Exercise 1 question 1.4
Limitations and Differences of using IPsec, TLS/SSL or SSH as VPN-solution

Ole Martin Dahl [ole.dahl@hig.no]

October 29, 2004

Abstract

Virtual private networks (VPNs) [1] [6] provide low-cost and secure access between hosts and/or networks. IPsec, TLS/SSL and SSH are popular technologies used to create VPNs. This article will point out some of the differences and limitations of using IPsec, TLS/SSL or SSH as VPN-solution.

Introduction

When selecting a VPN-solution three basic requirements must be assessed:

Confidentiality Is necessary to protect the information that is being sent between the communicating parties. A strong encryption algorithm is necessary to prevent an eavesdropper in reading confidential information in clear text.

Integrity It is important to verify that the received information is the same as it was when it was sent to you. In the digital world this is solved through digital signatures and hash functions.

Authentication It is necessary to verify that the information has come from whom it is supposed to, and that it is received by who is supposed to receive it, i.e. mutual authentication.

All these requirements can be offered with IPsec[3], TLS/SSL[5] or SSH[4], but each requirement is not always necessary to fulfill in every context. In some cases the integrity requirement is the most important or in other cases confidentiality and getting perfect forward secrecy (PFS)[1] is required. When choosing either IPsec, TLS/SSL or SSH care must be taken to achieve the required security from the protocol, each protocol can be configured to match different requirements.

The main problems with VPN-solutions have been and are implementation issues, processing overhead and packet overhead. IPsec, TLS/SSL and SSH all have such problems, but to different extents.

[1] In the attribute PFS lies that if a long term secret in a protocol is compromised an attacker will not be able to decrypt previously made encrypted communication

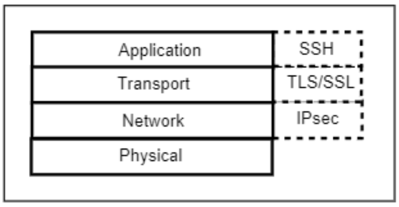

Figure 1: TCP model with SSH, TLS/SSL and IPsec

Main differences with IPsec, TLS/SSL and SSH

In the TCP networking model[2], with it's four layers, application, transport, network and physical layer, the different technologies IPsec, TLS/SSL and SSH has it's individual place [6] as described in figure 1.

SSH fits in at the top of the model at the application layer. This makes SSH an application by nature and work beside other network applications like ftp, http and others. SSH can be used in a port-forwarding mode to create a tunnel for other applications.

TLS/SSL provides security for the transport layer. TLS/SSL is not a single application like SSH, but provide security through implementation into applications. SSL was designed by Netscape[2] with HTTP usage in mind. TLS is the latest version of the SSL technology.

IPsec provides security at the IP packet layer; it is not integrated at higher levels like TLS/SSL. IPsec is a network-level protocol incorporated into servers and/or clients, e.g. into a router, dedicated VPN concentrator, a firewall or into an operating systems' kernel.

It is important to remember that there is no interoperability between SSH, TLS/SSL and IPsec, they all operate in different levels in the TCP model and are designed with different uses in mind. Of course it is possible to tunnel traffic through SSH running through TLS on a trusted network using IPsec, but it is not practical since the traffic is then encrypted and decrypted three times and uses a huge amount of CPU bandwidth. Selecting one of the technologies is normal.

SSH

SSH [7] is a very inexpensive, in fact it is free for non-commercial use and costs little for commercial use. SSH is available in two versions SSH-1 and SSH-2. SSH-2 is the latest and most secure version. SSH-1 is still very popular (SSH-1 can be found as GPL license for all major platforms) but have some limitations in features and it has some dangerous security issues, e.g. uses a CRC for integrity protection which is not secure. SSH have high availability and runs on almost every platform. SSH-2 support a lot of encryption algorithms like 3DES, IDEA, Blowfish, Twofish and Cast. SSH VPN in its simplest form uses the capabilities of SSH to tunnel service ports across the Internet inside an SSH session. Although it has limitations, it is easy to setup, need non-administrative access and work reliably.

[2] http://www.netscape.com

Benefits A major benefit with SSH is that it is possible to tunnel TCP based applications through SSH, e.g. email protocols, programming tools and even business applications like Oracle. To most users SSH appears to be terminal emulator similar to Telnet. The users do not see the encryption and therefore the security is transparent for the user. For system administrators SSH is a popular remote administration platform.

