SHARKFEST '12

Wireshark Developer and User Conference

SSL Troubleshooting with Wireshark and Tshark

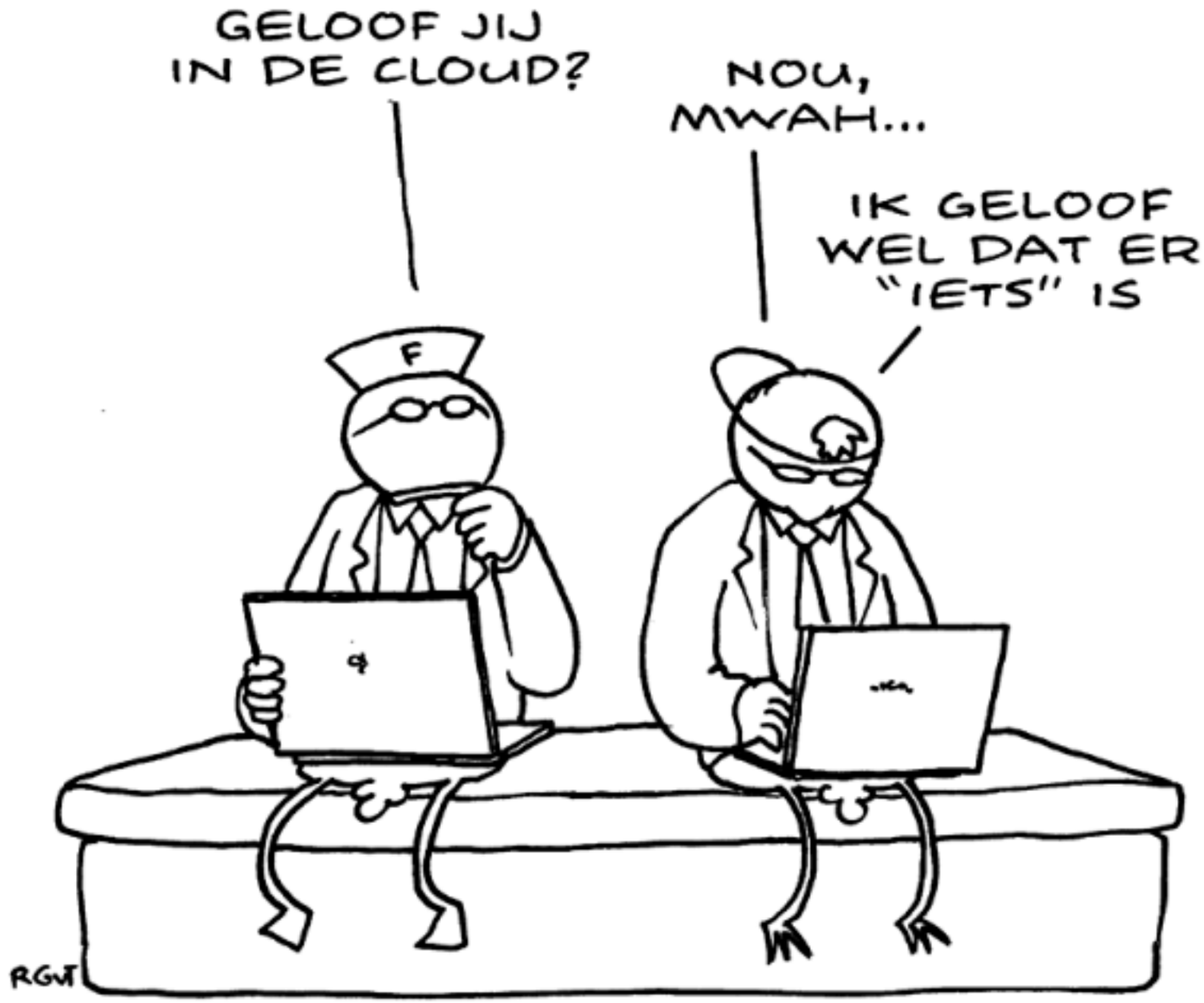
Sake Blok

Application Delivery Networking Consultant and Troubleshooter

sake.blok@SYN-bit.nl

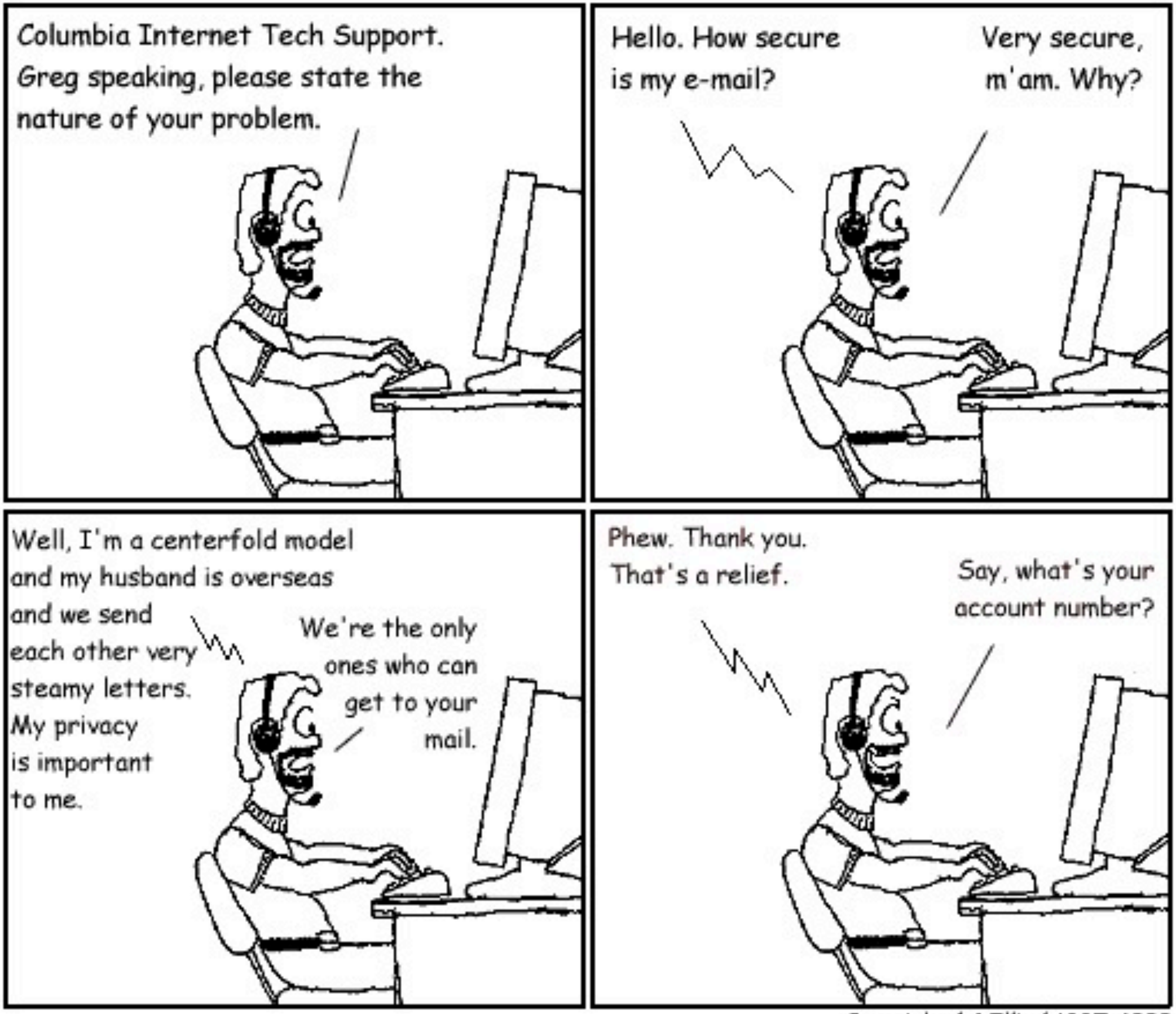
FOKKE & SUKKE

SLUITEN DE MAAND VAN DE FILOSOFIE AF



www.foksuk.nl

USER FRIENDLY by Illiad



About you?

- Who…
  - …thinks SSL is just about encryption?
  - …troubleshooted SSL traffic before?
  - …knows the purpose of each handshake message?
  - …tried to decrypt SSL traffic before?
  - …and ran into problems decrypting?
  - …troubleshooted client authentication problems?

About me?

- Started SYN-bit in 2009
- Application Delivery Networking Consultant & Troubleshooter (F5 Networks, Cisco ACE, Alteon)
- Have used SSL extensively in customer projects
- Using Ethereal since 1999, developing since 2006, member core-developers since 2007
- Enjoy scuba diving and art-house movies

Sake Blok sake.blok@SYN-bit.nl

Challenges

- Confidentiality
  - Encryption and Decryption
- Message Integrity
  - Message Digest and Message Signing
- Endpoint Authentication & Non-repudiation
  - Certificates and Certificate Authorities

SSL

Agenda

- Cryptology overview
- The SSL protocol
- Analyzing SSL with Wireshark
- Analyzing SSL with Tshark
- Common SSL connection problems
- Further reading
- Questions & Discussion

Agenda

- Cryptology overview
- The SSL protocol
- Analyzing SSL with Wireshark
- Analyzing SSL with Tshark
- Common SSL connection problems
- Further reading
- Questions & Discussion

Symetric Encryption

- Same key for encryption and decryption
- Computatively "cheap"
- Short keys (typically 40-256 bits)
- DES, 3DES, AESxxx, RC4

- Confidentiality?
- One-to-many?

Asymetric Encryption

- One key for encryption, second key for decryption (both keys form a pair)
- Computatively "expensive"
- Long keys (typically 512-4096 bits)
- RSA, DSA

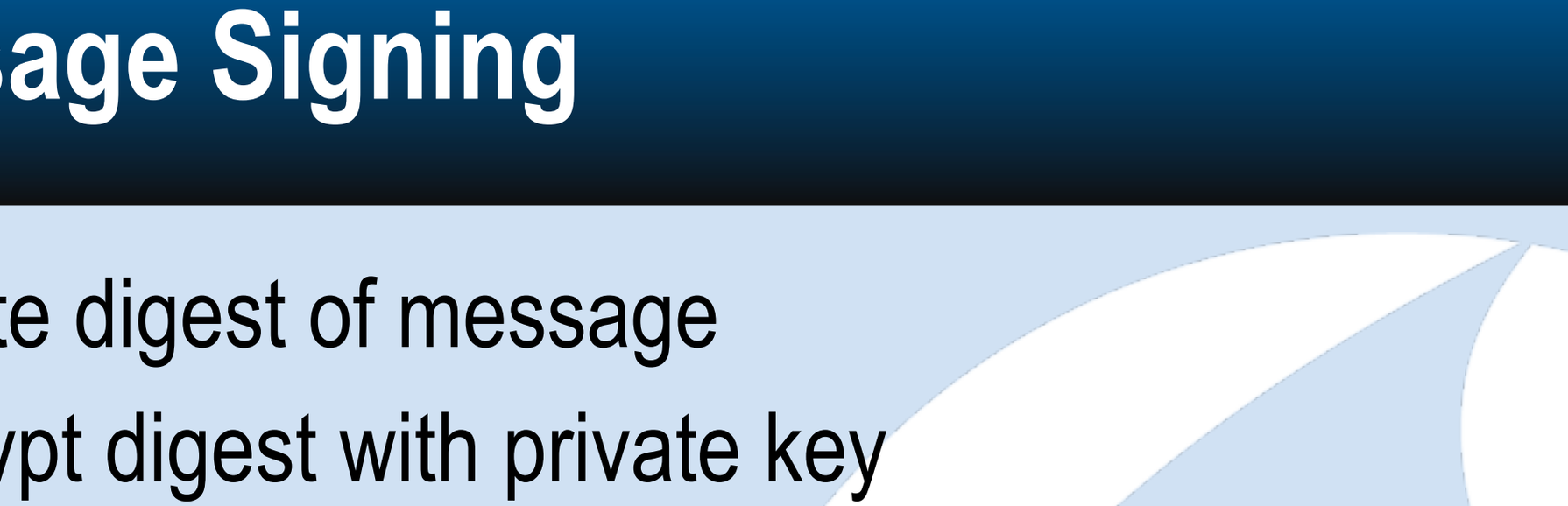
- Confidentiality?
- Authentication?
- One-to-many?

Hashing / Message Digest

- Irreversible
  - original text not reproducable from the digest
- Collision-resistance
  - "Not possible" to create a message M' so that it has the same digest as message M
- MD5, SHA-1, SHA-2

Message Signing

- Create digest of message
- Encrypt digest with private key
- Authenticity and sender of message can be checked with public key



Digital Certificates

"In cryptography, a public key certificate (or identity certificate) is an **electronic document** which utilizes a **digital signature** to bind together a **public key** with an **identity**."

