Alex Shah
EN.695.741.81.SP25 Information Assurance Analysis
Mod 1 Assignment 1 Part 2
January 26, 2025

1.
Name: “Apple Multiple Products Use-After-Free Vulnerability” CVE-2025-24085

Links:
https://www.cve.org/CVERecord?id=CVE-2025-24085
https://cwe.mitre.org/data/definitions/416
https://nvd.nist.gov/vuln/detail/CVE-2025-24085

Date reported: January 27, 2025, though active exploitation was acknowledged as before iOS 17.2 released on December 12, 2023

Exploit target: Apple VisionOS, iOS, iPadOS, MacOS, watchOS, and tvOS

Severity: 7.8 NVD, CISA 7.3 (High) for CVE-2025-24085

Advisories:
https://support.apple.com/en-us/122073
https://support.apple.com/en-us/122072
https://support.apple.com/en-us/122068
https://support.apple.com/en-us/122071
https://support.apple.com/en-us/122066

Propagation: A malicious application could leverage this use after free exploit in CoreMedia to elevate privileges such as a fake media player. The user would have to download the malicious application without being aware it contains the exploit and run it on their unpatched device. These applications could be sideloaded, make their way onto the app store without being detected, or otherwise use the exploit on CoreMedia functions which may be exploitable through other applications.

Mitigation: Apple has released an update to the effected platforms that users should update their devices to in order to mitigate the use after free exploit. Users should install reputable applications in order to avoid the exploit on unpatched devices.

Detection: Analysis of the source code or debugging the memory of the processes would reveal malicious code that uses the exploit. Users can also determine which applications are using the CoreMedia permission effected by the exploit. There are limited ways to evaluate applications on Apple platforms.

Patches: In the most recent update of Apple software, the vulnerability has been “addressed with improved memory management”. The update versions are visionOS 2.3, iOS 18.3 and iPadOS 18.3, macOS Sequoia 15.3, watchOS 11.3, and tvOS 18.3.

Actors: It is not reported what actors were using the exploit or against what targets but it suggested that targeted attacks could have used this exploit in the past and that fake media applications could be suggested to users to trick them into installing an application with the exploit.

TTP: Social engineering such as phishing or malicious links would allow an attacker to trick a user into installing an application on an unpatched device. The user would need to install and run the application so local access to the device by user interaction would be required. The exploit would allow for privilege escalation and execution of code such as downloading further payloads from controlled servers, or manipulating or exfiltrating data from the device.

2.
Name: Mirai (Mirai Botnet) including CVE-2024-45163

Links:
https://web.archive.org/web/20161021003956/https://securityintelligence.com/news/leaked-mirai-malware-boosts-iot-insecurity-threat-level/
https://www.fortinet.com/blog/threat-research/iot-botnet-more-targets-in-okirus-cross-hairs
https://arstechnica.com/information-technology/2017/12/100000-strong-botnet-built-on-router-0-day-could-strike-at-any-time/
https://www.justice.gov/usao-nj/pr/justice-department-announces-charges-and-guilty-pleas-three-computer-crime-cases

Date reported: August 2016

Exploit target: Linux based devices and IoT devices

Severity: N/A Mirai does not have a specific severity as a botnet

Advisories:
https://nvd.nist.gov/vuln/detail/CVE-2024-45163
https://www.cloudflare.com/learning/ddos/glossary/mirai-botnet/
https://www.avast.com/c-mirai
https://spectrum.ieee.org/mirai-botnet

Propagation: The Mirai author gathered common default credentials, common passwords, and brute force to attempt to log in to devices exposed to the internet such as routers and cameras in order to create a botnet. A server ran to initially sweep IP addresses and hammer for login access, then on controlled devices the same method was used to try IP addresses against a list of credentials. Later variants used further exploits on common hardware to create a larger target such as CVE-2019-3929 which affected display controllers and projectors by leveraging vulnerabilities in firmware/software of devices that often go unpatched.

Mitigation: Mirai primarily attempts to use default credentials to take over internet connected devices so changing default credentials and using strong passwords, limiting exposure to connected devices, as well as staying up to date on security patches can prevent Mirai infections.

