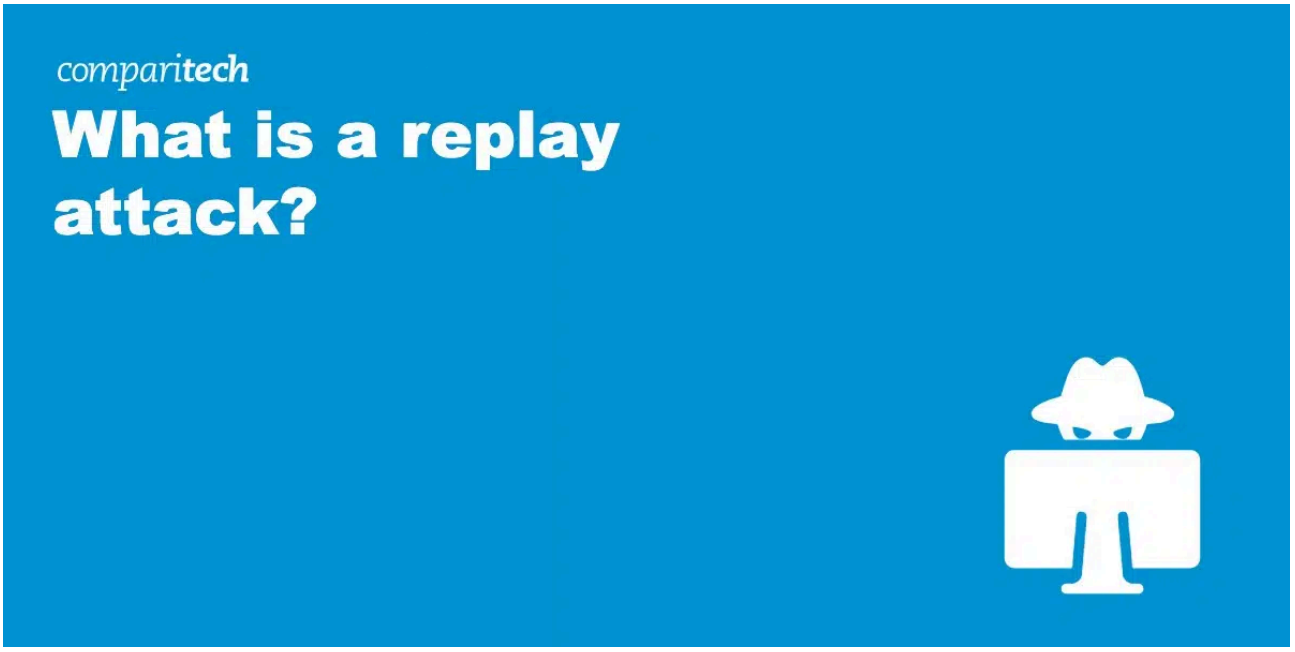
What is a replay attack?

By Justin Schamotta

Published: 2022-10-28 · Archived: 2026-04-05 17:12:45 UTC



In the story *Ali Baba and the Forty Thieves*, the robber captain utters “Open Sesame!” to gain entrance to the cave containing the thieves’ stolen goods. After overhearing the phrase, Ali Baba reuses it to open the doorway to the cave himself, thus engaging in a replay attack.

In computing, a replay attack occurs when transmitted authentication or access control information is intercepted and then re-transmitted to either produce an unauthorized effect or gain unauthorized access.

Replay attacks are listed as entry 294 in the Common Weakness Enumeration (CWE), and described as:

> “[A] flaw [that] exists when the design of the software makes it possible for a malicious user to sniff network traffic and bypass authentication by replaying it to the server in question to the same effect as the original message.”

Replay attacks are particularly concerning because the attacker doesn’t need to decrypt the intercepted messages to trick the receiver into acknowledging them as genuine.

Overview: What is a replay attack?

A replay attack — also known as a playback attack or repeat attack — typically begins with an attacker ‘sniffing’ the stream of small data packets being sent from one IP address to another. Replay attacks are a form of man-in-the-middle attack, with the ‘man’ being the packet sniffer.

What are data packets?

What might start off as a web page or email is split up into smaller pieces, sent through local networks or the wider internet, and reassembled into the original web page or email upon reaching the intended destination.

While they travel through the internet, data packets pass through many traffic control devices, such as switches and routers. At these points, they can be intercepted and analyzed by a packet sniffer — this is a catch-all term for small pieces of software or hardware that can copy data to the attacker's hard drive for subsequent examination.

The attacker can set the packet sniffer — also known as a protocol analyzer, packet analyzer, or network analyzer — to filter only packets that contain specific data elements that they are interested in. Wireshark is one of the most popular packet sniffers and is designed for use by legitimate IT professionals for network analysis.

How much usable data a packet sniffer captures depends on the strength of the network security protocols, but it might include which websites a user visits and what is viewed while there, or the contents and destination of an email. Or it might simply be a password hash.

In a replay attack, the attacker can make use of both encrypted and unencrypted data.

- **Unencrypted data**

Unencrypted data is the easiest for an attacker to manipulate when captured. Let's say Brad (B) sends a message to Charles (C) asking for $50. As Charles trusts Brad, he transfers the money. However, an attacker has captured the initial transfer request from Brad and resent it (with altered bank details). Charles again thinks the request is from Brad, so again sends the money, but this time it goes to the attacker's account.

- **Encrypted data**

While capturing encrypted data might not seem particularly useful, it still has value to an attacker. All they need to do is intercept and resend the entire package — message and key — to fool the receiving server into acknowledging them.

