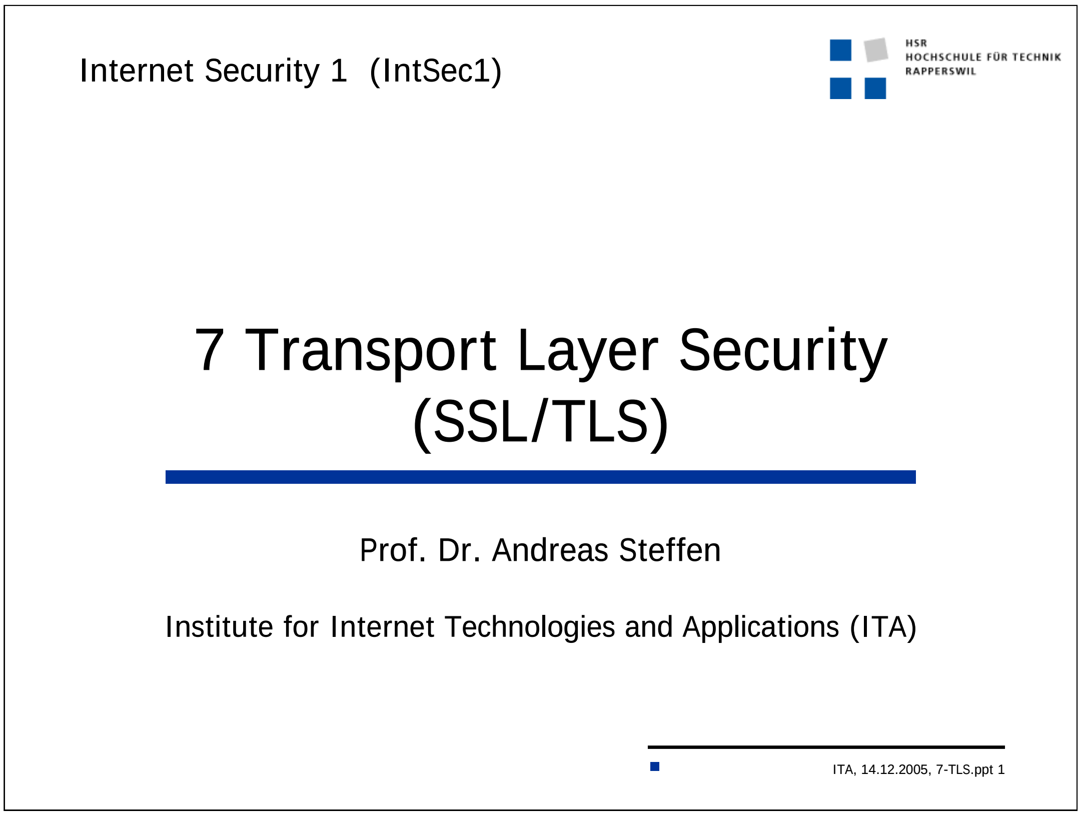

Internet Security 1 (IntSec1)

HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

7 Transport Layer Security (SSL/TLS)

Prof. Dr. Andreas Steffen

Institute for Internet Technologies and Applications (ITA)

