Ing. Ondřej Ševeček | PM Windows Server | GOPAS a.s. |
MCM: Directory Services | MVP: Enterprise Security | CEH |
ondrej@sevecek.com | www.sevecek.com |

TLS

1

Agenda

- What is TLS
- Algorithms and certificates
- Operating system support
- Attacks and patches
- Client certificate authentication

2

TLS

PROTOCOL BASICS

Transport Layer Security

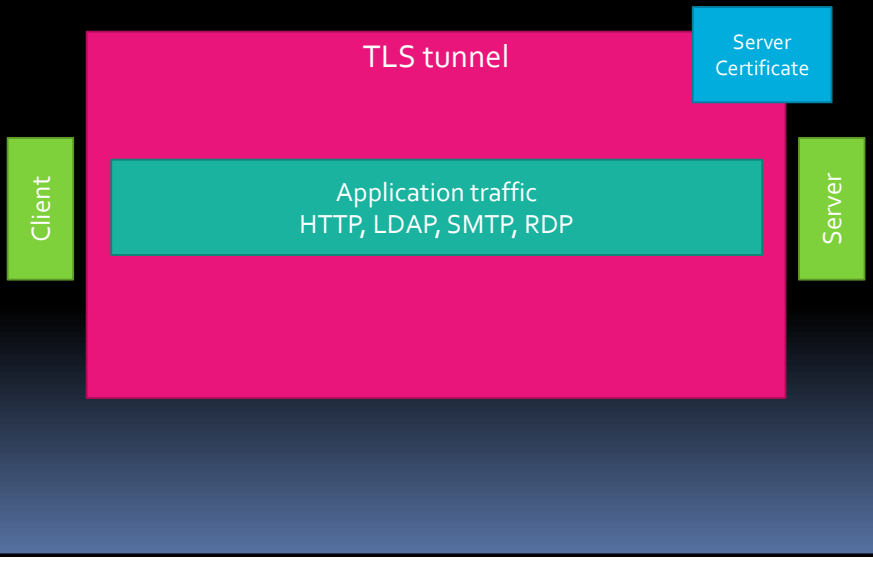
- Standard cryptographic protocol for secure transmissions
  - RSA/DSA/EC, RC4, DES, AES, MD5, SHA1, …
- Encryption and server identity authentication
  - HTTPS, SSTP, IPHTTPS, LDAPS, SQL, RDPS, SMTPS, Hyper-V replication, 802.1x EAP, IPSec IKEExt
- Client certificate authentication
- Requires public key certificate on the server

SSL vs. TLS vs. DTLS

- SSL 2.0 (1995) - Windows 2000+
  - MITM can downgrade cipher suite to 40-bit
  - MAC hashes can be downgraded to 40-bit
- SSL 3.0 (1996) - Windows 2000+
  - Support for DH, Fortezza key exchanges
  - Support for non RSA certificates
- TLS 1.0 (1999) - Windows 2000+
  - Security same as SSL 3.0
  - Protocol not compatible with SSL 3.0
  - IETF and US FIPS standard
- TLS 1.1 and 1.2 (2006, 2008) - Windows 7/2008 R2
  - More recent standards offering SHA2 and ECDH suites
  - Can fallback to TLS 1.0 without TCP RST
- TLS 1.3 (2019)
- DTLS 1.0 (based on TLS 1.0) and 1.2 (based on TLS 1.2) - Windows 8/2012
  - Update available for Windows 7/2008 R2 (KB2574819)
  - UDP datagram based communications such as RDP-UDP

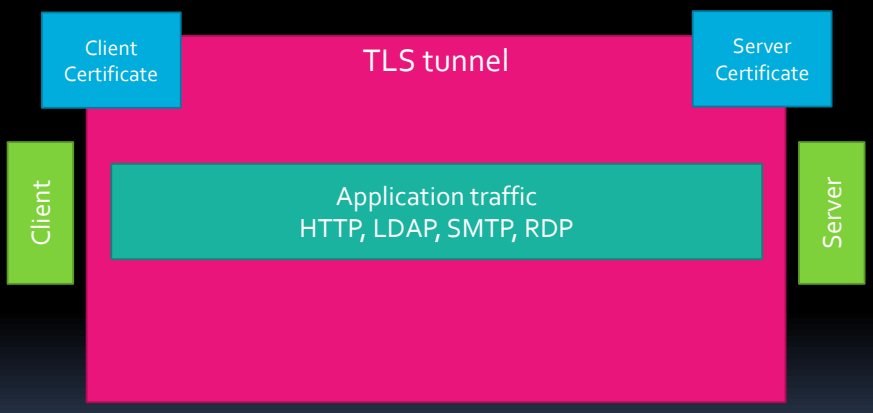
5

TLS with server certificate only



6

TLS with client certificate



Server certificate

- Encryption key "transport"
  - RSA key exchange
  - DSA/DH key agreement
  - ECDSA/ECDH key agreement
- Server identity authentication
  - Subject and SAN names
  - time validity
  - trusted issuer chain
  - revocation checking with CRL/OCSP

SChannel

- COM library for establishing TLS communications
- SCHANNEL Security Provider
  - HKLM\System\CCS\Control\SecurityProviders\SCHANNEL
- Group Policy
  - Policies / Administrative Templates / Network / SSL

SSL 2.0 cipher suites

- SSL_RC4_128_WITH_MD5
- SSL_DES_192_EDE3_CBC_WITH_MD5
- SSL_RC2_CBC_128_CBC_WITH_MD5
- SSL_DES_64_CBC_WITH_MD5
- SSL_RC4_128_EXPORT40_WITH_MD5

Disable SSL 2.0 and PCT
disable SSL 3.0 recommended

- HKLM\System\CurrentControlSet\Control\SecurityProviders \SCHANNEL\Protocols
  - \PCT 1.0
    - …
  - \SSL 2.0
    - \Client
      - Enabled = DWORD = 0
    - \Server
      - Enabled = DWORD = 0
  - \SSL 3.0
    - …

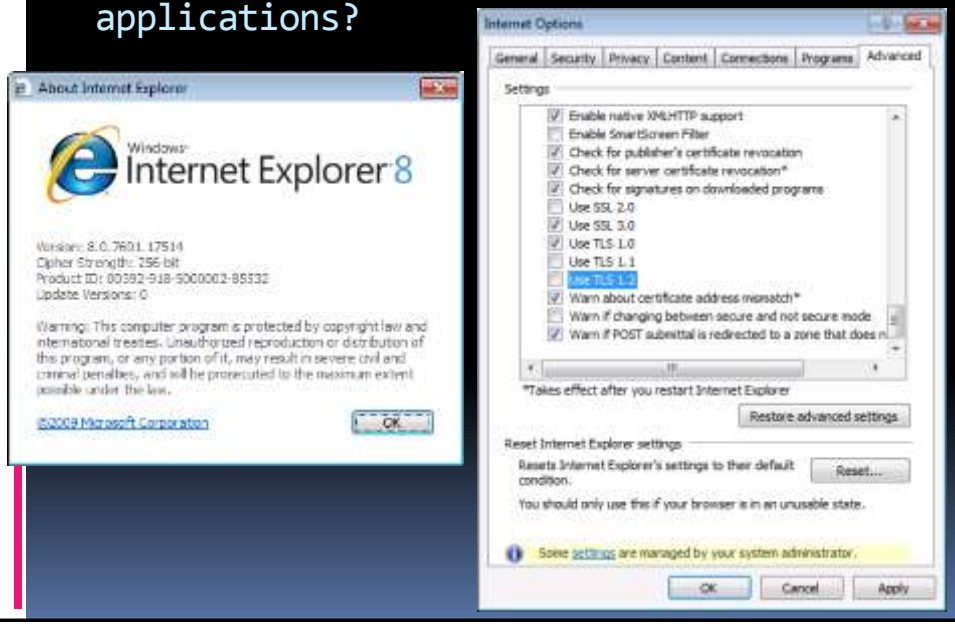
11

Enable TLS 1.1 and 1.2

- HKLM\System\CurrentControlSet\Control\SecurityProviders \SCHANNEL\Protocols
  - \TLS 1.1
  - \TLS 1.2
    - \Client
      - Enabled = DWORD = 1
      - DisabledByDefault = DWORD = 0
    - \Server
      - Enabled = DWORD = 1
      - DisabledByDefault = DWORD = 0

12

Enable TLS 1.1 and newer in applications?