Detection: Mirai can cause excessive resource usage or large volumes of traffic as it performs actions as part of the botnet or scanning for new devices to infect. Monitoring device resource usage as well as network usage, including using an IDS/IPS can detect Mirai activity. Logs may show access attempts including unfamiliar logins.

Patches: Vendors more frequently use randomized default credentials to prevent this type of attack, and implement security patches to fix the exploits in network and IoT hardware like ISP issued routers that were commonly targeted by Mirai and other botnets.

Actors: The author of Mirai was identified and the worm was initially used to target the author's rivals and competitors as they also offered services protecting from DDoS attacks. The author later released the source code which enabled variants and other actors

TTP: The attacker would first need to exploit publicly accessible devices to log in with default or common credentials then execute commands on the device to download payloads to be part of the botnet or search for more devices. The scripts are placed on the device and set to run automatically to keep persistence. Mirai scans local devices and attempts to move laterally as well as find new devices on the internet to infect. The botnet formed by Mirai is used to DDoS web sites and services.

3.
Name: Heartbleed CVE-2014-0160

Links:
https://web.archive.org/web/20170123161742/https://www.shodan.io/report/DCPO7BkV
https://cwe.mitre.org/data/definitions/126.html
https://www.csoonline.com/article/562859/the-heartbleed-bug-how-a-flaw-in-openssl-caused-a-security-crisis.html
https://www.cvedetails.com/cve/CVE-2014-0160/

Date reported: April 7 2014

Exploit target: Cryptography using OpenSSL versions 1.0.1 through 1.0.1f

Severity: 7.5 High for CVE-2014-0160

Advisories:
https://openssl-library.org/news/secadv/20140407.txt
https://access.redhat.com/security/vulnerabilities/heartbleed
https://nvd.nist.gov/vuln/detail/CVE-2014-0160
https://www.cisa.gov/news-events/alerts/2014/04/08/openssl-heartbleed-vulnerability-cve-2014-0160

Propagation: N/A The vulnerability was present in unpatched installations using the very widely distributed OpenSSL library for cryptography and doesn't propagate or spread. This could leak keys and passwords as well as other sensitive data in memory which could lead to further compromise. Researchers disclosed the vulnerability in April of 2014, and while operating system level patches were issued quickly, it wasn't until June 2014 before a patch to OpenSSL resolved the exploit was released. In the meantime, existing installations could be vulnerable and exploitable by bad actors, as well as those who chose not to update after the patch was released. Shodan tracked the still unpatched installations years later in 2017 to find that nearly two hundred thousand discoverable installations were not patched against the exploit including 42,032 in the United States, with a majority of the installations using the library for serving HTTPS traffic. This large quantity of exposed devices could be exploited by anyone who was able to discover the devices and try to extract sensitive information from leaked memory, which could be done repeatedly.

Mitigation: The OpenSSL library was patched to close the vulnerability by a software update 1.0.1g, but prior to the patch other cryptographic libraries could be used instead for cryptographic key generation and exchange or OpenSSL can be used with disabling the “heartbeat” feature when compiling the package. After disabling the feature or patching the installation keys and certificates should be regenerated.

Detection: The vulnerability was present in all installations between the introduction of the bug in 1.0.1 and the patch in 1.0.1g. There may be memory leaks in logs or unexpected logins due to the usage of stolen keys and passwords. Heartbleed can be detected by scanning with tools like nmap.

Patches: The OpenSSL library was patched to resolve the Heartbleed vulnerability in update 1.0.1g and later.

Actors: N/A While patches were quickly rolled out, as the 2017 Shodan scans revealed, there were many devices years after the disclosure of the vulnerability that were still unpatched to the Heartbleed exploit discoverable over the Internet. Opportunistic hackers and researchers would be able to exploit the vulnerability in order to gather sensitive data like passwords and keys that could be used to compromise the device or to perform reconnaissance. But no specific actors created or exploited the vulnerable feature in OpenSSL.

TTP: Reconnaissance is performed to determine exploitable servers which can then be accessed with the exploit, then the leaked memory checked for sensitive data. The stolen credentials can be used to login to steal or manipulate data.