pl`.

Module · Ipfw (2)

WEBMIN-0409 — Authenticated IPFW rule input executes shell commands as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

In the format-0 IPFW application path, Webmin serializes every rule to a shell command. Tokens containing parentheses are enclosed in raw single quotes, but embedded single quotes are not escaped. apply_rules then treats any token beginning and ending with a quote as already safe and excludes it from quotemeta before interpolating it into a command passed to backquote_logged. save_rule.cgi accepts custom interface values containing all non-whitespace shell syntax. An authorized IPFW-module user can therefore submit a value such as one that closes the quote, runs a no-whitespace command using ${IFS}, comments out the trailing quote, and obtains arbitrary command execution as the Webmin service account, normally root. Applying the saved configuration triggers execution.

Profile-based footprint: 84.99% observed module signal (5,742/6,756 usable profiles; observed gate: ipfw).

Recorded conditions: The host runs the IPFW BSD Firewall backend. [environmental]

Audited source: `ipfw/ipfw-lib.pl`, `ipfw/save_rule.cgi`.

WEBMIN-0727 — IPFW save-file configuration enables root command execution through pipe-open

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

An authenticated user delegated the BSD Firewall module with permission to edit module configuration can set the save_file path to shell text ending in a pipe. The module stores that value without a path-only constraint; get_config() resolves it as the default rules source and passes it to Perl's two-argument open as a command pipe. Triggering a no-argument read executes the shell text with the Webmin process identity, which is root in the standard privileged installation. The IPFW module declares FreeBSD and macOS support; the path also requires authentication and module-configuration permission.

Profile-based footprint: 84.99% observed module signal (5,742/6,756 usable profiles; observed gate: ipfw).

Recorded conditions: The host uses the IPFW backend supported on FreeBSD or macOS. [environmental]; The delegated BSD Firewall user can edit module configuration. [environmental]

Audited source: `ipfw/config.info`, `config_save.cgi`.

Module · LDAP Client (1)

WEBMIN-0610 — LDAP-controlled uid command injection can execute as Webmin root during configuration validation

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

In the LDAP Client configuration check, Webmin searches the configured base for posixAccount objects and takes the first entry. If getpwnam does not resolve that entry's uid, check.cgi constructs id -a by directly concatenating the LDAP uid and passes the string to backquote_command. That helper executes with Perl backticks, so shell metacharacters in the LDAP uid are interpreted. A directory administrator who can place and prioritize a crafted posixAccount can therefore gain OS command execution when an authorized Webmin user runs Check Configuration. The command runs with the Webmin CGI's privileges; the latest-stable container exercise confirmed effective UID 0. The path requires LDAP data control and an authorized user running Check Configuration; possession of an LDAP root password alone does not grant host root.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: ldap-client).

Recorded conditions: The attacker can place and prioritize a crafted posixAccount in the configured LDAP base. [environmental]; The selected LDAP uid is unresolved by local getpwnam. [environmental]; An authorized Webmin user runs Check Configuration. [environmental]

Audited source: ldap-client/check.cgi, web-lib-funcs.pl.

Module · Lpadmin (1)

WEBMIN-0315 — SMB printer configuration permits command execution as the print-service account

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:N/SI:N/SA:N.

The lpadmin Windows-printer save path accepts SMB server, share, user, password, and workgroup fields from an authenticated add/edit request and passes them to create_webmin_windows_driver. That function writes the values into an executable shell interface without shell-safe argument construction. A delegated lpadmin user can therefore store shell syntax that runs when the printer service invokes the interface. Frozen Webmin 2.670 still contains the path. The controlled reproduction executed only id as the typical Linux lp service account; it did not demonstrate root execution.

Profile-based footprint: 4.14% observed module signal (280/6,756 usable profiles; observed gate: lpadmin).

Recorded conditions: The printer service later invokes the generated shell interface. [environmental]

Audited source: lpadmin/save_printer.cgi, lpadmin/lpadmin-lib.pl.

Module · Lvm (1)

WEBMIN-0379 — Authenticated root command injection in LVM pvmove

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The LVM pvmove CGI accepts request-controlled physical-volume selectors and checks only that the two strings differ. move_logical_volume embeds both strings in a command and invokes it through Perl's shell-opening pipe form. An authenticated user who has LVM module access can therefore append shell syntax and run arbitrary commands as the root Webmin service account. The frozen 2.670 source retains the path, and a container test using that exact stable helper produced a marker and UID 0 output. The path requires authenticated LVM-module authorization.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: lvm).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: lvm/pvmove.cgi, lvm/lvm-lib.pl.

Module · Mailboxes (2)

WEBMIN-0153 — Mail recipient accepted as a privileged sendmail option

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

The local-delivery path parses recipient strings and appends shell-escaped values to a sendmail command without an option terminator. Shell escaping preserves one argv element but does not stop a leading hyphen from being interpreted by sendmail. A compose-authorized Webmin user can therefore inject mailer options into a process commonly launched with Webmin's elevated privileges. The safe latest-stable reproduction used a recorder instead of a real MTA and observed -X/evidence/attacker-controlled.log as argv[2].

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: mailboxes/mailboxes-lib.pl, mailboxes/boxes-lib.pl.

WEBMIN-0925 — Shell command injection in local Dovecot IMAP process launch

CVSS 4.0 Base (CVSS-B): 9.3 (Critical) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

When imap_login receives a folder whose server is '', *it chooses a fixed Dovecot executable, concatenates ' -u ' and the folder user (or remote-user fallback) into one scalar string, and passes that string to IPC::Open3::open3. Perl applies shell interpretation to a one-string process command when shell metacharacters are present. A caller that lets a less-trusted actor control the selected user can therefore execute commands with the invoking Webmin process privileges. The frozen built-in status monitor's form requires an IP host and does not itself expose server=''*, so this finding is constrained to other local-IMAP callers or integrations that satisfy the stated preconditions.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: A local-IMAP caller or integration invokes imap_login with server='*' and a less-trusted user value. [nondefault]

Audited source: `mailboxes/folders-lib.pl`, `status/imap-monitor.pl`.

Module · Minecraft (4)

WEBMIN-0369 — Java argument editing enables host shell command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The configuration page accepts a free-form Java argument string, saves it as java_args, and the server start builder concatenates it unquoted into a shell command executed by system_logged. Shell substitutions and metacharacters are therefore evaluated as the configured Minecraft Unix account. The frozen stable lab placed command substitution in java_args and observed its host marker after requesting server start.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: The configured Minecraft server start path consumes the editable java_args value. [environmental]; The server is started after the value is saved. [environmental]

Audited source: `minecraft/save_conf.cgi`, `minecraft/minecraft-lib.pl`.

WEBMIN-0481 — Minecraft-only Webmin user can inject commands into the root server-start shell

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The ordinary Minecraft configuration form accepts an unrestricted args value and save_conf.cgi persists it as java_args. get_start_command concatenates java_args into a shell command without quoting, and start_minecraft_server executes that string with system_logged. The shipped Minecraft configuration selects unix_user=root, so no lower-privilege user wrapper is applied. A Webmin identity delegated only the minecraft module can therefore save shell syntax in Java arguments and trigger start.cgi to run a command as UID 0. The latest-stable lab used harmless command substitution that wrote only the effective UID and observed 0.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `minecraft/edit_conf.cgi`, `minecraft/save_conf.cgi`.

WEBMIN-0555 — Minecraft-only Webmin user can execute root commands through boot-wrapper Java arguments

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Minecraft module accepts an authenticated user's Java command-line flags without shell validation, concatenates them into get_start_command, and writes that string into an executable /bin/sh boot wrapper. With the shipped unix_user=root default, a user assigned only the Minecraft module can enable the boot wrapper and cause shell substitution or metacharacters in java_args to execute as root. The isolated Webmin 2.670 lab reproduced this path using a Minecraft-only delegate and a harmless id marker.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `minecraft/save_conf.cgi`, `minecraft/minecraft-lib.pl`.

WEBMIN-0596 — Minecraft module environment setting permits authenticated root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Minecraft module exposes java_envs as free text to authenticated users who have module-configuration access. Generic configuration parsing persists the text without shell validation, and get_start_command concatenates it directly ahead of the Java executable. The module's shipped configuration uses unix_user=root, and atboot.cgi writes the constructed command into an executable /bin/sh wrapper. In the isolated Webmin 2.670 lab, an account assigned only the Minecraft module persisted a harmless command substitution in java_envs and the generated wrapper executed it as UID 0. This is authenticated vertical privilege escalation and direct root command execution, not an unauthenticated issue.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `minecraft/config.info`, `config-lib.pl`.

Module · Mount (3)

WEBMIN-0313 — Authenticated root command injection through CIFS workgroup configuration

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Workgroup-mode CIFS browsing reads the free-form browse_group module configuration value and concatenates it into an nmblookup command passed to backquote_command. A Webmin user with mount-module configuration access can persist shell metacharacters in that field and trigger execution by loading the CIFS server chooser. Frozen latest-stable source retains the unsafe concatenation, and a latest-stable container reproduced a harmless marker-file command through the authenticated CGI path.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mount/smb_server.cgi`, `mount/config.info`.

WEBMIN-0677 — Delegated mountpoint delimiters append arbitrary root fstab entries

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The Linux Mount module accepts a request-derived mountpoint after checking that its string begins within a delegated directory, but it does not reject line or field delimiters. create_mount and change_mount encode literal spaces while preserving line feeds and tabs before writing the root-owned fstab. A create-capable delegated operator can therefore keep the allowed directory as a prefix and append a second fstab entry for an arbitrary mountpoint. For example, an attacker-controlled NFS export can be configured over /etc/cron.d and

subsequently processed as root by boot or a normal mount-all operation, crossing the module's directory ACL and enabling persistent root code execution.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: The host uses the Linux fstab mount-record path. [environmental]; Privileged mount or boot tooling later processes the injected fstab entry. [environmental]; The attacker controls the NFS export used for the root-execution example. [environmental]

Audited source: `mount/save_mount.cgi`, `web-lib-funcs.pl`.

WEBMIN-0965 — Unescaped mount-module arguments permit privileged shell command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The mount module constructs shell command strings from dynamic mount directories, device identifiers, filesystem types, NFS host and export values, SMB server and option fields, and swap-file paths. The reviewed hardening patch shows that several of these values entered privileged command helpers without shell escaping, and frozen-stable inspection shows raw mount tuple interpolation remains in the FreeBSD, NetBSD, OpenBSD, HP-UX, Solaris, and OSF1 mount_dir implementations. An authenticated user delegated sufficient mount-module capability can include shell metacharacters in a reachable field and cause commands to run with Webmin's operating-system privileges, extending beyond the intended filesystem-management authorization.

Profile-based footprint: 95.17% observed module signal (6,430/6,756 usable profiles; observed gate: mount).

Recorded conditions: The host uses an affected non-Linux mount backend: FreeBSD, NetBSD, OpenBSD, HP-UX, Solaris, or OSF1. [environmental]

Audited source: `mount/freebsd-lib.pl`, `mount/save_mount.cgi`.

Module · MySQL (7)

WEBMIN-0211 — Database charset and collation fields are concatenated into CREATE DATABASE

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

The database-creation handler validates the database name but concatenates submitted charset and collation tokens into CREATE DATABASE. The server-generated select list is only a UI constraint. A forged authenticated request can therefore add SQL syntax to the statement; the mysql-client execution backend consumes the composed text as a script, while other backend behavior depends on driver settings.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL delegate can create a database; stacked-statement impact uses the supported mysql-client backend. [environmental]

Audited source: `mysql/newdb.cgi`.

WEBMIN-0212 — Greater-than and less-than searches treat the operand as SQL syntax

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

The shared search builder treats the operands of its greater-than and less-than modes as SQL text. Any authorized database viewer can supply an expression instead of a numeric value, altering the generated WHERE predicate and potentially reaching broader SQL behavior available to the backend credential. This is especially relevant to constrained module roles because execute_sql_safe is used for the query.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL delegate can search an authorized database using greater-than or less-than mode. [environmental]

Audited source: mysql/view-lib.pl, mysql/view_table.cgi.

WEBMIN-0214 — Storage-engine input can add unintended mysqld configuration lines

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

The configuration-save handler accepts stor from the request and hands it to save_directive without checking the storage-engine option set. The helper embeds that value in one my.cnf line, so an embedded newline adds another directive. This crosses the intended Webmin field boundary and can change MySQL service behavior after reload or restart.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL delegate can save server configuration and the changed file is later reloaded or restarted. [environmental]

Audited source: mysql/save_cnf.cgi, mysql/mysql-lib.pl.

WEBMIN-0265 — Limited MySQL permission editors can inject administrative DELETE SQL

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N.

Bulk privilege-deletion handlers accept NUL-separated request tuples, split each tuple on spaces, check restricted-database authorization only against the parsed database component, and concatenate the tuple components into SQL executed against MySQL's grant tables. A space-free comment payload in a different component can change the DELETE predicate while the authorized database check still receives an allowed name. This crosses the per-database Webmin ACL boundary.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL delegate may edit permissions for one database but not the administrative grant tables or other databases. [nondefault]

Audited source: mysql/delete_dbs.cgi.

WEBMIN-0266 — Unvalidated table engine text permits cross-database SQL injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N.

Table creation trusts the request-supplied engine value even though the ordinary form renders a finite selector. The value is appended verbatim to CREATE TABLE SQL. In the supported non-DBI execution path, Webmin writes that SQL to a command file consumed by the MySQL client under configured administrative credentials, so a stacked statement can escape the database boundary enforced by can_edit_db.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL delegate can create tables; stacked SQL uses the supported non-DBI mysql-client path. [environmental]

Audited source: mysql/mysql-lib.pl, mysql/create_table.cgi.

WEBMIN-0374 — Field-option SQL injection can escape a database-scoped module ACL

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N.

The field-save handler authorizes only the selected database, validates a numeric width, then appends the request's opts value verbatim to the column type and sends the assembled text to execute_sql_logged. A semicolon in opts therefore creates another SQL statement. The supported non-DBI path writes the complete text to a temporary file consumed by the MySQL client, so a user scoped by Webmin to one database can

submit SQL evaluated under a broader configured database account. The actor must already have schema-edit access to at least one database; the impact is the escape from that Webmin database scope, not unauthenticated access.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL delegate has schema-edit access to one database while the configured database login can reach others. [nondefault]; Stacked SQL uses the supported non-DBI mysql-client path. [environmental]

Audited source: `mysql/save_field.cgi`, `mysql/mysql-lib.pl`.

WEBMIN-0880 — Inline SQL scripts permit MySQL client meta-commands as the Webmin worker account

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The SQL textarea recognizes multi-statement input as a script, writes it verbatim to a temporary file, and runs it through the native mysql/mariadb client. Native client source files accept meta-commands such as shell escape. Unix-user demotion is conditional on the optional buser ACL setting; without it, a Webmin worker running as root executes the meta-command as root. The frozen latest-stable container reproduced this with a harmless id command.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mysql/exec.cgi`, `mysql/mysql-lib.pl`.

Module · Net (4)

WEBMIN-0020 — VLAN identity fields allow root command execution through executable interface configuration

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The boot-interface save handler accepts VLAN physical-device and VLAN-ID fields without constraining them to interface-name and numeric syntax. Debian persistence combines those values into an interface stanza name; embedded newlines therefore create executable directives such as pre-up. An authenticated network-module user with interface-edit rights can persist a root command that runs during interface activation.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: The host uses Debian network-interface persistence. [environmental]; A later interface activation executes the injected pre-up directive. [environmental]

Audited source: `net/save_bifc.cgi`, `net/debian-linux-lib.pl`.

WEBMIN-0023 — DNS resolution-order input can inject independent nsswitch.conf directives

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The DNS order parser accepts any non-whitespace string, including URL-decoded newlines. Linux and FreeBSD nsswitch writers embed the value directly after hosts:. An authenticated DNS-client administrator can escape that field and add another name-service database directive to a root-owned system configuration file.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: The host uses a Linux or FreeBSD nsswitch.conf writer for DNS client configuration. [environmental]

Audited source: web-lib-funcs.pl, net/freebsd-lib.pl.

WEBMIN-0257 — Delegated routes editor can inject root shell commands through Red Hat IPv6 gateway configuration

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

On Red Hat-family backends that support per-interface gateways, save_routes.cgi requires the narrow routes-edit ACL but parse_routing validates only the companion IPv4 gateway. It stores request-controlled gateway6 text verbatim in IPV6_DEFAULTGW and passes the value to write_env_file. That serializer encloses whitespace-containing values in double quotes without escaping embedded quotes or newlines. A delegated routes editor can therefore terminate the assignment and insert a shell line into the root-owned /etc/sysconfig/network file. Legacy Red Hat network initialization sources this shell-style file, yielding deferred arbitrary command execution as the Webmin OS account, normally root. The path requires authentication, a routes-edit grant, a Red Hat-family installation, and a later network restart or boot.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: The host uses a Red Hat-family network backend that writes per-interface IPv6 gateways. [environmental]; A later privileged network restart or boot sources the generated configuration. [environmental]

Audited source: net/save_routes.cgi, net/redhat-linux-lib.pl.

WEBMIN-1008 — Ifupdown VLAN interface-name injection can create root-executed directives

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The Debian boot-interface request path accepts VLAN physical and VLAN ID fields without excluding line breaks, combines them into an interface identity, and passes that identity to ifupdown stanza serializers. The serializers call validate_interface_options only for option parameter/value pairs and then interpolate the independent name into an iface line. A newline in the VLAN-derived name can therefore add another directive to the root-owned interfaces file. A pre-up, up, down, or post-down directive is executable by ifupdown with root privileges when the configuration is applied. The isolated latest-stable reproduction used a harmless /bin/true marker and did not apply networking or execute the injected directive.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: The host uses Debian ifupdown boot-interface persistence. [environmental]; The saved interface configuration is applied by privileged ifupdown tooling. [environmental]

Audited source: net/save_bifc.cgi, net/debian-linux-lib.pl.

Module · Openslp (1)

WEBMIN-0351 — Delegated OpenSLP configuration permits root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

OpenSLP module configuration exposes start_cmd and stop_cmd as optional free text. The common configuration saver accepts those values for any delegated module user whose ACL does not set noconfig, and OpenSLP start.cgi and stop.cgi pass the stored strings to shell-capable execution helpers. Because Webmin normally serves these CGIs as root, such a delegated user can execute an arbitrary command as root. This audit makes no claim about the exact introduction or fix commit.

Profile-based footprint: 85.08% observed module signal (5,748/6,756 usable profiles; observed gate: openslp).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: openslp/config.info.no, openslp/start.cgi.

Module · Package Updates (1)

WEBMIN-0201 — Newline injection executes privileged YUM shell transactions during preview

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The update endpoint forwards request-derived package selectors to dependency preview without validating them against the available-update list. On YUM systems, update_system_operations writes that string verbatim after an install command in a YUM shell program. Newline characters therefore terminate the intended command and inject directives such as remove and run. Because this occurs during preview, injected transactions execute with Webmin's root privileges before the normal confirmation and package allowlist enforcement, enabling arbitrary configured-repository package installation/removal and serious system integrity or availability impact. This is YUM command-language injection, not demonstrated OS-shell RCE.

Profile-based footprint: 95.00% observed module signal (6,418/6,756 usable profiles; observed gate: package-updates).

Recorded conditions: The host uses the YUM package backend. [environmental]

Audited source: `package-updates/update.cgi`, `package-updates/package-updates-lib.pl`.

Module · PHP Configuration (4)

WEBMIN-0239 — Custom error_reporting value injects arbitrary php.ini directives

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The error settings endpoint accepts a custom error_reporting request value after checking only that it contains non-whitespace. It invokes save_directive with noquote enabled, causing embedded newlines to be written as independent php.ini lines. A delegated module user whose manual-edit ACL is disabled can therefore inject arbitrary PHP directives into an otherwise authorized configuration file. Depending on the injected directives and local application files, this can weaken PHP security controls or enable code loading in the PHP service account; it is not direct root command execution.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: PHP Configuration delegate can save error settings but is denied manual file editing. [nondefault]

Audited source: `phpini/save_errors.cgi`, `phpini/phpini-lib.pl`.

WEBMIN-0254 — PHP Configuration option values permit authenticated arbitrary directive injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

A Webmin account restricted to the PHP Configuration module can send embedded line breaks and quote delimiters in option parameters handled by save_misc.cgi. save_directive treats the value as one string but preserves embedded newlines when the line array is flushed, allowing the request to add directives beyond the UI's delegated controls. The latest-stable reproduction injected auto_prepend_file into an allowed php.ini. If a PHP runtime consumes the file and the delegate supplies a reachable PHP payload (or another executable PHP configuration target), this can become code execution in that PHP service's operating-system context. It is not inherently root execution.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: PHP Configuration delegate saves an option with embedded line breaks; code execution needs a PHP runtime that consumes the injected directive. [environmental]

Audited source: `phpini/save_misc.cgi`, `phpini/phpini-lib.pl`.

WEBMIN-0516 — PHP default-character-set newline injection permits arbitrary directive insertion

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The PHP Configuration module accepts default_charset as free text and passes it unchanged to save_directive. A value containing both a quote delimiter and line breaks can terminate the generated quoted value, insert another php.ini directive, and hide the writer's trailing quote in a comment. Because the selected file is written using the module ACL's configured Unix identity (root by default), a delegated PHP Configuration user can inject directives such as auto_prepend_file into an allowed php.ini. When a PHP web or CLI process loads that configuration and the referenced PHP file is attacker-controlled/readable, code runs as that PHP consumer account. The reproduced primitive does not by itself establish root execution.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: PHP Configuration delegate can save the default charset with a quote and newline. [environmental]

Audited source: `phpini/save_misc.cgi`, `phpini/phpini-lib.pl`.

WEBMIN-0698 — Redis session path injects arbitrary PHP directives across the manual-edit ACL

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The structured session endpoint accepts session.save_path from an authenticated user authorized for the selected PHP configuration. For Redis and Memcache it intentionally skips the filesystem-directory check, then save_directive quotes whitespace-bearing input without escaping embedded line breaks or the selected quote delimiter. A value containing a double quote, a single quote, and a newline closes the generated single-quoted value and emits a second PHP ini directive. This crosses the module's ACL boundary because save_manual.cgi separately requires the manual privilege but save_session.cgi does not. The injected directive can alter arbitrary PHP behavior; for example, auto_prepend_file pointing to attacker-controlled readable PHP causes code execution in PHP processes that load the affected configuration.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: PHP Configuration delegate can edit structured session options but is denied manual file editing. [nondefault]; The selected session backend is Redis or Memcache. [environmental]

Audited source: `phpini/save_session.cgi`, `phpini/phpini-lib.pl`.

Module · Postfix (3)

WEBMIN-0398 — Postfix recipient BCC map path command injection executes as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The authenticated Postfix BCC options handler accepts recipient_bcc_maps from a user with bcc permission and saves it before regenerating the map. get_maps_types_files permits shell metacharacters in the absolute path, and regenerate_any_table concatenates that path into a backquote_logged command without quoting. A slash-free command/redirection suffix also passes ensure_map's precursor file creation. Because Webmin normally runs this CGI as root, a crafted map path executes arbitrary commands as root. The Postfix module default ACL enables bcc access and uses dir=/, so this is reachable by an account delegated that module page, not only by an unrestricted administrator.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `postfix/save_opts_bcc.cgi`, `postfix/postfix-lib.pl`.

WEBMIN-0515 — Unquoted Postfix map paths allow privileged shell command execution during table regeneration

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Postfix module parses configurable map specifications and concatenates their path components directly into command strings used to rebuild alias and lookup tables. regenerate_aliases appends the alias map path to the configured generator command, while regenerate_any_table appends a type:path token to postmap; both invoke backquote_logged, whose implementation uses Perl shell backticks. The map parser requires an absolute path but does not reject shell metacharacters. An authenticated Webmin user entrusted with Postfix alias or relevant map administration can therefore embed a command separator in a map filename and execute an arbitrary command with the Webmin CGI service privileges, normally root. This crosses the scoped module-administration boundary.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `postfix/postfix-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0686 — SNI map path permits root shell-command injection during rebuild

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The SNI map rebuild obtains tls_server_sni_maps, accepts an absolute map path whose parser preserves shell metacharacters, and concatenates the map type and path into a single backquote_logged command. A delegated user who can configure the SNI map can append a shell command to an otherwise permitted path. Because Webmin normally performs module actions as root, the command crosses both the map-directory restriction and the intended configuration-only privilege boundary.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `postfix/postfix-lib.pl`.

Module · PostgreSQL (7)

WEBMIN-0166 — Unchecked column names permit SQL injection during field deletion

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

The multi-field deletion endpoint splits submitted d values, compares only their count with the table description, and inserts each value directly between double quotes in an ALTER TABLE command. The value need not be an actual column and embedded quotes are not escaped, so a database editor can terminate the identifier and append another SQL statement.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: PostgreSQL delegate can delete fields and supplies a crafted field name to the SQL builder. [environmental]

Audited source: `postgresql/delete_fields.cgi`, `postgresql/postgresql-lib.pl`.

WEBMIN-0167 — Plaintext password synchronization permits cross-module SQL injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

The PostgreSQL User and Groups synchronization hook builds create and alter user statements by placing the caller-provided plaintext password directly inside a single-quoted SQL literal. When synchronization is enabled,

a delegated Unix-user administrator can supply a quote, append another statement, and comment out the generated suffix, causing SQL to run as the global PostgreSQL module login.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: Unix-user synchronization to PostgreSQL is enabled for a delegated account administrator. [nondefault]

Audited source: `postgresql/useradmin_update.pl`.

WEBMIN-0210 — Delegated PostgreSQL user editor SQL injection reaches server-account command execution

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:L/VA:L/SC:N/SI:N/SA:N`.

An authenticated Webmin user allowed to manage PostgreSQL users can supply a quote and additional statements in edit_user.cgi's user parameter. Webmin 2.670 places that value directly inside a pg_shadow SELECT and execute_sql_safe sends the resulting statement as the configured PostgreSQL administrator. In the shipped superuser configuration, a stacked COPY TO PROGRAM statement runs a command as the PostgreSQL server OS account. This does not directly execute as root, but it crosses a delegated Webmin ACL into database-superuser control and local postgres-account command execution.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: The configured PostgreSQL administrator role has superuser authority and the server supports COPY TO PROGRAM. [environmental]

Audited source: `postgresql/edit_user.cgi`, `postgresql/postgresql-lib.pl`.

WEBMIN-0319 — Request-controlled PostgreSQL username alters the edit-user lookup query

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

The PostgreSQL edit-user handler checks the module's users capability, but it concatenates the request-controlled user parameter between single quotes in a pg_shadow query and calls execute_sql_safe without bound parameters. A crafted value can change the SELECT, including adding a UNION projection whose values are rendered into the edit form. This crosses the distinct delegated users capability into the privileges of the configured PostgreSQL login.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: PostgreSQL delegate has users permission and supplies a crafted username to the edit-user lookup. [environmental]

Audited source: `postgresql/edit_user.cgi`.

WEBMIN-0320 — Bulk PostgreSQL role deletion allows SQL statement injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

The PostgreSQL bulk-user deletion handler requires the users capability but accepts each request-controlled d value without role-name validation, disables the library's PostgreSQL escaping, places the value inside a double-quoted identifier, and calls execute_sql_logged. A crafted double quote terminates the identifier and permits additional SQL in the same command. The added SQL runs with the module's configured database identity, which may have privileges beyond delegated role management.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: PostgreSQL delegate has users permission and supplies a quote-bearing role name to bulk deletion. [environmental]

Audited source: `postgresql/delete_users.cgi`.

WEBMIN-0554 — Delegated User and Groups password SQL injection creates a PostgreSQL superuser

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

When PostgreSQL account synchronization is enabled, the User and Groups module passes an operator-controlled plaintext password to postgresql/useradmin_update.pl. That hook executes with the configured global PostgreSQL login and inserts the password verbatim between SQL string quotes. An authenticated operator allowed to create or modify Unix users and invoke other-module hooks can close the password literal and append PostgreSQL role options. In the frozen 2.670 lab, the exact hook emitted a CREATE USER statement containing an injected SUPERUSER option, PostgreSQL accepted it, and the created test role had rolsuper=true. This grants full PostgreSQL cluster control and can expose server-side PostgreSQL superuser capabilities; it does not by itself establish root execution on the host.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: PostgreSQL synchronization from User and Groups is enabled for a delegated account operator. [nondefault]; The configured PostgreSQL login has enough authority to create a superuser role. [environmental]

Audited source: `postgresql/useradmin_update.pl`, `useradmin/save_user.cgi`.

WEBMIN-0797 — Authenticated SQL injection in PostgreSQL index metadata lookup

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

edit_index.cgi passes the request-supplied index value to index_structure after module ACL checks. index_structure inserts that value directly inside a single-quoted pg_indexes query and sends the resulting string to execute_sql_safe. A quote-bearing value can therefore change the SQL executed with Webmin's configured PostgreSQL credential. In the isolated 2.670 reproduction, a UNION payload supplied controlled table and index-definition values that the stable function accepted and parsed. Exploitation requires an authenticated user with PostgreSQL database and index-management access, but it can exceed that operation-specific authorization boundary.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: PostgreSQL delegate can edit indexes and supplies a quote-bearing index name. [environmental]

Audited source: `postgresql/postgresql-lib.pl`.

Module · Proftpd (1)

WEBMIN-0680 — Delegated ProFTPD configuration enables root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Webmin exposes proftpd_path as a free-text module configuration setting to authenticated ProFTPD-module users whose ACL permits module-configuration editing. get_proftpd_version concatenates that setting with fixed arguments and passes the complete string to backquote_command, which ultimately evaluates it through shell backticks. A module-scoped delegated user can therefore save shell metacharacters and trigger the apply path to execute an arbitrary command with Webmin's service privileges. In the frozen 2.670 lab the restricted account was denied the Webmin Users module, yet its payload executed as uid=0. The path requires valid Webmin credentials, ProFTPD module assignment, and noconfig=0.

Profile-based footprint: 5.28% observed module signal (357/6,756 usable profiles; observed gate: proftpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `proftpd/config.info`, `config_save.cgi`.

Module · Raid (5)

WEBMIN-0045 — Authenticated RAID disk parameter command injection executes as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

In mdadm mode, raid/save_raid.cgi forwards request-controlled disk and rdisk fields to add_partition and remove_partition without validating that the submitted value belongs to the UI-generated device set. raid/raid-lib.pl concatenates that value into shell command strings for mdadm --add, --fail, and --remove. A forged value can terminate the intended command and run an arbitrary command as the root Webmin process. Frozen Webmin 2.670 retained the flow, and the latest-stable container probe executed id and recorded uid=0(root).

Profile-based footprint: 85.98% observed module signal (5,809/6,756 usable profiles; observed gate: raid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `raid/save_raid.cgi`, `raid/raid-lib.pl`.

WEBMIN-0158 — RAID request fields enable authenticated root OS command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The RAID module relies on UI selects rather than server-side validation for values such as grow count, conversion count, and layout. save_raid.cgi/create_raid.cgi pass those request values into grow(), convert_raid(), and make_raid(), which concatenate them into shell command strings executed by backquote_logged. An authenticated account granted the RAID module can submit shell syntax directly and execute arbitrary commands with Webmin's privileges, normally root. This remains present in Webmin 2.670 and was reproduced with inert id commands from the exact stable functions.

Profile-based footprint: 85.98% observed module signal (5,809/6,756 usable profiles; observed gate: raid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `raid/save_raid.cgi`, `raid/raid-lib.pl`.

WEBMIN-0455 — Authenticated RAID chunk-size injection executes commands as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The RAID creation endpoint accepts a request-controlled chunk value and make_raid concatenates that value unquoted into an mdadm command executed by Webmin. A Webmin account authorized for the RAID module can therefore inject shell syntax and execute arbitrary commands as root. This is authenticated command execution, not an unauthenticated RCE.

Profile-based footprint: 85.98% observed module signal (5,809/6,756 usable profiles; observed gate: raid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `raid/create_raid.cgi`, `raid/raid-lib.pl`.

WEBMIN-0473 — Authenticated RAID action parameters allow root-context OS command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The RAID action handler accepts disk identifiers and disk-count values from authenticated requests and passes them unchanged to grow, convert_raid, remove_partition, and replace_partition. Those helpers concatenate the values into command strings executed by backquote_logged through a shell. An authenticated user who is authorized for the RAID module can place shell metacharacters in a submitted value and execute arbitrary

commands with the Webmin service's operating-system privileges, normally root. The UI's select lists and removal confirmation do not provide server-side value validation. The frozen 2.670 runtime probe exercised the actual grow and replace helpers and created benign UID markers containing 0.

Profile-based footprint: 85.98% observed module signal (5,809/6,756 usable profiles; observed gate: raid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: raid/save_raid.cgi, raid/raid-lib.pl.

WEBMIN-0484 — Authenticated RAID device injection executes commands as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

create_raid.cgi places the request's device parameter into raid.value without validating it as an md device. In mdadm mode, make_raid concatenates raid.value into a command executed by backquote_logged, the UUID workflow calls shell-form open on mdadm --detail with the same value, and readwrite_raid has another shell-form interpolation. Because Webmin module CGIs run with root privilege, an authenticated RAID-module user who tampers with the hidden device field can execute arbitrary commands as root. The same unvalidated value is subsequently appended to the root-owned mdadm configuration. The frozen Webmin 2.670 lab reproduced the request as uid=0(root).

Profile-based footprint: 85.98% observed module signal (5,809/6,756 usable profiles; observed gate: raid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: raid/create_raid.cgi, raid/raid-lib.pl.

Module · Samba (4)

WEBMIN-0057 — Delegated Samba group editing executes injected shell commands as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The Samba save_group.cgi endpoint requires maint_groups but accepts the submitted type field without constraining it to the form's expected one-character choices. When the configured smbgroupedit program exists, create_group and modify_group append that value unquoted to a command string executed by backquote_logged. A Samba-only delegated account can therefore submit shell metacharacters and run commands with Webmin's root privileges. The condition is specific to installations retaining the legacy smbgroupedit path; the net fallback quotes the value.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: The Samba installation uses the legacy smbgroupedit program rather than the quoted net fallback. [environmental]

Audited source: samba/save_group.cgi, samba/samba-lib.pl.

WEBMIN-0248 — Delegated Samba user editing permits root command execution through opts

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The Samba save_euser.cgi endpoint requires the view_users and maint_users module ACLs but accepts every submitted opts value without restricting it to known one-character account flags. When pdbedit is available, modify_user joins those values inside the shell fragment -c '[...]' and passes the whole command to backquote_logged while Webmin is running as root. A crafted opts value containing a single quote and shell separators escapes the intended pdbedit argument. In the frozen Webmin 2.670 lab, an authenticated request caused id to run as uid=0(root).

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: The Samba backend has the pdbedit executable available. [environmental]

Audited source: samba/save_euser.cgi, samba/samba-lib.pl.

WEBMIN-0814 — Authenticated Samba sync setting permits root command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The Samba synchronization settings endpoint permits any non-whitespace group SID/RID and stores it as sync_gid. During pdbedit-backed Samba user creation, create_user concatenates this value directly into a command string that backquote_logged executes through a shell. Because maint_sync is an independently assignable Samba module ACL, an authenticated principal limited to synchronization management can supply shell metacharacters that execute with the Webmin service's operating-system privileges when synchronized user creation occurs. The frozen 2.670 source retains the full data flow, and the required latest-stable Docker test reproduced marker creation through the actual frozen create_user and backquote_logged routines.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: samba/save_sync.cgi, samba/samba-lib.pl.

WEBMIN-2462 — Samba share-name validation permits line breaks in section headers

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

Both Samba share save handlers accept decoded names against /^[\p{L}\p{N}_$-.\s]+$/. Perl's \s includes line breaks, while create_share and modify_share interpolate the original, unnormalized name into an smb.conf section header. An authenticated Webmin user who is allowed by the relevant Samba module ACL to create or modify a share can therefore submit an embedded-line-break name and make Webmin write a split or malformed header. This can corrupt the intended Samba configuration or impair service availability.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: Samba delegate can create or rename shares and submits a line-break-bearing share name. [environmental]

Audited source: samba/save_fshare.cgi, samba/save_pshare.cgi.

Module · Sarg (1)

WEBMIN-0500 — Authenticated Sarg report generation permits root command injection through log filenames

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

Sarg's configuration UI allows a module user to select an arbitrary readable access-log path. Report generation enumerates matching filenames, interpolates them without shell quoting into one command string, and executes that string through Perl's pipe form while Webmin runs as root. An authenticated Sarg delegate who can create a readable local filename containing shell syntax can therefore execute arbitrary root commands. The requirement for both Sarg-module access and local file creation makes this conditional rather than an unauthenticated or default remote compromise, but successful exploitation crosses directly to root.

Profile-based footprint: 85.20% observed module signal (5,756/6,756 usable profiles; observed gate: sarg).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: sarg/save_log.cgi, sarg/sarg-lib.pl.

Module · Sendmail (2)

WEBMIN-0041 — Network-ports ACL allows arbitrary Sendmail configuration-directive injection

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

save_ports.cgi authorizes a caller using only the network-ports ACL, validates the visible port name/address/number, but copies mod_i and other_i request fields into a DaemonPortOptions string without rejecting control characters. save_directives then writes that string into root-managed sendmail.cf without removing embedded newlines and the endpoint restarts Sendmail. A restricted authenticated user can therefore terminate the intended option line and add independent Sendmail directives, gaining configuration authority far beyond network ports. Arbitrary Sendmail configuration can alter exposure, relaying, routing, mailers, and command-backed delivery behavior. The reproduction demonstrates directive injection without executing a command payload.

Profile-based footprint: 85.49% observed module signal (5,776/6,756 usable profiles; observed gate: sendmail).

Recorded conditions: Sendmail delegate has network-port editing permission while other configuration controls are withheld. [nondefault]

Audited source: `sendmail/save_ports.cgi`, `sendmail/sendmail-lib.pl`.

WEBMIN-0156 — Selected Sendmail queue flush allows authenticated root command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The selected-message flush handler parses the request-controlled file parameter and checks only the dedicated flushq module ACL. It does not verify that the submitted path is under a configured mail-queue directory or that its basename is a valid Sendmail queue identifier. After mail_from_queue preserves the path, del_mailqs.cgi strips directory components and concatenates the basename directly after -qI in a command string. proc::safe_process_exec_logged ultimately runs that string through /bin/sh -c with uid and gid zero. An authenticated account granted flushq, combined with the ability to place a readable file carrying shell metacharacters in its basename, can therefore execute arbitrary commands as root. The frozen 2.670 lab reproduced the issue through the actual authenticated HTTP handler.

Profile-based footprint: 85.49% observed module signal (5,776/6,756 usable profiles; observed gate: sendmail).

Recorded conditions: The caller can place a readable file whose basename contains shell metacharacters. [environmental]

Audited source: `sendmail/del_mailqs.cgi`, `mailboxes/boxes-lib.pl`.

Module · Shorewall (1)

WEBMIN-0072 — Delegated Shorewall params editing permits root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Shorewall save route allows an authenticated module user whose file ACL includes params to supply a parameter value. params_validate restricts only the variable name and emits the value unquoted into /etc/shorewall/params, a shell-style file consumed by Shorewall. The same module exposes check.cgi, which invokes Shorewall check without the apply/start nochange restriction, giving the delegated user an immediate trigger. The default Shorewall ACL includes all files. Frozen Webmin 2.670 retains the complete chain. In the controlled no-network lab, the exact stable validator serialized a harmless command substitution and shell consumption recorded uid=0(root).

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: shorewall).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: shorewall/shorewall-lib.pl, shorewall/save.cgi.

Module · Shorewall6 (1)

WEBMIN-0343 — Authenticated Shorewall6 table traversal enables root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Shorewall6 comment-saving endpoint accepts request-controlled table and msg values. Its only table authorization check returns true for every string under the wildcard files ACL and does not enforce the nochange read-only flag. The table value is appended directly to config_dir for locking and writing, without canonicalization, '..' rejection, or membership in the known Shorewall6 table list. The comment message is prefixed with COMMENT but embedded newlines are retained when Webmin's table cache flushes the row. An authenticated Shorewall6 module user can therefore traverse to a path such as an /etc/cron.d file and place a controlled cron line after the prefixed first line, yielding root command execution. The locked Webmin 2.670 reproduction harmlessly demonstrated the actual stable helper creating a controlled second line outside config_dir; it did not execute any command.

Profile-based footprint: 84.72% observed module signal (5,724/6,756 usable profiles; observed gate: shorewall6).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Module · Smart Status (1)

WEBMIN-0052 — SMART module extra arguments permit authenticated root command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The SMART module exposes config extra as free-form smartctl arguments, then concatenates it directly into shell-evaluated command strings used by drive status and self-test operations. Root config.cgi and config_save.cgi permit an authenticated user assigned the module to edit its configuration unless noconfig is set. Shell metacharacters in extra therefore execute commands in the Webmin worker context, normally UID 0. A network-disabled 2.670 container proof placed a harmless id command in extra, invoked get_drive_status, and recorded uid=0(root).

Profile-based footprint: 4.23% observed module signal (286/6,756 usable profiles; observed gate: smart-status).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: smart-status/config.info, smart-status/smart-status-lib.pl.

Module · Smf (1)

WEBMIN-0134 — Authenticated SMF state target command injection executes as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The SMF index accepts applyto as request data, splits it into FMRI values, and passes the values to svc_state_cmd. That function appends each FMRI without quoting to a privileged svcadm or svccfg command string. run_smf_cmds then treats semicolons as command separators and executes each command through backquote_logged. A Webmin user authorized for the SMF module can therefore place shell syntax in applyto and execute arbitrary commands as the Webmin process, normally root. The module declares Solaris 10 support, so exploitability requires that module and platform; referer checking limits cross-origin delivery but not an authenticated crafted request.

Profile-based footprint: 84.92% observed module signal (5,737/6,756 usable profiles; observed gate: smf).

Recorded conditions: The host supports and runs the Solaris 10 SMF module backend. [environmental]

Audited source: smf/index.cgi, smf/smf-lib.pl.

Module · Software (2)

WEBMIN-0490 — pkgsrc package and search parameters are interpolated into executable SQLite statements

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N.

The pkgsrc backend builds SQL by placing caller-supplied package names directly in an IN list and search text directly in two LIKE strings. execute_pkgin_sql then invokes the sqlite3 CLI with the complete statement. An authenticated Software Packages user on a pkgsrc-configured host can terminate the intended literal and submit additional SQLite statements under the Webmin process's database permissions. The search route is nominally a package lookup, while both routes can read or modify pkgin metadata. The constrained latest-stable reproduction inserted harmless markers through both actual functions into a disposable database.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: Software Packages runs the pkgsrc backend and an authenticated module user controls package or search text. [environmental]

Audited source: software/pkgsrc-lib.pl.

WEBMIN-0909 — Insufficient PHP module-name validation permits shell command injection in RPM queries

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The PHP module installation handler permits a module value after only a non-empty/trim check in latest stable, and php_module_packages embeds each token into RPM-style package names. software/rpm-lib.pl then places those names directly into shell command strings in package_info and virtual_package_info. open_execute_command executes those strings through a shell-backed pipe. On an RPM-selected host, an authenticated principal with global PHP Configuration access can therefore use shell metacharacters in the module field to run commands with Webmin's root privileges. The frozen latest-stable container reproduction invoked the actual RPM virtual-package function with the same derived package-name shape and confirmed creation of a UID-0 marker under an isolated, network-disabled lab configuration.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: The host uses the RPM-selected PHP module package path. [environmental]

Audited source: phpini/install_mod.cgi, phpini/phpini-lib.pl.

Module · Spam (3)

WEBMIN-0339 — Delegated SpamAssassin manual editing can escalate a low-privilege plugin to root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The SpamAssassin module's default ACL includes the manual capability. save_manual.cgi accepts request data and replaces a parsed SpamAssassin configuration file while Webmin is running as root. SpamAssassin supports loadplugin with a filesystem path; consequently, an authenticated delegated module user who can plant a readable Perl plugin can reference it from the replaced root-owned configuration and obtain root-context Perl execution when SpamAssassin next loads the configuration. The 2.670 Docker run used the exact frozen endpoint, wrote a root-owned local.cf, referenced a plugin owned by UID/GID 65534, and observed ruid=0 euid=0 from that plugin.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: spam).

Recorded conditions: The delegated account can place a readable Perl plugin on the host. [environmental]; SpamAssassin later loads the changed configuration and referenced plugin. [environmental]

Audited source: spam/defaultacl, spam/save_manual.cgi.

WEBMIN-0612 — Delegated Amavisd form permits executable Perl configuration injection

CVSS 4.0 Base (CVSS-B): 8.6 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The Amavisd save CGI is guarded by the separately delegable amavisd feature, then reads directive values from the request and sends them to save_amavis_directive. The validator only examines a value when its second argument equals the string '*', but the CGI consistently passes numeric 1, so arbitrary text is accepted. For existing computed directives the writer places the value directly after the Perl assignment operator; for string directives it interpolates into double quotes without full Perl-literal escaping. Consequently, a user authorized for the limited Amavisd settings page can write executable Perl syntax to Amavisd's configuration, crossing the intended feature-ACL boundary. Evaluation or reload of that configuration executes the injected statements in the daemon startup context.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: spam).

Recorded conditions: SpamAssassin delegate has the optional Amavisd feature permission. [nondefault]; The edited Amavis configuration is later loaded by the service. [environmental]

Audited source: spam/spam-amavis-lib.pl, spam/save_amavisd.cgi.

WEBMIN-0903 — Restricted header editor can inject additional SpamAssassin directives

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

The SpamAssassin header editor is independently grantable through the module ACL. MIME-header rows are passed to header_parser, but its ordinary-operator branch does not restrict the operator field to the UI choices. Webmin's request parser retains percent-decoded newlines, so one crafted row can become multiple configuration lines when save_directives persists it. This lets an authenticated principal limited to header editing write unrelated SpamAssassin directives and weaken filtering policy; the isolated latest-stable test persisted required_score 999 as a separate line.

Profile-based footprint: 96.64% observed module signal (6,529/6,756 usable profiles; observed gate: spam).

Recorded conditions: SpamAssassin delegate can edit MIME headers but is denied broader rule changes. [nondefault]

Audited source: spam/edit_header.cgi, spam/save_header.cgi.

Module · Squid (4)

WEBMIN-0238 — COSS cache path newline injection crosses Squid module permissions

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

The Squid cache-options save endpoint accepts request-controlled cache type and path values. When the type is coss, it skips the directory check and passes the path unchanged to the generic squid.conf serializer. A copts-authorized user can include a newline and a hash terminator in that path so the serialized cache_dir line is followed by an independent directive such as http_access allow all. Because copts and actrl are separately delegable module permissions, this permits configuration changes outside the user's intended scope. Applying the injected policy requires a later Squid configuration reload.

Profile-based footprint: 85.82% observed module signal (5,798/6,756 usable profiles; observed gate: squid).

Recorded conditions: Squid delegate can edit cache options but is denied access-control policy editing. [nondefault]; Injected Squid policy takes effect after a configuration reload. [environmental]

Audited source: squid/save_cache.cgi, squid/parser-lib.pl.

WEBMIN-0444 — Squid effective identity enables shell injection in ownership change

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The administrative form stores effective user and group values without shell-safe validation. The ownership workflow later reads those values and interpolates them into a one-string chown system call, allowing an authorized Squid-options user to inject commands into the normally root Webmin CGI context.

Profile-based footprint: 85.82% observed module signal (5,798/6,756 usable profiles; observed gate: squid).

Recorded conditions: An authorized Squid options editor controls effective user or group text used in a later ownership change. [environmental]

Audited source: `squid/save_admin.cgi`, `squid/chown.cgi`.

WEBMIN-0447 — Calamaris log filename enables root-context shell injection

CVSS 4.0 Base (CVSS-B): 7.0 (High) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Calamaris accepts every log-directory entry whose basename starts with access.log and interpolates its pathname into shell-form decompression or tail pipelines. A log writer able to create a metacharacter-bearing filename can execute a command in the Webmin CGI context when an authorized user generates the report.

Profile-based footprint: 85.82% observed module signal (5,798/6,756 usable profiles; observed gate: squid).

Recorded conditions: An attacker can create a metacharacter-bearing access.log filename and an operator generates Calamaris output. [environmental]

Audited source: `squid/calamaris.cgi`.

WEBMIN-0482 — Stored Squid cache path permits root shell command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The cache ownership helper collects paths from parsed Squid configuration and builds one shell command with system(STRING). For Squid 2.3 and later, cache_dir values[1] is appended without quoting or argument separation. A user who can control that configuration and invoke the permission-gated rebuild operation can place shell metacharacters in the path; command substitution is evaluated by the root Webmin process before chown runs. The frozen latest-stable lab confirmed the behavior with a harmless marker.

Profile-based footprint: 85.82% observed module signal (5,798/6,756 usable profiles; observed gate: squid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `squid/squid-lib.pl`, `squid/init_cache.cgi`.

