Man-in-the-middle attack

In cryptography and computer security, a **man-in-the-middle**, **monster-in-the-middle**,[1][2] **machine-in-the-middle**, **monkey-in-the-middle**,[3] **meddler-in-the-middle**[4] (**MITM**) or **person-in-the-middle**[5] (**PITM**) attack is a cyberattack where the attacker secretly relays and possibly alters the communications between two parties who believe that they are directly communicating with each other, as the attacker has inserted themselves between the two parties.[6] One example of a MITM attack is active eavesdropping, in which the attacker makes independent connections with the victims and relays messages between them to make them believe they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker.[7] The attacker must be able to intercept all relevant messages passing between the two victims and inject new ones. This is straightforward in many circumstances; for example, an attacker within the reception range of an unencrypted Wi-Fi access point could insert themselves as a man-in-the-middle.[8][9][10] As it aims to circumvent mutual authentication, a MITM attack can succeed only when the attacker impersonates each endpoint sufficiently well to satisfy their expectations. Most cryptographic protocols include some form of endpoint authentication specifically to prevent MITM attacks. For example, TLS can authenticate one or both parties using a mutually trusted certificate authority.[11][9]

Contents

Example

Suppose Alice wishes to communicate with Bob. Meanwhile, Mallory wishes to intercept the conversation to eavesdrop and optionally to deliver a false message to Bob.

First, Alice asks Bob for his public key. If Bob sends his public key to Alice, but Mallory is able to intercept it, an MITM attack can begin. Mallory sends Alice a forged message that appears to originate from Bob, but instead includes Mallory's public key.

Alice, believing this public key to be Bob's, encrypts her message with Mallory's key and sends the enciphered message back to Bob. Mallory again intercepts, deciphers the message using her private key, possibly alters it if she wants, and re-enciphers it using the public key she intercepted from Bob when he originally tried to send it to Alice. When Bob receives the newly enciphered message, he believes it came

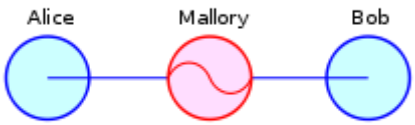
from Alice.

1. Alice sends a message to Bob, which is intercepted by Mallory:

   Alice *"Hi Bob, it's Alice. Give me your key."* → Mallory Bob

2. Mallory relays this message to Bob; Bob cannot tell it is not really from Alice:

   Alice Mallory *"Hi Bob, it's Alice. Give me your key."* → Bob



An illustration of the man-in-the-middle attack

3. Bob responds with his encryption key:

   Alice Mallory ← *[Bob's key]* Bob

4. Mallory replaces Bob's key with her own, and relays this to Alice, claiming that it is Bob's key:

   Alice ← *[Mallory's key]* Mallory Bob

5. Alice encrypts a message with what she believes to be Bob's key, thinking that only Bob can read it:

   Alice *"Meet me at the bus stop!" [encrypted with Mallory's key]* → Mallory Bob

6. However, because it was actually encrypted with Mallory's key, Mallory can decrypt it, read it, modify it (if desired), re-encrypt with Bob's key, and forward it to Bob:

   Alice Mallory *"Meet me at the van down by the river!" [encrypted with Bob's key]* → Bob

7. Bob thinks that this message is a secure communication from Alice.

This example[12] shows the need for Alice and Bob to have some way to ensure that they are truly each using each other's public keys, rather than the public key of an attacker. Otherwise, such attacks are generally possible, in principle, against any message sent using public-key technology. A variety of techniques can help defend against MITM attacks.

Defense and detection

MITM attacks can be prevented or detected by two means: authentication and tamper detection. Authentication provides some degree of certainty that a given message has come from a legitimate source. Tamper detection merely shows evidence that a message may have been altered.

Authentication

All cryptographic systems that are secure against MITM attacks provide some method of authentication for messages. Most require an exchange of information (such as public keys) in addition to the message over a secure channel. Such protocols, often using key-agreement protocols, have been developed with different

security requirements for the secure channel, though some have attempted to remove the requirement for any secure channel at all.[13]

A public key infrastructure, such as Transport Layer Security, may harden Transmission Control Protocol against MITM attacks. In such structures, clients and servers exchange certificates which are issued and verified by a trusted third party called a certificate authority (CA). If the original key to authenticate this CA has not been itself the subject of a MITM attack, then the certificates issued by the CA may be used to authenticate the messages sent by the owner of that certificate. Use of mutual authentication, in which both the server and the client validate the other's communication, covers both ends of a MITM attack. If the server or client's identity is not verified or deemed as invalid, the session will end.[14] However, the default behavior of most connections is to only authenticate the server, which means mutual authentication is not always employed and MITM attacks can still occur.