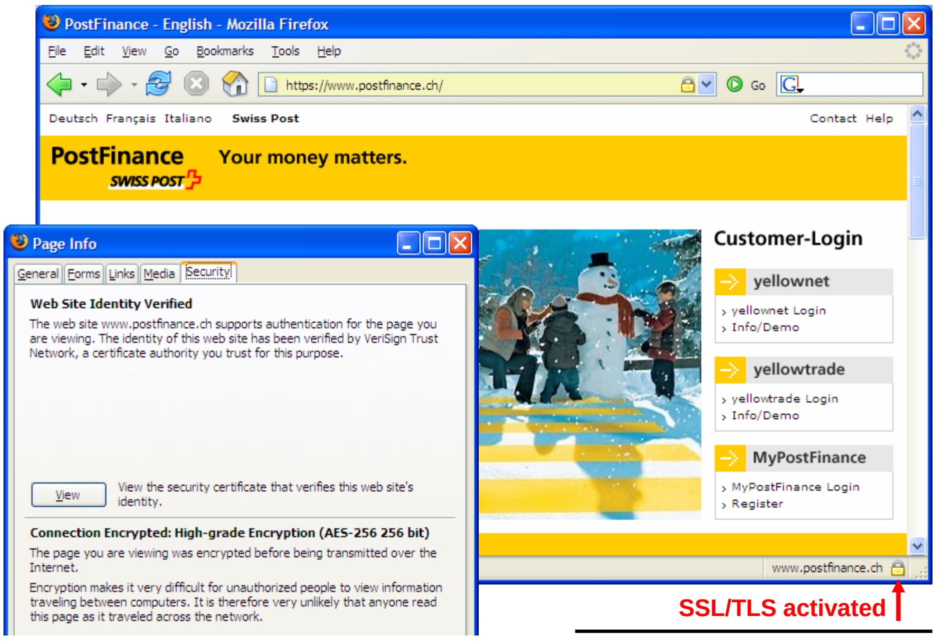
ITA, 14.12.2005, 7-TLS.ppt 1

7 Transport Layer Security

- TLS example
- Layer 3 versus layer 4 security
- SSL/TLS protocol layers
- SSL/TLS handshake protocol
- Resuming a SSL/TLS session
- Implemented SSL/TLS protocol versions
- SSL/TLS configuration options (Mozilla Firefox and Internet Explorer)
- SSL/TLS enhanced TCP-based application protocols

TLS Session Example

HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL



Web Site Identity Verified

The web site www.postfinance.ch supports authentication for the page you are viewing. The identity of this web site has been verified by VeriSign Trust Network, a certificate authority you trust for this purpose.

View the security certificate that verifies this web site's identity.

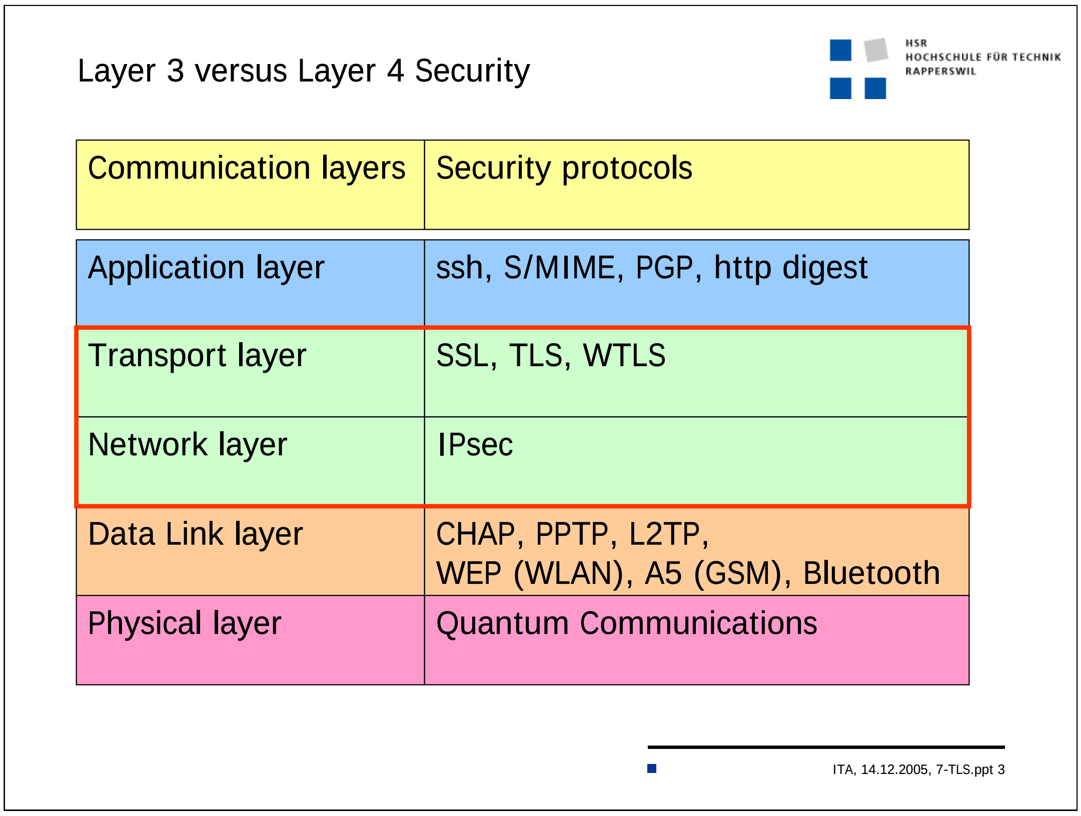
Connection Encrypted: High-grade Encryption (AES-256 256 bit)

The page you are viewing was encrypted before being transmitted over the Internet.