13

Application support for TLS 1.1 and newer

- Windows XP/2003 only TLS 1.0
- IE 9+ by default
- RDP client and server since Windows 8/2012

- NetFx 2.0/3.x TLS 1.0 only
- NetFx 2.0/3.x SHA1 only

14

Windows XP/2003- TLS 1.0/SSL cipher suites (no AES)

TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
TLS_RSA_WITH_DES_CBC_SHA
TLS_DHE_DSS_WITH_DES_CBC_SHA
TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA
TLS_DHE_DSS_EXPORT1024_WITH_DES_CBC_SHA
TLS_RSA_EXPORT_WITH_RC4_40_MD5
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5
TLS_RSA_WITH_NULL_MD5
TLS_RSA_WITH_NULL_SHA

SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_RC4_128_MD5

15

Server certificate: RSA + RSA key exchange + Key encipherment

TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
TLS_RSA_WITH_DES_CBC_SHA
TLS_DHE_DSS_WITH_DES_CBC_SHA
TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA
TLS_DHE_DSS_EXPORT1024_WITH_DES_CBC_SHA
TLS_RSA_EXPORT_WITH_RC4_40_MD5
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5
TLS_RSA_WITH_NULL_MD5
TLS_RSA_WITH_NULL_SHA

SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_RC4_128_MD5

16

Server certificate:
DSA + DH key agreement + Digital signature

TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
TLS_RSA_WITH_DES_CBC_SHA
TLS_DHE_DSS_WITH_DES_CBC_SHA
TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA
TLS_DHE_DSS_EXPORT1024_WITH_DES_CBC_SHA
TLS_RSA_EXPORT_WITH_RC4_40_MD5
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5
TLS_RSA_WITH_NULL_MD5
TLS_RSA_WITH_NULL_SHA

SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_RC4_128_MD5

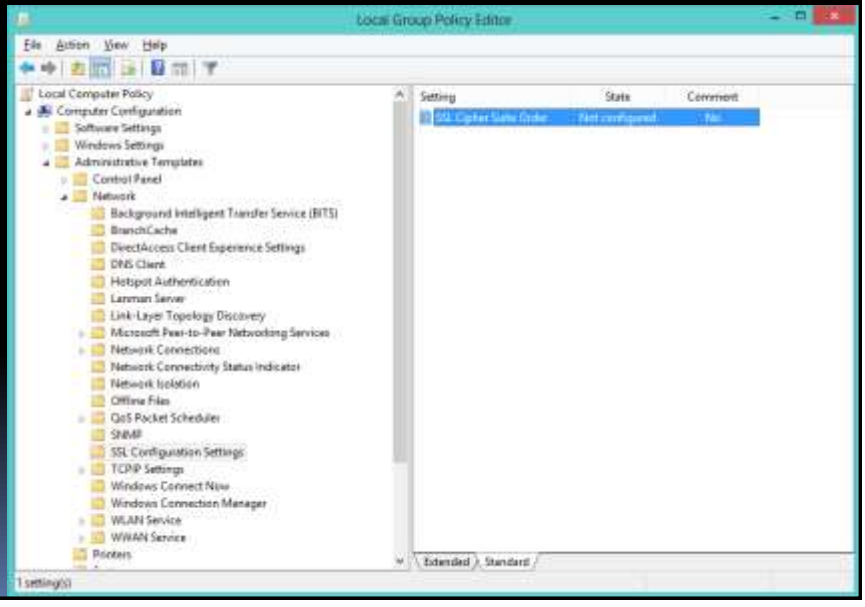
AES support on Windows 2003

- KB948963
- TLS_RSA_WITH_AES_128_CBC_SHA
  AES128-SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
  AES256-SHA

Disable/Enable Suites

- KB245030
- HKLM\SYSTEM\CCS\Control\SecurityProviders\SCHANNEL\Ciphers\NULL
  - Enabled = DWORD = 0
- RC4 40/128, RC2 56/56, RC2 56/128, RC4 56/128, RC4 64/128, RC2 128/128, Triple DES 168/168, RC4 128/128, …

TLS cipher suite order (Vista+)



Windows Vista/2008+ TLS v1.0 cipher suites (AES/EC/SHA1)

TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
TLS_DHE_DSS_WITH_AES_128_CBC_SHA
TLS_DHE_DSS_WITH_AES_256_CBC_SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
TLS_RSA_WITH_RC4_128_MD5

21

Windows Vista/2008+ TLS v1.0 cipher suites (AES/EC/SHA1)

TLS_RSA_WITH_AES_128_CBC_SHA	RSA + RSA-KE + Key Encipherment
TLS_RSA_WITH_AES_256_CBC_SHA	RSA + RSA-KE + Key Encipherment
TLS_RSA_WITH_RC4_128_SHA	RSA + RSA-KE + Key Encipherment
TLS_RSA_WITH_3DES_EDE_CBC_SHA	RSA + RSA-KE + Key Encipherment
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	RSA + ECDH + Digital Signature
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	RSA + ECDH + Digital Signature
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	ECDSA + ECDH + Digital Signature
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	ECDSA + ECDH + Digital Signature
TLS_DHE_DSS_WITH_AES_128_CBC_SHA	DSA + DH-KA + Digital Signature
TLS_DHE_DSS_WITH_AES_256_CBC_SHA	DSA + DH-KA + Digital Signature
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	DSA + DH-KA + Digital Signature
TLS_RSA_WITH_RC4_128_MD5	RSA + RSA-KE + Key Encipherment

22

Windows 7/2008 R2 TLS v1.1 cipher suites (AES/EC/SHA2)

