Lab 1 Report

OWASP Broken Web App (10.0.2.4)

Basic Network Scan

Samba Badlock Vulnerability (Nessus Plugin ID 90509)
A high-risk vulnerability detected during the Basic Network Scan is the “Samba Badlock vulnerability.” The specified vulnerability has a CVSS v3.0 base score of 7.5 and affects the version of Samba running on this remote service. The “Samba Badlock Vulnerability” is found in the Security Account Manager (SAM) and Local Security Authority (LSAD). As a result of this vulnerability, the SAM database is susceptible to man-in-the-middle attacks that allow the attacker to lower the user authentication level via Remote Procedure Call (RPC) channels. This attack permits the bad actor to take control of the victim machine, grants the bad actor access to sensitive data via the SAM database, and enables the bad actor to cause damage to the machine.

Countermeasure
A countermeasure for the Samba Badlock Vulnerability is to update the Samba version and allow for security patches to be installed. Updating Samba should ensure that any potential known weaknesses in the SAM are closed off and should prevent bad actors from taking advantage of the same exploit.

Web Application Test

Web Application Potentially Vulnerable to Clickjacking (Nessus Plugin ID 85582)
A medium risk vulnerability detected during the Web Application Test scan is “Web Application Potentially Vulnerable to Clickjacking.” The specified vulnerability has a CVSS v2.0 base score of 4.3 and affects this remote web server since it fails to set a X-Frame-Options response header. If a user accesses a website using this vulnerable remote web server, they may be at risk of a clickjacking attack. A clickjacking attack is caused by the user clicking on an infected part of the web page they are visiting. Clickjacking can have different results spanning from leading the user to a malicious site or adjusting security settings on the victim machine.

Countermeasure
A countermeasure to the web application clickjacking vulnerability is to ensure that the remote web server is using the X-Frame-Options response header. This solution should prevent content from being layered and embedded on a targeted website by an attacker. This can be accomplished by using the “DENY” setting in the X-Frame-Options header.

DVWA (10.0.2.5)

Basic Network Scan

SSL Certificate Signed Using Weak Hashing Algorithm (Nessus Plugin ID 35291)

A high-risk vulnerability detected during the Basic Network Scan is “SSL certificate signed using weak hashing algorithm.” The specified vulnerability has a CVSS v3.0 base score of 7.5 and affects this remote service since it uses a flawed cryptographic hashing algorithm such as MD2, MD4, MD5, or SHA1. Attackers can exploit this vulnerability by using collision attacks against the targeted SSL certificates. If a collision attack is successful, the bad actor can create SSL certificates that replicate the digital signature of the original certificate but contain different data. These additional SSL certificates can be used for malicious purposes such as posing as legitimate websites or sending fraudulent emails under the affected digital signature.

Countermeasure

The best countermeasure to prevent collision attacks on SSL certificates using weak hashing algorithms is to update the system and allow for the latest security patches to be installed. Security updates should ensure that the system is using newer hashing algorithms such as SHA-2 or SHA-3.

Web Application Test

CGI Generic Cookie Injection Scripting (Nessus Plugin ID 44136)

A medium risk vulnerability detected during the Web Application Test Scan is “CGI Generic Cookie Injection Scripting.” The specified vulnerability has a CVSS v2.0 base score of 4.3 and affects this remote web server since it uses a common gateway interface (CGI) script that does not sanitize URLs containing malicious JavaScript. Attackers can take advantage of this vulnerability by using session fixation to inject cookies into the victim’s browser. Session fixation is used for malicious purposes such as gaining unauthorized access to a victim’s account on a targeted website. Session fixation can be achieved by using methods such as cross-site scripting to inject a cookie containing the attacker’s fixed session ID from the targeted website into the victim’s browser. Once the victim logs into their account on the targeted website, the attacker can use the injected session ID cookie to hijack the victim’s account and use it for nefarious purposes.

Countermeasure

Session fixation is a security vulnerability that is dependent on the web application developer implementing a patch. However, without a patch, a countermeasure that could be implemented in response to the CGI script failing to sanitize URLs is to prevent access to the affected web application. By restricting access to the application, the risk of session fixation should be reduced since that attack vector is closed off.