Encryption makes it very difficult for unauthorized people to view information traveling between computers. It is therefore very unlikely that anyone read this page as it traveled across the network.

SSL/TLS activated

ITA, 14.12.2005, 7-TLS.ppt 2

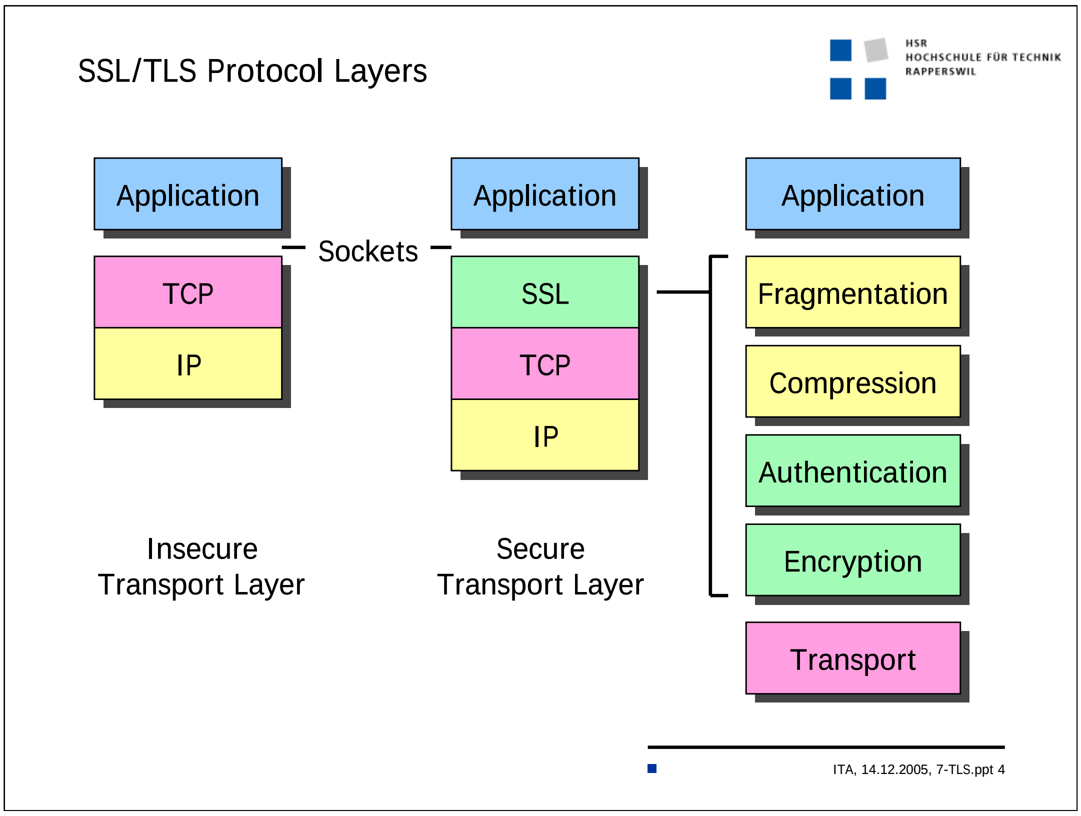


Layer 3 versus Layer 4 Security

HSR HOCHSCHULE FÜR TECHNIK RAPPERSWIL

Communication layers	Security protocols
Application layer	ssh, S/MIME, PGP, http digest
Transport layer	SSL, TLS, WTLS
Network layer	IPsec
Data Link layer	CHAP, PPTP, L2TP, WEP (WLAN), A5 (GSM), Bluetooth
Physical layer	Quantum Communications