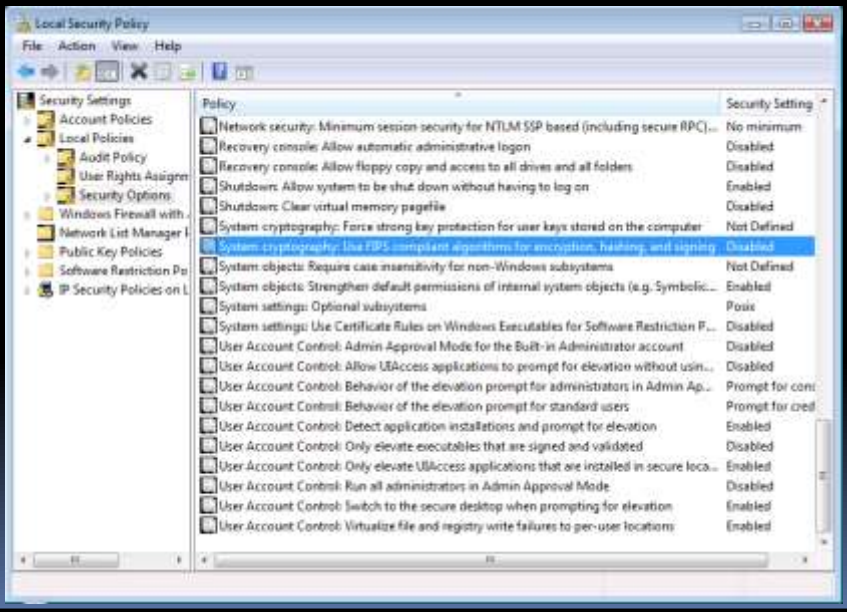
TLS_RSA_WITH_AES_128_CBC_SHA256
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256_P256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256_P384
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA_P256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA_P384
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA_P256
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA_P384
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256_P256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256_P256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA384_P384
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA384_P384
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA_P256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA_P384
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA_P256
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA_P384
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256
TLS_DHE_DSS_WITH_AES_128_CBC_SHA
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256
TLS_DHE_DSS_WITH_AES_256_CBC_SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
TLS_RSA_WITH_RC4_128_MD5
SSL_CK_RC4_128_WITH_MD5
SSL_CK_DES_192_EDE3_CBC_WITH_MD5
TLS_RSA_WITH_NULL_SHA256
TLS_RSA_WITH_NULL_SHA

SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_RC4_128_MD5

Third-party support

- Java SE 7
  - TLS 1.1, TLS 1.2
- Java 1.4.2
  - SHA-256 in crypto provider

FIPS compatibility



FIPS compatibility

- Severe compatibility impact
  - KB811833
- Disables SSL 2.0 and SSL 3.0
- Allows only TLS 1.0 and newer
  - RDP support since Windows 2003 SP1
  - RDP client 5.2+
- Cannot use RC4
- Cannot use MD5

TLS

PROTOCOL CONFIGURATION AND OPERATION

Server certificate

- RSA encryption + Key encipherment
  - RSA key exchange
  - Exchange requires signature as well
- DSA/ECDSA signature + Digital signature
  - DH key agreement

Comparable Algorithm Strengths (SP800-57)

Strength	Symetric	RSA	ECDSA	SHA
80 bit	2TDEA	RSA 1024	ECDSA 160	SHA-1
112 bit	3TDEA	RSA 2048	ECDSA 224	SHA-224
128 bit	AES-128	RSA 3072	ECDSA 256	SHA-256
192 bit	AES-192	RSA 7680	ECDSA 384	SHA-384
256 bit	AES-256	RSA 15360	ECDSA 512	SHA-512

Server certificate Subject

- Single name
- Wildcard name
- EV company identification

Server certificate SAN

- If SAN present, Subject is ignored
- Always repeat the Subject value in SAN

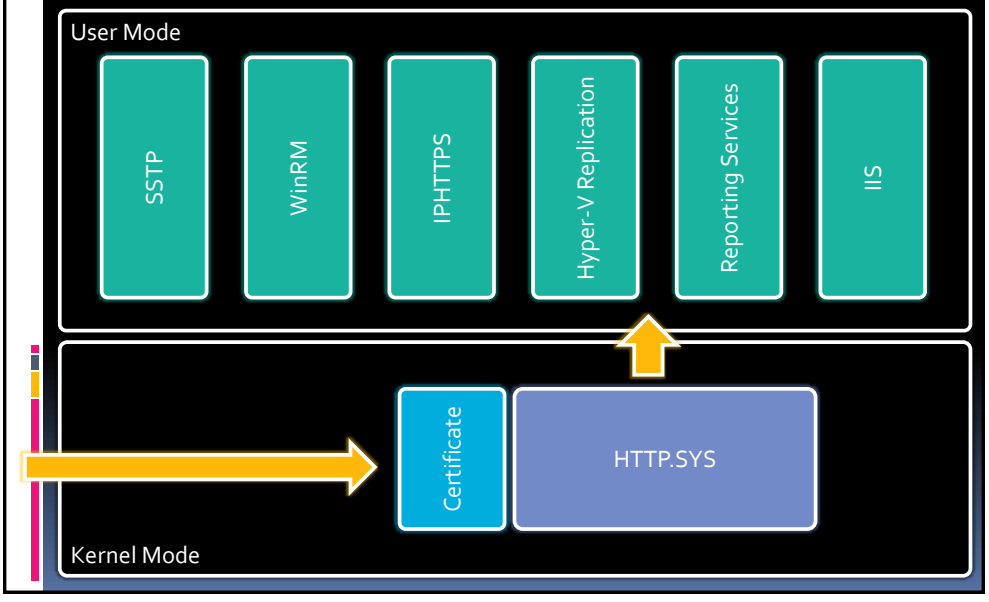
31

CSP vs. CNG

- Cryptographic Service Provider (CSP)
  - Windows 2003 require RSA SChannel Cryptographic Service Provider or DH SChannel Cryptographic Service Provider
  - System Center clients require CSP
  - SQL Server 2012 and older require CSP
- Cryptography Next Generation (CNG)
  - Windows Vista and newer
  - HTTPS.SYS, LDAPS, RDPS