Module · T (1)

WEBMIN-1007 — Compile test executes shell substitutions from repository filenames

CVSS 4.0 Base (CVSS-B): 7.0 (High) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The repository compile test walks the checkout for Perl source files and interpolates each repository-controlled filename into a string-form qx command. Because qx invokes a shell and the filename is merely enclosed in double quotes, shell command substitutions in a crafted filename are evaluated. The risk applies to developers and CI workers that execute the test on source supplied by an untrusted contributor; it does not require the Webmin service to be running. Frozen latest-stable source retains the sink, and the constrained Docker lab used the actual stable test with a harmless touch sentinel, which was created.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: A developer or CI worker runs the compile test on a checkout with a shell-substitution-bearing Perl filename. [environmental]

Audited source: t/compile.t.

Module · Time (1)

WEBMIN-0535 — Free-text hwclock flags allow delegated Time users to execute root shell commands

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The System Time module exposes hwclock_flags as a type-10 option that accepts free text. Webmin's configuration parser stores the free-text branch unchanged, and the module returns that value unchanged before interpolating it into backquote_command/backquote_logged shell strings. Module configuration access is available when noconfig is false, which is the module default, and simply rendering the Time page calls get_hardware_time when hwclock is supported. A delegated authenticated module user can therefore save shell metacharacters and execute arbitrary commands with the Webmin daemon's privileges, normally root. Frozen stable 2.670 source has the same flow, and an isolated Docker probe of its exact get_hardware_time function recorded effective UID 0 from an injected command.

Profile-based footprint: 5.37% observed module signal (363/6,756 usable profiles; observed gate: time).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: time/config.info, config-lib.pl.

Module · Useradmin (1)

WEBMIN-0624 — Authenticated option injection selects alternate login-accounting files

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:H/VA:N/SC:N/SI:N/SA:N.

The Linux recorded-logins endpoint accepts a request-supplied username when the module ACL grants logins=*. open_last_command applies quotemeta, which prevents shell metacharacter execution, but passes the resulting token to last without --. A username such as -f/path is therefore interpreted as last's alternate-file option. Because the CGI and last process run with Webmin service privileges, the authenticated module user can select a root-readable wtmp-format file and receive records that last can parse. The latest-stable reproduction used a mode-0600 controlled fixture and observed its unique host only in the crafted request, not in the benign control. The impact is constrained to formats and behaviors understood by last rather than arbitrary byte-for-byte file disclosure.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: User and Groups delegate has logins=* permission. [nondefault]; The host uses Linux last and the request supplies an option-shaped username. [environmental]

Audited source: useradmin/linux-lib.pl, useradmin/list_logins.cgi.

Module · Usermin (1)

WEBMIN-0967 — Usermin module archives can inject commands through top-level entry names

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The reviewed install_usermin_module context parses any non-slash top-level archive entry into a module name and inserts that value twice, without shell escaping, into a string-form system command used to extract module.info. The sink executes before valid module metadata is required. Additional commands later derive removal paths, find/chown paths, and copyconfig arguments from the same names. A Webmin principal with Usermin module-install permission can therefore use a crafted archive entry name to execute commands with Webmin process privileges. Frozen 2.670 retains the line-560 sink, and the latest-stable lab executed id as root through it.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: usermin/install_mod.cgi, usermin/usermin-lib.pl.

Module · Webalizer (1)

WEBMIN-0074 — Unquoted per-log config path permits root command injection

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

Webalizer derives a per-log configuration filename directly from the selected logfile path, replacing slashes but preserving shell metacharacters. generate_report escapes its output-directory and logfile arguments, yet inserts the readable config pathname into the shell command without quoting before open_execute_command executes it. A delegated Webalizer user who can place or select a readable metacharacter-bearing logfile, enable the per-log config, and choose or inherit root as the report user can execute arbitrary commands as root during report generation.

Profile-based footprint: 85.60% observed module signal (5,783/6,756 usable profiles; observed gate: webalizer).

Recorded conditions: The vulnerable per-log configuration-path feature must be enabled. [nondefault]

Audited source: webalizer/webalizer-lib.pl, webalizer/save_log.cgi.

Module · Xterm (1)

WEBMIN-0904 — Stored xterm configuration injection can drive a root terminal WebSocket

CVSS 4.0 Base (CVSS-B): 8.5 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N.

The xterm module stores a nominally numeric fontsize setting in shared module configuration and interpolates it verbatim into the JavaScript object passed to new Terminal(). The UI presents fixed choices, but Webmin's type-4 parser directly accepts any posted value. An authenticated xterm user who was also delegated module-configuration access can therefore store a JavaScript expression. On a later non-XHR xterm response, that expression executes in the visiting user's Webmin origin and inside the callback that owns the terminal WebSocket. If the visitor is a full administrator, the default xterm ACL starts a root terminal, so an expression can call socket.send() with shell commands. The path requires explicit configuration delegation and a privileged victim visit.

Profile-based footprint: 76.87% observed module signal (5,193/6,756 usable profiles; observed gate: xterm).

Recorded conditions: Xterm user can edit module configuration and store an arbitrary fontsize value. [nondefault]; A privileged administrator later opens the affected terminal page. [environmental]

Audited source: xterm/index.cgi, xterm/config.info.

Module · Zones (1)

WEBMIN-0192 — Solaris zone path permits root command injection after install

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The Solaris Zones creation handler accepts any absolute zone path and later interpolates that path without shell quoting into a scalar system_logged command intended to remove a post-install marker. Perl system() invokes a shell for this scalar command. An authenticated Zones-module user can place shell metacharacters in the path and execute arbitrary commands as the root Webmin worker when a successful installation reaches the .UNCONFIGURED branch. The frozen 2.670 source retains the data flow, and the latest-stable image reproduced it with a harmless id marker reporting uid=0(root).

Profile-based footprint: 0.64% observed module signal (43/6,756 usable profiles; observed gate: zones).

Recorded conditions: The host uses the Solaris Zones backend. [environmental]; The installation succeeds and reaches the post-install marker cleanup branch. [environmental]

Audited source: zones/forms-lib.pl, zones/create_zone.cgi.

File and path handling (44)

Core · Runtime (3)

WEBMIN-0417 — Privileged lock-file creation follows an attacker-replaced symlink

CVSS 4.0 Base (CVSS-B): 5.8 (Medium) · CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

lock_file uses a predictable sibling path, unlinks it, and then opens it with a truncating write. When an unprivileged user controls the containing directory, that user can replace the path with a symlink in the gap and redirect the privileged write. The frozen latest-stable Docker reproduction synchronized the real unlink/open sequence and changed a root-owned sentinel file to the privileged process ID.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: An unprivileged local user controls the directory containing the predictable lock file. [environmental]

Audited source: web-lib-funcs.pl.

WEBMIN-0633 — File-diff snapshot uses unsafe permissions and path recreation

CVSS 4.0 Base (CVSS-B): 2.0 (Low) · CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N.

During action-log diff generation, Webmin writes prior file contents to realfile.webminorig after setting umask(0700). Applied to Perl's default 0666 create mode, that mask yields 0066: the owner receives no permission while group and other retain read and write access. The preceding unlink followed by an ordinary path-based open also leaves a replacement-symlink race when the target directory is attacker-writable. A local user may disclose or alter the transient snapshot, and under the race precondition may redirect the privileged write.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: Action-log file diffs run in a directory writable by a local adversary or readable by others. [environmental]

Audited source: web-lib-funcs.pl.

WEBMIN-2466 — Reversible-encryption API exposes secrets in temporary files and command metadata

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

encrypt_phrase writes plaintext to Webmin temporary storage, passes the passphrase in an OpenSSL -pass command argument, and invokes the command through backquote_logged. decrypt_phrase uses the same passphrase channel. The latest-stable implementation defaults execution to nobody, requiring the source file to be readable by that account, and the command logging path receives the complete command string. A constrained latest-stable run observed the synthetic passphrase in process arguments and a mode-0644 plaintext temporary file readable as nobody while encryption succeeded.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: A caller uses encrypt_phrase or decrypt_phrase while a local observer can read the temporary file, command arguments, or action log. [environmental]

Audited source: web-lib-funcs.pl.

Core · Webmin (1)

WEBMIN-0256 — Delegated Webmin configuration actions can overwrite root-owned paths

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

Two delegated Webmin Configuration actions trust authenticated request fields as root filesystem destinations. CSR generation accepts complete key and CSR paths and writes generated material there; clone-title saving accepts a hidden clone identifier and interpolates it into a path beneath the configuration directory without containment validation. A principal limited to the webmin module and the ssl/descs pages can write outside /etc/webmin, producing root-owned file replacement with constrained content and causing host integrity or availability loss.

Profile-based footprint: 5.39% observed module signal (364/6,756 usable profiles; observed gate: webmin).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `webmin/newcsr.cgi`, `webmin/webmin-lib.pl`.

Module · Backup Config (2)

WEBMIN-0138 — Failed local backups retain umask-readable archive content

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

Local backup output is opened by the archive shell command under the process umask, but owner-only mode is assigned only after a successful command. When the archive pipeline fails, execute_backup returns while retaining local output. With a permissive configured or inherited umask, a partial archive containing sensitive configuration data can remain readable by non-root users. A frozen latest-stable Docker test forced a controlled post-redirection gzip failure and observed a retained mode-0644 file containing the synthetic secret marker.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: backup-config).

Recorded conditions: A local backup fails after writing the archive, with a permissive umask and a local reader. [environmental]

Audited source: `backup-config/backup-config-lib.pl`.

WEBMIN-0713 — Concurrent configuration backups can share and overwrite one temporary archive

CVSS 4.0 Base (CVSS-B): 2.0 (Low) · `CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

For non-local backup destinations, execute_backup creates the archive at a path returned by transname_timestamped. That helper formats only the current local time to second resolution and passes the resulting basename to transname; the supplied-filename path omits the random, PID, and counter components used by the no-filename branch. Backup operations sharing a Webmin temporary directory during the same second can therefore target the same file. Their shell redirections, permission changes, uploads, and unlinks can interleave, allowing one operation's archive content to replace or be consumed by another operation. A cross-user confidentiality consequence is configuration-dependent, but archive corruption and request interference follow from the shared path.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: backup-config).

Recorded conditions: Two non-local configuration backups share a temporary directory and begin within one second. [environmental]

Audited source: `backup-config/backup-config-lib.pl`, `web-lib-funcs.pl`.

Module · Filemin (1)

WEBMIN-0887 — Directory-based Unix-user selection uses a non-canonical path

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The per-directory work_as_dir selector evaluates the raw request path and falls back to work_as_user when no mapping matches. is_under_directory rejects any raw path containing dot-dot, but get_paths later simplifies that same path before its allowed-path decision. A path that canonicalizes inside a mapped directory can therefore miss the mapping and execute as the fallback Unix user, which may be more privileged.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: Filemin work_as_dir mapping differs from fallback work_as_user. [nondefault]; The raw path misses the mapping but canonicalizes inside its directory. [environmental]

Audited source: `filemin/filemin-lib.pl`, `web-lib-funcs.pl`.

Module · Grub2 (1)

WEBMIN-1011 — Background import can republish root-readable files as mode 0644

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

save_theme.cgi requires only the GRUB 2 edit ACL and sends the submitted background path to grub2_install_background_source. The stable validator checks an absolute readable path and an image-like suffix but follows symlinks. The copy helper resolves the submitted path with abs_path, reads the resolved target as the Webmin process, derives the destination basename from the attacker-supplied path, and applies mode 0644 to the result. A delegated editor with a local symlink-creation capability can therefore turn a root-readable file into a world-readable copy in the GRUB background directory. The isolated stable-image lab confirmed a mode-0600 synthetic source was copied through leak.png and read successfully after dropping privileges to UID 65534.

Profile-based footprint: 0.01% observed module signal (1/6,756 usable profiles; observed gate: grub2).

Recorded conditions: The same attacking principal can create a local symlink with an allowed image suffix. [environmental]

Audited source: `grub2/save_theme.cgi`, `grub2/grub2-lib.pl`.

Module · Help (1)

WEBMIN-2401 — Help module traversal discloses constrained root-readable HTML

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

An authenticated Webmin account allowed to use only the Help module can submit a selected-module value containing parent-directory components. The stable search handler passes that value to module_root_directory, enumerates a derived help subdirectory, and reads a second derived .html path without checking that either remains under an installed module. Because Webmin runs as root, a compatible root-readable HTML file can be rendered to the delegate. Exploitation is constrained: the target must have an .html suffix, contain the expected header structure and search term, and have a matching basename in the separately derived help directory. The stable lab reproduced disclosure of a root:root mode-0600 inert marker with a Help-only account.

Profile-based footprint: 0.01% observed module signal (1/6,756 usable profiles; observed gate: help).

Recorded conditions: A Help module user supplies a traversing module name resolving to a readable HTML file. [environmental]

Audited source: `help/search.cgi`, `web-lib-funcs.pl`.

Module · Htaccess Htpasswd (1)

WEBMIN-0107 — Webmin-created password files default to world-readable mode

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The htaccess-htpasswd module uses oct($config{'perms'}) || 0644 when it creates the users file and optional groups file. Because 0644 grants read permission to users outside the owner and group, a local unprivileged account that can traverse the parent directory can obtain password-verification material for offline guessing. The risk is conditional on local access and directory traversal, and an administrator can mitigate it by configuring a restrictive mode such as 0600; nevertheless, the default behavior exposes sensitive authentication data.

Profile-based footprint: 5.42% observed module signal (366/6,756 usable profiles; observed gate: htaccess-htpasswd).

Recorded conditions: Webmin creates a password file under a directory traversable by local users; the default umask permits mode 0644. [environmental]

Audited source: `htaccess-htpasswd/config.info`, `htaccess-htpasswd/save_dir.cgi`.

Module · Kea Dhcp (1)

WEBMIN-1009 — Manual Kea editor can be retargeted to arbitrary root files

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The manual editor intends to confine a restricted Kea user to configured Kea files, but its allow-list also includes absolute Control Agent password-file values read from the Control Agent JSON. The same manual capability permits editing that JSON. A user can therefore set a password-file to an arbitrary absolute path, reload the editor so the path becomes allow-listed, read it through edit_text.cgi, and overwrite it through save_text.cgi as the Webmin process. Password-type entries are not subject to the JSON root check. This crosses the delegated module ACL boundary and provides root-level arbitrary file read/write.

Profile-based footprint: 0.00% observed (0/6,756 usable profiles; gate: kea-dhcp). No numeric deployment bound follows from a negative asset probe.

Recorded conditions: The editable Kea Control Agent JSON contains an absolute password-file entry that the manual editor can add to its allowlist. [nondefault]

Audited source: `kea-dhcp/kea-dhcp-lib.pl`, `kea-dhcp/edit_text.cgi`.

Module · LDAP Client (1)

WEBMIN-0549 — LDAP root password can be written to a world-readable client configuration

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When Webmin saves a root LDAP password, `if (&find_svalue("rootpwmoddn", $conf), 2)` evaluates to 2 even if the new-format directive is absent. Old-format deployments therefore use `save_directive` to place the password in `rootpwmodpw` in the main client configuration instead of `save_rootbinddn_secret`, which deliberately creates a root-owned mode-0600 file. If local users can read the main configuration, they can obtain a directory-root credential; direct host code execution was not demonstrated.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: ldap-client).

Recorded conditions: LDAP Client uses the legacy configuration format without the new-format root password directive. [nondefault]; The main LDAP client configuration is readable by a local user who can inspect the stored root password. [environmental]

Audited source: `ldap-client/save_server.cgi`, `ldap-client/ldap-client-lib.pl`.

Module · LDAP Server (1)

WEBMIN-0637 — Delegated LDAP certificate generation overwrites and chowns arbitrary existing PEM targets

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The classic LDAP certificate generator accepts any configuration-derived key and certificate path whose first byte is '/'. A user delegated the ldap-server module's slapd permission can set the TLS fields to an existing valid PEM anywhere readable by root. The generator then opens those paths for replacement as root and calls set_ownership_permissions with the configured LDAP account. Stable reproduction against 2.670 replaced a harmless root-owned mode-0600 PEM outside the LDAP configuration area and changed its ownership to UID/GID 65534. The generated content is constrained to a private key and certificate, so this audit does not claim direct RCE; the impact is cross-service TLS material replacement, denial of service, and expansion of write control to the LDAP account.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: The classic LDAP certificate generator is used. [environmental]; The selected key and certificate path names an existing valid PEM target readable by root. [environmental]

Audited source: `ldap-server/gencert.cgi`, `ldap-server/save_slapd.cgi`.

Module · LDAP Users and Groups (1)

WEBMIN-0229 — Automatic LDAP home paths accept traversal components

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The LDAP user administration automatic-home flow accepts username and group-derived values that can contain path separators, concatenates them beneath a configured base, and does not canonicalize the result before root-privileged mkdir, chmod, chown, and optional skeleton-copy operations. A delegated account operator can use dot-dot components to place a new home directory outside the intended base when the outside parent exists. This conclusion is based on the reviewed path construction and sinks plus a frozen-stable container reproduction.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: Delegated LDAP account creation uses automatic home paths and an existing out-of-base parent. [environmental]

Module · Logrotate (2)

WEBMIN-0910 — Latest stable retains a local root-execution race in immediate log rotation

CVSS 4.0 Base (CVSS-B): 7.3 (High) · `CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

rotate_log_now creates a temporary configuration with an ordinary umask-dependent write open, closes it, changes its mode, and only then invokes logrotate as the privileged Webmin process. If the initial mode grants a local principal write access, that principal can open the file before the mode change; chmod does not revoke the writable descriptor. The principal can therefore overwrite the root-owned, apparently non-writable configuration before logrotate reads it. In the constrained latest-stable lab, UID 65534 retained the descriptor across chmod 0644, injected a firstaction directive, and logrotate executed it as UID 0. The deterministic test hook widens the scheduling point but does not alter the underlying descriptor-permission semantics.

Profile-based footprint: 5.34% observed module signal (361/6,756 usable profiles; observed gate: logrotate).

Recorded conditions: The initial temporary configuration mode permits another local principal to open it for writing. [environmental]; The local principal wins the open-to-chmod race and retains the writable descriptor. [environmental]

Audited source: `logrotate/logrotate-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0963 — Local users can read force-rotation temporary configuration

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

When Webmin force-rotates one log, rotate_log_now serializes global logrotate directives and the selected log stanza into a transname temporary file, chmods that file to 0644, and supplies it to logrotate with -f. Webmin enforces mode 0755 on the containing temporary directory, so a local unprivileged account can traverse/list the directory and read the file while the command runs. This can disclose privileged filesystem paths and the text of administrator-configured pre/post-rotation commands. The exposure is a local, timing-dependent window and the registered path is removed during Webmin cleanup.

Profile-based footprint: 5.34% observed module signal (361/6,756 usable profiles; observed gate: logrotate).

Recorded conditions: A force-rotation runs while a local user can read the temporary directory and logrotate configuration. [environmental]

Audited source: `logrotate/logrotate-lib.pl`, `web-lib-funcs.pl`.

Module · Mailboxes (2)

WEBMIN-0401 — Root mbox rewrite follows a pre-created predictable symlink

CVSS 4.0 Base (CVSS-B): 5.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The root-side mbox rewrite helper uses a predictable .del sibling and opens it with O_CREAT|O_WRONLY but without exclusive creation or symlink protection. A mailbox owner who controls that directory can pre-place a symlink; the privileged operation follows it, writes the target, and assigns it to the mailbox user.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `mailboxes/boxes-lib.pl`.

WEBMIN-0826 — Palette header permits constrained local stylesheet path traversal

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

Both displayed-message and quote iframe helpers interpolate HTTP_X_COLOR_PALETTE_FILE directly into a filesystem path before appending .min.css. Dot components and path separators are not rejected. An authenticated requester can traverse outside the intended theme iframe directory and cause a known readable .min.css file to be embedded in the response, creating constrained local-file disclosure and style-content injection.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: A requester supplies a traversing palette header that resolves to a readable .min.css file. [environmental]

Audited source: `mailboxes/folders-lib.pl`.

Module · Minecraft (3)

WEBMIN-0368 — World path traversal permits out-of-tree deletion and disclosure

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

World deletion, download, and clone paths do not consistently apply the creation branch's name allowlist or require the requested name to match an enumerated world. The confirmed-delete route joins any non-empty name to minecraft_dir and calls Webmin's recursive deletion helper. A traversal name can therefore target a sibling or ancestor path; related routes can copy or archive out-of-tree data. In the frozen stable lab, an authenticated delete request with ../victim removed a populated sibling directory.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: Minecraft module user can delete, clone, or download worlds and supplies an out-of-tree traversal name. [environmental]

Audited source: minecraft/save_world.cgi, web-lib-funcs.pl.

WEBMIN-0520 — Minecraft world archive round trip discloses root-readable files through symlinks

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

Webmin 2.670 handles Minecraft world ZIP imports in the root Webmin process. It extracts an uploaded or server-side ZIP, accepts a readable level.dat, and recursively copies the resulting tree without rejecting symbolic links. Its authenticated world-download branch then runs zip -r as the same privileged process, which follows a stored symlink and archives the target contents. A Minecraft-only Webmin user can use this chain to retrieve an arbitrary root-readable file. The network-isolated stable lab reproduced disclosure of a mode-0600 root-owned marker. The demonstrated effect is disclosure of root-readable files; command execution was not demonstrated.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: minecraft/save_world.cgi.

WEBMIN-0790 — Remote Minecraft chat can drive root arbitrary file overwrite through playtime log parsing

CVSS 4.0 Base (CVSS-B): 8.8 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

The playtime parser searches each time-only server-log line with unanchored login and logout regexes. A remote player can place matching substrings in ordinary chat text and make the captured token, including ../ components, become a usage-map key. check_playtime_limits joins that key directly to the module playtime directory and passes the resulting path to write_file under the scheduled job's root privilege. The attacker controls the target path but not arbitrary file contents: output is constrained to the module's playtime key/value records. This supports high host-integrity and availability impact through creation or replacement of root-writable files, but the reviewed and reproduced path does not establish remote code execution.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: Minecraft playtime-limit tracking and its scheduled parser must be configured. [non-default]

Audited source: minecraft/minecraft-lib.pl.

Module · Net (1)

WEBMIN-0469 — Static-route device traversal permits root-owned writes outside the route directory

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

Red Hat static-route parsing accepts the submitted device value with only a non-whitespace check and uses it to form a per-device route filename. On branches using the $dev.route form, parent-directory components escape the intended route directory before the root-privileged write helper is called. A routes-level delegated administrator can therefore create or overwrite a suffix-form path outside the authorized network-route location with constrained route configuration content.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: The host uses the Red Hat static-route file backend. [environmental]

Audited source: net/redhat-linux-lib.pl.

Module · Pam (1)

WEBMIN-0387 — Relative PAM helper path can execute a planted Perl file

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The PAM edit and save endpoints form a Perl helper path by concatenating a request-controlled module name with ./ and .pl, then execute a readable result with do. Parent-directory components are not rejected. If a readable attacker-influenced Perl file is present at a reachable relative path, a PAM-authorized actor can cause it to execute with the Webmin CGI process privileges. The latest-stable test used a harmless planted file, observed its response marker and file side effect, and removed the entire lab afterward.

Profile-based footprint: 86.04% observed module signal (5,813/6,756 usable profiles; observed gate: pam).

Recorded conditions: A readable attacker-influenced Perl file exists at a path reachable through PAM's relative helper lookup. [environmental]

Audited source: pam/edit_mod.cgi, pam/save_mod.cgi.

Module · PHP Configuration (2)

WEBMIN-0324 — Restricted PHP Configuration user can overwrite root-owned files through an allowlisted php.ini symlink

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

The manual PHP configuration workflow authorizes the submitted path string through can_php_config, then performs a whole-file replacement that follows symlinks. For a restricted domain owner whose only allowed php.ini resides in a user-controlled directory, replacing that path with a symlink redirects the save into a root-owned destination. In Webmin 2.670, save_manual.cgi calls write_file_contents_as_user, but should_switch_user resolves the link and intentionally does not switch to the configured non-root ACL user when the target is under /etc, leaving the write in root context. The latest-stable lab authenticated a phpini-only Webmin delegate with anyfile=0, global=0, manual=1, user=tenant, and one allowlisted path; a manual save through that symlink replaced a root-owned mode-0600 /etc file while preserving root ownership. This is an arbitrary root-file overwrite primitive and can lead to root command execution when a suitable root-consumed file is targeted, but direct command execution was not exercised in the proof.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: The path requires a configured PHP file allowlist and a staged symlink. [nondefault]

Audited source: phpini/list_ini.cgi, phpini/save_manual.cgi.

WEBMIN-0456 — Module-only PHP Configuration access permits arbitrary root file overwrite and root command execution

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

An authenticated account granted the PHP Configuration module inherits a default ACL that enables manual editing of any pathname as root. save_manual.cgi accepts the multipart file parameter, can_php_config treats every path as allowed when anyfile is enabled, and write_file_contents_as_user writes in the root Webmin process for the default user=root setting. An attacker can therefore replace or create an arbitrary root-owned file with chosen content; selecting a root-consumed target yields root command execution. The network-disabled Webmin 2.670 reproduction wrote an attacker-selected /etc/profile.d script as UID 0 and a subsequent root login shell executed it as uid=0.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: phpini/save_manual.cgi, phpini/phpini-lib.pl.

Module · Postfix (2)

WEBMIN-0206 — Postfix log action path traversal can execute hooks outside action_script_dir

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

save_opts.cgi accepts _log_form from the request and passes it as the webmin_log action code after the existing Postfix general-settings permission check. When action scripts are configured, webmin_log constructs a directory beneath action_script_dir using this action and executes every executable file found there. Because path components are not validated, an authenticated user can submit ../ components and select a directory outside the configured hook tree. A suitable executable in that directory is then invoked by the Webmin process, normally as root.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: Webmin action scripts and a suitable external hook must be configured. [nondefault]

Audited source: postfix/save_opts.cgi, web-lib-funcs.pl.

WEBMIN-0801 — Default Postfix map ACL permits arbitrary root-owned file creation and overwrite

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

An authenticated identity with Postfix module access, header=1, manual=1, and an unrestricted directory ACL can submit an absolute regexp: or pcre: path through save_opts_header.cgi. The generic save path stores the parameter and ensure_map creates a missing target as the root-running Webmin process with mode 0755. save_manual.cgi then regards that path as a member of the chosen map and replaces it with caller-controlled bytes, without imposing a narrower path check. The shipped Postfix default ACL grants both capabilities and dir=/. This yields arbitrary root-owned file creation or overwrite outside the Postfix configuration area and can enable root compromise if the selected file is interpreted by a privileged component. The latest-stable reproduction used a benign disposable /tmp target and did not execute payload code.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: postfix).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: postfix/save_opts_header.cgi, postfix/postfix-lib.pl.

Module · PostgreSQL (1)

WEBMIN-0377 — World-writable PostgreSQL HBA directory permits local configuration replacement

CVSS 4.0 Base (CVSS-B): 6.9 (Medium) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

When the alternate PostgreSQL HBA sample is readable but the configured HBA parent does not exist, Webmin creates the parent as 0777 and copies the sample while operating with root filesystem authority. The world-writable, non-sticky directory lets any local UID unlink and replace pg_hba.conf. Frozen stable 2.670 retains the behavior and the container attempt reproduced replacement by UID 65534. This can corrupt bootstrap or weaken database authentication if the hostile file is later used, but it is not a demonstrated Webmin RCE or root privilege escalation and is constrained by the uncommon fresh/missing-directory state plus PostgreSQL's own data-directory checks.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: The configured PostgreSQL HBA parent directory is missing while the alternate sample is readable. [nondefault]

Audited source: `postgresql/index.cgi`, `postgresql/config-freebsd`.

Module · Proc (1)

WEBMIN-0863 — Orphaned trace terminator trusts a caller-selected PID file

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The trace terminator derives a .tail path from caller-controlled id, reads the first line as a PID, sends HUP and KILL to that PID, and unlinks the chosen path without checking whether the Webmin user may manage the process. A traversal id can select an attacker-prepared PID file outside the module configuration directory. The latest-stable lab used only a disposable container child and confirmed signaling and unlinking.

Profile-based footprint: 20.43% observed module signal (1,380/6,756 usable profiles; observed gate: proc).

Recorded conditions: A caller can place or select a PID file at a traversed trace ID path and target a live process. [environmental]

Module · Proftpd (1)

WEBMIN-0027 — Delegated ProFTPD configuration enables arbitrary root-owned file append

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

Webmin exposes add_file as a free-text ProFTPD module setting to authenticated module users whose ACL permits configuration editing. create_virt.cgi uses that path in privileged read/flush file operations, while ServerName validation only requires non-whitespace and permits embedded newlines. A ProFTPD-only delegated user can therefore select a root-writable target and append attacker-controlled complete lines between the fixed virtual-host wrapper lines. Frozen Webmin 2.670 reproduced the primitive against a root-owned mode-0600 file while denying the account access to Webmin Users. Privileged files can be corrupted or poisoned. The path requires valid Webmin credentials, explicit module delegation, noconfig=0, and a suitable target; direct command execution was not demonstrated.

Profile-based footprint: 5.28% observed module signal (357/6,756 usable profiles; observed gate: proftpd).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `proftpd/config.info`, `config_save.cgi`.

Module · Samba (3)

WEBMIN-0384 — Samba roaming-profile cleanup permits authenticated path traversal to root directory deletion

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

When Samba roaming-profile cleanup is enabled, useradmin_delete_user concatenates the local account name beneath the Profiles share path and executes rm -rf for directories, including the .v2 profile added by the reviewed change. Useradmin's default account-name checks do not reject slash or dot-dot components. A Webmin principal allowed to create and delete affected Unix users with cross-module deletion enabled can therefore make the root Webmin process recursively delete a chosen directory outside the configured profile root. Shell quotemeta prevents command injection but does not constrain filesystem traversal. The path requires the optional synchronization setting and sufficient delegated Useradmin rights.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: Samba roaming-profile cleanup and cross-module Unix-user deletion are enabled. [nondefault]

Audited source: samba/useradmin_update.pl, useradmin/save_user.cgi.

WEBMIN-0502 — Samba share creation can chown and chmod an arbitrary root-owned file through a symlink

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

A Webmin user granted only Samba file-share creation can request automatic directory creation at an attacker-controlled pathname. If a local attacker pre-positions that pathname as a symlink to a regular root-owned file, the !-d condition is true, make_dir's mkdir fails with EEXIST but the helper still chmods the pathname and returns success, and set_ownership_permissions then chowns and chmods it again as root. The attacker can choose an existing Unix owner/group and a permissive mode; assigning the target to their local account makes it writable. A root-consumed file such as an authentication or service configuration file turns this primitive into root code execution. The path requires both authenticated Samba c_fs delegation and local symlink placement.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: A local attacker can place a symlink at the selected automatic-share directory path before Webmin creates it. [environmental]

Audited source: samba/save_fshare.cgi, web-lib-funcs.pl.

WEBMIN-0676 — Samba account passwords can be exposed through a world-readable temporary file

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

When the Samba module uses pdbedit to create an account, create_user writes the supplied plaintext password twice to a transname file and opens it through open_tempfile's direct-write branch. Webmin's temp directory is mode 0755, and the direct branch does not set a restrictive file mode, so a default 0022 umask creates the password file as 0644 until the command finishes and process cleanup removes it. An unprivileged local user who monitors the searchable directory can read the credential during that window. A restrictive Webmin umask mitigates the path; exploitation also requires local access and favorable timing.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: Samba uses pdbedit account creation and a local user can read its 0755 temporary directory during a 0022-umask write. [environmental]

Audited source: samba/samba-lib.pl, web-lib-funcs.pl.

Module · Sendmail (2)

WEBMIN-0197 — Alias-only Webmin delegate can write root-owned files and escalate to root

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

A user permitted to create Sendmail aliases can submit an arbitrary afile parameter. save_alias.cgi copies it into the new alias object without checking that it equals one of the server-derived AliasFile paths; create_alias then reads, appends, and flushes that path with Webmin's root privileges. The cmt field accepts an embedded newline and make_table_comment preserves it, so the attacker controls an entire additional line before the generated alias line. This is not useful escalation for a master Webmin administrator who is already root-equivalent, but it is a real boundary break for delegated Sendmail/alias accounts. A standard one-file setup remains reachable by a forged POST even though the selector is shown only when multiple files are configured. Root execution requires a suitable root-consumed file, such as authorized_keys or a scheduler configuration, that accepts the injected line.

Profile-based footprint: 85.49% observed module signal (5,776/6,756 usable profiles; observed gate: sendmail).

Recorded conditions: The selected root-owned target is consumed by a privileged component that accepts the injected line. [environmental]

WEBMIN-0924 — Inbound autoreply macros can override the reply-tracking DBM path

CVSS 4.0 Base (CVSS-B): 6.9 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:L/VA:L/SC:N/SI:N/SA:N`.

autoreply.pl substitutes $SUBJECT, $FROM, $TO, $DATE, and $BODY into each template line before determining whether that line is an internal directive. If sender-controlled expansion occurs while template header parsing remains active, the expansion can supply a new Reply-Tracking directive and replace the operator's configured path. The script then resolves that sender-selected value, opens it as a writable DBM, writes tracking entries, and may remove DBM sidecar suffixes on an error. Exploitation requires an unsafe but supported template layout and is limited to locations writable by the mail-delivery identity; it nevertheless gives an unauthenticated message sender a reproducible file-operation primitive and storage-exhaustion avenue.

Profile-based footprint: 85.49% observed module signal (5,776/6,756 usable profiles; observed gate: sendmail).

Recorded conditions: Sendmail autoreply expands sender-controlled macros while template header parsing remains active. [environmental]

Audited source: `sendmail/autoreply.pl`.

Module · Shell (1)

WEBMIN-2433 — Root host-side chdir leaks directory existence outside the configured Shell chroot

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

When a chroot ACL is configured, the Shell CGI accepts request-carried working-directory state and parsed cd paths, concatenates them with the jail root, and performs chdir while Webmin still has root's host filesystem view. Parent-directory components can resolve to host directories outside the jail. Successful traversal is reflected in the returned pwd and failure emits an error, allowing a delegated Shell module user to test host-directory existence and root traversability. Command execution occurs only afterward under the selected user and chroot or jail shell, so the demonstrated impact is limited to a directory-existence oracle.

Profile-based footprint: 4.56% observed module signal (308/6,756 usable profiles; observed gate: shell).

Recorded conditions: Shell delegate is confined by a configured chroot ACL. [nondefault]; A traversing working-directory path resolves to an existing directory outside the jail. [environmental]

Audited source: `shell/index.cgi`, `shell/shell-lib.pl`.

Module · Shorewall (1)

WEBMIN-1029 — Directory traversal in Shorewall table helpers escapes config_dir

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The Shorewall module authorizes table names with can_access. When the files ACL is '*', can_access returns success for every string; defaultacl selects that wildcard. Multiple row helpers then concatenate the untrusted table string after config_dir and open, cache, flush, or reorder the resulting path without rejecting parent-directory components or checking canonical containment. Authenticated module requests can therefore direct privileged table operations to files outside the Shorewall configuration directory.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: shorewall).

Recorded conditions: Shorewall files ACL is wildcard, as in defaultacl. [nondefault]; A module user supplies a traversing table name to a file operation. [environmental]

Audited source: `shorewall/defaultacl`, `shorewall/shorewall-lib.pl`.

Module · Software (1)

WEBMIN-1020 — Package-listed symlink targets are disclosed by the file viewer

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

Exact package-list membership is treated as sufficient authorization even when the listed path is a symlink. The viewer then uses an ordinary open that follows the link, so the privileged process returns target content outside the path/type boundary the package listing presents. The gap between membership enumeration and open also lacks descriptor-identity enforcement.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: A package file list contains a symlink to a target outside the listed path boundary. [environmental]

Audited source: `software/software-lib.pl`, `software/debian-lib.pl`.

Module · SSH Server (2)

WEBMIN-0462 — SSH user-creation hook permits root arbitrary file overwrite through authorized_keys symlink

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The hook checks only that .ssh is absent before running key generation as the new account. While key generation is in progress, that account can place authorized_keys as a symlink. Webmin then copies a generated public key and changes ownership as root without rejecting the symlink, so both operations affect the target file. The reproduced primitive replaces an existing root-owned file and hands it to the lower-privilege UID; redirecting it to a root-consumed authentication or scheduled-command configuration can produce root execution.

Profile-based footprint: 95.20% observed module signal (6,432/6,756 usable profiles; observed gate: sshd).

Recorded conditions: Automatic SSH key generation for new users must be enabled. [nondefault]

Audited source: `sshd/useradmin_update.pl`, `web-lib-funcs.pl`.

WEBMIN-0961 — New-user Ed25519 authorization race can chown an attacker-selected file

CVSS 4.0 Base (CVSS-B): 5.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

When automatic SSH key creation and self-authorization are enabled, Webmin generates a key as the newly created user, then resumes as root and copies a readable public-key pathname from that user's .ssh directory. For id_ed25519.pub, a concurrent account can replace the generated public key with a symlink. copy_source_dest preserves that link at authorized_keys, and set_ownership_permissions then follows authorized_keys while assigning the new user's UID and GID. The result is a race-dependent ownership change of an attacker-selected root-owned target. The new account is already saved before the module callback runs, so it can act concurrently. A sensitive target may turn the ownership change into local root privilege escalation. The feature is opt-in and exploitation requires winning the provisioning-time race.

Profile-based footprint: 95.20% observed module signal (6,432/6,756 usable profiles; observed gate: sshd).

Recorded conditions: New-user Ed25519 key generation/authorization must be enabled. [nondefault]

Audited source: `sshd/useradmin_update.pl`, `web-lib-funcs.pl`.

Module · Status (1)

WEBMIN-0576 — Status monitor clone path traversal loads planted Perl as root

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Status monitor save path accepts clone from a user with monitor-edit authority and stores it without checking that it names an installed clone or remains beneath Webmin's module roots. service_status later passes that saved value to the Fail2ban monitor handler added by the reviewed commit. That handler calls foreign_require, whose module path is built by concatenation and whose requested library is executed with Perl do. A Status user who can plant a module.info and fail2ban-lib.pl in a writable local directory can use parent-directory components in clone to load that Perl in Webmin's root process. Frozen Webmin 2.670 retains the complete chain; the locked network-disabled reproduction created the planted directory as UID 65534, saved the crafted monitor through authenticated Webmin, and recorded the planted code executing as uid=0/euid=0.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: The attacker can plant module.info and Perl code in a local directory read by the clone loader. [environmental]; A Status monitor later handles the stored clone value. [environmental]

Audited source: `status/fail2ban-monitor.pl`, `status/save_mon.cgi`.

Module · Updown (1)

WEBMIN-0120 — Directory ZIP symlink traversal discloses files outside the ACL

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

fetch.cgi authorizes the selected directory but does not apply the directory ACL to recursively encountered entries. It then executes zip -r, which follows a nested symlink in the tested stable environment. Under mode 0 the PATH_INFO fetch branch does not switch away from the Webmin root identity, so a delegate can retrieve root-readable files outside the allowed tree through a symlink inside it.

Profile-based footprint: 5.60% observed module signal (378/6,756 usable profiles; observed gate: updown).

Recorded conditions: Updown delegate is confined to an allowed upload tree in mode 0. [nondefault]; The downloadable tree contains a symlink to an out-of-scope file. [environmental]

Audited source: `updown/fetch.cgi`, `updown/updown-lib.pl`.

Module · Useradmin (1)

WEBMIN-0054 — Batch home-directory confinement accepts sibling-prefix paths

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N`.

The batch validator treats the allowed home root as a raw string prefix rather than a directory boundary. If delegated scope is /srv/allowed, a path such as /srv/allowed-escape/member passes and can become the target of privileged home creation, ownership, or skeleton-file operations.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: Delegated batch user administration is confined to an allowed home directory. [nondefault]

Audited source: `useradmin/batch_exec.cgi`.

Module · Vendor Perl (1)

WEBMIN-0996 — TNEF privilege separation permits root to consume mailbox-user-controlled paths

CVSS 4.0 Base (CVSS-B): 4.3 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:P/VC:H/VI:N/VA:L/SC:N/SI:N/SA:N`.

The TNEF path changes ownership of a private parser directory to the mailbox user and runs Convert::TNEF under that effective identity, but restores effective root before reading decoded attributes. Convert::TNEF stores every attribute larger than 4096 bytes in a predictable file inside that user-owned directory; MIME::Body later reopens the stored pathname without link protection. The mailbox owner can replace a completed spill path

while the parser processes later input, causing the root-context data() call to follow a symbolic link. In the frozen latest-stable container, a nobody-owned parser directory and crafted TNEF stream allowed nobody to replace the AttachData path with a link to a synthetic root-owned mode-0600 marker: nobody's direct read was denied, yet decode_tnef_attachment returned the marker after root restoration. The same boundary is also skipped for nested message objects because extract_mail does not preserve open_user or a parent link. Exploitation requires coordination with a privileged Webmin mailbox parse and access to the associated local account's parser directory; the demonstrated impact is privileged file disclosure, with denial of service also possible by redirecting the unbounded read to a special file.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: The mailbox owner can access the associated parser directory and replace a completed spill path. [environmental]; A privileged Webmin mailbox parse continues after the path replacement. [environmental]

Audited source: `mailboxes/boxes-lib.pl`, `vendor_perl/Convert/TNEF.pm`.

Information exposure and secrets (41)

Core · ACL (2)

WEBMIN-0226 — Authentication debug logging records credential material

CVSS 4.0 Base (CVSS-B): 5.7 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When miniserv debugging is enabled, the current server writes full POST data, decoded Basic-auth credentials, validate_user password arguments, failed submitted passwords alongside stored password fields, and external-database password fields. The debug log is opt-in and locally restricted, but it becomes a secondary store for plaintext credentials and password verifier material that can be exposed through local access, backups, or log collection. The latest-stable container observed a controlled credential marker in that log; only the boolean observation was retained outside the disposable container.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: MiniServ authentication debug logging is enabled. [nondefault]; A local or backup/log-collection observer can read the debug output. [environmental]

Audited source: `miniserv-lib.pl`.

WEBMIN-0394 — Database configuration endpoints enforce the wrong ACL privilege

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The ACL module exposes a dedicated sql privilege on the display endpoint, but save_sql.cgi and the database setup helpers authorize with the unrelated pass privilege. Those endpoints accept backend hosts, credentials, and database/LDAP identifiers; they validate or create remote structures and persist the selected user database in miniserv configuration. An authenticated delegated administrator who can manage password policy but is denied database configuration can therefore cross that intended privilege boundary.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: ACL delegate can manage password policy but is denied external database configuration. [nondefault]

Audited source: `acl/edit_sql.cgi`, `acl/save_sql.cgi`.

Core · Command-line utilities (1)

WEBMIN-2441 — Webmin password supplied on the command line is exposed in process arguments

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The supported --password option carries the new Webmin-only password in the process argument vector. On systems where another local account can inspect that process, or where administrative commands are recorded with arguments, the cleartext Webmin credential can be disclosed. Hidden interactive entry avoids placing the secret in argv. This is a local operational credential exposure through process arguments or command recording.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: An administrator passes a Webmin password through the --password CLI option while a local observer can inspect argv. [environmental]

Audited source: `bin/passwd`.

Core · Runtime (2)

WEBMIN-0012 — Debug modes write plaintext credentials to log files

CVSS 4.0 Base (CVSS-B): 5.7 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When miniserv or network debugging is enabled, diagnostic messages interpolate supplied login, sudo, HTTP, FTP, or user-database password values. The plaintext credentials persist in debug files and may be disclosed through log access, backups, or support collection. Frozen 2.670 retains the network-helper behavior, and the controlled latest-stable Docker probe observed a dummy credential in webmin.debug.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: MiniServ request or authentication debug logging is enabled. [nondefault]; An observer can read the debug log or a copy of it. [environmental]

Audited source: `web-lib-funcs.pl`, `miniserv-lib.pl`.

WEBMIN-0745 — Tokenless FastRPC transfer sockets permit file theft and content injection

CVSS 4.0 Base (CVSS-B): 7.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`.

After an authenticated tcpread or tcpwrite request, the auxiliary listener accepts the first network peer without a session identifier, transfer nonce, credential, or expected-peer check. The first peer can steal the selected file or replace bytes written to the authenticated client's chosen destination; an empty acknowledgment is incorrectly treated as success.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: An authenticated transfer creates a FastRPC auxiliary listener reachable by a competing network peer. [environmental]

Audited source: `fastrpc.cgi`.

Core · Cross-cutting (1)

WEBMIN-0672 — Configured SMTP STARTTLS silently downgrades to plaintext

CVSS 4.0 Base (CVSS-B) · path `publication:module:mailboxes:primary`: 6.0 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

CVSS 4.0 Base (CVSS-B) · path `publication:module:status:primary`: 6.0 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When STARTTLS mode is selected, the mail sender treats a non-success upgrade response as permission to continue without TLS, then proceeds to optional authentication and message delivery. The SMTP monitor also tests an unrelated variable before its STARTTLS branch, allowing the configured check to remain plaintext. A rejecting or actively manipulated peer can therefore expose mail contents and authentication exchanges.

Profile-based footprint: 96.18% observed module signal (6,498/6,756 usable profiles; observed gate: mailboxes, status).

Recorded conditions: Mail sender or SMTP monitor is configured for STARTTLS. [nondefault]; The peer rejects or an attacker suppresses the upgrade response. [environmental]

Audited source: mailboxes/boxes-lib.pl, status/smtp-monitor.pl.

Core · Integration (1)

WEBMIN-0743 — Servers JSON endpoint discloses remote credentials when pass=0

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

The Servers module JSON endpoint returns complete stored server records after checking only whether the account may use each server. Because it does not honor the separate pass ACL or redact the pass field, a delegated account denied password visibility can retrieve remote management credentials. The behavior was reproduced against the frozen 2.670 source in an isolated latest-stable container.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: Servers delegate can use a stored server but is denied password visibility. [nondefault]; The selected linked-server record contains a credential. [environmental]

Audited source: servers/index-json.cgi.

Core · Packaging (3)

WEBMIN-0679 — Unofficial Windows helper provisions a known Webmin administrator credential without TLS

CVSS 4.0 Base (CVSS-B): 9.3 (Critical) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

setup.bat supplies login=admin, password=admin, and ssl=0 to setup.pl. It also supplies nostart=nostart, which delays rather than removes the exposure: if the operator later starts the staged service or enables it at boot without first changing the generated account and transport settings, a network peer can use the shipped credential for full Webmin administration and an on-path observer can recover it from plaintext HTTP.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: An operator uses the unofficial Windows setup.bat helper and later starts Webmin before changing admin/admin or enabling TLS. [environmental]

Audited source: setup.bat, setup.pl.

WEBMIN-0768 — setup.sh leaks the initial Webmin administrator password through process arguments

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

The shell installer supplies the plaintext initial administrator password as ARGV[1] to a short-lived Perl hashing process. On hosts where procfs command lines are readable across users, a local observer can poll the process list and recover the credential during installation. The frozen 2.670 source retains this construction, and a network-isolated Docker reproduction showed UID 65534 reading a synthetic password marker from the hashing process command line.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: An operator runs setup.sh with a plaintext initial password while a local user can inspect process arguments. [environmental]

Audited source: setup.sh.

WEBMIN-0901 — Authenticated RPM repository file exposes credentials to local users

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N.

For authenticated RPM repositories, the setup script embeds the supplied username and password in the repository base URL and writes the repository file without a restrictive mode. Under a standard root umask, local users can read and reuse that external repository credential.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: An authenticated RPM repository is installed with a credential and local users can read its mode-permissive file. [environmental]

Audited source: `webmin-setup-repo.sh`.

Module · Cluster Webmin (1)

WEBMIN-0108 — Optional verification permits cleartext cluster-upgrade package substitution

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

At the reviewed snapshot, the Cluster Webmin latest-release flow obtained release information and the selected Webmin package over cleartext HTTP. Signature verification was exposed as an unchecked optional control, and the no-check branch continued to the package installer. Because Webmin performs the upgrade with administrative privileges and can distribute it to multiple managed hosts, a network-position attacker able to substitute the download could gain code execution on every selected target.

Profile-based footprint: 4.19% observed module signal (283/6,756 usable profiles; observed gate: cluster-webmin).

Recorded conditions: A cluster upgrade downloads over HTTP while an on-path attacker substitutes the package. [environmental]; Package signature verification is left unselected for the cluster upgrade. [nondefault]

Audited source: `cluster-webmin/upgrade.cgi`, `cluster-webmin/index.cgi`.

Module · Fsdump (2)

WEBMIN-0124 — Background filesystem backup writes remote password to action log

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

Filesystem-backup actions pass a dump object containing the remote password to Webmin's persistent action logger unless the pass member is explicitly removed. The frozen 2.670 background-backup branch calls webmin_log with the unredacted object. The latest-stable Docker test created a password-bearing backup definition, invoked the authenticated background CGI, and observed a pass field in webmin.log; retained evidence redacts the random sentinel and session value.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: fsdump).

Recorded conditions: A background filesystem backup uses a remote password and an observer can read the action log. [environmental]

Audited source: `fsdump/backup.cgi`, `web-lib-funcs.pl`.