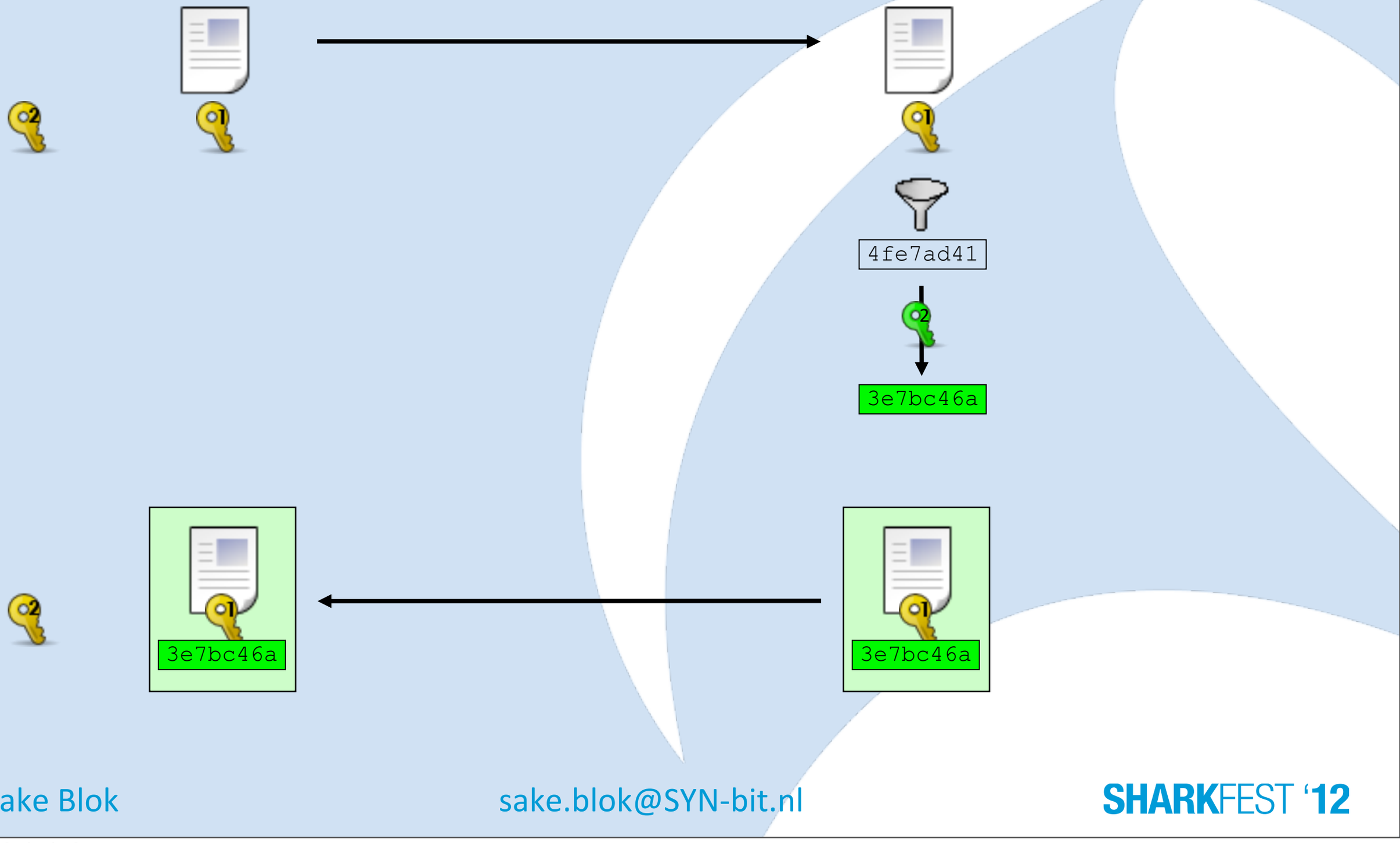
(From http://en.wikipedia.org/wiki/Digital_certificate)

But who is signing???

Certificate Authorities

- Mutually trusted by sender and receiver
- "Solves" key exchange problems
- CA's can be chained
- Top of chain is "self-signed" (and is called the "Root CA")

Creating a certificate



Agenda

- Cryptology overview
- The SSL protocol
- Analyzing SSL with Wireshark
- Analyzing SSL with Tshark
- Common SSL connection problems
- Further reading
- Questions & Discussion

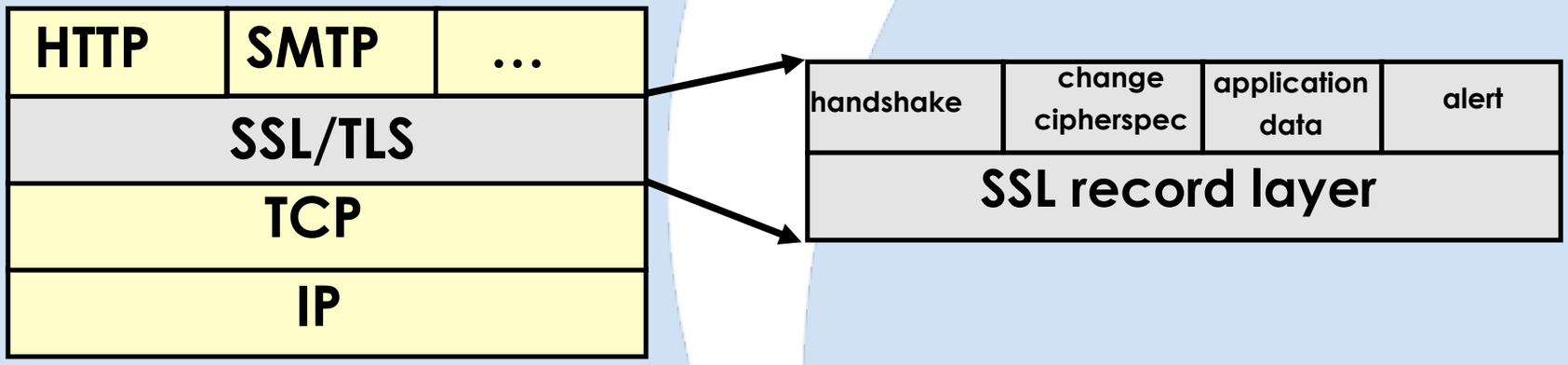
SSL History

- SSLv1 by Netscape (unreleased, 1994)
- SSLv2 by Netscape (v2-draft,1994)
- SSLv3 by Netscape (v3-draft, 1995)
- TLSv1.0, IETF (RFC 2246, 1999)
- TLSv1.1, IETF (RFC 4346, 2006)
- TLSv1.2, IETF (RFC 5246, 2008)
- Risks and differences explained at:

  http://www.yaksman.org/~lweith/ssl.pdf

Place in TCP/IP stack

- Between transport and application layer
- Protocol independent



SSL Record Layer

- Provides fragmentation (max size 2^14)
- Multiple SSL messages (of one content type) per SSL Record allowed

- SSL Record can be split over multiple TCP-segments (2^14 > MSS!)
- One TCP-segment can contain multiple SSL Records (or fragments)

SSL Content Types

- Handshake Protocol (0x16)
  - responsible for authentication and session key setup
- ChangeCipherSpec Protocol (0x14)
  - Notify start of encryption
- Alert Protocol (0x15)
  - Reporting of warnings and fatal errors
- Application Protocol (0x17)
  - Actual encryption and transport of data

Agenda

- Cryptology overview
- The SSL protocol
- Analyzing SSL with Wireshark
- Analyzing SSL with Tshark
- Common SSL connection problems
- Further reading
- Questions & Discussion

Lab setup

Client
WinXP - Firefox

Client-cert 1: Neo
Client-cert 2: Trinity

.1

192.168.3.0/24

.3

.4

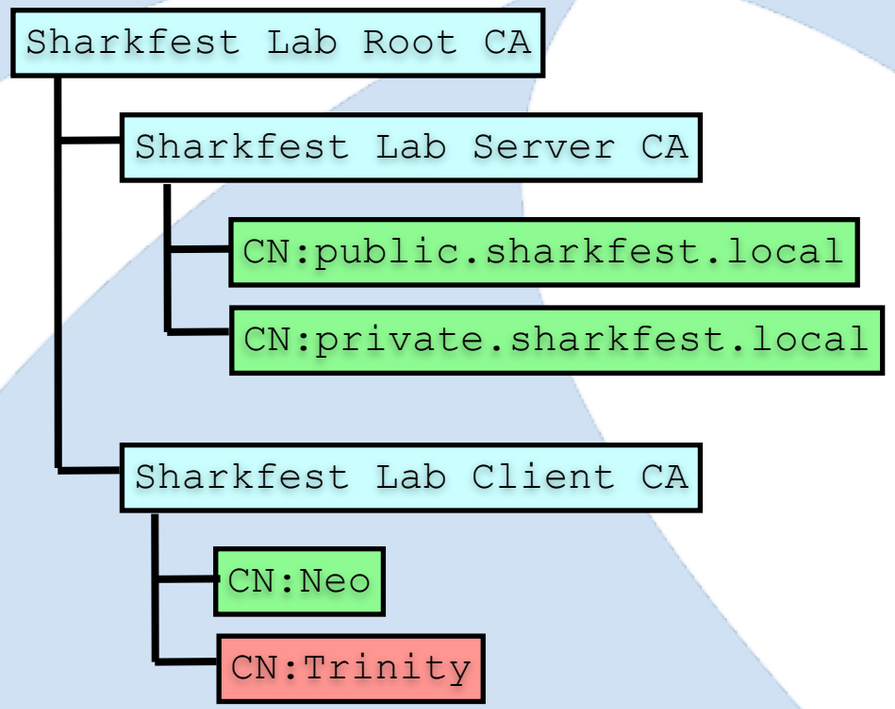
(virtual) Server 1
Ubuntu - Apache

Server-cert:
public.sharkfest.local

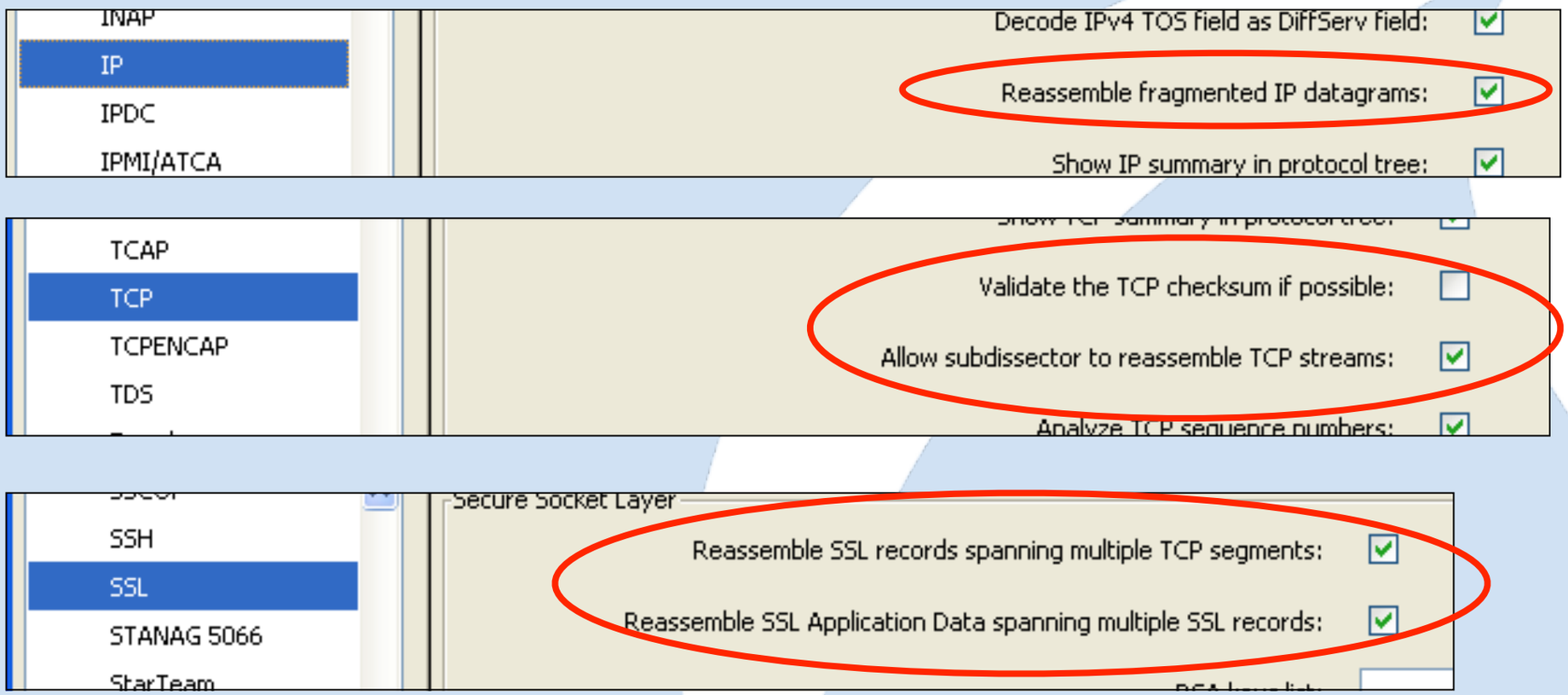
(virtual) Server 2
Ubuntu - Apache

Server-cert:
private.sharkfest.local

Require Client-Certificate



Choosing the right settings




```
ip.defragment: TRUE
tcp.check_checksum: FALSE
tcp.desegment_tcp_streams: TRUE
ssl.desegment_ssl_records: TRUE
ssl.desegment_ssl_application_data: TRUE
```

Analyzing the SSL record layer (1)

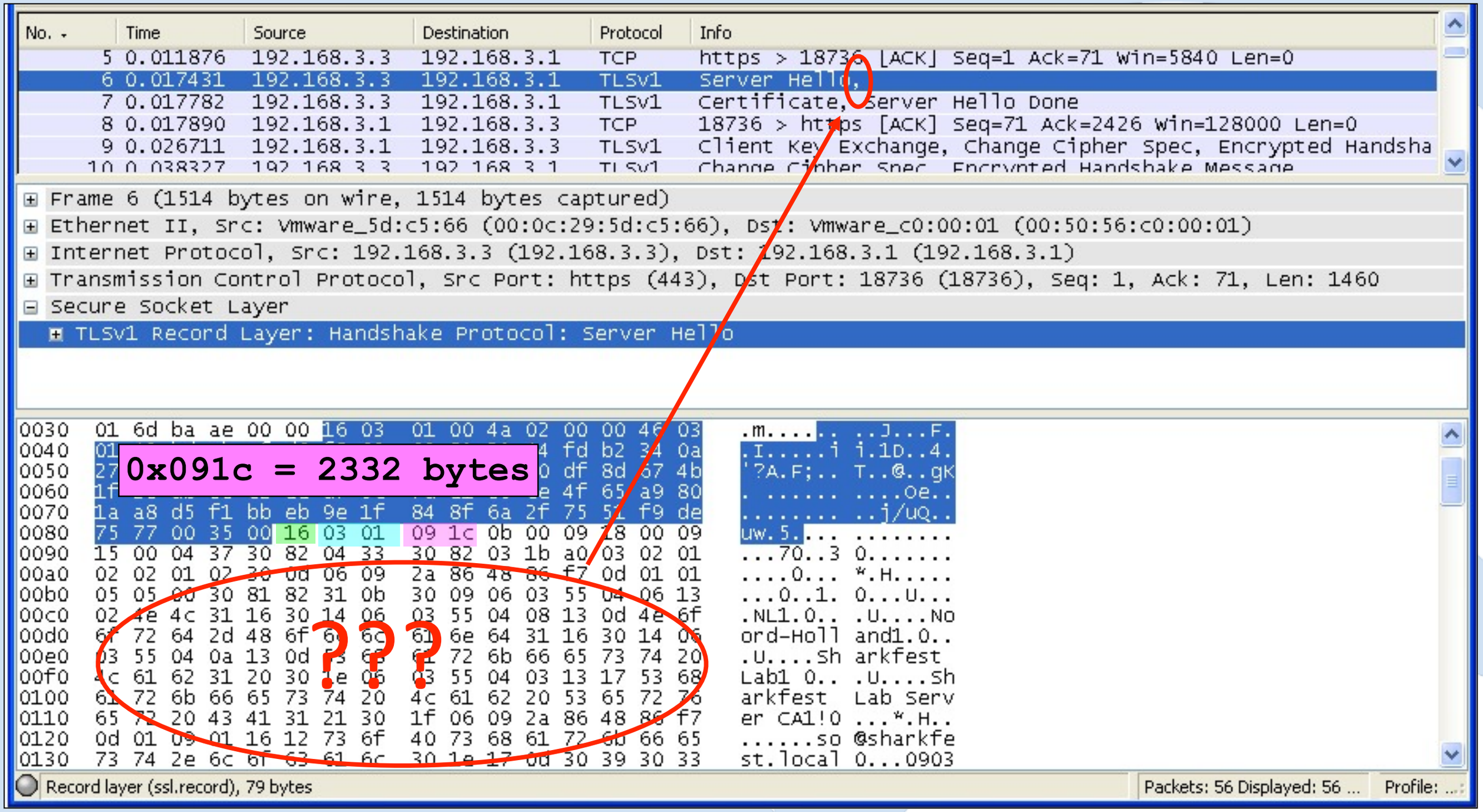
No.	Time	Source	Destination	Protocol	Info
4	0.011511	192.168.3.1	192.168.3.3	SSL	Client Hello
5	0.011876	192.168.3.3	192.168.3.1	TCP	https > 18736 [ACK] Seq=1 Ack=71 Win=5840 Len=0
6	0.017431	192.168.3.3	192.168.3.1	TLSv1	Server Hello,
7	0.017782	192.168.3.3	192.168.3.1	TLSv1	Certificate, Server Hello Done
8	0.017890	192.168.3.1	192.168.3.3	TCP	18736 > https [ACK] Seq=71 Ack=2426 Win=128000 Len=0

```
⊞ Frame 4 (124 bytes on wire, 124 bytes captured)
⊞ Ethernet II, Src: Vmware_c0:00:01 (00:50:56:c0:00:01), Dst: Vmware_5d:c5:66 (00:0c:29:5d:c5:66)
⊞ Internet Protocol, Src: 192.168.3.1 (192.168.3.1), Dst: 192.168.3.3 (192.168.3.3)
⊞ Transmission Control Protocol, Src Port: 18736 (18736), Dst Port: https (443), Seq: 1, Ack: 1, Len: 70
⊟ Secure Socket Layer
  ⊟ TLSv1 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 65
    ⊞ Handshake Protocol: Client Hello
```

