MANAGING RISK FROM TRANSPORT LAYER SECURITY INSPECTION

WITH GREAT POWER…

To protect enterprise data and intellectual property, network security administrators enforce encryption policies to secure traffic to and from their networks. However, adversaries also use encryption, often using it to hide their activities. Normally, these activities—like command and control, loading malware into a network, and exfiltration of sensitive data—would be detected by traffic inspection devices, but those devices typically cannot inspect encrypted traffic.

Transport Layer Security Inspection (TLSI), also known as TLS break and inspect, is a security process that allows enterprises to decrypt traffic, inspect the decrypted content for threats, and then re-encrypt the traffic before it enters or leaves the network. Introducing this capability into an enterprise enhances visibility within boundary security products, but introduces new risks. These risks, while not inconsequential, do have mitigations.

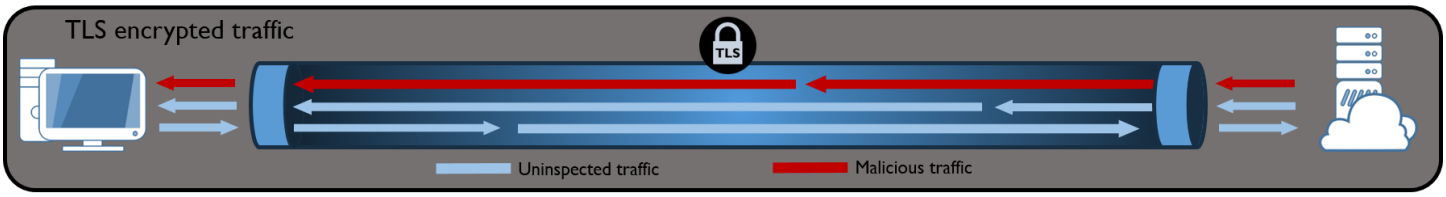

Figure 1a: Encrypted Traffic

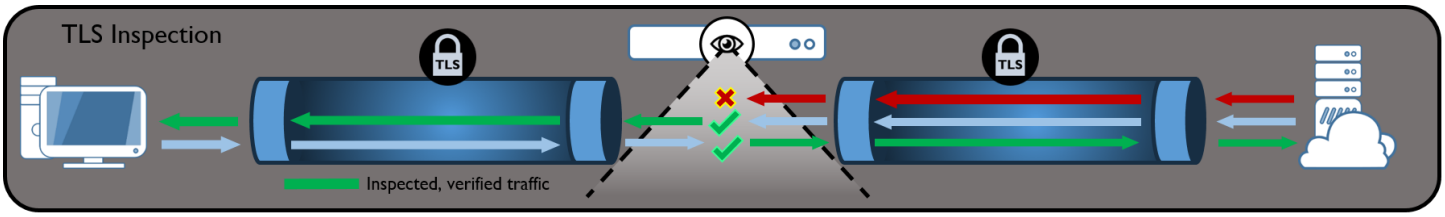

Figure 1b: TLS Inspection

a.) A TLS channel is negotiated between the client and the server. All of the data that passes through the channel is encrypted and therefore not subject to inspection.
b.) The device conducting TLSI replaces the TLS channel with a TLS chain, breaking the encryption on the data, inspecting it, and blocking it, if it is malicious.

DIGGING IN – WHAT IS TLSI?

TLSI is typically performed by a proxy device to expose the underlying plaintext of a TLS session. This enables traffic inspection devices like firewalls, intrusion detection systems, and intrusion prevention systems (IDS/IPS) to detect indicators of threat or compromise. Here, TLSI also includes the inspection of legacy Secure Sockets Layer (SSL) traffic. Discussed in detail are the three main functions of the TLSI mechanism in a forward proxy: managing forward proxy traffic flows, establishing TLS sessions, and issuing trusted certificates. Risks become apparent as the detailed mechanism TLSI employs is understood.

