

NetSec by Proxy

A MODEL FOR ENCRYPTED TRAFFIC MANAGEMENT ON
AN INTERNAL NETWORK

Evan Sayles

COMP 116, FALL 2016 | PROFESSOR MING CHOW | MENTOR EDWARD RAYMOND

Abstract

This paper will discuss the SSL/TLS encryption model and how the security it offers benefits both everyday internet users and criminals. I will explore, explain, and discuss the implications of one model of encrypted traffic management which restores the ability of system administrators and security professionals to monitor the contents of encrypted network traffic in real-time as it passes into and out of their networks, without causing out-of-network loss of security for users. This model uses a specialized piece of hardware sitting in front of a server or network.

Introduction

Much of modern-day internet traffic is now being transferred using Secure Socket Layer (SSL) or Transport Layer Security (TLS) network communications. This traffic is encrypted such that no third party can view or efficiently access the data being transferred between two computers, preventing theft of information or modification of the data while it is in transit. It is difficult to break this security without having direct access to one of the endpoint machines. This is certainly a positive for everyday internet users: their data is much safer in transit, so identity theft and certain invasions of privacy become much less likely.

However, SSL/TLS also serves cyber-criminals. Their traffic is encrypted as well, so it is possible to send malicious traffic across the internet with a lower risk of discovery. Many, cyber-crimes now occur over encrypted channels such as SSL/TLS. System administrators and security professionals, who would otherwise be able to monitor traffic matching certain patterns such as database information exfiltration or remote code execution, are not able to view that data as it passes in and out of their networks.

SSL/TLS cannot be quickly decrypted by anyone besides the intended recipient, so trying to break encryption on a network scale is not a feasible way of maintaining security. However, there are secure models which make monitoring of encrypted traffic feasible while also maintaining security for users as their data travels through the network and across the internet. This paper will discuss one model and its security implications.

To the Community

Encryption of our web traffic over protocols like HTTPS, which uses SSL/TLS, has greatly improved the security of our personal information as it goes from our devices to the services we use online. However, criminals will still be able to access our information by gaining access to either our devices or the servers which hold our data, and they will use the same protected, encrypted protocols to transfer that data to themselves or some paying customer. We need a way to efficiently monitor the transfer of data to protect ourselves. However, it's imperative that we avoid the transmission of decrypted data between computers at all – even within an internal network. The model I discuss in this paper is excellent because all its work is done locally so that all traffic before and after the monitoring stage is safely encrypted. The protection of data in transit is imperative, and this model keeps the data safe while also ensuring that the data being passed into and out of the network is not maliciously intended.

Encrypted Traffic Management

SSL and TLS overview

SSL and TLS offer secured means of communication between a client and a server. Their models are very similar, but TLS was launched later to address security holes in SSL. Throughout this paper I will refer to these two suites as SSL/TLS, unless I am specifying one of them specifically. To establish an encrypted means of communication, a client and server perform an SSL/TLS handshake where they decide on a cipher to use for the session, compute keys, and verify that they can successfully communicate using the decided-upon hashes and keys. If all these steps are performed correctly, the client and server can use their hashes and keys to communicate data for the remainder of the session.

SSL/TLS offers several security benefits:

- A third party monitoring traffic using a tool like WireShark can only see encrypted gibberish as traffic on the network.
- A third party cannot modify data being transmitted because it does not know the private keys of either side.
- Certificate authorities prevent many phishing schemes by ensuring users are truly visiting the sites they think they are.

However, it is not a perfect security model:

- There are several known attacks against both SSL and TLS, like forcing a downgrade or using malicious embedded JavaScript in web sites to leak information.
- SSL/TLS are not uniformly universally used on all websites, and different versions mean users are subjected to varying levels of security as they navigate the internet.