Limitations SSH is not designed to be incorporated into network gateways such as routers or firewalls as a complete VPN solution. It is possible to create a VPN by tunneling PPP through SSH, but it require much overhead and is not meant to handle a connection with a lot of bandwidth requirements like IPsec. TLS/SSL and IPsec is almost totally transparent to use, but SSH is not, to use SSH you have to be logged on to user account to utilize the transport layer security. SSH is used for scripting applications, whereas TLS/SSL and IPsec is incorporated into applications and the TCP/IP stack. UDP and ICMP is also a problem with SSH. It is not possible to tunnel UDP or ICMP traffic. These protocols can indeed be useful in some VPNs, e.g. securing audio streaming through a VPN. Another problem with SSH is that there are so many different implementations of the protocol that interoperability problems is starting to arise, e.g. different implementations of the server may crash the client and vice versa. This is happening despite that SSH is being standardized by IETF[4].

TLS/SSL

When a transport layer protocol like TLS/SSL is used, the application must have built in TLS/SSL support. Almost all new web browsers have TLS/SSL support[6]. Therefore a VPN solution with TLS/SSL is often used via web browser communication. It is possible to implement TLS/SSL support into other applications also, but TLS/SSL VPNs mainly use applications that run inside a web browser.

Benefits TLS/SSL is designed to be transparent to higher level protocols. A major benefit with TLS/SSL is its popularity in web and e-commerce. TLS/SSL provide a session oriented security. An TLS/SSL aware application opens a session and the server responds. When the application quits, the server quite the session. There is no permanent TLS/SSL connectively, as with IPsec between two hosts.

Another benefit with TLS/SSL is that certificates from root CA's is included in new web browsers, so that the user can verify a server certificate with a trusted CA. Like SSH it is possible to port forwarding to tunnel applications with TLS/SSL, e.g. with stunnel[3].

Limitations TLS/SSL lack support for UDP traffic, like SSH it require a stateful connection. There are also some limitations on the applications that support TLS/SSL, mainly web browsers and e-mail applications support TLS/SSL as standard. Another issue with TLS/SSL is that not all setups have implemented both server and client authentication. This is an aspect that should be taken into consideration if it is necessary to authenticate both the server and

[3] http://www.stunnel.org

the client in a connection. When using TLS/SSL in tunnel mode it can become expensive if the setup requires an external certification authority to sign many digital certificates.

IPsec

Full transparency is achieved when using IPsec. Every IP packet is secured regardless if it is a UDP, TCP or another type of packet. IPsec is probably the most efficient and secure VPN solution, but it has some limitations especially when it comes to implementation in different environments. Using NAT and IPsec could become an administration nightmare[6]. IPsec have four different "modes" that it can operate in AH (Authentication Header) transport mode, AH tunnel mode, ESP (Encapsulating Security Payoad) transport mode and ESP tunnel mode. These modes provide different packet security and all modes do not fit into every network solution.

Benefits IPsec provide security directly on the IP network layer and secure everything that is put on top of the IP network layer. The protocol has also been an Internet standard for quite some time and has been proven to be a secure and trusted method of securing data.

IPsec support the use of nested tunnels, that is if a user must pass through two or more secure gateways the tunnels can be double encrypted.

Limitations Although IPsec has more features than SSH and TLS/SSL it is often more difficult to implement and require special support in routers etc. Like SSH there are also some interoperability issues, different IPsec implementations do not always follow the standard and communicate problem-free between each other.

Further work

Further work that should be discussed is key management. How easy is it to adapt different key management solutions into VPNs for IPsec, TLS/SSL or SSH. Deciding how to manage keys is crucial to get a secure VPN-solution. This article has not covered this topic.

Also seeing how different firewall and IDS systems react to the different VPN-solutions are not covered in this article.

Conclusions

Selecting the right VPN-solution for different settings/environments can be difficult. Not all VPN-solutions fit into a specific network or host configuration. Knowing the limitations and differences is important.

References

[1] Becta ICT Advice. Virtual private networks. http://cnscenter.future.co.kr/resource/security/vpn/virtual_private_networks.pdf, June 2003.

[2] V. G. Cerf and E. Cain. The dod internet architecture model. *Computer Networks*, pages pages 307–318, 1983.

[3] IETF. Ip security protocol. http://www.ietf.org/html.charters/ipsec-charter.html, Visited 2004.

[4] IETF. Secure shell. http://www.ietf.org/html.charters/secsh-charter.html, Visited 2004.

[5] IETF. Transport layer security. http://www.ietf.org/html.charters/tls-charter.html, Visited 2004.

[6] Stephen Northcutt, Lenny Zeltser, Scott Winters, Karen Kent Frederick, and Ronald W. Ritchey. *Inside Network Perimeter Security.* New Riders, 2003.

[7] SSH. Ssh main homepage. http://www.ssh.com, Visited 2004.