Systems Affected

Networked systems

Overview

Securing end-to-end communications plays an important role in protecting privacy and preventing some forms of man-in-the-middle (MITM) attacks. Recently, researchers described a MITM attack used to inject code, causing unsecured web browsers around the world to become unwitting participants in a distributed denial-of-service attack. That same code can be employed to deliver an exploit for a particular vulnerability or to take other arbitrary actions.

Description

A MITM attack occurs when a third party inserts itself between the communications of a client and a server. MITM attacks as a general class are not new. Classic MITM attacks (e.g., ARP Spoofing) focus on redirecting network communications. By definition, network infrastructure under attacker control is vulnerable to MITM. However, as technology evolves, new methods for performing MITM attacks evolve as well.

Currently, there is no single technology or configuration to prevent all MITM attacks. However, increasing the complexity with multiple layers of defense may raise the cost for the attacker. Increasing the attacker's cost in time, effort, or money can be an effective deterrent to avoiding future network compromise.

Generally, encryption and digital certificates provide an effective safeguard against MITM attacks, assuring both the confidentiality and integrity of communications. As a result, modern MITM attacks have focused on taking advantage of weaknesses in the cryptographic infrastructure (e.g., certificate authorities (CAs), web browser certificate stores) or the encryption algorithms and protocols themselves.

Impact

MITM attacks are critical because of the wide range of potential impacts—these include the exposure of sensitive information, modification of trusted data, and injection of data.

Solution

Employing multiple network and browser protection methods forces an attacker to develop different tactics, techniques, and procedures to circumvent the new security configuration.

CERT recommends reviewing the following mitigations to reduce vulnerability to MITM attacks:

Update Transport Layer Security and Secure Socket Layer (TLS/SSL)

CERT recommends upgrading TLS to 1.1 or higher and ensuring TLS 1.0 and SSL 1, 2, 3.x are disabled, unless required. TLS 1.0 clients can fall back to version 3.0 of the SSL protocol, which is vulnerable to a padding oracle attack when Cypher-Block Chaining mode is used. This method is commonly referred to as the "POODLE" (Padding Oracle on Downgraded Legacy Encryption) attack. Vulnerable TLS implementations can be updated by applying the patch provided by the vendor. Vendor information is available in the National Vulnerability Database (NVD) entry for CVE-2014-3566 [1] or in CERT Vulnerability Note VU#577193 [2]. See US-CERT TA14-290A [3] for additional information on this vulnerability.

Utilize Certificate Pinning

Certificate pinning [4] is a method of associating X.509 certificate and its public key to a specific CA or root. Typically, certificates are validated by checking a verifiable chain of trust back to a trusted root certificate. Certificate pinning bypasses this validation process and allows the user to trust "this certificate only" or "trust only certificates signed by this certificate." Please use the following resources to configure your browser for certificate pinning:

Microsoft Certificate Trust

The Microsoft Enhanced Mitigation Experience Toolkit (EMET) 5.2 employs a feature named "Certificate Trust" for SSL/TLS certificate pinning. This feature is intended to detect and stop MITM attacks that leverage Public Key Infrastructure. [5]

To use the Certificate Trust, you must provide a list of websites you want to protect and certificate pinning rules applicable to those websites. In order to do this, work with the Certificate Trust Configuration feature of the graphical application or use the Configuration Wizard to automatically configure EMET with the recommended settings. [6] Also, ensure period defaults are updated through patching.

Browser Certificate Pinning

Google Chrome and Mozilla Firefox, among others, perform certificate pinning. They conduct a variation of certificate pinning using the HTTP Strict Transport Security (HSTS), which pre-loads a specific set of public key hashes into the HSTS configuration, limiting valid certificates to only those with the specified indicated public key. Chrome uses HTTPS pins for most Google properties. It uses whitelisted public keys which include keys from Verisign, Google Internet Authority, Equifax, and GeoTrust. Thus, Chrome will not accept certificates for Google properties from other CAs.

Firefox 32 on desktop and later (Firefox 34 and later on Android) has the ability to use certificate pinning. It also has the ability to enforce built-in pinsets (mapping of public keys) information to domains. Firefox will pin all sites that Chrome already does, pin their own sites after audit and cleansing, and pin other popular sites that are already in good standing. Please visit this site on How to Use Pinning [7] and for more information.

Implement DNS-based Authentication of Named Entities (DANE)

DANE is a protocol that allows certificates (X.509) commonly used for TLS. DANE is bound to DNS which uses Domain Name System Security Extensions (DNSSEC). A working group in the Internet Engineering Task Force of DANE developed a new type of DNS record that allows a domain itself to sign statements about which entities are authorized to represent it. [8]

Google Chrome does not use DANE but uses an add-on [9] for support. Mozilla Firefox also uses an add-on(link is external) [10] to check the existence and validity of DNSSEC.

Use Network Notary Servers

Network notary servers aim to improve the security of communications between computers and websites by enabling browsers to verify website authenticity without relying on CAs. CAs are often considered a security risk because they can be compromised. [11] As a result, browsers can deem fraudulent sites trustworthy and are left vulnerable to MITM attacks.

Each network notary server, or group of servers, is public and can be operated by public/private organizations or individuals. These servers regularly monitor websites and build a history of each site's certificate data over time. When a browser equipped with a network notary add-on communicates with a website and obtains its certificate information, a user-designated network notary server supplies the browser with historical certificate data for that site. If certificate information provided by the website is inconsistent with the notary's historical data, a MITM attack could be at play. [12]

References

- [1] https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-3566
- [2] http://www.kb.cert.org/vuls/id/577193
- [3] https://www.us-cert.gov/ncas/alerts/TA14-290A
- [4] https://www.owasp.org/index.php/Certificate_and_Public_Key_Pinning
- [5] https://support.microsoft.com/en-us/kb/2458544(link is external)
- [6] https://technet.microsoft.com/en-us/library/cc700843.aspx(link is external)
- [7] https://wiki.mozilla.org/SecurityEngineering/Public_Key_Pinning
- [8] http://www.internetsociety.org/articles/dane-taking-tls-authentication-next-level-using-dnssec
- [9] http://www.internetsociety.org/deploy360/resources/how-to-add-dnssec-support-to-google-chrome/
- [10] https://www.dnssec-validator.cz/(link is external)
- [11] http://perspectives-project.org/
- [12] http://arstechnica.com/information-technology/2008/08/network-notary-system-thwarts-man-in-the-middle-attacks/(link is external)

Revisions

- April 30, 2015: Initial Release