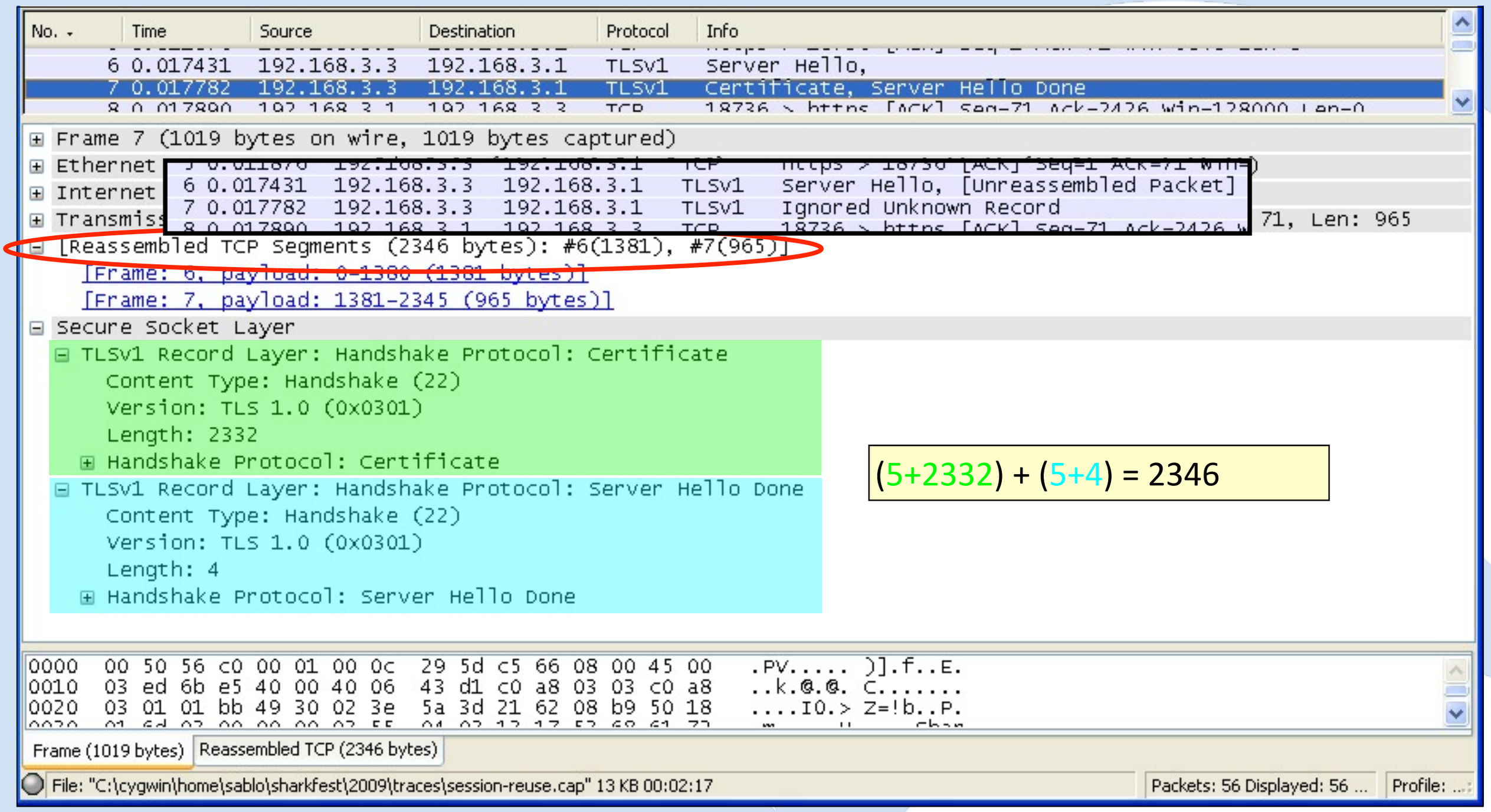
```
0000  00 0c 29 5d c5 66 00 50  56 c0 00 01 08 00 45 00   ..)].f.P V.....E.
0010  00 6e 42 29 40 00 80 06  31 0c c0 a8 03 01 c0 a8   .nB)@... 1.......
0020  03 03 49 30 01 bb 21 62  08 73 02 3e 54 89 50 18   ..I0..!b .s.>T.P.
0030  fa 00 67 eb 00 00 16 03  01 00 41 01 00 00 3d 03   ..g..... ..A...=.
0040  01 49 eb 46 9e dd 81 95  16 fc 5d dd d0 97 42 8d   .I.F.... ..]...B.
0050  41 0d 78 52 e2 57 9e 2e  89 03 cd b3 31 c7 63 dc   A.xR.W.. ....1.c.
0060  a9 00 00 10 00 84 00 35  00 41 00 04 00 05 00 2f   .......5 .A...../
0070  fe ff 00 0a 01 00 00 04  00 23 00 00               ........ .#..
```

File: "C:\cygwin\home\sablo\sharkfest\2009\traces\session-reuse.cap" 13 KB 00:02:17 | Packets: 56 Displayed: 56 ... | Profile: ...

Analyzing the SSL record layer (2)



Analyzing the SSL record layer (3)

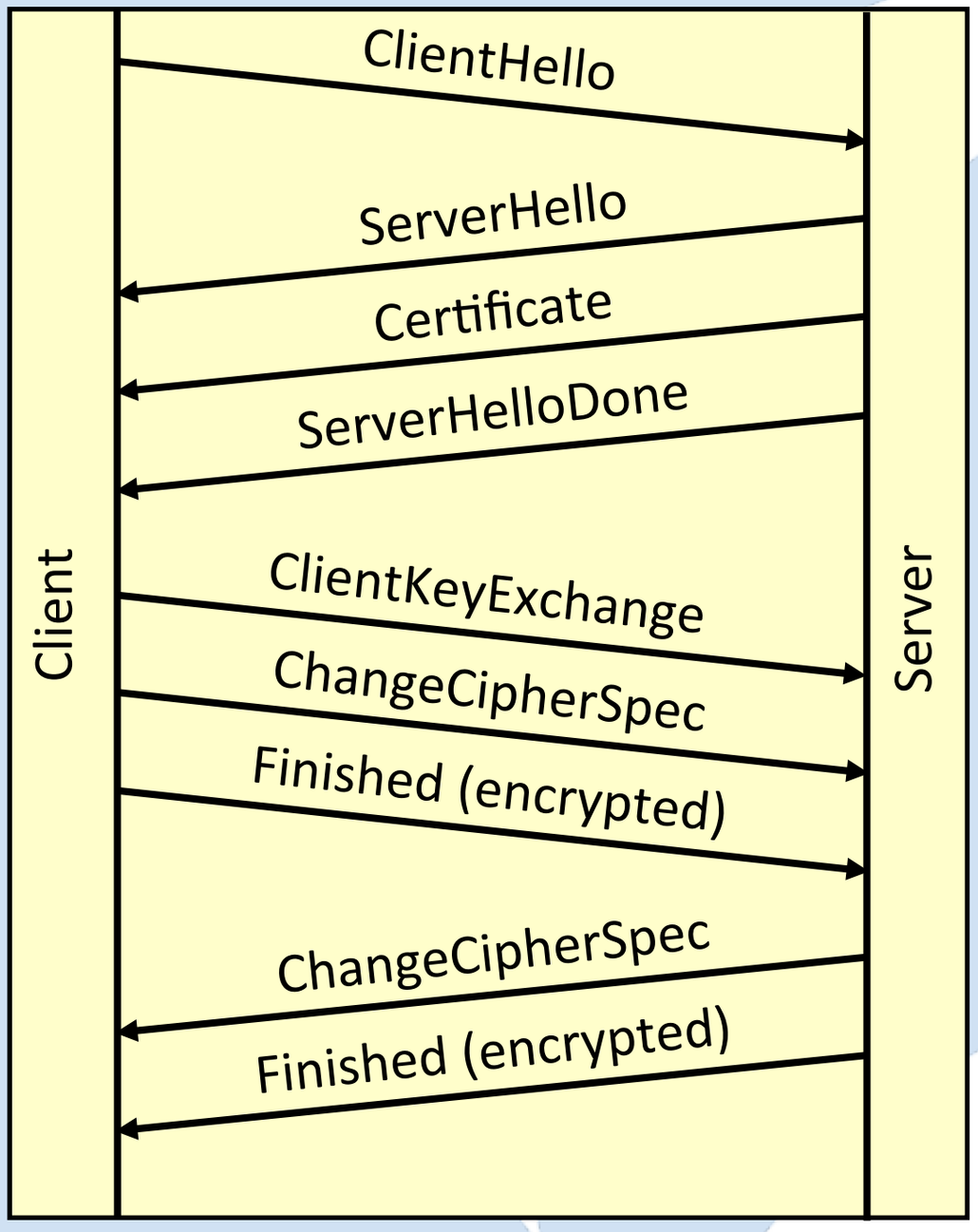


(5+2332) + (5+4) = 2346

Analyzing the SSL handshake

- Normal RSA handshake
- Ephemeral RSA (or DH) handshake
- SSL session with client authentication
- Reusing SSL sessions
  - Reused SSL session (partial handshake)
  - Expired SSL session
  - No SSL reuse

Normal RSA handshake



… in Wireshark

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.3	TCP	18736 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000309	192.168.3.3	192.168.3.1	TCP	https > 18736 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=
3	0.000357	192.168.3.1	192.168.3.3	TCP	18736 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.011511	192.168.3.1	192.168.3.3	TLSv1	Client Hello
5	0.011876	192.168.3.3	192.168.3.1	TCP	https > 18736 [ACK] Seq=1 Ack=71 Win=5840 Len=0
6	0.017431	192.168.3.3	192.168.3.1	TLSv1	Server Hello,
7	0.017782	192.168.3.3	192.168.3.1	TLSv1	Certificate, Server Hello Done
8	0.017890	192.168.3.1	192.168.3.3	TCP	18736 > https [ACK] Seq=71 Ack=2426 Win=128000 Len=0
9	0.026711	192.168.3.1	192.168.3.3	TLSv1	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Mes
10	0.038327	192.168.3.3	192.168.3.1	TLSv1	Change Cipher Spec, Encrypted Handshake Message

ClientHello

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 65
    Handshake Protocol: Client Hello
        Handshake Type: Client Hello (1)
        Length: 61
        Version: TLS 1.0 (0x0301)
      Random
          gmt_unix_time: Apr 19, 2009 17:43:26.000000000
          random_bytes: DD819516FC5DDDD097428D410D7852E2579E2E8903CDB331...
        Session ID Length: 0
        Cipher Suites Length: 16
      Cipher Suites (8 suites)
          Cipher Suite: TLS_RSA_WITH_CAMELLIA_256_CBC_SHA (0x0084)
          Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
          Cipher Suite: TLS_RSA_WITH_CAMELLIA_128_CBC_SHA (0x0041)
          Cipher Suite: TLS_RSA_WITH_RC4_128_MD5 (0x0004)
          Cipher Suite: TLS_RSA_WITH_RC4_128_SHA (0x0005)
          Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
          Cipher Suite: SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA (0xfeff)
          Cipher Suite: TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000a)
        Compression Methods Length: 1
      Compression Methods (1 method)
          Compression Method: null (0)
        Extensions Length: 4
      Extension: SessionTicket TLS
          Type: SessionTicket TLS (0x0023)
          Length: 0
          Data (0 bytes)
```

ServerHello

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 74
    Handshake Protocol: Server Hello
        Handshake Type: Server Hello (2)
        Length: 70
        Version: TLS 1.0 (0x0301)
      Random
          gmt_unix_time: Mar 16, 2009 02:30:23.000000000
          random_bytes: D6F56969813144FDB2340A273F419E463BF915549B0740DF...
        Session ID Length: 32
        Session ID: DB00C2AAD79CFDA109CE4F65A9801AA8D5F1BBEB9E1F848F...
        Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
        Compression Method: null (0)
```

Certificate (1)

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Certificate
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 2332
    Handshake Protocol: Certificate
        Handshake Type: Certificate (11)
        Length: 2328
        Certificates Length: 2325
      Certificates (2325 bytes)
          Certificate Length: 1079
        Certificate ()
          Certificate Length: 1240
        Certificate ()
  TLSv1 Record Layer: Handshake Protocol: Server Hello Done
```

Certificate (2)