Forward Proxy Traffic Flows

A forward proxy is a network device that intercepts requests from internal network clients and forwards those requests to servers on external networks. When the external servers respond, the responses are sent to the forward proxy and then the forward proxy sends the responses to the internal network clients. A TLSI capability implemented within a forward proxy between the edge of the enterprise network and an external network such as the Internet protects enterprise clients from the high risk environment outside the forward proxy.

A risk associated with TLSI within a forward proxy is improper control and external processing of the decrypted traffic at or near the enterprise boundary. A forward proxy that forwards decrypted traffic to external inspection devices could misroute the traffic and result in exposing sensitive traffic to unauthorized or weakly protected networks.

Deploying firewalls and monitoring network traffic flow on all network interfaces to the forward proxy helps protect a TLSI implementation from potential exploits. Implementing analytics on the logs helps ensure the system is operating as expected. Both also help detect intentional and unintentional abuse by security administrators as well as misrouted traffic.

TLS Sessions

TLSI occurs in real-time as TLS clients establish encrypted connections to external servers. It decrypts traffic by replacing the end-to-end TLS session with a "TLS chain" consisting of two independently negotiated TLS connections: one is negotiated between an external server and the forward proxy, and the second between the forward proxy and the TLS client that attempted to initiate the TLS session to the external server. The two TLS connections allow for decision-making around how to handle the traffic (e.g. blocking, bypassing, inspecting, or forwarding traffic) in one connection before passing the traffic to the next connection. While there are two separate connections, TLS traffic flows as if there were a single connection. This TLS chaining risks a potential downgrade of TLS protection from what was accepted by the client. The TLS version or cipher suites used in one independently negotiated connection can be weaker than those negotiated for the second connection. This could result in passive exploitation of the session, or exploitation of vulnerabilities associated with weaker TLS versions or cipher suites.

TLS security settings, including version, cipher suites, and certificates, should be properly configured to prevent TLS downgrade. Disable weak TLS versions and cipher suites on the server-side. Prevent clients from forcing the usage of weak TLS versions and cipher suites. For enterprises that have clients with outdated technologies that require weak TLS versions and cipher suites, such as outdated browsers, constrain the usage of the weaker TLS security parameters so the proxy negotiates them only for exempted clients. Some TLSI vendor solutions may have features that allow weaker TLS versions and cipher suites by exception only.

Unexpected changes in TLS certificates received from external servers might indicate man-in-the-middle attacks against the proxy. Apply certificate pinning to detect unauthorized changes and alert the security administrator. Apply certificate transparency to report the unauthorized certificates to the external servers' owners.

Certification Authority

TLSI forward proxy devices incorporate a certification authority (CA) function that creates and signs new certificates that represent the external servers to the client: the CA embedded in the forward proxy issues a certificate indicating the properties of the requested external server's certificate. The TLSI system uses this certificate during the processing of TLS traffic in the connection between the TLS clients and forward proxy. The TLS clients are configured to trust the CA.

The primary risk involved with TLSI's embedded CA is the potential abuse of the CA to issue unauthorized certificates trusted by the TLS clients. Abuse of a trusted CA can allow an adversary to sign malicious code to bypass host IDS/IPSs or to deploy malicious services that impersonate legitimate enterprise services to the hosts.

The embedded CA must be protected from abuse, and remediation for potential compromise must be readily available. Issue the embedded CA's signing certificate from an external CA trusted only for TLS inspection purposes. Do not use default or self-signed certificates. Monitor enterprise traffic for unexpected and unauthorized certificates