Attestments, such as verbal communications of a shared value (as in ZRTP), or recorded attestments such as audio/visual recordings of a public key hash[15] are used to ward off MITM attacks, as visual media is much more difficult and time-consuming to imitate than simple data packet communication. However, these methods require a human in the loop in order to successfully initiate the transaction.

In a corporate environment, successful authentication (as indicated by the browser's green padlock) does not always imply secure connection with the remote server. Corporate security policies might contemplate the addition of custom certificates in workstations' web browsers in order to be able to inspect encrypted traffic. As a consequence, a green padlock does not indicate that the client has successfully authenticated with the remote server but just with the corporate server/proxy used for SSL/TLS inspection.

HTTP Public Key Pinning (HPKP), sometimes called "certificate pinning," helps prevent a MITM attack in which the certificate authority itself is compromised, by having the server provide a list of "pinned" public key hashes during the first transaction. Subsequent transactions then require one or more of the keys in the list must be used by the server in order to authenticate that transaction.

DNSSEC extends the DNS protocol to use signatures to authenticate DNS records, preventing simple MITM attacks from directing a client to a malicious IP address.

Tamper detection

Latency examination can potentially detect the attack in certain situations,[16] such as with long calculations that lead into tens of seconds like hash functions. To detect potential attacks, parties check for discrepancies in response times. For example: Say that two parties normally take a certain amount of time to perform a particular transaction. If one transaction, however, were to take an abnormal length of time to reach the other party, this could be indicative of a third party's interference inserting additional latency in the transaction.

Quantum cryptography, in theory, provides tamper-evidence for transactions through the no-cloning theorem. Protocols based on quantum cryptography typically authenticate part or all of their classical communication with an unconditionally secure authentication scheme. As an example Wegman-Carter authentication.[17]

Forensic analysis

Captured network traffic from what is suspected to be an attack can be analyzed in order to determine whether there was an attack and, if so, determine the source of the attack. Important evidence to analyze when performing network forensics on a suspected attack includes:[18]

- IP address of the server
- DNS name of the server
- X.509 certificate of the server
  - Whether the certificate has been self signed
  - Whether the certificate has been signed by a trusted certificate authority
  - Whether the certificate has been revoked
  - Whether the certificate has been changed recently
  - Whether other clients, elsewhere on the Internet, received the same certificate

Notable instances

A notable non-cryptographic MITM attack was perpetrated by a Belkin wireless network router in 2003. Periodically, it would take over an HTTP connection being routed through it: this would fail to pass the traffic on to its destination, but instead itself responded as the intended server. The reply it sent, in place of the web page the user had requested, was an advertisement for another Belkin product. After an outcry from technically literate users, this feature was removed from later versions of the router's firmware.[19]

In 2011, a security breach of the Dutch certificate authority DigiNotar resulted in the fraudulent issuing of certificates. Subsequently, the fraudulent certificates were used to perform MITM attacks.[20]

In 2013, Nokia's Xpress Browser was revealed to be decrypting HTTPS traffic on Nokia's proxy servers, giving the company clear text access to its customers' encrypted browser traffic. Nokia responded by saying that the content was not stored permanently, and that the company had organizational and technical measures to prevent access to private information.[21]

In 2017, Equifax withdrew its mobile phone apps following concern about MITM vulnerabilities.[22]

Other notable real-life implementations include the following:

- DSniff – the first public implementation of MITM attacks against SSL and SSHv1
- Fiddler2 HTTP(S) diagnostic tool
- NSA impersonation of Google[23]
- Qaznet Trust Certificate
- Superfish malware
- Forcepoint Content Gateway – used to perform inspection of SSL traffic at the proxy
- Comcast uses MITM attacks to inject JavaScript code to 3rd party web pages, showing their own ads and messages on top of the pages[24][11][8]
- 2015 Kazakhstan man-in-the-middle attack

See also

- ARP spoofing – a technique by which an attacker sends Address Resolution Protocol messages onto a local area network
- Aspidistra transmitter – a British radio transmitter used for World War II "intrusion" operations, an early MITM attack.
- Babington Plot – the plot against Elizabeth I of England, where Francis Walsingham intercepted the correspondence.
- Computer security – the design of secure computer systems.