```
Handshake Protocol: Certificate
    Handshake Type: Certificate (11)
    Length: 2328
    Certificates Length: 2325
  Certificates (2325 bytes)
      Certificate Length: 1079
    Certificate ()
      signedCertificate
          version: v3 (2)
          serialNumber: 2
        signature (shaWithRSAEncryption)
        issuer: rdnSequence (0)
        validity
        subject: rdnSequence (0)
        subjectPublicKeyInfo
        extensions: 4 items
      algorithmIdentifier (shaWithRSAEncryption)
        Algorithm Id: 1.2.840.113549.1.1.5 (shaWithRSAEncryption)
      Padding: 0
      encrypted: 739D20C79873ADD406549E824AE1304525EEA1A5E185FB0B...
  Certificate Length: 1240
Certificate ()
```

Certificate (3)

```
validity
subject: rdnSequence (0)
  rdnSequence: 5 items ()
    RDNSequence: 1 item ()
      RelativeDistinguishedName
          Id: 2.5.4.6 (id-at-countryName)
          CountryName: NL
    RDNSequence: 1 item ()
      RelativeDistinguishedName
          Id: 2.5.4.8 (id-at-stateOrProvinceName)
        DirectoryString: printableString (1)
            printableString: Noord-Holland
    RDNSequence: 1 item ()
      RelativeDistinguishedName
          Id: 2.5.4.10 (id-at-organizationName)
        DirectoryString: printableString (1)
            printableString: Sharkfest Lab
    RDNSequence: 1 item ()
      RelativeDistinguishedName
          Id: 2.5.4.3 (id-at-commonName)
        DirectoryString: printableString (1)
            printableString: public.sharkfest.local
    RDNSequence: 1 item ()
      RelativeDistinguishedName
          Id: 1.2.840.113549.1.9.1 (pkcs-9-at-emailAddress)
          SyntaxIA5String: co@sharkfest.local
subjectPublicKeyInfo
```

Certificate (4)

```
Certificate ()
signedCertificate
    version: v3 (2)
    serialNumber: 2
  ⊞ signature (shaWithRSAEncryption)
  ⊟ issuer: rdnSequence (0)
    ⊟ rdnSequence: 5 items ()
      ⊞ RDNSequence: 1 item ()
      ⊞ RDNSequence: 1 item ()
      ⊞ RDNSequence: 1 item ()
      ⊟ RDNSequence: 1 item ()
        ⊟ RelativeDistinguishedName
            Id: 2.5.4.3 (id-at-commonName)
          ⊟ DirectoryString: printableString (1)
              printableString: Sharkfest Lab Server CA
      ⊞ RDNSequence: 1 item ()
  ⊞ validity
  ⊞ subject: rdnSequence (0)
  ⊞ subjectPublicKeyInfo
  ⊞ extensions: 4 items
⊞ algorithmIdentifier (shaWithRSAEncryption)
  Padding: 0
  encrypted: 739D20C79873ADD406549E824AE1304525EEA1A5E185FB0B...
Certificate Length: 1240
⊟ Certificate ()
⊟ signedCertificate
    version: v3 (2)
    serialNumber: 1
  ⊞ signature (shaWithRSAEncryption)
  ⊞ issuer: rdnSequence (0)
  ⊞ validity
  ⊟ subject: rdnSequence (0)
    ⊟ rdnSequence: 5 items ()
      ⊞ RDNSequence: 1 item ()
      ⊞ RDNSequence: 1 item ()
      ⊞ RDNSequence: 1 item ()
      ⊟ RDNSequence: 1 item ()
        ⊟ RelativeDistinguishedName
            Id: 2.5.4.3 (id-at-commonName)
          ⊟ DirectoryString: printableString (1)
              printableString: Sharkfest Lab Server CA
```

ServerHelloDone

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Certificate
  TLSv1 Record Layer: Handshake Protocol: Server Hello Done
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 4
    Handshake Protocol: Server Hello Done
      Handshake Type: Server Hello Done (14)
      Length: 0
```

ClientKeyExchange

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Client Key Exchange
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 134
    Handshake Protocol: Client Key Exchange
        Handshake Type: Client Key Exchange (16)
        Length: 130
  TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message
```

ChangeCipherSpec (C)

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Client Key Exchange
  TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
      Content Type: Change Cipher Spec (20)
      Version: TLS 1.0 (0x0301)
      Length: 1
      Change Cipher Spec Message
  TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message
```

Finished (C)

Without decryption:

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Client Key Exchange
  TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 48
      Handshake Protocol: Encrypted Handshake Message
```

With decryption:

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Client Key Exchange
  TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  TLSv1 Record Layer: Handshake Protocol: Finished
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 48
    Handshake Protocol: Finished
        Handshake Type: Finished (20)
        Length: 12
        Verify Data
```

ChangeCipherSpec (S)

```
Secure Socket Layer
  TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
      Content Type: Change Cipher Spec (20)
      Version: TLS 1.0 (0x0301)
      Length: 1
      Change Cipher Spec Message
  TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message
```

Finished (S)

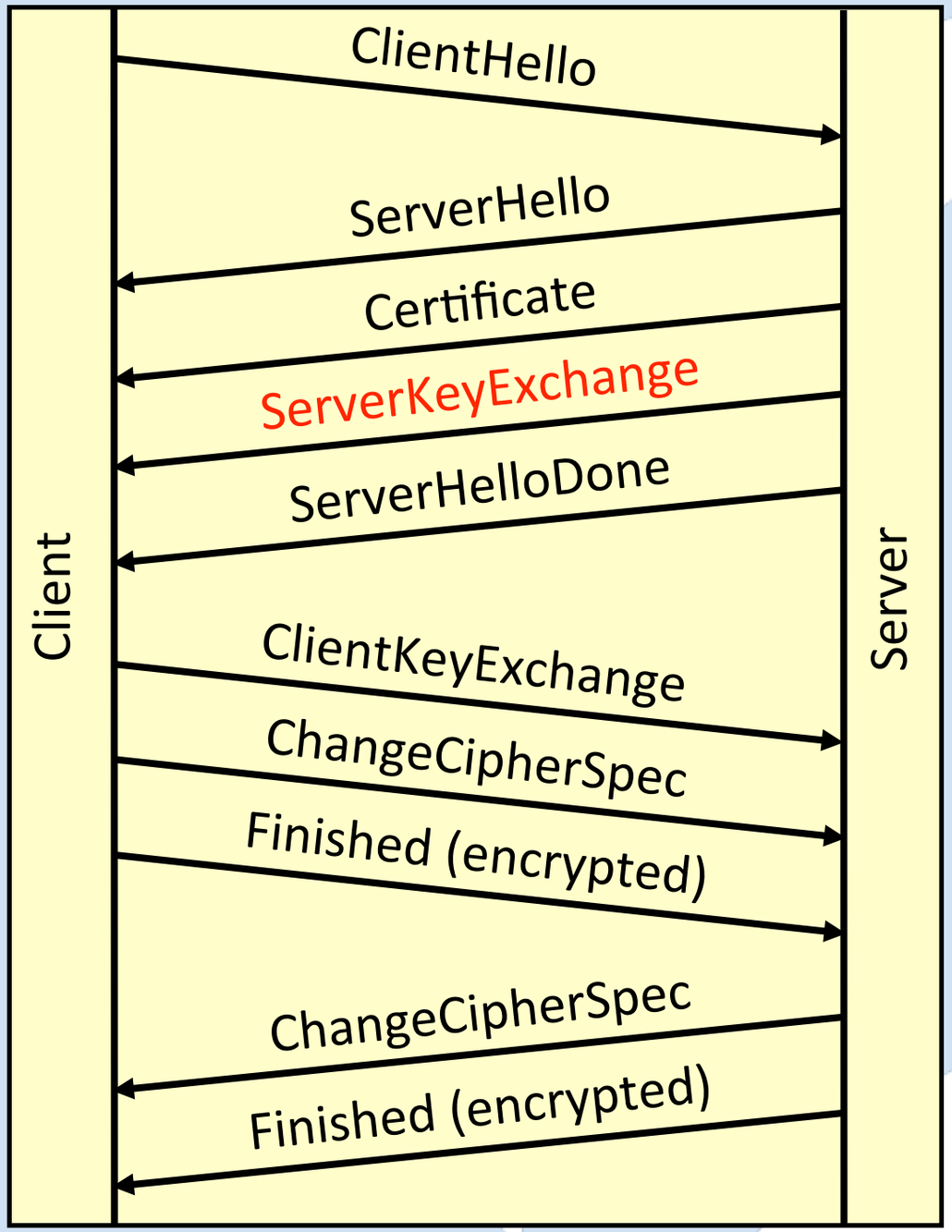
Without decryption:

```
Secure Socket Layer
  TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 48
      Handshake Protocol: Encrypted Handshake Message
```

With decryption:

```
Secure Socket Layer
  TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  TLSv1 Record Layer: Handshake Protocol: Finished
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 48
    Handshake Protocol: Finished
        Handshake Type: Finished (20)
        Length: 12
        Verify Data
```

Ephemeral RSA (or DH) handshake



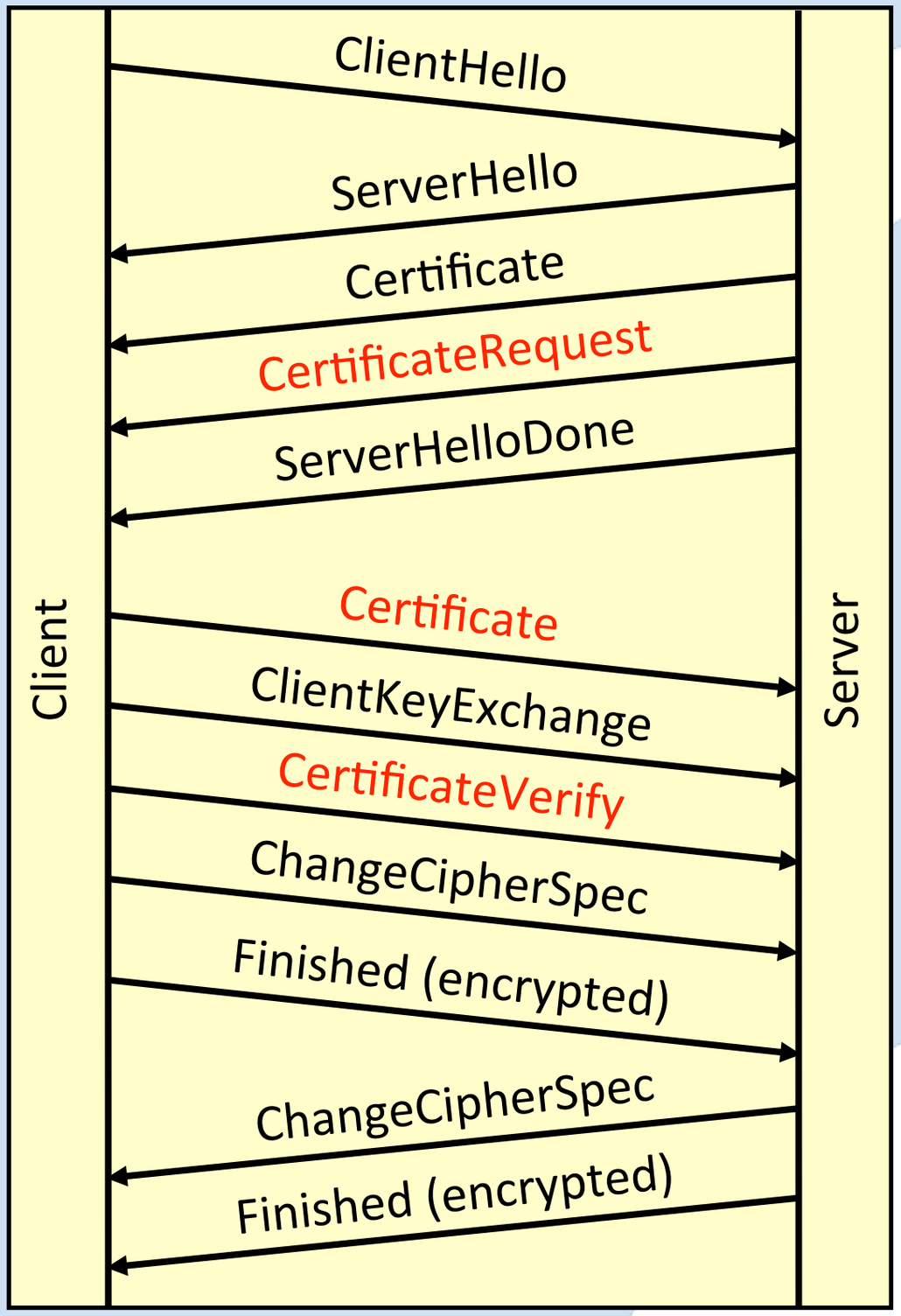
… in Wireshark

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.3	TCP	42370 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000577	192.168.3.3	192.168.3.1	TCP	https > 42370 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=146
3	0.000618	192.168.3.1	192.168.3.3	TCP	42370 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.026109	192.168.3.1	192.168.3.3	SSL	Client Hello
5	0.026465	192.168.3.3	192.168.3.1	TCP	https > 42370 [ACK] Seq=1 Ack=107 Win=5840 Len=0
6	0.070925	192.168.3.3	192.168.3.1	TLSv1	Server Hello,
7	0.071108	192.168.3.3	192.168.3.1	TLSv1	Certificate, Server Key Exchange, Server Hello Done
8	0.071172	192.168.3.1	192.168.3.3	TCP	42370 > https [ACK] Seq=107 Ack=2828 Win=128000 Len=0
9	0.090279	192.168.3.1	192.168.3.3	TLSv1	Client Key Exchange, Change Cipher Spec, Encrypted Handshak
10	0.090657	192.168.3.3	192.168.3.1	TCP	https > 42370 [ACK] Seq=2828 Ack=305 Win=6912 Len=0
11	0.110494	192.168.3.3	192.168.3.1	TLSv1	Change Cipher Spec, Encrypted Handshake Message

ServerKeyExchange

```
Secure Socket Layer
  TLSv1 Record Layer: Handshake Protocol: Certificate
  TLSv1 Record Layer: Handshake Protocol: Server Key Exchange
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 397
    Handshake Protocol: Server Key Exchange
        Handshake Type: Server Key Exchange (12)
        Length: 393
  TLSv1 Record Layer: Handshake Protocol: Server Hello Done
```