32

IIS and HTTP.SYS



33

NETSH HTTP

- netsh http show sslcert
- netsh http add sslcert <thumbprint> <appid>

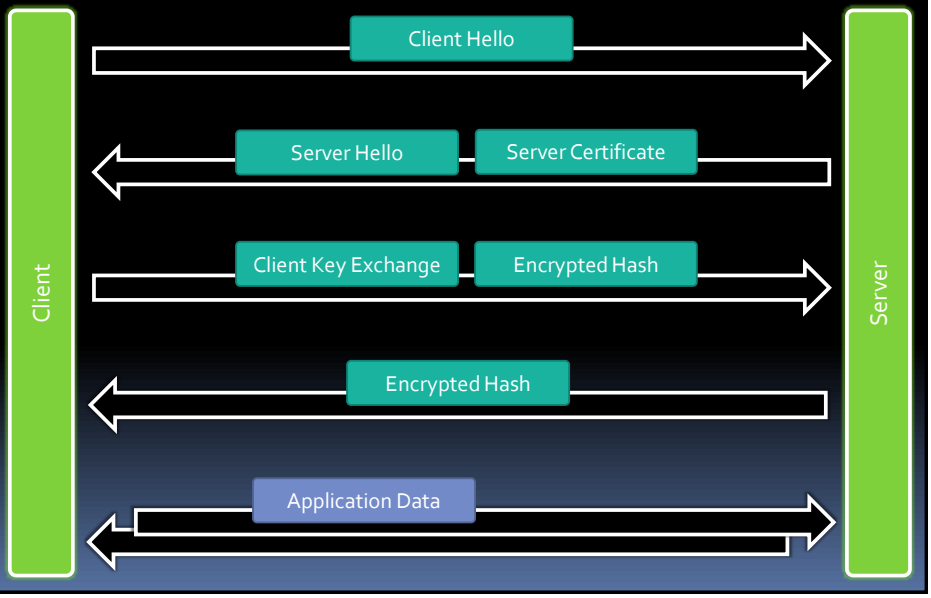
34

HTTP.SYS AppId

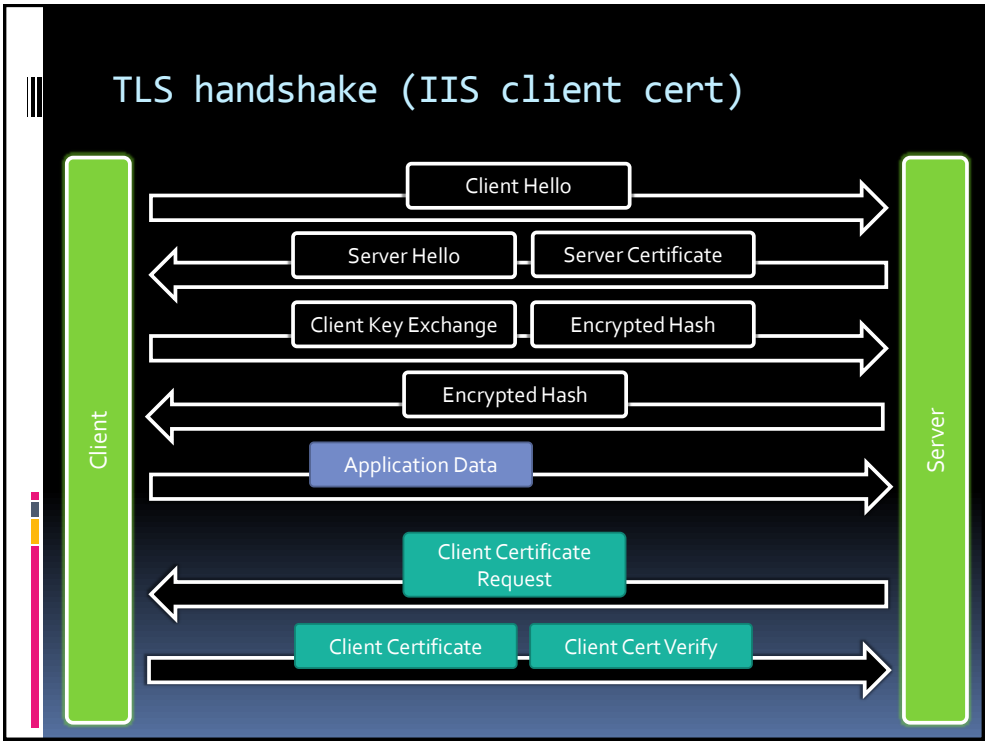
- http://www.sevecek.com/Lists/Posts/Post.aspx?ID=9
- IIS
  - {4dc3e181-e14b-4a21-b022-59fc669b0914}
- SSTP
  - {ba195980-cd49-458b-9e23-c84ee0abcd75}
- SQL RS
  - {1d40ebc7-1983-4ac5-82aa-1e17a7ae9a0e}
- WinRM
  - {afebb9ad-9b97-4a91-9ab5-daf4d59122f6}
- Hyper-V
  - {fed10a98-8cb9-41e2-8608-264b923c2623}

35

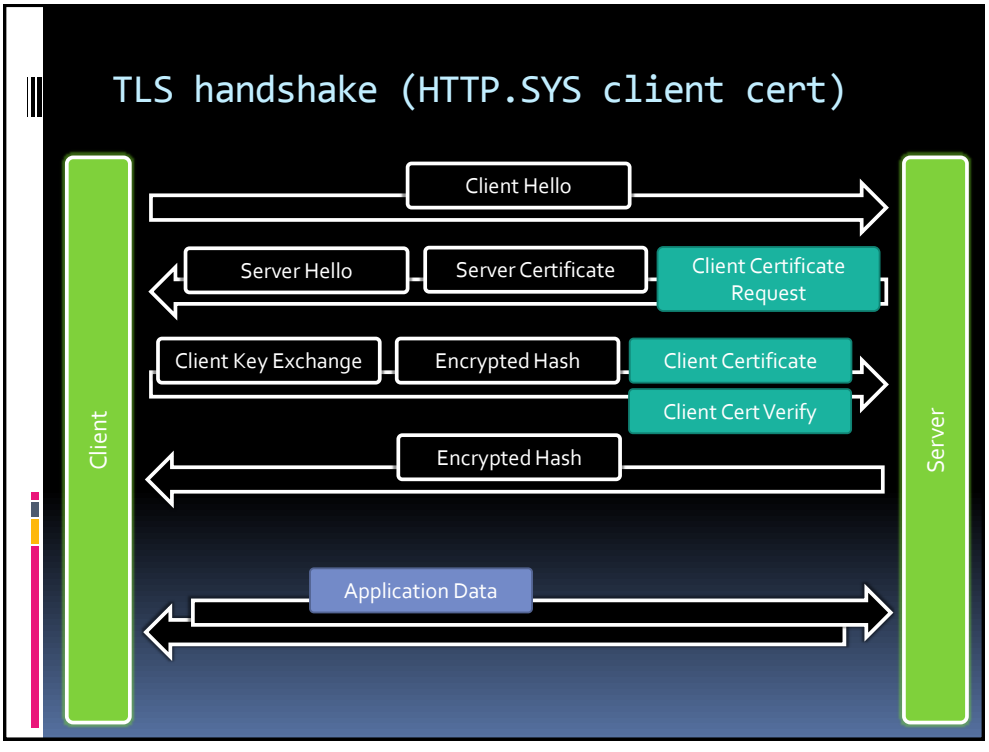
TLS handshake (no client cert)



36


TLS handshake (IIS client cert)
Client
Server
Client Hello
Server Hello
Server Certificate
Client Key Exchange
Encrypted Hash
Encrypted Hash
Application Data
Client Certificate Request
Client Certificate
Client Cert Verify

37


TLS handshake (HTTP.SYS client cert)
Client
Server
Client Hello
Server Hello
Server Certificate
Client Certificate Request
Client Key Exchange
Encrypted Hash
Client Certificate
Client Cert Verify
Encrypted Hash
Application Data

