# SSL and TLS: Theory and Practice

# Chapter 5 – Firewall Traversal

Rolf Oppliger

May 28, 2023

# Terms of Use

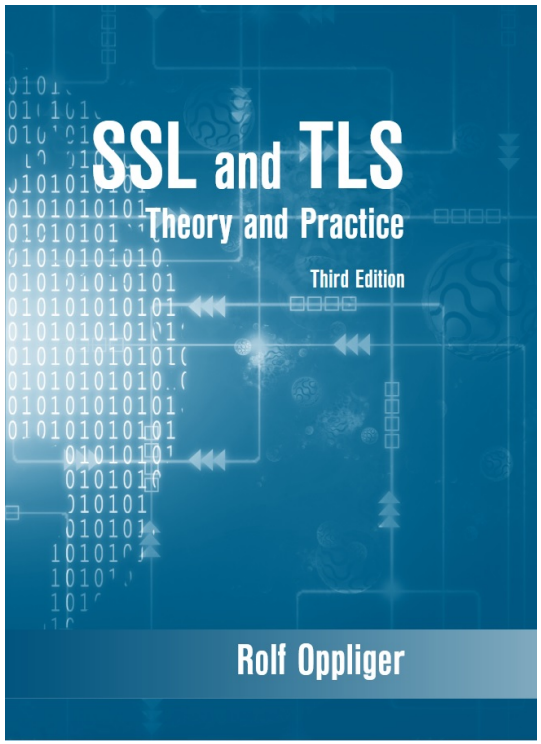
- This work is published with a CC BY-ND 4.0 license (  )
  - CC = Creative Commons ()
  - BY = Attribution ()
  - ND = No Derivatives ()

# whoami

rolf-oppliger.ch
rolf-oppliger.com

- eSECURITY Technologies Rolf Oppliger (founder and owner)
- Swiss National Cyber Security Centre NCSC (scientific employee)
- University of Zurich (adjunct professor)
- Artech House (author and series editor for information security and privacy)

# Reference Book





ISBN 978-1-68569-015-1

https://www.esecurity.ch/Books/ssltls3e.html

# Challenge Me

# Outline

## 5. Firewall Traversal

# 5. Firewall Traversal

# 5. Firewall Traversal

## 5.1 Introduction

- There are many possibilities to define the term **Internet firewall**, or **firewall** in short
- According to RFC 4949, it refers to "an internetwork gateway that restricts data communication traffic to and from one of the connected networks (the one said to be 'inside' the firewall) and thus protects that network's system resources against threats from the other network (the one that is said to be 'outside' the firewall)"
- This definition is fairly broad and not precise in mathematical terms

# 5. Firewall Traversal

## 5.1 Introduction

- In the early days of the firewall technology, Bill Cheswick and Steve Bellovin defined a firewall (system) as a collection of components placed between two networks that have the following three properties
  1. All traffic from inside to outside, and vice versa, must pass through the firewall
  2. Only authorized traffic, as defined by the local security policy, is allowed to pass through
  3. The firewall itself is immune to penetration