WEBMIN-0972 — FSdump FTP transport lacks peer authentication and fail-closed TLS

CVSS 4.0 Base (CVSS-B): 6.0 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The frozen stable FTP backup client upgrades sockets with SSL_verify_mode set to 0, so TLS provides encryption without authenticating the configured server. AUTH TLS is also optional: if the server rejects it, the client proceeds to USER/PASS and data transfer in plaintext. An on-path adversary can therefore present an untrusted TLS endpoint or suppress the upgrade and observe or alter FTP credentials and backup data. The latest-stable container test executed the unmodified ftp.pl with controlled socket and server responses; accepted TLS recorded verification mode 0, while rejected TLS recorded both PASS and STOR on plaintext transport.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: fsdump).

Recorded conditions: FSdump backup uses FTP with optional AUTH TLS and stored credentials. [nondefault]; An on-path attacker impersonates the FTPS peer or rejects the upgrade. [environmental]

Audited source: `fsdump/ftp.pl`.

Module · Init (1)

WEBMIN-0117 — Arbitrary root file disclosure in Webmin 2.670 init module

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

Webmin 2.670 permits a bootup=2 start/stop-only init-module delegate to pass an absolute path as the type-0 action name. action_filename preserves the absolute path, edit_action.cgi reads it under the root Webmin process, and the supposedly read-only view returns the file contents. The isolated exact-stable lab returned a marker from /root with mode 0600 over authenticated HTTP.

Profile-based footprint: 5.45% observed module signal (368/6,756 usable profiles; observed gate: init).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `init/edit_action.cgi`, `init/init-lib.pl`.

Module · Iscsi Client (2)

WEBMIN-0354 — iSCSI CHAP credentials are rendered in cleartext Web UI controls and details

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The iSCSI Client authentication page reads stored node and discovery CHAP passwords and renders them with ui_textbox, whose default output is an ordinary type=text input containing the supplied value. The per-session connection form does the same for a newly entered secret, and connection details print password and password_in values as table content. Although access requires an authenticated Webmin principal with this module, these displays unnecessarily reveal reusable storage credentials in clear text and in the page DOM.

Profile-based footprint: 4.11% observed module signal (278/6,756 usable profiles; observed gate: iscsi-client).

Recorded conditions: iSCSI Client has saved CHAP credentials and an authenticated module user can open the affected views. [environmental]

Audited source: `iscsi-client/edit_auth.cgi`, `iscsi-client/add_form.cgi`.

WEBMIN-0355 — Per-session CHAP passwords enter Webmin execution logs and command arguments

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

Per-session connection setup concatenates the CHAP password into an iscsiadm --value argument and passes the full command to backquote_logged. That wrapper forwards the command string to additional_log as an exec record before executing it through a shell. Consequently the secret enters Webmin command-audit data and also exists in command arguments during execution. An observer with access to execution logs or suitable local process visibility can recover the reusable iSCSI credential.

Profile-based footprint: 4.11% observed module signal (278/6,756 usable profiles; observed gate: iscsi-client).

Recorded conditions: Per-session iSCSI CHAP is configured; a log reader or process observer can access the command. [environmental]

Audited source: `iscsi-client/iscsi-client-lib.pl`, `web-lib-funcs.pl`.

Module · Kea Dhcp (1)

WEBMIN-1010 — Kea structured editors persist submitted credentials in action logs

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

save_options.cgi accepts a replacement lease-database password and save_ddns.cgi accepts replacement TSIG secrets, but both handlers pass the complete request map to webmin_log after saving. Webmin serializes scalar

parameters into its persistent action log, and the Actions Log detail view renders stored parameters to identities authorized to view Kea actions. Newly submitted credentials therefore persist outside the Kea configuration and cross into a broader logging access and retention boundary.

Profile-based footprint: 0.00% observed (0/6,756 usable profiles; gate: kea-dhcp). No numeric deployment bound follows from a negative asset probe.

Recorded conditions: Kea structured editor is configured with lease-database or TSIG credentials. [nondefault]; An Actions Log reader can inspect the saved raw request parameters. [environmental]

Audited source: `kea-dhcp/save_options.cgi`, `kea-dhcp/save_ddns.cgi`.

Module · LDAP Server (4)

WEBMIN-0141 — Automatic LDAP encryption detection falls back to plaintext with bind credentials

CVSS 4.0 Base (CVSS-B): 7.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`.

When the LDAP Server module manages a remote server and encryption is left at Detect automatically, connect_ldap_db attempts LDAPS, then StartTLS, then plain LDAP. Each attempt receives the same configured bind DN and password, and generic_ldap_connect performs the bind after transport setup. If encrypted attempts are unavailable while plain LDAP is reachable, the administrative bind can therefore traverse the network without transport encryption, allowing a network-positioned adversary to capture or reuse the credential.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: A remote LDAP client uses Detect encryption; encrypted attempts fail while plaintext bind remains available. [environmental]

Audited source: `ldap-server/ldap-server-lib.pl`, `ldap-client/ldap-client-lib.pl`.

WEBMIN-0142 — LDIF configuration endpoints enforce the stale slapd ACL instead of ldif

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

On an LDIF-based OpenLDAP installation, Webmin exposes ldif as the configuration permission and uses it to decide whether the LDIF page appears in the module index. The LDIF view, save, and certificate endpoints instead check the separate slapd flag. A delegated user with ldif=0 but slapd=1 can directly reach configuration functionality that the active ACL model denies, including LDAP administrative identity/password, database limits, TLS file paths, protocol settings, and certificate generation.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: LDIF-based OpenLDAP configuration is used, with a delegate allowed slapd but denied ldif administration. [nondefault]

Audited source: `ldap-server/acl_security.pl`, `ldap-server/index.cgi`.

WEBMIN-0811 — LDAP administrator password exposed in process arguments and command-debug output

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When Webmin saves a new OpenLDAP administration password and slappasswd is available, hash_ldap_password builds a shell command using slappasswd -s followed by the plaintext submitted password. The command is executed through backquote_command, making the credential visible in process command lines for the duration of the call. If Webmin command debugging is enabled, backquote_command also records the translated command, causing the plaintext credential to persist in debug output. Shell escaping limits command interpretation but does not protect the credential from these observation channels. A local process observer or a party with later access to the relevant debug record could recover the LDAP administrator password.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: LDAP administrator password is changed with slappasswd present and a local observer can inspect argv. [environmental]; Command-debug logging is enabled for the secondary log exposure path. [nondefault]

Audited source: `ldap-server/ldap-server-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0882 — LDAP schema import exposes its administrative password in argv

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The LDIF schema-import endpoint reads the configured or discovered LDAP administrative password and builds a shell command containing `ldapadd -w <password>`. Webmin's execution helper starts that string as a child command, so the secret becomes a process argument; the helper can also copy the full command into command-debug output. In the frozen 2.670 lab, a schema-authorized import using a synthetic LDAP password produced a live argv that uid 65534 could read from `/proc`. The endpoint's local-server, schema-ACL, schema-directory, and executable checks are relevant controls over invocation and file choice, but they do not prevent credential disclosure after the command is spawned.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: LDAP schema import runs with a configured administrator password while a local observer can inspect argv. [environmental]; Command-debug logging is enabled for the secondary log exposure path. [nondefault]

Audited source: `ldap-server/import_sfile.cgi`, `web-lib-funcs.pl`.

Module · LDAP Users and Groups (3)

WEBMIN-0014 — Plaintext LDAP passwords exposed in slappasswd process arguments

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The password hashing helper uses backquote_command to invoke slappasswd with the plaintext password supplied after -s for SHA, SSHA, and LDAP-MD5 modes. Process arguments are observable independently of shell quoting. In the latest-stable container, an observer running as UID 65534 read the synthetic marker from /proc while the command executed. A local account can therefore capture a password at creation or change time on systems whose procfs exposes other users' command lines.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: LDAP password hashing uses slappasswd and a local observer can inspect process arguments. [environmental]

Audited source: `ldap-useradmin/ldap-useradmin-lib.pl`, `web-lib-funcs.pl`.

WEBMIN-0682 — LDAP SHA-512 crypt salts repeat for password operations in the same second

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

When LDAP password encryption selector value 6 is active, the module delegates to useradmin::encrypt_sha512 without supplying a salt. That helper constructs a missing salt from the last eight decimal digits of time(), so all calls during one second use the same SHA-512 crypt salt. Interactive creation can encounter the issue, and the batch create and update paths make clustered same-second operations practical. If an adversary obtains the stored hashes, reused salts expose equal-password reuse among affected accounts and allow work against the shared salt to be amortized. An attacker must first obtain stored password hashes. Salt reuse alone does not bypass authentication or reveal plaintext passwords.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: LDAP password encryption selector is SHA-512 crypt mode 6. [nondefault]; Multiple password operations occur within one second and an attacker obtains the hashes. [environmental]

Audited source: `ldap-useradmin/ldap-useradmin-lib.pl`, `useradmin/md5-lib.pl`.

WEBMIN-0711 — Delegated module configuration can exfiltrate privileged local files through LDAP bind

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When ldap_pass_file is configured, ldap_connect opens that path with the privileges of the Webmin process, reads the entire file without constraining its location, and supplies the bytes as the password to generic_ldap_connect. A delegated ldap-useradmin user who retains module-configuration permission can also choose the LDAP host and bind identity. By selecting a privileged local file and an observable LDAP endpoint, that user can cause otherwise unreadable local content to be transmitted as a bind password. The module default ACL sets noconfig=0, while administrators can prevent this path by explicitly denying module-configuration access.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: LDAP Users and Groups delegate can edit module configuration and set ldap_pass_file. [nondefault]; The configured LDAP host is attacker-observable and Webmin can read the selected local file. [environmental]

Audited source: `ldap-useradmin/ldap-useradmin-lib.pl`, `ldap-client/ldap-client-lib.pl`.

Module · Mailboxes (1)

WEBMIN-0869 — STARTTLS rejection falls through to plaintext SMTP authentication

CVSS 4.0 Base (CVSS-B): 6.0 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When STARTTLS mode is selected, send_mail issues STARTTLS and switches to TLS only for a 2xx response. Any other response merely clears the ssl flag; execution then loads configured SMTP credentials, creates the SASL client, and sends AUTH on the unchanged plaintext socket. The latest-stable loopback reproduction returned 454 to STARTTLS and observed AUTH PLAIN followed by a response containing the marker username and password. This is a fail-open transport downgrade; the default CRAM-MD5 choice is less revealing than PLAIN, but configured PLAIN and comparable mechanisms expose reusable credentials.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: Mailboxes SMTP delivery is configured for STARTTLS with authentication. [nondefault]; The SMTP peer rejects the STARTTLS command. [environmental]

Audited source: `mailboxes/boxes-lib.pl`.

Module · MySQL (3)

WEBMIN-0738 — MYSQL_PWD is inherited by the lower-privileged backup writer

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

make_authstr and set_authstr_env keep the MySQL password in MYSQL_PWD. In backup_database, Webmin wraps only the output writer with command_as_user, reasserts MYSQL_PWD, and then launches the complete mysqldump pipeline. command_as_user's env=0 form uses su without a login environment, allowing the lower-privileged writer to inherit the database password. In the isolated latest-stable harness, the real set_authstr_env and command_as_user functions launched UID 65534, and that child's environment contained the exact non-secret sentinel under MYSQL_PWD.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL backup uses a lower-privileged output writer. [nondefault]; A MySQL password is set in MYSQL_PWD and the writer inherits the environment. [environmental]

WEBMIN-2412 — MySQL action details can retain plaintext database credentials

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The MySQL logging wrapper expands SQL parameters and supplies the complete statement to Webmin's auxiliary action logging. Credential-management paths also send SQL containing password material through the same wrapper. When action detail logging is enabled, this creates a durable secondary copy of database credentials. Access requires local log access or appropriately delegated Webmin-log access, so this is reported as a security observation rather than an unauthenticated disclosure.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MySQL action detail logging is enabled while a credential-changing SQL statement is executed. [nondefault]; A local or delegated action-log reader can inspect the logged SQL. [environmental]

Audited source: `mysql/mysql-lib.pl`.

WEBMIN-2424 — MySQL password can appear in local process arguments

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

When Webmin cannot use MYSQL_PWD for a MySQL client invocation, make_authstr falls back to a --password= argument containing the configured database password. Under process-visibility configurations that allow another local principal to inspect command lines, the secret is exposed for the lifetime of the client process. The impact requires local process observation and the fallback condition; no remote command execution is claimed.

Profile-based footprint: 97.10% observed module signal (6,560/6,756 usable profiles; observed gate: mysql).

Recorded conditions: MYSQL_PWD cannot be used, so MySQL client authentication falls back to --password=; another local user can see argv. [environmental]

Audited source: `mysql/mysql-lib.pl`.

Module · PHP Configuration (1)

WEBMIN-0674 — Delegated PHP Configuration user can disclose root-only directive values

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The PHP Configuration ACL promises to read and write configuration files as a configured Unix user, but several authenticated structured edit handlers call get_config directly. Because Webmin normally runs as root and can_php_config also permits arbitrary paths when anyfile is enabled, a delegated user can have those handlers parse a file the ACL Unix identity cannot read and render recognized directive values. The exposed directives are limited to those surfaced by the affected forms.

Profile-based footprint: 95.40% observed module signal (6,445/6,756 usable profiles; observed gate: phpini).

Recorded conditions: PHP Configuration delegate is constrained to a Unix identity while anyfile allows an otherwise unreadable path. [nondefault]

Audited source: `phpini/phpini-lib.pl`, `phpini/edit_db.cgi`.

Module · PostgreSQL (1)

WEBMIN-0511 — PostgreSQL user edit and synchronization paths interpolate values into SQL

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N`.

PostgreSQL user-management code touched by this batch builds SQL by interpolating role-related values rather than binding or applying PostgreSQL-aware quoting. The edit_user.cgi request parameter is placed directly into a catalog lookup after the users ACL check, and the synchronization hook directly assembles role names and passwords into lookup and role-management commands. A principal already delegated Webmin user-management access can therefore alter SQL semantics under the module's configured database identity. In the frozen latest-stable lab, a control role lookup returned no rows while a crafted role value caused the real Webmin execute_sql_safe helper to return an injected marker row. The demonstrated effect is query manipulation and data exposure; broader impact depends on the configured PostgreSQL role and which eligible path the principal can reach.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: PostgreSQL role editing or Unix-user synchronization accepts attacker-controlled role or password text. [environmental]

Audited source: `postgresql/edit_user.cgi`, `postgresql/useradmin_update.pl`.

Module · Project Meta (1)

WEBMIN-0906 — Mutable external CI workflow receives package-publication and review secrets

CVSS 4.0 Base (CVSS-B): 9.2 (Critical) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Package publication delegates execution to webmin/webmin-ci-cd using a branch name rather than an immutable commit. Across the reviewed changes this appears as downloaded bootstrap code and then a reusable workflow; the caller forwards SSH, GPG, upload-target, and later code-review credentials. Anyone able to maliciously change or compromise the referenced branch can execute code in subsequent caller runs with those secrets, potentially stealing credentials or altering published artifacts. The finding is limited to the demonstrated trust configuration and does not claim an observed compromise.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: Package CI invokes a mutable external workflow while publication or review secrets are supplied. [environmental]

Audited source: `.github/workflows/webmin.dev+webmin.yml`.

Module · Samba (1)

WEBMIN-0564 — Samba domain-join password is exposed in process arguments and response HTML

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The Samba domain-join handler concatenates a submitted password into the `net join -U user%password` command, prints that command into the authenticated result page, and executes it through a shell-backed helper. This exposes a potentially privileged domain credential through both local process arguments and response content. The frozen 2.670 lab used only a synthetic marker and a controlled fake `net`; UID 65534 read the marker from the process cmdline, and the real CGI response contained the same marker.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: Samba domain join runs with a supplied password while a local process observer or response viewer can see it. [environmental]

Audited source: `samba/save_winbind.cgi`, `web-lib-funcs.pl`.

Module · Servers (2)

WEBMIN-0760 — Delegated Servers users can retrieve stored remote Webmin credentials as JSON

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The JSON endpoint added by the manifest commit passes every field of each ACL-selected server record to print_json. Those records contain the remote Webmin username and plaintext password and also gain an internal file path. Server selection is the only authorization applied: the endpoint does not enforce the module's separate edit or links permissions. An authenticated user delegated only visibility of selected servers can therefore retrieve credentials that may grant administrative access to another managed Webmin host. The path requires authentication and server-selection access.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: Servers delegate can view selected linked servers while edit or password-related permissions are withheld. [nondefault]; Those server records contain reusable remote credentials. [environmental]

Audited source: `servers/index-json.cgi`, `servers/servers-lib.pl`.

WEBMIN-0919 — Default fast RPC exposes privileged management traffic on raw TCP

CVSS 4.0 Base (CVSS-B): 7.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`.

The reviewed defaults cause newly registered servers to select fast RPC, and the accompanying editor/save behavior can apply that selection without a visible form field. In the frozen implementation, Webmin authenticates a request to fastrpc.cgi over HTTP(S), receives a port and session identifier, closes the HTTP(S) connection, and opens a direct TCP socket. Serialized privileged RPC requests and replies, including the session identifier used in framing, then traverse that socket without a TLS wrapper. A passive network-path adversary can disclose management arguments and results; an active network-path adversary may alter calls or responses despite TLS and certificate checking on the initial negotiation.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: A linked Webmin server uses default fast RPC and an observer can reach its raw TCP connection. [environmental]

Module · Squid (1)

WEBMIN-0645 — Delegated Squid manual editor can read arbitrary root files

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

An authenticated Webmin user restricted to the Squid module and its manual-editor capability can supply an arbitrary file query parameter to edit_manual.cgi. The endpoint reads that path as the Webmin operating-system account, normally root, without checking get_all_config_files or the delegated root prefix. This exposes arbitrary root-readable files, including material well outside the user's Squid configuration scope.

Profile-based footprint: 85.82% observed module signal (5,798/6,756 usable profiles; observed gate: squid).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `squid/edit_manual.cgi`, `squid/save_manual.cgi`.

Module · Status (2)

WEBMIN-0812 — Webhook notifications disclose stored monitor credentials

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

Status monitor definitions can hold service authentication material in ordinary keys such as user and pass. send_status_webhook enumerates every non-empty service key except underscore-prefixed internal keys, labels each as service_<key>, URL-encodes only the value, and adds the complete set to the outbound request target. Consequently, a webhook-enabled transition sends stored HTTP, FTP, LDAP, or database monitor passwords to the webhook destination; request and debug logs can retain them, and the feature permits plaintext HTTP. The destination must be configured and the monitor must enable webhook notification, but neither action communicates or limits this unintended credential export.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: Status monitor has a credential-bearing service definition and webhook notifications enabled. [nondefault]; A monitor transition causes a webhook request to an observable receiver. [environmental]

Audited source: `status/monitor.pl`, `status/http-monitor.pl`.

WEBMIN-0865 — Status webhooks and debug output disclose saved monitor credentials

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The status webhook builder copies every non-private, non-empty monitor property into service_<key> URL query parameters. Credential-bearing SMTP and IMAP monitor records store pass and user properties, so webhook delivery includes those secrets; debug mode prints the same serialized parameter string to its output. This unnecessarily exposes credentials to the webhook receiver and common URL/debug logging surfaces. The frozen 2.670 function reproduced service_pass in both the request path and debug output using a synthetic monitor record.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: A credential-bearing Status monitor has webhooks or debug mode enabled. [nondefault]

Audited source: `status/monitor.pl`, `status/smtp-monitor.pl`.

Module · Useradmin (1)

WEBMIN-0056 — macOS password material is written to diagnostic output

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The macOS account support path historically printed plaintext password state to STDERR and its password-hash setter still prints the supplied password hash. Diagnostic collectors or operators with log access can receive credential material that should remain confined to the password update transaction and protected account store.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: macOS account password workflow runs while a local observer can read diagnostic output. [environmental]

Audited source: `useradmin/macos-lib.pl`.

Module · Usermin (3)

WEBMIN-0678 — Usermin self-upgrade downloads privileged packages over plaintext HTTP

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Usermin latest-package upgrade workflow constructs a plaintext HTTP SourceForge URL, passes the parsed non-TLS setting into Webmin's downloader, and then installs the downloaded Debian or RPM package or executes setup from extracted tar content. A network attacker controlling the unauthenticated first hop can replace the response or redirect and thereby supply privileged installation content. Triggering the path requires an upgrade-authorized Webmin principal to initiate the operation.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: An upgrade-authorized principal initiates Usermin latest-package upgrade across an attacker-controlled network path. [environmental]

Audited source: `usermin/upgrade.cgi`, `web-lib-funcs.pl`.

WEBMIN-0725 — Usermin Authy API key is retained in raw action-log parameters

CVSS 4.0 Base (CVSS-B): 6.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The Authy provider accepts an application API key in authy_apikey. After saving, the Usermin handler passes the complete input hash to webmin_log; the generic logger serializes every scalar parameter, and the Actions Log raw view displays parsed parameters. This duplicates the credential into a second storage and disclosure surface. Frozen latest-stable runtime testing submitted an explicit non-secret canary through the real handler and confirmed that it appeared in the action log.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin Authy provider is configured with an API key. [nondefault]; A Webmin Log reader can inspect raw action parameters. [environmental]

Audited source: `webmin/twofactor-funcs-lib.pl`, `usermin/change_twofactor.cgi`.

WEBMIN-0750 — Unconstrained Usermin redirect target receives a one-time switch-user credential

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

In frozen Webmin 2.670, Usermin's change_web.cgi stores redirect_url after checking only for whitespace, and get_usermin_login_base_url returns that value verbatim. create_usermin_login_url then appends /session_login.cgi?session=<sid>, where the SID represents a five-minute, one-time, source-IP-bound login as the selected Usermin user; switch.cgi redirects the operator's browser to that URL. A hostile redirect target therefore receives the credential in its request URL. The save CGI also contains no inline $access{'web'} authorization check before writing the value. The path requires configuration influence and a later privileged switch-user action. Redemption must also satisfy the token's one-time and source-IP constraints.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin redirect_url is set to an attacker-controlled endpoint. [nondefault]; An operator follows switch-user while its source-IP-bound one-time session remains valid. [environmental]

Audited source: `usermin/usermin-lib.pl`, `usermin/change_web.cgi`.

Authentication and session handling (14)

Core · ACL (2)

WEBMIN-0720 — Authenticated-session ID generation has a non-cryptographic fallback

CVSS 4.0 Base (CVSS-B): 9.2 (Critical) · `CVSS:4.0/AV:N/AC:H/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

generate_random_session_id prefers 16 bytes from /dev/urandom but falls back to Perl rand when that read path fails. create_session_user uses the resulting 32-hex value as the key for a session already authenticated as the supplied username. Under entropy-source failure, reproducible PRNG state therefore yields reproducible authentication tokens. Exploitation additionally depends on a session-creation caller and applicable session-IP behavior.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: The entropy source read fails during authenticated-session creation and PRNG state is predictable. [environmental]

Audited source: `acl/acl-lib.pl`, `miniserv.pl`.

WEBMIN-2457 — Deduplicated password history can preserve stale password age

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

The old-password helper updates lastchange only when it inserts a prior hash that is not already in the history list. If the actual current hash is already present and the password changes, the hash is correctly not duplicated but lastchange also remains stale. Miniserv uses that timestamp for configured password expiry and account locking, so a new credential can immediately inherit an old expiry or lock state. This is an availability and policy-accuracy observation; password verification and history comparison remain in place.

Profile-based footprint: 93.83% observed module signal (6,339/6,756 usable profiles; observed gate: acl).

Recorded conditions: Webmin password-history or expiry policy is enabled. [nondefault]; The old hash is already present in the history list when a password changes. [environmental]

Audited source: `acl/acl-lib.pl`, `miniserv-lib.pl`.

Core · Webmin (2)

WEBMIN-0030 — Timed user login blocks do not expire automatically

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

The per-user unblock loop uses the blocked-host array as its loop bound and lookup source while splicing the blocked-user array. With no blocked hosts, an expired user entry is never inspected, allowing repeated failed logins to deny that account beyond the configured finite duration until an administrator clears it.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: A finite Webmin user-login block interval is configured. [nondefault]

Audited source: `miniserv.pl`.

WEBMIN-0872 — Proxy certificate headers could authenticate users without a trusted-peer allowlist

CVSS 4.0 Base (CVSS-B): 9.3 (Critical) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

In proxy certificate-trust mode, Miniserv accepted an SSL client subject and verification marker from HTTP headers and used the subject to find a Webmin user. Without binding those headers to a known direct proxy peer, a client able to reach Miniserv directly could forge a mapped subject and authenticate without the corresponding certificate.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: Proxy certificate-header trust is enabled without a trusted direct-proxy source list. [nondefault]; A client can reach MiniServ directly and forge a mapped subject header. [environmental]

Audited source: `webmin/change_access.cgi`, `miniserv-lib.pl`.

Module · Exports (1)

WEBMIN-0459 — Editing a single-flavor GSS export silently removes its Kerberos restriction

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

Existing-export links contain only an index. The editor can infer NFSv4 and display the selected krb5p flavor, but it places the original missing request ver into the hidden form field. On save, mode 3 first produces an empty host, and the code that would encode a single security flavor as gss/krb5p is skipped because the posted version is empty. A normal authenticated save of a single-flavor gss/krb5p(rw) export consequently writes an all-client (rw) export with the default AUTH_SYS security flavor. This removes both the client restriction and Kerberos privacy requirement without warning.

Profile-based footprint: 4.11% observed module signal (278/6,756 usable profiles; observed gate: exports).

Recorded conditions: An existing NFSv4 export has a single gss/krb5p security flavor and is saved through Webmin's editor. [environmental]

Audited source: exports/index.cgi, exports/edit_export.cgi.

Module · Fail2Ban (1)

WEBMIN-2449 — GET-based jail unban becomes CSRF-capable when no-Referer requests are allowed

CVSS 4.0 Base (CVSS-B): 6.9 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N.

The Fail2Ban status UI exposes an unban operation as a GET link containing the jail and IP, and the target CGI accepts ReadParse input without requiring POST or a request token before invoking the privileged fail2ban-client unban command. Frozen stable 2.670 normally rejects foreign and absent Referer requests and issues a SameSite=Lax session cookie, so a fresh install is protected. However, when an administrator enables Webmin's documented no-Referer allowance (referers_none=0), a cross-site top-level GET can carry the Lax cookie and the endpoint performs the requested unban. This is a conditional defense-in-depth observation, not an unconditional bypass of the stable defaults.

Profile-based footprint: 93.75% observed module signal (6,334/6,756 usable profiles; observed gate: fail2ban).

Recorded conditions: No-Referer requests are allowed or equivalent request-origin checks are relaxed. [nondefault]; A Fail2Ban-admin session follows an attacker-chosen GET unban URL. [environmental]

Audited source: fail2ban/list_status.cgi, fail2ban/unblock_jailed_ip.cgi.

Module · Fastrpc (1)

WEBMIN-1037 — An unauthenticated first peer can destroy a FastRPC control session

CVSS 4.0 Base (CVSS-B): 8.7 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N.

FastRPC accepts one raw control connection before validating its session identifier. An unauthenticated peer that reaches the predictable dynamic listener can connect first, submit an invalid identifier to terminate the child, or withhold a frame until timeout, denying the legitimate authenticated operation.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: An authenticated FastRPC operation exposes a predictable dynamic listener to an unauthenticated first peer. [environmental]

Module · File (1)

WEBMIN-0024 — Configured File Manager access root can be deleted

CVSS 4.0 Base (CVSS-B): 7.1 (High) · CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:L/SC:N/SI:N/SA:N.

The File Manager path boundary treats the configured allowed root as an accessible path, but deletion needs an additional invariant that the anchor itself cannot be removed. The historical commit adds that exact guard. Frozen stable 2.670's rewritten filemin module accepts '.' as equal to the current allowed root and passes it to recursive unlink_file; the isolated authenticated reproduction deleted a non-empty configured root. A write-capable File Manager user can therefore remove the boundary anchor and its contents, causing destructive data loss or service disruption.

Profile-based footprint: 0.41% observed module signal (28/6,756 usable profiles; observed gate: file).

Recorded conditions: File Manager delegate is confined to an allowed root but can perform write or delete operations there. [nondefault]

Audited source: filemin/delete.cgi, filemin/filemin-lib.pl.

Module · Iscsi Server (1)

WEBMIN-0349 — Bulk deletion can preserve a selected iSCSI credential and delete another user

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The bulk iSCSI-user handler stores each selected record's original auths-file line number, then attempts reverse deletion with string comparison rather than numeric comparison. Mixed-width line numbers sort incorrectly: line 2 is processed before line 10. The first splice shifts later records, while the second selected object still carries line 10, so a different record is deleted and the selected line-10 credential remains. Webmin logs the selected deletion count even though the resulting credential set differs from the request.

Profile-based footprint: 0.00% observed (0/6,756 usable profiles; gate: iscsi-server). No numeric deployment bound follows from a negative asset probe.

Recorded conditions: iSCSI auths-file entries span mixed-width line numbers, such as 2 and 10, during bulk deletion. [environmental]

Audited source: `iscsi-server/delete_users.cgi`, `iscsi-server/iscsi-server-lib.pl`.

Module · Lvm (1)

WEBMIN-2432 — LV deletion cleanup can detach unrelated numeric device-mapper entries

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N`.

The LV objects returned by list_logical_volumes store the logical-volume identifier in `name`. The cleanup added to delete_logical_volume instead concatenates the VG with the undefined `lv` field, reducing the intended mapper name to a VG-only prefix. Its anchored expression then accepts that prefix followed by optional `p` and decimal digits and passes every match to dmsetup remove before invoking lvremove on the requested device. Consequently, deleting one LV can attempt to detach an unrelated numeric mapper object. Exploitation requires authenticated LVM-module access, a suitable mapper-name collision, and a mapping that dmsetup can remove; busy devices commonly fail closed. The demonstrated effect is limited to availability and device-scope integrity for a removable mapper object; no RCE, shell injection, data overwrite, or authentication bypass was demonstrated.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: lvm).

Recorded conditions: An LV deletion uses a VG name that prefixes an unrelated numeric device-mapper entry. [environmental]

Audited source: `lvm/lvm-lib.pl`.

Module · Samba (1)

WEBMIN-0308 — Failed Samba password changes are reported as successful

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

set_password invokes smbpasswd through execute_command and stores its exit status in local $ex, but returns the negation of unrelated $rv. Callers such as save_euser.cgi interpret a true value as success, so a non-zero smbpasswd exit can leave the old credential active while Webmin completes and logs the account update without the intended failure error.

Profile-based footprint: 84.81% observed module signal (5,730/6,756 usable profiles; observed gate: samba).

Recorded conditions: A Samba password-change command fails while Webmin saves the associated account edit. [environmental]

Audited source: `samba/samba-lib.pl`, `samba/save_euser.cgi`.

Module · Useradmin (1)

WEBMIN-2420 — Dictionary-password policy fails open when dictionary files cannot be read

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

When `passwd_dict` is enabled, `check_password_restrictions` rejects a candidate only if `is_dictionary_word` returns true. That helper tries configured or default word files without checking whether each `open_readfile` call succeeded, then returns false after the list is exhausted. Consequently an unavailable dictionary source silently disables the promised dictionary-word rejection for otherwise eligible passwords. The behavior weakens an optional local password-policy control; it does not itself bypass Webmin authentication or grant privileges.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: Dictionary-password checking is enabled. [nondefault]; Configured or fallback dictionary files cannot be opened during password validation. [environmental]

Audited source: `useradmin/config.info`, `useradmin/user-lib.pl`.

Module · Usermin (2)

WEBMIN-0506 — Usermin authentication form can silently re-enable persistent login

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The Usermin authentication editor loads /etc/usermin/config into uconfig but renders the permanent-login checkbox from the distinct Webmin-global gconfig namespace. The save endpoint then writes the submitted checkbox back to uconfig.noremember. When uconfig.noremember is 1 and gconfig.noremember is absent, the form incorrectly shows permanent login as enabled; saving any setting changes uconfig.noremember to 0. This is an authorized-administrator workflow error rather than an authorization bypass, but it silently weakens a session-persistence policy.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Usermin has noremember=1 while Webmin's global noremember setting is absent, and an operator saves authentication settings. [nondefault]

Audited source: `usermin/edit_session.cgi`, `usermin/change_session.cgi`.

WEBMIN-2427 — Privileged Usermin upgrade endpoint accepts state-changing GET parameters

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:A/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The Usermin upgrade endpoint accepts ordinary query-string parameters on GET and proceeds into the same privileged upgrade logic used by form submissions. The upgrade ACL remains mandatory and referer validation is an important defense, so this is not an unauthenticated or direct authorization bypass. When an installation disables or relaxes referer validation, however, an authenticated administrator can be navigated to a URL that requests an official-source upgrade or forced reinstall. Requiring POST at the endpoint would remove that configuration-dependent request channel.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: Referer validation is disabled or relaxed for the Usermin upgrade endpoint. [nondefault]; An upgrade-authorized administrator has an active Webmin session that can be induced to follow a GET URL. [environmental]

Audited source: `usermin/upgrade.cgi`, `web-lib-funcs.pl`.

Transport and cryptography (19)

Core · Packaging (1)

WEBMIN-0900 — Repository compatibility wrapper executes mutable upstream code as root

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The legacy repository setup wrapper downloads a shell script from the mutable master branch and executes it with the caller's privileges without binding the content to a release, revision, hash, or signature. A compromised or unintentionally changed upstream response therefore becomes privileged shell code when an administrator uses the wrapper.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: An administrator executes the legacy repository wrapper while its mutable upstream response is compromised. [environmental]

Audited source: `webmin-setup-repos.sh`.

Core · Webmin (2)

WEBMIN-0533 — Remote module and theme installs skip signature verification by default

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The module and theme installation forms initialize the signature choice to off. Their handlers fetch remote archives and call check_update_signature only when the submitted checksig field is true; otherwise the same archive is passed directly to install_webmin_module. Because installed Webmin modules contain server-side executable code, archive substitution on an administrator-initiated remote install can become privileged code execution. The isolated frozen-2.670 lab installed a benign unsigned module while making no signature-file request.

Profile-based footprint: 5.39% observed module signal (364/6,756 usable profiles; observed gate: webmin).

Recorded conditions: An administrator installs a remote module or theme while an attacker can substitute the downloaded archive. [environmental]

WEBMIN-0641 — TOTP seeds use a predictable non-cryptographic PRNG

CVSS 4.0 Base (CVSS-B): 9.2 (Critical) · `CVSS:4.0/AV:N/AC:H/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The default TOTP seed is generated by repeated calls to Perl rand after seed_random invokes srand with time, process ID, and a raw four-byte string. Perl rand is not a cryptographic generator; the numeric XOR normally coerces non-numeric random bytes to zero, and seeds separated by 2^32 yield the same generated secret. This sharply reduces the effective entropy of a long-lived authentication key and can enable second-factor prediction when enrollment state can be narrowed.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: TOTP enrollment generates a secret and an attacker can estimate the PRNG seed inputs. [environmental]

Audited source: `webmin/twofactor-funcs-lib.pl`, `web-lib-funcs.pl`.

Module · Cluster Usermin (1)

WEBMIN-0758 — Unauthenticated Debian download feeds privileged Usermin cluster upgrade

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The Cluster Usermin latest-package path requests the release index and Debian artifact through http_download without setting the function's HTTPS-mode argument, then passes the downloaded artifact to dpkg --install without signature or digest validation. Correcting the Debian filename makes the intended artifact path operational but does not authenticate its bytes. An active network intermediary can therefore substitute a package whose maintainer scripts execute with upgrade privileges on selected managed hosts.

Profile-based footprint: 85.07% observed module signal (5,747/6,756 usable profiles; observed gate: cluster-usermin).

Recorded conditions: A Cluster Usermin Debian upgrade downloads through an attacker-controlled HTTP path. [environmental]

Module · Cluster Webmin (1)

WEBMIN-0358 — Default cluster upgrade downloads executable packages over unauthenticated HTTP

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

Cluster Webmin's default latest-version upgrade option constructs the package location with plain HTTP and leaves package-signature verification unselected. Verification runs only when the sig request parameter is supplied. The downloaded artifact is then copied to selected managed hosts and passed to RPM, DEB, or tar setup installation logic with administrative privileges. A network-positioned attacker able to substitute the cleartext response could therefore supply installation code that compromises the selected cluster members. The controlled latest-stable reproduction confirmed the transport trust failure without executing a package payload.

Profile-based footprint: 4.19% observed module signal (283/6,756 usable profiles; observed gate: cluster-webmin).

Recorded conditions: A cluster administrator initiates latest-version upgrade over an attacker-controlled HTTP path. [environmental]

Audited source: `cluster-webmin/index.cgi`, `cluster-webmin/upgrade.cgi`.

Module · Cpan (1)

WEBMIN-0499 — Plain-HTTP CPAN retrieval enables pre-confirmation root code execution

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The CPAN module obtains its module index and selected archives over plain HTTP without an authenticated integrity check. An on-path attacker can replace the index entry and archive for a module an authorized user requests. download.cgi extracts that archive and runs its Makefile.PL while calculating prerequisites, before displaying the final install form. Because Webmin normally runs as root, this supplies attacker-controlled UID-0 command execution. The path requires an authorized module installation request and an attacker able to intercept the HTTP download.

Profile-based footprint: 4.25% observed module signal (287/6,756 usable profiles; observed gate: cpan).

Recorded conditions: No additional deployment prerequisite is stored; actor and trigger details are in the finding narrative.

Audited source: `cpan/config`, `cpan/cpan-lib.pl`.

Module · Fsdump (1)

WEBMIN-1036 — FTPS peer certificates are not authenticated

CVSS 4.0 Base (CVSS-B): 7.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`.

The fsdump FTP helper upgrades its control and data sockets with IO::Socket::SSL while explicitly setting SSL_verify_mode to 0. No CA trust or endpoint-name verification is supplied. The helper then transmits USER and PASS after the control handshake, so TLS encrypts the session without authenticating the FTPS server. An interceptor or redirected endpoint can therefore present an arbitrary certificate and receive the

configured backup credentials; the reused control context also does not add an independent authenticated identity for the data channel.

Profile-based footprint: 5.31% observed module signal (359/6,756 usable profiles; observed gate: fsdump).

Recorded conditions: FSdump backup uses FTPS with configured server credentials. [nondefault]; An interceptor or redirected endpoint supplies an arbitrary certificate. [environmental]

Audited source: `fsdump/ftp.pl`.

Module · LDAP Server (1)

WEBMIN-2402 — LDAP administrator password hashing can fall back to legacy crypt

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

The historical password-change handler unconditionally hashes the LDAP administrator password with unix_crypt using a two-character time-derived salt and stores it as {crypt}. The frozen newest-stable helper prefers slappasswd but retains the same legacy method as its fallback. This verifier is fast, minimally salted, and truncates password significance after eight characters on the reproduced platform, so disclosure of slapd.conf or a backup materially lowers the cost of offline password recovery.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: LDAP administrator password hashing falls back to legacy crypt when slappasswd is unavailable. [environmental]

Audited source: `ldap-server/save_slapd.cgi`, `ldap-server/ldap-server-lib.pl`.

Module · LDAP Users and Groups (2)

WEBMIN-0016 — LDAP TLS connections do not require server-certificate verification

CVSS 4.0 Base (CVSS-B): 7.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`.

The shared LDAP client enables StartTLS or LDAPS but does not request required server-certificate verification. In the StartTLS path, even a configured CA file is passed only when a client certificate exists, and neither branch supplies verify=require before the password bind. Encryption without peer authentication permits a network-positioned attacker to impersonate the LDAP endpoint and receive the bind credential. The latest-stable call-flow harness recorded no verify or cafile argument before a password-bearing bind.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: ldap-client).

Recorded conditions: Webmin binds to an LDAP server over StartTLS or LDAPS across an attacker-controlled network path. [environmental]

Audited source: `ldap-client/ldap-client-lib.pl`.

WEBMIN-0307 — Optional unsalted LDAP {SHA} password storage weakens offline resistance

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

When the LDAP Users and Groups module is configured for the optional Unix SHA method, password creation and reset store an LDAP {SHA} value derived from plain SHA-1 without a salt. Identical passwords therefore have identical stored representations, and the fast digest provides substantially less resistance to offline guessing than a salted adaptive password hash. Exploitation requires access to LDAP password attributes; the consequence can include recovery of weak passwords and compromise of the affected LDAP identities or reused credentials.

Profile-based footprint: 4.17% observed module signal (282/6,756 usable profiles; observed gate: ldap-useradmin).

Recorded conditions: LDAP password encryption is set to the optional unsalted Unix SHA method. [nondefault]; An attacker obtains stored LDAP password attributes for offline guessing. [environmental]

Module · Mailboxes (2)

WEBMIN-0402 — SMTP SSL accepts unauthenticated server certificates

CVSS 4.0 Base (CVSS-B): 7.6 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`.

The SMTP SSL helper completes a Net::SSLeay handshake without enabling certificate-chain verification or checking the server hostname. A network attacker can present an arbitrary certificate, impersonate the configured SMTP server, and observe authentication material and message content.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: Mailboxes sends SMTP over SSL through a network path where an attacker can impersonate the server. [environmental]

Audited source: `mailboxes/boxes-lib.pl`.

WEBMIN-0695 — IMAP TLS does not authenticate the server before sending credentials

CVSS 4.0 Base (CVSS-B): 6.0 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The SSL-mode IMAP client establishes TLS with Net::SSLeay but does not verify the certificate chain or configured server name before sending the configured username and password in an IMAP LOGIN command. An active network intermediary or attacker-controlled endpoint can present an arbitrary certificate, complete the handshake, and obtain reusable mailbox credentials. The frozen latest-stable Docker reproduction confirmed this behavior with a self-signed certificate for a mismatched hostname.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: Mailboxes uses SSL-mode IMAP with stored credentials over a network path an attacker controls. [environmental]

Audited source: `mailboxes/folders-lib.pl`.

Module · Pptp Client (1)

WEBMIN-0501 — MPPE form loses explicit ban on 40-bit encryption

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:P/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

mppe_options_form assigns both its positive and negative state variables by looking up the positive MPPE directive. With an existing nomppe-40 entry, neither lookup succeeds, so the form selects Default (Allowed). Submitting that value makes parse_mppe_options delete both spellings, including nomppe-40. This silently weakens an explicit encryption policy. Exploitation is conditional rather than automatic: an authorized configuration save and subsequent negotiation with a peer willing to use 40-bit MPPE are required.

Profile-based footprint: 0.00% observed (0/6,756 usable profiles; gate: pptp-client). No numeric deployment bound follows from a negative asset probe.

Recorded conditions: PPTP client configuration already contains nomppe-40 and an operator saves the MPPE form. [environmental]

Audited source: `pptp-client/pptp-client-lib.pl`, `pptp-client/save_opts.cgi`.

Module · Servers (2)

WEBMIN-0504 — Selected certificate checking is skipped by the credential-bearing automatic fast-RPC probe

CVSS 4.0 Base (CVSS-B): 6.0 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

When a linked server is saved with explicit credentials and automatic fast-RPC detection, save_serv.cgi opens the SSL probe without passing any certificate requirements. It then writes the submitted username and password as HTTP Basic credentials, even if the form's checkssl option was selected; that option is only stored later. A network or DNS attacker can therefore terminate TLS with an arbitrary attacker-rooted certificate and capture privileged remote Webmin credentials during the save action.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: Linked-server save selects certificate checking with explicit credentials and automatic fast-RPC detection. [nondefault]; An attacker can intercept or redirect the TLS probe. [environmental]

Audited source: `servers/save_serv.cgi`.

WEBMIN-0505 — Linked-server hostname validation overmatches wildcard certificates

CVSS 4.0 Base (CVSS-B): 5.9 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:L/UI:P/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The TLS hostname validator treats a certificate name of *.example.test as matching the bare suffix and any hostname ending in .example.test. Wildcard identities are expected to cover one leftmost DNS label, not the parent itself or arbitrarily deep names. An attacker who can redirect a linked-server connection and present a CA-acceptable wildcard for the broader suffix can therefore pass the Webmin hostname check for an out-of-scope target and receive credential-bearing RPC traffic.

Profile-based footprint: 95.66% observed module signal (6,463/6,756 usable profiles; observed gate: servers).

Recorded conditions: A linked-server connection is redirected to a host presenting a CA-acceptable wildcard for a broader suffix. [environmental]

Audited source: `web-lib-funcs.pl`.

Module · Status (2)

WEBMIN-0099 — SSL certificate monitor reports an untrusted wrong-host certificate as Up

CVSS 4.0 Base (CVSS-B): 6.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The SSL URL monitor invokes openssl s_client without making certificate-chain verification failures fatal, extracts any parseable peer PEM, and bases success primarily on certificate dates. Its optional hostname check runs only when a narrow legacy Subject parser matches and then treats certificate DNS names as Perl regular expressions without escaping ordinary dots. If the parser does not match, no mismatch description is set and the function falls through to Up. A remote endpoint operator or network interceptor can therefore present a self-signed and wrong-host certificate with acceptable dates and suppress the configured certificate alarm. This is a monitoring-integrity failure, not direct code execution or compromise of the Webmin host.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: An SSL monitor checks a remote endpoint whose certificate is untrusted or for the wrong host. [environmental]

Audited source: `status/sslcert-monitor.pl`.

WEBMIN-0864 — SMTP monitor credentials are not protected by authenticated encryption

CVSS 4.0 Base (CVSS-B): 8.2 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`.

The SMTP status monitor offers STARTTLS mode 2 but tests package global $ssl instead of the saved $serv->{'ssl'} setting. Authentication then uses SASL PLAIN, allowing credential material to be transmitted without the selected upgrade. The immediate-TLS helper also establishes Net::SSLeay without peer-certificate or hostname verification, so an active intermediary can impersonate that endpoint. The frozen 2.670 source retains both conditions, and its exact SMTP function reproduced successful authentication with no STARTTLS command or TLS switch in a network-disabled protocol harness.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: SMTP status monitor is configured for STARTTLS or immediate TLS with credentials. [nondefault]; A rejecting or impersonating SMTP peer is reachable during the monitor check. [environmental]

Audited source: `status/smtp-monitor.pl`.

Module · Useradmin (1)

WEBMIN-0234 — Generated account passwords are predictable from low-entropy process state

CVSS 4.0 Base (CVSS-B): 9.2 (Critical) · `CVSS:4.0/AV:N/AC:H/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The shared 15-character account-password generator calls Webmin's seed_random and then Perl rand. In frozen 2.670, seed_random reads four bytes from /dev/urandom but numerically XORs the raw byte string with time and PID; common non-numeric buffers coerce to zero and numeric prefixes contribute only a small decimal value. The locked runtime probe generated with the stable algorithm and predicted all eight observed outputs from time and PID alone. The weakness reduces a nominally large generated-password space to an enumerable seed search for attackers who can estimate creation time and PID.

Profile-based footprint: 95.63% observed module signal (6,461/6,756 usable profiles; observed gate: useradmin).

Recorded conditions: Account creation uses generated passwords and an attacker can estimate the creation time and process ID. [environmental]

Audited source: `useradmin/user-lib.pl`, `web-lib-funcs.pl`.

Module · Usermin (1)

WEBMIN-0839 — Unauthenticated HTTP package delivery in Usermin Debian self-upgrade

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The automatic Usermin Debian upgrade path constructs a cleartext HTTP URL for a newkey-prefixed package, downloads the response to a local file, checks only the archive's package name and version, and invokes dpkg --install. There is no explicit signature or digest verification between network delivery and installation. A network-position attacker able to replace the response can supply a malicious Debian archive whose maintainer scripts execute with the upgrade process's privileges.

Profile-based footprint: 94.75% observed module signal (6,401/6,756 usable profiles; observed gate: usermin).

Recorded conditions: A Usermin Debian upgrade follows an attacker-controlled plaintext HTTP package response. [environmental]

Audited source: `usermin/upgrade.cgi`, `web-lib-funcs.pl`.

Request forgery and origin checks (4)

Core · Cross-cutting (1)

WEBMIN-0874 — Late address-policy callback permits blind HTTP requests to non-public services

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

When an affected URL-download feature is configured to allow only public destinations, Webmin opens the connection and writes the HTTP request inside make_http_connection before http_download invokes callback mode 7. Filemin, Mailboxes, and Updown enforce the non-public-address policy from that callback, so the response is rejected but the request has already reached the destination. An authenticated delegated user with the relevant fetch capability can therefore issue blind GET requests from the Webmin host to loopback, private,

link-local, or reserved HTTP services, potentially activating state-changing endpoints reachable only from the managed server.

Profile-based footprint: 100.00% assigned core gate (6,756/6,756 usable profiles). Core is a classification in the applicability model, not an observed module probe.

Recorded conditions: A URL downloader is configured to reject non-public addresses through the late callback. [nondefault]; An authenticated URL input targets a non-public HTTP service reachable from Webmin. [environmental]

Audited source: `web-lib-funcs.pl`, `filemin/http_download.cgi`.

Module · Grub2 (1)

WEBMIN-1012 — Remote theme installation permits requests to non-public destinations

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

save_theme.cgi requires only the GRUB 2 edit ACL and accepts a remote theme source. The stable helper recognizes HTTP, HTTPS, and FTP URLs, parses the destination, and passes it directly to Webmin's download routines. It does not attach Webmin's available public-address checker, so loopback and private addresses are permitted. The network-isolated stable-image lab started a service only on container loopback; the module issued GET /theme.txt to 127.0.0.1 and installed the exact 34-byte response. This gives a delegated editor a server-side request primitive against services reachable from the Webmin host.