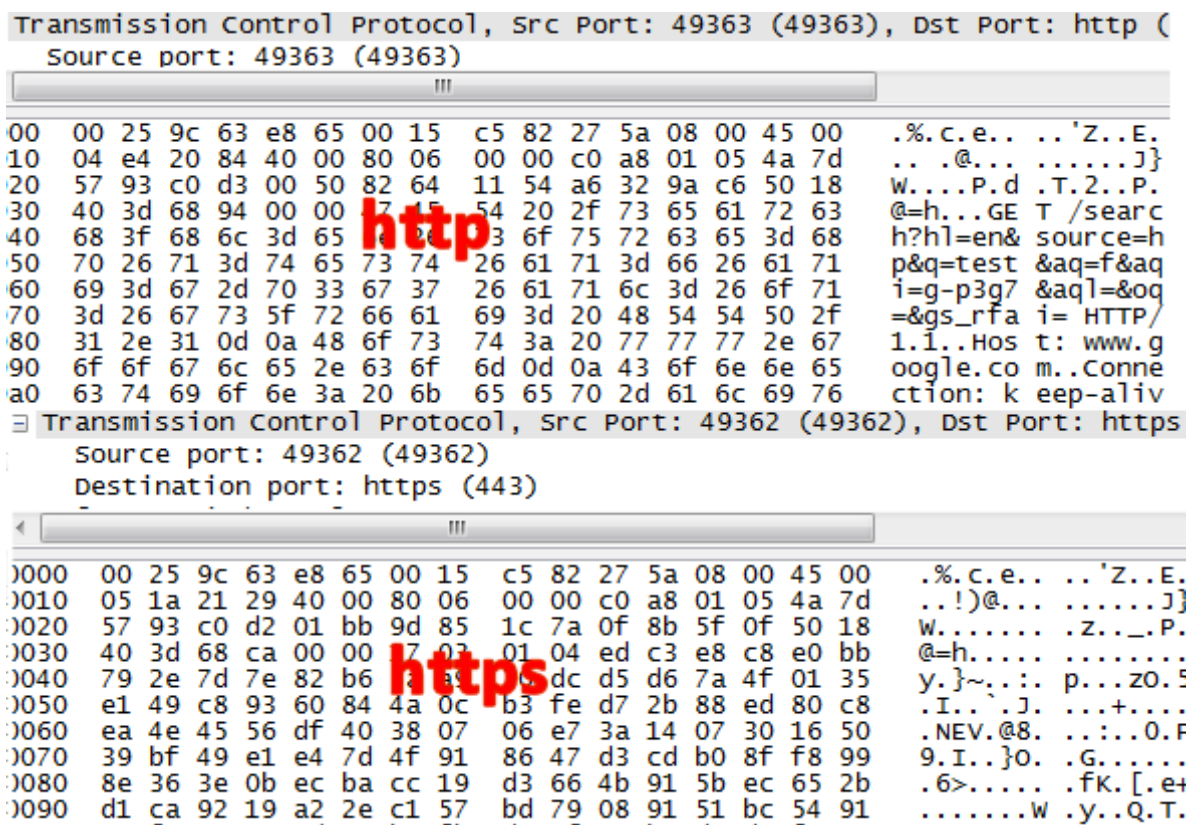


Figure 1. Captured packets over unsecured HTTP at the top, compared to SSL-encrypted packets over HTTPS.

Breaking SSL/TLS

Trying to break communications over SSL/TLS using a brute-force method would take eons. Since SSL/TLS chooses from multiple secure algorithms, an attacker must first determine which algorithm is being used, then proceed to try to break it.

To be able to decrypt messages over SSL/TLS, an attacker needs access to the device receiving the data, and because forward secrecy is guaranteed in modern TLS specifications, the attacker would need access to the device while the communication is happening. Private keys are the only effective way of decrypting messages sent over SSL/TLS.

Quickly and Securely Decrypting Network Traffic

In this section, I will discuss one model of monitoring encrypted traffic, implemented using a specialized piece of hardware sitting on a network, called an SSL Visibility Appliance.

The SSL Visibility Appliance (SVA) sits in front of a network, such that all traffic into and out of that network passes through it. When a client (a user inside the network) starts an SSL/TLS handshake with an outside server, the SVA intercepts that handshake transparently, forming an SSL/TLS session with the user and also a unique SSL/TLS session with the target server. Because the SVA sits between two SSL/TLS sessions, it can (and must) decrypt the data being sent through it from one endpoint before re-encrypting it and sending it to the other. It can then use that decrypted data with any number of security platforms connected to it in a local “daisy-chain” for analysis, monitoring or filtering.

Because the data is only decrypted locally, it is always safe while in transit (assuming no adversary has access to a platform using data decrypted by the SVA).

Because the data is being intercepted the SVA, which is not the final destination of the client's request, the SVA must issue a digital certificate to the client. Clients on the network must be configured to trust the certificate issued by the SVA, because otherwise the appliance would be considered an unauthorized computer attempting a man-in-the-middle attack. This means appliances like the SVA can be used to decrypt and analyze data passing into and out of one network, but not the whole internet. No actor outside the network is capable of performing the same kind of interception without somehow forging the digital certificate. Thus, users can feel more confident their data is safe while in transit over the internet.

Conclusion and Action Items

Encrypted communication is getting more and more common online. This offers tremendous amounts of security to users worldwide, but also creates new opportunities for criminals to act maliciously in the shadows of cipher text. Decrypting SSL/TLS traffic is challenging to impossible without direct access to one of the machines doing the sending or receiving, but appliances such as the SVA serve as a useful model for how this traffic can still be securely monitored on a network level. Everyday people and companies are safer, and criminals are forced to work that much harder to try to gain access to information they should not have.

References

Figure 1 via <https://googlesystem.blogspot.com/2010/05/google-secure-search.html>

SSL Support Team, SSL.com . "Pros And Cons Of SSL / HTTPS / TLS - SSL.com." *SSL.com*. N.p., n.d. Web. 14 Dec. 2016.

"SSL Visibility Appliance Overview." *SSL Visibility Appliance Overview*. N.p., n.d. Web. 14 Dec. 2016.

Ristic, Ivan. *Bulletproof SSL and TLS*. London: Feisty Duck, 2014. Print.

"SSL/TLS Overview." Stanford, CA: Stanford Secure Computer Systems Group, 2004. Online PDF. 13 Dec. 2016.