THREAT ADVISORY *CRITICAL*

July 5, 2024

THREAT ADVISORY *CRITICAL* -- regreSSHion: Critical Vulnerability in OpenSSH (CVE-2024-6387)

SUMMARY

The critical OpenSSH vulnerability impacts almost every Unix-like and Linux system (except OpenBSD), is due to a serious flaw known as "regreSSHion," (CVE-2024-6387), and exposes Linux environments to remote unauthenticated code execution. The potential impact of this vulnerability is extensive, posing a risk to numerous servers and infrastructure components worldwide.

RISK SCORE

CVE-ID	CVSSv3 Score
CVE-2024-6387	8.1

THREAT INTELLIGENCE

PoC available on GitHub

VULNERABILITY DETAILS

"regreSSHion," exposes Linux environments to remote unauthenticated code execution, potentially leading to root-level access and does not require user interaction. The vulnerability stems from a signal handler race condition in OpenSSH's server (sshd) affecting versions 8.5p1 to 9.8p1 on glibc-based Linux systems because the syslog() function calls async-signal-unsafe functions like malloc() and free(), resulting in unauthenticated remote code execution with root privileges. This issue arises because sshd's privileged code is not sandboxed and operates with full privileges. OpenBSD is not affected because its signal alarm (SIGALRM) handler uses syslog_r(), an async-signal-safe version of syslog().

This issue is a regression of an older vulnerability (CVE-2006-5051) and can be exploited by attackers through precise timing to manipulate system memory. The complexity of the exploit requires multiple attempts and significant skill, but the potential impact is severe.

POC EXPLOIT

The "regreSSHion" proof of concept exploit on GitHub uses a complex race condition in OpenSSH that requires precise timing and potentially thousands of attempts to succeed. It targets 32-bit systems and has been tested on Debian-based, glibc-based Linux distributions. Key aspects of the exploit include:

- Timing and Duration: The exploit takes about 10,000 attempts to succeed, approximately 3-4 hours to win the race condition, and 6-8 hours to bypass ASLR and gain remote root shell access.
- prepare_heap() Function: Sets up memory in a specific way, creating and freeing small chunks, then creating pairs of large and small chunks filled with specific data to create a predictable memory layout.

- attempt_race_condition() Function: Tries to exploit the race condition by sending a crafted packet to the server, timing the final byte to be sent just before the server times out the connection. This aims to manipulate the server's memory, allowing the attacker to run code with root permissions.

AFFECTED PRODUCTS

- OpenSSH version 8.5p1 to 9.7p1
- Older versions prior to 4.4p1 if unpatched for CVE-2006-5051 and CVE-2008-4109

SOLUTION

- OpenSSH version 9.8p1

RECOMMENDATIONS

- Update to OpenSSH version 9.8p1 or later.
- If immediate updating is not possible, administrators can set the login timeout to zero (LoginGraceTime=0 in sshd_config) as a temporary mitigation. However, developers warn that this makes the SSH server more susceptible to DDoS attacks.
- Limit SSH access to necessary IP addresses and networks using firewall rules.
- Use jump hosts or bastion servers for additional access control.
- Deploy host-based intrusion prevention tools like fail2ban to monitor and block suspicious SSH activity.
- Adjust sshd_config settings:
  - Set LoginGraceTime to 0.
  - Reduce MaxStartups to limit unauthenticated connections (e.g., MaxStartups 10:30:100).
  - Set PerSourceMaxStartups to a small number (e.g., 5) to limit connections from a single IP.
- Implement strict network segmentation using VLANs or network zones to isolate critical systems.
- Implement multi-factor authentication (MFA) for SSH access.
- Establish robust logging and monitoring for SSH services with alerts for unusual activity.
- Consider alternative secure remote access methods that don't rely on SSH, such as VPN solutions with strong authentication.

REFERENCES

- https://www.splunk.com/en_us/blog/security/cve-2024-6387-regresshion-vulnerability.html
- https://unit42.paloaltonetworks.com/threat-brief-cve-2024-6387-openssh/
- https://sec1.io/blog/cve-2024-6387-critical-rce-openssh/?gclid=EAIaIQobChMImrbTnt2OhwMVSBeDAx0gvwVGEAAYASAAEgK3i_D_BwE
- https://www.qualys.com/regresshion-cve-2024-6387/
- https://www.kaspersky.com/blog/openssh-vulnerability-mitigation-cve-2024-6387-regresshion/51603/