Profile-based footprint: 0.01% observed module signal (1/6,756 usable profiles; observed gate: grub2).

Recorded conditions: A GRUB 2 edit-authorized user submits a private or loopback theme URL reachable from Webmin. [environmental]

Audited source: `grub2/save_theme.cgi`, `grub2/grub2-lib.pl`.

Module · Mailboxes (1)

WEBMIN-0914 — Mailbox image proxy sends private-network HTTP request before destination-policy rejection

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`.

The authenticated mailbox image-proxy action accepts a caller-selected URL and performs the HTTP operation from the Webmin host. At the reviewed historical commit, the action supplied no destination validator. Frozen stable 2.670 adds a global download-address policy, but http_download invokes that callback only after make_http_connection has connected and written the HTTP request. Consequently, a restricted mail-module user can still cause blind HTTP GET requests to loopback and other non-public destinations even though stable withholds the response and reports a policy error. A network-none Docker reproduction using the frozen stable source and a safe user configured for public-only downloads observed the loopback request and confirmed response non-disclosure.

Profile-based footprint: 94.32% observed module signal (6,372/6,756 usable profiles; observed gate: mailboxes).

Recorded conditions: An authenticated Mailboxes image-proxy user supplies a private HTTP target reachable from Webmin. [environmental]

Audited source: `mailboxes/xhr-lib.pl`, `web-lib-funcs.pl`.

Module · Minecraft (1)

WEBMIN-2445 — Minecraft URL installer permits blind SSRF from the Webmin host

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The authenticated Minecraft JAR installer parses an operator-controlled URL and passes its host, port, path, and scheme directly to Webmin's HTTP downloader. No destination policy excludes loopback, link-local, or private addresses. The downloaded body is processed as a JAR candidate rather than disclosed directly, making

this blind SSRF. The path requires module authorization and an accepted request under normal origin checks; the downloaded response is processed as a JAR candidate.

Profile-based footprint: 0.04% observed module signal (3/6,756 usable profiles; observed gate: minecraft).

Recorded conditions: An authenticated Minecraft installer submits a private or loopback JAR URL reachable from Webmin. [environmental]

Audited source: `minecraft/add_version.cgi`, `web-lib-funcs.pl`.

Availability and race conditions (7)

Module · Bind8 (1)

WEBMIN-0538 — Zone comment parsing has reproducible superlinear CPU cost

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N`.

The fallback expression used to strip zone-file comments contains nested repetition and is explicitly identified in source as very slow. A malformed long zone line produced approximately fourfold CPU growth each time input length doubled. A user who can place raw text in a zone file and induce Webmin parsing can consume request CPU and degrade the administrative service.

Profile-based footprint: 94.63% observed module signal (6,393/6,756 usable profiles; observed gate: bind8).

Recorded conditions: An attacker can place a long malformed line in a BIND zone file and induce Webmin to parse it. [environmental]

Audited source: `bind8/records-lib.pl`.

Module · Cron (2)

WEBMIN-0642 — Cron next-run display permits authenticated CPU exhaustion through wildcard schedules

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

Cron's next_run routine expands each schedule field into arrays, iterates the Cartesian product of minute, hour, day, month, and up to eight candidate years, stores all candidate timestamps, and sorts them. Rendering a normal * * * * * job therefore performs at least 535,680 date combinations rather than stopping at the first future minute. In Webmin 2.670 the routine remains called from authenticated Cron list and edit pages when next-run display is enabled. A permitted user can repeatedly request such views, or a delegated Unix user can populate broad schedules that make an administrator's list view consume CPU per job. This is a transient availability issue; it does not cross an integrity or confidentiality boundary.

Profile-based footprint: 96.86% observed module signal (6,544/6,756 usable profiles; observed gate: cron).

Recorded conditions: A Cron module user opens next-run display for a broad wildcard job. [environmental]

Audited source: `cron/cron-lib.pl`, `cron/index.cgi`.

WEBMIN-0657 — Zero-step cron schedule can exhaust a Webmin request worker

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

Cron schedule expansion uses the numeric step as the loop increment. A zero step leaves the iterator unchanged while the upper-bound test remains true. The manifest change guards cron_range, but the frozen newest-stable get_times_input and show_times_input copies still use the unguarded pattern. In the isolated newest-stable container, a */5 control returned immediately and */0 stayed in get_times_input until a three-second alarm, demonstrating per-request availability impact.

Profile-based footprint: 96.86% observed module signal (6,544/6,756 usable profiles; observed gate: cron).

Recorded conditions: An authenticated Cron user submits or renders a */0 schedule. [environmental]

Module · DHCP (1)

WEBMIN-0492 — Authenticated DHCP range expansion can exhaust lease-viewer memory

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N`.

The DHCP lease viewer expands every configured range into an in-memory array and then iterates that array into additional tracking structures. Range-saving paths validate endpoint syntax but do not cap the number of addresses between endpoints. A DHCP module user with range-write and lease-view permissions can therefore save a large valid range and trigger memory exhaustion by opening the lease viewer. The primary demonstrated impact is availability; no command execution, data disclosure, or privilege escalation was established.

Profile-based footprint: 84.87% observed module signal (5,734/6,756 usable profiles; observed gate: dhcpd).

Recorded conditions: DHCP delegate can save a very large valid range and view leases. [environmental]

Audited source: `dhcpd/dhcpd-lib.pl`, `dhcpd/list_leases.cgi`.

Module · Filemin (1)

WEBMIN-0717 — Extract-on-upload permits decompression resource amplification

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N`.

Extract-on-upload dispatches compressed content to external tools without limiting expanded bytes, extracted entry count, elapsed work, or destination filesystem consumption. The request-size limit applies to compressed input, so an authenticated uploader can amplify a small permitted request into substantially larger host storage and processing use.

Profile-based footprint: 94.83% observed module signal (6,407/6,756 usable profiles; observed gate: filemin).

Recorded conditions: Filemin extract-on-upload is enabled for an authenticated uploader. [nondefault]; The uploaded compressed archive expands beyond available processing or storage resources. [environmental]

Audited source: `filemin/filemin-lib.pl`, `filemin/upload.cgi`.

Module · Package Updates (1)

WEBMIN-2435 — Stale progress marker can hide a pending security update after PID reuse

CVSS 4.0 Base (CVSS-B): 2.1 (Low) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:H/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

Package update progress is stored in persistent PID-named files and removed only on normal completion. get_update_progress accepts a record whenever kill(0, stored_pid) succeeds, without checking process identity, marker age, or deleting dead markers. If an updater terminates before cleanup and its PID is later reused, System Information treats the stale package list as active and removes those packages before calculating security-update counts. This does not block the Package Updates page itself, but it can make the dashboard under-report or entirely omit a pending security update and delay patching.

Profile-based footprint: 95.00% observed module signal (6,418/6,756 usable profiles; observed gate: package-updates).

Recorded conditions: An updater leaves a PID-named progress marker and that PID is later reused by a live process. [environmental]

Audited source: `package-updates/package-updates-lib.pl`, `system-status/system_info.pl`.

Module · Passwd (1)

WEBMIN-0736 — Shared rate-limit lock is held during exponential backoff

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N`.

apply_rate_limit uses one shared state file and holds its exclusive Webmin file lock across `sleep(2 ** count)`. Concurrent requests therefore wait for the sleeping holder before applying their own delay. Because the endpoint invokes this helper before its anonymous-mode check, a Webmin-authenticated caller can reach the behavior even when anonymous integration is not enabled; deployments exposing the intended integration broaden reachability. Repeated concurrency can accumulate waiting CGI processes and impair availability.

Profile-based footprint: 5.54% observed module signal (374/6,756 usable profiles; observed gate: passwd).

Recorded conditions: A caller can reach the password rate-limit endpoint and concurrent requests accumulate while backoff holds the lock. [environmental]

Other security defects (16)

Core · Webmin (1)

WEBMIN-0365 — Upgrade signature verification is unchecked when GnuPG setup succeeds

CVSS 4.0 Base (CVSS-B): 7.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The upgrade form passes gnupg_setup's status directly to the signature checkbox. Success is zero, while ui_checkbox checks only a truthy selected value, so a functional GnuPG installation produces an unchecked verification control. A privileged administrator who accepts that default can submit an upgrade without signature verification, turning compromise or substitution of the selected package source into privileged Webmin code execution.

Profile-based footprint: 5.39% observed module signal (364/6,756 usable profiles; observed gate: webmin).

Recorded conditions: GnuPG setup succeeds and an administrator accepts the unchecked verification box during upgrade. [environmental]

Audited source: `webmin/edit_upgrade.cgi`, `webmin/webmin-lib.pl`.

Module · Cluster Usermin (1)

WEBMIN-2448 — Wrong regex capture can block Debian security-update deployment

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

The Debian metadata expression has one capture group for the Version value, but the matched path assigns $version from $2. For valid dpkg metadata, $1 contains the release and $2 is undefined. Numeric host filtering and remote-version enforcement then operate on an undefined value, which can prevent the cluster workflow from deploying a valid Usermin Debian update. The security consequence is reduced reliability of a mechanism administrators may use to deliver security updates.

Profile-based footprint: 85.07% observed module signal (5,747/6,756 usable profiles; observed gate: cluster-usermin).

Recorded conditions: Cluster Usermin deploys a Debian package with valid Version metadata through the affected host filter. [environmental]

Module · Cluster Webmin (1)

WEBMIN-2410 — Leading-@ usernames collide with nested-group membership encoding

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The clustered user-edit validator accepts a username beginning with @ and stores a renamed user verbatim in group membership data. The ACL library reserves the same leading marker for nested groups and strips it before selecting a group. A member token such as @ops can therefore identify an actual user at creation/rename time but be processed as group ops during recursive module or ACL propagation. This can skip the intended user or modify a same-named group's descendants. The path requires an authenticated administrator with clustered identity-management authority.

Profile-based footprint: 4.19% observed module signal (283/6,756 usable profiles; observed gate: cluster-webmin).

Recorded conditions: A clustered user name begins with @ and collides with nested-group encoding during ACL propagation. [environmental]

Module · LDAP Client (1)

WEBMIN-0450 — LDAP fallback-path bug directs security configuration to the wrong file

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

get_ldap_config_file() splits configured paths into lexical @confs but iterates @$confs, which is a different scalar that Perl autovivifies as an empty array reference. No existence test runs, so the function always returns the first configured path. Where that path is absent and a later path is the active LDAP client file, Webmin reads, validates, and writes against the wrong target. This can silently leave active bind credentials, transport-security, certificate-validation, search-base, or identity-filter settings unchanged.

Profile-based footprint: 84.68% observed module signal (5,721/6,756 usable profiles; observed gate: ldap-client).

Recorded conditions: LDAP client has multiple configured file paths, with the first absent and a later one active. [nondefault]

Audited source: `ldap-client/ldap-client-lib.pl`.

Module · LDAP Server (1)

WEBMIN-0328 — Sorted ACL display indexes target unsorted handler entries

CVSS 4.0 Base (CVSS-B): 7.1 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N`.

The LDIF ACL list sorts olcAccess objects by parsed numeric {n} order and then emits positional indexes for edit, bulk delete, move-up, and move-down actions. Each receiver independently reloads olcAccess in parser order and uses the submitted position without applying the list's sort. If physical value order differs from numeric ACL order, the position names another rule. An authorized operator can consequently edit, delete, or move a different rule; deleting a restrictive rule can broaden directory access.

Profile-based footprint: 85.01% observed module signal (5,743/6,756 usable profiles; observed gate: ldap-server).

Recorded conditions: OpenLDAP ACL values are physically ordered differently from their numeric {n} prefixes. [environmental]

Audited source: `ldap-server/edit_acl.cgi`, `ldap-server/acl_form.cgi`.

Module · Net (1)

WEBMIN-0338 — Unvalidated interface fields can execute root shell code through generated network configuration

CVSS 4.0 Base (CVSS-B): 8.7 (High) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`.

The reviewed network-interface save paths trust request-derived bond-slave and VLAN interface fields more than their downstream uses permit. Those values are embedded in root-executed Debian pre-up directives, direct activation command strings, or Red Hat ifcfg assignments that network scripts interpret as shell. The module

requires authenticated interface-creation permission, but that permission is intended to authorize network configuration rather than arbitrary root shell execution. A forged request can therefore cross the network-management authorization boundary and run chosen shell commands as root when configuration is activated or later sourced.

Profile-based footprint: 95.80% observed module signal (6,472/6,756 usable profiles; observed gate: net).

Recorded conditions: The affected interface save path uses Debian network scripts or a Red Hat ifcfg backend. [environmental]; The saved configuration is activated or later sourced by privileged network tooling. [environmental]

Audited source: `net/save_bifc.cgi`, `net/debian-linux-lib.pl`.

Module · Package Updates (1)

WEBMIN-0944 — Existing Package Updates schedules can silently stop emailing security notices

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

The scheduled-update mail path now treats '*' as the sole marker for using Webmin's global notification address. The module's shipped defaults do not create that marker, so an existing schedule whose module-specific sched_email key is absent or empty yields no recipient even when webmin_email_to is configured. Because sending is guarded by a truthy recipient, update reports—including reports for available security updates—are silently skipped until the setting is explicitly saved with either the global option or a module-specific address.

Profile-based footprint: 95.00% observed module signal (6,418/6,756 usable profiles; observed gate: package-updates).

Recorded conditions: Package Updates schedule lacks sched_email while a global webmin_email_to address is configured. [nondefault]

Audited source: `package-updates/update.pl`, `package-updates/index.cgi`.

Module · Pam (2)

WEBMIN-0091 — PAM option forms silently discard security-relevant flags

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

Several PAM module option forms render names or enumerated values that do not match their parsers. Two security-relevant examples are pam_tally, where no_reset is displayed as current value 2 but the choices are 0, 1, and 0, and pam_rhosts_auth, where the display reads no_hosts while the parser stores no_rhosts. Opening and saving these forms can silently discard authentication hardening even when the operator did not select that policy change.

Profile-based footprint: 86.04% observed module signal (5,813/6,756 usable profiles; observed gate: pam).

Recorded conditions: An operator edits and saves a PAM option entry that has security-relevant flags. [environmental]

Audited source: `pam/pam_tally.so.pl`, `pam/pam_rhosts_auth.so.pl`.

WEBMIN-0092 — Bulk @include removal can delete an adjacent PAM rule

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N`.

The bulk @include saver walks parsed include objects from lowest to highest position and calls delete_module with their original line offsets. Each splice shifts later lines, so a later deletion may target an unselected neighboring authentication rule. The exact newest-stable control flow removed a following pam_deny.so rule from a three-line fixture while leaving one requested include behind.

Profile-based footprint: 86.04% observed module signal (5,813/6,756 usable profiles; observed gate: pam).

Recorded conditions: A PAM configuration contains adjacent @include and ordinary entries, and the operator bulk-deletes an include. [environmental]

Audited source: pam/save_incs.cgi, pam/pam-lib.pl.

Module · PostgreSQL (1)

WEBMIN-0808 — PostgreSQL backup pipeline masks dump-process failures

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N.

backup_database streams pg_dump into cat, gzip, or bzip2 and then relies on the shell pipeline status plus captured output. A normal POSIX shell returns the final writer's status; the appended stderr redirection applies to that final command, so a failing pg_dump can emit an error outside the captured string while a successful writer yields status zero. Webmin can consequently return success for an empty or partial backup, weakening backup integrity and recovery assurance.

Profile-based footprint: 96.11% observed module signal (6,493/6,756 usable profiles; observed gate: postgresql).

Recorded conditions: A PostgreSQL backup runs pg_dump through a pipeline whose final writer succeeds while pg_dump fails. [environmental]

Audited source: postgresql/postgresql-lib.pl.

Module · Proc (1)

WEBMIN-2461 — Process argv can spoof Webmin's hide-self collector marker

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:N/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N.

The Linux process parser marks any row whose argument text contains the complete ps collector command, and the process visibility predicate rejects every marked row whenever hide_self is enabled. Because a local process can choose its argv, an unrelated process can imitate that text and disappear from the affected Webmin process views. The impact is limited to Webmin monitoring visibility; the process remains visible to operating-system tools and this behavior does not grant additional privilege.

Profile-based footprint: 20.43% observed module signal (1,380/6,756 usable profiles; observed gate: proc).

Recorded conditions: Webmin Process Manager hide_self option is enabled. [nondefault]; A local process can spoof the collector command text in argv. [environmental]

Audited source: proc/linux-lib.pl, proc/proc-lib.pl.

Module · Proftpd (1)

WEBMIN-2439 — Translated commas corrupt ProFTPD access-order form values

CVSS 4.0 Base (CVSS-B): 4.8 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:P/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N.

ProFTPD edit_Allow_Deny_Order concatenates a translated label, a comma, and the intended machine value. choice_input splits this string at its first comma. When the label itself contains a comma, the remainder of the label becomes part of the form value, so the valid existing access order is not preselected and choosing the option yields unintended text. parse_choice returns that value without removing the translated prefix. Frozen stable source retains affected labels, including German, Norwegian, Polish, Portuguese automatic, Czech, Russian, Ukrainian, and Turkish resources. This is a configuration-integrity risk for an authorized user editing access-order policy; any daemon rejection, availability effect, or policy-default consequence depends on subsequent configuration handling and was not tested. A network-disabled latest-stable helper check reproduced the rendering/preselection and serialization defect in 12 options from seven locale files, with two English controls passing. No unauthorized access or end-to-end server failure is asserted.

Profile-based footprint: 5.28% observed module signal (357/6,756 usable profiles; observed gate: proftpd).

Recorded conditions: The ProFTPD locale translates an access-order label with a comma and an operator saves the affected form. [environmental]

Audited source: proftpd/proftpd-lib.pl, proftpd/mod_core.pl.

Module · Project Meta (1)

WEBMIN-0834 — Secret-bearing package workflow executes dependencies from mutable Git references

CVSS 4.0 Base (CVSS-B): 9.2 (Critical) · CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N.

The active package workflow executes mutable action references before a later step runs repository-controlled build scripts with deployment SSH keys, a repository-signing passphrase, and the upload destination. A moved or compromised dependency ref can modify the workspace or persist code between steps, giving it a path to those later secrets and package outputs. Immutable action commit pins and isolation of the secret-bearing deployment phase are absent.

Profile-based footprint: N/A — this path cannot be measured with the current running-host module probes.

Recorded conditions: Package workflow runs mutable third-party action references before steps receiving publication secrets. [environmental]

Audited source: .github/workflows/webmin.dev+webmin.yml.

Module · Software (1)

WEBMIN-0710 — APT fallback can skip security updates in security-only scheduling

CVSS 4.0 Base (CVSS-B): 5.1 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N.

When Webmin falls back to `apt list --upgradable`, it accepts a comma-separated suite field but removes everything from the first comma onward before deriving the `security` boolean. If a non-security suite is first and a security suite is later, the record is treated as non-security. The Package Updates scheduler installs only records with a true security field when configured for security-only action, so the affected update is skipped even though the accepted apt source list includes a security suite. This can delay security patch deployment under the policy intended to install those patches automatically.

Profile-based footprint: 1.12% observed module signal (76/6,756 usable profiles; observed gate: software).

Recorded conditions: Package Updates schedule is security-only. [nondefault]; APT fallback reports a non-security suite before a security suite for the same upgrade. [environmental]

Audited source: software/apt-lib.pl, package-updates/update.pl.

Module · SSH Server (1)

WEBMIN-0908 — Keyboard-interactive alias mismatch can falsely report authentication disabled

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:P/VC:N/VI:L/VA:N/SC:N/SI:N/SA:N.

The authentication editor and save handler switch to KbdInteractiveAuthentication on OpenSSH 6.2+, but Webmin's parser and writer match literal directive names and do not coalesce the legacy ChallengeResponseAuthentication alias. With an earlier legacy Yes present, submitting No retains that line and appends KbdInteractiveAuthentication no. The next Webmin view reads only the new spelling and reports No, creating false assurance while the earlier enabled alias can remain the first effective value for the same OpenSSH authentication setting.

Profile-based footprint: 95.20% observed module signal (6,432/6,756 usable profiles; observed gate: sshd).

Recorded conditions: OpenSSH 6.2+ configuration retains legacy ChallengeResponseAuthentication Yes when an operator saves the new spelling as No. [environmental]

Module · Status (1)

WEBMIN-0609 — Alert throttle persists across recovery and can suppress renewed-outage notification

CVSS 4.0 Base (CVSS-B): 5.3 (Medium) · `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:N/VA:L/SC:N/SI:N/SA:N`.

The Status monitor records one last-sent timestamp per monitor and gates every notification channel on email_interval. After processing a successful status, the cleanup branch checks $ok == 1, but $ok is otherwise absent from monitor.pl and status-lib.pl; the computed per-remote status is $up. The stale timestamp therefore survives recovery. If the endpoint fails again before the interval expires, the new event is detected and logged but its email, pager, SMS, SNMP, and webhook notification construction is skipped. An actor able to flap a monitored endpoint or its network path can exploit the timing to hide a renewed outage for the remainder of the configured interval. The demonstrated effect is loss of monitoring availability without OS command execution or a Webmin privilege escalation.

Profile-based footprint: 94.79% observed module signal (6,404/6,756 usable profiles; observed gate: status).

Recorded conditions: Status monitor has a nonzero email_interval and a notification channel. [nondefault]; A monitored service recovers and fails again before the interval expires. [environmental]

Audited source: `status/monitor.pl`, `status/status-lib.pl`.

Confirmed historical security issues (567)

These internally discovered issues have historical source confirmation and a recorded fix comparison at the audited Webmin 2.670 revision. The source links identify historical locations or evidence commits. Selection required a confirmed historical defect and an upstream or audited-revision fix comparison; 21 suspected leads and five local-branch fix records remain outside this cohort.

Access control and ACL bypass (112)

ACL · Core · ACL (5)

- **WEBMIN-0078 — Delegated ACL users can change global password policy without a dedicated permission.** The review's new global password-restriction form and save handler do not test a feature-specific ACL capability. Historical source.
- **WEBMIN-0397 — Bulk user-to-group assignment bypasses delegated assignment limits.** The bulk joingroup action validates that each selected user is editable but accepts any submitted group, updates its membership, copies its ACLs, and merges its modules without applying the caller's gassign restriction. Historical source.
- **WEBMIN-0508 — Restricted ACL operator could self-assign to an unauthorized group.** In the reviewed historical state, acl/delete_users.cgi used delete permission and can_edit_user to authorize the bulk endpoint, exempted joingroup requests from the self-protection check, and then accepted the requested target group without enforcing the ACL module's gassign allowlist. Historical source.
- **WEBMIN-0625 — Delegated ACL operator could assign editable users to unauthorized groups.** The bulk Add To Group endpoint accepted a directly submitted group name after validating only that each target user was editable. It did not apply the independently configured gassign group allowlist before removing the target from its old group, adding it to the requested group, copying group ACL files, and saving inherited modules. Historical source.
- **WEBMIN-2451 — Delegated ACL administration can fall back to classic DES password hashes.** In the reviewed password-hash implementation, the mode-zero branch calls useradmin's system-aware hash routine only when foreign_available('useradmin') succeeds. That helper combines installation/support checks with the current Webmin user's module ACL. Historical source.

ACL · Core · Runtime (10)

- **WEBMIN-0180 — Mandatory PAM startup can fail open.** The service can continue running when PAM is configured as mandatory but cannot be enabled, allowing a weaker authentication path to remain active. Historical source.

- **WEBMIN-0181 — Symlink branch can bypass a filesystem containment predicate.** The symlinked-directory branch can return a truthy resolved path rather than evaluating whether the requested file remains within the permitted directory. Historical source.
- **WEBMIN-0630 — Forwarded client identity was not re-evaluated against source-IP policy.** When trusted-proxy mode replaces the direct peer with a forwarded client address, access decisions must be evaluated against that effective identity. Historical source.
- **WEBMIN-0632 — RPC request deserialization occurred before authorization.** rpc.cgi parsed a recursive, object-capable serialized request before checking whether the authenticated user had RPC permission. That ordering exposed parser and class-loading behavior to callers who should have been rejected, creating pre-authorization denial-of-service and gadget risk. Historical source.
- **WEBMIN-0707 — Absent users-file configuration caused authentication to fail open.** The entire request authentication pipeline was conditional on a configured users file. If the setting was absent because of a broken or incomplete Miniserv configuration, privileged requests could proceed without any user authentication instead of failing closed. Historical source.
- **WEBMIN-0746 — Encoded-colon script URIs bypass untrusted-HTML filtering.** The untrusted-HTML sanitizer recognizes literal javascript: and vbscript: schemes but not HTML entity encodings of the colon. A browser decodes those entities after filtering, permitting active content in the administrator origin. Historical source.
- **WEBMIN-0747 — RPC ACL branch inversion admits explicitly restricted users.** The shared RPC admin predicate maps the documented not-allowed ACL value to success and the allowed value to failure. Both classic and fast RPC endpoints trust this predicate, allowing an authenticated user explicitly denied RPC to cross the RPC authorization boundary. Historical source.
- **WEBMIN-0895 — Anonymous recovery rate limit is bypassed by an undefined guard variable.** The anonymous password-recovery loop initializes its configured attempt limit as $pfailures but tests the unrelated undefined $failures variable in the disable guard. Perl treats that value as false, so next executes for every IP, user, and email key. Historical source.
- **WEBMIN-0934 — Password-only Basic authentication bypassed account 2FA policy.** The HTTP Basic/RPC authentication path accepted a valid username and password without consulting the Webmin account's configured two-factor provider. Credentials for a 2FA-protected account could therefore authenticate through this alternate password-only path, defeating the account's multi-factor expectation. Historical source.
- **WEBMIN-0936 — RPC-only accounts were not confined to RPC entry points.** The global ACL value representing an RPC-only account was honored by RPC authorization but ordinary module initialization did not reject that account early enough. Such credentials could reach normal interactive Webmin modules, expanding their usable surface beyond the administrator-selected RPC-only scope, subject to the account's remaining ACLs. Historical source.

ACL · Core · Cross-cutting (1)

- **WEBMIN-0244 — IPv6 per-user deny restrictions can fail open after ACL serialization.** Per-user ACL records use colons as field delimiters, while IPv6 addresses contain literal colons. The historical diff demonstrates the required substitution at that serialization boundary. Historical source.

ACL · Core · Integration (4)

- **WEBMIN-0948 — Untrusted proxy identity headers could bypass authentication or IP ACLs.** Forwarded client IP and certificate headers were trusted under proxy mode without binding them to a configured direct peer. Directly reachable attackers could spoof IP identity or a certificate subject used for access decisions. Historical source.
- **WEBMIN-0949 — Pooled keep-alive requests could inherit another user's ACL identity.** Persistent per-request globals survived on a keep-alive backend connection. With reverse-proxy connection pooling, a later client's request could inherit baseauthuser or related authentication state and be authorized as the earlier client. Historical source.
- **WEBMIN-0952 — Restricted operators could change group membership for out-of-scope users.** The User and Groups save path accepted member lists without applying per-user edit ACLs. A

delegated operator able to edit a group could add or remove users outside their allowed account scope and change Unix privileges. Historical source.

- **WEBMIN-0953 — File Manager administrator shortcut overrode explicit safe and strict ACLs.** File Manager's write check returned success for administrator-named Webmin users before evaluating module safe or strict mode. Explicitly restricted privileged accounts could therefore write files their delegated ACL intended to deny. Historical source.

ACL · Module · Ajaxterm (1)

- **WEBMIN-0294 — Ajaxterm proxy skips module ACL and exposes localhost services.** ajaxterm/proxy.cgi deliberately omits init_config(), which is the Webmin initialization routine that enforces whether the current HTTP user may access the module. The CGI still accepts a numeric port and path from PATH_INFO, forwards the caller's method and body to that port on localhost, and returns the response. Historical source.

ACL · Module · Apache (4)

- **WEBMIN-0661 — Apache directive newline injection can execute a root-context piped command.** The Apache module's generic parse_opt, parse_choice, parse_handler, and parse_filters helpers accepted carriage returns, line feeds, and NUL bytes from request values. parse_inputs then passed the returned strings to save_directive. Historical source.
- **WEBMIN-0688 — Apache option saves bypassed the separate Apply ACL.** In the reviewed historical snapshot, both Apache global-option and virtual-server option save endpoints invoke restart_apache immediately after saving while checking only edit-oriented permissions. The module defines Apply as a distinct ACL, and restart.cgi checks that ACL before invoking the same service-control function. Historical source.
- **WEBMIN-0956 — Apache file parameters were not constrained to authorized targets.** Multiple Apache administration endpoints accepted a request file parameter and reached root-privileged read or write operations without first proving that the file was the configured MIME target or within the caller's allowed Apache directory. Historical source.
- **WEBMIN-2471 — Nested directive line accounting could rewrite adjacent security controls.** Apache directive objects carried physical line offsets used by generic save and delete helpers. Counting only immediate nested members, and later treating comments as independent dummy entries, could make those offsets disagree with serialized files.

ACL · Module · At (2)

- **WEBMIN-0708 — Scheduled job body can execute immediately with Webmin privileges.** In the reviewed Linux create_atjob state, the function separates the submitted command body (cmd) from the constructed at invocation (atcmd), including the command_as_user wrapper for a selected non-Webmin user. It then gives cmd itself to open_execute_command. Historical source.
- **WEBMIN-0709 — atd lifecycle operations lack action-level authorization.** The reviewed atd lifecycle handlers perform host-wide operations after loading the Scheduled Commands module but without a distinct action-level authorization decision. bootup.cgi enables or disables atd at boot, and start.cgi and stop.cgi mutate the service's runtime state. Historical source.

ACL · Module · Bind8 (1)

- **WEBMIN-0018 — Batch creation lacked directory authorization for local files.** The review zone creation workflow accepted a request-supplied local pathname and read it with Webmin's privileges without applying the module's directory authorization helper. Historical source.

ACL · Module · Certmgr (1)

- **WEBMIN-0070 — Manual configuration page ACL can be bypassed.** The module models manual configuration editing as a separately delegated page, but manual.cgi and save_manual.cgi do not enforce the corresponding access flag. A user with some Certificate Manager access can directly reach these endpoints even when manual editing is disabled and can read or replace the configured OpenSSL file. Historical source.

ACL · Module · Change User (1)

- **WEBMIN-0693 — Authenticated language preference permits global Webmin configuration injection.** In the reviewed historical source, an authenticated user allowed to change their language can submit a percent-encoded newline in the lang parameter. ReadParse decodes it, change-user stores it without validation, acl::modify_user places it in the global gconfig map, and the historical write_file serializes the embedded newline unchanged. Historical source.

ACL · Module · Cluster Passwd (2)

- **WEBMIN-0272 — UnixWare lock marker bypasses cluster old-password verification.** The cluster password-change handler added useradmin::encrypt_password as an old-password verifier. On UnixWare, useradmin deliberately sets no_encrypt_password because the native passwd program handles encryption, causing encrypt_password to return its input verbatim. Historical source.
- **WEBMIN-0325 — Cluster password validation could consult unrelated local user state.** The reviewed cluster password-update paths called useradmin::check_password_restrictions without the user-context argument even though cluster target information was already available. In the helper, an absent context activates a lookup in the Webmin host's local user list; the passwd_mindays comparison occurs only if a same-named local record is found. Historical source.

ACL · Module · Cluster Webmin (1)

- **WEBMIN-2411 — Cluster group ACL edits used the user ACL namespace.** The historical cluster group-ACL editor used the user-oriented get_module_acl path for both principal types and saved every result as $who.acl, even though group ACLs occupy the separate .gacl namespace.

ACL · Module · Cron (3)

- **WEBMIN-0568 — Restricted Cron-module users can reach the global manual file editor.** In the reviewed historical manual-editor code, edit_manual.cgi and save_manual.cgi authorize access with $access{'acl_users'} == 0. The cron ACL form and can_edit_user logic represent user scope with the mode and users fields, leaving acl_users undefined; Perl's numeric comparison treats that missing value as zero. Historical source.
- **WEBMIN-0757 — Direct Cron allow/deny save bypassed the module's policy-control ACL.** In the historical state evidenced by the reviewed diff, cron/save_allow.cgi accepted parsed request data and proceeded to lock, unlink, and rewrite the configured system cron allow/deny files without checking the caller's existing module-level allow permission. Historical source.
- **WEBMIN-0918 — Cron clone form could disclose jobs outside the caller's Unix-user ACL.** On the pre-change side of the reviewed diff, edit_cron.cgi accepted a caller-controlled clone index, selected from the complete cron-job array, and copied the selected job into a rendered creation form without first applying can_edit_user to the source job's Unix owner. Historical source.

ACL · Module · DHCP (7)

- **WEBMIN-0050 — Scoped DHCP editor can inject additional server directives.** The newly exposed dhcp-server-identifier field used the generic string serializer, which added quote delimiters but did not escape quote or directive syntax in the request value. A user constrained to editing Client Options could therefore add unrelated dhcpd directives to root-owned configuration. Historical source.
- **WEBMIN-0051 — Nameless-group ACL collision bypasses per-object authorization.** Per-object ACL keys were derived from a group name or domain-name option. Groups with neither value all mapped to ACLgrp_, so a grant for one nameless group authorized access to other colliding groups. Historical source.
- **WEBMIN-0407 — DHCP raw configuration editor bypasses the noconfig ACL.** In the reviewed DHCPd raw-configuration path, edit_text.cgi and save_text.cgi both test access{'noconfig'} without first calling get_module_acl. The intended denial therefore operates on an uninitialized ACL value. save_text.cgi then accepts multipart request text and replaces the configured dhcpd.conf through Webmin's privileged file helpers. Historical source.
- **WEBMIN-0563 — Hidden DHCP objects leaked through detailed list rendering.** Before commit f8e042beacb86b7c9873793c4f977471e2b86a09, the DHCP index filtered denied objects out of link,

title, and checkbox arrays but passed the unfiltered subnet/host collections to detailed table renderers. Historical source.

- **WEBMIN-0831 — DHCP quoted-hash parser differential can bypass per-object configuration ACLs.** The DHCP tokenizer's earlier substitution order applied hash-comment expressions before converting escaped quotes. For a valid quoted value such as an option containing " followed by whitespace and #, the escaped quote was counted as a closing quote and the remainder of the line was discarded. Historical source.
- **WEBMIN-1024 — Raw DHCP configuration read and write bypassed noconfig authorization.** The raw DHCP configuration display and save handlers checked the noconfig flag without first loading the requesting user's module ACL. Because the display path reads the complete configured file and the save path replaces it, direct endpoint access could bypass the intended restriction for an authenticated DHCP-module user assigned noconfig=1. Historical source.
- **WEBMIN-1025 — TSIG key disclosure and modification bypassed global authorization.** The TSIG key display and save handlers lacked an active server-side check of the DHCP module's global permission. A user with module access but global=0 could directly reach a view that renders shared secret values or submit a replacement set of key directives. Historical source.

ACL · Module · Dovecot (1)

- **WEBMIN-2447 — Saving Network Options could drop an existing TLS-required setting.** The pre-change Network Options page could read an active modern Dovecot `ssl` value of `required` but offered only `yes`, `no`, and default radio choices. Webmin's radio helper checks only an exact option-value match, leaving no SSL input selected for `required`.

ACL · Module · Fail2Ban (2)

- **WEBMIN-2425 — Sequential directive removal could delete an adjacent Fail2Ban setting.** The line-oriented configuration writer maintained both a start offset (line) and end offset (eline) for each directive. In the reviewed deletion sequence, it reduced line for later members and then retested that already-mutated line before reducing eline.
- **WEBMIN-2456 — Fail2ban enforcement can desynchronize after a FirewallD transition.** A FirewallD restart or start can rebuild firewall state without Fail2ban's dynamic ban rules even while Fail2ban remains running and believes those bans are active. In a host using both services, an administrator-triggered FirewallD transition can therefore create a bounded enforcement gap for already-banned addresses until Fail2ban re-establishes its rules. Historical source.

ACL · Module · Fdisk (1)

- **WEBMIN-0892 — Partition Manager hdparm request fields reach a shell command unescaped.** In the reviewed historical source, apply_hdparm.cgi accepts a set of hdparm option values from the request, concatenates each non-empty value into one command string without validation or shell escaping, appends the selected drive, and passes the result to backquote_logged. The drive is subject to can_edit_disk, but the option fields are not. Historical source.

ACL · Module · File (5)

- **WEBMIN-0262 — Chroot path expansion could bypass a denied-root ACL in file display and archive requests.** In the pre-change side of the reviewed file/show.cgi hunk, the request path was expanded with unmake_chroot before can_access compared it with module allowed and denied roots. Historical source.
- **WEBMIN-0318 — Missing path authorization before HTML editor read.** The reviewed HTML editor read the caller-selected file before applying the File Manager's allowed-root and denied-root predicate. Identity switching and an optional chroot can limit some deployments, but they do not enforce the delegated root policy in configurations where the module identity retains broader filesystem read permission. Historical source.
- **WEBMIN-0405 — File search executes after an allowed-root authorization denial.** The reviewed search.cgi state tests can_access for the requested directory but only prints an error when autho-

rization fails. It then launches find, returns path metadata, and can open matching files to test a supplied content pattern. Historical source.

- **WEBMIN-0406 — File-manager root ACLs can be bypassed with dot-dot path components.** The parent-side file-lib.pl source reviewed for the manifest change compares raw slash-separated request path components to allowed and denied root prefixes. A path can begin with an allowed root and then contain dot-dot components that the filesystem resolves outside that root. Historical source.
- **WEBMIN-0475 — Legacy File Manager relative paths could bypass configured root restrictions.** In the legacy File Manager code reviewed by this review, can_access formerly simplified caller-controlled input before comparing it with allowed and denied roots, but path-taking CGI handlers subsequently used the original input in filesystem operations. Historical source.

ACL · Module · Filemin (9)

- **WEBMIN-0525 — Unchecked action filenames bypass Filemin allowed-path restrictions.** The durable Filemin import validates the requested working directory against allowed_paths, but its action endpoints then append independent file, name, or multipart filename values without normalizing and rechecking the resulting target. The gap covers read, create, edit, save, upload, rename, delete, permission, and extraction entry points. Historical source.
- **WEBMIN-0558 — File operations lacked uniform allowed-directory confinement.** Multiple historical File Manager operations converted request names into privileged filesystem paths without uniform canonicalization and allowed-directory checks. Historical source.
- **WEBMIN-0647 — Inconsistent allowed-path canonicalization could bypass Filemin path restrictions.** Filemin simplified the requested working directory but, before the reviewed canonicalization hunk, compared it with raw allowed-path ACL strings. Historical source.
- **WEBMIN-0715 — Root-mapped File Manager accounts bypassed configured path restrictions.** The reviewed path setup treated any root Unix execution identity as implicitly authorized for the entire filesystem, replacing explicit configured allowed paths with /. A restricted Webmin File Manager account mapped to root therefore inherited root-wide Filemin scope rather than its configured directory list. Historical source.
- **WEBMIN-0807 — File downloads can follow an allowed-path symlink to out-of-scope data.** The reviewed download flow demonstrates that the prior source authorized a simplified lexical path and only then opened it. Because the open follows symbolic links, a link located inside an allowed directory could select a readable target outside that directory, bypassing the configured File Manager path boundary and disclosing the target contents. Historical source.
- **WEBMIN-0848 — Unchecked archive operands bypass File Manager allowed paths.** In the reviewed compression snapshot, selected names are passed to tar relative to the allowed current directory without rejecting .. components, allowing readable files outside that tree to be packaged for download. Historical source.
- **WEBMIN-0991 — Allowed-path identity misbinding in File Manager ACL expansion.** At the reviewed boundary, get_allowed_paths could expand *HOME*/ROOT using the connected Webmin identity rather than the ACL-selected work-as Unix identity and could consult the unpopulated global @allowed_paths for root-token and base selection. Historical source.
- **WEBMIN-1033 — Manual ACL arguments bypassed Filemin allowed-path enforcement.** The reviewed ACL endpoint accepted free-form manual text, split it on whitespace, shell-escaped each token, and appended the tokens to the setfacl command independently. Shell escaping protected the shell parser but not setfacl's own option and operand parser. Historical source.
- **WEBMIN-1034 — Search results crossed configured Filemin path boundaries.** The reviewed search endpoint ran find from the accepted current directory and immediately processed every returned pathname with stat and MIME classification for display. Filemin permits a common ancestor of configured allowed directories as a navigation location, so the search scope could include disallowed siblings. Historical source.

ACL · Module · Firewall (3)

- **WEBMIN-2403 — Virtual-interface selections can yield ineffective firewall scoping.** The firewall rule editor previously allowed virtual interface aliases to be chosen even though such interface matches

do not work for this rule path. A displayed but ineffective interface scope can cause an administrator to believe that a rule protects or restricts alias traffic when the kernel rule is rejected or never matches as intended.

- **WEBMIN-2434 — Restrictive IPv6 firewall presets contained an untyped all-ICMPv6 allow rule.** The restrictive IPv6 preset generator added an ACCEPT rule for protocol icmpv6 without an icmpv6-type constraint, despite describing the policy as permitting safe ICMP types. The broad rule widened the generated external-interface policy beyond the specific echo-request allowance used by the more permissive preset.
- **WEBMIN-2438 — Non-operative Fail2ban merge path can leave dynamic bans absent after firewall apply.** In the reviewed historical commit state, apply_configuration snapshots the live rules before applying Webmin's managed ruleset and calls merge_fail2ban_rules afterward in non-direct mode.

ACL · Module · Fsdump (1)

- **WEBMIN-0126 — Delegated backup directory name permits root command injection.** Linux and Solaris backup command builders place configured directory names inside literal single quotes and then execute the assembled command. Directory existence and ACL checks do not reject a quote within an otherwise permitted directory name. Historical source.

ACL · Module · Htaccess Htpasswd (2)

- **WEBMIN-1030 — Credential-file operations could escape delegated allowed directories.** The protected-directory save flow accepted direct form paths and, in additional paths, credential filenames obtained from existing AuthUserFile/AuthGroupFile directives without consistently applying can_access_dir. A Webmin or Usermin account delegated this module for selected directories could therefore direct credential-file operations beyond that scope. Historical source.
- **WEBMIN-1031 — Privileged protected-target access preceded directory authorization.** save_dir.cgi derived the .htaccess target and performed lock_file and get_htaccess_config before enforcing the module's allowed-directory decision. In a root-running Webmin request, locking can create an adjacent .lock file and capture target contents for action logging, while the Apache helper opens and parses the target. Historical source.

ACL · Module · Init (1)

- **WEBMIN-0739 — Systemd non-root boot helper omitted the User= boundary.** The historical systemd implementation of enable_at_boot_as_user placed the requested account in a top-level option, but enable_at_boot forwarded only the nested options object to create_systemd_service. Historical source.

ACL · Module · Itsecur Firewall (2)

- **WEBMIN-0288 — Delegated user management can modify arbitrary Webmin accounts.** The user-management UI presents only non-root accounts assigned to this module and only modules available to the current user. The save handler does not enforce either boundary: it selects any global Webmin account named by the request, including root, and accepts any submitted module list before invoking global ACL create, modify, or delete operations. Historical source.
- **WEBMIN-0438 — Separator editor can overwrite or delete the first firewall rule.** The separator form emits empty new, idx, and insert values because it reads a literal input key for each hidden field. The state-changing handler interprets the empty edit index as rule position zero. Historical source.

ACL · Module · LDAP Server (1)

- **WEBMIN-2414 — LDAP ACL editing can discard explicit rule order.** The reviewed historical ACL edit transformation parsed an olcAccess rule and serialized it again without carrying the explicit numeric {N} order prefix. An authorized edit could therefore rewrite a rule without the metadata OpenLDAP uses to represent access-rule precedence.

ACL · Module · Lpadmin (1)

- **WEBMIN-0139 — AIX printer deletion can damage another queue configuration.** In the parent version of the reviewed change, AIX delete_printer expects a printer name but dereferences that scalar as a hash when matching qconfig entries. For an ordinary name without a matching symbolic hash value, no entry is selected; undefined line offsets in the later splice are coerced to zero and one, removing the first qconfig line. Historical source.

ACL · Module · Mailboxes (1)

- **WEBMIN-2431 — Untrusted markup outside BODY was retained by the HTML mail helper.** safe_html temporarily retained markup located before an opening BODY element and after a closing BODY element in HTML email. Historical source.

ACL · Module · MySQL (3)

- **WEBMIN-0213 — MySQL user SSL type is interpolated into an UPDATE statement.** The SSL account-control change placed the submitted ssl_type value directly inside an UPDATE of MySQL's user table. A forged request bypasses the form's fixed option set and can change the statement syntax under the configured MySQL permission-management credential. Historical source.
- **WEBMIN-0789 — Database chooser could grant privileges to unintended databases.** The database chooser submitted literal database names directly into mysql.db privilege rows. In that table, percent and underscore have wildcard meaning, so selecting a concrete database with either character could authorize a database user for additional matching databases. Historical source.
- **WEBMIN-0881 — MySQL service-control CGI endpoints lacked direct module ACL enforcement.** The independently requestable start, stop, and restart CGI handlers proceeded from module initialization to privileged service helpers without checking the module's stop permission. An authenticated module user who lacked that action-level permission could invoke the endpoints directly and change database service state, including causing downtime. Historical source.

ACL · Module · Net (3)

- **WEBMIN-0441 — Authenticated VLAN field injection executes privileged shell commands.** At the audited historical state, save_bifc.cgi accepts VLAN physical and tag fields without validation and activate_interface interpolates them, along with derived interface names, into strings executed by privileged shell helpers. Historical source.
- **WEBMIN-0468 — Active-route interface command injection permits root command execution.** In the reviewed historical state, create_route.cgi copies the local-interface request field into route.iface without validation. Linux create_route then appends that value to either a route or ip route shell command and executes the assembled string as Webmin's privileged process. Historical source.
- **WEBMIN-0552 — Delegated interface edit can delete configuration for out-of-scope interfaces.** In the historical Debian network backend immediately preceding the reviewed change, get_interface_defs did not treat a top-level source line as the end of an iface stanza, and modify_interface_def likewise did not terminate its replacement region at source. Historical source.

ACL · Module · Nginx (1)

- **WEBMIN-1002 — Direct Nginx restart bypassed the service-control ACL.** The Nginx restart CGI reached configuration testing and the privileged apply command without enforcing access.stop. Hiding service controls in the UI was insufficient because an authenticated module user denied that ACL could request the endpoint directly. Historical source.

ACL · Module · PHP Configuration (3)

- **WEBMIN-0531 — Virtualmin PHP discovery bypassed PHP Configuration global-file ACL.** The manifest change added Virtualmin-discovered global php.ini files to list_php_configs() without applying the global-file ACL that protects the normal system-wide list. can_php_config() uses membership in this list as its authorization decision, so an authenticated PHP Configuration module user configured with global=0 and anyfile=0 could still select and modify those global files. Historical source.

- **WEBMIN-0675 — Delegated PHP Configuration user could edit Virtualmin global configs despite global=0.** At the historical boundary before the reviewed guard, list_php_configs appended Virtualmin global PHP configuration paths whenever Virtualmin was installed, even when the delegated user's global ACL was disabled. Those paths then passed can_php_config and could be changed with the configured Unix identity. Historical source.
- **WEBMIN-0777 — Delegated PHP configuration editor could load an arbitrary native extension.** In the reviewed extension-selection snapshots, edit_dirs.cgi presented a server-generated set of shared-object names, but save_dirs.cgi trusted the client-supplied ext and NUL-separated exts parameters. It wrote those values as extension_dir and one or more extension directives after checking only whether the user could edit the target PHP configuration. Historical source.

ACL · Module · Postfix (1)

- **WEBMIN-2429 — Sender restriction edits can leave the compiled lookup policy stale.** The sender-restriction editor's regeneration helper queries a Postfix parameter named regenerate_sender_restrictions_table rather than smtpd_sender_restrictions. Consequently, saving an indexed sender restriction map can update its text source and reload Postfix without rebuilding the database actually used for lookups. Historical source.

ACL · Module · PostgreSQL (3)

- **WEBMIN-0623 — PostgreSQL host-authentication policy write bypasses module ACL.** The manual editor's display page rejects PostgreSQL-module users whose users ACL is disabled, but the newly added save_manual.cgi endpoint lacks that check and directly writes submitted data to the configured pg_hba.conf file. Historical source.
- **WEBMIN-0726 — Incomplete PostgreSQL identifier quoting enables SQL injection through crafted object names.** In the reviewed historical snapshot, the database-deletion endpoint uses quote_table for a request-supplied database identifier and catalog-derived table identifiers. quote_table splits on dots and surrounds each segment with double quotes but does not escape an embedded double quote. Historical source.
- **WEBMIN-0984 — PostgreSQL host-access editors omitted function-level authorization.** The structured allowed-host editor/save flow and manual HBA replacement flow lacked their own PostgreSQL users-permission checks. Direct requests from an authenticated Webmin principal with module access but without that delegated permission could reach pg_hba.conf display or mutation and, on the structured path, restart PostgreSQL. Historical source.