ITA, 14.12.2005, 7-TLS.ppt 3



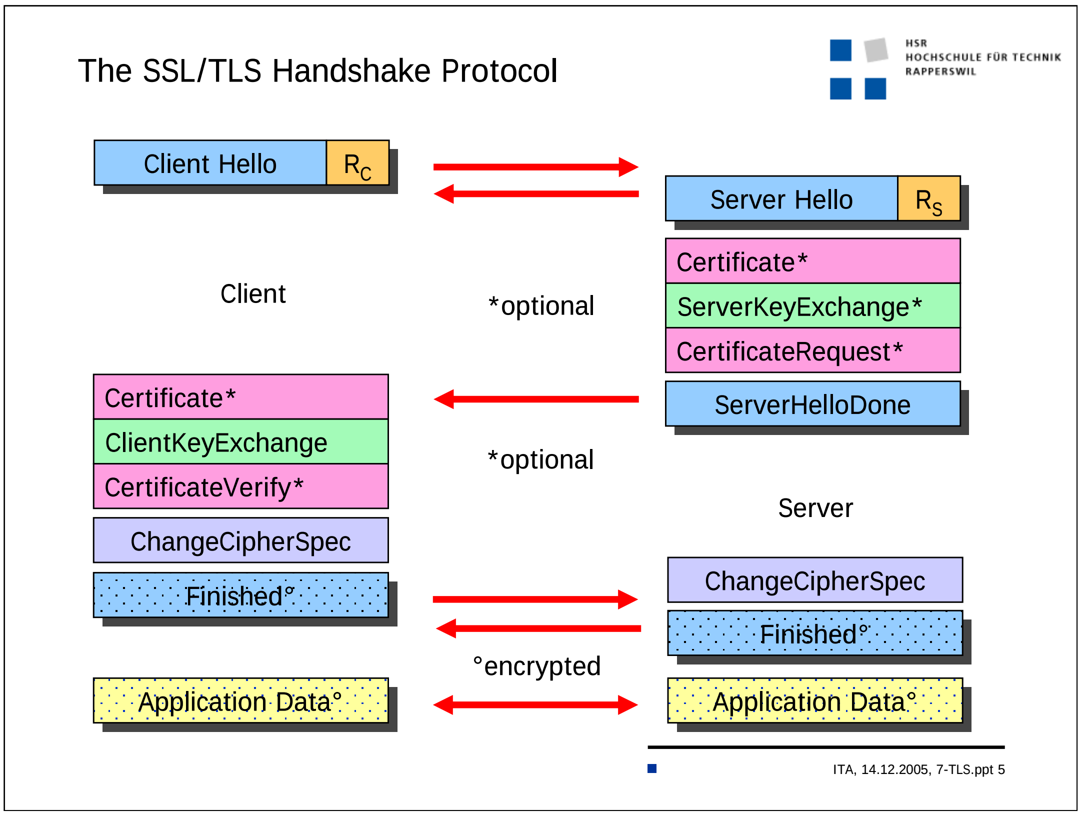
SSL/TLS Protocol Layers

HSR HOCHSCHULE FÜR TECHNIK RAPPERSWIL

ITA, 14.12.2005, 7-TLS.ppt 4

SSL/TLS Protocol Layer

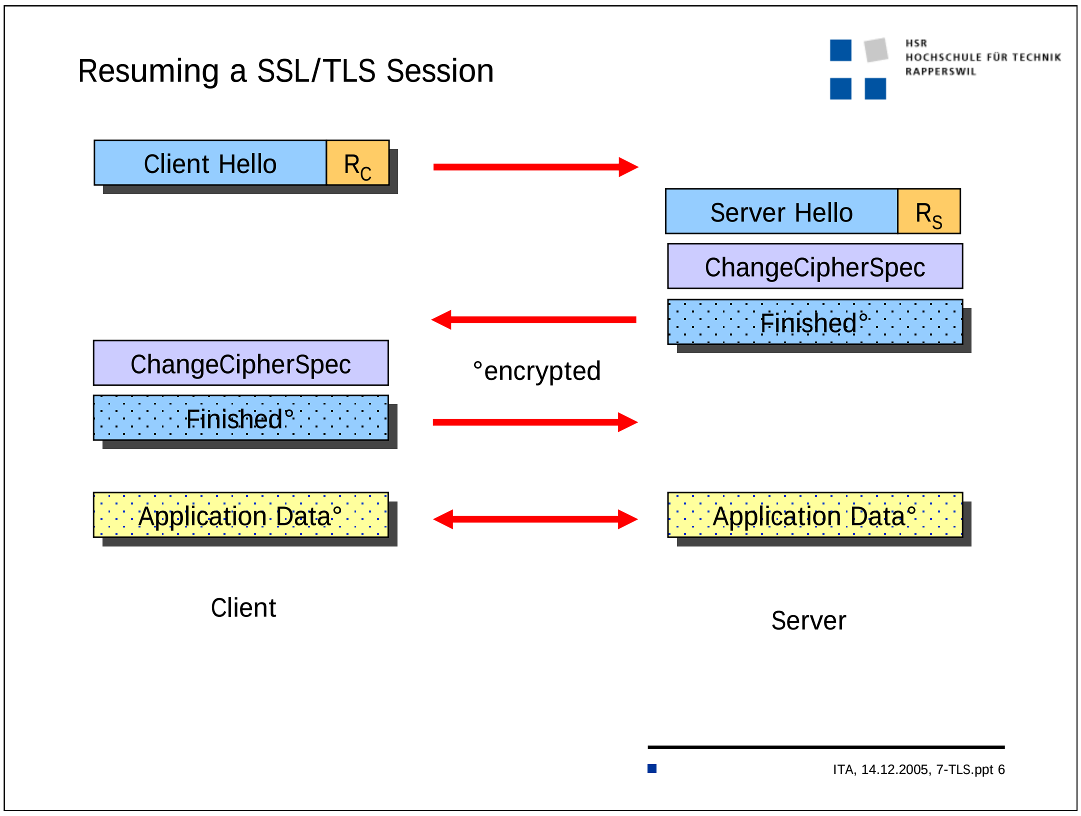
- The **Secure Sockets Layer** (SSL) is inserted between the **Transport layer** and the **Application Layer** (with communication layers defined according to Tanenbaum !). In contrast to **IPSec** which is a **Layer 3+** protocol based directly on IPv4 or IPv6, **SSL** is a **Layer 4+** protocol based directly on a TCP transport mechanism.
- The **SSL protocol** offers secure sockets to **SSL-aware** applications. The TCP/IP stack of the SSL client and server platforms do not have to be modified!
- The **IPSec protocol** offers secure communication to any existing IP based service or application. It is the IP stacks of the IPsec clients or IPsec security gateways that must be modified in order to support IPsec transport mode or IPSec tunnel mode, respectively.
- The SSL protocol is responsible for the following tasks:
  - Fragmentation of application data streams into SSL PDUs
  - Compression of PDUs before encryption
  - Authentication of PDUs
  - Encryption of PDUs



SSL/TLS Handshake Protocol

- The SSL session state is controlled by the SSL handshake protocol that runs on top of the SSL record layer. When a SSL client and a SSL server first start communicating, they agree on a protocol version, select cryptographic algorithms, optionally authenticate each other, and use public-key encryption techniques to generate shared secrets.
