ON2IT AND AMS-IX PRESENT: THREAT TALKS

Threat Talks
Encryption

The Hidden Dangers of Encrypted Traffic

Encrypted traffic used to be considered the safest way to browse the internet and conduct business online. The idea was simple: if data is encrypted, it's protected from prying eyes.

However, encryption is now a double-edged sword. Cybercriminals have learned to weaponize it, using it to evade detection, conceal malicious activity, and bypass security measures. Today, the very technology meant to safeguard information is being exploited to facilitate ransomware attacks, data breaches, and command-and-control operations.

With cyberthreats now lurking within encrypted channels, how do we protect ourselves?

threat-talks.com

In this Threat Talks infographic we will discuss the following threats:

- Decryption
- Public Key Infrastructure
- Post Quantum Cryptography

89.9% of encrypted threats involve malware

Source: New ThreatLabz Report Reveals Over 85% of Attacks Are Encrypted

95% + of web traffic is encrypted (2022)

Source: New ThreatLabz Report Reveals Over 85% of Attacks Are Encrypted

Only 3.5% of organizations actively decrypt network traffic

Source: Why You Need the Ability to Inspect Encrypted Network Traffic

70% + of attacks use encryption to evade detection

Source: Why You Need the Ability to Inspect Encrypted Network Traffic

Decryption

The Elephant in the Room of Network Security

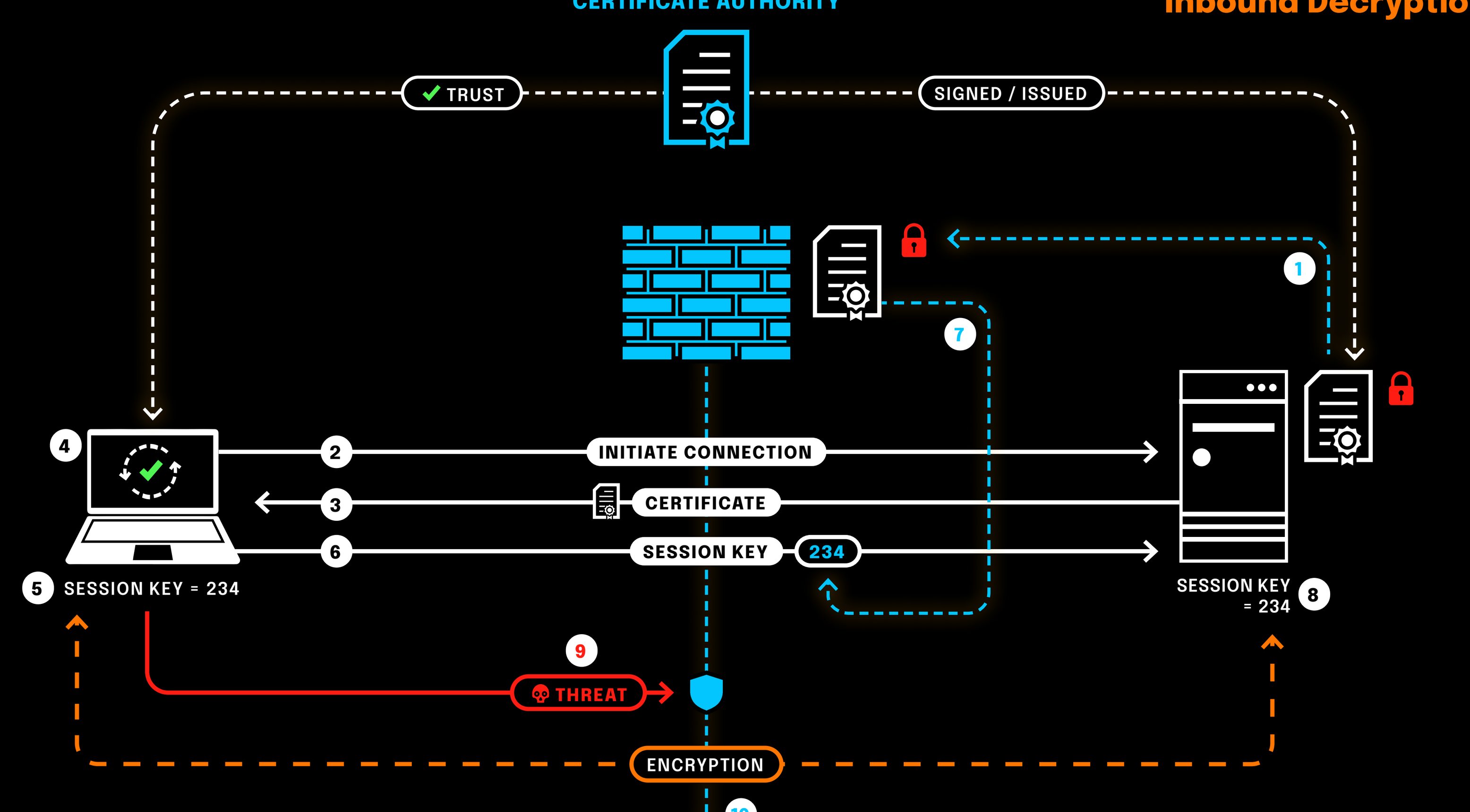
Despite the growing volume of encrypted traffic, most organizations still fail to decrypt and inspect it, leaving them blind to hidden threats. Cybercriminals exploit this blind spot to deliver malware, execute phishing attacks, and exfiltrate sensitive data undetected. While decryption is critical for network security, many hesitate due to concerns over complexity, performance, and privacy. However, without it, security tools are powerless against encrypted threats. Organizations that ignore decryption risk missing a crucial layer of protection–making it the elephant in the room of network security.

