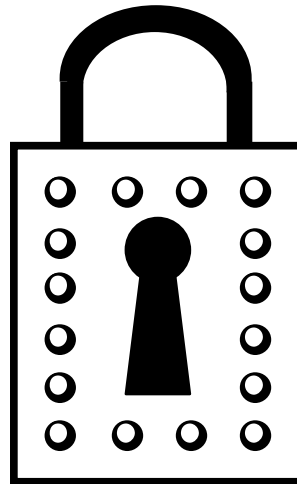


SSL and TLS



License

Copyright © 2008 Ciaran McHale.

Permission is hereby granted, free of charge, to any person obtaining a copy of this training course and associated documentation files (the "Training Course"), to deal in the Training Course without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Training Course, and to permit persons to whom the Training Course is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Training Course.

THE TRAINING COURSE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE TRAINING COURSE OR THE USE OR OTHER DEALINGS IN THE TRAINING COURSE.

What are SSL and TLS?

- In 1994, the world wide web was new and immature:
 - Web browsers and web servers sent only plaintext messages
 - Unsafe to use your credit-card to buy something from a website
- Netscape designed SSL to provide encrypted communication for the web
 - SSL stands for *secure sockets layer*
 - Netscape had a patent for SSL, but made SSL open
- SSL matured quickly with the help of the web community:
 - In 1995, SSL 3.0 was released
 - In 1996, Netscape handed over responsibility for SSL to the IETF
 - IETF = *Internet Engineering Task Force*
 - IEFT is an international standards organization
 - IETF renamed the next version of SSL to TLS 1.0
 - You can think of TLS 1.0 as being SSL 3.1

An extra layer in the protocol stack

- An application-level protocol normally talks directly to TCP/IP
- SSL was designed so:
 - It could be used with HTTP (an application-level protocol)
 - It could be used with other application-level protocols too
- For example:
 - CORBA is a remote procedure call (RPC) mechanism
 - The insecure CORBA protocol is called IIOP
 - The secure CORBA protocol is called IIOP/TLS

Simplified overview of SSL/TLS

■ Recall:

- Symmetric ciphers:
 - Are fast
 - But how do the two parties *securely* agree on a secret key?
- Asymmetric ciphers have the opposite properties:
 - Are 100–1000 times slower than symmetric ciphers
 - Can safely exchange public keys, even if other people overhear

■ Slightly simplified explanation of how SSL works:

- SSL uses both symmetric and asymmetric ciphers
- Uses an asymmetric cipher to securely communicate a private key for use with a symmetric cipher
- Then switches over to using the symmetric cipher with the agreed-upon private key

Simplified overview of SSL/TLS (cont')

- Actually, SSL uses six secret keys rather than just one:
 - Three are for client-generated messages
 - And three are for server-generated messages
 - The use of multiple keys makes life even harder for hackers
- Each group of three secret keys consists of:
 - A key used by the encryption cipher
 - A key used by the MAC cipher
 - A key used to initialize the encryption cipher

SSL/TLS supports many ciphers

- SSL/TLS uses one of each of the following:
 - A symmetric cipher
 - An asymmetric cipher
 - A MAC cipher
- But there are many competing ciphers in each category
 - Which one should be used?
- During the initial SSL/TLS handshaking:
 - Client sends a list of ciphers it understands to the server
 - The server picks one from each category and notifies client of its choice
- Benefits:
 - SSL/TLS can adapt whenever better ciphers are developed in the future
 - SSL/TLS can adapt to legal restrictions on ciphers in some countries

Summary

- TLS is the new name for SSL
 - TLS 1.0 = SSL 3.1
- SSL was first used to secure communication via HTTP
 - But can be used to secure other protocols
- SSL uses both symmetric and asymmetric ciphers:
 - Uses an asymmetric cipher to securely communicate a private key for use with a symmetric cipher
 - Then switches over to using the symmetric cipher with agreed-upon private keys
- SSL is *not* hardcoded to use a particular set of
 - Client and server negotiate on which set of ciphers to use
 - SSL can evolve to support newer, better ciphers when they are developed