ACL · Module · Proc (2)

- **WEBMIN-0106 — macOS process lookup evaluates an unquoted PID before authorization.** In the reviewed macOS state, Process module endpoints pass the requested PID through process_info to list_processes before process-level authorization. list_processes interpolates the PID into two Perl pipe-open command strings without validation or quoting. Historical source.
- **WEBMIN-0862 — Legacy trace streaming endpoint omitted process authorization and trusted its state-file identifier.** The reviewed legacy streaming endpoint accepted a process PID without the process lookup and can_edit_process gate used by the ordinary trace page. It also formed a writable state-file path directly from the caller's id. Historical source.

ACL · Module · Proftpd (1)

- **WEBMIN-2473 — ProFTPD TLS policy controls were assigned to the wrong Webmin configuration context.** Within the assigned history, the ProFTPD TLS descriptor table classified TLSEngine, TLSRequired, TLSRSACertificateFile, TLSRSACertificateKeyFile, and TLSCACertificateFile for Webmin's global editor collection. Webmin renders and saves that collection against <Global> members, while its root collection targets the top-level server configuration.

ACL · Module · Project Meta (1)

- **WEBMIN-0905 — Package publication bypasses SSH server authentication.** The reworked CI publisher configures both ssh and scp with StrictHostKeyChecking=no and UserKnownHosts-

File=/dev/null. The same unauthenticated channel uploads package artifacts and sends a remote signing command containing the configured GPG passphrase. Historical source.

ACL · Module · Samba (1)

- **WEBMIN-1026 — Missing endpoint authorization allowed restricted users to access full Samba configuration editing.** The Samba module's manual-edit permission controlled whether the UI exposed the full-configuration editor, but the direct editor and save handlers did not themselves enforce that permission in the reviewed patch context. Historical source.

ACL · Module · Sendmail (2)

- **WEBMIN-0989 — Sendmail auxiliary-file handlers used a prefix-only directory authorization check.** The historical Sendmail address-file, filter-file, and autoreply-file edit and save handlers authorized a caller-provided file path by comparing only its leading bytes with the configured apath. This accepted sibling names that merely shared the prefix (for example, /etc/mailcap when apath was /etc/mail). Historical source.
- **WEBMIN-1027 — Missing Sendmail subfeature authorization on bulk deletion.** The Sendmail module has independent ACL controls for domain mappings and mailertable entries, but the two bulk-delete CGI paths shown by the reviewed change did not enforce the corresponding flag before selecting and deleting records. Historical source.

ACL · Module · Software (1)

- **WEBMIN-1019 — Authenticated unrestricted file read in Software Packages viewer.** The reviewed viewer snapshot uses PATH_INFO as a filesystem path and opens it without verifying that it belongs to any selected software package. An authenticated module user can therefore read any local path accessible to the privileged Webmin process. Historical source.

ACL · Module · Spam (2)

- **WEBMIN-0028 — Configured AWL user/group restrictions were ineffective.** The historical can_edit_awl authorization function received a requested username but attempted to obtain awl_users and awl_groups restrictions by dereferencing that scalar instead of reading the module ACL hash. Historical source.
- **WEBMIN-0251 — Usermin Auto-Whitelist page allowlist bypass.** The historical Usermin branch of can_use_page first loaded the administrator-controlled avail_icons list and then unconditionally inserted awl. Because Auto-Whitelist view and mutation handlers authorize with can_use_check("awl"), direct requests remained allowed even when the administrator had disabled that page. Historical source.

ACL · Module · Squid (2)

- **WEBMIN-0791 — Direct Squid apply request bypasses the module restart permission.** At the reviewed 166df0cb92a9e45902fc000d6a9fb12dcc26077f snapshot, squid/restart.cgi parses a request and directly invokes apply_configuration() without checking $access{'restart'}. The module defines that distinct permission in acl_security.pl and consults it when rendering the apply control. Historical source.
- **WEBMIN-1028 — Restricted Squid-module users could bypass the configuration-apply ACL.** The reviewed Squid restart.cgi handler could reach apply_configuration() without checking the module's dedicated restart permission. Although the UI conditionally displayed the apply button, a restricted authenticated module user could call the CGI directly. Historical source.

ACL · Module · SSH Server (2)

- **WEBMIN-0282 — SSHD global security settings can remain trapped inside Match scope.** The historical `save_directive` change attempted to put newly created global SSH directives before conditional Match configuration, but it iterated through all Match entries and retained the last one. Historical source.
- **WEBMIN-0321 — Saving SSH authentication settings could silently re-enable public-key login.** The historical OpenSSH authentication form read PubkeyAuthentication into $rsa but rendered

the uninitialized $dsa variable. The yes/no helper therefore displayed the setting as enabled even when sshd_config explicitly contained PubkeyAuthentication no. Historical source.

ACL · Module · Syslog (2)

- **WEBMIN-1021 — Pre-authorization metadata rewrite crossed restricted Syslog destination scope.** On the earlier side of the reviewed change, save_log.cgi opened a request-selected local-log destination and unconditionally passed it to set_ownership_permissions before can_edit_log evaluated the account's allowed destination scope. An authenticated delegated Syslog editor could therefore target an existing file outside the permitted log paths. Historical source.
- **WEBMIN-1022 — Pre-validation file creation crossed restricted Syslog destination scope.** On the earlier side of the reviewed change, local destination parsing opened the request-selected path in append mode before message-type validation and before can_edit_log evaluated the account's allowed path scope. Historical source.

ACL · Module · Updown (1)

- **WEBMIN-2460 — Whole-file dangerous-document inspection permits authenticated memory pressure.** The assigned dangerous-document path reads an entire HTML, XML, or PDF file into memory and creates a filtered comparison copy before responding. Request memory therefore scales with selected file size rather than the normal bounded send buffer.

ACL · Module · Useradmin (3)

- **WEBMIN-1014 — Bulk home-directory deletion bypassed the dedicated permission.** The user and group management module exposed two bulk user-deletion paths that could invoke home-directory deletion without rechecking the dedicated delhome ACL at the destructive sink. Historical source.
- **WEBMIN-1015 — Local batch imports crossed global file-access boundaries.** Server-side user and group review import accepted a caller-selected local path using only the module's review directory boundary and read the file directly under Webmin's process identity. Historical source.
- **WEBMIN-1016 — Group membership changes crossed the editable-user ACL.** The group editor did not authoritatively apply the editable-user ACL to membership deltas. A delegated operator allowed to edit a group could submit additions or removals for users outside that operator's configured user scope, potentially changing access or privilege through group membership. Historical source.

ACL · Module · Usermin (1)

- **WEBMIN-1017 — Usermin restart endpoint did not enforce the stop ACL.** The reviewed commit diff shows that usermin/restart.cgi previously loaded the Usermin module library and invoked restart_usermin_miniserv() without checking the configurable stop ACL. An authenticated account assigned the Usermin Configuration module but denied stop permission could therefore request restart.cgi and restart the Usermin service. Historical source.

ACL · Module · Vnc (1)

- **WEBMIN-0082 — Application mode does not enforce VNC authentication or a local-only listener.** When the optional X application mode is enabled, the CGI launches Xvnc :1 without specifying a password or a local-only binding and starts the configured program on that display. The source leaves password creation as a TODO. Historical source.

ACL · Module · Webminlog (1)

- **WEBMIN-0921 — Safe-mode user could reconfigure global audit-email notifications.** In the historical state evidenced by the reviewed review patch, a Webmin account restricted to safe modules could inherit notify=1 from webminlog/defaultacl because the module's safeacl did not explicitly override that field. Historical source.

ACL · Module · Xterm (2)

- **WEBMIN-0792 — Direct WebSocket listener exposed an unauthenticated privileged shell.** The reviewed initial Terminal server accepted a numeric TCP port and registered PTY input/output callbacks without a WebSocket handshake authentication check. The bundled server's numeric-listen path bound by LocalPort without LocalAddr, while the module chose port 555 by default and selected root in its default ACL. Historical source.
- **WEBMIN-0799 — Shell WebSocket backend listened beyond loopback before authenticating its single client.** The reviewed shellserver passed a bare numeric port to the bundled Net::WebSocket::Server, whose IO::Socket::INET default did not set LocalAddr. The server assigned the first connection to its sole-client slot and disabled the startup alarm before the handshake key callback ran. Historical source.

Cross-site scripting and HTML injection (150)

XSS · Core · ACL (2)

- **WEBMIN-0719 — ACL editor header rendered an unescaped user/group identifier.** acl/edit_acl.cgi placed the request-selected user or group identifier directly inside an HTML <tt> fragment used by ui_print_header. Without contextual encoding, crafted markup in that identifier could execute in an authenticated ACL editor's Webmin origin. Historical source.
- **WEBMIN-2474 — ACL audit-log rendering accepted unencoded structured values.** The ACL audit-log formatter historically inserted multiple structured values into HTML without sink-local encoding. If markup-bearing data reached an object, rename, group, two-factor, bulk-delete, or clone field, a privileged viewer of rendered Webmin logs could receive active markup.

XSS · Core · Runtime (9)

- **WEBMIN-0011 — Core browser endpoints rendered untrusted values without context encoding.** PAM login, search, file/date/module, upload, user, and group chooser paths placed request, conversation, or account-derived values into HTML attributes or JavaScript strings without the encoding required by those contexts. Historical source.
- **WEBMIN-0185 — Chooser filenames can inject script into an inline handler.** A quote-bearing local filename can escape the chooser's inline JavaScript attribute and execute script in an authenticated user's Webmin origin. Historical source.
- **WEBMIN-0414 — Reflected script injection through the Webmin search title.** The Webmin search endpoint placed the request-controlled title in the page header without HTML-context encoding. A crafted authenticated request could therefore inject markup or script into the Webmin origin. Historical source.
- **WEBMIN-0415 — Script injection in chooser and upload rendering contexts.** Chooser and upload pages embedded filenames, usernames, group data, and progress text into JavaScript strings and document.write output using encoders that did not match those contexts. The reviewed series introduces hexadecimal JavaScript escaping and separate HTML escaping for generated table cells. Historical source.
- **WEBMIN-0631 — Forwarded-address header allowed HTML injection.** Forwarded client-address headers are attacker-controlled unless strictly validated at the trust boundary. The reviewed path permitted arbitrary header text to flow into host/error presentation, enabling markup injection when an administrator encountered or reviewed the resulting output. Historical source.
- **WEBMIN-0634 — Unescaped filenames reached HTML error rendering.** File and directory names can contain HTML metacharacters and may be influenced by lower-privileged users or request data. Rendering those names directly in file-open and replacement error pages creates a markup injection path in a privileged administrator session. Historical source.
- **WEBMIN-0704 — Unescaped progress URLs could inject markup into authenticated responses.** Download progress rendering passed a redirect- or cache-derived URL directly into HTML-bearing translated text. A URL containing markup-significant content could therefore render attacker-controlled HTML or script in the authenticated Webmin origin. Historical source.
- **WEBMIN-0795 — Core UI renderers lacked context-specific output encoding.** Core UI paths emitted select labels and account chooser values into HTML text and inline JavaScript without consistent

context-specific encoding. Attacker-influenced module labels or local identity metadata could therefore become markup or script in a privileged user's Webmin origin. Historical source.

- **WEBMIN-0937 — Script-embedded redirect URL could break out of the script element.** The UI redirect helper embedded a caller-provided URL inside a script element. JavaScript quote escaping did not stop the HTML parser from recognizing a less-than sequence such as a closing script tag, allowing attacker-controlled markup to terminate the element and execute in the Webmin origin when a controllable URL reached the helper. Historical source.

XSS · Core · Cross-cutting (2)

- **WEBMIN-0570 — Historically unescaped request values reached privileged HTML pages.** Several administrative pages rendered invalid usernames, file/package search values, or invalid referer hostnames inside HTML without output encoding. A crafted request shown to an authenticated Webmin user could therefore execute markup in the Webmin origin. Historical source.
- **WEBMIN-0571 — Historically unescaped SQL result cells could execute stored markup.** The MySQL and PostgreSQL query pages passed database result cells directly to the HTML column renderer. If a database contained attacker-controlled markup, a privileged database administrator displaying that row could execute it in the Webmin origin. Historical source.

XSS · Core · Integration (5)

- **WEBMIN-0585 — Historical Majordomo list content enables stored Webmin-origin XSS.** Majordomo index and detail views emitted stored list configuration, information, introductory text, message headers/footers, and members without HTML escaping. A delegated list operator could store markup that executes when another Webmin user views those pages, crossing from scoped list management into the privileged origin. Historical source.
- **WEBMIN-0587 — Historical Certificate Manager metadata enables stored Webmin-origin XSS.** Certificate Manager parsed subject, issuer, and subjectAltName strings from certificate text and placed them directly into UI grid values. Crafted certificate material could therefore inject markup or script when viewed by an authenticated Webmin user. Historical source.
- **WEBMIN-0589 — Historical Filemin edit view permits reflected Webmin-origin XSS.** Filemin inserted the request-controlled file name into a single-quoted inline JavaScript call without JavaScript-context escaping. A crafted permitted file value could terminate the string and execute script when an authenticated Filemin user opened the URL. Historical source.
- **WEBMIN-0817 — Upload/Download inline viewer relies on bypassable active-content regexes.** When Upload and Download display mode is enabled, browser-active files can be returned from the authenticated Webmin origin. The reviewed sequence first emits original bytes inline, then attempts to protect HTML/XML/PDF by comparing the output of regular-expression filtering and forcing attachment only when bytes change. Historical source.
- **WEBMIN-0947 — Pre-authentication login-banner reflected XSS.** When a configured login banner substituted LOGINURL, the unauthenticated page parameter reached HTML without filtering. A crafted login URL could execute script in the Webmin origin before authentication. Historical source.

XSS · Module · Apache (3)

- **WEBMIN-0004 — Stored HTML injection through Apache Require identities.** The legacy Require editor derives a user/group list from Apache configuration words and historically placed it directly in a quoted input value. A quote-bearing identity could therefore escape the attribute and inject browser content when an authorized user opens the editor. Historical source.
- **WEBMIN-0957 — Configuration-derived Apache values enabled stored XSS.** The Apache define and global MIME views inserted configuration-derived strings into HTML labels, table cells, or query strings without the required context encoding. An actor who can influence those persistent Apache values could supply markup that executes when a privileged Webmin user opens the affected page. Historical source.
- **WEBMIN-2418 — Apache config-test diagnostic HTML injection in the reviewed historical path.** In the reviewed manual-save path, Apache configuration-test diagnostics entered a preformatted Webmin error response without contextual HTML encoding on the removed side of the delta. Because a

user with manual-edit permission can influence the directives reported by Apache, diagnostic text could contain active markup. Historical source.

XSS · Module · Bacula Backup (2)

- **WEBMIN-0487 — BareOS console resource markup yields stored XSS in historical Bacula forms.** The alternate BareOS console-output branches accept quoted job, client, storage, pool, address, and related resource fields with greedy captures and no output encoding. Those values populate selection labels in Bacula forms. Historical source.
- **WEBMIN-0728 — Bacula catalog labels can execute markup in the restore chooser.** The audited chooser flow copies Bacula catalog directory basenames and filenames into JSON node text. Its jsTree configuration does not enable text-only rendering, and the bundled library's false default assigns node text through innerHTML. Historical source.

XSS · Module · Bind8 (3)

- **WEBMIN-0188 — Reflected XSS in BIND zone-search output.** The zone-search request value reached both a Perl regular expression and a rendered HTML fragment without literalization or HTML encoding. A crafted search could therefore alter expression semantics and inject active markup into an authenticated Webmin response. Historical source.
- **WEBMIN-0189 — Stored attribute injection through edit-record identifiers.** The edit-record link used record_id, a string containing DNS name, type, and values, directly inside a quoted href. Malicious zone data could escape that attribute context when an operator viewed editable records. Historical source.
- **WEBMIN-0539 — Unescaped record values enabled stored XSS in audit-log rendering.** The bind8 log parser embedded persisted record names, zone objects, and new record values directly inside HTML. A restricted DNS user could store markup in a record value and have it rendered when a more privileged user viewed the action log, allowing script execution in that viewer's Webmin session. Historical source.

XSS · Module · Blue Theme (1)

- **WEBMIN-0003 — Webmin search reflects an unescaped no-results term into authenticated HTML.** The review snapshot's shared Webmin search endpoint reads and URL-decodes the search parameter, trims it, and in the no-results branch places it inside a tt element through the text substitution helper. Neither the sink nor text performs HTML encoding. Historical source.

XSS · Module · Burner (1)

- **WEBMIN-0335 — Norwegian Burner validation errors reflect raw source paths into HTML.** The Norwegian Burner catalog contains several validation messages that interpolate a source-path placeholder. In the historical Burner save_profile.cgi flow, an authenticated module user controls source_N, and failed path or directory-ACL validation passes that value to text() and then error(). Historical source.

XSS · Module · Certmgr (2)

- **WEBMIN-0071 — Unencoded certificate data and request fields allow cross-site scripting.** Certificate subject and issuer components parsed from OpenSSL output and multiple direct request fields are concatenated into HTML without contextual output encoding. Malicious certificate identity data or crafted request values can therefore inject active markup into an authenticated Webmin page. Historical source.
- **WEBMIN-0413 — Certificate metadata enables stored HTML injection.** show_cert_info extracts certificate and CSR subject/issuer values from OpenSSL output and concatenates them into an HTML table without calling html_escape. UI table helpers treat these values as markup. Historical source.

XSS · Module · Cfengine (1)

- **WEBMIN-0463 — Stored XSS through managed-server metadata in CFEngine host details.** The historical Servers save path persisted an administrator-supplied server description without HTML

validation. The touched CFEngine host-detail renderer placed that description, or the host when no description existed, directly in the display-text argument to ui_link. Historical source.

XSS · Module · Cluster Cron (2)

- **WEBMIN-0638 — Reflected markup injection in the legacy new-job id field.** The removed-side new-job form shown by the reviewed diff emitted the request parameter id directly inside a single-quoted hidden-input value. Because the new=1 path did not constrain id through a job lookup, a crafted value could close the attribute and inject active markup into an authenticated Cluster Cron page. Historical source.
- **WEBMIN-2468 — Historical cluster-job result rendering lacked HTML output encoding.** In the historical state immediately before the first reviewed change, the cluster-job callback embedded its remote command output or remote error string directly into the authenticated execution-results page. A malicious or compromised managed endpoint could therefore return HTML or script-capable markup when an administrator triggered the job.

XSS · Module · Cluster Shell (4)

- **WEBMIN-0130 — Linked-server command output can execute script in the authenticated Webmin origin.** Before output encoding was added, Cluster Shell placed the raw result returned by remote_eval directly inside the Webmin results page. A compromised or malicious linked server, or a remote command whose output contains attacker-controlled markup, could return HTML containing script. Historical source.
- **WEBMIN-0270 — Delegated linked-server metadata enables stored XSS in Cluster Shell.** The historical Cluster Shell index emitted stored linked-server host and description fields directly inside an HTML option element. A Servers-module delegate who could create or modify a linked server could persist markup and cause it to execute for a Cluster Shell user viewing the command form. Historical source.
- **WEBMIN-0331 — Cluster Shell renders persisted server descriptions as active HTML.** In the audited historical snapshot, Cluster Shell concatenates each credentialed server's persisted description into a selector display label and passes it to the shared ui_select helper. That helper emits option display text without HTML escaping. Historical source.
- **WEBMIN-0689 — Stored XSS through Cluster Shell command history.** Cluster Shell accepted an authenticated command value, appended it verbatim to a module-global history file, and later supplied each stored command as a scalar option to the historical ui_select helper. Historical source.

XSS · Module · Cluster Shutdown (1)

- **WEBMIN-0466 — Persistent server metadata creates stored cross-site scripting.** The module places persistent registered-server host and description fields directly into HTML table cells, and it later joins selected hostnames into confirmation markup without encoding. The server editor allows arbitrary description content and permits non-whitespace markup-like hostname content in same-user mode. Historical source.

XSS · Module · Cluster Software (3)

- **WEBMIN-0620 — Reflected XSS through the package-view search hidden field.** In the historical package-details implementation, the search request parameter was copied directly into double-quoted hidden value attributes for list and uninstall forms. A quote-bearing value could terminate the attribute and add active markup in the authenticated Webmin origin. Historical source.
- **WEBMIN-0842 — Reflected HTML injection through Cluster Software search input.** The reviewed search response placed the request-controlled search string directly into localized HTML messages on both match and no-match branches and also copied it unencoded into a link attribute. An attacker able to induce an authenticated Cluster Software user to open a crafted search request could execute markup in the Webmin origin. Historical source.
- **WEBMIN-0843 — Package description could inject markup into installation-completion details.** The installation-completion details embedded package description metadata directly inside a

preformatted HTML block. A package or managed package system able to return crafted description text could cause markup execution when an authenticated user completed the installation. Historical source.

XSS · Module · Cluster Usermin (1)

- **WEBMIN-0845 — Historical reflected markup injection in Cluster Usermin install errors.** The reviewed data flow shows request parameters local and url being interpolated by text() into messages that download_error() prints inside HTML. Without output encoding at the two affected error branches, crafted markup in either value can be reflected into the module response. Historical source.

XSS · Module · Cluster Webmin (1)

- **WEBMIN-0846 — Reflected XSS in Cluster Webmin module-install error fields.** In the audited patch preimages, cluster-webmin/install.cgi used the request-derived local path and malformed URL directly as substitutions in localized installation errors. download_error printed its message argument into the HTML response without sink-level encoding. Historical source.

XSS · Module · Cpan (5)

- **WEBMIN-0104 — Authenticated reflected XSS in the historical CPAN module chooser.** The historical side of the reviewed cpan.cgi hardening diff interpolated the request-controlled search value directly inside an HTML <tt> fragment. It also split the request-controlled selection value without validation and reflected it into page text and single-quoted href attributes/fragments. Historical source.
- **WEBMIN-0274 — Authenticated reflected XSS in CPAN missing-module display.** The historical missingok flow builds each missing-module display item as <tt> plus the request-derived CPAN token plus </tt>, then emits the joined list into the options table. A crafted nonexistent module token can therefore become active markup in an authenticated Webmin response. Historical source.
- **WEBMIN-0381 — RPM fallback reflects an arbitrary CPAN token as active Webmin-origin HTML.** In the reviewed historical source, source=3 splits the cpan request parameter into arbitrary tokens and the RPM fallback treats every unmatched token as a package capability. The status response places the raw token in a translated HTML message; text() substitutes it without encoding. Historical source.
- **WEBMIN-0697 — Cross-site scripting through unencoded CPAN archive diagnostics.** In the reviewed historical CPAN handler, combined tar listing and diagnostic output was embedded directly inside an HTML error response after archive validation returned a non-zero status. A malformed archive could emit attacker-controlled member text before failing, causing markup to execute in the authenticated Webmin origin. Historical source.
- **WEBMIN-0875 — Reflected HTML injection in CPAN source-mode error handling.** The CPAN download handler accepted a request-controlled source selector, compared it numerically, and placed unsupported-mode values directly into a generic error message rendered as HTML. Values beginning with an unsupported numeric mode could retain trailing markup, allowing reflected HTML or script interpretation in an authenticated Webmin response. Historical source.

XSS · Module · Cron (5)

- **WEBMIN-0064 — Stored XSS in Cron job-list environment and comment cells.** The audited job-list builder concatenates persistent cron environment names, values, and job comments directly into HTML. A delegated user able to write one of those fields can persist active markup that executes when an administrator or another more privileged Webmin user views the combined job listing, enabling Webmin-session privilege escalation. Historical source.
- **WEBMIN-0433 — Stored XSS from user crontab environment values reaches privileged Webmin viewers.** The historical Cron Jobs index reads environment assignments from per-user crontabs and renders the stored name and value directly as ui_link display text. ui_link interpolates display text without escaping, even though neighboring command rows explicitly call html_escape. Historical source.
- **WEBMIN-0480 — Delegated Cron user can plant stored Webmin-origin XSS through run-parts filenames.** In the assigned historical snapshot, is_run_parts removes the trailing shell comment and recognizes a run-parts directory, expand_run_parts returns every non-dot entry from that directory, and edit_cron.cgi concatenates those paths into HTML without escaping. Historical source.

- **WEBMIN-0658 — Run-parts filename can inject markup into cron job editor.** The reviewed run-parts expansion accepted every non-dot filename, and the cron job editor joined the returned paths directly into HTML. A markup-bearing filename in a recognized run-parts directory could therefore become active markup in an authorized viewer's Webmin page. Historical source.
- **WEBMIN-0853 — Reflected cross-site scripting in Cron username error messages.** The historical Cron save endpoint inserted the request-derived user parameter into three localized error messages without HTML neutralization. The text helper performs placeholder replacement without encoding, and the CGI error path emits the substituted message into HTML. Historical source.

XSS · Module · Custom (1)

- **WEBMIN-0611 — Stored command descriptions reached unencoded privileged HTML contexts.** A persisted Custom Commands description was rendered without HTML encoding in two privileged UI contexts: as an option label in the ACL editor after the first manifest change, and as run-page header subtext immediately before the second manifest change. Historical source.

XSS · Module · Dhcp Dns (1)

- **WEBMIN-0121 — Delegated DHCP metadata executes as stored XSS in DHCP/DNS Hosts.** The DHCP/DNS Hosts index renders a host's parent description without HTML escaping. For group parents this is the DHCP group comment; for shared-network parents it is the shared-network name. Historical source.

XSS · Module · Disk Usage (1)

- **WEBMIN-0276 — Disk Usage tree stores and renders attacker-controlled directory names as active HTML.** The historical Disk Usage module recursively records filesystem directory paths and later interpolates each path directly into an anchor name attribute and a tt element. Historical source.

XSS · Module · Documentation (4)

- **WEBMIN-0249 — Authenticated reflected XSS in Documentation no-results search text.** The historical no-results branch rendered the request-controlled Documentation search text inside a <tt> element without HTML encoding. A crafted link viewed by an authenticated Webmin user with module access could execute script in the Webmin origin, becoming root-equivalent when the victim has privileged Webmin rights. Historical source.
- **WEBMIN-0513 — Documentation search results heading reflected unescaped terms.** In the parent state of the reviewed change, man/search.cgi split the request search term and embedded each raw token in the results heading. The default UI emitted that subtext as HTML. Historical source.
- **WEBMIN-0561 — Unescaped Manual Pages parameters permit reflected XSS.** The historical Manual Pages handler rendered untrusted page and section parameters into HTML without output encoding. A crafted request viewed by an authenticated module user could execute script in the Webmin origin and act with that victim's Webmin privileges. Historical source.
- **WEBMIN-0913 — Man opts parameter allowed authenticated reflected XSS in generated links.** The reviewed historical Man module had two request-to-link dataflows. The older converter branch inserted opts directly into rewritten anchor markup, and manual search results concatenated opts into a URL passed to a UI helper that emitted an unencoded single-quoted href. Historical source.

XSS · Module · Exports Nfs4 (1)

- **WEBMIN-0136 — Authenticated stored XSS in the NFSv4 export editor.** In the introduced NFSv4 export module, hostname-mode input is accepted whenever it contains an asterisk and is persisted verbatim to the exports file. The parser restores that token, and edit_export.cgi places it directly inside a single-quoted value attribute. Historical source.

XSS · Module · Fail2Ban (1)

- **WEBMIN-0712 — Fail2ban custom protocol stored XSS in default-jail editor.** At the reviewed historical commit, edit_jaildef.cgi began asking ui_select to add an existing protocol value that was

absent from its fixed choices. save_jaildef.cgi accepted a forged protocol value without validation and save_directive preserved it in Fail2ban configuration. Historical source.

XSS · Module · Fetchmail (1)

- **WEBMIN-0171 — Stored XSS through a Fetchmail password in the historical poll editor.** The historical parent poll editor inserted the parsed Fetchmail password directly into a single-quoted HTML value attribute. A local account that can write a .fetchmailrc can store an apostrophe-bearing password that adds an event-handler attribute when a Webmin operator opens that poll entry. Historical source.

XSS · Module · File (2)

- **WEBMIN-0025 — Request and command output reached File Manager HTML without encoding.** Historical File Manager error pages interpolated request-derived filenames and archive command output into HTML without encoding, and the module index placed the request-derived open value in a quoted applet attribute without encoding. Crafted authenticated requests could inject markup or script into the Webmin origin. Historical source.
- **WEBMIN-0830 — Legacy Xinha editor executes active content from opened HTML files.** The legacy File module's HTML editor reads an access-authorized file into a textarea and starts Xinha with the default configuration. Xinha creates an iframe without a sandbox and writes inwardHtml(textarea.value) into that same-origin document. Historical source.

XSS · Module · Filemin (4)

- **WEBMIN-0606 — Invalid chown identity values reached HTML error output without escaping.** The reviewed chown.cgi patch shows invalid owner and group request values moving from direct interpolation inside bold HTML fragments to html_escape-wrapped output. Because print_errors deliberately renders markup-bearing fragments, the prior direct interpolation allowed active markup in the authenticated Webmin origin. Historical source.
- **WEBMIN-0608 — Crafted filenames entered the editor's inline JavaScript string.** The reviewed editor code inserts the requested filename directly into a single-quoted inline JavaScript change(...) call. A filename containing a quote and JavaScript syntax can break out of that string when an authenticated user opens an existing crafted file within an allowed path. Historical source.
- **WEBMIN-0648 — Filemin request path was reflected into executable browser contexts.** Filemin assigned the request's path parameter directly to `$path` and interpolated that raw value into single-quoted hidden fields, href attributes, and JavaScript navigation attributes across current and legacy templates. Historical source.
- **WEBMIN-0684 — Filesystem-controlled breadcrumb path was interpolated into an HTML attribute.** The reviewed historical breadcrumb renderer placed the accumulated filesystem path directly into a single-quoted href while escaping only the visible component. Historical source.

XSS · Module · Filter (4)

- **WEBMIN-0067 — Stored filter fields can execute markup in the Usermin filter list.** Filter condition header names, condition values, raw regular expressions, and forwarding actions were embedded into list-page markup without HTML encoding. A malicious value persisted through a quick-filter/imported/provisioned rule is interpreted as active content when the filter list is opened. Historical source.
- **WEBMIN-0514 — Stored XSS through an unescaped forwarding action in the Usermin filter list.** In the parent snapshot of commit 1760032, filter/filter-lib.pl::describe_action inserted a parsed forwarding action directly into a <tt> element. filter/index.cgi rendered that description in the authenticated Usermin filter table, while save.cgi allowed non-whitespace forwarding text without HTML-safe validation. Historical source.
- **WEBMIN-0840 — Stored folder-name HTML injection in reviewed filter UI sinks.** At the pre-change sides of the reviewed folder-label hunks, mailbox folder display names were supplied directly to select-option or link/table HTML renderers. Because those helpers treat caller text as HTML, a crafted

stored folder name could become active markup when the relevant authenticated filter page was viewed. Historical source.

- **WEBMIN-0841 — Stored filter-action HTML injection in the reviewed action column.** The reviewed filter action path loaded request-derived stored action content and could interpolate a file action into action-column HTML without encoding. Markup-bearing action content could therefore become active HTML or script when the authenticated filter listing rendered that path. Historical source.

XSS · Module · Firewall (1)

- **WEBMIN-0081 — Authenticated reflected XSS in historical firewall index parameters.** In the historical parent snapshot reviewed for the assigned change, firewall/index.cgi embedded the reset request value directly in a single-quoted hidden-field attribute and embedded the table request value directly in hidden fields and quoted links. Historical source.

XSS · Module · Fsdump (1)

- **WEBMIN-0127 — Legacy OS-specific backup forms allow stored cross-site scripting.** The reviewed FreeBSD, Solaris, and IRIX forms interpolate persisted backup destination and label fields directly into HTML attributes. Their parsers store fields that may contain attribute-breaking characters. Historical source.

XSS · Module · Gray Theme (2)

- **WEBMIN-0360 — Authenticated reflected XSS through Gray Theme frame target.** In the reviewed Gray Theme frame controller, ReadParse accepts page, the controller concatenates that value into goto without HTML-attribute encoding, and the frameset writes goto into a double-quoted src attribute. A quote-bearing page value can end src and add an event handler. Historical source.
- **WEBMIN-0507 — Reflected script injection through gray theme page parameter.** At the reviewed commit, ReadParse decodes the page request parameter, index.cgi appends it to the right-frame URL, and the frameset emits that URL directly inside a double-quoted src attribute. A crafted value containing a quote can break out of the attribute and inject markup or an event handler when a victim opens the URL. Historical source.

XSS · Module · Htaccess Htpasswd (1)

- **WEBMIN-0613 — Stored HTML injection in protected-directory index labels.** The protected-directory index formerly inserted directory paths, htpasswd usernames, and group names directly into HTML link text. Because those persisted values can contain markup delimiters, a crafted entry could execute attacker-controlled HTML or script when an authorized operator viewed the index. Historical source.

XSS · Module · Init (2)

- **WEBMIN-0597 — Stored HTML injection through init module command configuration.** The examined historical reboot and shutdown confirmation pages inserted the configured reboot_command or shutdown_command directly into an HTML fragment. Historical source.
- **WEBMIN-2464 — Stored HTML injection through service metadata in action lists.** The action-list renderer passed OS-derived service names and descriptions directly into HTML rows for Win32, FreeBSD RC, Upstart, and systemd modes. An actor able to control service metadata could place active markup there; when a user authorized for Bootup and Shutdown opened the list, it could execute in that user's Webmin origin. Historical source.

XSS · Module · Inittab (1)

- **WEBMIN-0431 — Stored init-process ID permits privileged-context XSS.** In the historical code, the Init Configuration list concatenated a parsed init-process ID directly into the href argument of ui_link. The helper placed that string verbatim inside a single-quoted href attribute, while the module's save path accepted any non-empty ID and persisted it to inittab. Historical source.

XSS · Module · Itsecur Firewall (1)

- **WEBMIN-0287 — Persistent cross-site scripting in firewall configuration views.** Group names, rule and separator descriptions, service names, and time names accept HTML-significant characters and are stored in module configuration. List and edit pages then concatenate those values into HTML text, link targets, option elements, and quoted attributes without context-appropriate encoding. Historical source.

XSS · Module · LDAP Users and Groups (2)

- **WEBMIN-0228 — LDAP group descriptions rendered as unescaped option HTML.** In the reviewed historical snapshot, LDAP group descriptions are appended to the visible label of selected secondary groups. The contemporaneous generic select renderer quote-escapes option values but places display labels directly into option HTML. Historical source.
- **WEBMIN-0292 — Stored HTML injection in LDAP custom-field edit controls.** The reviewed UI migration diff removes HTML assembled with LDAP custom-field values inserted directly into a single-quoted input value and directly into a textarea body. Those contexts lacked output encoding, so a crafted directory value could inject active content when a privileged Webmin operator opened a user or group edit form. Historical source.

XSS · Module · Link (1)

- **WEBMIN-0247 — Authenticated stored XSS in Custom Link ACL URL field.** The historical Custom Link ACL editor rendered its persisted per-user or per-group link URL directly inside a single-quoted input value. A delegated ACL administrator able to save this field could store a quote-breaking payload; when a more-privileged administrator later opened the Custom Link ACL form, the payload could execute in the Webmin origin. Historical source.

XSS · Module · Mailboxes (10)

- **WEBMIN-0033 — Successful spam-report response embeds an unescaped folder value in JavaScript.** Historical commit 38ada7346c94b0b03a4155b84ef05f4477d67374 added a post-report script redirect that interpolates the raw folder request parameter inside a single-quoted JavaScript string. Historical source.
- **WEBMIN-0151 — MIME attachment filename rendered without HTML encoding.** The attachment table and slideshow used decoded MIME filename text directly in HTML. A sender able to deliver a crafted attachment name could inject markup when a privileged mailbox reader viewed or forwarded the message. Historical source.
- **WEBMIN-0152 — Mail sender text could become active HTML in the reply composer.** Sender name and address text derived from the From header was concatenated into HTML reply content before the reviewed change applied html_escape at each HTML-context use. When the HTML editor materialized the textarea value, crafted sender markup could become active content in the authenticated Webmin origin. Historical source.
- **WEBMIN-0399 — Stored HTML injection through an unparsed mailbox Date header.** A message Date header that did not parse was returned verbatim by simplify_date and inserted into the mailbox list without HTML encoding. Because a remote sender controls that header, opening the list could execute stored markup in the authenticated administrative origin. Historical source.
- **WEBMIN-0562 — Stored HTML injection through MIME attachment filenames.** attachments_table decoded the MIME attachment filename and used that sender-controlled string as an HTML anchor label without output encoding. A crafted filename could therefore inject persistent markup when an authenticated recipient viewed the message's attachments. Historical source.
- **WEBMIN-0774 — Stored cross-site scripting through an unescaped mail-folder selector label.** The reviewed historical folder selector built each displayed option label from a mailbox folder name without HTML output encoding. Those names include filenames recursively discovered under per-user mail storage, while the contemporaneous shared select renderer encoded the option value but emitted its label as raw HTML. Historical source.
- **WEBMIN-0824 — Raw HTML mail was rendered in an unsandboxed same-origin iframe.** The reviewed HTML rendering change passed raw attacker-controlled email HTML into a same-origin

srcdoc iframe whose generated markup had no sandbox attribute at that snapshot. Script in a malicious message could therefore reach the authenticated parent Webmin page. Historical source.

- **WEBMIN-0825 — Email head content could influence the containing Webmin page.** Email head markup was extracted, passed through JavaScript filtering, and inserted into the containing mail page. JavaScript filtering does not make arbitrary style, metadata, and remote-resource elements safe for parent-page inclusion; malicious mail could alter the authenticated UI or trigger tracking requests. Historical source.
- **WEBMIN-0827 — Unencoded folder parameter enabled pagination-link XSS.** The folder request parameter was concatenated directly into pagination href query strings. Without URL encoding, crafted delimiter and markup characters could escape the intended value context and produce reflected script or markup in the authenticated mail list. Historical source.
- **WEBMIN-0997 — Active SVG attachment content under the Webmin origin.** The reviewed individual-attachment response accepted attachment-controlled SVG content and could serve it as an active document from the authenticated Webmin origin. A malicious message sender could place script-capable SVG content in a mailbox and rely on an authorized reader opening the attachment. Historical source.

XSS · Module · Mailcap (1)

- **WEBMIN-0432 — Mailcap index renders stored program and comment markup in the Webmin origin.** In the historical manifest snapshot, mailcap/index.cgi sends each entry's program and comment (or description fallback) directly to the HTML table renderer. save.cgi permits a module-authorized user to persist active-HTML characters in those fields, and mailcap-lib.pl also reads equivalent values from the configured file. Historical source.

XSS · Module · Majordomo (2)

- **WEBMIN-0582 — List-scoped content is rendered as stored script in privileged Webmin views.** The overview and redesigned detail page concatenate persisted list info, description, intro, message directives, reply-to data, owner/approval values, and member lines into HTML without contextual escaping. A list-scoped module principal can persist these values through ACL-gated list save operations. Historical source.
- **WEBMIN-0599 — Delegated Majordomo global-option values enabled stored Webmin-origin XSS.** The historical global-options save endpoint allowed a Webmin user holding the Majordomo module's delegated global ACL to persist whoami and whoami_owner strings containing quote and markup delimiters because validation required only a non-whitespace value. Historical source.

XSS · Module · Mon (1)

- **WEBMIN-0175 — Legacy MON status CGI renders daemon-controlled status data as active HTML.** The preimage deleted by reviewed commit 3c44db87eaf8639bc4da1a24df190ea594765af3 retrieves group, service, description, summary, alert, and status data from a MON daemon and inserts those strings directly into Webmin-origin HTML and attributes. Historical source.

XSS · Module · Mount (1)

- **WEBMIN-0037 — Historical reflected XSS in the NFS export chooser.** Before the assigned XSS-hardening change, nfs_export.cgi rendered the request-supplied server value directly in the popup heading and failure message. A crafted URL viewed by an authenticated Webmin user could execute attacker markup in the Webmin origin. Historical source.

XSS · Module · MySQL (1)

- **WEBMIN-0046 — Historical MySQL UI output paths lacked HTML encoding.** Reviewed MySQL UI deltas show two authenticated output paths that embedded attacker-influenced text in HTML without context-appropriate encoding: the database search status banner and the SQL execution result heading. Either path could execute markup in the Webmin origin when a privileged operator loads the crafted content. Historical source.

XSS · Module · Old Blue Theme (1)

- **WEBMIN-0485 — Old Blue theme page parameter enables authenticated reflected XSS.** In the historical Old Blue Framed Theme, index.cgi appends the request-controlled page parameter to a frame target and prints it unencoded inside a double-quoted src attribute. An authenticated victim following a crafted URL can therefore receive attacker-chosen attributes or script-bearing markup in the Webmin origin. Historical source.

XSS · Module · Package Updates (2)

- **WEBMIN-0200 — Crafted package selector causes reflected XSS before validation.** update.cgi splits the request-controlled u selector into package and system fields, then interpolates the package portion into a tt element without HTML encoding. This output occurs before package_install checks whether the package exists, so an invalid selector still reaches the response. Historical source.
- **WEBMIN-0802 — Unencoded package identifiers can become active progress-page markup.** Package identifiers originate in the update request and are placed into HTML progress messages in both multi-package and per-package paths. Without contextual encoding, a markup-bearing identifier is interpreted as active content before the package operation finishes, creating reflected script execution in the authenticated Webmin origin. Historical source.

XSS · Module · Pam (1)

- **WEBMIN-0888 — Stored HTML injection in the PAM services list.** In the parent snapshot reviewed for c607990c48f5e544e3928506fa4a91ae12b2997c, get_pam_config read a service-file description comment and pam/index.cgi passed it directly to an HTML table-cell builder. The reviewed change applies html_escape at this list-page sink, preventing a crafted description from being interpreted as markup there. Historical source.

XSS · Module · Passwd (2)

- **WEBMIN-0075 — Account real-name data is rendered as active HTML in the password-change form.** The reviewed password-change form reads the selected account through getpwnam and renders the login plus real-name/GECOS field as the raw value of ui_table_row. That historical helper interpolates its value argument directly into a table cell. Historical source.
- **WEBMIN-0364 — Stored XSS risk from unescaped account metadata in the password UI.** In the reviewed historical snapshot, passwd/edit_passwd.cgi placed the getpwnam login and GECOS/real-name values directly into an HTML table-cell value, while passwd/index.cgi escaped the visible username but interpolated the raw username into a double-quoted link target. Historical source.

XSS · Module · Postfix (1)

- **WEBMIN-0994 — Reflected XSS in Postfix mail-queue search results.** The historical Postfix mail-queue search response placed the request-controlled match value directly inside an HTML <tt> element. An attacker who induces an authenticated user with mailq module access to open a crafted search can execute active content in the Webmin origin. Historical source.

XSS · Module · PostgreSQL (2)

- **WEBMIN-0040 — PostgreSQL index filters allowed authenticated reflected XSS.** The reviewed historical PostgreSQL module index had two request-to-markup paths. The search value was placed directly in an HTML body context, while show values replaced the permission-filtered database-title list and were then used to build href values. Historical source.
- **WEBMIN-0448 — PostgreSQL grantee name can trigger stored XSS in the historical grant editor.** The grant editor consumes PostgreSQL ACL grantee names through extract_grants and supplies them to ui_select, including the add-if-missing path enabled by the reviewed change. In the historical snapshot, ui_select escapes submitted values but not visible option labels. Historical source.

XSS · Module · Proc (2)

- **WEBMIN-0565 — Historical authenticated reflected XSS in Process command output.** In the historical parent state reviewed through this manifest change, proc/run.cgi inserted the request-controlled

cmd value directly into a tt element on the authenticated foreground command-output page. Historical source.

- **WEBMIN-0660 — Stored XSS through an NSS-controlled Process ACL run-as username.** The historical Process module ACL form derives its configured run-as username from getpwuid and inserts it directly into a single-quoted input value. An NSS identity containing a quote plus event attributes can escape that value and execute script when an authorized Webmin ACL administrator views the form. Historical source.

XSS · Module · Proftpd (1)

- **WEBMIN-0252 — Denied FTP users editor allowed stored Webmin-origin XSS.** The historical denied-FTP-users save endpoint persisted module-user-controlled text in the configured ftpusers file, and the editor emitted that file directly inside literal textarea markup. A whitespace-free closing-textarea payload could therefore persist active markup and execute in the Webmin origin when another authenticated user opened the editor. Historical source.

XSS · Module · Quota (1)

- **WEBMIN-0110 — Predictable /tmp repquota input enables privileged stored XSS.** At the reviewed historical state, Linux user quota enumeration runs `cat /tmp/repquota.txt` as root. A local user can create that predictable file with a report record whose username is no-whitespace HTML. Historical source.

XSS · Module · Raid (1)

- **WEBMIN-0160 — Historical RAID action-log descriptions allow stored XSS.** The conversion/grow logging change records request-derived count text and the RAID log parser returns it inside HTML without escaping. In the reviewed historical action-log search and detail views, that description was rendered directly. Historical source.

XSS · Module · Servers (3)

- **WEBMIN-0089 — Delegated server editor can store script for privileged Servers-page viewers.** In the reviewed historical code, save_serv.cgi persisted request-controlled host, description, and group values, while index.cgi embedded those fields into generated HTML without output encoding. Historical source.
- **WEBMIN-0245 — Authenticated stored XSS through no-port registered-server hostname.** In the assigned historical change, a no-port server record is rendered by appending its stored host or realhost directly into the Servers table. save_serv.cgi accepts any non-whitespace hostname and stores it unchanged when port_def is used with mode 0. Historical source.
- **WEBMIN-0772 — Stored Servers Index metadata rendered as active HTML.** Servers Index host/realhost, description, group, and icon-title values were persisted from module edit operations and rendered into HTML text contexts without output encoding. A user granted server-record edit/add capability could store active markup that would be interpreted when another authorized user viewed the table or icon presentation. Historical source.

XSS · Module · Shell (2)

- **WEBMIN-0454 — Persisted shell ACL username could escape its HTML value attribute.** The reviewed historical parent form interpolates the persisted shell ACL username directly inside a single-quoted HTML value attribute. The same file stores the submitted username when the default option is not selected. Historical source.
- **WEBMIN-2437 — Stored command markup could alter Command Shell audit-log presentation.** The reviewed shell log-parser delta shows that recorded command text was placed directly inside an HTML tt element used to describe Command Shell run actions. Because the audit UI consumes module descriptions as HTML, an authenticated shell user could store markup in a command and have that structure rendered for a later log viewer.

XSS · Module · Software (2)

- **WEBMIN-0723 — Remote package-download error could execute script in the Webmin origin.** In the reviewed historical Software Packages URL-installation flow, a remote endpoint could place a crafted invalid HTTP status line or redirect Location value into the download error scalar. The error handler embedded that scalar directly in an HTML response. Historical source.
- **WEBMIN-0820 — Reflected HTML injection in Software Packages error and search responses.** Two Software Packages response paths placed parsed request text into HTML without context-appropriate encoding: the invalid package URL error and the no-match package search response. A crafted value rendered for an authenticated module user could be interpreted as markup or script in the Webmin origin. Historical source.

XSS · Module · Squid (1)

- **WEBMIN-0443 — Squid operational command output rendered as active HTML.** In the reviewed historical states, multiple Squid lifecycle and cache-initialization paths placed external command output inside preformatted Webmin responses without HTML escaping. Attacker-influenced output could therefore become active content in the authenticated administration origin. Historical source.

XSS · Module · Ssh (1)

- **WEBMIN-0085 — SSH Login endpoint configuration is inserted into HTML without escaping.** The SSH Login CGI renders the configured host and port directly into quoted applet parameter attributes. The generic module configuration handler accepts these values as free text. Historical source.

XSS · Module · Ssh2 (2)

- **WEBMIN-0086 — SSH2 configuration values are inserted into applet HTML without escaping.** A user allowed to edit SSH2 module configuration can enter non-numeric text in the optional port or font-size fields. The CGI inserts these values directly into applet parameter markup. Historical source.
- **WEBMIN-0220 — Delegated SSH2 configuration editing permits stored XSS.** The SSH2 module's generic configuration exposes foreground, background, terminal type, and later encoding as free-form text. Historical source.

XSS · Module · Status (1)

- **WEBMIN-0923 — Email-template descriptions could execute stored markup in the Webmin origin.** A status editor could save arbitrary nonblank template-description text, and the template list supplied that stored value directly as the HTML body of a link. Since `ui_link` does not encode its text argument, script-capable markup could execute in the Webmin origin for another status editor who viewed the list. Historical source.

XSS · Module · Syslog (1)

- **WEBMIN-0096 — Authenticated reflected XSS in the historical System Logs viewer header.** In the historical System Logs viewer, the explicit file query parameter could be assigned to $file after module-ACL checks and then concatenated directly into the HTML page header. A crafted value containing markup could therefore execute script in the authenticated Webmin origin when an authorized victim opened the link. Historical source.

XSS · Module · Time (2)

- **WEBMIN-0467 — Delegated timezone editing enables stored XSS in the historical Time module.** At the historical commit, an authenticated user granted only the Time module's timezone capability can directly submit an arbitrary zone value. Webmin, running as root, stores that value in the /etc/localtime symlink target. Historical source.
- **WEBMIN-2398 — Authenticated reflected markup through unescaped date-command diagnostics.** In the reviewed historical Time handler, date and hardware-clock utilities receive an argument assembled from request fields, with shell quoting applied, and their stderr is captured on failure. The

handler forwards that diagnostic to Webmin's legacy HTML error renderer without output encoding. Historical source.