Undetected Vulnerabilities

Basic Network Scan

SSL DROWN Attack Vulnerability (Nessus Plugin ID 89058)

A vulnerability detected on DVWA that was not found on OWASP Broken Web App is the “SSL DROWN attack vulnerability.” The specified vulnerability has a CVSS v3.0 base score of 5.9 and affects this remote service since it supports SSLv2. DROWN attacks enable bad actors to get around the encryption on HTTPS and TLS protocols using SSLv2. Once these protocols are decrypted, the bad actor can gain access to sensitive data such as passwords and banking information. In addition, DROWN attacks can allow bad actors to display false secure websites on the victim's machine as well as intercept and manipulate services that the victim is attempting to access.

Countermeasure

A countermeasure for the DROWN attack vulnerability is to have the server operator disable SSLv2. This can be accomplished by updating the servers to use newer versions of software such as OpenSSL and NSS. These newer software versions no longer support SSLv2 or have it disabled. In addition, sever operators should monitor private keys and prevent them from being used on servers that support SSLv2.

Web Application Test

WordPress Pingback File Information Disclosure (Nessus Plugin ID 24237)

A vulnerability detected on OWASP Broken Web App that was not found on DVWA is the "WordPress pingback file information disclosure." The specified vulnerability has a CVSS v2.0 base score of 5 and affects this remote web server since the remote host's version of WordPress fails to sanitize the sourceURI when processing pingbacks. An attacker can take advantage of this exploit to locate files on the server and gain readability access based on the permissions of the remote host. In addition, a bad actor can use this exploit to perform DOS attacks.

Countermeasure

A countermeasure to the WordPress pingback file information disclosure is to ensure that the remote host has WordPress version 2.1 or better installed. If the remote host does not have the correct version of WordPress installed, they should be denied access to the remote web server and prompted to update WordPress. This ensures that the remote host is no longer a viable attack vector.

Sources

[1] "Samba Badlock Vulnerability." Tenable. https://www.tenable.com/plugins/nessus/90509 (accessed April 18, 2024).

[2] "CVE-2016-2118.html." Samba. https://www.samba.org/samba/security/CVE-2016-2118.html (accessed April 18, 2024).

[3] "SSL Certificate Signed Using Weak Hashing Algorithm." Tenable. https://www.tenable.com/plugins/nessus/35291 (accessed April 18, 2024).

[4] "Microsoft Security Advisory 961509." Microsoft. https://learn.microsoft.com/en-us/security-updates/SecurityAdvisories/2008/961509 (accessed April 18, 2024).

[5] Josh Lake. "What is a collision attack? Threats to digital signature security." Comparitech. https://www.comparitech.com/blog/information-security/what-is-a-collision-attack/ (accessed April 18, 2024).

[6] "SSL DROWN Attack Vulnerability (Decrypting RSA with Obsolete and Weakened eNcryption)." Tenable. https://www.tenable.com/plugins/nessus/89058 (accessed April 18, 2024).

[7] “The DROWN Attack.” Drown Attack. https://drownattack.com/ (accessed April 18, 2024).

[8] “CGI Generic Cookie Injection Scripting.” Tenable. https://www.tenable.com/plugins/nessus/44136 (accessed April 18, 2024).

[9] Mitja Kolšek, “Session Fixation Vulnerability in Web-based Applications,” ACROS Security, Slovenia, version 1.0 – revision 1, 2007.

[10] “Web Application Potentially Vulnerable to Clickjacking.” Tenable. https://www.tenable.com/plugins/nessus/85582 (accessed April 19, 2024).

[11] Gustav Rydstedt. “Clickjacking.” OWASP. https://owasp.org/www-community/attacks/Clickjacking (accessed April 19, 2024).

[12] “Clickjacking Defense Cheat Sheet.” OWASP Cheat Sheet Series. https://cheatsheetseries.owasp.org/cheatsheets/Clickjacking_Defense_Cheat_Sheet.html (accessed April 19, 2024).

[13] “Web Application Potentially Vulnerable to Clickjacking.” Tenable. https://www.tenable.com/plugins/nessus/85582 (accessed April 19, 2024).