Shell shocked - the Bash bug - blueAPACHE

Canonical: https://www.blueapache.com/blog/shell-shocked-the-bash-bug-2

Details:

Historical article: 2014. Vulnerability and operating-system descriptions below reflect that period; assess actual exposure using the affected version, configuration and vendor patches.

A critical security flaw known as ‘Shell Shock’ has recently been found in Bash that is worrying a lot of people, and rightly so. The US National Vulnerability Database rate this new flaw as a Level 10, and CERT Australia has noted plenty of online chatter around exploiting the bug.

What is Bash?

Bash (Bourne Again Shell) is a pervasive command line utility used in many Unix-based operating systems including Linux and OS X. Bash is not a new command line tool – it was first released in 1989 and is now distributed open source software under the GNU project. Its design can be directly traced back to the origins of Unix in the late 1960s.

What is Shell Shock?

Shell Shock was first discovered by Edinburgh-based programmer Stephane Chazelas weeks ago. The bug, present in all versions of Bash dating back at least to 1994, relates to the handling of configuration information. A maliciously formatted configuration string can cause Bash to do literally anything the user has permission to do. Remote exploitation is possible when a vulnerable service passes attacker-controlled data to Bash. Commands run with the privileges of the affected process; root access is not an automatic outcome.

Who is affected?

Unpatched affected versions of Bash can be at risk where the vulnerable processing path is exposed. This includes Linux, Unix and Mac OS users (including Mac desktop and lap top users). Most Mac desktop systems do not have many network-accessible server programs running on them by default, which limits the ways the bug could be exploited. However, email attachments represent one possibility the bug can be leveraged, as do malicious Wi-Fi hot spots. Mac laptop users connecting to untrusted hot spots risk an attacker exploiting the flaw and securing full access to their computer. Windows-based desktops and laptops do not include Bash by default, so only those that have deliberately installed Bash need be concerned.

How to mitigate the risk?

Most operating system vendors have already released updates that completely or at least partially mitigate the risk of being exploited. CERT Australia recommend that all system administrators and consumers ensure their software updates are applied as soon as they become available.

References

CISA hosts the archived US-CERT/DHS Shellshock alert about the vulnerability. The US National Vulnerability Database rates this as severity 10 . To better understand your security risks, contact our Account Management Team .

What is 'Shell Shock' as described in the blueAPACHE blog post?

Shellshock is a historical Bash vulnerability that can allow commands embedded in environment variables to execute. Remote exploitation depends on an exposed service passing attacker-controlled data to vulnerable Bash; the commands run with that process's privileges, not automatically as root.

Who discovered the Shell Shock bug and when was it present in Bash?

Shell Shock was first discovered by Edinburgh-based programmer Stephane Chazelas. The bug has been present in all versions of Bash dating back at least to 1994.

What is Bash, according to the blueAPACHE article?

Bash (Bourne Again Shell) is a pervasive command line utility used in many Unix-based operating systems, including Linux and OS X. It was first released in 1989, is distributed as open source software under the GNU project, and its design can be traced back to the origins of Unix in the late 1960s.

How severe was the Shell Shock vulnerability rated?

The US National Vulnerability Database rated the Shell Shock flaw as a Level 10 (severity 10), the highest rating, and CERT Australia noted plenty of online chatter around exploiting the bug.

Who is at risk from the Shell Shock bug?

The archived alert identifies affected GNU Bash versions through 4.3 on relevant Linux, Mac OS X and other Unix-like systems. Exposure depends on version, configuration and patch status; installing any version of Bash does not by itself establish vulnerability.

How could Mac users be exploited by the Shell Shock bug?

The historical article discussed possible paths through network-accessible services and untrusted networks. Whether a particular Mac is exposed depends on the installed Bash version, reachable services and applied patches.

What mitigation does the blueAPACHE article recommend for the Shell Shock bug?

Most operating system vendors had already released updates that completely or partially mitigate the risk of exploitation. CERT Australia recommended that all system administrators and consumers ensure their software updates are applied as soon as they become available.

What references did the blueAPACHE blog post cite regarding Shell Shock?

The article cites an Advisory issued by CERT Australia about the vulnerability, and notes that the US National Vulnerability Database rates the flaw as severity 10 (referencing CVE-2014-6271).