- The client starts with a **ClientHello** message to which the server must respond with a **ServerHello** message – otherwise a fatal error occurs and the connection fails. The following attributes are established: Protocol Version, Session ID, Cipher Suite, and Compression Method. Additionally, two random values are generated and exchanged ClientHello-Random R_C and ServerHello-Random R_S.
- Next the server usually sends its X.509 server certificate in an optional **Certificate** message. If no certificate is sent, then an optional **ServerKeyExchange** message may be sent instead, containing the server part of a Diffie-Hellman (DH) secret. If the server insists on a **client side authentication** an optional **CertificateRequest** message is appended. The server indicates the end of the server hello phase by sending a **ServerHelloDone** message.
- If the server has sent a CertificateRequest message, the client must send either its X.509 client certificate in a **Certificate** message or a ‘no certificate‘ alert. If the client has received a server certificate containing the server‘s public RSA key, the client encrypts a randomly chosen premaster secret with it and sends it to the server in a **ClientKeyExchange** message. Alternatively the clients can send its part of a DH key exchange. Each side can now form a shared master secret.
- The client then emits a **ChangeCipherSpec** message announcing that the new parameters have been loaded, followed by a **Finished** message already encrypted with the new settings. The server does the same on its side.
- The encrypted exchange of application data can now be started.