- Cryptanalysis – the art of deciphering encrypted messages with incomplete knowledge of how they were encrypted.
- Digital signature – a cryptographic guarantee of the authenticity of a text, usually the result of a calculation only the author is expected to be able to perform.
- Evil maid attack – attack used against full disk encryption systems
- Interlock protocol – a specific protocol to circumvent an MITM attack when the keys may have been compromised.
- Key management – how to manage cryptographic keys, including generation, exchange and storage.
- Key-agreement protocol – a cryptographic protocol for establishing a key in which both parties can have confidence.
- Man-in-the-browser – a type of web browser MITM
- Man-on-the-side attack – a similar attack, giving only regular access to a communication channel.
- Mutual authentication – how communicating parties establish confidence in one another's identities.
- Password-authenticated key agreement – a protocol for establishing a key using a password.
- Quantum cryptography – the use of quantum mechanics to provide security in cryptography.
- Secure channel – a way of communicating resistant to interception and tampering.
- Spoofing attack – Cyber attack in which a person or program successfully masquerades as another by falsifying data

References

1. Gabbi Fisher; Luke Valenta (March 18, 2019). "Monsters in the Middleboxes: Introducing Two New Tools for Detecting HTTPS Interception" (https://blog.cloudflare.com/monsters-in-the-middleboxes/).
2. Matthias Fassl (April 23, 2018). "Usable Authentication Ceremonies in Secure Instant Messaging" (http://www.ifs.tuwien.ac.at/~weippl/Thesis/2018/Matthias%20Fassl%20-%20Usable%20Authentication%20Ceremonies%20in%20Secure%20Instant%20Messaging.pdf) (PDF).
3. John R Richter (November 24, 2019). "Monkey In The Middle" (https://sites.psu.edu/hacking/2017/02/24/monkey-in-the-middle/).
4. Poddebniak, Damian; Ising, Fabian; Böck, Hanno; Schinzel, Sebastian (August 13, 2021). *Why TLS Is Better Without STARTTLS: A Security Analysis of STARTTLS in the Email Context* (https://www.usenix.org/system/files/sec21-poddebniak.pdf) (PDF). 30th USENIX Security Symposium (https://www.usenix.org/conference/usenixsecurity21/technical-sessions). p. 4366. ISBN 978-1-939133-24-3. "When a *Meddler-in-the-Middle (MitM)* attacker removes the STARTTLS capability from the server response, they can easily downgrade the connection to plaintext."
5. "Person-in-the-middle" (https://www.cyber.gov.au/acsc/view-all-content/glossary/person-middle). 2020-10-11.
6. Elakrat, Mohamed Abdallah; Jung, Jae Cheon (2018-06-01). "Development of field programmable gate array–based encryption module to mitigate man-in-the-middle attack for nuclear power plant data communication network" (https://linkinghub.elsevier.com/retrieve/pii/S173857331730565X). *Nuclear Engineering and Technology*. **50** (5): 780–787. doi:10.1016/j.net.2018.01.018 (https://doi.org/10.1016%2Fj.net.2018.01.018).