Types of Decryption

There are two primary types of network decryption:

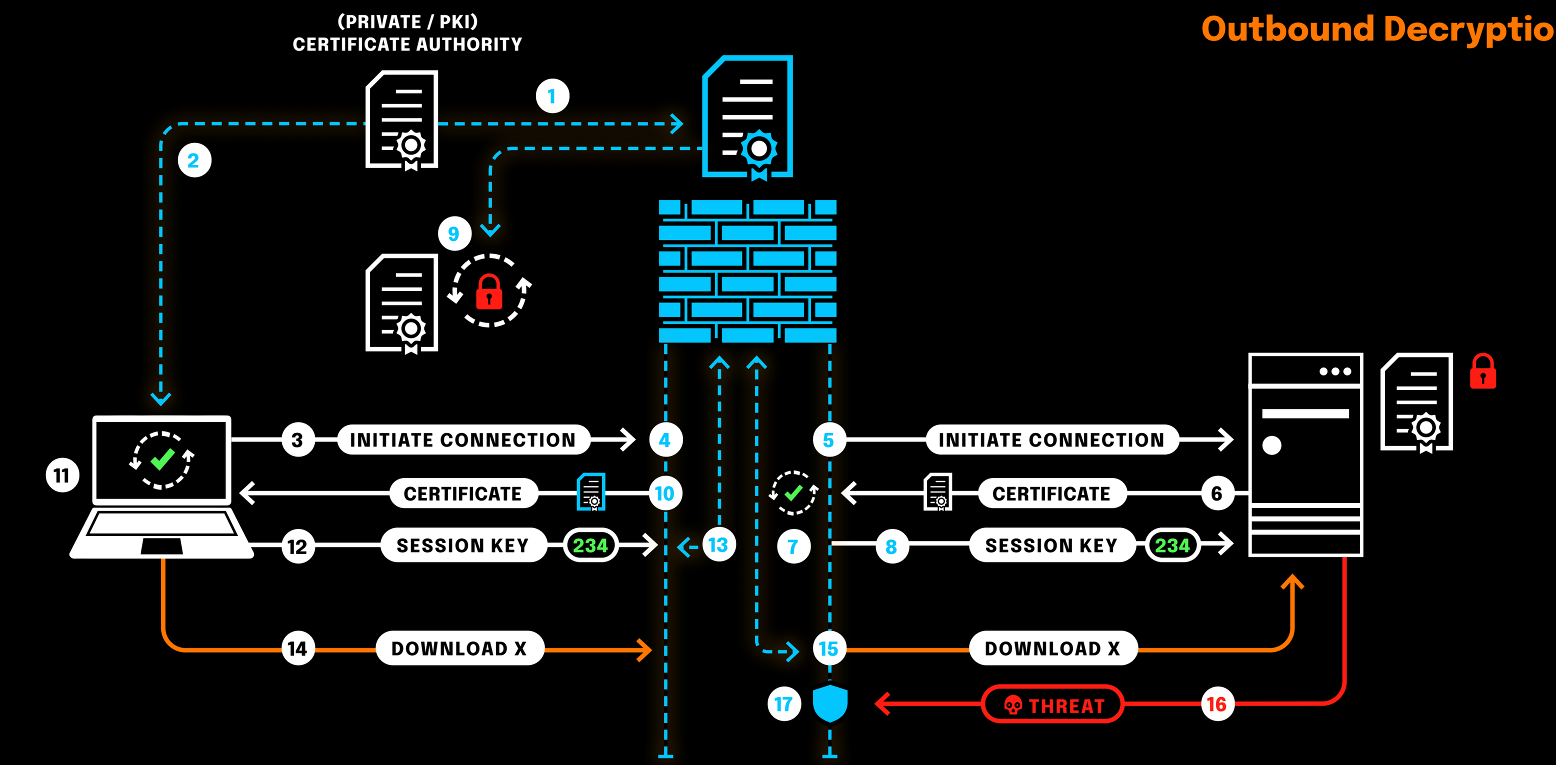
Inbound Decryption → Protect internal servers against external threats.
Outbound Decryption → Prevent internal users from accessing malicious or unauthorized content.