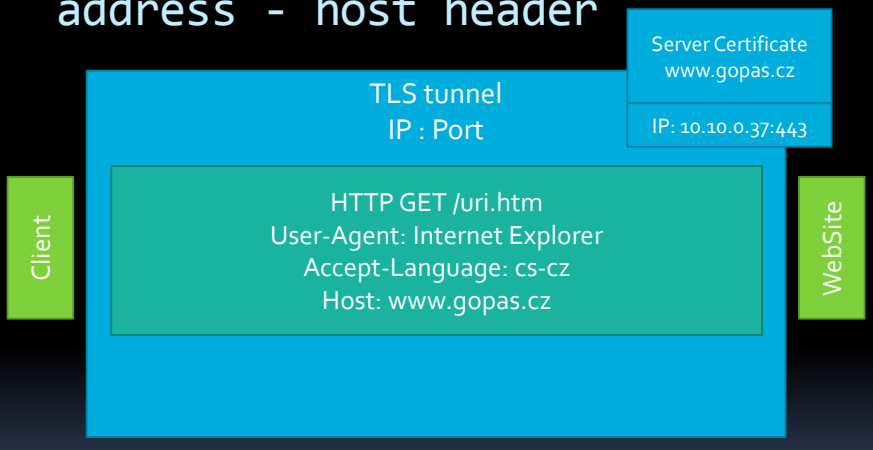
38

TLS

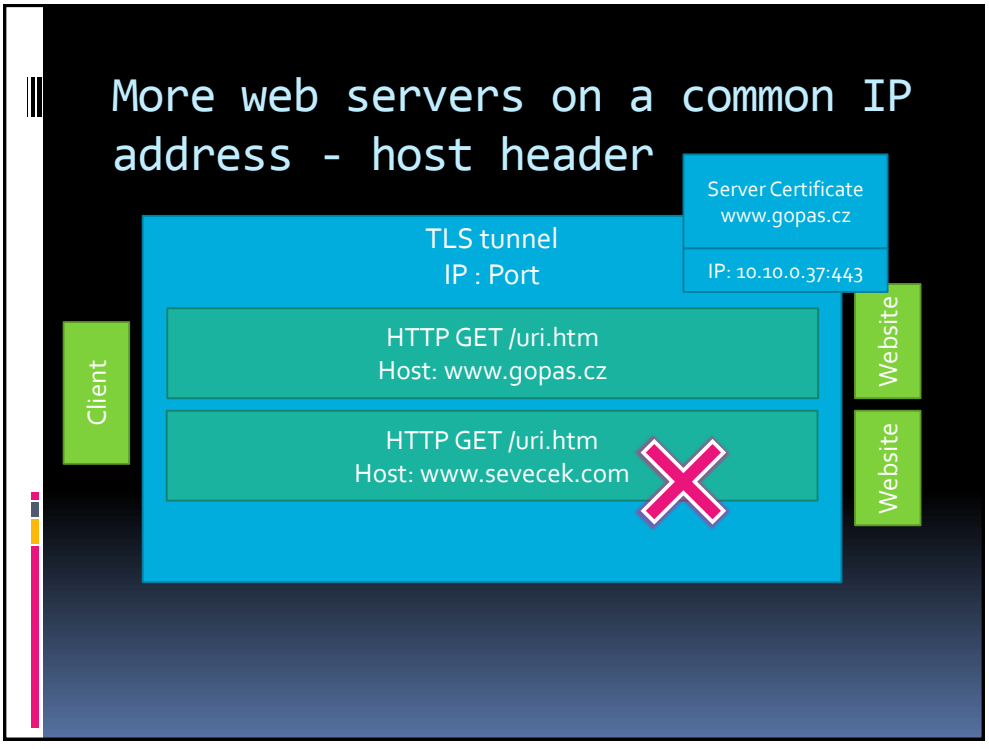
SERVER NAME INDICATION

39

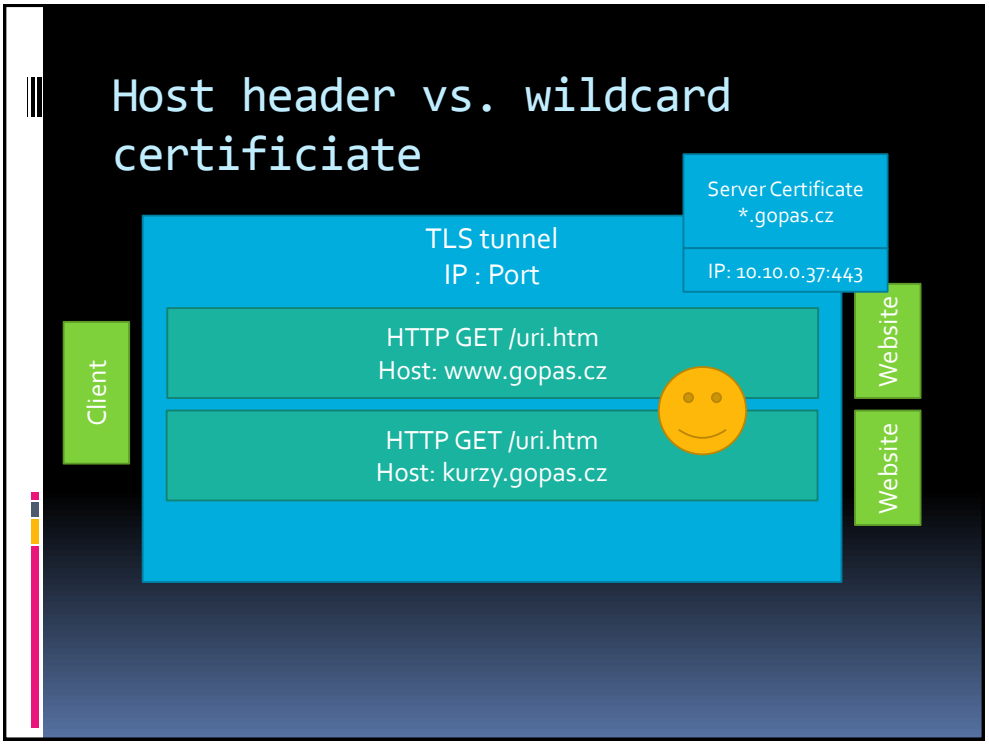
More web servers on a common IP address - host header



40

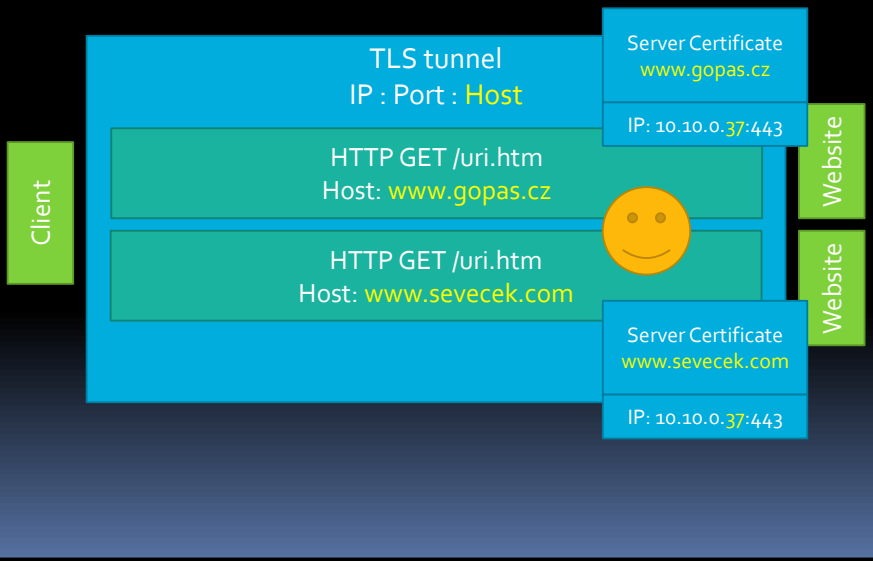

More web servers on a common IP address - host header
Server Certificate
www.gopas.cz
IP: 10.10.0.37:443
TLS tunnel
IP : Port
HTTP GET /uri.htm
Host: www.gopas.cz
HTTP GET /uri.htm
Host: www.sevecek.com
Client
Website
Website

41


Host header vs. wildcard certificiate
Server Certificate
*.gopas.cz
IP: 10.10.0.37:443
TLS tunnel
IP : Port
HTTP GET /uri.htm
Host: www.gopas.cz
HTTP GET /uri.htm
Host: kurzy.gopas.cz
Client
Website
Website

42

Server Name Indication (SNI)



Server Name Indication (SNI)