# 5. Firewall Traversal

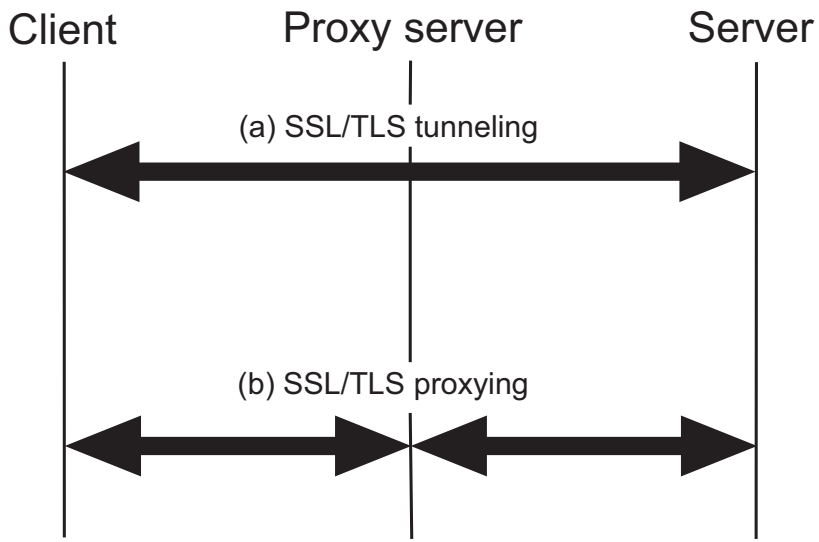
## 5.1 Introduction

- There are many technologies that can be used (and combined) to implement a firewall
  - Static packet filtering
  - Dynamic packet filtering (aka stateful inspection)
  - Circuit-level gateways
  - Application-level gateways
- These technologies can be combined and configured in many ways (e.g., dual-homed firewall, screened subnet firewall, . . . )
- Firewalls can be operated in a centralized or decentralized way (i.e., personal firewalls)

# 5. Firewall Traversal

## 5.1 Introduction

- Different application protocols may have different requirements with regard to proxy servers
- An application protocol can either be proxied or tunneled



# 5. Firewall Traversal

## 5.1 Introduction

- In the past, it has been been common practice for companies and organizations to
  - Tunnel outbound SSL/TLS connections
  - Proxy inbound SSL/TLS connections
- This practice is about to change as the deployment settings are getting more involved, and content screening and data loss prevention (DLP) are required

# 5. Firewall Traversal

## 5.2 SSL/TLS Tunneling

- SSL tunneling is enabled by the HTTP CONNECT method
- The method can be used to establish an end-to-end tunnel across an HTTP proxy server
- It is invoked by `CONNECT www.esecurity.ch:443 HTTP/1.0`
- It can be combined with the "normal" authentication and authorization mechanisms (e.g., HTTP Basic or Digest authentication)

# 5. Firewall Traversal

## 5.2 SSL/TLS Tunneling

- SSL/TLS tunneling has a few practical disadvantages
- Many things that are possible with unencrypted data become impossible if data is end-to-end encrypted
- The proxy server cannot even ensure that a particular application protocol (e.g., HTTP) is used on top of SSL/TLS
- The HTTP proxy server can control neither the protocol in use nor the data that is being transmitted
- In some application settings, this level of ignorance is dangerous and cannot be accepted

# 5. Firewall Traversal

## 5.3 SSL/TLS Proxying

- SSL/TLS proxying mandates the following procedure
  1. The user has his or her client establish a first SSL/TLS connection to the proxy server
  2. The proxy server may authenticate and authorize the client
  3. The proxy server establishes a second SSL/TLS connection to the origin server
  4. The proxy server then mediates data between the two SSL/TLS connections, optionally doing content screening and caching

# 5. Firewall Traversal

## 5.3 SSL/TLS Proxying

- The distinguishing feature of an SSL/TLS proxy server is that it terminates all SSL/TLS connections and hence that no SSL/TLS tunneling occurs
- This makes everything transparent to the proxy server
- It also means that end-to-end security cannot be achieved and that the proxy server yields a (legitimate) MITM
- Due to its ability to intercept data traffic, an SSL/TLS proxy server is often called interception proxy or middlebox
- Middleboxes are omnipresent and pose many problems in the field

# 5. Firewall Traversal

## 5.3 SSL/TLS Proxying

- When the client establishes a first SSL/TLS connection to the proxy server, then the proxy server must authenticate itself to the client
- It typically sends a CERTIFICATE message to the client as part of the handshake
- This message comprises a server certificate that can be verified by the client
- Either the proxy server shares the origin server's private key and certificate, or — more likely — it must be able to issue a suitable certificate on the fly

# 5. Firewall Traversal

## 5.3 SSL/TLS Proxying

- Backed by an 2015 CERT Coordination Center (CERT/CC) post, the U.S. Cybersecurity & Infrastructure Security Agency (CISA) issued a security alert about HTTPS interception weakening TLS security in 2017
- Most importantly, the alert warns about the use of HTTPS inspection products that do not correctly perform certificate validation
- This weakens the end-to-end security that HTTPS originally aims to provide