Inbound Decryption



1. FIREWALL — Imports the certificate and private key.
2. CLIENT — Initiates a session with the server.
3. SERVER — Sends its public certificate, containing the public key.
4. CLIENT — Validates the certificate.
5. CLIENT — Generates a session key, encrypts it with the public key, and sends it to the server.
6. SERVER — Recieves the encrypted session key.
7. FIREWALL — Uses the private key to decrypt the session key and stores it for the session.
8. SERVER — Retrieves the session key, establishing a secure connection
9. ATTACKER — Sends encrypted malicious traffic (e.g., SQL injection, XSS attack) to the server.
10. FIREWALL — Reads and inspects the encrypted traffic using the stored session key, detects the malicious payload, and blocks the session.

Outbound Decryption



1. FIREWALL — Creates a subordinate certificate authorized to sign certificates on behalf of the company's Certificate Authority (CA).
2. IT OPS — Ensures all internal clients trust this certificate authority.
3. CLIENT — Initiates a secure connection to an external web server.
4. FIREWALL — Intercepts and halts the connection.
5. FIREWALL — Initiates a separate connection to the requested web server.
6. SERVER — Sends its certificate with the public key.
7. FIREWALL — Validates the certificate and generates a session key.
8. FIREWALL — Encrypts the session key with the public key and sends it to the server.
9. FIREWALL — Creates an on-the-fly certificate mirroring the original details but signed by the subordinate CA.
10. FIREWALL — Sends this newly generated certificate to the client.
11. CLIENT — Validates the certificate and creates a session key with the firewall-generated certificate's public key.
12. CLIENT — Encrypts the session key with the firewall-generated certificate's public key.
13. FIREWALL — Receives the session key and has now established two separate encrypted sessions:
    - **Client → Firewall** (Internal secure session)
    - **Firewall → Server** (External secure session)

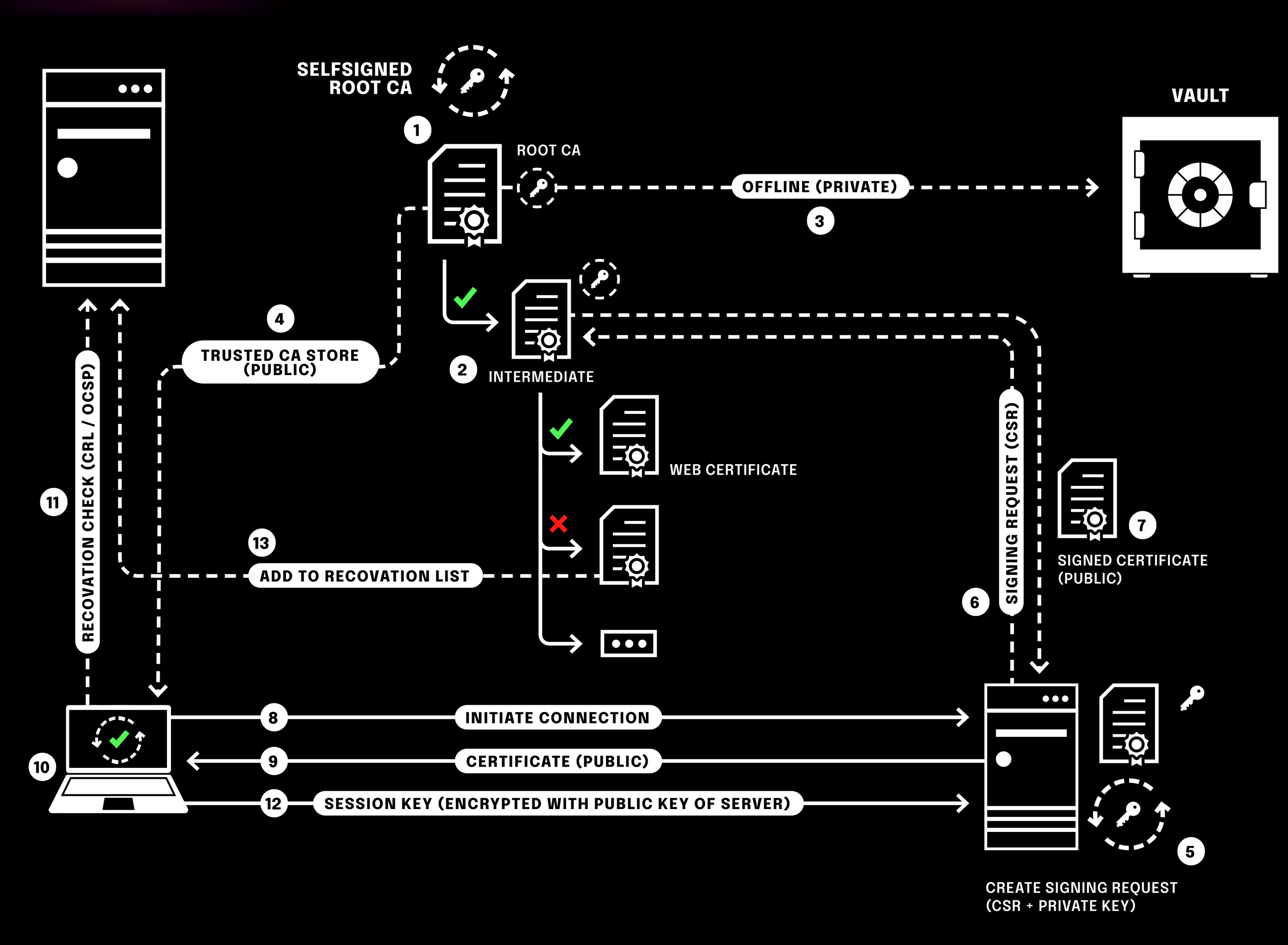
Since the firewall now sits in the middle, this is referred to as a "**man-in-the-middle**" **(MITM)** decryption approach.