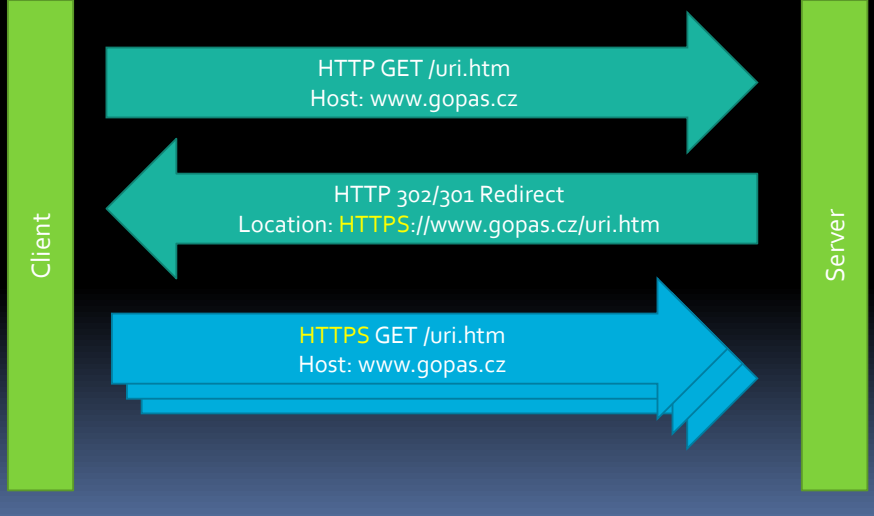
- Supported by Windows 2012 HTTP.SYS
- Supported by Windows Vista/2008 client SCHANNEL
  - IE 7
  - Firefox 2.0
  - Opera 8.0
  - Opera Mobile 10.1
  - Chrome 6
  - Safari 2.1
  - Windows Phone 7

TLS

PROTOCOL ATTACKS AND FIXES

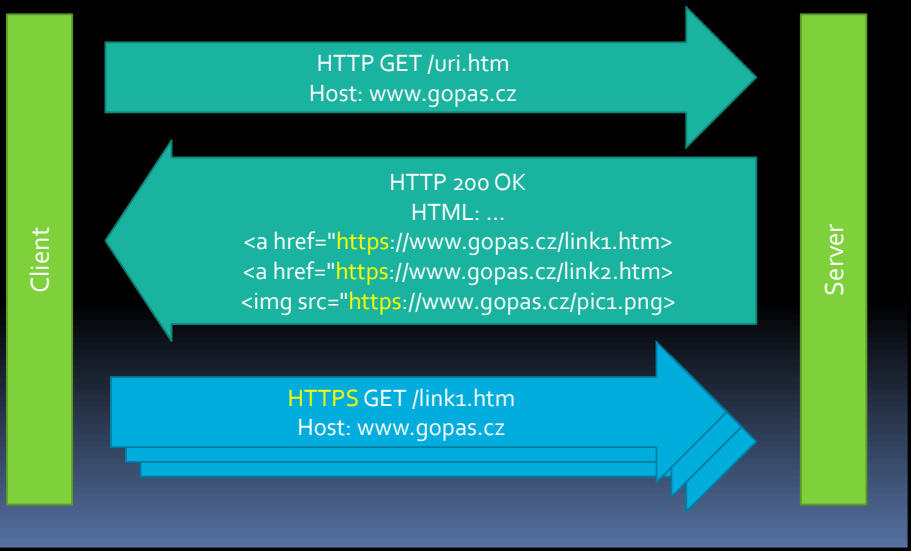
45

Client started HTTP and 302/301 redirect



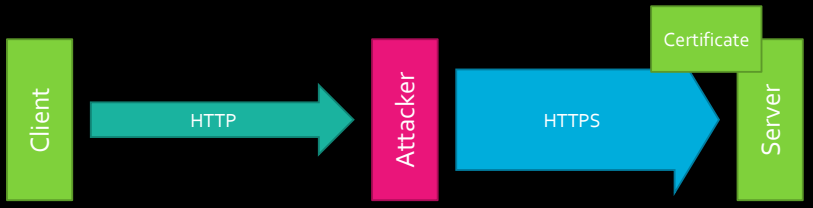
46

Client started HTTP and HTTPS links <a href>

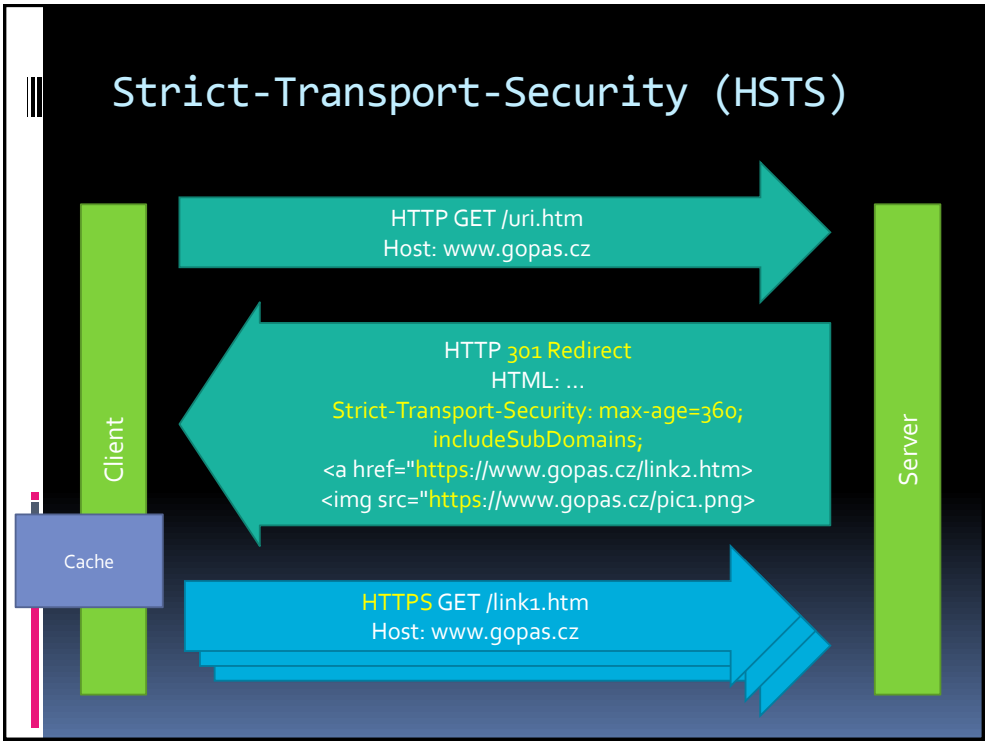


47

MITM attack with HTTPS strip (SSL strip)