Source: Stephen Thomas, SSL and TLS Essentials, Wiley Computer Publishing

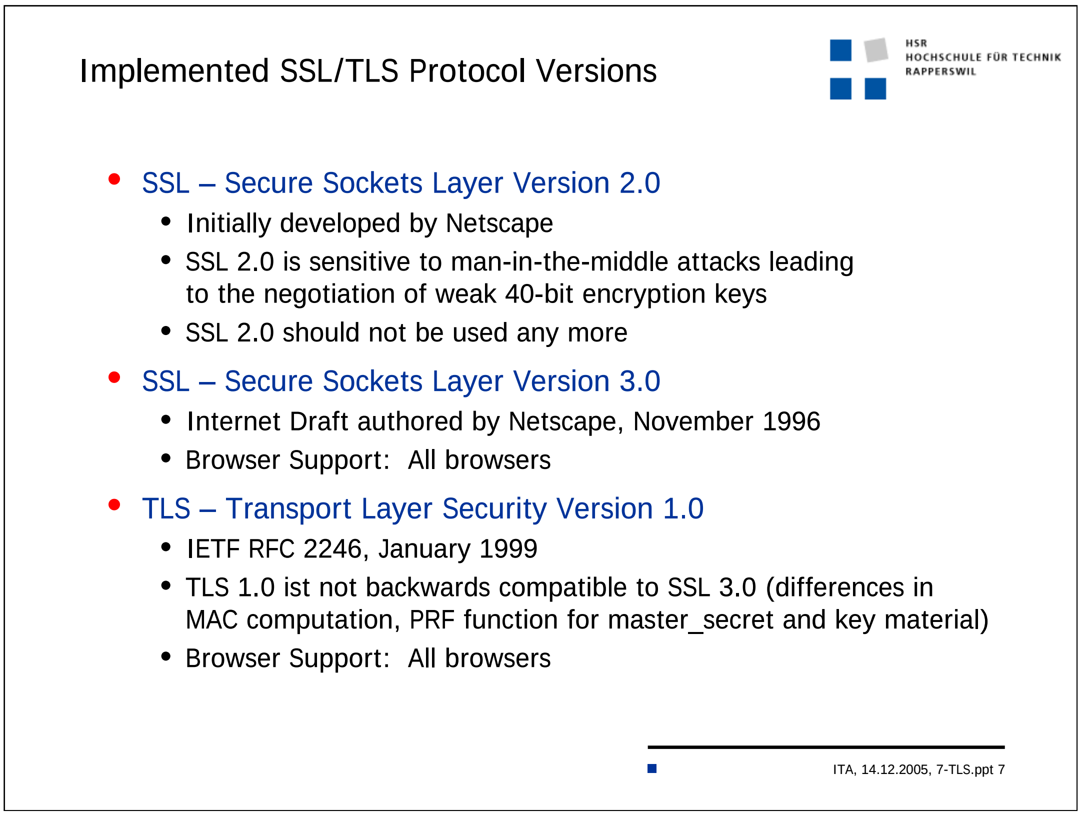


Resuming a SSL/TLS Session

Resuming a SSL/TLS Session

- When the client and server decide to resume a previous session or duplicate an existing session (instead of negotiating new security parameters) the message flow is as follows:
- The client sends a ClientHello using the Session ID of the session to be resumed. The server then checks its session cache for a match. If a match is found, and the server is willing to re-establish the connection under the specified session state, it will send a ServerHello with the same Session ID value. Using the cached master secret and the fresh client hello and server hello nonces, new session key material is generated. At this point, both client and server must send change cipher spec messages and proceed directly to the finished messages. Once the re-establishment is complete, the client and server may begin to exchange application layer data.
  If a Session ID match is not found, the server generates a new session ID and the TLS client and server perform a full handshake.

Source: RFC 2246 – TLS Protocol Version 1.0



Implemented SSL/TLS Protocol Versions

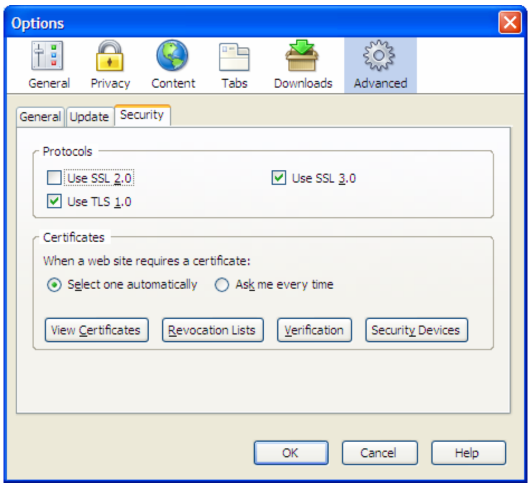
HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