14. CLIENT — Requests a file download.
15. FIREWALL — Inspects the request and forwards it to the server.
16. SERVER — Sends the file (which is injected with malware).
17. FIREWALL — Decrypts and scans the file, detects the malware, and blocks the download.

Public Key Infrastructure

The Digital Trust Framework

Public Key Infrastructure (PKI) is a setup that enables secure communication and authentication through encryption and digital certificates. It provides a way to verify the identity of entities (such as websites, users, and devices) using cryptographic keys. PKI is essential for securing both internal as external (internet) transactions, protecting sensitive data, and ensuring trust in digital interactions by establishing a verifiable chain of trust.



Setting up the PKI

1. Create a selfsigned root certificate this will be the foundation of the chain of trust.
2. Generate an intermediate certificate, which is signed by the root certificate. This certificate has the authority to issue other certificates.
3. Secure the root certificate (private key), store it offline and in a safe place, if lost the entire chain of trust is compromised.
4. Ensure client trust. Distribute the root certificate to the trusted CA store (e.g., in Windows) so clients recognize it as a valid authority.

Creating certificates

5. 5. Generate a private key and Certificate Signing Request (CSR) on the web server. This request contains the necessary details for the certificate.
6. Send the CSR to the PKI for signing. Only send the CSR, never the private key.
7. Receive the signed certificate from the PKI. This is the public part of the certificate, now signed by the intermediate CA.

Setting up an encrypted connection

8. The client initiates a connection to the webserver.
9. The webserver responds by sending the public part of the certificate including the public key.
10. The client verifies the certificate. It checks whether the entire trust chain is valid, ensuring the root CA is trusted.
11. The client will also check if the certificate is not revoked, by using Certificate Revocation List (CRL) or the Online Certificate Status Protocol (OCSP).
12. If everything is ok, the client will create a session key and send this encrypted with the public key to the webserver. They now have established a secure connection.