XSS · Module · Tunnel (1)

- **WEBMIN-0775 — Tunnel malformed-path error allowed reflected XSS.** The pre-change Tunnel link.cgi embedded an invalid PATH_INFO value directly in error(). The historical Webmin error renderer inserted the original argument into an HTML h3 without output encoding, so a crafted invalid Tunnel URL could execute attacker-controlled markup in an authenticated victim's Webmin origin. Historical source.

XSS · Module · Updown (5)

- **WEBMIN-0207 — Crafted LHA member name executes in the authenticated upload result.** The historical LHA upload path parses filename text from lha command output and embeds that text in an HTML fragment without escaping. The upload result then emits the fragment in the authenticated Webmin page. Historical source.
- **WEBMIN-0614 — Authenticated HTML injection through an unescaped upload destination.** The pre-change upload handler placed the destination path directly between HTML <tt> tags in its saving-status response. That path came from the submitted destination and, for directory uploads, the basename of the multipart filename, so HTML-significant characters were not neutralized before rendering. Historical source.
- **WEBMIN-0821 — Inline active-document preview can execute in the Webmin origin.** In the assigned history, fetch.cgi could return a readable file inline using a guessed active MIME type when preview was enabled. The reviewed changes successively used type coercion, blacklist filtering, broader HTML/XML/PDF classification, and attachment fallback, which confirms that the response crosses an active-content trust boundary. Historical source.
- **WEBMIN-0856 — Reflected self-XSS in download error messages.** The before-change download handler inserted a submitted invalid URL, destination directory, and run-as user, as well as an immediate-download error string, into HTML-rendered messages without output encoding. The localized text helper only performs placeholder substitution, and the shared CGI error path renders message fragments as HTML. Historical source.
- **WEBMIN-0993 — Unsafe inline file preview allowed same-origin active-content execution.** The historical Up/Down preview response selected active MIME types such as HTML, XML, or PDF, ran the original bytes through a regular-expression JavaScript filter, and forced download only when the filtered bytes differed. It then returned the original bytes. Historical source.

XSS · Module · Useradmin (4)

- **WEBMIN-0055 — Chooser pages embedded unencoded values into JavaScript.** Historical user and group chooser pages placed request-derived names and account-derived display values inside JavaScript string literals without JavaScript-context encoding. A crafted chooser request or stored display value could break out of the string and execute script in an authenticated Webmin origin. Historical source.
- **WEBMIN-0235 — Account and group metadata was emitted unencoded into privileged Webmin pages.** The assigned history contains several account and group views that emitted real names, home paths, shells, login values, group descriptions, group names, and member labels without HTML encoding. An attacker able to influence such system identity metadata could cause script execution when a privileged Webmin user opened the affected list or form. Historical source.
- **WEBMIN-0457 — Useradmin chooser inline-script escaping permits script termination.** The historical user and group multi-select chooser base frames interpolate caller-selected tokens and resolved account metadata into a raw inline script. Their quote_javascript helper backslash-prefixes angle brackets but leaves the angle-bracket bytes visible to the HTML script-data parser. Historical source.
- **WEBMIN-0816 — Mass account-action status messages rendered local identifiers without HTML encoding.** The reviewed mass group-deletion and mass user-action paths inserted local account or group identifiers into translated HTML status messages without encoding at the changed sinks. Historical source.

XSS · Module · Usermin (4)

- **WEBMIN-0460 — Stored Webmin-origin XSS through a crafted Usermin IPv6 netmask suffix.** In the audited historical tree, save_ipkey.cgi accepts per-IP SSL-key tokens through check_ip6address. That validator removes everything after the slash into a netmask string and checks it only through Perl numeric comparisons, so a suffix beginning with an in-range number may retain arbitrary markup. Historical source.
- **WEBMIN-0569 — Delegated Usermin category editor can plant stored JavaScript in users' landing pages.** Commit da6b64277820e5851e6244e36bc25ddc2e3b8b06 made the built-in category editor's submitted field names line up with the save handler. The handler requires only the independently delegatable categories permission and writes the supplied label verbatim to webmin.catnames. Historical source.
- **WEBMIN-0837 — Stored XSS through unencoded Usermin metadata and log renderers.** Usermin stored module-description overrides and consumed module, theme, language, dependency, and log metadata in administrator-facing views. Before output encoding was applied in the reviewed sink set, those values reached HTML markup directly. Historical source.
- **WEBMIN-0838 — Stored XSS through a custom Usermin category label.** The Usermin category editor stored custom category descriptions verbatim in webmin.catnames. The module-assignment page read those values and supplied them as displayed option text to a selector implementation that did not encode the label. Historical source.

XSS · Module · Webminlog (4)

- **WEBMIN-0304 — Historical stored XSS from local process arguments in privileged action-log searches.** When full action descriptions are requested, the historical Webmin Actions Log search passes a process-kill event through proc/log_parser.pl, which surrounds the recorded process arguments with formatting tags but does not encode them. The search result then embeds that description directly as HTML. Historical source.
- **WEBMIN-0426 — Webminlog view footer reflected nested search state into an active HTML attribute.** In the parent state of reviewed commit 3b6234b511f97d18f4941c618ac594e73d8f7bda, view.cgi decoded the search query parameter with ReadParse and concatenated it into the destination passed to ui_print_footer. The footer routine inserted that destination into double-quoted href attributes without HTML-attribute escaping. Historical source.
- **WEBMIN-0559 — Stored action-description XSS in Webmin Actions Log search and detail views.** Historically, Webmin Actions Log rendered module-generated action descriptions as HTML in both search results and the detailed action page without active-content filtering. For a concrete delegated path, MySQL user management accepted any non-whitespace host value, stored it in the audit record, and its log parser embedded that host without HTML escaping. Historical source.
- **WEBMIN-0666 — Forwarded client address enabled stored XSS in action-log views.** In the reviewed historical snapshot, Webmin's action logger could persist a request-controlled forwarding-header value as the client-IP token when trust_real_ip was enabled. The action-log search and detail pages parsed that token and passed it directly to table helpers whose values are raw HTML. Historical source.

XSS · Module · Wuftpd (1)

- **WEBMIN-0353 — Norwegian WU-FTPd validation errors allow authenticated reflected XSS.** The Norwegian locale added by the reviewed commit defines net_eip with a raw $1 placeholder. Historical save_net.cgi forwards an invalid request-controlled aip_N value to this message, text() performs literal substitution, and error() writes the message into HTML. Historical source.

XSS · Module · Xterm (1)

- **WEBMIN-0793 — Invalid shell-user input was reflected without HTML escaping.** The reviewed invalid-account branch copied the request-derived user value into a translated error response without HTML output encoding. A crafted Terminal request could therefore execute markup or script in the authenticated Webmin origin when an authorized victim followed it. Historical source.

Command and data injection (145)

Injection · Core · Command-line utilities (1)

- **WEBMIN-0748 — Remote version response can inject terminal control sequences.** The historical detailed version-reporting path downloaded a key/value list from virtualmin.com using Webmin's HTTPS helper without passing certificate requirements. It accepted numeric-leading remote version strings and printed a qualifying value directly among terminal formatting sequences. Historical source.

Injection · Core · Runtime (4)

- **WEBMIN-0184 — Login redirect parameter permits HTTP response injection.** An unvalidated post-login page parameter is inserted into the Location header, allowing response splitting or attacker-directed response manipulation. Historical source.
- **WEBMIN-0420 — Dangerous function-shaped headers reached CGI environments.** HTTP request headers are transferred into CGI process environments. Function-shaped values could therefore reach a vulnerable shell runtime invoked by a CGI and trigger command interpretation before the intended script logic. Historical source.
- **WEBMIN-0705 — Compatibility password tool allowed shell interpretation of its arguments.** The compatibility changepass.pl rewrite constructed a single shell command containing the config directory, username, and new password. A metacharacter-bearing argument could be interpreted as shell syntax and cause unintended command execution with the invoking operator's privileges. Historical source.
- **WEBMIN-0896 — Password-recovery emails trust an attacker-controlled Host header.** The recovery-email URL helper prefers HTTP_HOST from the anonymous request and places it directly into the emailed origin. A forged Host can therefore place the bearer reset identifier in a link to an attacker-controlled server. Historical source.

Injection · Core · Cross-cutting (1)

- **WEBMIN-0673 — Authenticated command injection through the Usermin SSL key output path.** The Usermin SSL-key workflow accepts any non-empty newfile value and embeds it inside single quotes in shell command strings. A permitted Usermin user can break out of those quotes and execute arbitrary commands in the privileged process context. Historical source.

Injection · Core · Integration (5)

- **WEBMIN-0521 — Filemin metadata listing permits filename-driven command execution.** The reviewed historical File Manager listing builds lsattr and SELinux commands by surrounding raw full pathnames with literal double quotes and executing the resulting scalar with Perl backticks. A crafted directory-entry name can terminate the intended argument or invoke shell substitution. Historical source.
- **WEBMIN-0584 — Historical Majordomo list-name injection executes root shell commands.** The Majordomo list creator accepted every non-whitespace list name. The list index later inserted the list-derived members filename into a backticked cat/wc pipeline. Historical source.
- **WEBMIN-0586 — Historical Certificate Manager signing permits root command injection.** The signcsr handler gated access with its module ACL but validated only that signing parameters were present. It interpolated multiple request fields directly into a backticked OpenSSL command, so a principal delegated certificate-signing permission could append shell syntax and execute commands as root. Historical source.
- **WEBMIN-0588 — Historical Synology IPKG backend permits root command injection.** The IPKG backend received the package parameter from Software Packages CGI handlers. Although package_info computed a quotemeta value, it executed the raw input; installation and deletion helpers also interpolated raw package values. Historical source.
- **WEBMIN-0742 — Usermin certificate path reaches an unescaped shell redirection.** The reviewed Usermin certificate-generation CGI accepts a request-controlled newfile path, checks only that it is non-empty, and interpolates it inside single quotes in a shell output-redirection command. An embedded quote can escape that word and append shell commands under Webmin's management privileges. Historical source.

Injection · Core · Webmin (1)

- **WEBMIN-0193 — Parsed OpenPGP key names enter privileged shell commands without escaping.** The reviewed GnuPG helper reads user ID text from GnuPG output and reuses the parsed key name in shell-interpreted fingerprint, encryption-recipient, trust, and deletion commands. Wrapping the name in double quotes does not disable command substitutions. Historical source.

Injection · Module · Apache (1)

- **WEBMIN-0005 — Shell injection in htaccess directory discovery.** The htaccess discovery endpoint accepts a requested directory, checks that the literal path is a directory, and historically interpolated it directly into a string-form pipe open. A directory name containing shell metacharacters can pass the filesystem check and alter the shell command, yielding command execution in the Webmin process context. Historical source.

Injection · Module · At (2)

- **WEBMIN-0215 — macOS At working-directory injection executes outside su as root.** In the historical macOS At backend, create_job.cgi accepts any existing working directory and create_atjob interpolates it directly into a nested double-quoted su -c command. A delegated At user can reference an existing directory whose name contains a closing double quote and shell syntax. Historical source.
- **WEBMIN-1013 — Outer-shell command injection through At job working directory.** On the historical macOS and OpenServer At backends represented by the reviewed diff, create_job.cgi accepted a caller-selected working directory after only verifying that it existed. The backend interpolated that value into a double-quoted outer command of the form su user -c "cd DIR ; at ...". Historical source.

Injection · Module · Bacula Backup (1)

- **WEBMIN-0061 — Authenticated SQL injection in Bacula catalog file listing.** In the reviewed historical list.cgi, the job request parameter is copied into an SQL fragment and interpolated into four DBI queries without a numeric constraint or bound placeholder. Because the same statements bind volume and path values, the unbound job fragment is a distinct request-to-query trust-boundary failure. Historical source.

Injection · Module · Bind8 (1)

- **WEBMIN-0780 — Privileged shell command injection through zone-derived values.** The reviewed history contains privileged BIND operations that constructed shell command strings from zone-related domains, filenames, or temporary paths without consistently separating those values from shell syntax. Historical source.

Injection · Module · Certmgr (3)

- **WEBMIN-0068 — Authenticated shell command injection in Certificate Manager operations.** Certificate generation, CSR generation/signing, certificate inspection, overwrite previews, and PKCS#12 export construct shell command strings containing request-controlled values. Perl pipe-open, backtick, and system-string forms invoke a shell, so crafted values can add shell syntax and run commands with the Webmin process privileges. Historical source.
- **WEBMIN-0411 — Certmgr file parameters allow root command execution.** Certmgr handlers concatenate request-controlled certificate/key paths, output paths, day counts, and related values into Perl pipe opens or backtick command strings. In view.cgi, direct file parameters reach both OpenSSL command strings and a two-argument open used by downloads. Historical source.
- **WEBMIN-0578 — Certificate-manager request fields reach privileged shell commands.** In the reviewed historical snapshots, certmgr/signcsr.cgi builds an OpenSSL CA invocation with Perl backticks from form-controlled file and duration values, while certmgr/view.cgi builds an OpenSSL CRL invocation with pipe-open from a direct crlfile parameter. Neither sink uses an argument-vector execution API or shell-safe validation. Historical source.

Injection · Module · Cluster Shutdown (1)

- **WEBMIN-0465 — Stored registered hostname enables local shell command injection.** The module's parallel status checker constructs a backtick shell command containing a registered server's host value without quoting. The contemporaneous server editor permits any non-whitespace host in same-user mode, stores it, and sets the user field that makes the Cluster Shutdown module process the record. Historical source.

Injection · Module · Cluster Software (2)

- **WEBMIN-0986 — Remote Perl code injection through unescaped package paths.** The parallel installer embedded a request-controlled local package path in a remote stat Perl expression, and both parallel and serial installers embedded a package-derived remote temporary path in a remote unlink expression. These strings were passed to remote_eval, which evaluates Perl in the remote software-module session. Historical source.
- **WEBMIN-0987 — HTTP and FTP protocol injection through unvalidated download fields.** The install confirmation endpoints accepted host, port, HTTP path, and FTP path fields from the request and forwarded them to download helpers on managed hosts. The earlier URL parsing step did not bind those hidden fields to the later request. Historical source.

Injection · Module · Cpan (1)

- **WEBMIN-0696 — Shell command injection through an unescaped CPAN archive pathname.** In the reviewed historical CPAN handler, archive paths derived from a local request value or from supplied upload and URL basenames were interpolated into a shell-form open used to run gunzip and tar. Shell metacharacters in a crafted readable pathname could alter the command executed by the Webmin CGI process before archive contents were validated. Historical source.

Injection · Module · Cron (1)

- **WEBMIN-0974 — Cron shell-command operands were not escaped at the reviewed sites.** At the pre-change side of the reviewed diff, dynamic username and file-path operands are concatenated into shell-interpreted system or execute_command strings without escaping, and cron_editor.pl uses two-argument open forms. Historical source.

Injection · Module · Custom (4)

- **WEBMIN-0035 — Authenticated NSS username command injection in historical Custom Commands run-as execution.** The historical Custom Commands su path built `su - $user ...` by directly interpolating the authenticated or configured NSS username, then handed that string to safe_process_exec, which invokes `/bin/sh -c`. Historical source.
- **WEBMIN-0223 — Delegated Custom Commands runner can inject shell commands into the su environment prelude.** Historical set_parameter_envs serialized each request-derived parameter as NAME='VALUE' without escaping embedded single quotes. execute_custom_command wrote that serialization into an executable temporary shell script whenever the custom command used its su option. Historical source.
- **WEBMIN-0299 — Date parameter injection in generated user-environment script.** In the reviewed historical path, the year, month, and day fields of a custom-command date parameter were concatenated directly from the request. The resulting value was inserted between literal single quotes in environment-assignment text. Historical source.
- **WEBMIN-0990 — Run-as environment assignments allowed shell injection.** Custom Commands constructed an environment assignment by placing a request-controlled parameter directly between single quotes. For commands using the su path, that assignment was written to a temporary shell script and executed under the configured run-as account. Historical source.

Injection · Module · Fdisk (10)

- **WEBMIN-0093 — Delegated fdisk user could inject root shell commands through hdparm fields.** The historical fdisk apply_hdparm.cgi handler accepted authenticated request values for hdparm

options and drive, concatenated them without quoting into shell commands, and executed them in the Webmin process context. Historical source.

- **WEBMIN-0094 — Delegated fdisk user could submit an arbitrary root fsck command.** The historical fsck.cgi endpoint trusted the request's cmd field after checking only whether the same request's dev field passed the fdisk disk ACL. It forwarded cmd to proc.safe_process_exec_logged with UID and GID zero; the helper executed the string through /bin/sh -c without dropping the Webmin process identity. Historical source.
- **WEBMIN-0155 — Authenticated fdisk device parameter command injection executes as root.** In the reviewed historical fdisk state, mkfs.cgi accepts a request-controlled device value. The module's default disks=* ACL makes can_edit_disk accept any such value, mkfs_parse embeds it without shell escaping in the filesystem command, and safe_process_exec ultimately invokes /bin/sh -c without dropping from Webmin's root identity. Historical source.
- **WEBMIN-0283 — Parted request parameters allow authenticated root command injection.** The parted-backed partition creation and relabel flows accept request-controlled partition type, partition number, and partition-table format values after checking only that the authenticated module user may edit the selected server-enumerated disk. The historical helpers concatenate those values into command strings passed to backquote_logged. Historical source.
- **WEBMIN-0284 — Filesystem labels allow authenticated root command injection.** Partition edit and creation requests forward an authenticated disk editor's filesystem label to set_label. The historical helper places the label directly into single-quoted e2label syntax and a double-quoted xfs_db command string, then invokes the shell through system_logged. Historical source.
- **WEBMIN-0449 — Authenticated fdisk delegate can inject commands into parted execution.** In the reviewed historical parted path, save_part.cgi passes request-controlled type and newpart values to create_partition. That function concatenates type into the mkpart command and, when type is raid, concatenates the partition number into a second parted command before executing the string via backquote_logged. Historical source.
- **WEBMIN-0893 — Partition-table request value reaches the parted shell command unescaped.** In the reviewed historical source, relabel.cgi checks whether the requested device is allowed and then forwards the independent request field table to set_partition_table. That helper concatenates table directly into a parted command string and passes it to backquote_logged. Historical source.
- **WEBMIN-0945 — Unquoted hdparm option values enabled authenticated command injection.** On the pre-change side of the reviewed diff, apply_hdparm.cgi concatenates each submitted hdparm flag value directly into a command string. backquote_logged executes the resulting string with Perl backticks, so shell metacharacters in a crafted flag value can become commands. Historical source.
- **WEBMIN-0946 — Wildcard disk access exposed raw drive command injection.** In the reviewed post-change snapshot, apply_hdparm.cgi still appends the request-derived drive value to the apply command without quoting and also embeds it directly in the timing branch's Perl backticks. can_edit_disk returns true for any value when the fdisk disk ACL contains '*'. Historical source.
- **WEBMIN-0968 — Historical fdisk shell-command argument injection surface.** The reviewed direct diff shows multiple fdisk command builders placing dynamic drive, device, label, partition, extent, and filesystem-option values into shell command strings used by backticks, pipes, backquote_logged, system_logged, or pty_process_exec. Historical source.

Injection · Module · Fetchmail (1)

- **WEBMIN-0969 — Shell injection through Fetchmail command and cron value handling.** The reviewed Fetchmail code constructed commands for configuration checks, cron display/save, file ownership and mode changes, and daemon control using relevant values that were raw, merely enclosed in literal single quotes, or decoded by executing echo in backticks. Historical source.

Injection · Module · File (5)

- **WEBMIN-0026 — ACL filename allowed authenticated shell injection.** Historical getfacl.cgi constructed a backtick shell command by placing the request-derived filename between literal single quotes. An allowed filename containing a single quote could terminate that quoting and inject shell syntax, executing commands with the File Manager module's effective Unix identity, potentially root. Historical source.

- **WEBMIN-0316 — Conditional shell injection in legacy file metadata fallback.** In the reviewed legacy File Manager metadata path, the fallback used a shell command containing a filename inside literal single quotes. A filename containing its own single quote can terminate that quoting and append shell syntax. Historical source.
- **WEBMIN-0317 — Command execution through command-capable file open.** The reviewed file-serving path took PATH_INFO and supplied it as the entire expression to Perl's two-argument open after only a filesystem-root authorization check. Pipe-shaped values are process-open syntax, allowing an authenticated File Manager principal to invoke a command with the module's operating identity. Historical source.
- **WEBMIN-0404 — Authenticated file-search filters permit shell command injection.** The parent-side search.cgi source reviewed for the manifest change appends request-controlled type, user, group, and size filters directly to a find command and invokes the resulting string through a shell pipe. ReadParse URL-decodes those fields without an allowlist. Historical source.
- **WEBMIN-0669 — Authenticated command injection through File Manager attribute paths.** The legacy getattrs.cgi and getext.cgi handlers accepted a request-derived file path, checked only whether its absolute location was within an allowed root, and then embedded it inside a single-quoted shell fragment executed through Perl backticks. Historical source.

Injection · Module · Filemin (2)

- **WEBMIN-0556 — Shell substitution through metadata-listing filenames.** The historical File Manager metadata listing constructed backtick-executed lsattr and SELinux commands by placing full directory-entry names inside double quotes. Historical source.
- **WEBMIN-0805 — Authenticated command injection in File Manager ACL handling.** The reviewed ACL handler constructs a shell command from request-controlled action, permission, account, entry-type, recursion, and manual fields. Its metacharacter deny-list does not remove shell pipe or redirection operators, and backquote_logged ultimately executes the string using Perl backticks. Historical source.

Injection · Module · Filter (1)

- **WEBMIN-0326 — Autoreply subject injection permits Unix-account command execution.** The historical full and simplified filter save endpoints accept an autoreply subject without rejecting line breaks. The subject serializer writes that value into a directive-oriented file as `Subject: value`; an injected newline can add a later `From:` directive. Historical source.

Injection · Module · Firewalld (1)

- **WEBMIN-0765 — Authenticated command injection through FirewallD rule-listing zone.** In the reviewed historical list_rules.cgi snapshot, ReadParse places the request-controlled zone parameter in $dzone. The page concatenates that value directly into a firewall-cmd command string and passes the string, including shell redirections, to open_execute_command. Historical source.

Injection · Module · Format (1)

- **WEBMIN-0985 — Historical shell command injection in Solaris filesystem actions.** The reviewed diff shows that the Solaris format module's fsck, newfs, and tunefs action handlers had request-derived values incorporated into strings executed through Perl pipe opens. Historical source.

Injection · Module · Fsdump (4)

- **WEBMIN-0233 — Historical FreeBSD custom remote-shell input permits root shell injection.** In the exact reviewed FreeBSD snapshot, the custom remote-shell parser validates only the first whitespace-delimited executable token and returns the full string. The FreeBSD tar builder appends that string without quoting to a command executed by the root backup path. Historical source.
- **WEBMIN-0347 — FreeBSD fsdump directory shell injection in pre-hardening command construction.** The FreeBSD backup command builder represented the configured source directory by embedding it inside literal single quotes in a shell command. The save path accepted any existing ACL-permitted directory, including a Unix pathname containing a single quote. Historical source.

- **WEBMIN-0392 — Filesystem Backup remote-file input permits authenticated root command injection.** At the manifest commit state, the Linux and FreeBSD Filesystem Backup implementations accept a non-empty remote-file path (hfile) from an authenticated module user and concatenate it inside single quotes when constructing backup or restore shell commands. A quote-bearing hfile can terminate that argument and append shell syntax. Historical source.
- **WEBMIN-0470 — Delegated Filesystem Backup editor can inject root shell commands through a planted source directory.** In the historical Linux tar and XFS backup command builders, date-expanded source directories were serialized by surrounding each value with a literal single quote. A delegated Filesystem Backup editor could select an existing, ACL-authorized directory whose Unix name contained a quote followed by shell syntax. Historical source.

Injection · Module · Gray Theme (1)

- **WEBMIN-2454 — Authenticated navigation parameter can forge diagnostic log lines.** In the reviewed historical state, Gray theme left.cgi writes the raw authenticated request's mode value to STDERR before checking whether it names an allowed menu mode. Historical source.

Injection · Module · Init (6)

- **WEBMIN-0116 — Authenticated root command injection in runlevel editor.** In the historical request path reviewed around the UI rewrite, type-1 edit_action.cgi arguments are used to construct a runlevel filename and the filename is interpolated into a backtick cat command. Historical source.
- **WEBMIN-0148 — Historical authenticated root command injection in Bootup action execution.** At the assigned historical commit snapshot, init/start_stop.cgi accepts any authenticated init-module user with a truthy bootup ACL, concatenates the request-controlled file parameter with a selected action, and sends the resulting string to safe_process_exec_logged. That helper reaches safe_process_exec, which invokes /bin/sh -c without reducing privileges. Historical source.
- **WEBMIN-0241 — Init start/stop file parameter command injection executes as root.** In the reviewed historical start_stop.cgi path, any authenticated Init-module user with bootup action permission can submit an arbitrary file value. The endpoint concatenates that value with a fixed action and sends the string to safe_process_exec_logged without allowlisting or shell quoting. Historical source.
- **WEBMIN-0345 — Run-only init-module users can inject root shell commands through start_stop.cgi.** At the reviewed historical snapshot, start_stop.cgi treats both full boot-action access (bootup=1) and the deliberately restricted run-only tier (bootup=2) as authorized. Historical source.
- **WEBMIN-0977 — Command injection through submitted init action path.** The reviewed start_stop.cgi delta shows that request field file was previously concatenated with an action to form a command executed by Webmin's privileged process. A principal with bootup permission could supply shell syntax rather than a known init action path, turning service control into arbitrary command execution. Historical source.
- **WEBMIN-0978 — Command injection through submitted runlevel.** The reviewed change_rl.cgi delta shows that request field level was interpolated into a shell command invoking telinit without first requiring a real runlevel. A principal with bootup permission could break out of the intended argument and execute arbitrary commands with Webmin's operating-system privileges. Historical source.

Injection · Module · Ipfw (1)

- **WEBMIN-0111 — Authenticated IPFW rule interface injection permits root command execution.** In the reviewed numeric-format IPFW apply path, authenticated BSD Firewall module users can store a custom via, xmit, or recv interface containing shell metacharacters because validation rejects only whitespace. rule_lines places the value into a command string, and apply_rules executes that string through backquote_logged. Historical source.

Injection · Module · Ipsec (1)

- **WEBMIN-0980 — Authenticated OS command injection in the IPsec connection-up action.** On the pre-change side of the reviewed diff, ipsec/up.cgi reads the conn CGI value and interpolates it into a single-quoted command string. proc::safe_process_exec_logged delegates to safe_process_exec, which executes the string with /bin/sh -c. Historical source.

Injection · Module · Itsecur Firewall (2)

- **WEBMIN-0286 — Root-context command injection across firewall utility workflows.** Several authenticated module operations concatenate request-controlled values into shell command strings. Historical source.
- **WEBMIN-0329 — Scoped firewall-restore user can inject commands through the ZIP password.** The historical IT Security and Firewall restore endpoint accepts an authenticated user's custom ZIP password and constructs `-P '$in{'pass'}'` inside a larger command string. It then supplies that string to backquote_logged, whose implementation executes it with Perl backquotes. Historical source.

Injection · Module · Lpadmin (2)

- **WEBMIN-0280 — Historical CUPS PPD parser path injection allowed root command execution.** The historical parse_cups_ppd implementation interpolated the submitted gzip PPD path into a shell-form open pipeline. Historical source.
- **WEBMIN-0281 — Historical CUPS driver file operations allowed root shell injection.** The historical CUPS create_driver and delete_driver routines built rm, gunzip, and cp shell commands from the request-selected PPD path and printer-derived destination. Historical source.

Injection · Module · Lvm (2)

- **WEBMIN-0077 — Authenticated LVM device-path shell injection executes commands as root.** The historical LVM physical-volume request path accepts an explicit readable device path and passes it to helpers that build shell command strings with hand-written single quotes. The LVM2 migration branch reviewed in this review likewise constructs `yes | pvmove '$device'`. Historical source.
- **WEBMIN-0168 — Delegated LVM user can inject commands through physical-volume percentage selection.** The historical physical-volume percentage branch in save_lv.cgi copies the request parameter pvof into the LV size_of field. create_logical_volume then appends /dev/<size_of> without shell escaping to an lvcreate command passed to backquote_logged. Historical source.

Injection · Module · Mailboxes (1)

- **WEBMIN-0400 — Local MTA command injection through the envelope sender.** The local-MTA delivery branch constructed a shell pipe command containing an unquoted envelope sender derived from the message's From header. An authorized sender able to place shell metacharacters in that parsed address could change the command executed by the mailbox process. Historical source.

Injection · Module · Mailcap (1)

- **WEBMIN-0973 — Mailcap configuration path reached command-interpreting two-argument open.** The reviewed historical code passed the configured mailcap pathname to Perl's two-argument open by concatenating it with the read marker. Perl can interpret special pipe/open-mode syntax in that single argument. Historical source.

Injection · Module · Majordomo (2)

- **WEBMIN-0581 — Delegated Majordomo list creation leads to root shell command injection.** The list-creation endpoint accepts any non-whitespace name, including shell syntax, and creates the associated list files for a principal holding only the module's create capability. The overview later enumerates that name, builds the members pathname, and interpolates it unquoted into a shell backtick used to run `cat ... | wc -l`. Historical source.
- **WEBMIN-0583 — Global taboo values escape a fixed heredoc into executable Majordomo configuration.** The global settings endpoint passes multiline taboo-header and taboo-body request data to a serializer that always emits `END` as a Perl heredoc delimiter. It transforms leading hyphens and blank lines but leaves a line equal to `END` intact. Historical source.

Injection · Module · Mon (1)

- **WEBMIN-0174 — Legacy MON command CGI permits root shell injection through the username.** The preimage deleted by reviewed commit 3c44db87eaf8639bc4da1a24df190ea594765af3 con-

tains an executable MON CGI that reads the `user` request field and embeds it unquoted in the string form of a Perl `open` pipe. Shell metacharacters therefore break out of the intended moncmd.pl arguments and execute arbitrary commands under the privileged Webmin CGI worker. Historical source.

Injection · Module · Mount (4)

- **WEBMIN-0036 — Historical mounted-source command injection through vol_id.** The historical Linux mount enumerator passed a mounted-source string beginning with /dev/ directly into a shell command invoking vol_id. A local unprivileged FUSE user able to control the recorded source could include shell syntax; filesystem enumeration by the root Webmin process would execute that syntax as root. Historical source.
- **WEBMIN-0039 — Historical mounted-path command injection in disk-space reporting.** Historical disk_space implementations interpolated live mountpoint paths directly into df or bdf shell command strings. An unprivileged local user able to create a FUSE mount at a shell-metacharacter-bearing path could cause the root Webmin process to execute additional commands when Filesystem Manager collected usage data. Historical source.
- **WEBMIN-0312 — Authenticated root command injection through swap-file name.** The swap creation CGI forwards the request-controlled cswap_file value to OS-specific create_swap functions after validating only cswap_size. In the reviewed Linux, FreeBSD, NetBSD, and OpenBSD implementations, the filename was embedded directly in dd and mkswap shell commands. Historical source.
- **WEBMIN-0575 — Authenticated NFS mount fields reached root shell commands in historical platform backends.** In the reviewed historical source, save_mount.cgi parsed authenticated request fields and called operating-system-specific check_location routines after mount-module authorization. Historical source.

Injection · Module · MySQL (7)

- **WEBMIN-0049 — Historical import paths permitted authenticated shell command injection.** The reviewed import path used a user-selected server-side filename directly in a shell-form cp invocation and later concatenated the same path unquoted into the mysqlimport command. Historical source.
- **WEBMIN-0375 — Backslash-aware quote reinterpretation in historical password SQL.** The historical password-update flow doubles apostrophes, places the result inside a single-quoted MySQL password expression, and then applies a helper that doubles backslashes only inside double-quoted text. A backslash before an apostrophe can therefore change how MySQL parses the doubled quote. Historical source.
- **WEBMIN-0644 — Authenticated MySQL backup WHERE clause reaches a privileged shell.** In the reviewed historical snapshot, backup_db.cgi accepts a user-supplied mysqldump WHERE clause after database and backup ACL checks and passes it to backup_database. The helper takes the global request value, surrounds it with shell double quotes, appends it to a pipeline command, and invokes backquote_logged. Historical source.
- **WEBMIN-0663 — Global-variable form values can alter SQL executed by the CLI backend.** In the reviewed system-variable save path, an authenticated account with MySQL permission-management access can place quote and statement syntax into a value that is concatenated into SET GLOBAL. The module's supported no-DBI execution path writes the resulting SQL text to a temporary file and sends it to the mysql client. Historical source.
- **WEBMIN-0664 — Backup filtering exposed shell command substitution.** The reviewed backup change replaces a request-derived mysqldump where clause that had been placed inside shell double quotes with a quotemeta-escaped argument. Shell command substitutions remain active inside double quotes, and backup_database concatenates that fragment into a command executed by backquote_logged. Historical source.
- **WEBMIN-0787 — Authenticated local SQL-file path injection could execute host commands.** The local-file restore handler accepted a readable request-supplied path and, for compressed inputs, interpolated it directly into a shell redirection passed to backquote_command. A delegated database administrator who could arrange a readable crafted filename could cause shell metacharacters to execute with the Webmin CGI's host privileges. Historical source.

- **WEBMIN-0879 — Administrative MySQL queries interpolate attacker-influenced values instead of binding them.** Multiple reviewed MySQL module paths assembled SQL by inserting request-derived keys or values directly into administrative queries. In the strongest constrained-user path, can_edit_db authorizes only olddb while olduser, oldhost, table, or column keys shape DELETE predicates on MySQL privilege tables. Historical source.

Injection · Module · Net (6)

- **WEBMIN-0021 — IPv6 prefix validation permits root shell command injection in active-interface activation.** The historical IPv6 prefix validator accepts non-numeric suffixes because it only performs Perl numeric comparisons. The active-interface workflow concatenates the accepted prefix into an unquoted ifconfig shell command, allowing an authorized interface editor to append a root command. Historical source.
- **WEBMIN-0022 — Hostname validation allows command substitution in a privileged shell call.** The historical hostname filter uses the ASCII range A-z, which includes backticks. Gentoo hostname saving interpolates that value unquoted into a system_logged shell command. Historical source.
- **WEBMIN-0258 — Historical VLAN activation parameters permit root shell command injection.** The historical VLAN save path accepts physical and vlanid request fields without syntactic validation. When an authorized interface editor requests activation, activate_interface concatenates those values into vconfig, ifup/ifdown, or ifconfig shell command strings without quoting. Historical source.
- **WEBMIN-0527 — Delegated network editors can inject root-executed ifupdown directives.** The audited bonding-primary and bridge-timing additions copy authenticated request values into Debian interface option structures without rejecting carriage returns or newlines. The historical Debian writer interpolates each option directly into /etc/network/interfaces. Historical source.
- **WEBMIN-0629 — Netplan hostname update accepted shell command substitution.** In the reviewed historical Netplan path, a user with Network Configuration DNS write permission could submit a hostname containing backticks. The historical `[A-z]` validator accepted the backtick character, and save_hostname embedded the value unquoted in a single-string system command executed with Webmin's service privileges. Historical source.
- **WEBMIN-0732 — Unescaped bond and VLAN fields reached privileged shell commands.** The historical Linux network activation flow accepted posted bond and VLAN attributes into an interface object and constructed privileged shell command strings by direct interpolation. Bond timing and mode values and VLAN ID lacked syntax constraints in the reviewed request path. Historical source.

Injection · Module · Nis (2)

- **WEBMIN-0259 — Authenticated NIS slave parameter command injection.** The historical IPv6 validation path accepts an IPv6-looking NIS slave value whose netmask begins with an in-range number but continues with shell syntax. to_ip6address returns that accepted input unchanged, and the NIS server configuration path interpolates it into an unquoted ypinit shell command. Historical source.
- **WEBMIN-1005 — OpenLinux prior-domain command injection and destructive path control.** The OpenLinux server configuration treated old_N as a client-supplied hidden field and used it to identify an existing NIS domain. In the domain-removal branch shown by the reviewed change, that value was interpolated into a shell-parsed rm -rf command without validating its syntax or confinement. Historical source.

Injection · Module · Package Updates (2)

- **WEBMIN-0528 — Authenticated bulk package update command injection executes with Webmin privileges.** In the audited historical snapshot, update.cgi derives package names from request-controlled u fields and the bulk helper forwards their joined value without the enumerated-package validation used by the single-package helper. Historical source.
- **WEBMIN-0643 — Authenticated Package Updates user can inject a root shell command through an unknown APT package name.** The historical new-package workflow accepts a request-supplied package name that is absent from package metadata and forwards it unchanged to the APT installer. The historical APT backend builds its executed shell command before escaping the package string, so shell metacharacters in the name are interpreted by the shell. Historical source.

Injection · Module · Postfix (2)

- **WEBMIN-0378 — Delegated Postfix SMTP-server user can inject root shell commands through parameter lookup.** The historical edit_access.cgi endpoint is available to a Postfix user holding the smtpd ACL and forwards the request-controlled name parameter to get_current_value. If the supplied name is absent from main.cf, the helper invokes postconf through a shell command containing the unquoted name. Historical source.
- **WEBMIN-0857 — Postfix compatibility-level value could inject Perl during General Options rendering.** The historical resolver obtains compatibility_level through get_current_value(), which first reads the value directly from Postfix main.cf, then interpolates that value into eval "$compatibility_level $op $lvl" while resolving a compatibility conditional. Historical source.

Injection · Module · PostgreSQL (4)

- **WEBMIN-0164 — Delegated PostgreSQL group management permits SQL injection.** The group-management save endpoint accepts any non-whitespace group name, uses it inside a quoted lookup literal, and then concatenates it and selected PostgreSQL usernames as syntax in create and alter statements. Historical source.
- **WEBMIN-0165 — Perl magic-open in SQL history enables authenticated root command execution.** The SQL-history form reads a filename formed from a fixed commands-file prefix plus the request db value using Perl's two-argument open without an explicit mode. A trailing pipe activates command-pipe semantics. Historical source.
- **WEBMIN-0269 — PostgreSQL role-management fields are interpolated into administrative SQL.** The reviewed role save handler validates pname and ppass only as non-whitespace, then inserts them directly into double-quoted identifiers or single-quoted password literals. Historical source.
- **WEBMIN-0982 — PostgreSQL user/group operations embedded request values into SQL.** The reviewed user/group management diff shows request-derived identifiers, passwords, validity values, and membership data being replaced with strict validation, identifier/literal quoting, or bound parameters. Historical source.

Injection · Module · Proc (5)

- **WEBMIN-0105 — Process renice parameters reach a shell command without quoting.** In the reviewed historical state, renice_proc.cgi accepts pid and nice request fields and, after authorizing management of the selected process, passes them to platform helpers that concatenate them into shell command strings without quoting. Historical source.
- **WEBMIN-0291 — Authenticated Process Manager file search could execute privileged shell commands.** The historical Process Manager search endpoint accepted authenticated request values for filesystem and file searches and passed them to operating-system fuser helpers. In the unsafe forms exposed by the assigned diffs, those values were concatenated into shell command strings and executed with Perl backticks in Webmin's privileged service context. Historical source.
- **WEBMIN-0692 — Shell command injection through lsof-backed process search selectors.** The parent-side implementation shown by the reviewed diff built lsof pipe commands by embedding process-search protocol/port and IP values inside single quotes. The process-search CGI supplied those values directly from parsed request parameters and did not require the module's separate command-run permission. Historical source.
- **WEBMIN-0860 — Process-trace syscall filter reached a privileged shell without quoting.** In the reviewed process-trace flow, the syscalls request value was only split on whitespace before the Linux trace helper joined it into a string-form shell pipe. A permitted Process Manager user could attach shell syntax to the filter and run commands with the Webmin service account. Historical source.
- **WEBMIN-0926 — Authenticated shell injection through Process Manager command parameters.** Within the reviewed historical state transition, Proc backend functions assembled ps, renice, strace or truss, fuser, and Windows process command strings from PID, nice-level, syscall-list, or mount-point arguments without shell quoting on multiple supported operating systems. Historical source.

Injection · Module · Qmailadmin (1)

- **WEBMIN-0451 — Qmail autoreply sender header reaches an unescaped shell command.** At the audited commit, qmail autoreply aliases execute the shared sendmail autoreply handler. That handler derives its default reply sender from unauthenticated message To/Cc/Bcc headers, and its address parser accepts tokens containing shell separators and redirection characters. Historical source.

Injection · Module · Quota (8)

- **WEBMIN-0044 — Quota-limit fields allowed authenticated root command injection.** The user and group quota setters constructed configured setquota invocations as shell strings and historically concatenated four quota-limit values without numeric normalization. Historical source.
- **WEBMIN-0209 — Delegated Quota user can inject root commands through the filesystem parameter.** On the reviewed Linux path, check_quotas.cgi accepts the request-controlled filesys value after can_edit_filesys. Historical source.
- **WEBMIN-0271 — Quota activation filesystem parameter permits authenticated root command injection.** In the reviewed Linux quota initialization code, activate.cgi takes dir from an authenticated request and permits any value when the Quota module filesystem ACL is the default wildcard. quotaon then concatenates that value into convertquota -u and -g command strings passed to system_logged. Historical source.
- **WEBMIN-0348 — Delegated FreeBSD quota viewer can execute shell commands through the filesystem selector.** In the reviewed historical FreeBSD quota helpers, list_users.cgi and list_groups.cgi accept a request-controlled filesystem string. can_edit_filesys authorizes every value when the delegated Disk Quotas ACL grants all filesystems, including for read-only quota users. filesystem_users and filesystem_groups then interpolate that string directly into backtick-based repquota commands. Historical source.
- **WEBMIN-0408 — Delegated Disk Quotas deactivation allows root shell-command injection.** In the historical snapshot reviewed at 72bb51c7514bea72c283cfafc84ec9cba75aff82, activate.cgi accepts a request-controlled dir whenever the delegated Disk Quotas ACL has filesys=* and enable=1. Historical source.
- **WEBMIN-0471 — Historical authenticated XFS quota identifier command injection as root.** The historical XFS quota query and user-limit paths assembled shell command strings by inserting an authenticated request's user or group identifier directly inside a single-quoted xfs_quota -c argument. Disk Quotas ACL mode 0 accepted arbitrary submitted identifiers, and the query CGI did not require the identifier to resolve to an operating-system account. Historical source.
- **WEBMIN-0894 — Authenticated shell command injection through XFS quota user or group names.** The XFS quota query and limit-setting paths constructed shell command strings containing user or group names and passed those strings to backquote_logged, which evaluates commands using Perl backticks. Historical source.
- **WEBMIN-0970 — Shell injection through unescaped Quota filesystem arguments.** The reviewed manifest diff shows filesystem values placed directly into shell-interpreted command strings used by quota-management and filesystem-space operations across seven platform backends. A reachable value containing shell metacharacters could change the executed command under the Webmin process privileges. Historical source.

Injection · Module · Samba (1)

- **WEBMIN-0135 — Samba password update permits root command injection through output redirection.** The password-error propagation change declares a third output-reference argument for set_password, but supplies the second argument—the submitted password—as execute_command's stdout and stderr values. The historical command wrapper treats non-reference output values as filenames, concatenates them unquoted into redirection syntax, and invokes /bin/sh -c. Historical source.

Injection · Module · Sendmail (1)

- **WEBMIN-0636 — Remote mail headers could execute shell commands through Sendmail autoreply.** When a Webmin-configured Sendmail autoreply alias processed a message, the script parsed

attacker-controlled To, Cc, and Bcc headers, selected one address for the reply From value, and concatenated that parsed value into a string-form pipe command without escaping. Historical source.

Injection · Module · Software (12)

- **WEBMIN-0042 — APT package parameter command injection executes as root.** In the audited historical APT backend, authenticated install_pack.cgi requests with source=3 pass the update parameter to update_system_install. The function constructs $cmd from raw update before applying quotemeta to a separate variable, then passes the prebuilt raw command to open_execute_command. Historical source.
- **WEBMIN-0146 — APT update package parameter reaches a root shell command before quoting.** In the reviewed APT installation flow, install_pack.cgi passes the request-derived update field to update_system_install. apt-lib.pl then interpolates that value into the command string before calculating a quotemeta-transformed copy, and later executes the already-built string. Historical source.
- **WEBMIN-0491 — APT package input is executed through the shell before quoting.** In the audited APT snapshot, `install_pack.cgi` passes request-derived update text to `update_system_install`. That function constructs `$cmd` with the raw value, only afterwards transforms `$update` with `quotemeta`, and then executes the already constructed command through `open_execute_command`. Historical source.
- **WEBMIN-0615 — Authenticated command injection through raw IPKG package arguments.** In the reviewed IPKG backend snapshot, the package-details path passes a decoded package request value to package_info, which interpolates it into an ipkg shell command before package validation. The same backend snapshot also concatenates package arguments in install and removal helpers. Historical source.
- **WEBMIN-0616 — HTTP response splitting in IPKG tree redirects.** The reviewed IPKG tree open and close endpoints concatenate the decoded what parameter into a redirect fragment. The default Webmin redirect helper emits the assembled value directly in the Location header. Historical source.
- **WEBMIN-0650 — APT package input can alter the privileged shell command.** The reviewed APT install path accepts a package string from the authenticated Software Packages flow, applies quotemeta token-by-token, and then removes the escape from every non-word character before constructing the shell command. Historical source.
- **WEBMIN-0652 — YUM package input can alter the privileged shell command.** The reviewed YUM/DNF install path constructs a combined command for the optional repository prerequisite and requested package transaction. That command uses raw request-derived package text and is passed to a shell-backed execution helper; the former per-name quoted value is not used. Historical source.
- **WEBMIN-0756 — RPM wildcard pathname allowed shell metacharacter interpretation.** The reviewed RPM file_packages wildcard path accepted a CGI request value and constructed a shell-interpreted rpm command. In the relevant prior-side code, that branch interpolated the raw pathname rather than the function's existing quotemeta value, allowing shell metacharacters in a crafted wildcard pathname to alter command execution. Historical source.
- **WEBMIN-0778 — Authenticated package-name shell injection in APT and YUM/DNF install backends.** The Software Packages install handler accepts a package update field and routes it to the active package backend. In the historical APT path, a quoting pass was followed by logic that removed the quoting backslashes; in the historical YUM/DNF path, expanded package names were joined without shell quoting. Historical source.
- **WEBMIN-0975 — Privileged command injection through unescaped Software module arguments.** Multiple Software module backend functions constructed shell commands by directly interpolating package identifiers, package-file paths, alternate root paths, and temporary paths. Values from authenticated module requests and package metadata could therefore introduce shell metacharacters at the audited sinks. Historical source.
- **WEBMIN-1018 — Authenticated command injection in package file-type probing.** The reviewed viewer snapshot takes attacker-controlled PATH_INFO and embeds it inside a double-quoted backtick command used to invoke the file utility. A path containing quote and shell syntax can escape the intended argument and execute commands with Webmin service privileges when MIME fallback is reached. Historical source.
- **WEBMIN-2417 — FreeBSD ports selector reaches a privileged shell command without validation.** The reviewed FreeBSD ports backend accepts a package selector from the source=3 installation

request, splits it on whitespace, and places each token directly inside a shell command of the form cd /usr/ports/$w && make package. Shell metacharacters in a token are not validated or quoted before open_execute_command runs the command.

Injection · Module · Squid (2)

- **WEBMIN-0446 — Cache-manager CGI arguments enable shell command injection.** In the reviewed historical state, cache-manager request arguments from CGI ARGV were joined unquoted into shell-form pipe opens. A user allowed to use the cache manager could place shell metacharacters in those arguments and execute commands with the Webmin CGI's privileges. Historical source.
- **WEBMIN-0930 — Authenticated OS command injection in Squid cache-manager argument handling.** The pre-change Squid cache-manager wrapper concatenated decoded CGI command-line arguments directly into the command string used by two-argument piped open. An authenticated Webmin user granted Squid cache-manager (cms) access could supply shell metacharacters in a no-equals query string and cause the Webmin process to execute attacker-selected shell syntax. Historical source.

Injection · Module · SSH Server (3)

- **WEBMIN-0260 — Authenticated malformed IPv6 listener injects root-owned sshd_config directives.** In the reviewed historical SSH Server module, a selected listener address was passed through check_ip6address and then written verbatim as a ListenAddress value. The validator split the last block at /, numerically compared the unrestricted suffix, and validated only the prefix block. Historical source.
- **WEBMIN-0461 — Delegated Unix-user creation reaches root shell through SSH key synchronization.** In the manifest commit state, useradmin_create_user constructs a root shell pipeline containing the newly-created username directly after su. The user-creation endpoint strips line terminators but otherwise permits shell metacharacters unless an optional administrator regex is configured. Historical source.
- **WEBMIN-0730 — Plain SSH key passphrase could escape a privileged shell command.** The reviewed SSH new-user hook constructed a privileged shell string by inserting plain password text into a double-quoted ssh-keygen argument nested inside a single-quoted su command. A password containing a single quote could escape the outer quoting context and append shell syntax executed by the Webmin service account, ordinarily root. Historical source.