# 5. Firewall Traversal

## 5.4 Middlebox Mitigation

- Instead of SSL/TLS tunneling, SSL/TLS proxying is often required in the field
- This means that SSL/TLS is not really an end-to-end security technology
- Rather, it is an end-to-middlebox or even end-to-first-middlebox security technology
- What is going on behind the (first) middlebox is not necessarily visible from the end

# 5. Firewall Traversal

## 5.4 Middlebox Mitigation

- A generalized setting with $n$ client-side middleboxes (CSM) and $m$ server-side middleboxes (SSM)

# 5. Firewall Traversal

## 5.4 Middlebox Mitigation

- There are two obvious technologies to make sure that a client is connected to an origin server (instead of a middlebox)
  - Certificate pinning
  - Ensure that the client and origin server use the same TLS connection (e.g., using keying material exporters)
- Unfortunately, both technologies are difficult to administer and have severe disadvantages when used in the field
- Consequently, there is a lot of research and development to find possibilities to mitigate middleboxes

# 5. Firewall Traversal

## 5.4 Middlebox Mitigation

- Solution categories
  1. TLS extensions and specifically crafted certificates, such as proxy certificates, delegated credentials, and many more
  2. Special encryption technologies, such as searchable or order-preserving encryption
  3. Confidential Computing, i.e., trusted execution envrionments (TEEs) and secure enclaves
- None of these solutions is ready for prime time
- The solutions from the first category seem to be the most promising ones

# 5. Firewall Traversal

## 5.4 Middlebox Mitigation

- Due to the trend of enforcing forward secure key exchange methods only, i.e., (EC)DHE, the use of middleboxes has become more challenging
- This became obvious in the standardization process of TLS 1.3
- Many organizations (mainly from the banking industry) opposed to the banishment of static keys
- They managed to have a version of TLS that still supports static keys standardized by the European Telecommunications Standards Institute (ETSI)

# 5. Firewall Traversal

## 5.4 Middlebox Mitigation

- The protocol was first named enterprise TLS (eTLS), and was later renamed to Enterprise Transport Security (ETS)
- ETS is part of a middlebox security protocol (MSP) series — aka transport layer MSP (TLMSP)
- Due to its inability to provide forward secrecy, eTLS/ETS/ MSP/TLMSP is controversially discussed in the community
- It is unlikely that it will prevail in the long term
- CVE-2019-9191 states that the protocol yields a vulnerability because it does not provide forward secrecy

# 5. Firewall Traversal

## 5.4 Middlebox Mitigation

- An interesting situation occurs in the realm of content delivery networks (CDNs), such as the ones provided by Akamai or CloudFlare
- If an origin server is operated inside a CDN, then SSL/TLS proxying should be enforced by the edge servers
- This means that the edge servers must have access to the private keys of the origin servers
- Either the keys are deposited and securely stored on the edge servers, or the keys are made otherwise available and accessible to them (e.g., Keyless SSL for CloudFlare)

# 5. Firewall Traversal

## 5.5 Final Remarks

- There are two possibilities for SSL/TLS to (securely) traverse a firewall, i.e., SSL/TLS tunneling and SSL/TLS proxying
- From a security perspective, SSL/TLS proxying is the preferred choice
- There are many web application firewalls (WAFs) that represent SSL/TLS proxy servers with complementary features, such as content inspection and DLP
- The design and implementation of WAFs that are resistant to various types of attacks is a timely and important topic

# 5. Firewall Traversal

## 5.5 Final Remarks

- Today, many companies and organizations still use SSL/TLS tunneling for oubound connections and SSL/TLS proxying for inbound connections
- But due to content-driven attacks (e.g., malware), this practice is about to change
- Many security professionals opt for proxying outbound SSL/TLS connections, as well
- Depending on the application setting, this can be simple or difficult to deploy and put in place
- Proxying is particularly challenging for DTLS

# Questions and Answers

# Thank you for your attention