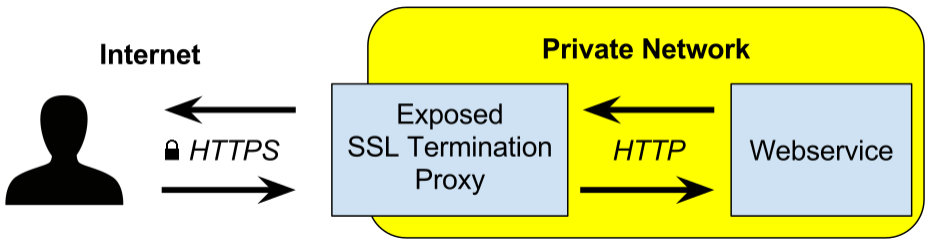
TLS termination proxy

A **TLS termination proxy** (or **SSL termination proxy**,[1] or **SSL offloading**[2]) is a proxy server that acts as an intermediary point between client and server applications, and is used to terminate and/or establish TLS (or DTLS) tunnels by decrypting and/or encrypting communications. This is different to **TLS pass-through proxies** that forward encrypted (D)TLS traffic between clients and servers without terminating the tunnel.



Incoming HTTPS traffic gets decrypted and forwarded to a web service in the private network.

Uses

TLS termination proxies can be used to:

- secure plaintext communications over untrusted networks by tunnelling them in (D)TLS,
- allow inspection of encrypted traffic by an intrusion detection system to detect and block malicious activities,
- allow network surveillance and analysis of encrypted traffic,
- enable otherwise unsupported integration with other applications that provide additional capabilities such as content filtering or Hardware security module,
- enable (D)TLS protocol versions, extensions, or capabilities (e.g. OCSP stapling, ALPN, DANE, CT validation, etc.) unsupported by client or server applications to enhance their compatibility and/or security,
- work around buggy/insecure (D)TLS implementations in client or server applications to improve their compatibility and/or security,
- provide additional certificate-based authentication unsupported by server and/or client applications or protocols,
- provide an additional defence-in-depth layer for centralised control and consistent management of (D)TLS configuration and associated security policies, and
- reduce the load on the main servers by offloading the cryptographic processing to another machine.

Types

TLS termination proxies can provide three connectivity patterns:[3]

- **TLS Offloading** of inbound encrypted (D)TLS connection from a client and forwarding communications over a plain text connection to the server.
- **TLS Encryption** of inbound plaintext connection from a client and forwarding communications over an encrypted (D)TLS connection to the server.
- **TLS Bridging** of two encrypted (D)TLS connections to allow inspection and filtering of encrypted traffic by decrypting inbound (D)TLS connection from a client and re-encrypting it with another (D)TLS connection to the server.

Combining a TLS Encrypting proxy in front of a client with a TLS Offloading proxy in front of a server, can allow (D)TLS encryption and authentication for protocols and applications that don't

otherwise support it, with two proxies maintaining a secure (D)TLS tunnel over untrusted network segments between client and server.

A proxy used by clients as an intermediary gateway for all outbound connections is typically called a Forward proxy, while a proxy used by servers as an intermediary gateway for all inbound connections is typically called a Reverse proxy. Forward TLS bridging proxies that allow intrusion detection system to analyse all client traffic are typically marketed as "SSL Forward Proxy".[4][5][6]

TLS Offloading and TLS Bridging proxies typically need to authenticate themselves to clients with a digital certificate using either PKIX or DANE authentication. Usually the server operator supplies to its reverse proxy a valid certificate for use during (D)TLS handshake with clients. A forward proxy operator, however would need to create their own private CA, install it into the trust store of all clients and have the proxy generate a new certificate signed by the private CA in real time for each server that a client tries to connect to.

When network traffic between client and server is routed via a proxy, it can operate in transparent mode by using the client's IP address instead of its own when connecting to the server and using the server's IP address when responding to the client. If a **Transparent TLS Bridging Proxy** has a valid server certificate, neither client nor server would be able to detect the proxy presence. An adversary that has compromised the private key of the server's digital certificate or is able to use a compromised/coerced PKIX CAs to issue a new valid certificate for the server, could perform a man-in-the-middle attack by routing TLS traffic between client and server through a Transparent TLS Bridging Proxy and would have the ability to copy decrypted communications, including logon credentials, and modify content of communications on the fly without being detected.

References

1. *SSL Termination (https://f5.com/glossary/ssl-termination)* , *F5 Networks*.
2. *"Setup IIS with URL Rewrite as a reverse proxy" (https://docs.microsoft.com/fr-fr/archive/blogs/friis/setup-iis-with-url-rewrite-as-a-reverse-proxy-for-real-world-apps)* . *Microsoft*.
3. *"Infrastructure Layouts Involving TLS" (https://www.haproxy.com/documentation/hapee/latest/deployment-guides/tls-infrastructure/)* . *HAProxy Technologies*.
4. *"SSL Forward Proxy Overview" (https://www.juniper.net/documentation/en_US/junos-space15.2/topics/concept/junos-space-ssl-forward-proxy-overview.html)* . *Juniper Networks*.
5. *"SSL Forward Proxy" (https://www.paloaltonetworks.com/documentation/71/pan-os/pan-os/decryption/ssl-forward-proxy)* . *Palo Alto Networks*.

6. *"Overview: SSL forward proxy client and server authentication" (https://support.f5.com/kb/en-us/products/big-ip_ltm/manuals/product/bigip-ssl-administration-11-6-0/13.html)* . *F5 Networks.*

Last edited 1 year ago by MaxEnt