Injection · Module · Status (2)

- **WEBMIN-0551 — Custom scheduled-SMS subject permits authenticated root command execution.** The historical custom scheduled-SMS subject handler stores a value after checking only that it contains non-whitespace. The historical generic configuration writer emits embedded LF characters verbatim, so a subject can append arbitrary name=value records to the Status module configuration. Historical source.
- **WEBMIN-0922 — Delegated query-monitor configuration could reach Perl code evaluation.** The reviewed query-monitor parser interpolated the request's `driver` value into string-form Perl evaluation before storing it. Monitor-save dispatch invokes this parser for an authenticated status user with edit capability. Historical source.

Injection · Module · Syslog (1)

- **WEBMIN-0626 — Historical authenticated Syslog filter option injection into root grep.** On the historical Linux filtered-log path, Webmin applied Perl quotemeta to a viewer-controlled filter and interpolated it as grep's first non-fixed argument without an option terminator. The shell removed the escaping before grep parsed arguments, so a leading value such as -f/path remained a grep option. Historical source.

Injection · Module · Updown (2)

- **WEBMIN-0119 — Shell injection through directory ZIP temporary filename.** The directory-download branch derives its temporary archive filename from the requested directory's basename and places that value unquoted in a shell command. A permitted directory basename containing command substitution causes the shell to execute attacker-selected commands. Historical source.

- **WEBMIN-0231 — Authenticated Up/down directory fetch permits root shell command injection.** In the historical directory-fetch path adjacent to the reviewed change, the final directory component becomes a temporary ZIP filename. tempname preserves supplied names lacking dot-dot, and the temporary path was interpolated unquoted into a shell command. A permitted directory basename containing shell syntax could therefore append commands. Historical source.

Injection · Module · Useradmin (3)

- **WEBMIN-0877 — FreeBSD login-history username reached a privileged shell command unescaped.** In the directly reviewed parent-side FreeBSD implementation, `open_last_command` embedded the request-derived username in a string-form pipe open as `last $user`. `list_logins.cgi` accepted the username selector from the request and, for wildcard login-history access, allowed any value before forwarding it through `list_last_logins`. Historical source.
- **WEBMIN-0897 — Login-history arguments can inject shell commands on macOS.** The reviewed login-history flow takes the username request parameter and, when the Webmin ACL grants logins='*', accepts any value without requiring it to match a configured username. On macOS, open_last_command places that value directly into a string-form Perl pipe open, which invokes a shell. Historical source.
- **WEBMIN-0976 — OS command injection through unescaped Useradmin command operands.** On affected Useradmin platform and operation paths, account-controlled values such as username, real name, home directory, shell, group data, and related identifiers were interpolated into command strings consumed by helpers that invoke a shell. Historical source.

Injection · Module · Usermin (2)

- **WEBMIN-0966 — Authenticated Usermin command injection through unescaped request and path arguments.** On the pre-change side of the reviewed patch, Usermin Configuration operations place request-derived module names and filesystem paths directly into shell-parsed strings used for tar, gzip, OpenSSL/cat, package, archive, setup, and removal commands. Historical source.
- **WEBMIN-2436 — Usermin sort option accepts configuration-line injection.** The historical advanced-options save path stores the decoded sortconfigs request value directly, and the historical write_file implementation serializes that value without removing embedded newlines. A crafted value can therefore create additional records in the global Usermin configuration rather than changing only the intended boolean option.

Injection · Module · Vgetty (1)

- **WEBMIN-0833 — Authenticated command injection through Vgetty upload filename.** In the reviewed historical Vgetty upload flow, ReadParseMime copied the client-supplied multipart filename into wav_filename. upload.cgi removed path components and a trailing .wav suffix but did not reject or shell-quote metacharacters before interpolating the value into a single-string system("mv ...") call. Historical source.

File and path handling (32)

Files and paths · Core · Runtime (3)

- **WEBMIN-0267 — File chooser root restriction can be dropped by a sibling chroot prefix.** chooser.cgi clears its configured root restriction whenever the request chroot merely begins with the root string. Because the test has no equality-or-directory-separator boundary, /srv/user2 is treated as being under /srv/user. Historical source.
- **WEBMIN-0931 — Multipart direct-write path traversal.** ReadParseMime direct-write mode used the client-supplied multipart filename as a filesystem path below a caller-provided directory. A filename containing separators, traversal components, or an absolute path could select an unintended writable destination. Historical source.
- **WEBMIN-0939 — Module identifier path traversal.** get_module_info rejected only module identifiers beginning with a dot before joining the value to a module root and reading module.info. Identifiers containing separators or traversal components could escape the installed-module namespace and cause metadata to be loaded from unintended filesystem locations. Historical source.

Files and paths · Core · Cross-cutting (1)

- **WEBMIN-0670 — Authenticated arbitrary root-context file truncation through reboot-notification dismissal.** The reboot-notification branch accepts removenotify_value from a user with reboot permission and passes it directly to write_file_contents without content. In the privileged Webmin process this selects an arbitrary path to create or truncate, violating the limited reboot ACL boundary. Historical source.

Files and paths · Core · Integration (4)

- **WEBMIN-0741 — File Manager upload path traversal enables writes outside allowed directories.** The reviewed File Manager upload flow accepts slash-separated multipart filenames, creates their directory components relative to the selected working directory, and opens the concatenated target without rejecting parent-directory components or revalidating the resulting path against File Manager allowed paths. Historical source.
- **WEBMIN-0782 — Filemin ACL operation can escape configured allowed paths.** The reviewed historical Filemin ACL endpoint authorizes the current directory but constructs each target from that directory and a request-controlled filename without canonicalizing the result or rechecking it against Filemin's allowed paths. Historical source.
- **WEBMIN-0852 — Direct-upload multipart filename permits root-context path traversal.** In the audited historical delta, `ReadParseMime` directly concatenates a client-supplied multipart filename with the direct-upload temporary directory and opens that path for writing. Because the filename is not reduced to a basename before the open, traversal sequences can escape the temporary directory. Historical source.
- **WEBMIN-0950 — SSH authorized_keys management could follow user-controlled symlinks.** Privileged user management appended an SSH key through a path below the target home without rejecting .ssh or authorized_keys symlinks. A malicious local user or pre-positioned home could redirect that write to an attacker-selected path. Historical source.

Files and paths · Core · Packaging (1)

- **WEBMIN-0929 — Root repository setup used a predictable /tmp file for sources.list replacement.** The reviewed root-only repository setup snapshot writes filtered APT configuration to /tmp/sources.list.$$ using ordinary shell redirection and then moves that pathname over /etc/apt/sources.list. The PID-derived pathname is predictable and is neither created exclusively nor placed in a private directory. Historical source.

Files and paths · Core · Webmin (1)

- **WEBMIN-2430 — World-writable ACME response permits local validation tampering.** The bundled ACME HTTP-01 path writes the key-authorization response and changes the file to mode 0777. The response must be web-readable, but granting universal write permission allows another local account to replace it during validation.

Files and paths · Module · Ajaxterm (1)

- **WEBMIN-0196 — Predictable Ajaxterm log path permits privileged file truncation.** The Ajaxterm CGI child opens the fixed pathname /tmp/ajaxterm.out with truncating output redirection before executing the backend. Historical source.

Files and paths · Module · Cron (1)

- **WEBMIN-0434 — Pre-validation cron username path enables root arbitrary file overwrite.** In the historical pre-change side of the reviewed Cron security patch, save_cron.cgi and save_env.cgi construct cron_dir/user from the request and lock it before checking whether the username exists or whether the module ACL permits that target. Historical source.

Files and paths · Module · File (1)

- **WEBMIN-0403 — Recursive chmod can modify a symbolic-link target outside allowed roots.** In the reviewed chmod.cgi state, recursive mode 1 authorizes only the requested directory, then iterates its

direct children. Its link and directory tests use $full without assigning it in that branch, so a symbolic-link child is not skipped and is passed to update. update applies chown and chmod to the child pathname, which follows the link target. Historical source.

Files and paths · Module · Filemin (5)

- **WEBMIN-0716 — Multipart upload paths could write outside allowed directories.** The directory-upload handler copied multipart filename text, created slash-delimited directories, and opened *cwd*/file without rejecting traversal components or authorizing the resulting target. An authenticated restricted uploader could select a path outside allowed directories and write with the Filemin process identity. Historical source.
- **WEBMIN-0806 — File Manager ACL changes can traverse outside allowed paths.** The reviewed ACL endpoint authorizes the requested directory through get_paths but appends each request-controlled file name without rejecting traversal or rechecking the combined target against allowed paths. A readable target outside the configured File Manager root can consequently receive ACL changes under the module's effective Unix identity. Historical source.
- **WEBMIN-0849 — Archive output traversal can write outside allowed paths.** The reviewed compression handler forms the output path by concatenating the allowed current directory, the request-controlled archive name, and a server-selected extension. Historical source.
- **WEBMIN-0885 — File mutations can traverse beyond File Manager allowed paths.** In the review revision, delete.cgi, rename.cgi, and paste.cgi construct filesystem operands by appending names to an already-authorized working or source directory but do not validate each resulting operand. A dot-dot-bearing name can therefore resolve outside the configured File Manager allowed paths. Historical source.
- **WEBMIN-0992 — File action path traversal outside authorized directories.** Multiple File Manager actions constructed operation targets by concatenating request-controlled or paste-buffer names with $cwd without a common lexical current-directory and resolved allowed-path check. Historical source.

Files and paths · Module · LDAP Client (1)

- **WEBMIN-0278 — Delegated LDAP Client configuration can overwrite arbitrary files as root.** In the reviewed historical state, a Webmin user delegated the LDAP Client module can, unless noconfig is explicitly set, edit the tab-delimited secret-file setting and then submit chosen rootbindpw content. save_rootbinddn_secret replaces every selected path and sets it to root:root mode 0600. Historical source.

Files and paths · Module · Mailboxes (3)

- **WEBMIN-0150 — Bulk attachment download allowed MIME filename path traversal.** The reviewed bulk-download implementation decoded a message-supplied attachment filename and appended it directly to a privileged temporary-directory path. Slash and dot segments were not constrained, so a crafted attachment name could escape the request directory when an authorized reader selected Download All. Historical source.
- **WEBMIN-0998 — Bulk attachment staging accepted traversal filenames.** Bulk attachment download staged each message-supplied filename below a temporary directory using a direct file open. Slash traversal in a malicious attachment name could redirect the attachment bytes outside that directory with the Webmin process's filesystem authority when an authorized reader requested the ZIP. Historical source.
- **WEBMIN-1000 — Server-side attachment detachment accepted traversal filenames.** The server-side detach workflow combined an operator-selected directory with a decoded, message-controlled attachment filename. Slash traversal could direct the attachment bytes outside the selected directory. Historical source.

Files and paths · Module · Nginx (2)

- **WEBMIN-1001 — Raw configuration symlinks could separate authorization from the file written.** The raw Nginx configuration editor authorized a submitted pathname against its allowed list

and then followed symlinks for the write. A restricted module user could therefore reach a target whose resolved identity had not been compared with the authorized file set. Historical source.

- **WEBMIN-1003 — Unvalidated password-file path allowed access outside delegated roots.** The password-user form, list, and save CGI paths accepted a request-supplied file without applying the Nginx module's access.root restriction. Because htpasswd operations run as the configured write identity and that identity defaults to root, a restricted authenticated module user could read usernames or modify arbitrary accessible htpasswd-format files. Historical source.

Files and paths · Module · PostgreSQL (1)

- **WEBMIN-2459 — Compressed restore retains decompressed temporary SQL data.** The compressed-restore path creates a decompressed SQL tempfile and assigns it to $path. After restore_database returns, cleanup calls unlink(\$file)$, although $file is not assigned by restore.cgi. Historical source.

Files and paths · Module · Qmailadmin (1)

- **WEBMIN-0981 — Legacy qmailadmin save handlers permitted authenticated arbitrary file overwrite.** The before-side of reviewed commit 3934ae3e73b9cd154a850c7ca9fde278fceef80e shows executable legacy qmailadmin save CGIs passing the request-controlled file parameter directly to a truncating Perl open. old_save_rfile.cgi then writes the request-controlled text field, yielding a chosen-path, chosen-content overwrite with the CGI process's filesystem permissions. Historical source.

Files and paths · Module · Syslog (1)

- **WEBMIN-0786 — Restricted Syslog editors can make arbitrary privileged files world-readable.** In the reviewed historical Syslog save path, an authenticated module user with editing enabled and the syslog capability can select local-file mode and supply an arbitrary path. The handler opens that path and calls set_ownership_permissions with the configured owner/group and mode 0644 before can_edit_log checks the delegated logs allowlist. Historical source.

Files and paths · Module · Updown (2)

- **WEBMIN-0883 — Multipart filename traversal permits pre-authorization arbitrary file overwrite.** When direct upload mode is selected, ReadParseMime uses the multipart filename verbatim in *tmp*/file and opens that path with truncating mode. In Webmin, this parse and write occur before upload authorization, destination-directory ACL validation, and the selected-user UID switch. Historical source.
- **WEBMIN-0942 — Root direct-upload write can follow a fixed user's staging symlink.** In the reviewed commit state, ACL mode 3 selects the fixed Unix user's home .tmp directory as the direct multipart write directory while the CGI is still root. ReadParseMime takes the client filename's basename and opens that path with truncation before can_upload is checked. Historical source.

Files and paths · Module · Useradmin (2)

- **WEBMIN-0731 — Privileged SSH-key append can escape the selected home through symbolic links.** The reviewed new-user SSH-key workflow builds <home>/.ssh/authorized_keys and appends the submitted key through the privileged Webmin process. It checks that the selected home is ACL-permitted, but it neither rejects a symbolic-link .ssh directory nor a symbolic-link authorized_keys file; append mode opens the resolved target directly. Historical source.
- **WEBMIN-2452 — World-traversable defaults exposed new home directories.** Seven operating-system templates used 0755 as the default mode passed to create_home_directory for newly created home directories. That grants every local user directory read and execute permission, permitting listing and traversal and access to contained files that retain world-readable modes.

Files and paths · Module · Webalizer (1)

- **WEBMIN-0453 — Webalizer report viewer could disclose root-only files through report-owner-controlled paths.** In the reviewed historical state, view_log.cgi constructed a report asset path from the configured report directory and requested asset, then opened it directly with the Webmin CGI's

privileges. A non-root report owner able to place a file or symlink in that directory could make an authenticated viewer dereference a root-only target and return its contents. Historical source.

Information exposure and secrets (40)

Information exposure · Core · ACL (1)

- **WEBMIN-0395 — Certificate retrieval allowed arbitrary local-file disclosure and deletion.** The reviewed certificate-output diff shows that cert_output.cgi previously opened the request's file value, returned its bytes as a client certificate, and unlinked that same path without constraining it to Webmin's temporary directory. Historical source.

Information exposure · Core · Runtime (7)

- **WEBMIN-0179 — PAM challenge response can be disclosed as a failed username.** A later secret-bearing PAM response can flow into failed-login handling as the username and be written to audit, syslog, or external failure handlers. Historical source.
- **WEBMIN-0182 — Per-user Usermin configuration directories are locally traversable.** New per-user and module configuration directories use mode 0755, exposing directory metadata and traversal to other local accounts. Historical source.
- **WEBMIN-0187 — Rendered response capture is copied to the service error log.** Captured action output is echoed unconditionally to stderr, potentially copying sensitive rendered response bodies to a more broadly accessible or longer-lived log. Historical source.
- **WEBMIN-0418 — Rejected FTP login errors could disclose passwords.** The FTP error channel could retain the complete PASS command after rejected authentication. Callers that displayed or logged that error could disclose the supplied FTP password. Historical source.
- **WEBMIN-0703 — 2FA retry redirects exposed the submitted primary password in a URL.** When a primary password was valid but 2FA did not complete, Miniserv preserved the password as failed_pass in a query string for the login-page retry. URL encoding only protects query syntax; it does not prevent retention in browser history, request logs, diagnostic traces, or downstream URL processors. Historical source.
- **WEBMIN-0706 — 2FA validation logged the submitted token and provider API key.** The 2FA-file support path printed the complete wrapper invocation to STDERR, including the submitted one-time token and the provider API key. Service error logs and support captures could consequently disclose both transient and long-lived authentication material. Historical source.
- **WEBMIN-2397 — Session table retained replayable bearer IDs.** Active session IDs were retained directly as session-table keys. A process-memory, session-database, or diagnostic disclosure containing those keys would expose replayable bearer values rather than non-replayable lookup digests. Historical source.

Information exposure · Core · Cross-cutting (1)

- **WEBMIN-0383 — TOTP enrollment secret disclosed to an external QR service.** The reviewed TOTP enrollment rendering builds a Google Charts image URL containing the username and complete reusable TOTP seed. Rendering the enrollment page sends that secret to a third party and may place it in browser, proxy, or service URL logs. Historical source.

Information exposure · Core · Integration (5)

- **WEBMIN-0667 — Primary password copied into an internal CGI query-string environment.** On a two-factor authentication failure, the reviewed miniserv path copies the submitted primary password into an unencoded internal GET query string and exports that string as CGI QUERY_STRING. Historical source.
- **WEBMIN-0818 — Mailboxes image relay lacks non-public destination controls.** The authenticated Mailboxes XHR handler accepts a URL, resolves its host through the normal HTTP client, downloads the body, and relays the upstream content type and response without an address-policy callback. Historical source.

- **WEBMIN-0902 — External QR fallback discloses the new TOTP seed.** When local QR generation is unavailable, the enrollment page embeds an api.qrserver.com image whose query contains the complete otpauth URI and TOTP seed. Loading the page sends the second-factor secret to that external service. Historical source.
- **WEBMIN-0951 — Server-side downloads could reach non-public network services.** A download-capable account could make Webmin connect to attacker-selected HTTP, HTTPS, or FTP destinations without a non-public-address policy. This exposed loopback, link-local, private, and translated internal targets to SSRF. Historical source.
- **WEBMIN-2470 — Login redirect hook exposed plaintext Webmin passwords.** The optional post-login redirect hook interpolated the submitted plaintext password into a shell command. Configured deployments exposed that secret to the hook and potentially to local command-line observation.

Information exposure · Core · Packaging (1)

- **WEBMIN-0899 — RPM builder discloses environment secrets to build output.** The reviewed RPM builder unconditionally printed every process environment variable and value. When builds run in CI or signing environments, this copies credentials into logs that may have a broader retention period and readership than the original secret store. Historical source.

Information exposure · Core · Webmin (2)

- **WEBMIN-0640 — TOTP enrollment discloses the seed through an external QR image request.** Historical TOTP enrollment renderers sent the complete otpauth enrollment value, including the reusable TOTP seed, to third-party QR services in an image URL. Reviewed variants used chart.googleapis.com and, when local QR support was unavailable, api.qrserver.com. Historical source.
- **WEBMIN-0871 — TOTP enrollment secrets were placed in QR endpoint query strings.** The historical local QR flow placed the complete otpauth enrollment value, including the TOTP seed, in the request URL for qr.cgi. Request targets are routinely retained in Webmin, reverse-proxy, browser-history, and diagnostic records. Historical source.

Information exposure · Module · Ajaxterm (3)

- **WEBMIN-0194 — Authenticated Ajaxterm access exposes an unrestricted loopback HTTP proxy.** The Webmin-facing proxy parses any decimal TCP port from PATH_INFO, forwards the client-selected method, path, content metadata, and body to localhost, and relays the backend response. It does not verify that the port belongs to an Ajaxterm process started for the current page or user. Historical source.
- **WEBMIN-0544 — Local listener race can impersonate the Ajaxterm backend.** The reviewed startup code chooses a predictable high port by binding it, closes that socket, and only afterward forks and execs Ajaxterm. The parent then treats any successful connection to the port as proof that its child is ready and embeds that port in a same-origin Webmin proxy iframe. Historical source.
- **WEBMIN-2407 — GET terminal mode places sensitive keystrokes in request URLs.** Ajaxterm's browser client offers a GET toggle that appends buffered terminal input to the k query parameter. When enabled, all typed data becomes part of request URLs.

Information exposure · Module · Apache (1)

- **WEBMIN-0662 — Legacy virtual-host display exposes out-of-scope Apache configuration.** show_virt.cgi accepted a request-selected virtual-host index or name, loaded that configuration block, and rendered its directive names and values without calling can_edit_virt. An authenticated Apache-module user restricted to selected hosts by the virts ACL could directly request an excluded host and read its configuration. Historical source.

Information exposure · Module · Certmgr (3)

- **WEBMIN-0069 — Certificate Manager permits arbitrary privileged file operations.** The import handler accepts posted destination directory and filename strings for privileged writes, while the view handler accepts direct paths for plaintext download, export, and unlink. The handlers do not require

paths to remain within the configured SSL directory, so delegated page access becomes arbitrary host file access at Webmin's privilege level. Historical source.

- **WEBMIN-0412 — Certmgr view access exposes arbitrary root-level files.** view.cgi accepts keyfile, certfile, csrfile, or keycertfile directly from the request. Its download branch opens the supplied path, and its delete branch passes the path directly to unlink, without requiring that it remain below any configured certificate directory. Historical source.
- **WEBMIN-0579 — CRL download can disclose arbitrary root-readable files.** The historical CRL download branch accepts a direct crlfile request value and passes it to output_cert, which opens that path and streams its bytes to the response. The code does not bind the path to ssl_dir or enforce an allowed file type before reading. Historical source.

Information exposure · Module · Cron (1)

- **WEBMIN-0063 — Delegated Cron user can clone and read another user's job.** The audited clone workflow reads from the module-wide job list and copies the selected source into the new-job form without applying the source user's ACL check. An authenticated delegated Cron-module user can therefore select a job belonging to a disallowed Unix user and receive its command, input, comment, and schedule. Historical source.

Information exposure · Module · Documentation (1)

- **WEBMIN-0560 — Manual-page path input exposes readable local files.** The historical Manual Pages handler allowed an authenticated module user to supply path-shaped page input to configured manual-page commands. This could disclose local files readable in the command's operating-system context; required authentication, module access, command acceptance, and filesystem permissions limit the impact. Historical source.

Information exposure · Module · Fsdump (1)

- **WEBMIN-0285 — Remote-backup password could be disclosed through echoed PTY output.** In the audited preimage, run_ssh_command wrote the configured remote-backup password to a PTY and forwarded captured PTY text to foreground browser output or scheduled backup-result email without removing that secret. proc::pty_process_exec attempted to disable echo only when optional IO::Stty loaded, so the source did not guarantee that the password would stay out of the captured stream. Historical source.

Information exposure · Module · Itsecur Firewall (1)

- **WEBMIN-0289 — Authentication logs are exposed through firewall-log ACL checks.** The authentication-log page is protected by the dedicated authlogs ACL feature, but its direct download and streaming handlers check the ordinary logs feature. A Webmin user granted firewall-log access but denied authentication-log access can call those handlers directly and retrieve the configured authentication log outside the intended feature boundary. Historical source.

Information exposure · Module · LDAP Client (1)

- **WEBMIN-0133 — LDAP user administration can send bind credentials without configured StartTLS.** The reviewed ldap-useradmin connector parses the NSS LDAP configuration into a hash keyed by directive name, so a normal `ssl start_tls` line becomes ssl => start_tls. Historical source.

Information exposure · Module · LDAP Server (1)

- **WEBMIN-0101 — StartTLS control flow can expose LDAP administrator credentials.** The historical connection routine interprets the Net::LDAP StartTLS result backwards: failed negotiation advances through the mode loop, while a successful response returns an error. The subsequent bind uses the configured LDAP administrator DN and password. Historical source.

Information exposure · Module · MySQL (2)

- **WEBMIN-0047 — Historical root-password form disclosed the configured database credential.** The reviewed root-password form sends the configured MySQL administration password back to an authenticated browser as clear text. Historical source.
- **WEBMIN-0788 — MySQL administration password could leak through process listings.** supports_env_pass used find to obtain a password directive object and compared that object with the requested password. When a readable client section contained a password, the false mismatch disabled MYSQL_PWD transport and make_authstr embedded the requested secret in a --password argument. Historical source.

Information exposure · Module · Pptp Client (1)

- **WEBMIN-0123 — Historical PPTP connect password disclosure through GET submission.** The optional connect-time PPTP password introduced by commit 67fa6cacfbda2190ae1bf46befdae9769b684d21 was rendered in a form without a method attribute. Browsers therefore sent cpass in the conn.cgi query string. Historical source.

Information exposure · Module · Project Meta (2)

- **WEBMIN-0836 — Repository-signing passphrase is exposed in SSH process arguments.** The WEBMIN_DEV___GPG_PH secret is copied into cloud_upload_gpg_passphrase, concatenated into the remote shell command, and passed as an ssh process argument via eval. Redirecting output does not hide process arguments. Historical source.
- **WEBMIN-0873 — Unauthenticated SSH upload channel exposes a signing passphrase to endpoint impersonation.** The historical build upload helper disables SSH server identity checks with StrictHostKeyChecking=no and an empty known-hosts file. It then constructs a remote signing command containing the GPG passphrase and executes that command over the unauthenticated SSH channel. Historical source.

Information exposure · Module · Status (1)

- **WEBMIN-0700 — Forced Status refresh passed CGI environment state to privileged monitor execution.** The historical forced-refresh paths invoked the monitor wrapper from a Webmin CGI process without first clearing miniserv request state. Request-derived HTTP variables and sensitive CGI metadata therefore remained in %ENV for the privileged monitor process and any check programs it launched. Historical source.

Information exposure · Module · Vnc (2)

- **WEBMIN-0083 — VNC desktop traffic is sent over a plain connection.** The bundled viewer uses a plain TCP socket for VNC protocol streams, and the module itself identifies the session as unencrypted. A network observer on an unprotected path can see screen content and input even when the Webmin page uses HTTPS. Historical source.
- **WEBMIN-0084 — Host header steers server-side VNC port probe.** When no VNC host is configured, the CGI uses SERVER_NAME as its TCP connection target and exposes failure versus success. Miniserv derives that value from the HTTP Host header unless a musthost policy rejects it. Historical source.

Information exposure · Module · Xterm (2)

- **WEBMIN-0867 — Session-bearing values were written to terminal diagnostic files.** Two reviewed diagnostic paths persisted session-bearing data. The WebSocket mismatch log interpolated both the received key and the expected base64 session identifier into a per-port log, and an environment debug call serialized the complete CGI environment immediately after SESSION_ID was assigned. Historical source.
- **WEBMIN-0958 — Websocket authentication mismatch log disclosed session-bearing secrets.** On a websocket handshake mismatch, the historical code shown by the reviewed diff logged the supplied key and the base64-encoded Webmin session ID. Both values participate in terminal websocket authentication,

so persistent logs unnecessarily exposed credentials to anyone able to read connection logs. Historical source.

Authentication and session handling (27)

Authentication · Core · ACL (1)

- **WEBMIN-0509 — Routine account edits could silently disable per-user two-factor enforcement.** The historical existing-user save path constructed a replacement object from form inputs and selected retained fields, then modify_user rewrote the complete user record. Historical source.

Authentication · Core · Runtime (11)

- **WEBMIN-0010 — Static unauthenticated paths could enter CGI execution.** The reviewed request path treated a configured static unauthenticated namespace as ordinary anonymous validation, so a matching resource that resolved as internal/cgi could enter CGI dispatch. A remote unauthenticated requester could execute that CGI with Webmin's service privileges. Historical source.
- **WEBMIN-0183 — RBAC deny policy is looked up under the wrong identity.** The foreign-module gate can fail to consult the authenticated base user's RBAC deny entry and therefore allow a module contrary to explicit policy. Historical source.
- **WEBMIN-0186 — Fixed-password verifier accepts any non-empty computed hash.** The fixed-password login path checks only whether password_crypt returns a truthy string, so an incorrect password can authenticate without matching the stored verifier. Historical source.
- **WEBMIN-0416 — Mapped-user authentication could omit required two-factor validation.** For sudo-capable username mappings, password validation could return an effective identity different from the underlying Webmin account. Two-factor lookup used the wrong identity, so the protected account's second-factor requirement could be omitted. Historical source.
- **WEBMIN-0422 — Executable link.cgi inherited unauthenticated static-resource matches.** The unauthenticated-path loop included a special case that tested the suffix of /servers/link.cgi against static unauthenticated resource patterns. A matching suffix could therefore approve execution of link.cgi without normal authentication. Historical source.
- **WEBMIN-0828 — Proxy client-certificate identity lacked verification proof.** When proxy address trust was enabled, the reviewed authentication path could map X-SSL-Client-DN directly to a Webmin certificate user, including on a non-TLS MiniServ connection. It did not require an upstream verification-success assertion or prove that the direct peer was an approved proxy. Historical source.
- **WEBMIN-0932 — Forwarded client certificate accepted without positive verification.** When trusted-proxy certificate authentication was enabled, the request predicate consulted a misspelled verification header and rejected only explicit failure-like values. Historical source.
- **WEBMIN-0933 — Forwarding headers influenced URL and WebSocket origin decisions without a trust gate.** Browser URL construction and the WebSocket allowed-origin set consumed Forwarded and X-Forwarded components regardless of the configured reverse-proxy trust setting. A directly connected requester able to supply those headers could influence generated scheme/host values and add a chosen origin to WebSocket validation. Historical source.
- **WEBMIN-0935 — Keep-alive backend connections retained authentication state across requests.** Miniserv reused process globals while serving multiple HTTP requests on one keep-alive connection. Historical source.
- **WEBMIN-0940 — Configured password helper could accept a new password without old-password confirmation.** In configured passwd_cmd mode, the conversation driver sent the new password whenever the helper requested it, even if the helper never requested the old/current password. A helper with that prompt order could therefore change a password without the Webmin flow establishing knowledge of the existing password. Historical source.
- **WEBMIN-2396 — Referer warning offered a parameter-replaying continuation.** The referer defense rendered a warning page that serialized the rejected request parameters into a form and allowed the authenticated user to continue, with an option to disable future checks. This reduced cross-site request protection to a click-through and could complete a crafted state-changing request after social interaction. Historical source.

Authentication · Core · Cross-cutting (3)

- **WEBMIN-0001 — Shell command endpoint was exempt from the referer gate.** In the reviewed historical source, shell/index.cgi parses a command-bearing non-GET request and executes the cmd value for the Unix identity authorized by the shell module. The shared referer rejection condition exempted module index paths, so this command-capable index did not receive that cross-site request defense. Historical source.
- **WEBMIN-0002 — Temporary-password enforcement depended on password-aging configuration.** The reviewed temporary-password implementation checks the temppass flag only inside the password-aging branch. When pass_maxdays is disabled, last-change metadata is missing, or nochange is set, successful authentication skips the forced-change return and receives a normal session. Historical source.
- **WEBMIN-0776 — Historical terminal proxy accepted cross-origin authenticated WebSockets.** In the reviewed historical snapshot, xterm creates a predictable low-sequential-port WebSocket proxy path bound to the authenticated username. The handshake checks that username and then connects to the shell backend, but it does not validate the browser Origin or require a per-route secret. Historical source.

Authentication · Core · Integration (1)

- **WEBMIN-0819 — Forwarded client-certificate DN can override an unmatched verified certificate.** In a configuration that enables client-certificate authentication and trusts forwarded client information, MiniServ first attempts to map the directly verified certificate. If that certificate is valid but unmapped, the reviewed fallback maps X-SSL-Client-DN from the request without proxy verification and marks a matching account certificate-authenticated. Historical source.

Authentication · Module · Ajaxterm (1)

- **WEBMIN-0654 — Cross-site Ajaxterm requests could inject input into a root shell.** Before the reviewed security restoration, ajaxterm/proxy.cgi skipped init_config() and immediately forwarded an attacker-selected request path and POST body to a numeric localhost port. The Ajaxterm browser protocol places encoded keystrokes in the k form field, and the local service writes that field to its terminal process. Historical source.

Authentication · Module · Images (1)

- **WEBMIN-0100 — Blue-theme logo opens an unprotected HTTP destination with an opener.** The blue-theme logo links to http://www.webmin.com/ using target=_new without rel=noopener. If a user clicks it and an attacker can tamper with the unencrypted destination response before any HTTPS upgrade, the attacker can present a counterfeit site. Historical source.

Authentication · Module · Minecraft (1)

- **WEBMIN-0370 — Unauthenticated metadata can select executable server code.** The reviewed download logic obtains the executable JAR URL from a plain-HTTP metadata page and accepts either HTTP or HTTPS links from that response. The selected artifact is installed based on a checksum difference but has no trusted signature or pinned digest. Historical source.

Authentication · Module · Passwd (1)

- **WEBMIN-0735 — Password API can substitute an empty value for the requested new password.** In the fa5fc3a94b6889ad4aa04ad57beb9014da115fc2 snapshot, change_passwd.cgi checks that `new` is non-empty but passes the different unset `pass` field to password-policy validation and change_password. Historical source.

Authentication · Module · SSH Server (1)

- **WEBMIN-0322 — SSH hardening directives could be scoped to the wrong Match block.** When an SSH directive was absent, the historical save_directive helper searched every Match entry and inserted the new line immediately before the last one. In a configuration with multiple Match blocks, that location is inside the preceding conditional block rather than in the global section. Historical source.

Authentication · Module · Useradmin (1)

- **WEBMIN-0390 — Empty validator output converted rejection into success.** The external password-checking path returned only captured command output when the checker exited non-zero. If the checker rejected silently, the empty return value was false to save_user.cgi, which then continued password processing rather than enforcing the configured validator's rejection. Historical source.

Authentication · Module · Xterm (5)

- **WEBMIN-0800 — Predictable tokenless terminal routes enabled cross-site WebSocket hijacking.** The xterm route allocator selected sequential ports beginning at 555 and registered /xterm/ws-<port> without a route nonce. miniserv authenticated the browser by its Webmin session and matched the route user, but it neither validated the WebSocket Origin nor required a per-route token. The historical session cookie did not carry SameSite. Historical source.
- **WEBMIN-0866 — Terminal shell WebSocket backend accepted network-wide connections.** The reviewed shell server supplied only a numeric port to the bundled Net::WebSocket::Server. That library created IO::Socket::INET without LocalAddr, exposing the transient privileged-shell backend beyond loopback. Historical source.
- **WEBMIN-0927 — Public xterm backend listener permits unauthenticated terminal-session denial of service.** Before the reviewed change, xterm's shell backend passed only an allocated port to Net::WebSocket::Server. The bundled server consequently created an IO::Socket::INET listener without a LocalAddr restriction. Historical source.
- **WEBMIN-0960 — Browser session identifier was duplicated into websocket route configuration.** The historical behavior visible in the reviewed diff passed the active Webmin browser session value as backend_session when allocating an xterm websocket, causing it to be serialized into miniserv route configuration. This unnecessarily expanded the local persistence footprint of a live credential. Historical source.
- **WEBMIN-2472 — Terminal index lacked referer enforcement despite launching a shell backend.** The historical behavior visible in the reviewed change treated xterm/index.cgi like a normally safe module index even though a request allocates a websocket route and launches shellserver. A cross-site request made under an authenticated victim session could therefore trigger terminal backend side effects. Historical source.

Transport and cryptography (14)

Transport · Core · Runtime (2)

- **WEBMIN-0419 — SSLv3 remained enabled despite the configured prohibition.** The no_ssl3 configuration was not applied to the Net::SSLeay server context. A deployment could therefore advertise that SSLv3 was disabled while the protocol remained negotiable in supporting TLS stacks. Historical source.
- **WEBMIN-2446 — Source-distributed TLS private key cannot identify installations.** The default miniserv.pem includes static private-key material in the source tree. A deployment retaining it shares a publicly recoverable identity, enabling impersonation or interception where that certificate is accepted.

Transport · Core · Integration (2)

- **WEBMIN-0332 — Cleartext Minecraft metadata enables executable JAR substitution.** The reviewed Minecraft download flow obtains trusted download metadata over cleartext HTTP, extracts an arbitrary matching HTTP/HTTPS server-JAR URL from that response, downloads the artifact without a signature or trusted digest, and places it at the executable server-JAR path. Historical source.
- **WEBMIN-0781 — Bundled fallback TLS private key removes per-installation key uniqueness.** The reviewed historical setup behavior deploys reusable bundled certificate and private-key material when local certificate generation fails. An installation taking that fallback can share its Webmin server private key with other copies and with anyone able to obtain the bundled material, undermining endpoint authentication and transport confidentiality. Historical source.

Transport · Core · Packaging (1)

- **WEBMIN-2450 — Historical installer generated a 512-bit RSA TLS private key.** The reviewed setup.pl diff shows the certificate-generation command using rsa:512 before changing that command to a stronger key size. A 512-bit RSA modulus is below acceptable modern security strength and can be factored in practice; a retained key could allow recovery of the private key and TLS endpoint impersonation.

Transport · Core · Webmin (1)

- **WEBMIN-0477 — Built-in ACME private-key copies lacked explicit 0600 modes.** The built-in ACME path generated a domain private key and copied it into the module configuration directory without explicitly setting a restrictive per-file mode. Security therefore depended on inherited umask and directory traversal protections. Historical source.

Transport · Module · Certmgr (1)

- **WEBMIN-2400 — Generated RSA keys can be cryptographically weak.** Both certificate and CSR generation present 512-bit RSA as a valid choice and select 1024-bit RSA by default. These sizes do not meet contemporary security requirements, making generated credentials susceptible to key recovery or otherwise unsuitable for security-sensitive use. Historical source.

Transport · Module · Documentation (1)

- **WEBMIN-0655 — Installation and security guidance linked over unencrypted HTTP.** The authored README directs administrators to HTTP download and setup guidance, and the 1.930 notice links to an HTTP security advisory. Historical source.

Transport · Module · Dovecot (2)

- **WEBMIN-0699 — Inverted version gating could misapply Dovecot TLS configuration.** The reviewed version_atleast implementation previously placed the requested minimum version on the left side of compare_version_numbers and the detected Dovecot version on the right. That reverses the intended minimum-version predicate. Historical source.
- **WEBMIN-2408 — Dovecot TLS enablement choice could be saved with opposite polarity.** The historical Dovecot networking form used the inverse values appropriate to `ssl_disable` even when the active directive was modern `ssl`. Consequently, selecting Yes could submit `no` and disable intended TLS.

Transport · Module · Htpasswd File (1)

- **WEBMIN-0804 — Default htpasswd Crypt path hard-coded traditional DES password hashes.** In the before-state shown by the reviewed commit, the htpasswd-file module's shipped md5=0 Crypt path generated only two uppercase salt characters and invoked unix_crypt directly for newly supplied passwords on non-Windows hosts. Historical source.

Transport · Module · Project Meta (1)

- **WEBMIN-0835 — Package upload and signing do not authenticate the SSH server.** The upload and signing helpers pass StrictHostKeyChecking=no and UserKnownHostsFile=/dev/null to scp and ssh. The client therefore accepts an unrecognized substitute server key instead of authenticating the deployment endpoint, permitting an active network-path attacker to divert uploads and receive the remote command text. Historical source.

Transport · Module · SSH Server (1)

- **WEBMIN-2442 — Old OpenSSH absent-directive form state selected SSH1 only.** In the reviewed protocol-form logic, an absent Protocol directive caused the detected OpenSSH 2.0-through-2.8 branch to select only protocol 1. Because the value is rendered as the editable form state, an authorized operator could carry that legacy-only selection into a subsequent save. Historical source.

Transport · Module · Translations (1)

- **WEBMIN-0137 — Russian update instructions point to an unencrypted alternate site.** The change authored by Jamie Cameron replaces links in Russian Webmin update and upgrade instructions with HTTP links to cpplus.info and labels the product ZTL. An administrator who follows these links and installs a package from the site or a response modified in transit may run untrusted code. Historical source.

Request forgery and origin checks (3)

Request forgery · Module · Ajaxterm (1)

- **WEBMIN-0543 — Authenticated arbitrary loopback HTTP proxy in Ajaxterm.** The reviewed proxy parses an arbitrary decimal port and downstream path from PATH_INFO, then forwards the caller's HTTP method and optional body to localhost at that port and returns the response. It never verifies that the port was allocated by index.cgi, belongs to the requesting Ajaxterm instance, or is even an Ajaxterm listener. Historical source.

Request forgery · Module · Flashterm (1)

- **WEBMIN-0253 — Authenticated Host header controls a server-side Telnet-port connection.** For an authenticated principal authorized to open FlashTerm, index.cgi trusts HTTP_HOST as the destination hostname. Historical Miniserv accepts the request Host header when the optional musthost restriction is unset and exports it as HTTP_HOST. Historical source.

Request forgery · Module · Ssh2 (1)

- **WEBMIN-0087 — Optional SSH2 server test permits request-selected TCP destinations.** When the optional connectivity test is enabled without a configured host, the CGI uses SERVER_NAME from the incoming Host header to choose the address, connects from the server to the configured port (22 by default), and changes its response according to success. Historical source.

Availability and race conditions (8)

Availability · Core · Runtime (1)

- **WEBMIN-0938 — Idle pre-TLS connections could retain server children indefinitely.** After accepting an SSL-enabled connection, the server peeked for the first byte without a readiness deadline. An unauthenticated remote client could connect and send nothing, retaining a child process. Historical source.

Availability · Module · Dhcp Dns (1)

- **WEBMIN-0122 — IP chooser exhausts resources by materializing unbounded subnet address space.** The popup free-IP chooser walks every integer address in every configured subnet, inserts all candidates into an in-memory hash, builds a second available-address list, and sorts it before responding. There is no subnet-size cap, request budget, pagination, or streaming. Historical source.

Availability · Module · Itsecur Firewall (1)

- **WEBMIN-0290 — Broad CIDR report filters cause uncontrolled resource consumption.** Report filtering accepts a user-selected source or destination network and expands a CIDR by iterating from its first address to its last, appending every address to an in-memory array. There is no prefix-size, iteration, time, or memory bound. Historical source.

Availability · Module · Procmail (1)

- **WEBMIN-0659 — Authenticated malformed Procmail configuration causes parser resource exhaustion.** The historical Procmail parser repeatedly reads past EOF when a configuration ends inside a double-quoted environment assignment. Historical source.

Availability · Module · Proftpd (1)

- **WEBMIN-0890 — Unconditional shared-library refresh enables authenticated resource amplification.** At the reviewed snapshot, proftpd-lib.pl declares %oldsite but loads the existing cache into %site, then tests the unpopulated %oldsite. The missing version fields make the refresh condition true on every library load. Historical source.

Availability · Module · Servers (1)

- **WEBMIN-0733 — Zero-minute discovery interval can cause authenticated CPU exhaustion.** The reviewed scheduled-discovery save path derives a cron minute list with for(my $i=0; $i<60; $i+ =in{'mins'}). The diff demonstrates that digits-only validation admitted zero, for which i never advances and the request remains in the loop. Historical source.

Availability · Module · Shell (1)

- **WEBMIN-2423 — Line-oriented capture can allocate beyond the configured output limit.** In the reviewed line-oriented output-capture state, Perl reads a complete record from the command pipe before applying max_output. If the first record is much larger than the configured threshold, the process materializes that record and appends it because the retained buffer was still below the threshold. Historical source.

Availability · Module · Updown (1)

- **WEBMIN-2422 — Historical download streaming permits authenticated memory exhaustion.** In the parent side of the reviewed historical change, both directory-archive and ordinary-file responses stream with scalar <FILE> reads. Perl scalar readline consumes through the next input record separator, so a large newline-free file can be materialized as one scalar before output. Historical source.

Other security defects (36)

Other · Core · ACL (2)

- **WEBMIN-0079 — User IP deny rules are serialized as allow rules.** The user editor assigns the same submitted value to its allow-list and deny-list radio choices. The save handler interprets that value only as allow mode, so an administrator selecting the deny label stores the listed addresses as the exclusive allowed set. Historical source.
- **WEBMIN-0080 — Existing-user updates corrupt password expiry and reuse state.** The existing-user serializer omits the colon between allowed-time data and the password last-change timestamp. The parser expects last-change and old-password hashes in separate colon-delimited fields, so a routine user update can shift or erase this security state. Historical source.

Other · Core · Runtime (2)

- **WEBMIN-0268 — IPv6 CIDR access rules are mis-evaluated as IPv4 networks.** miniserv's IP gate accepts any address text in its first CIDR expression, converts the prefix through an IPv4-only dotted-mask helper, and then applies the IPv4 network loop. That branch consumes IPv6 CIDR rules before the later IPv6-network branch, which is left unimplemented. Historical source.
- **WEBMIN-0794 — Proxy-supplied webprefix lacked input filtering.** When remote proxy-prefix mode was enabled, the request-supplied X-Webmin-Webprefix value was copied directly into the global webprefix used throughout link and attribute generation. A client able to supply that header without a trusted proxy overwriting it could inject URL or HTML metacharacters into privileged Webmin pages. Historical source.

Other · Core · Webmin (1)

- **WEBMIN-0255 — IPv4 CIDR /9-/15 access rules widen to /8.** The reviewed IPv4 CIDR conversion helper has two consecutive >=16 branches. Prefixes /9 through /15 take the final branch, where the computed first-octet mask is fractional and later integer coercion yields 255, so the effective comparison covers the whole /8. Historical source.

Other · Module · Apache (1)

- **WEBMIN-2426 — Generated virtual hosts enabled directory indexing by default.** The reviewed virtual-host generator emitted Options +Indexes in a newly created Directory section. On an Apache deployment with directory indexing available and no index document, that default can disclose filenames and downloadable content under the document root. Historical source.

Other · Module · Fail2Ban (3)

- **WEBMIN-2469 — Webmin could represent custom Fail2Ban jail overrides with stale precedence.** Before the audited jail-merge change, list_jails appended jail.d/*.conf sections directly to the base array while merge_local_files applied later overlays only to the first same-named section.
- **WEBMIN-2475 — Jail editor round trips could corrupt defensive action definitions.** Two action parser conditions in the reviewed history made the Fail2Ban jail editor unable to preserve some valid action definitions: delimiter splitting fragmented quoted option values, and a greedy bracket expression collapsed a parameterless action with the parameterized action after it.
- **WEBMIN-2476 — Default jail saves could discard explicit defensive timing policy.** The former default-jail editor represented maxretry, findtime, and bantime as optional values, and the matching save path removed a timing directive when its default marker was submitted. That behavior could silently replace the concrete 3/600/600 policy shown by Webmin with package-dependent implicit behavior. Historical source.

Other · Module · Filemin (1)

- **WEBMIN-2440 — Allowed-path listing used unquoted pathname regexes.** The directory-list filter interpolated filesystem entry names and configured path prefixes as Perl regular expressions. Metacharacters could make a disallowed sibling entry satisfy the ancestor predicate and appear with listing metadata. Historical source.

Other · Module · Filter (1)

- **WEBMIN-2415 — Bounce-suppressed forwarding rule could remain active after UI disable or replacement.** In the reviewed historical code, list_filters removed a leading bounce-suppression condition from its normalized @conds array but classified nocond using the raw recipe conditions.

Other · Module · Firewall (10)

- **WEBMIN-0162 — Automatic hosting profile can generate invalid multiport firewall controls.** The automatic hosting-server profile contains snapshots in which comma-separated mail service ports are represented as singular dport values without the multiport matcher, and a related snapshot contains unterminated multiport module arrays. These concrete representations cannot reliably yield the intended iptables rules. Historical source.
- **WEBMIN-0163 — Negated multiport rules can be saved with positive match semantics.** When a comma-separated port condition is converted from the singular sport/dport representation to multiport sports/dports, failing to copy array element zero discards the explicit negation mode. The saved rule then matches the listed ports rather than ports outside the list. Historical source.
- **WEBMIN-0337 — Automatic firewall setup can lose its ingress policy when the filter table is missing.** In automatic blocking modes, setup.cgi searches for the filter table. If that table is absent, the historical path creates an object named nat, then assigns it an INPUT DROP default and the intended inbound allow rules. save_table renders the object's name as the iptables-save table header and appends or restores the block. Historical source.
- **WEBMIN-0389 — Firewall rule predicates can be stripped by mismatched condition-mode form fields.** In the reviewed historical snapshot, the rule editor's print_mode helper declares but does not assign its formal arguments and renders selectors without the `${name}_mode` names expected by save_rule.cgi. The save-side parse_mode helper treats a missing mode as numeric zero and deletes the corresponding match. Historical source.
- **WEBMIN-0601 — IPv6 firewall rules could fail to persist because boot restore used the IPv4 save file.** The Debian network-interface startup path compared and wrote the IPv4-only $iptables_save_file even when configuring the IPv6 firewall. Historical source.