- SSL – Secure Sockets Layer Version 2.0
  - Initially developed by Netscape
  - SSL 2.0 is sensitive to man-in-the-middle attacks leading to the negotiation of weak 40-bit encryption keys
  - SSL 2.0 should not be used any more
- SSL – Secure Sockets Layer Version 3.0
  - Internet Draft authored by Netscape, November 1996
  - Browser Support: All browsers
- TLS – Transport Layer Security Version 1.0
  - IETF RFC 2246, January 1999
  - TLS 1.0 ist not backwards compatible to SSL 3.0 (differences in MAC computation, PRF function for master_secret and key material)
  - Browser Support: All browsers

ITA, 14.12.2005, 7-TLS.ppt 7

SSL/TLS Configuration Options
Firefox 1.5

HSR HOCHSCHULE FÜR TECHNIK RAPPERSWIL


Options

Protocols

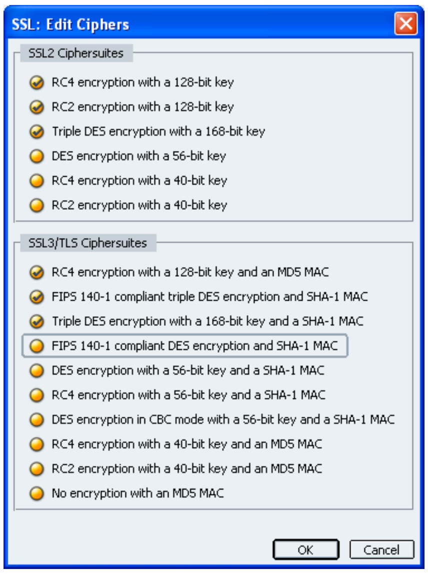
[ ] Use SSL 2.0 [✓ Use SSL 3.0]
[✓ Use TLS 1.0]

Certificates

When a web site requires a certificate:

[✓ Select one automatically] Ask me every time

SSL/TLS Configuration Options
Mozilla 1.x



SSL2 Ciphersuites

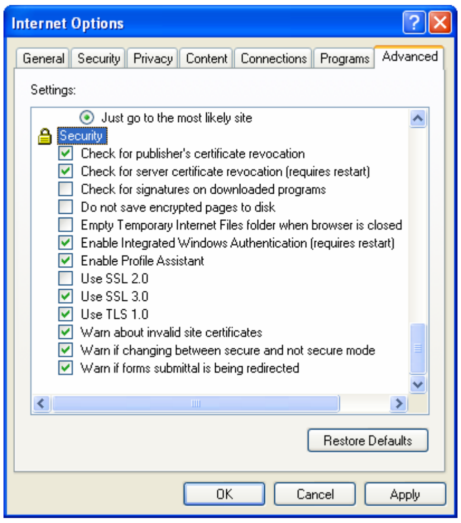
- [✓] RC4 encryption with a 128-bit key
- [✓] RC2 encryption with a 128-bit key
- [✓] Triple DES encryption with a 168-bit key
- DES encryption with a 56-bit key
- RC4 encryption with a 40-bit key
- RC2 encryption with a 40-bit key

SSL3/TLS Ciphersuites

- [✓] RC4 encryption with a 128-bit key and an MD5 MAC
- [✓] FIPS 140-1 compliant triple DES encryption and SHA-1 MAC
- [✓] Triple DES encryption with a 168-bit key and a SHA-1 MAC
- FIPS 140-1 compliant DES encryption and SHA-1 MAC
- DES encryption with a 56-bit key and a SHA-1 MAC
- RC4 encryption with a 56-bit key and a SHA-1 MAC
- DES encryption in CBC mode with a 56-bit key and a SHA-1 MAC
- RC4 encryption with a 40-bit key and an MD5 MAC
- RC2 encryption with a 40-bit key and an MD5 MAC
- No encryption with an MD5 MAC