Client Authentication



… in Wireshark

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.4	TCP	14980 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000372	192.168.3.4	192.168.3.1	TCP	https > 14980 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=14
3	0.000400	192.168.3.1	192.168.3.4	TCP	14980 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.015645	192.168.3.1	192.168.3.4	SSLv2	Client Hello
5	0.015824	192.168.3.4	192.168.3.1	TCP	https > 14980 [ACK] Seq=1 Ack=52 Win=5840 Len=0
6	0.017894	192.168.3.4	192.168.3.1	SSLv3	Server Hello,
7	0.017988	192.168.3.4	192.168.3.1	SSLv3	Certificate, Certificate Request, Server Hello Done
8	0.018015	192.168.3.1	192.168.3.4	TCP	14980 > https [ACK] Seq=52 Ack=2590 Win=128000 Len=0
9	4.089191	192.168.3.1	192.168.3.4	TCP	[TCP segment of a reassembled PDU]
10	4.089622	192.168.3.4	192.168.3.1	TCP	https > 14980 [ACK] Seq=2590 Ack=1512 Win=8768 Len=0
11	4.089949	192.168.3.1	192.168.3.4	SSLv3	Certificate, Client Key Exchange, Certificate Verify, Chan
12	4.107141	192.168.3.4	192.168.3.1	SSLv3	Change Cipher Spec, Encrypted Handshake Message

CertificateRequest

```
Secure Socket Layer
  SSLv3 Record Layer: Handshake Protocol: Certificate
  SSLv3 Record Layer: Handshake Protocol: Multiple Handshake Messages
      Content Type: Handshake (22)
      Version: SSL 3.0 (0x0300)
      Length: 167
    Handshake Protocol: Certificate Request
        Handshake Type: Certificate Request (13)
        Length: 159
        Certificate types count: 2
      Certificate types (2 types)
        Certificate type: RSA Sign (1)
        Certificate type: DSS Sign (2)
      Distinguished Names Length: 154
    Distinguished Names (154 bytes)
        Distinguished Name Length: 152
      Distinguished Name: ()
        RDNSequence: 1 item ()
          RelativeDistinguishedName
              Id: 2.5.4.3 (id-at-commonName)
            DirectoryString: printableString (1)
                printableString: Sharkfest Lab Root CA
        RDNSequence: 1 item ()
        RDNSequence: 1 item ()
        RDNSequence: 1 item ()
        RDNSequence: 1 item ()
        RDNSequence: 1 item ()
  Handshake Protocol: Server Hello Done
```

Certificate (C)

```
Secure Socket Layer
  SSLv3 Record Layer: Handshake Protocol: Multiple Handshake Messages
      Content Type: Handshake (22)
      Version: SSL 3.0 (0x0300)
      Length: 2579
    Handshake Protocol: Certificate
        Handshake Type: Certificate (11)
        Length: 2309
        Certificates Length: 2306
      Certificates (2306 bytes)
          Certificate Length: 1060
        Certificate ()
          Certificate Length: 1240
        Certificate ()
    Handshake Protocol: Client Key Exchange
    Handshake Protocol: Certificate Verify
  SSLv3 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  SSLv3 Record Layer: Handshake Protocol: Encrypted Handshake Message
```

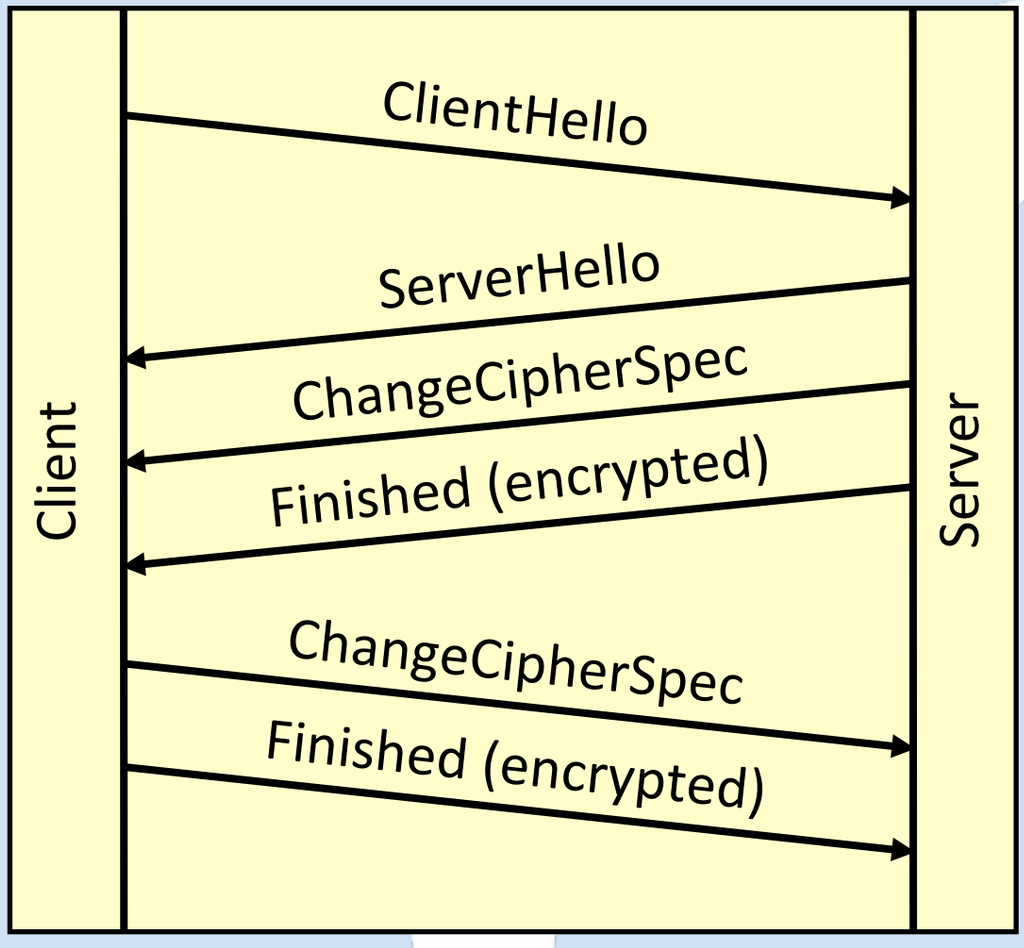
CertificateVerify

```
Secure Socket Layer
  SSLv3 Record Layer: Handshake Protocol: Multiple Handshake Messages
      Content Type: Handshake (22)
      Version: SSL 3.0 (0x0300)
      Length: 2579
    Handshake Protocol: Certificate
    Handshake Protocol: Client Key Exchange
    Handshake Protocol: Certificate Verify
        Handshake Type: Certificate Verify (15)
        Length: 130
  SSLv3 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  SSLv3 Record Layer: Handshake Protocol: Encrypted Handshake Message
```

Reusing SSL sessions

- Key negotiation "expensive"
- Cache session keys and re-use for new TCP sessions
- SSL session ID is used as Index
- Timeout on SSL session ID is an "absolute timeout" not an "idle timeout"
  - Old IE: 2 minutes, now 10 hours

Handshake of a Reused Session



No.	Time	Source	Destination	Protocol	Info
23	39.68772(	192.168.3.1	192.168.3.3	TCP	18774 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
24	39.688101	192.168.3.3	192.168.3.1	TCP	https > 18774 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=
25	39.688149	192.168.3.1	192.168.3.3	TCP	18774 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
26	39.688711	192.168.3.1	192.168.3.3	TLSv1	Client Hello
27	39.688983	192.168.3.3	192.168.3.1	TCP	https > 18774 [ACK] Seq=1 Ack=103 Win=5840 Len=0
28	39.694301	192.168.3.3	192.168.3.1	TLSv1	Server Hello, Change Cipher Spec, Encrypted Handshake Message
29	39.717354	192.168.3.1	192.168.3.3	TLSv1	Change Cipher Spec, Encrypted Handshake Message, Application Dat

SSL session reuse (new, reused and expired)

Filter: ssl.handshake — Expression... Clear Apply

No.	Time	Source	Destination	Protocol	ssl-id len	ssl-id	Info
4	0.011511	192.168.3.1	192.168.3.3	SSL	0		Client Hello
6	0.017431	192.168.3.3	192.168.3.1	TLSv1	32	DB00C2	Server Hello,
7	0.017782	192.168.3.3	192.168.3.1	TLSv1			Certificate
9	0.026711	192.168.3.1	192.168.3.3	TLSv1			Client Key Exchange, Change Cipher Spec, Encry
10	0.038327	192.168.3.3	192.168.3.1	TLSv1			Change Cipher Spec, Encrypted Handshake Messag
26	39.688711	192.168.3.1	192.168.3.3	SSL	32	DB00C2	Client Hello
28	39.						Spec, Encrypted Ha
29	39.						ed Handshake Messag
41	111						
43	111						
44	111						
46	111						Cipher Spec, Encry
47	111						ed Handshake Messag

```
#   Inter-Process Session Cache:
#   Configure the SSL Session Cache: First the mechanism
#   to use and second the expiring timeout (in seconds).
#SSLSessionCache         dbm:/var/run/apache2/ssl_scache
SSLSessionCache        shmcb:/var/run/apache2/ssl_scache(512000)
SSLSessionCacheTimeout  60
```

Destination Dest
Protocol Proto
ssl-id len Custom (ssl.handshake.session_id_length)
ssl-id Custom (ssl.handshake.session_id)
Info Information

Add
Remove

Properties
Format: Custom
Field name: ssl.handshake.session_id

Partial Handshake

No SSL session reuse

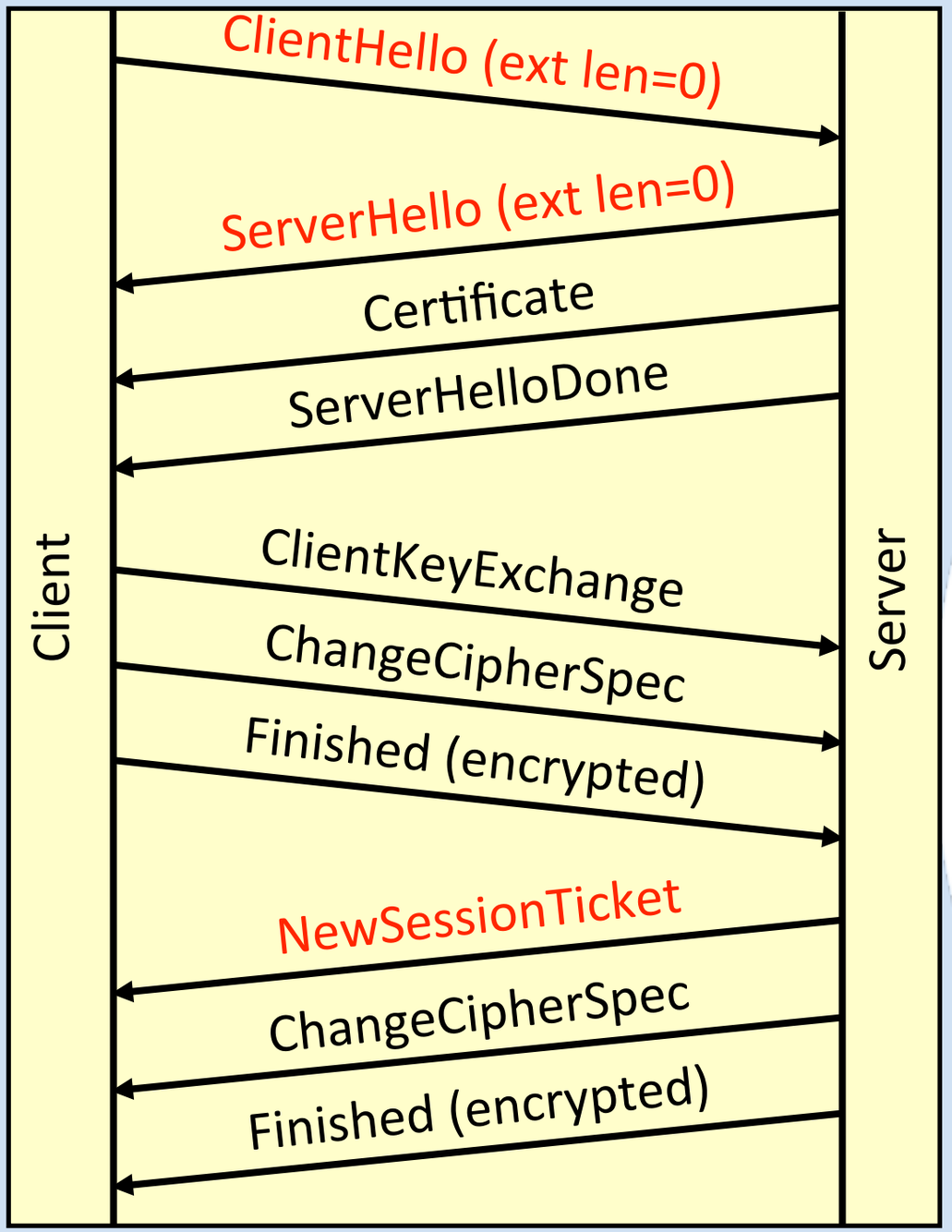
No.	Time	Source	Destination	Protocol	ssl-id len	ssl-id	Info
4	0.011833	192.168.3.1	192.168.3.3	TLSv1	32	5186BC	Client Hello
6	0.018800	192.168.3.3	192.168.3.1	TLSv1	0		Server Hello,
7	0.019128	192.168.3.3	192.168.3.1	TLSv1			Certificate
9	0.026392	192.168.3.1	192.168.3.3	TLSv1			Client Key Exchange, Change Cipher Spec, Encryp
10	0.037500	192.168.3.3	192.168.3.1	TLSv1			Change Cipher Spec, Encrypted Handshake Message