- **WEBMIN-0602 — Netplan systems could lose Webmin-managed firewall enforcement after reboot.** The Debian startup-method selection treated Netplan systems as if legacy ifupdown configuration controlled interface activation. Enabling the firewall at boot could therefore write restore directives to an inactive mechanism. Historical source.
- **WEBMIN-0752 — Generic boot integration can omit firewall restoration.** In the reviewed generic boot-integration path, create_webmin_init calls a helper name that is not defined; the available helper differs by one character. On systems that rely on this fallback, enabling firewall restoration can terminate before the init action is created. Historical source.
- **WEBMIN-0753 — Debian persistence argument mismatch can omit the restore hook.** The Debian boot-persistence path receives an interface descriptor whose final fields are configuration file and source line, while net::modify_interface_def expects mode and configuration file. Passing the descriptor without remapping can treat the file path as a delete-mode flag and the line number as the output target. Historical source.
- **WEBMIN-0754 — Automatic ingress baseline can target an incompatible table.** Automatic firewall modes 2 and above create a default DROP policy for INPUT plus a set of ingress allow rules. When no filter table is loaded, a fallback named nat is incompatible with those INPUT filtering semantics. Historical source.
- **WEBMIN-2404 — Invalid negated limit choices can prevent intended rate controls.** The rule editor offered an 'Above' mode for packet-rate and burst-rate limits even though that negated form is invalid for the generated rule. Applying it can cause rule-load failure or leave an intended throttling control absent, which is a security-control correctness problem for anti-abuse and rate-limiting policies.

Other · Module · Firewall6 (1)

- **WEBMIN-0592 — IPv6 lifecycle consolidation can operate on IPv4 service state.** The consolidated distro lifecycle code selected IP-family state too late or retained an IPv4-specific save-file variable. On Coherent/Red Hat style systems the IPv6 module could bind its stored init-script path to the IPv4 iptables service; on the Debian network-interface boot path it could generate ip6tables-restore against the IPv4 rules file. Historical source.

Other · Module · Fsdump (1)

- **WEBMIN-0125 — Scheduled filesystem-backup program cannot compile.** The reviewed scheduled-backup program contains two adjacent boolean AND operators in its post-command failure condition. Perl rejects the exact manifest blob before execution, so scheduled backups using that program cannot run and expected recovery data may not be produced. Historical source.

Other · Module · Ipfilter (1)

- **WEBMIN-0009 — IPFilter port-range UI generated the inverse firewall predicate.** The historical IPFilter rule editor described its range choice as matching ports between two endpoints, but its parser serialized the <> not-between comparator. Applying a rule created through that option therefore targeted the complementary port set. Historical source.

Other · Module · Itsecur Firewall (2)

- **WEBMIN-0439 — Secondary NAT saves can silently replace internal mapping targets.** The secondary NAT editor renders both an existing map's external and internal address fields from the external value. Its save handler later treats the two submitted fields as separate endpoints and persists them. Historical source.
- **WEBMIN-0440 — Firewall user-access administration cannot reliably identify or update principals.** The module-local user list combines all principals into one malformed row, while the editor submits an empty old identity and displays inverted or unconditional access-state selections. The save handler relies on that old identity to select the account. Historical source.

Other · Module · Nftables (1)

- **WEBMIN-1032 — Legacy migration can silently deactivate intended nftables policy.** The historical module-install migration reads private nftables tables, silently omits any table whose family/name

key already exists in the service-loaded destination, writes remaining tables without checking that nft can load the combined result, and then retires the private boot path. Historical source.

Other · Module · Passwd (1)

- **WEBMIN-2428 — Mode-5 ACL editor can clear the effective allowed-group list.** At the reviewed historical state, the mode-5 passwd ACL form populates its group selector from the 'groups' option, while the save routine stores the submitted group list in 'users' and the authorization predicate reads 'users'. Historical source.

Other · Module · Samba (1)

- **WEBMIN-0310 — Windows networking form fails to preserve the configured Samba authentication mode.** The reviewed Windows networking UI passes an uninitialized $security variable to its selector and lists passwd rather than Samba's server mode. The unchanged save handler writes the submitted selection directly. Historical source.

Other · Module · Sentry (1)

- **WEBMIN-0131 — Czech configuration can start Portsentry without the intended UDP mode.** The Czech configuration metadata supplies TCP mode tokens for the UDP mode choices. If an administrator selects one and uses the generated default startup command, Webmin passes that TCP token in the UDP daemon position, so intended UDP monitoring may not start. Historical source.

Other · Module · Servers (1)

- **WEBMIN-2444 — Hostname-only status RPC can cross registered-server authorization identity.** The edit page authorizes a server object loaded by its unique ID. In the hostname-only status-call form exposed by the reviewed diff, remote_rpc_call discards that identity and looks up a cache entry keyed by hostname.

Other · Module · Software (1)

- **WEBMIN-0550 — Incorrect package-version ordering can hide security updates.** In the first manifest source state, Webmin compares number-string-number package versions through its generic numeric-plus-string branch before reaching the specialized trailing-number logic. The resulting lexicographic suffix comparison ranks versions such as 4ubuntu8 above 4ubuntu10. Historical source.

Other · Module · SSH Server (1)

- **WEBMIN-0147 — Public-key authentication disablement is rendered as enabled.** The historical users-and-groups form reads PubkeyAuthentication into $rsa but renders its dsa radio from undefined $dsa. The radio therefore defaults to enabled even when the daemon directive is explicitly no, and the save handler writes that displayed dsa choice back to PubkeyAuthentication. Historical source.

Other · Module · Translations (1)

- **WEBMIN-0303 — German quota exclusion labels can mislead delegated access configuration.** The German quota catalog labels mode 2 as “all excluded users” and “all excluded groups”, rather than “all except the listed users or groups”. An administrator who reads the listed subjects as the permitted set can instead delegate quota administration for all other users or groups. Historical source.

Other · Module · Usermin (1)

- **WEBMIN-2419 — Usermin SSL form can silently clear server cipher preference.** The new Usermin SSL control persists its submitted value as ssl_honorcipherorder, and miniserv consults that same key to enable server cipher preference. The edit form instead chooses the radio state from no_honorcipherorder, which is not written by this path.

Other · Module · Webmincron (1)

- **WEBMIN-2465 — Generated Webmin cron jobs can be invisible to digit-only loaders.** At the reviewed historical snapshot, save_webmin_cron generated underscore-separated filename stems, while list_webmin_crons and the miniserv loader both accepted only digit-only .cron stems. The file write succeeds and requests a reload, but the scheduled function is excluded from the collection used for visibility and execution. Historical source.

Security hardening observations without demonstrated impact (95)

These records identify concrete hardening opportunities from hypotheses whose available evidence showed no security impact. They are design and maintenance candidates for preventing future defects.

Other security defects (95)

Core · Command-line utilities (2)

- **WEBMIN-1967 — Caller-selected configuration reaches code loading and shell-form service commands.** The reviewed command is a local administrative CLI and the manifest changes supply no setuid behavior, privileged wrapper, sudo rule, or other authority boundary. A caller that can invoke the command directly already executes programs with the same OS identity. Reviewed source.
- **WEBMIN-2217 — Config directory is interpolated into a restart system call.** No lesser-privileged or remote source for --config was identified: it is an explicit local path selected by the same root operator who already has unrestricted command execution and whose configuration is loaded before the restart. This is a hardening concern, not a distinct security boundary violation. Reviewed source.

Core · Runtime (8)

- **WEBMIN-1305 — Malformed crypt salt handling.** This is robustness hardening without evidence that malformed salt input permits an incorrect password to authenticate. Reviewed source.
- **WEBMIN-1527 — Absence of a same-origin script CSP.** The policy is valuable defense in depth, but its addition does not by itself prove a distinct vulnerability in the changed server code. Reviewed source.
- **WEBMIN-1528 — TRACE was not explicitly rejected.** The change is conservative method filtering; no reflection primitive or security impact was demonstrated. Reviewed source.
- **WEBMIN-1773 — NUL byte accepted in parsed request parameters.** The change is prudent defense in depth, but this review does not contain a concrete downstream truncation sink or demonstrated security consequence. Reviewed source.
- **WEBMIN-1989 — WebSocket dispatch lacked an explicit base-user condition.** Ordinary paths already pass the preceding authentication flow. Only deliberately configured unauthenticated paths have no base user, so the patch is defense-in-depth rather than evidence of a generic bypass. Reviewed source.
- **WEBMIN-2304 — Predictable var_dump path under /tmp.** No production call site exists in the frozen tree, the helper is documented for internal debugging, and the review supplies no unprivileged route that invokes it. The concern is retained as defense-in-depth rather than a supported finding. Reviewed source.
- **WEBMIN-2306 — Session lookup keys used an unkeyed digest fallback.** Session identifiers are already generated as high-entropy bearer values, and read access to the local database is a separate privileged boundary. HMAC and tighter file modes are meaningful defense-in-depth, but this review does not show a remote exploit from the fallback alone. Reviewed source.
- **WEBMIN-2307 — ACL restoration followed filesystem links.** Physical restore is prudent hardening, but the reviewed patch does not establish attacker control over a relevant restore path or race window. No supported end-to-end boundary violation was found in this review. Reviewed source.

Core · Cross-cutting (2)

- **WEBMIN-1359 — Optional TLS compression remains configurable.** This is a defense-in-depth option. The patch alone does not establish the secret-bearing compression oracle and attacker-controlled request pattern needed for a product vulnerability. Reviewed source.

- **WEBMIN-1705 — Reserved webmin username rejected in internal APIs.** The change is sensible defense in depth, but the reviewed evidence does not demonstrate that the prior internal namespace collision yields unauthorized access or privilege escalation. Reviewed source.

Core · Integration (2)

- **WEBMIN-1417 — Optional LDAP Unix-SHA mode uses unsalted SHA-1.** The weak algorithm is an explicit legacy interoperability selection rather than an attacker-controlled downgrade or default transition in this change. It is retained as a hardening concern but not accepted as a vulnerability finding for this review. Reviewed source.
- **WEBMIN-2127 — Mailbox and Minecraft glob expansion escapes path boundaries.** The merge narrows expansion by applying \Q...\E to variable path components; the evidence supports hardening rather than a remaining expansion primitive. Reviewed source.

Core · Packaging (1)

- **WEBMIN-1156 — Predictable shared temporary paths in command-line module package generators.** Dismissed as a review finding because this commit adds offline command-line packaging utilities, not a Webmin request handler, and no lower-trust product input reaches them. Reviewed source.

Core · Webmin (1)

- **WEBMIN-1662 — Directory listing remains enabled unless administrators opt out.** The reviewed change did not expose a directory outside the existing Webmin authentication boundary, and no concrete sensitive index-less path accessible to an unauthorized requester was demonstrated. The new option is defense in depth rather than evidence of a review-specific vulnerability. Reviewed source.

Module · Apache (4)

- **WEBMIN-1155 — Permissive redirect destination scheme validation.** The prior regex already excludes whitespace/config-line injection, and the destination is an Apache redirect configured by an authorized editor. The change supports correctness hardening but not a distinct security finding on the available evidence. Reviewed source.
- **WEBMIN-1568 — Document-root ownership fallback to UID/GID zero.** The patch is prudent ownership and failure-handling hardening. The reviewed state does not show privilege gain, attacker-controlled chown, or escape from the permitted document-root boundary. Reviewed source.
- **WEBMIN-1740 — Virtual-server hostname metacharacter hardening.** The allowlist is useful defensive hardening, but this review did not establish a new authority boundary crossed by the prior syntax. Creation already requires the create ACL and includes an explicit user-selected configuration-file mode, while the edited ServerName remains a single non-whitespace token. Reviewed source.
- **WEBMIN-1832 — Empty SSL protocol selection could create invalid configuration.** Dismissed as a security finding because the behavior is confined to a caller already entrusted to modify the same Apache SSL configuration; the change is important validation and availability hardening but does not grant a lower-privileged or untrusted actor a new capability. Reviewed source.

Module · At (2)

- **WEBMIN-1983 — Broad suppression of subroutine-redefinition diagnostics.** The candidate is dismissed because the pragma only suppresses a warning category; Perl already permits the redefinition, and the patch does not alter code-loading sources, dispatch, authorization, data flow, or execution. The reviewed change provides no evidence that an attacker can influence a redefinition or that this warning is a security control. Reviewed source.
- **WEBMIN-2041 — Potential Init-module availability boundary on AT service controls.** The patch is dependency and UI-visibility hardening, not evidence of a privilege bypass. foreign_available incorporates module and user policy, but the reviewed change does not modify a state-changing handler. Reviewed source.

Module · Backup Config (3)

- **WEBMIN-1213 — Empty module element reaches foreign-module processing.** The change is useful hardening, but the reviewed empty value does not resolve to an attacker-selected module. The foreign loader checks the resolved directory and errors when it does not exist, so the candidate lacks a demonstrated security impact. Reviewed source.
- **WEBMIN-2220 — System hostname interpolated into immediate-backup form action.** The only variable component comes from host-level hostname sources, not the HTTP request. The form helper HTML-escapes the action, its prefix is a fixed relative backup.cgi path, and the receiving backup handler does not consume the filename portion. Reviewed source.
- **WEBMIN-2309 — Global notification address is rendered without HTML escaping in the backup editor.** The raw interpolation is a defense-in-depth weakness, but the review establishes no externally controlled or lower-privilege source for the value. The demonstrated writer is a privileged global Webmin configuration surface, and direct file modification requires host administration. Reviewed source.

Module · Bacula Backup (3)

- **WEBMIN-1201 — Request values are concatenated into Bacula console commands.** The reviewed module grants the same principal broad Bacula-administrative abilities, including editing job directives and triggering operations. This review did not establish a lower-privilege requester or a capability gain beyond that intended authority, so the pattern is retained as defense-in-depth rather than a separate finding. Reviewed source.
- **WEBMIN-1202 — File-list endpoint accepts unknown referrers.** The endpoint remains authenticated and read-only, and the change does not permit a cross-origin site to read catalog results. No independent confidentiality or integrity impact was demonstrated from this flag, so it is dismissed separately from the SQL issue. Reviewed source.
- **WEBMIN-1554 — Schedule chooser allowed unknown Referers before hardening.** Removing the override is valid defense-in-depth hardening, but the chooser only renders a form and its submit endpoint returns a client-side popup-opener assignment. With no server-side state mutation or demonstrated action in the victim's application context, the reviewed evidence does not support a CSRF security finding. Reviewed source.

Module · Bind8 (1)

- **WEBMIN-1312 — Semicolon handling in ACL editor warranted boundary review.** The validation is useful defense in depth, but no privilege gain or protected-boundary bypass is established because defaults permission also grants the manual configuration editor. Reviewed source.

Module · Bsdfdisk (1)

- **WEBMIN-2026 — Potential shell injection through unquoted fdisk device argument.** The apparent shell sink has no attacker-controlled token. Perl glob results must match one of six fixed patterns whose only variable characters are decimal digits, and symlinks are skipped before command execution. Reviewed source.

Module · Cluster Cron (1)

- **WEBMIN-1785 — Missing-job dereferences in edit, delete, and execute handlers.** The guard is sound robustness hardening, but lookup equality prevents a nonexistent id from selecting another job, and the inspected downstream calls do not establish a distinct security impact. Ordinary error/availability behavior for an authenticated module user is insufficient for a finding. Reviewed source.

Module · Cluster Software (1)

- **WEBMIN-2156 — Unreadable local package path in installation error.** The candidate does not reach a raw HTML sink in the reviewed code because install_error encodes its entire message argument at the final output boundary. The added inner call is defense in depth and may double-encode characters, but it does not establish a separate vulnerability. Reviewed source.

Module · Cpan (1)

- **WEBMIN-2196 — Shell-form cleanup of an internally generated temporary path.** The cleanup path comes from tempname() with no user-supplied filename, so no attacker-controlled command token was established. Using unlink_file is sound hardening, but the reviewed data flow does not support a security finding. Reviewed source.

Module · Cron (4)

- **WEBMIN-1699 — String-form shell copy in direct-write mode.** The reviewed request paths constrain the username to an existing system account, generate the temporary file internally, and treat cron_dir as administrator configuration. No attacker-controlled shell metacharacter source was established within this review, so the string-form call is a hardening concern rather than a supported security finding. Reviewed source.
- **WEBMIN-2005 — Unquoted temporary paths in crontab command construction.** Dismissed because the reviewed path arguments are created internally by transname and no request-derived component or lower-privilege path control is shown. Changing the configured temporary directory already requires an administrative configuration capability, so the diff supports prudent hardening but not a separate exploitable boundary violation. Reviewed source.
- **WEBMIN-2006 — Manual Cron execution inherited the server default PATH.** Dismissed because the reviewed flow already requires authorization for the job's Unix user, executes as that user, and provides no evidence that a restricted requester controls the inherited server PATH. Explicit per-job Cron environment values are intentionally applied after the default. Reviewed source.
- **WEBMIN-2216 — Implicit Perl argument forwarding in WebSocket module enumeration.** Although explicit parentheses prevent ampersand-call argument inheritance, both required security preconditions are absent from the concrete source path. This is correctness hardening without a demonstrated security consequence. Reviewed source.

Module · Custom (1)

- **WEBMIN-2350 — Repeated scalar parameters reached environment handling.** The added rejection is sound defensive hardening, but free-text scalar values were already attacker-controlled and constrained option types validate values before use. Without evidence of a validation or authorization bypass, this candidate is dismissed. Reviewed source.

Module · DHCP (1)

- **WEBMIN-2001 — Unbounded expansion of a configured shared-network pool.** Dismissed as a security finding because the controlling input is privileged service configuration, the page already materializes configured subnet ranges through the same helper, and no lower-privilege or unauthenticated boundary is crossed. This remains a defensive scaling and hardening consideration. Reviewed source.

Module · Fail2Ban (1)

- **WEBMIN-1795 — Fail2Ban action port validation accepts trailing shell metacharacters.** The validation is permissive and warrants hardening, but the required principal already has an explicit same-module interface for writing arbitrary root-consumed action command text. The observed behavior therefore does not provide a new privilege gain and is not a high or critical security finding. Reviewed source.

Module · Fetchmail (1)

- **WEBMIN-2186 — Unquoted IMAP folder value can alter fetchmail tokens.** The serializer does not quote or narrowly validate the folder value, but the identical authorized form already exposes preconnect and postconnect command controls and saves them into the same user block. Reviewed source.

Module · File (1)

- **WEBMIN-1493 — New recursive permission scopes are not implemented by chmod.cgi.** The mismatch is a genuine functional defect, but it does not give an attacker a new input path or cause an

unauthorized chmod/chown. Values 3 and 4 reach no server-side update branch at all, while implemented modes remain subject to module, read-only, path, and operating-system authorization. Reviewed source.

Module · Filemin (3)

- **WEBMIN-1646 — Current-directory text used in an allowed-path regular expression.** The change is security-relevant hardening, but the post-change hunk directly neutralizes regex interpretation and does not leave a candidate defect at this comparison. Reviewed source.
- **WEBMIN-1820 — Hidden dialog path values might permit attribute injection.** The prior urlized value was semantically inconvenient for form submission but syntactically safe in the attribute because all non-alphanumeric characters, including quotes, became percent escapes. The replacement html_escape value also encodes both quote types. Reviewed source.
- **WEBMIN-2394 — Strict write precheck could disagree with effective Unix permissions.** Dismissed as a security finding because the observed inconsistency was a defense-in-depth precheck: get_paths had already switched the process to the configured Unix UID, and the eventual file operation remained subject to kernel permission enforcement. The change is still security-positive hardening. Reviewed source.

Module · Firewall (1)

- **WEBMIN-1252 — Unbounded numeric IPv4 prefix accepted by firewall rule form.** The change is correctness hardening rather than a distinct vulnerability. The prefix branch is constrained to digits, so it does not create shell or option injection, and the requester already has authority to change the selected table's firewall behavior. Reviewed source.

Module · Grub2 (1)

- **WEBMIN-2363 — Manual editor symlink write-through concern.** Dismissed because the reviewed hardening and the frozen stable source perform lstat-based rejection during enumeration and again through save-time re-enumeration; the accompanying regression cases confirm that outside targets remain unchanged. Reviewed source.

Module · Init (3)

- **WEBMIN-1385 — Mass Upstart response emits service output without HTML escaping.** The reviewed change did not establish a credible lower-privilege source for the root service command output or a cross-user delivery route; the demonstrated case reduces to the same authorized user's own request. HTML escaping remains prudent hardening, but the review evidence does not support a security finding. Reviewed source.
- **WEBMIN-2255 — Caller-controlled paths and content reach systemd override writer.** The helper is a privileged write sink, but no functional caller exists in the reviewed snapshot and therefore no attacker-to-sink data path is established. Calling this internal API already requires trusted module code. Reviewed source.
- **WEBMIN-2256 — Slash-delimited systemd option filter reaches Perl string eval.** String eval is an unsafe API design, but the reviewed snapshot has no functional caller and the frozen stable tree's sole caller uses a fixed literal. No attacker-controlled flow to the sink is demonstrated, so this remains a hardening concern and is not accepted as a finding. Reviewed source.

Module · Kea Dhcp (1)

- **WEBMIN-2359 — CSV parser string eval appears dynamic but uses a constant loader string.** The change is worthwhile hardening and lint cleanup, but neither side of the one-line diff evaluates attacker-controlled content. Lease data is parsed only after the fixed module load, so the candidate does not represent an exploitable injection. Reviewed source.

Module · LDAP Client (1)

- **WEBMIN-2219 — Broken-pipe handling during LDAP connection establishment.** Dismissed as a security candidate. The reviewed change is defensive reliability hardening, and the remaining branch asymmetry does not establish a security boundary failure. Reviewed source.

Module · Logrotate (1)

- **WEBMIN-2376 — Vendor-overlay writes considered link-redirection hazards.** The reviewed hardening is sound defense in depth, but no supported unprivileged path to create the required link in /etc/logrotate.d or alter the module's configured roots was demonstrated. An actor with that filesystem authority is already equivalent to the configuration owner, so this does not establish an independent privilege crossing. Reviewed source.

Module · Logviewer (1)

- **WEBMIN-2334 — Systemd unit and executable values used in shell command construction.** The inputs originate from the system manager and trusted executable discovery, systemd represents unusual unit-name bytes with escapes, and changing those system-level sources requires privileges beyond the restricted Webmin journal reader. The change is sound defense in depth, but the reviewed evidence does not support a reachable vulnerability. Reviewed source.

Module · Mailboxes (3)

- **WEBMIN-1449 — Quoted display name could enter the bare-address parser branch.** The parser correction clearly improves recipient interpretation, but the reviewed change does not demonstrate an authorization bypass, code execution, or reliable attacker-selected misdelivery. It remains a correctness hardening change, not a complete security finding. Reviewed source.
- **WEBMIN-1520 — Maildir and MH size paths bypassed folder-user state restoration.** The state-restoration defect is concrete, but the selected identity is lower-privileged and the reviewed request flow does not demonstrate cross-user data access, elevated writes, or privilege gain. It is retained as reliability and defense-in-depth evidence rather than a separate finding. Reviewed source.
- **WEBMIN-1685 — Two-argument opens treated paths and modes as one expression.** The conversion is sound robustness and hardening, but direct call-site review did not substantiate the required attacker control and privilege transition. The candidate is therefore not promoted to a finding. Reviewed source.

Module · MySQL (1)

- **WEBMIN-1515 — Backtick termination in the CSV column parameter.** The query-construction behavior is real, but it does not cross a Webmin authorization boundary: both csv.cgi and the module's arbitrary-SQL exec.cgi require can_edit_db and reject edonly users. Reviewed source.

Module · Net (1)

- **WEBMIN-1972 — Cross-platform hostname and interface command interpolation.** The apparent sinks warranted review, but hostname input is allowlisted and no attacker-controlled shell-metacharacter path was demonstrated for the touched BSD-family interface-name commands. The changes are defense-in-depth and do not support a separate finding. Reviewed source.

Module · Nftables (2)

- **WEBMIN-2322 — Coarse module access lacked action and table separation.** Before granular ACLs, Webmin's coarse module assignment was the documented authorization boundary; the review found no path for a user lacking module access to invoke these handlers. The new ACL system enables safer delegation and is meaningful defense-in-depth, but absence of optional subdelegation controls is not itself an authorization bypass. Reviewed source.
- **WEBMIN-2325 — Concurrent rules and metadata operations could lose updates.** The commit is sound concurrency hardening, but the historical review did not establish a deterministic security boundary violation or an unprivileged attacker who can win the race. Newest-stable source retains the locking, so no actionable finding remains. Reviewed source.

Module · Nis (2)

- **WEBMIN-2353 — AIX and Solaris domain command escaping.** Dismissed as defense in depth. Before the added quotemeta calls, the request value had already passed a strict character allowlist that

excludes command separators, substitutions, quotes, whitespace, and escapes, and it was placed inside double quotes at the command sink. Reviewed source.

- **WEBMIN-2354 — OpenLinux new-domain copy helper conversion.** Dismissed as defense in depth for request-controlled input. The domain component was already validated with a strict alphanumeric/dot/hyphen allowlist before the prior cp construction; the helper conversion nevertheless improves command-boundary hygiene. Reviewed source.

Module · Passwd (1)

- **WEBMIN-1979 — Distinct errors reveal whether a submitted Unix username exists.** The distinction is an information-hardening concern, but local usernames are not established as confidential by this API contract, the operation is deployment-opt-in, and both guessed-name paths are throttled. The evidence does not support a separate substantive finding for this review. Reviewed source.

Module · PHP Configuration (1)

- **WEBMIN-1973 — Relative PHP configuration paths are ambiguous but remain inside explicit ACL authority.** The new leading-slash check is useful defense-in-depth, but the pre-change relative-path behavior does not cross the contemporaneous authorization boundary: a user without anyfile still needs an exact configured match, while a user with anyfile is expressly authorized for any file. No security finding is supported. Reviewed source.

Module · Postfix (2)

- **WEBMIN-1571 — Historical referrer-check bypass in read-only Postfix map chooser.** The diff is security-relevant hardening, but the reviewed historical flow does not support a concrete vulnerability finding. map_chooser.cgi is a read-only form renderer, an attacker origin cannot inspect its authenticated response under same-origin policy, and the paired state-changing handler retained the ordinary Referer policy. Reviewed source.
- **WEBMIN-2023 — Shell interpretation of a privileged Postfix configuration path.** Dismissed as a standalone vulnerability because this review provides no request-controlled or lower-privilege source for postfix_config_file. Reviewed source.

Module · PostgreSQL (1)

- **WEBMIN-1926 — Process-global HOME mutation could influence later child commands.** The commit establishes undesirable process-global state and corrects its scope, but the temporary path is generated inside Webmin's validated temp tree and no evidence shows that an untrusted party can control the path, place effective configuration there, read the 0600 credential file, or cross a privilege boundary. Reviewed source.

Module · Proc (1)

- **WEBMIN-1904 — Authorized command inherits a Webmin module working directory.** Dismissed as a defense-in-depth containment change rather than a separate vulnerability. The caller intentionally grants arbitrary command execution after checking run permission; changing the child's current directory limits where relative side effects land but does not grant a new command, identity, or filesystem privilege. Reviewed source.

Module · Proftpd (1)

- **WEBMIN-2320 — Failed module discovery may replace Webmin's cached capability list.** The change is defensive availability hardening for Webmin's local capability cache. It neither changes ProFTPD runtime configuration nor gives an untrusted principal control over the cache refresh in the reviewed path. Reviewed source.

Module · Project Meta (1)

- **WEBMIN-2153 — Usermin build setup clones the current Webmin branch.** The clone uses authenticated HTTPS and the same Webmin upstream authority whose source the build intentionally con-

sumes. Control sufficient to alter that repository already controls the build input, so this is reproducibility hardening rather than a separate exploit path in the reviewed review. Reviewed source.

Module · Quota (1)

- **WEBMIN-1334 — Blank quota principal accepted by filesystem-listing pages.** The affected endpoints are listing/form flows; downstream principal arguments are shell-quoted, and the change does not prevent access to any valid principal or demonstrate a security-boundary bypass. This is correctness and defensive hardening rather than a distinct security finding. Reviewed source.

Module · Samba (1)

- **WEBMIN-1390 — Raw Samba account name rendered in the user editor.** The raw rendering is a defense-in-depth concern, but the reviewed code does not give a lower-privilege actor a way to control the account name. The source is a root-controlled Samba backend, the module creates entries from local Unix identities, the edit endpoint cannot rename them, and the commit only changes typography. Reviewed source.

Module · Sendmail (1)

- **WEBMIN-1516 — Pre-conversion feature current-value display rendered stored text without escaping.** The UI conversion correctly adds escaping, but the historical sink does not establish an additional meaningful security boundary: controlling its payload already requires broad feature authority and arbitrary root-consumed M4 text control. It is therefore documented as security-relevant hardening rather than a separate finding. Reviewed source.

Module · Servers (2)

- **WEBMIN-1165 — Whitespace in server-group labels could create ambiguous membership data.** The commit prevents malformed group-label data, but group labels are not the authorization key in the reviewed server module. The caller already needs edit/add permission, existing-server changes also require can_use_server, and no group string reaches a shell or other execution sink in the traced path. Reviewed source.
- **WEBMIN-2037 — Unescaped server-discovery status values might permit reflected injection.** The request-controlled value is constrained to digits and dots ending in .0 before output, and the remaining displayed addresses are locally derived. The encoding change remains useful defense in depth but does not substantiate an exploitable finding in this path. Reviewed source.

Module · Smart Status (1)

- **WEBMIN-2371 — Potential stored markup through a configured device name.** Although the raw interpolation is a hardening weakness, the only identified input source is protected module configuration. A principal allowed to change it already controls the raw smartctl executable and extra shell arguments used by the root-running module, which is a stronger root-equivalent capability than browser script execution. Reviewed source.

Module · Software (1)

- **WEBMIN-1258 — Captured package-manager stderr reaches module response output.** The review did not establish an independent lower-privilege producer for this output: controlling repositories or initiating package installation already requires package-administration authority, and the review's APT path encodes output before rendering. This remains useful hardening context but is not a supported standalone finding. Reviewed source.

Module · Status (1)

- **WEBMIN-1453 — Unvalidated monitor type previously reached dynamic loads and remote Perl evaluation.** The input shape is dangerous and the guard is sound hardening, but it does not confer execution authority beyond the required status edit ACL: that ACL already exposes explicit arbitrary local and remote transition-command execution. It is therefore not reported as a separate privilege-escalation finding. Reviewed source.

Module · Syslog Ng (2)

- **WEBMIN-1652 — Unescaped network source IP in source-list description.** The renderer lacks defense-in-depth encoding, but the newly supported input path only persists a strict IPv4 address. Supplying active markup therefore requires an independent ability to modify the root-owned configuration or equivalent Syslog-NG module authority, so this change does not establish an attacker-controlled security boundary. Reviewed source.
- **WEBMIN-1931 — Configured line count reaches a shell command before integer normalization.** Dismissed as authority-equivalent hardening. The integer conversion is a sound defense-in-depth improvement, but the exact audited configuration surface already grants an authorized configuration editor control over tail_cmd, start_cmd, and restart_cmd. Reviewed source.

Module · System Status (1)

- **WEBMIN-1609 — Operating-system metadata previously entered an HTML-valued UI field without explicit encoding.** The encoding change is sound defense-in-depth, but this review does not establish a lower-privilege input path: controlling the fields requires global Webmin or host administration already capable of changing application code/configuration. The hypothesized script execution therefore does not add a demonstrated capability across a trust boundary. Reviewed source.

Module · Systemd (2)

- **WEBMIN-2366 — Automount selector label may reach HTML without encoding.** Dismissed because the shared ui_select sink already HTML-escapes display text. The explicit call-site escape is defense in depth and does not demonstrate a pre-existing unencoded output path. Reviewed source.
- **WEBMIN-2367 — Non-canonical linger boolean may choose the wrong authorized action.** Dismissed as an operation-correctness defect rather than a security boundary violation: the same checked linger ACL authorizes both outcomes, and the target user restriction is unchanged. Exact comparison is still appropriate hardening. Reviewed source.

Module · Useradmin (7)

- **WEBMIN-1194 — Invalid passwd-line lookup could delete the wrong record.** The guard is important hardening, but full caller review in this review did not establish attacker control over the stale internal line index; an exploit claim would be speculative. Reviewed source.
- **WEBMIN-1198 — Incompatible password salts accepted.** The evidenced impact is failed or malformed password generation. No authentication bypass or credential disclosure was demonstrated, so this remains correctness hardening. Reviewed source.
- **WEBMIN-1353 — Numeric account indices could become stale across concurrent file changes.** The change is valuable race and correctness hardening, but the reviewed endpoints still authorization-check the record they resolve. The review does not establish a cross-privilege target substitution, so no security finding is accepted. Reviewed source.
- **WEBMIN-1804 — Generic Perl evaluation from installed help-file <exec> directives.** The eval primitive was dangerous and appropriately retired, but no lower-privileged input reached it in the audited boundary. Miniserv canonicalized traversal segments before setting PATH_INFO, and the selected help content remained an installed module file. Reviewed source.
- **WEBMIN-1971 — Owner execute bit on newly created authorized_keys.** Dismissed as a security finding because 0700 was unconventional but still owner-only. The permission cleanup is sound, yet it does not itself establish unauthorized access and is separate from the accepted symbolic-link issue. Reviewed source.
- **WEBMIN-2116 — SHA-512 helper previously accepted salts for other crypt schemes.** Scheme-prefix enforcement is prudent, but the reviewed evidence does not show attacker control of the optional salt or a resulting privilege, confidentiality, or integrity violation; the candidate is dismissed as correctness hardening. Reviewed source.
- **WEBMIN-2117 — Yescrypt helper previously accepted salts for other crypt schemes.** The source supports algorithm-consistency hardening, not a concrete exploit path in this review. Without attacker control or demonstrated privilege, confidentiality, or integrity impact, the candidate is dismissed. Reviewed source.

Module · Usermin (1)

- **WEBMIN-2335 — Shell injection through bind-port or certificate-days numeric values.** Dismissed as a separate vulnerability because both values are constrained to decimal digits before command construction. Adding quotemeta remains sound defense in depth, but the reviewed source does not support shell-metacharacter injection through these two numeric fields. Reviewed source.

Module · Webminlog (1)

- **WEBMIN-2223 — Global notification address is placed in an HTML label without escaping.** The raw HTML sink is concrete, but this review does not establish attacker-controlled input across a meaningful privilege boundary. The only in-tree writer found for webmin_email_to is the Webmin Configuration mail-settings path, while a Webmin Actions Log user merely reads the value. Reviewed source.

Module · Xterm (1)

- **WEBMIN-2317 — Possible websocket data callback before session handshake completion.** The reviewed code exposes the backend only on 127.0.0.1, accepts one connection, and validates miniserv's session-bound key in the websocket handshake callback. Reviewed source.

Module · Zones (1)

- **WEBMIN-1318 — Forged brand value reaches the zonecfg command stream.** The input is insufficiently constrained, but the interpreter reached by this path is zonecfg, not a host shell. The module already exposes creation and modification of zone filesystems, networks, resources, and global zone properties, so no distinct privilege or confidentiality boundary was demonstrated. Reviewed source.

Prior CVE reference set (109 identifiers)

The researcher knew of published CVEs before this campaign. This table lists every CVE identifier currently linked to a Known Bugs record under the Webmin project: 107 published and two rejected. These records supplied prior context for the campaign. The public data archive retains their associated titles and states.

The two rejected IDs have different reasons. CVE-2015-1990 was in a CNA pool that was never assigned to an issue, even though a Webmin notice used that identifier for an xmlrpc.cgi XSS report. CVE-2019-15231 was rejected as a duplicate; the retained identifier is CVE-2019-15107. Neither rejection is a finding from this audit.

CVE identifier	Record status	Known Bugs ID
CVE-1999-1074	published	WEBMIN-1038
CVE-2001-0222	published	WEBMIN-1039
CVE-2001-1074	published	WEBMIN-1040
CVE-2001-1196	published	WEBMIN-1041
CVE-2001-1530	published	WEBMIN-1042
CVE-2002-0756	published	WEBMIN-1043
CVE-2002-0757	published	WEBMIN-1044
CVE-2002-1672	published	WEBMIN-1045
CVE-2002-1673	published	WEBMIN-1046
CVE-2002-1947	published	WEBMIN-1047
CVE-2002-2201	published	WEBMIN-1048
CVE-2002-2360	published	WEBMIN-1049
CVE-2003-0101	published	WEBMIN-1050
CVE-2004-0559	published	WEBMIN-1051
CVE-2004-0582	published	WEBMIN-1052
CVE-2004-0583	published	WEBMIN-1053
CVE-2004-1468	published	WEBMIN-1054
CVE-2005-1177	published	WEBMIN-1055
CVE-2005-3042	published	WEBMIN-1056
CVE-2005-3912	published	WEBMIN-1057
CVE-2006-3274	published	WEBMIN-1058
CVE-2006-3392	published	WEBMIN-1059

CVE identifier	Record status	Known Bugs ID
CVE-2006-4542	published	WEBMIN-1060
CVE-2007-1276	published	WEBMIN-1061
CVE-2007-3156	published	WEBMIN-1062
CVE-2007-5066	published	WEBMIN-1063
CVE-2008-0720	published	WEBMIN-1064
CVE-2009-4568	published	WEBMIN-1065
CVE-2011-1937	published	WEBMIN-1066
CVE-2012-2981	published	WEBMIN-1067
CVE-2012-2982	published	WEBMIN-0317
CVE-2012-2983	published	WEBMIN-0318
CVE-2012-4893	published	WEBMIN-1070
CVE-2014-0339	published	WEBMIN-1071
CVE-2014-3885	published	WEBMIN-1072
CVE-2014-3886	published	WEBMIN-1073
CVE-2014-3924	published	WEBMIN-1074
CVE-2015-1377	published	WEBMIN-1075
CVE-2015-1990	rejected	WEBMIN-1076
CVE-2017-15644	published	WEBMIN-1077
CVE-2017-15645	published	WEBMIN-1078
CVE-2017-15646	published	WEBMIN-1079
CVE-2017-17089	published	WEBMIN-1080
CVE-2017-2106	published	WEBMIN-1081
CVE-2017-9313	published	WEBMIN-1082
CVE-2018-19191	published	WEBMIN-1083
CVE-2018-8712	published	WEBMIN-1084
CVE-2019-12840	published	WEBMIN-1085
CVE-2019-15107	published	WEBMIN-1086
CVE-2019-15231	rejected-duplicate	WEBMIN-1086
CVE-2019-15641	published	WEBMIN-1087
CVE-2019-15642	published	WEBMIN-1088
CVE-2019-9624	published	WEBMIN-1089
CVE-2020-12670	published	WEBMIN-1090
CVE-2020-35606	published	WEBMIN-1091
CVE-2020-35769	published	WEBMIN-1092
CVE-2020-8820	published	WEBMIN-0689
CVE-2020-8821	published	WEBMIN-1094
CVE-2021-31760	published	WEBMIN-1095
CVE-2021-31761	published	WEBMIN-1096
CVE-2021-31762	published	WEBMIN-1097
CVE-2021-32156	published	WEBMIN-1098
CVE-2021-32157	published	WEBMIN-1099
CVE-2021-32158	published	WEBMIN-1100
CVE-2021-32159	published	WEBMIN-1101
CVE-2021-32160	published	WEBMIN-1102
CVE-2021-32161	published	WEBMIN-1103
CVE-2021-32162	published	WEBMIN-1104
CVE-2022-0824	published	WEBMIN-1105
CVE-2022-0829	published	WEBMIN-1106
CVE-2022-30708	published	WEBMIN-1107
CVE-2022-36446	published	WEBMIN-1108
CVE-2022-36880	published	WEBMIN-1109
CVE-2022-3844	published	WEBMIN-1110
CVE-2023-38303	published	WEBMIN-1111
CVE-2023-38304	published	WEBMIN-1112
CVE-2023-38305	published	WEBMIN-1113
CVE-2023-38306	published	WEBMIN-1114
CVE-2023-38307	published	WEBMIN-1115
CVE-2023-38308	published	WEBMIN-1116
CVE-2023-38309	published	WEBMIN-1117

CVE identifier	Record status	Known Bugs ID
CVE-2023-38310	published	WEBMIN-1118
CVE-2023-38311	published	WEBMIN-1119
CVE-2023-40982	published	WEBMIN-1120
CVE-2023-40983	published	WEBMIN-1121
CVE-2023-40984	published	WEBMIN-1122
CVE-2023-40985	published	WEBMIN-1123
CVE-2023-40986	published	WEBMIN-1124
CVE-2023-41155	published	WEBMIN-1125
CVE-2023-41163	published	WEBMIN-1126
CVE-2023-43309	published	WEBMIN-1127
CVE-2023-52046	published	WEBMIN-1128
CVE-2024-12828	published	WEBMIN-1129
CVE-2024-36450	published	WEBMIN-1130
CVE-2024-36451	published	WEBMIN-1131
CVE-2024-36452	published	WEBMIN-1132
CVE-2024-36453	published	WEBMIN-1133
CVE-2024-45692	published	WEBMIN-1134
CVE-2025-61541	published	WEBMIN-0896
CVE-2025-67738	published	WEBMIN-0930
CVE-2026-102337	published	WEBMIN-1137
CVE-2026-22678	published	WEBMIN-1138
CVE-2026-42210	published	WEBMIN-0934
CVE-2026-49102	published	WEBMIN-1140
CVE-2026-49103	published	WEBMIN-0998
CVE-2026-49243	published	WEBMIN-1142
CVE-2026-56020	published	WEBMIN-1143
CVE-2026-56021	published	WEBMIN-1144
CVE-2026-56022	published	WEBMIN-0934

Mapped upstream fixes (135)

The checked public upstream source revision is `0dd66e3d5b5c`. The following audited findings have an evidence-backed mapping to a public fix commit in that revision's ancestry.

Finding	Mapped public fix commit
WEBMIN-0007	`e2e54f4ea639`
WEBMIN-0013	`d30ecd28bbbb`
WEBMIN-0015	`f6137ff68fa2`
WEBMIN-0017	`ba5c385dcf96`
WEBMIN-0020	`ddf5f13a87af`
WEBMIN-0029	`8402301a08de`
WEBMIN-0041	`e329c228c9f6`
WEBMIN-0043	`baad8643ae02`
WEBMIN-0045	`96eae73b788f`
WEBMIN-0048	`960eed1cba10`
WEBMIN-0053	`3fcb9dd0406c`
WEBMIN-0058	`c06baf0755b1`
WEBMIN-0065	`218c3e8bc2cb`
WEBMIN-0073	`01b988dbf1e8`
WEBMIN-0074	`e52c5e0196a6`
WEBMIN-0095	`abb5b40323a7`
WEBMIN-0098	`400c326a1b71`
WEBMIN-0102	`a2a3b1a87b15`
WEBMIN-0117	`ace94bda95d9`
WEBMIN-0143	`6dc1809e28bf`
WEBMIN-0144	`872f7ec16eae`
WEBMIN-0145	`aa8ad9663e48`

Finding	Mapped public fix commit
WEBMIN-0156	8b420ba88c05
WEBMIN-0157	5d9bbd84016a
WEBMIN-0158	96eae73b788f
WEBMIN-0172	1c2cff183e13
WEBMIN-0191	ead31644e4f2
WEBMIN-0204	48731ccded0a
WEBMIN-0208	237d62194ce7
WEBMIN-0212	b22c7c87f27b
WEBMIN-0217	adbd9965a3b0
WEBMIN-0219	c853f121fda3
WEBMIN-0230	820a9f99a897
WEBMIN-0238	f0c28d6e0ed7
WEBMIN-0240	01b988dbf1e8
WEBMIN-0243	b8e0462f7b98
WEBMIN-0248	feeabbca6da2
WEBMIN-0264	c853f121fda3
WEBMIN-0265	fe563dcdc3ce
WEBMIN-0277	68713ca03428
WEBMIN-0293	16cf0b0cac02
WEBMIN-0296	c06baf0755b1
WEBMIN-0302	dd4f1bb93f90
WEBMIN-0309	d7ea434835b0
WEBMIN-0314	5fc1d76ddde3
WEBMIN-0315	5386c5a4aff7
WEBMIN-0323	c06baf0755b1
WEBMIN-0330	5fc1d76ddde3
WEBMIN-0336	020b1b95c127
WEBMIN-0338	f4344c9c245e
WEBMIN-0339	8402301a08de
WEBMIN-0343	f918d37cbcf5
WEBMIN-0344	01b988dbf1e8
WEBMIN-0346	68713ca03428
WEBMIN-0388	25e826f1759c
WEBMIN-0394	5e4b93741064
WEBMIN-0396	f936f44d2c39
WEBMIN-0398	0cd7a6ace3ff
WEBMIN-0401	d85a34dfa224
WEBMIN-0409	e912a82c2c3c
WEBMIN-0410	8391c7821c7d
WEBMIN-0429	ba5c385dcf96
WEBMIN-0430	224cbd91467c
WEBMIN-0442	b1a6ec3dfbed
WEBMIN-0444	8f880ab366ed
WEBMIN-0445	4e1c9abcea7f
WEBMIN-0447	d2eba12d0105
WEBMIN-0455	96eae73b788f
WEBMIN-0469	5e504f38ad70
WEBMIN-0473	96eae73b788f
WEBMIN-0479	c06baf0755b1
WEBMIN-0482	8f880ab366ed
WEBMIN-0490	05ec33f8f310
WEBMIN-0496	c06baf0755b1
WEBMIN-0498	f8f35b26c6cd
WEBMIN-0504	49c8b88de1ef
WEBMIN-0511	db232a368463
WEBMIN-0515	0cd7a6ace3ff
WEBMIN-0516	4b3ba2995867
WEBMIN-0522	4d179ac3659e
WEBMIN-0524	01b988dbf1e8

Finding	Mapped public fix commit
WEBMIN-0545	069798e036ad
WEBMIN-0547	99fd83e830e9
WEBMIN-0548	01b988dbf1e8
WEBMIN-0566	5bd2e5237c80
WEBMIN-0572	6ae856025078
WEBMIN-0590	1d8608d61b85
WEBMIN-0591	b29a5ac88b9a
WEBMIN-0610	ada632465f62
WEBMIN-0612	e777aa96e21a
WEBMIN-0635	c3a28bf2b319
WEBMIN-0645	60c55190431b
WEBMIN-0671	806872a89758
WEBMIN-0672	3ccad807df80
WEBMIN-0681	77bca6104682
WEBMIN-0686	0cd7a6ace3ff
WEBMIN-0687	05954bf7f351
WEBMIN-0701	3d69fee3dc2f
WEBMIN-0714	b850bae55ba5
WEBMIN-0718	3f1812458d6f
WEBMIN-0722	0843de05cb9c
WEBMIN-0724	c06baf0755b1
WEBMIN-0734	9869c5496dfc
WEBMIN-0737	0843de05cb9c
WEBMIN-0743	80738aaab675
WEBMIN-0745	0678fbfa6bfb
WEBMIN-0760	80738aaab675
WEBMIN-0773	5eaf180b88ab
WEBMIN-0784	0f9a539b5df6
WEBMIN-0785	a1c0a075e785
WEBMIN-0797	e9db7ba4f316
WEBMIN-0803	fe1f12347977
WEBMIN-0808	c1d819bcbd95
WEBMIN-0810	c06baf0755b1
WEBMIN-0814	2c89f7b097cd
WEBMIN-0847	6b78f7d8b5be
WEBMIN-0850	adbd9965a3b0
WEBMIN-0859	7840c1f15f83
WEBMIN-0861	d2831bad4d1c
WEBMIN-0863	7e2ad6f558ca
WEBMIN-0869	3ccad807df80
WEBMIN-0878	c3a28bf2b319
WEBMIN-0907	bf44976c2327
WEBMIN-0914	db5966c6ecbd
WEBMIN-0915	02c02214d7bd
WEBMIN-0920	35a71584cca6
WEBMIN-0943	5320d7d1572f
WEBMIN-0954	16cf0b0cac02
WEBMIN-0971	13cf1a32c3ae
WEBMIN-0988	87fbd16b6f1a
WEBMIN-1009	c130b0d32981
WEBMIN-1011	8c275ff0a6e0
WEBMIN-1029	1b8992052c58
WEBMIN-2405	3466d09069f7
WEBMIN-2462	a1e7d5cec96d

References

[1] FIRST. *Common Vulnerability Scoring System v4.0: Specification Document*. Specification.
[2] OWASP Foundation. *Web Security Testing Guide*, version 4.2. Guide.

[3] Dongliang Mu, Alejandro Cuevas, Limin Yang, Hang Hu, Xinyu Xing, Bing Mao, and Gang Wang. *Understanding the Reproducibility of Crowd-reported Security Vulnerabilities.* 27th USENIX Security Symposium, 2018. Proceedings.
[4] MITRE. *Common Weakness Enumeration.* CWE.
[5] Fengjie Li, Jiajun Jiang, Dongchi Chen, and Yingfei Xiong. *LLM-based Vulnerability Detection at Project Scale: An Empirical Study.* arXiv:2601.19239v2, 2026. Paper.
[6] Ze Sheng, Zhicheng Chen, Qingxiao Xu, Kewen Zhu, and Jeff Huang. *FuzzingBrain V2: A Multi-Agent LLM System for Automated Vulnerability Discovery and Reproduction.* arXiv:2605.21779, 2026. Paper.
[7] OpenAI. *OpenAI Daybreak - Trusted Access for Cyber overview.* Documentation.
[8] Webmin. *Security.* Webmin security notices.
[9] Webmin. *Security Advisories.* GitHub advisories.
[10] OpenAI. *Daybreak Blue model alias.* Model documentation.
[11] OpenAI. *API pricing*, Standard tier, 19 November 2026. Pricing table.