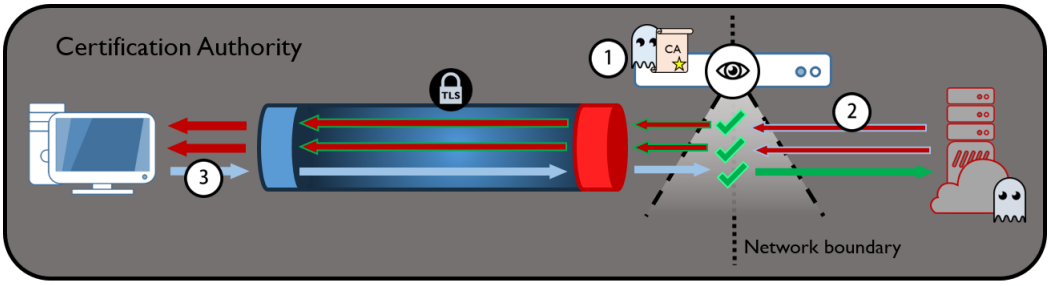

Figure 3: Compromised Certification Authority

1.) By compromising the device's embedded Certification Authority (CA), an attacker can forge certificates or validate untrustworthy certificates. 2.) If the CA is compromised, there is no way to verify where clients are receiving data from. 3.) This allows attackers to both view and exfiltrate data, and introduce malicious data. This applies to internal (non-inspected) traffic as well.

issued by the embedded CA. Enable certificate revocation services for certificates the TLSI system cached, and be able to revoke any unauthorized certificates, or the embedded CA's signing certificate itself, if unauthorized certificates are detected. Ensure the embedded CA is configured to issue only TLS server authentication certificates, as indicated in the value of their "extended key usage" field. Configure TLS clients to trust the external CA so they only trust the certificates the TLSI system issued for TLS server authentication. Issue the embedded CA with a certificate that has name constraints to reinforce limitations of the inspection authorization and prevent impersonation of enterprise services.

CONTROL ACCESS TO PLAINTEXT

A further risk of introducing TLSI is that an adversary can focus their exploitation efforts on a single device where potential traffic of interest is decrypted, rather than try to exploit each location where the data is stored. Setting a policy to enforce that traffic is decrypted and inspected only as authorized, and ensuring that decrypted traffic is contained in an out-of-band, isolated segment of the network prevents unauthorized access to the decrypted traffic.[1]

INSIDER THREATS

While TLSI allows data loss prevention tools to access the underlying plaintext of encrypted traffic, additional insider threat risk is associated with authorized security administrators responsible for managing the TLSI implementation. These authorized individuals could abuse their access to capture passwords or other sensitive data visible in the decrypted traffic.[2] Apply the principles of least privilege and separation of duties to ensure only authorized TLSI administrators have access to the data while other administrators, such as those responsible for device configuration, are prevented from accessing the data. Use a separate auditor role to detect modification of the TLSI policy and other potential administrator privilege abuse.

PRIVACY CONCERNS

In the US, enterprises operating TLSI capabilities are subject to privacy laws, policies, and regulations. Enterprises should be aware of applicable requirements and configure TLSI to prevent unauthorized exposure of data.

DO IT WELL, DO IT ONCE

To minimize the risks described above, breaking and inspecting TLS traffic should only be conducted once within the enterprise network. Redundant TLSI, wherein a client-server traffic flow is decrypted, inspected, and re-encrypted by one forward proxy and is then forwarded to a second forward proxy for more of the same, should not be performed. Inspecting multiple times can greatly complicate diagnosing network issues with TLS traffic. Also, multi-inspection further obscures certificates when trying to ascertain whether a server should be trusted. In this case, the "outermost" proxy makes the decisions on what server certificates or CAs should be trusted and is the only location where certificate pinning can be performed. Finally, a single TLSI implementation is sufficient for detecting encrypted traffic threats; additional TLSI will have access to the same traffic. If the first TLSI implementation detected a threat, killed the session, and dropped the traffic, then additional TLSI implementations would be rendered useless since they would not even receive the dropped traffic for further inspection. Redundant TLSI increases the risk surface, provides additional opportunities for adversaries to gain unauthorized access to decrypted traffic, and offers no additional benefits.