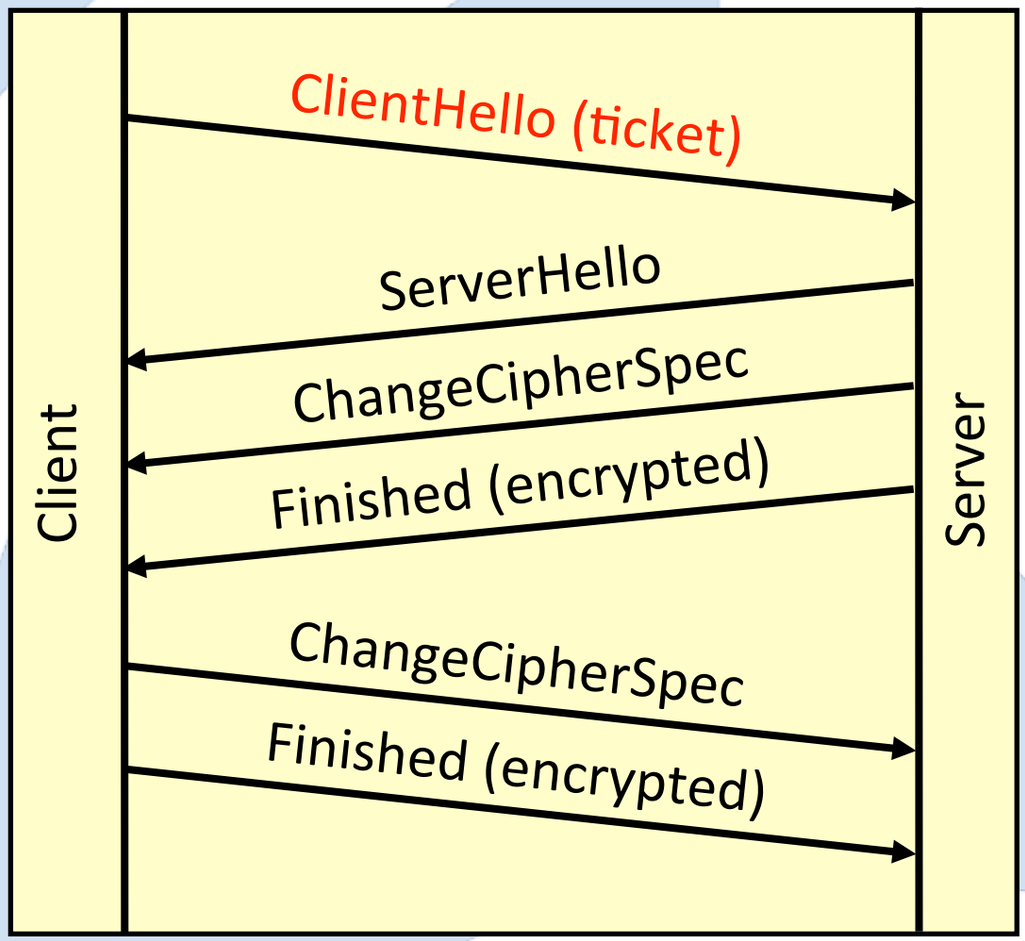
```
⊞ Frame 6 (1514 bytes on wire, 1514 bytes captured)
⊞ Ethernet II, Src: Vmware_5d:c5:66 (00:0c:29:5d:c5:66), Dst: Vmware_c0:00:01 (00:50:56:c0:00:01)
⊞ Internet Protocol, Src: 192.168.3.3 (192.168.3.3), Dst: 192.168.3.1 (192.168.3.1)
⊞ Transmission Control Protocol, Src Port: https (443), Dst Port: 17788 (17788), Seq: 1, Ack: 103, Len: 1460
⊟ Secure Socket Layer
  ⊟ TLSv1 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 42
    ⊟ Handshake Protocol: Server Hello
        Handshake Type: Server Hello (2)
        Length: 38
        Version: TLS 1.0 (0x0301)
      ⊞ Random
        Session ID Length: 0
        Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
        Compression Method: null (0)
```

TLS session Tickets (1)

- TLS session tickets (RFC 5077)
- Do not keep state on server, only on client
- TLS extension in ClientHello and ServerHello
- New SSL HandshakeType: **NewSessionTicket**

TLS session Tickets (2)





TLS session Tickets (3)

4	0.015145	192.168.1.22	74.125.132.19	TLSv1	164	Client Hello
6	0.032365	74.125.132.19	192.168.1.22	TLSv1	1484	Server Hello
7	0.032767	74.125.132.19	192.168.1.22	TLSv1	350	Certificate, Server Hello Done
9	0.033752	192.168.1.22	74.125.132.19	TLSv1	252	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
10	0.051951	74.125.132.19	192.168.1.22	TLSv1	292	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
22	2.363423	192.168.1.22	74.125.132.19	TLSv1	360	Client Hello
26	2.383264	74.125.132.19	192.168.1.22	TLSv1	199	Server Hello, Change Cipher Spec, Encrypted Handshake Message

```
▽ Handshake Protocol: Client Hello
    Handshake Type: Client Hello (1)
    Length: 89
    Version: TLS 1.0 (0x0301)
```

```
▽ Handshake Protocol: Server Hello
    Handshake Type: Server Hello (2)
    Length: 53
    Version: TLS 1.0 (0x0301)
  ▷ Random
    Session ID Length: 0
    Cipher Suite: TLS_RSA_WITH_RC4_128_SH
    Compression Method: null (0)
    Extensions Length: 13
  ▷ Extension: server_name
  ▷ Extension: renegotiation_info
  ▽ Extension: SessionTicket TLS
```

```
▽ TLSv1 Record Layer: Handshake Protoc
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 174
  ▽ Handshake Protocol: New Session Ticket
      Handshake Type: New Session Ticket (4)
      Length: 170
```

```
▽ Handshake Protocol: Client Hello
    Handshake Type: Client Hello (1)
    Length: 285
    Version: TLS 1.0 (0x0301)
  ▷ Random
    Session ID Length: 32
    Session ID: 73d2a649be4542fefe7b5cb6f4b15b5a48ae87f7597390b0...
    Cipher Suites Length: 20
  ▷ Cipher Suites (10 suites)
    Compression Methods Length: 1
  ▷ Compression Methods (1 method)
    Extensions Length: 192
  ▷ Extension: server_name
  ▽ Extension: SessionTicket TLS
      Type: SessionTicket TLS (0x0023)
      Length: 164
      Data (164 bytes)
```

Analyzing SSL Alerts

Without decryption:

14	12.494568	192.168.3.1	192.168.3.3	TLSv1	Application Data
15	12.495834	192.168.3.3	192.168.3.1	TLSv1	Application Data, Application Data
17	27.530927	192.168.3.3	192.168.3.1	TLSv1	Encrypted Alert
20	32.811207	192.168.3.1	192.168.3.3	TLSv1	Encrypted Alert

```
Secure Socket Layer
  TLSv1 Record Layer: Encrypted Alert
      Content Type: Alert (21)
      Version: TLS 1.0 (0x0301)
      Length: 32
      Alert Message: Encrypted Alert
```

With decryption:

14	12.494568	192.168.3.1	192.168.3.3	HTTP	GET / HTTP/1.1
15	12.495834	192.168.3.3	192.168.3.1	HTTP	HTTP/1.1 200 OK (text/html)
17	27.530927	192.168.3.3	192.168.3.1	TLSv1	Alert (Level: Warning, Description: Close Notify)
20	32.811207	192.168.3.1	192.168.3.3	TLSv1	Alert (Level: Warning, Description: Close Notify)

```
Secure Socket Layer
  TLSv1 Record Layer: Alert (Level: Warning, Description: Close Notify)
      Content Type: Alert (21)
      Version: TLS 1.0 (0x0301)
      Length: 32
    Alert Message
      Level: Warning (1)
      Description: Close Notify (0)
```

Analyzing SSL Application Data

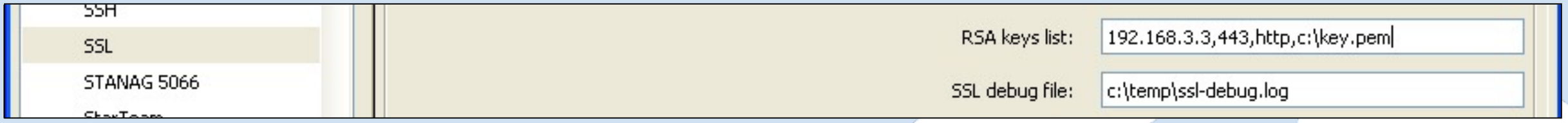
11	0.040173	192.168.3.1	192.168.3.3	TLSv1	491	Application Data
12	0.042446	192.168.3.3	192.168.3.1	TLSv1	496	Application Data, Application Data
14	12.494568	192.168.3.1	192.168.3.3	TLSv1	491	Application Data
15	12.495834	192.168.3.3	192.168.3.1	TLSv1	496	Application Data, Application Data
29	39.717354	192.168.3.1	192.168.3.3	TLSv1	550	Change Cipher Spec, Encrypted Handshake Message, Application Data
30	39.720262	192.168.3.3	192.168.3.1	TLSv1	496	Application Data, Application Data
48	111.230987	192.168.3.1	192.168.3.3	TLSv1	491	Application Data
49	111.233419	192.168.3.3	192.168.3.1	TLSv1	496	Application Data, Application Data

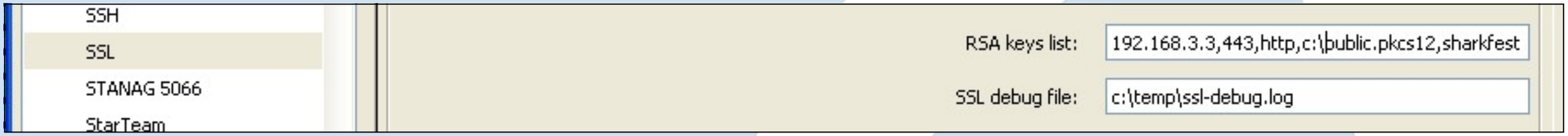
```
▽ Secure Sockets Layer
  ▽ TLSv1 Record Layer: Application Data Protocol: http
      Content Type: Application Data (23)
      Version: TLS 1.0 (0x0301)
      Length: 384
      Encrypted Application Data: 94c662e11c5c01813955dfc675754583ab4a70d65fddf8e9...
  ▽ TLSv1 Record Layer: Application Data Protocol: http
      Content Type: Application Data (23)
      Version: TLS 1.0 (0x0301)
      Length: 48
      Encrypted Application Data: 635e2a228ddc1aa5d7a2a89c809e6e693ec01f4cf5746fee...
```

Decrypting SSL traffic

- Provide **server** private key to Wireshark
- Only works when whole session (including **full handshake**) is in the tracefile
- Does **not** work with Ephemeral RSA or DH ciphers (ServerKeyExchange present)
- **Does** works with Client Authentication

Providing the server private key (1)





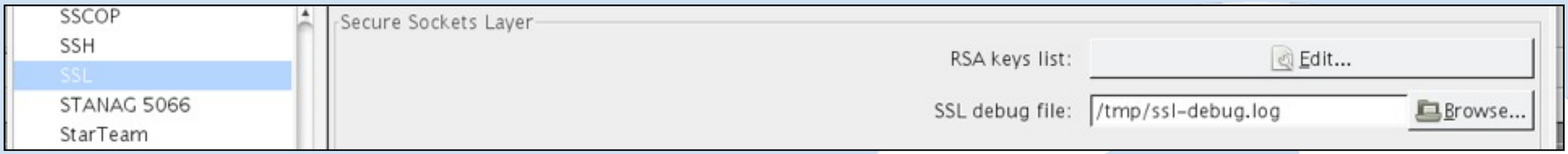
Wireshark preferences file:

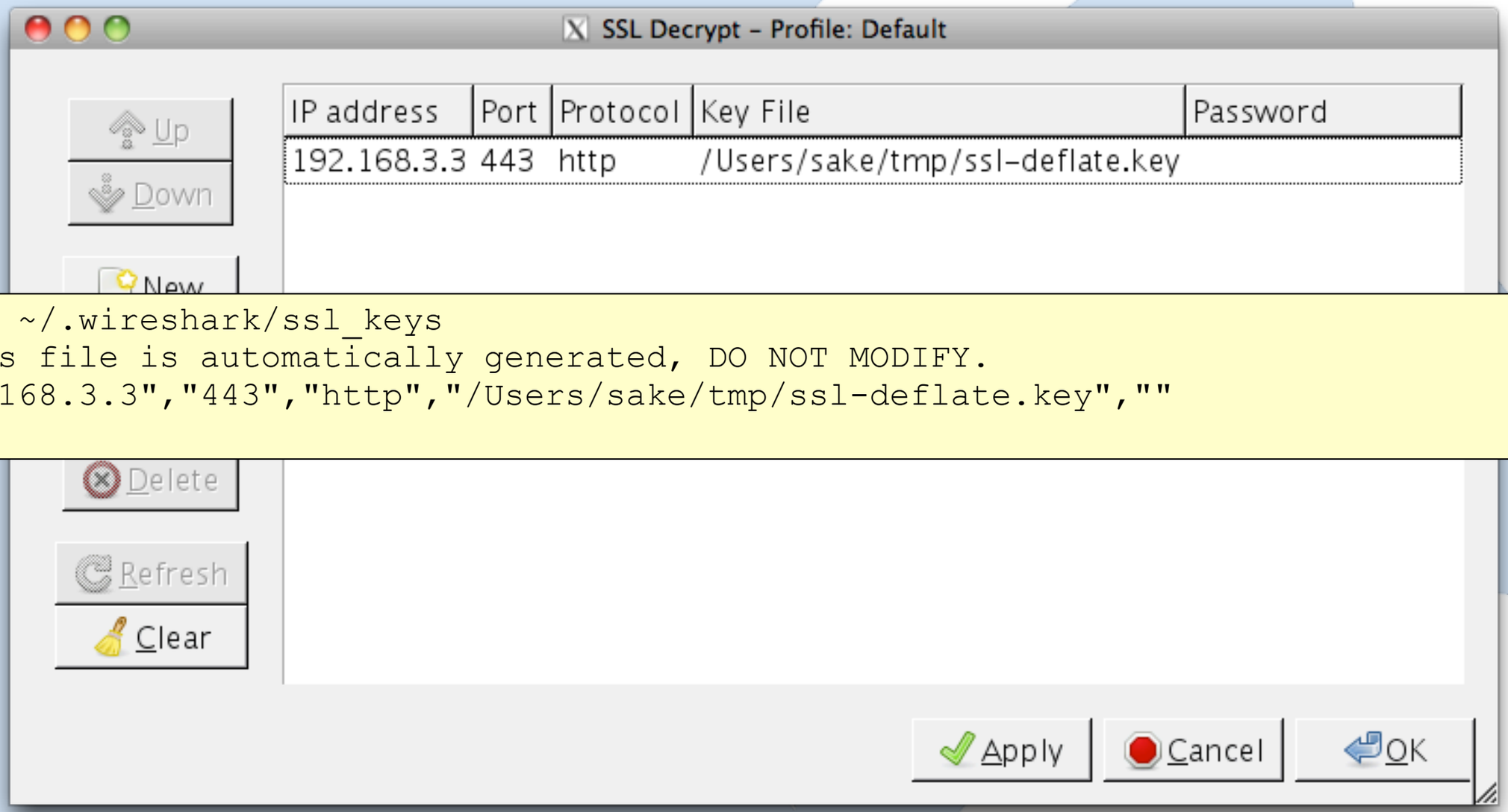
```
ssl.keys_list: 192.168.3.3,443,http,c:\key.pem
ssl.debug_file: c:\temp\ssl-debug.log
```