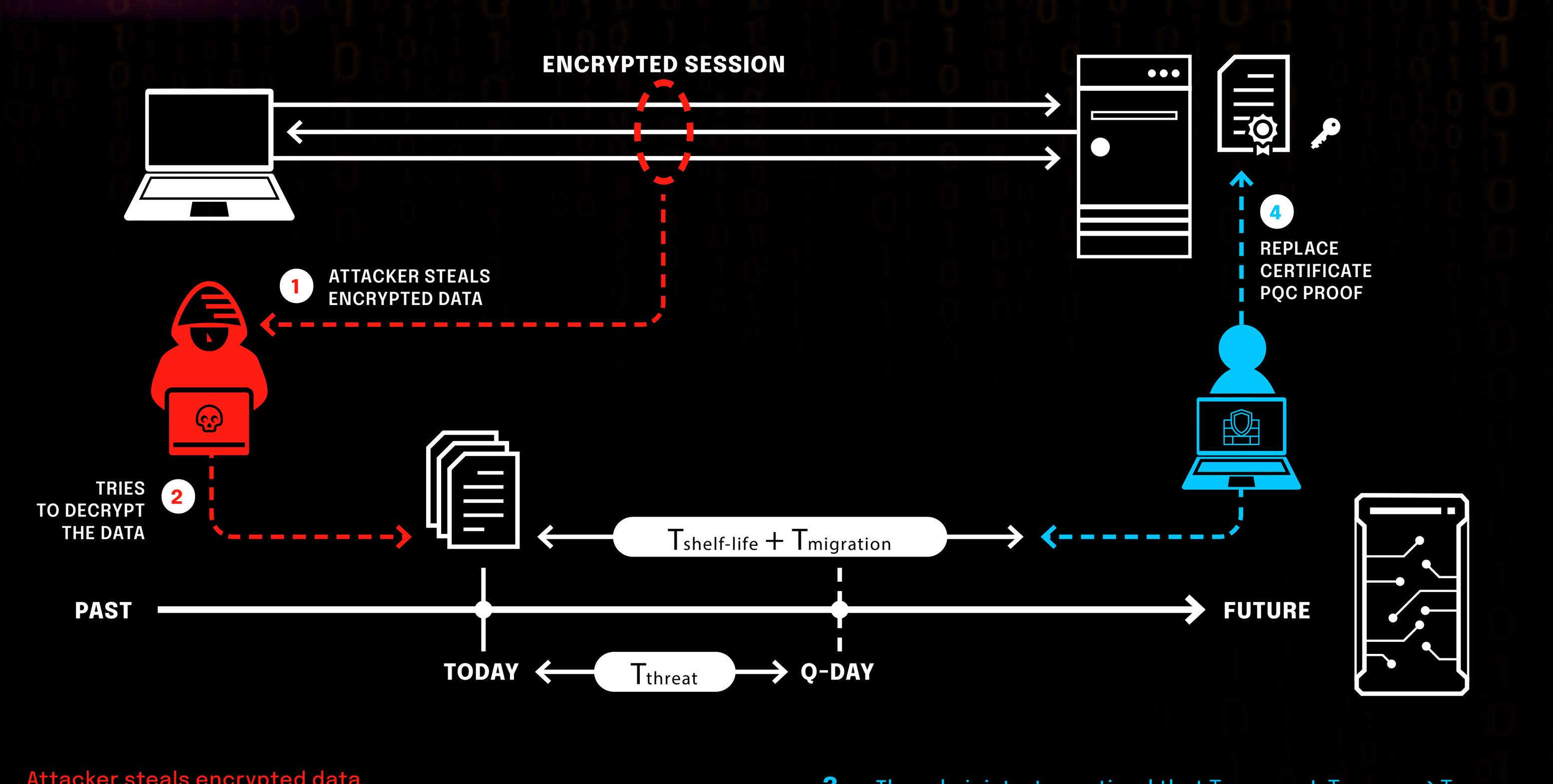
Revocation

13. Revoking a compromised certificate. If a certificate is no longer trustworthy (e.g., if its private key is leaked), the PKI can revoke it to prevent misuse.

Post Quantum Cryptography

Future-Proofing Encryption Against Quantum Threats

With the rapid advancement of quantum computing, traditional encryption methods based on RSA and ECC will eventually become vulnerable to quantum attacks. Using Michele Mosca's inequality–which compares the time it takes to migrate to quantum, you can determine whether your certificates need to be replaced. If $T_{shelf\text{-}life} + T_{migration} > T_{threat}$, it means an adversary could break your encryption before you have transitioned, making it critical to act now to safeguard sensitive data



1. Attacker steals encrypted data.
2. Attacker tries to decrypt the data, without quantum computing it takes longer than that the information needs to be secured.
3. The administrator noticed that $T_{shelf\text{-}life} + T_{migration} > T_{threat}$ and in order to secure the data it needs to take action.
4. The administrator replaces the certificate and the used cipher with a post quantum proof variant.

Q-Day
Q-Day is the anticipated moment when quantum computers become powerful enough to break widely used cryptographic algorithms such as RSA and ECC. Once Q-Day arrives, encrypted data that was previously considered secure could be instantly decrypted, exposing sensitive information. Organizations must proactively prepare by transitioning to post-quantum cryptographic algorithms before this day comes.

threat-talks.com

CONTACT US

ON2IT ZERO TRUST INNOVATORS
amsix