7. Wang, Le; Wyglinski, Alexander M. (2014-10-01). "Detection of man-in-the-middle attacks using physical layer wireless security techniques: Man-in-the-middle attacks using physical layer security" (https://onlinelibrary.wiley.com/doi/10.1002/wcm.2527). *Wireless Communications and Mobile Computing*. **16** (4): 408–426. doi:10.1002/wcm.2527 (https://doi.org/10.1002%2Fwcm.2527).
8. "Comcast continues to inject its own code into websites you visit" (https://thenextweb.com/insights/2017/12/11/comcast-continues-to-inject-its-own-code-into-websites-you-visit/). 2017-12-11.
9. Callegati, Franco; Cerroni, Walter; Ramilli, Marco (2009). "Man-in-the-Middle Attack to the HTTPS Protocol". *IEEE Security & Privacy Magazine*. **7**: 78–81. doi:10.1109/MSP.2009.12 (https://doi.org/10.1109%2FMSP.2009.12). S2CID 32996015 (https://api.semanticscholar.org/CorpusID:32996015).
10. Tanmay Patange (November 10, 2013). "How to defend yourself against MITM or Man-in-the-middle attack" (https://web.archive.org/web/20131124235452/http://hackerspace.lifehacker.com/how-to-defend-yourself-against-mitm-or-man-in-the-middl-1461796382). Archived from the original (https://hackerspace.lifehacker.com/how-to-defend-yourself-against-mitm-or-man-in-the-middl-1461796382) on November 24, 2013. Retrieved November 25, 2014.
11. "Comcast still uses MITM javascript injection to serve unwanted ads and messages" (https://www.privateinternetaccess.com/blog/2016/12/comcast-still-uses-mitm-javascript-injection-serve-unwanted-ads-messages/). 2016-12-28.
12. "diffie hellman - MiTM on RSA public key encryption" (https://crypto.stackexchange.com/questions/31224/mitm-on-rsa-public-key-encryption). *Cryptography Stack Exchange*.
13. Merkle, Ralph C (April 1978). "Secure Communications Over Insecure Channels". *Communications of the ACM*. **21** (4): 294–299. CiteSeerX 10.1.1.364.5157 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.364.5157). doi:10.1145/359460.359473 (https://doi.org/10.1145%2F359460.359473). S2CID 6967714 (https://api.semanticscholar.org/CorpusID:6967714). "Received August, 1975; revised September 1977"
14. Sasikaladevi, N. and D. Malathi. 2019. “Energy Efficient Lightweight Mutual Authentication Protocol (REAP) for MBAN Based on Genus-2 Hyper-Elliptic Curve.” Wireless Personal Communications 109(4):2471–88.
15. Heinrich, Stuart (2013). "Public Key Infrastructure based on Authentication of Media Attestments". arXiv:1311.7182v1 (https://arxiv.org/abs/1311.7182v1) [cs.CR (https://arxiv.org/archive/cs.CR)].
16. Aziz, Benjamin; Hamilton, Geoff (2009). "Detecting man-in-the-middle attacks by precise timing" (https://researchportal.port.ac.uk/portal/files/107556/Detecting_Man-in-the-Middle_Attacks_by_Precise_Timing.pdf) (PDF). *2009 Third International Conference on Emerging Security Information, Systems and Technologies*: 81–86. doi:10.1109/SECURWARE.2009.20 (https://doi.org/10.1109%2FSECURWARE.2009.20). ISBN 978-0-7695-3668-2. S2CID 18489395 (https://api.semanticscholar.org/CorpusID:18489395).
17. "5. Unconditionally secure authentication" (http://www.lysator.liu.se/~jc/mthesis/5_Unconditionally_secure_au.html). *liu.se*.
18. "Network Forensic Analysis of SSL MITM Attacks" (http://www.netresec.com/?page=Blog&month=2011-03&post=Network-Forensic-Analysis-of-SSL-MITM-Attacks). *NETRESEC Network Security Blog*. Retrieved March 27, 2011.
19. Leyden, John (2003-11-07). "Help! my Belkin router is spamming me" (https://www.theregister.co.uk/2003/11/07/help_my_belkin_router/). *The Register*.
20. Zetter, Kim (2011-09-20). "DigiNotar Files for Bankruptcy in Wake of Devastating Hack" (https://www.wired.com/2011/09/diginotar-bankruptcy/). *Wired*. ISSN 1059-1028 (https://www.worldcat.org/issn/1059-1028). Retrieved 2019-03-22.

21. Meyer, David (10 January 2013). "Nokia: Yes, we decrypt your HTTPS data, but don't worry about it" (http://gigaom.com/2013/01/10/nokia-yes-we-decrypt-your-https-data-but-dont-worry-about-it/). Gigaom, Inc. Retrieved 13 June 2014.
22. Weissman, Cale Guthrie (September 15, 2017). "Here's Why Equifax Yanked Its Apps From Apple And Google Last Week" (https://www.fastcompany.com/40468811/heres-why-equifax-yanked-its-apps-from-apple-and-google-last-week). *Fast Company*.
23. "NSA disguised itself as Google to spy, say reports" (http://news.cnet.com/8301-13578_3-57602701-38/nsa-disguised-itself-as-google-to-spy-say-reports/). *CNET*. 12 Sep 2013. Retrieved 15 Sep 2013.
24. "Comcast using man-in-the-middle attack to warn subscribers of potential copyright infringement" (https://www.techspot.com/news/62887-comcast-using-man-middle-attack-warn-subscribers-potential.html). *TechSpot*.

External links

- Finding Hidden Threats by Decrypting SSL (http://www.sans.org/reading-room/whitepapers/analyst/finding-hidden-threats-decrypting-ssl-34840) (PDF). SANS Institute.