When using Tshark:

```
tshark -r file.cap -o ssl.keys_list:192.168.3.3,443,http,"c:\key.pem" \
                   -o ssl.debug_file:"c:\ssl-debug.log" -V -R http
```

Providing the server private key (2)






```
$ cat ~/.wireshark/ssl_keys
# This file is automatically generated, DO NOT MODIFY.
"192.168.3.3","443","http","/Users/sake/tmp/ssl-deflate.key",""
$
```

Providing the server private key (3)

SSL debug log:

```
ssl_init keys string:
192.168.3.3,443,http,c:\temp\public.sharkfest.local.key
ssl_init found host entry 192.168.3.3,443,http,c:\temp\public.sharkfest.local.key
ssl_init addr '192.168.3.3' port '443' filename 'c:\temp\public.sharkfest.local.key' password(only for p12 file) '(null)'
ssl_load_key: can't import pem data
```

PEM keyfile *without* passphrase:

```
-----BEGIN RSA PRIVATE KEY-----
MIICXgIBAAKBgQDrHdbb+yGE6m6EZ03bXURpZCjch2H6g97ZAkJVGrjLZFfettBA
EYa8vYYxWsf8KBpEZeksSCsDA9MnU2H6QDjzqdOnaSWfeXMAr4OsCOpauStpreq7
q1hk8iOqy+f4KijRrhWplh1QW1A8gtSIg137pyUhW+WsfwxKwmzjGIC1SwIDAQAB
AoGBAMneA9U6KIxjb+JUg/99c7h9W6wEvTYHNTXjf6psWA+hpuQ82E65/ZJdszL6
...
b6QKMh16r5wd6smQ+CmhOEnqqyT5AIwwl2RIr9GbfIpTbtbRQw/EcQOCx9wFiEfo
tGSsEFi72rHK+DpJqRI9AkEA72gdyXRgPfGOS3rfQ3DBcImBQvDSCBa4cuU1XJ1/
MO93a8v9Vj87/yDm4xsBDsoz2PyBepawHVlIvZ6jDD0aXw==
-----END RSA PRIVATE KEY-----
```

PEM keyfile *with* passphrase:

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,F6C218D4FA3C8B66

FR2cnmkkFHH45Dcsty1qDiIUy/uXn+9m/xeQMVRxtiSAmBmnUDUFIFCDDiDc9yif
ERok2jPr2BzAazl5RBxS2TY/+7x0/dHD11sF3LnJUoNruo77TERxqgzOI0W1VDRA
...
ygw5JslxgiN18F36E/cEP5rKvVYvfEPMa6IsiRhfZk1jLAuZihVWc7JodDf+6RKV
yBXrK/bDtdEih+bOnYu+ZDvjAzVz9GhggCW4QHNboDpTxrrYPkj5Nw==
-----END RSA PRIVATE KEY-----
```

Converting keys

Removing passphrase:

```
root@mgmt# openssl rsa -in encrypted.key -out cleartext.key
Enter pass phrase for encrypted.key: <passphrase>
writing RSA key
root@mgmt#
```

Converting from DER to PEM (and removing passphrase):

```
root@mgmt# openssl rsa -inform DER -in der.key -out pem.key
Enter pass phrase for encrypted.key: <passphrase>
writing RSA key
root@mgmt#
```

Converting from PEM to PKCS12 (and adding passphrase):

```
root@mgmt# openssl pkcs12 -in pem.cert -inkey pem.key -export -out cert.pkcs12
Enter Export Password: <new-passphrase>
Verifying - Enter Export Password: <new-passphrase>
root@mgmt#
```

Decryption in Action

No.	Time	Source	Destination	Protocol	Info
10	0.038327	192.168.3.3	192.168.3.1	TLSv1	Change Cipher Spec, Finished
11	0.040173	192.168.3.1	192.168.3.3	HTTP	GET / HTTP/1.1
12	0.042446	192.168.3.3	192.168.3.1	HTTP	HTTP/1.1 200 OK (text/html)
13	0.238504	192.168.3.1	192.168.3.3	TCP	18736 > https [ACK] Seq=706 Ack=2927 Win=127498 Len=0

```
⊞ Frame 11 (491 bytes on wire, 491 bytes captured)
⊞ Ethernet II, Src: Vmware_c0:00:01 (00:50:56:c0:00:01), Dst: Vmware_5d:c5:66 (00:0c:29:5d:c5:66)
⊞ Internet Protocol, Src: 192.168.3.1 (192.168.3.1), Dst: 192.168.3.3 (192.168.3.3)
⊞ Transmission Control Protocol, Src Port: 18736 (18736), Dst Port: https (443), Seq: 269, Ack: 2485, Len: 43
⊟ Secure Socket Layer
  ⊟ TLSv1 Record Layer: Application Data Protocol: http
      Content Type: Application Data (23)
      Version: TLS 1.0 (0x0301)
      Length: 432
      Encrypted Application Data: C0D1C49A5E8119FC1B21EF547592476DF61AA48A11C44522...
⊟ Hypertext Transfer Protocol
  ⊞ GET / HTTP/1.1\r\n
    Host: 192.168.3.3\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.8) Gecko/2009032609 Firefox/3.0.8\r\
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
    Accept-Language: en-us,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
    Keep-Alive: 300\r\n
    Connection: keep-alive\r\n
    Pragma: no-cache\r\n
    Cache-Control: no-cache\r\n
    \r\n
```

Decrypting IMAPS

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.46	192.168.1.20	TCP	22446 > imaps [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.001820	192.168.1.20	192.168.1.46	TCP	imaps > 22446 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=7
3	0.001857	192.168.1.46	192.168.1.20	TCP	22446 > imaps [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.010231	192.168.1.46	192.168.1.20	SSL	Client Hello
5	0.011625	192.168.1.20	192.168.1.46	TCP	imaps > 22446 [ACK] Seq=1 Ack=103 Win=5888 Len=0
6	0.012351	192.168.1.20	192.168.1.46	TLSv1	Server Hello, Certificate, Server Hello Done
7	0.013831	192.168.1.46	192.168.1.20	TLSv1	Client Key Exchange, Change Cipher Spec, Finished
8	0.019822	192.168.1.20	192.168.1.46	TLSv1	Change Cipher Spec, Finished
9	0.168748	192.168.1.46	192.168.1.20	TCP	22446 > imaps [ACK] Seq=285 Ack=978 Win=127022 Len=0
10	0.170301	192.168.1.20	192.168.1.46	IMAP	Response: * OK Dovecot ready.
11	0.172574	192.168.1.46	192.168.1.20	IMAP	Request: g7fg CAPABILITY

```
Frame 10 (96 bytes on wire, 96 bytes captured)
Ethernet II, Src: JuniperN_bb:d1:3b (0
Internet Protocol, Src: 192.168.1.20 (192.168.1.20), Dst: 192.168.1.46 (192.168.1.46)
Transmission Control Protocol, Src Port: imaps (993), Dst Port: 22446 (22446), Seq: 978, Ack: 285, Len: 42
Secure Socket Layer
  TLSv1 Record Layer: Application Data Protocol: imap
      Content Type: Application Data (23)
      Version: TLS 1.0 (0x0301)
      Length: 37
      Encrypted Application Data: F8260B9E9D0597A3CE35E176BA3EDB28D588E004F6B57F74...
Internet Message Access Protocol
  * OK Dovecot ready.\r\n
      Response Tag: *
      Response: OK Dovecot ready.
```

ssl.keys_list: 192.168.1.20,993,imap,C:\key.pem

Decrypting "STARTTLS" (1)

Filter: smtp || ssl Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
4	0.021653	192.168.1.20	192.168.1.46	SMTP	S: 220 brutus.netcc.local ESMTP Postfix (Ubuntu)
5	0.023320	192.168.1.46	192.168.1.20	SMTP	C: EHLO HTRQ93J
7	0.025077	192.168.1.20	192.168.1.46	SMTP	S: 250-brutus.netcc.local | 250-PIPELINING | 250-SIZE 10240000 |
8	0.025868	192.168.1.46	192.168.1.20	SMTP	C: STARTTLS
9	0.027373	192.168.1.20	192.168.1.46	SMTP	S: 220 2.0.0 Ready to start TLS
11	0.262273	192.168.1.46	192.168.1.20	TLSv1	Client Hello
12	0.264832	192.168.1.20	192.168.1.46	TLSv1	Server Hello, Certificate, Server Hello Done
13	0.266373	192.168.1.46	192.168.1.20	TLSv1	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Mess.
14	0.281296	192.168.1.20	192.168.1.46	TLSv1	Change Cipher Spec, Encrypted Handshake Message

```
⊞ Frame 13 (236 bytes on wire, 236 bytes captured)
⊞ Ethernet II, Src: IntelCor_61:3a:ad (00:1c:bf:61:3a:ad), Dst: Junipe
⊞ Internet Protocol, Src: 192.168.1.46 (192.168.1.46), Dst: 192.168.1.20 (192.168.1.20)
⊞ Transmission Control Protocol, Src Port: 38477 (38477), Dst Port: smtp (25), Seq: 95, Ack: 1153, Len: 182
⊟ Secure Socket Layer
  ⊞ TLSv1 Record Layer: Handshake Protocol: Client Key Exchange
  ⊞ TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  ⊟ TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 32
      Handshake Protocol: Encrypted Handshake Message
```

ssl.keys_list:

Decrypting "STARTTLS" (2)

Filter: smtp || ssl Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
4	0.021653	192.168.1.20	192.168.1.46	SSL	Continuation Data
5	0.023320	192.168.1.46	192.168.1.20	SSL	Continuation Data
7	0.025077	192.168.1.20	192.168.1.46	SSL	Continuation Data
8	0.025868	192.168.1.46	192.168.1.20	SSL	Continuation Data
9	0.027373	192.168.1.20	192.168.1.46	SSL	Continuation Data
11	0.262273	192.168.1.46	192.168.1.20	SSL	Client Hello
12	0.264832	192.168.1.20	192.168.1.46	TLSv1	Server Hello, Certificate, Server Hello Done
13	0.266373	192.168.1.46	192.168.1.20	TLSv1	Client Key Exchange, Change Cipher Spec, Finished
14	0.281296	192.168.1.20	192.168.1.46	TLSv1	Change Cipher Spec, Finished

```
⊞ Frame 13 (236 bytes on wire, 236 bytes captured)
⊞ Ethernet II, Src: IntelCor_61:3a:ad (00
⊞ Internet Protocol, Src: 192.168.1.46 (192.168.1.46), Dst: 192.168.1.20 (192.168.1.20)
⊞ Transmission Control Protocol, Src Port: 38477 (38477), Dst Port: smtp (25), Seq: 95, Ack: 1153, Len: 182
⊟ Secure Socket Layer
  ⊞ TLSv1 Record Layer: Handshake Protocol: Client Key Exchange
  ⊞ TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  ⊟ TLSv1 Record Layer: Handshake Protocol: Finished
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 32
    ⊟ Handshake Protocol: Finished
        Handshake Type: Finished (20)
        Length: 12
        Verify Data
```

ssl.keys_list: 192.168.1.20,25,smtp,C:\key.pem

Decrypting "STARTTLS" (3)

Filter: smtp || ssl Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
4	0.021653	192.168.1.20	192.168.1.46	SMTP	S: 220 brutus.netcc.local ESMTP Postfix (Ubuntu)
5	0.023320	192.168.1.46	192.168.1.20	SMTP	C: EHLO HTRQ93J
7	0.025077	192.168.1.20	192.168.1.46	SMTP	S: 250-brutus.netcc.local | 250-PIPELINING | 250-SIZE 10240000 |
8	0.025868	192.168.1.46	192.168.1.20	SMTP	C: STARTTLS
9	0.027373	192.168.1.20	192.168.1.46	SMTP	S: 220 2.0.0 Ready to start TLS
11	0.262273	192.168.1.46	192.168.1.20	TLSv1	Client Hello
12	0.264832	192.168.1.20	192.168.1.46	TLSv1	Server Hello, Certificate, Server Hello Done
13	0.266373	192.168.1.46	192.168.1.20	TLSv1	Client Key Exchange, Change Cipher Spec, Finished
14	0.281296	192.168.1.20	192.168.1.46	TLSv1	Change Cipher Spec, Encrypted Handshake Message

```
⊞ Frame 13 (236 bytes on wire, 236 bytes captured)
⊞ Ethernet II, Src: IntelCor
⊞ Internet Protocol, Src: 19
⊞ Transmission Control Protocol, Src Port: 38477 (38477), Dst Port: smtp (25), Seq: 95, Ack: 1153, Len: 182
⊟ Secure Socket Layer
  ⊞ TLSv1 Record Layer: Handshake Protocol: Client Key Exchange
  ⊞ TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  ⊟ TLSv1 Record Layer: Handshake Protocol: Finished
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 32
    ⊟ Handshake Protocol: Finished
        Handshake Type: Finished (20)
        Length: 12
        Verify Data
```

ssl.keys_list: 192.168.1.20,start_tls,smtp,C:\key.pem

Decrypt-problem I (1)

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.3	TCP	18774 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000375	192.168.3.3	192.168.3.1	TCP	https > 18774 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=4
3	0.000423	192.168.3.1	192.168.3.3	TCP	18774 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.000985	192.168.3.1	192.168.3.3	SSL	Client Hello
5	0.001257	192.168.3.3	192.168.3.1	TCP	https > 18774 [ACK] Seq=1 Ack=103 Win=5840 Len=0
6	0.006575	192.168.3.3	192.168.3.1	TLSv1	Server Hello, Change Cipher Spec, Encrypted Handshake Message
7	0.029628	192.168.3.1	192.168.3.3	TLSv1	Change Cipher Spec, Encrypted Handshake Message, Application Data
8	0.032536	192.168.3.3	192.168.3.1	TLSv1	Application Data, Application Data

```
⊞ Frame 7 (550 bytes on wire, 550 bytes captured)
⊞ Ethernet II, Src: Vmware_c0:00:01 (00:50:56:c0:00:01), Dst: Vmware_5d:c5:66 (00:0c:29:5d:c5:66)
⊞ Internet Protocol, Src: 192.168.3.1 (192.168.3.1), Dst: 192.168.3.3 (192.168.3.3)
⊞ Transmission Control Protocol, Src Port: 18774 (18774), Dst Port: https (443), Seq: 103, Ack: 139, Len: 496
⊟ Secure Socket Layer
  ⊞ TLSv1 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
  ⊟ TLSv1 Record Layer: Handshake Protocol: Encrypted Handshake Message
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 48
      Handshake Protocol: Encrypted Handshake Message
  ⊞ TLSv1 Record Layer: Application Data Protocol: http
```