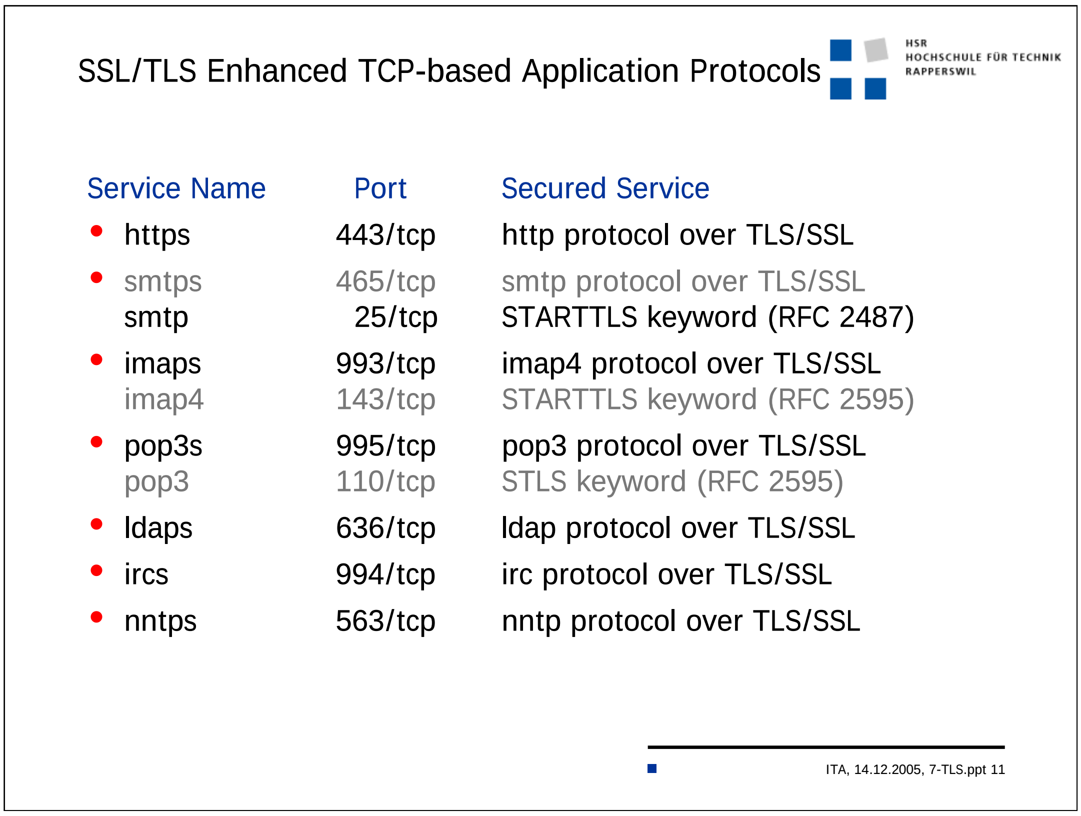
SSL/TLS Configuration Options
Internet Explorer 6.0

HSR HOCHSCHULE FÜR TECHNIK RAPPERSWIL



Settings:

- [✓] Just go to the most likely site
- Security
  - [✓] Check for publisher's certificate revocation
  - [✓] Check for server certificate revocation (requires restart)
  - [ ] Check for signatures on downloaded programs
  - [ ] Do not save encrypted pages to disk
  - [ ] Empty Temporary Internet Files folder when browser is closed
  - [✓] Enable Integrated Windows Authentication (requires restart)
  - [✓] Enable Profile Assistant
  - [ ] Use SSL 2.0
  - [✓] Use SSL 3.0
  - [✓] Use TLS 1.0
  - [✓] Warn about invalid site certificates
  - [✓] Warn if changing between secure and not secure mode
  - [✓] Warn if forms submittal is being redirected



SSL/TLS Enhanced TCP-based Application Protocols

HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

Service Name	Port	Secured Service
https	443/tcp	http protocol over TLS/SSL
smtps	465/tcp	smtp protocol over TLS/SSL
smtp	25/tcp	STARTTLS keyword (RFC 2487)
imaps	993/tcp	imap4 protocol over TLS/SSL
imap4	143/tcp	STARTTLS keyword (RFC 2595)
pop3s	995/tcp	pop3 protocol over TLS/SSL
pop3	110/tcp	STLS keyword (RFC 2595)
ldaps	636/tcp	ldap protocol over TLS/SSL
ircs	994/tcp	irc protocol over TLS/SSL
nntps	563/tcp	nntp protocol over TLS/SSL

ITA, 14.12.2005, 7-TLS.ppt 11