Say that person A wishes to login to a website. They send their authentication data, which might consist of a session ID, login credentials and password hash — all of which are encrypted. If our attacker, person B, intercepts this data and then forwards it on to the intended server, they can sit back and receive whatever information that person A's authentication data has unlocked.

Key elements of a replay attack

Replay attacks typically involve the following stages:

- Attacker installs a packet sniffing tool on a network
- Filtered data is copied to attacker's computer
- Attacker either immediately resends packets of interest, or manipulates them and then resends them
- Attacker receives response from the destination server, which now believes the attacker is the originator of the message

Examples of a replay attack

In September 2022, an offshoot of the Ethereum network — dubbed ETHPoW — suffered a replay attack when attackers duplicated transactions on both ETHPoW and Ethereum chains, enabling them to harvest approximately 200 ETHW.

Prior to this, a Linden Lab vulnerability (CVE-2007-4961) enabled attackers to capture password hashes and replay them to the Second Life authentication server in order to gain entry.

Even Tor has been shown to be susceptible to replay attacks. By hijacking an entry and exit onion router, attackers can copy a packet and send it along. When the packet reaches the now-compromised exit router, it's rejected as a duplicate. However, this tells the attacker the destination and the source of the message, and negates the supposed anonymity of the connection.

According to Microsoft's Threat Intelligence team, one of the most dangerous financial criminal groups – Octo Tempest – uses replay attacks with harvested tokens to bypass multi-factor authentication.

Microsoft also describes how attacks conducted by Midnight Blizzard — a Russian state-affiliated hacker group – in 2023 used session replay attacks so that they could gain initial access to cloud resources.

Outside of the online world, replay attacks have been used to compromise chip-on-card and voice recognition systems. They've also been successful in tricking car and garage doors into opening.

How to prevent replay attacks

There's no one solution to preventing replay attacks. Instead, you must incorporate several strategies:

- **Create random session keys**

In a two-way conversation, sender and receiver should establish a completely random session key. A session key is a string of characters used within an encryption algorithm to make it appear as if it were random, and only used during a particular session. Because it is only valid for the duration of the session, its usefulness to an attacker is limited.

Websites supporting TLS (transport layer security) generate session keys at the start of any communication session between the client and the server. This occurs during what is known as the TLS handshake.

- **One-time passwords**

These are similar to session IDs in that they either expire after initial use, or after a limited period of time. They can be used to authenticate individual transactions, as well as sessions. They are often used by banks to authenticate customers.

- **Timestamping**

You can timestamp messages to ensure that a particular request can't be used more than once. Servers receiving those falling outside an established threshold – which might be a few hundred milliseconds – will reject them.

The Kerberos protocol – used by Windows 2000 and later versions – includes the use of time stamps to limit the effectiveness of replay attacks. Messages which are past the "time to live (TTL)" limit are discarded.

A timestamp forms part of a strong digital signature. These are used to confirm that information has originated from the signer and has not been altered. They are essentially encrypted stamps of authentication used for emails or electronic documents.

- **Repeat messages**

Instruct servers to refuse to accept repeat messages. The Windows Communication Foundation (WCF), which provides a framework for building service-oriented applications, uses this type of defense measure. Servers with an updated Replay Cache can limit the number of messages they will accept.

- **Sequencing**

By sequentially numbering legitimate messages, the receiving server can reject packets which don't arrive in order.

Replay attack FAQs

What other methods can attackers use to intercept data?

ARP poisoning is a type of cyberattack that abuses weaknesses in the Address Resolution Protocol (ARP) to redirect network traffic. It can be used to initially capture the data for a replay attack, but will only work with networks using ARP.

ARP is designed to translate between Media Access Control (MAC) addresses (at the data link layer) and IP addresses (at the network layer). It lets devices on a network find out which devices are assigned particular IP addresses. The results are stored and used to create a list of current MAC-to-IP mappings.

Any device on the network can answer an ARP request, so an attacker who wants to attack computer A can respond to an ARP request with their own device's IP address, thus poisoning the ARP cache with a false entry. Traffic sent to the original IP address is now steered to the attacker's device, where they can initiate a replay attack.

Can I detect packet sniffers on my network?

If packet sniffing software has been installed on a device within your network, then yes. To work to its full potential the packet sniffing software requires that the host computer's network interface controller is set to 'promiscuous mode'.

This means that it can pick up all network traffic rather than just that sent to its MAC address. However, this can be detected by sending a Ping with the right IP address, but the wrong MAC address for each computer on the

network.

Unfortunately, most attackers will implement a packet sniffer as a stand-alone device that acts as a test access point (TAP) for the network. These are designed to be undetectable.

What is a password hash?

Replay attacks can involve the forwarding of captured password hashes – but what exactly are these?

Password hashing involves putting a password through a one-way encryption algorithm that transforms it into a random string of numbers and letters with a fixed length. The most common algorithm used to do this is the SHA (Secure Hash Algorithm).

The current standard in use is SHA-2, which is a family of two hash functions with differing word sizes: SHA-256 uses 32-bit words where SHA-512 uses 64-bit words. Use this calculator if you'd like to see the output for any word fed into SHA-256.

Websites and apps never store passwords in their original plain text format (or at least they shouldn't). Instead, they just store the encrypted hashes of passwords. When you type in your password on a login page, the text is hashed and compared with the original password hash stored on the server. If the two hashes match, the user is logged in.

Without proper precautions, a replay attack can occur in which an encrypted hash is intercepted and sent again by the attacker, granting them access to the victim's account.