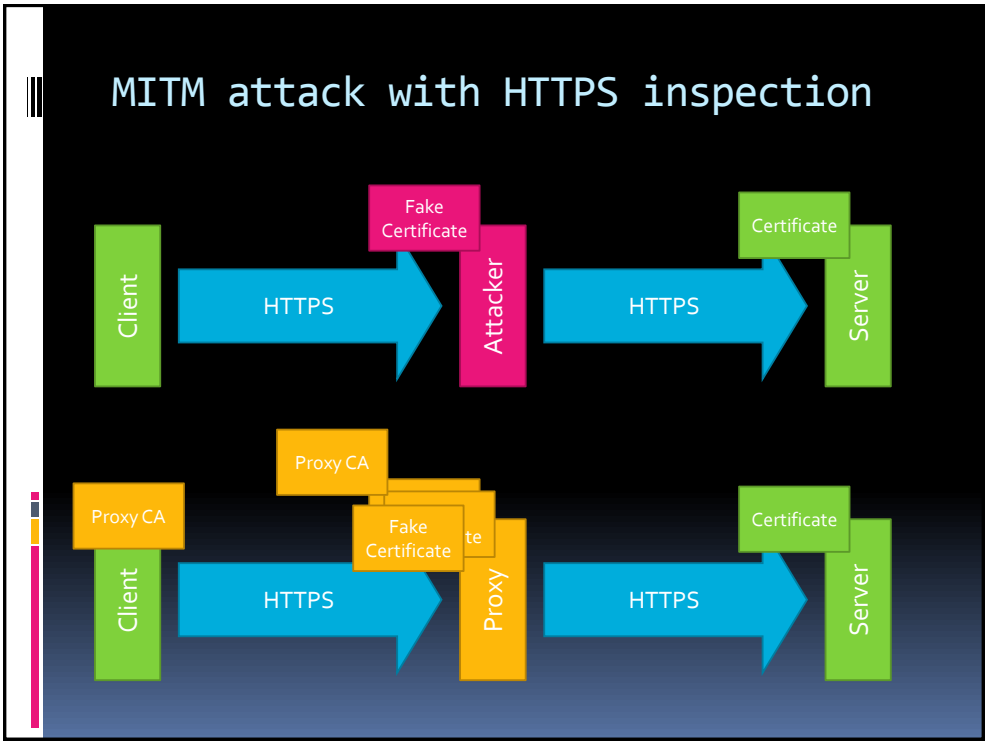


48

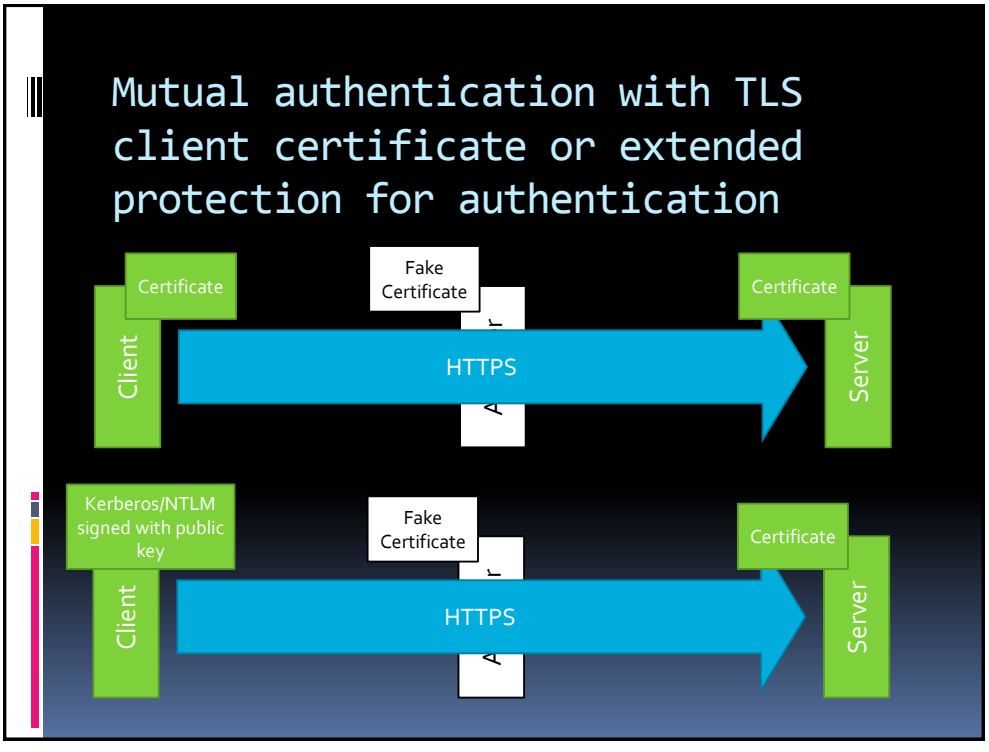


Chromium preload list

- Strict-Transport-Security: max-age=10886400; includeSubDomains; preload
- https://hstspreload.appspot.com/


MITM attack with HTTPS inspection
Client
HTTPS
Fake Certificate
Attacker
HTTPS
Certificate
Server
Proxy CA
Proxy CA
Client
HTTPS
Fake Certificate
Proxy
HTTPS
Certificate
Server

51


Mutual authentication with TLS client certificate or extended protection for authentication
Certificate
Client
Fake Certificate
HTTPS
Certificate
Server
Kerberos/NTLM signed with public key
Client
Fake Certificate
HTTPS
Certificate
Server