IMPLEMENTATION CHALLENGES

Many TLSI products cut corners to meet performance requirements. Choose products that are independently validated to properly implement data flow, TLS, and CA functions. NSA recommends products validated by National Information Assurance Partnership (NIAP)[3], and configured according to the vendor's instructions used during validation.

---

[1] Refer to "Segment Networks and Deploy Application-Aware Defenses," a Cybersecurity Top 10 Mitigations document, for more information.

[2] For information on privilege abuse mitigations, refer to "Defend Privileges and Accounts," a Cybersecurity Top 10 Mitigations document.

[3] For more information, please refer to the Protection Profile Module for SSL/TLS Inspection Proxy: www.niap-ccevs.org/Profile/PP.cfm

Some TLS protected applications are incompatible with simple TLSI implementations that do not also address application security features. Network clients using such applications may receive error messages or have their sessions unexpectedly drop or hang. If the TLSI implementation cannot properly inspect TLS sessions protecting these applications, the sessions should be bypassed or blocked, according to the risk associated with the traffic. For example:

- TLS 1.3 implements restrictions that do not allow certain shortcuts commonly used in TLSI products. TLSI can cause sessions to fail for applications that use TLS 1.3 exclusively.
- External servers requiring client authenticated TLS will not trust the TLSI's signing certificate and will reject sessions using client certificates issued by the embedded CA.
- HTTP Strict Transport Security (HSTS) includes a security feature that binds the HTTP session to the specific TLS session used. TLSI systems that ignore the underlying HTTP headers will cause HSTS sessions to be rejected by the client application, the server, or both.
- TLSI can also cause sessions to fail in unexpected ways if they use client-level certificate pinning, where the application requires a specific certificate in the TLS session. Certificate pinning is commonly used for automated software updates.

Once TLSI is implemented, security administrators must also consistently manage the implementation to ensure that legitimate network processing is not disrupted due to unintended blocking; administrators must balance usability with security. Conduct education and awareness campaigns to inform employees that they may not have access to high risk websites which were previously accessible prior to the TLSI implementation. Set up a corporate help desk to support employees experiencing issues accessing necessary websites. Some TLSI vendor solutions provide additional features for enhancing usability, such as bypassing traffic for known incompatible applications. Enterprises should enable these usability features when needed.

Network owners should be aware that TLS inspection is not a cure-all. It can only inspect SSL and TLS traffic where the proxy's certificate is trusted by clients (and servers for mutual authentication). While some break and inspect devices can block TLS sessions that do not allow inspection, this could also disrupt legitimate activity.

RISK, MINIMIZED

Security administrators cannot protect against what they cannot see. The latest tactics, techniques, and procedures (TTPs) have allowed attackers to leverage encrypted traffic to sneak past traffic monitoring tools. Security professionals have fought against these TTPs through the use of TLSI. TLSI capabilities implemented in enterprise forward proxies can provide visibility into encrypted network traffic to detect adversarial use of encryption, but the devices that break and inspect the TLS traffic may become high priority targets for exploitation and introduce additional risks into an enterprise network. Enterprises must carefully weigh these risks against the benefits and, if TLSI is implemented, address those risks. Moreover, while applications incompatible with TLSI may cause users to experience latency and errors, ongoing management and support helps administrators balance usability with security. The mitigations described above can reduce the risks introduced by a TLSI capability, provide indicators that alert administrators if the TLSI implementation may have been exploited, and minimize unintended blocking of legitimate network activity. In this way, security administrators can successfully add TLSI to their arsenal and continue to step up their methods to combat today's adversaries and TTPs.

DISCLAIMER OF WARRANTIES AND ENDORSEMENT

CONTACT INFORMATION

Client Requirements / General Cybersecurity Inquiries: Cybersecurity Requirements Center (CRC), 410-854-4200, Cybersecurity_Requests@nsa.gov
Media inquiries: Press Desk, 443-634-072, MediaRelations@nsa.gov