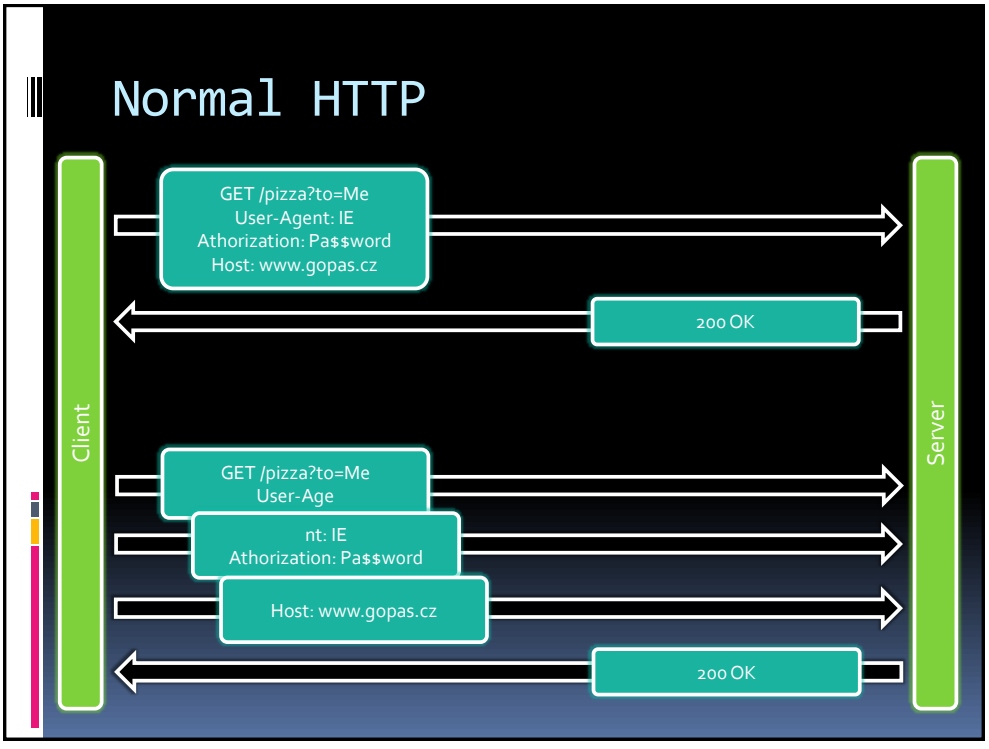
52

Cryptographic downgrade

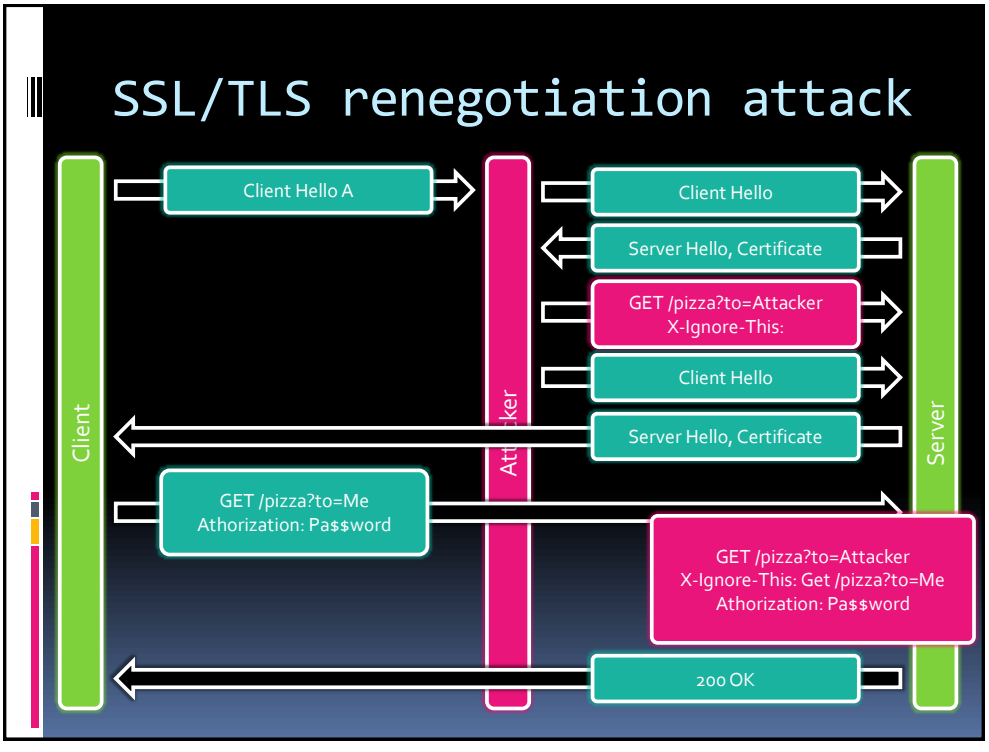
- active MITM can limit the client's offer to the least secure algorithm specified by the server
- Prevent by disabling insecure suites on the server side

SSL/TLS renegotiation attack

- TLS 1.0 and SSL 3.0 problem
  - TLS 1.1 and TLS 1.2 do not have this issue
- active MITM can prepend its own data before client's request


Normal HTTP
Client
Server
GET /pizza?to=Me
User-Agent: IE
Athorization: Pa$$word
Host: www.gopas.cz
200 OK
GET /pizza?to=Me
User-Age
nt: IE
Athorization: Pa$$word
Host: www.gopas.cz
200 OK

55


SSL/TLS renegotiation attack
Client
Server
Client Hello A
Client Hello
Server Hello, Certificate
GET /pizza?to=Attacker
X-Ignore-This:
Client Hello
Server Hello, Certificate
GET /pizza?to=Me
Athorization: Pa$$word
GET /pizza?to=Attacker
X-Ignore-This: Get /pizza?to=Me
Athorization: Pa$$word
200 OK

56

SSL/TLS renegotiation attack

- KB980436 enables/enforces RFC 5746
  - must install on both server and client

SSL/TLS renegotiation attack

- Renegotiation Info extension
  - sent by clients, required by servers
  - by default client and server are compatible
- Strict/Compatible SERVER
  - AllowInsecureRenegoClients = 0/1
- Strict/Compatible CLIENT
  - AllowInsecureRenegoServers = 0/1

SSL/TLS renegotiation attack

- Older TLS servers may have problems with Renegotiation Info extension
  - can be changed from an extension to a suite 00FF on client side
  - UseScsvForTls = DWORD = 1

SSL/TLS renegotiation attack

- KB977377 allows to disable renegotiation at all
  - problems with SSL Client Certificates if not required on the site level
- HKLM\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL
  - DisableRenegoOnClient = 1/0
  - DisableRenegoOnServer = 1/0

TLS Beast attack

- TLS 1.0 and SSL 3.0 problem
  - TLS 1.1 and TLS 1.2 do not have this issue
- CBC - next IV is taken as the last cipher-text block
  - if you can make the victim's requests split authentication cookie one by one character into different packets, you can guest the cookie
- Requires same-origin injection

TLS Beast attack

- Patched by RFC 2246
  - KB2585542 for Windows Vista and newer
  - KB2638806 for Windows 2003/XP
- TLS Application Data Fragmentation
  - splits application data into several packets
- Server application should be protected against script injection

TLS Beast attack

- Must be used willingly by a patched client (IE, Outlook, etc.)
  - patched servers only support the protection
- If the server replies with fragmented application data, some unpatched client applications may fail

TLS Beast attack

- Can enforce:
  HKLM\System\CCS\Control\SecurityProviders\SCHANNEL
  SendExtraRecord = DWORD = 1
- Can disable at all
  SendExtraRecord = DWORD = 2
  - but you are vulnerable again
- Default setting to let client apps decide and server protect itself
  SendExtraRecord = DWORD = 0

RC4 weakness

- capture 1 000 000 000 TLS connections
- first 220 bytes of TLS encrypted data starting at 37th byte

- RFC 7465 prohibited since Feb2015
- MS does not recommend since 2015

LogJam

- Downgrade DH to 512 (DH_EXPORT)
  - all Windows Servers since 2003 has minimum of 1024 bit DH keys and are not vulnerable
- patched since May 2015 (KB3061518)
  - client side prevents 512 bit DH keys by default
  - HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\KeyExchangeAlgorithms\Diffie-Hellman
    - ClientMinKeyBitLength = DWORD = 1024, …

Do I have the hotfix?

- PowerShell

```
gwmi win32_quickfixengineering |
? { $_.HotfixId -eq 'KB980436' }
```

67

TLS

SIDE CHANNEL ATTACKS

68

Side channel attacks

- SSL stripping
  - MITM downgrades HTTPS:// links to HTTP://
  - MITM downgrades 302 redirects to HTTP://
- Cross-site scripting (XSS)
  - malicious script in a trusted web page
- Cross-site request forgery (CSRF)
  - link/picture that does something in a different, still authenticated web page
  - XSS + POST can be even more severe

SSL Strip

- Enforce TLS on the server side

CRIME attack

- TLS compression
  - if you are able to inject something similar into the internal HTTP through client's own browser (CSS/CSRF), it will shrink the traffic
- SCHANNEL does not support TLS compression at all
  - RFC 3749 - also known as DEFLATE

BREACH attack

- Attacks HTTP (non S) compression
  - server side GZIP, DEFLATE
  - server must reflect user input, CSRF must be employed
  - OWA does!
- Disable compression on the server side

TLS

VALIDATING TLS CONFIGURATION

73

Validating public TLS servers

- http://www.ssllabs.com

74

TLS performance (TLS 1.2) Test-Tls function

- 1 server CPU, 8 client CPUs
  - no certificate validation on client
  - server certificate only
    - RSA 2048
    - RSA 4096
- client running at 90% CPU
- TCP 3389 - svchost.exe, 60% CPU
- TCP 443 - lsass.exe, 15% CPU
- ~126 sessions per second
  - full handshake (client hello, server hello, key ex)

General recommendation

- Go for TLS 1.1 and TLS 1.2
- Prefer ECDH suites (PFS)
- Support TLS 1.0 for backward compatibility
- Disable SSL 2.0 and SSL 3.0 and PCT
- Disable DES, 40bit, 1024bit, …
- Potentially disable RC4 and MD5

- UPDATE and TEST

Ing. Ondřej Ševeček | PM Windows Server | GOPAS a.s. |
MCM: Directory Services | MVP: Enterprise Security | CEH
ondrej@sevecek.com | www.sevecek.com |

THANK YOU!