```
ssl_init keys string:
192.168.3.3,443,http,c:\temp\public.sharkfest.local.key
ssl_init found host entry 192.168.3.3,443,http,c:\temp\public.sharkfest.local.key
ssl_init addr '192.168.3.3' port '443' filename 'c:\temp\public.sharkfest.local.key' password(only for p12
file) '(null)'
Private key imported: KeyID B8:2B:EA:B8:F8:BD:62:50:E3:0C:2D:3D:06:09:91:64:...
ssl_init private key file c:\temp\public.sharkfest.local.key successfully loaded
association_add TCP port 443 protocol http handle 04086228
```

Decrypt-problem I (2)

Checking ssl debug log:

```
[...]
dissect_ssl enter frame #7 (first time)
  conversation = 07411870, ssl_session = 07411BC8
  record: offset = 0, reported_length_remaining = 496
dissect_ssl3_record: content_type 20
dissect_ssl3_change_cipher_spec
association_find: TCP port 18774 found 00000000
packet_from_server: is from server - FALSE
ssl_change_cipher CLIENT
  record: offset = 6, reported_length_remaining = 490
dissect_ssl3_record: content_type 22
decrypt_ssl3
association_
packet_from_
decrypt_ssl3
decrypt_ssl3
dissect_ssl3
  record: offset = 59, reported_length_remaining = 437
dissect_ssl3_record: content_type 23
decrypt_ssl3_record: app_data len 432 ssl, state 0x17
association_find: TCP port 18774 found 00000000
packet_from_server: is from server - FALSE
decrypt_ssl3_record: using client decoder
decrypt_ssl3_record: no decoder available
association_find: TCP port 18774 found 00000000
association_find: TCP port 443 found 047AF518
[...]
```

Make sure that the whole SSL session (which can be made out of multiple TCP streams) is in the tracefile. Starting with the handshake and up to the current frame.

Decrypt-problem II (1)

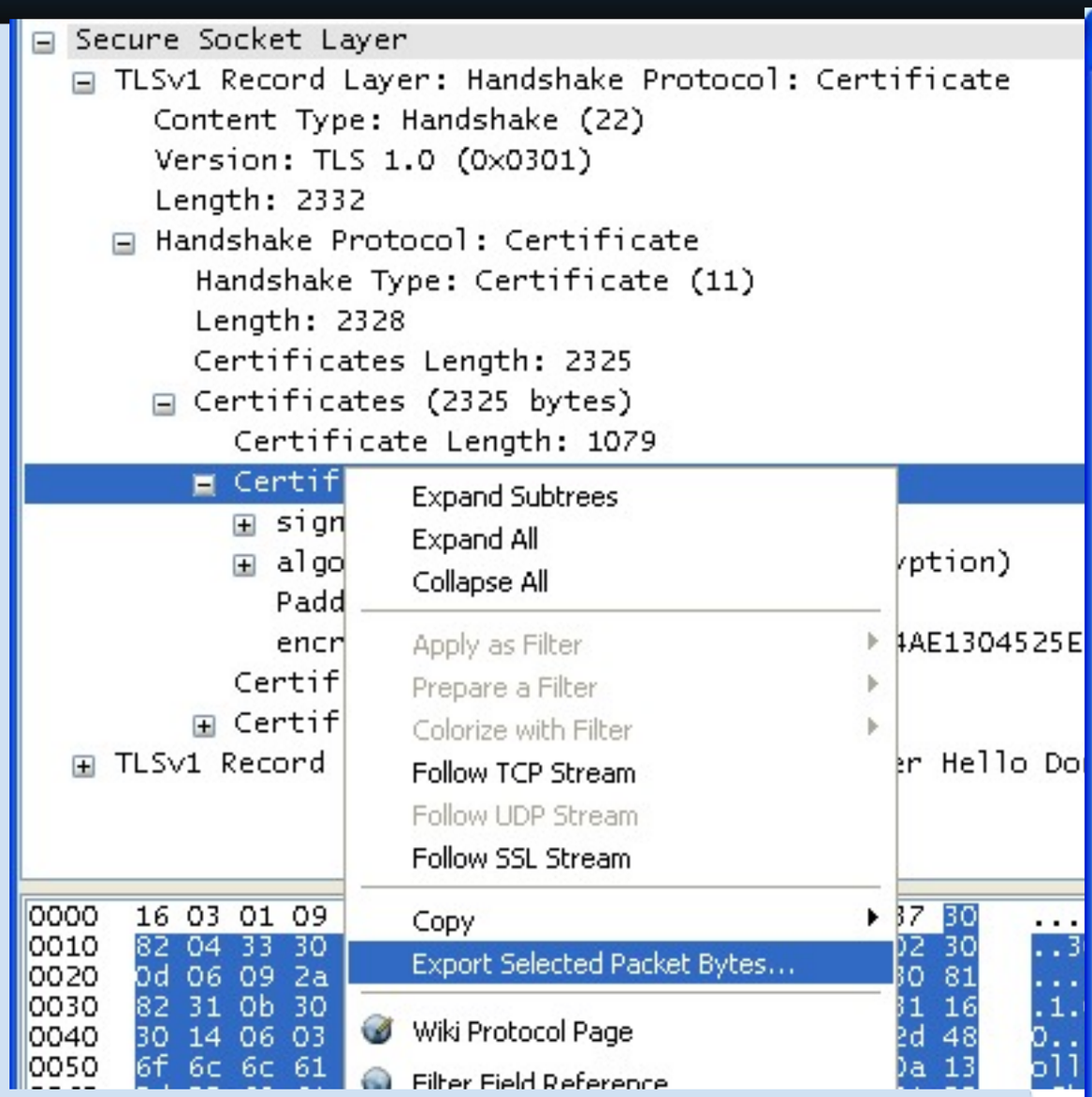
Checking ssl debug log:

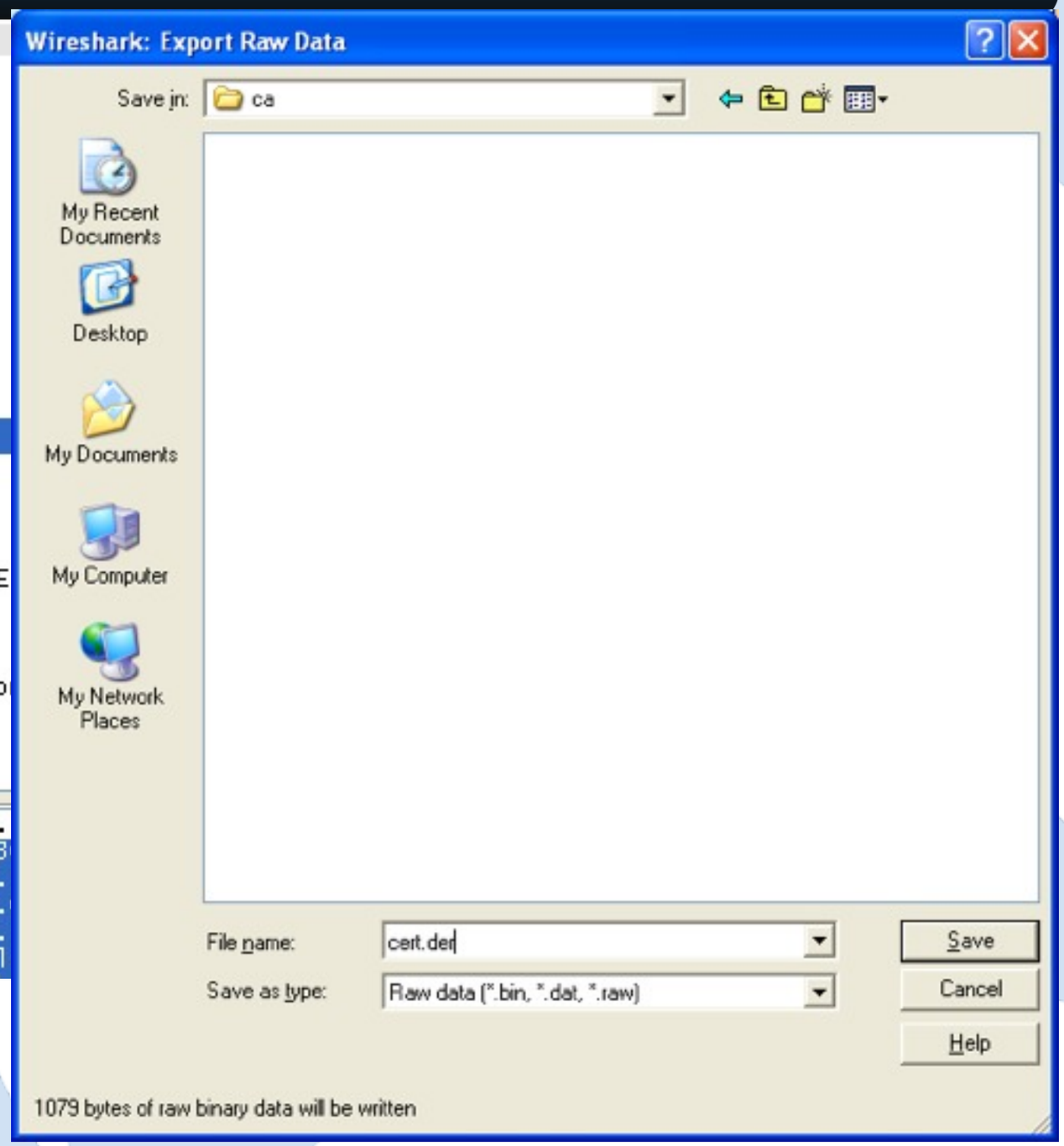
```
ssl_association_remove removing TCP 443 - http handle 04086F30
ssl_init keys string:
192.168.3.3,443,http,c:\temp\public.sharkfest.local.key
ssl_init found host entry 192.168.3.3,443,http,c:\temp\public.sharkfest.local.key
ssl_init addr '192.168.3.3' port '443' filename 'c:\temp\public.sharkfest.local.key' password(only for p12
file) '(null)'
Private key imported: KeyID FA:56:73:A4:38:9C:A1:4F:28:23:88:76:83:42:13:86:...
ssl_init private key file c:\temp\public.sharkfest.local.key successfully loaded
association_add TCP port 443 protocol http handle 04086F30

[...]

ssl_decrypt_pre_master_secret:RSA_private_decrypt
pcry_private_decrypt: stripping 0 bytes, decr_len zd
decrypted_unstrip_pre_master[128]:
6a f7 2a 4b 45 17 72 47 c2 11 d1 dd ad dc af b6
04 76 cb 3c 32 1c d1 01 57 4a 83 79 af d9 40 af
aa a8 71 1f bd 6f 70 d5 cc 49 e6 be 44 42 07 7c
45 b7 5b 5b 52 de 3e 58 d3 42 8d 5f bc 99 3e 13
f5 7d 27 a1 3e 7f b2 3f 8b 9d e5 fb 60 ec 40 26
87 8f 24 41 fb d4 ec f7 0e ea 04 46 c2 d7 5f 7b
4a d2 40 47 07 7b 0d 63 d8 d6 0f e6 9e 98 92 02
58 13 51 72 1b 85 69 04 52 42 74 12 40 e2 a5 bb
ssl_decrypt_pre_master_secret wrong pre_master_secret length (128, expected 48)
dissect_ssl3_handshake can't decrypt pre master secret
```

Decrypt-problem II (2)





Decrypt-problem II (3)

In wireshark preferences:

```
ssl.keys_list: 192.168.3.3,443,http,c:\temp\public.sharkfest.local.key
```

Checking whether certificate and key match:

```
$ openssl x509 -in cert.der -inform DER -noout -text | grep "Subject:"
        Subject: C=NL, ST=Noord-Holland, O=Sharkfest Lab, CN=public.sharkfest.local/
emailAddress=co@sharkfest.local
$
$ openssl
a29682af8
$
$ openssl
ce71158d3851a885314c264863142389
$
$ openssl rsa -noout -modulus -in private.sharkfest.local.key | openssl md5
a29682af822b4cd064d39d4ccd1e0e6c
$
```

Make sure that the private key matches the (server) certificate that is used in the tracefile.

Decryption Without The Private Key

- Using the unencrypted (pre-)Master-Secret
- First use: Debug version of Firefox/Chrome
  (see: https://bugs.wireshark.org/bugzilla/show_bug.cgi?id=4349)
- Second use: openssl s_client
- SSL preferences: (Pre)-Master-Secret log filename

```
RSA db00c2aad79cfda1 92cdc769c670ba6f48cfe756992ad435401a26d023590...f360c108df167ca6b6f443f4d2b118de0ccadb8

RSA Session-ID:fbcf322128ed0a0...61523189639f5ba189a Master-Key:bda6ea472f6c39a9fcfd5dc...228eb85744c9bf7cf2
```

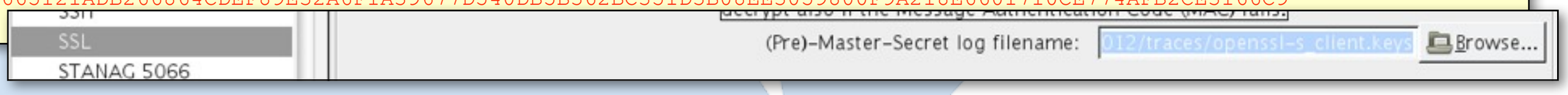
openssl s_client

```
$ openssl s_client -cipher AES256-SHA -no_ticket -connect imap.syn-bit.nl:993 | tee openssl-s_client.txt
depth=1 C = GB, ST = Greater Manchester, L = Salford, O = COMODO CA Limited, CN = COMODO High-Assurance Secure
Server CA
verify error:num=20:unable to get local issuer certificate
verify return:0
CONNEC
---
[...]
SSL-Se
    Pr
    Ci
    Se
    Se
    Ma
    Ke
    PS
    PS
    SF
    Co
    St
    Ti
    Ve
---
* OK I
HEL
*
QU
DO
```

```
    15 0.180186   46.30.211.94    192.168.1.22    IMAP    119   Response: * OK IMAP4 ready
    17 2.631302   192.168.1.22    46.30.211.94    IMAP    140   Request: HELP
    18 2.669607   46.30.211.94    192.168.1.22    IMAP    135   Response: * BAD invalid command

▷ Frame 15: 119 bytes on wire (952 bits), 119 bytes captured (952 bits)
▷ Ethernet II, Src: JuniperN_bb:d1:32 (00:12:1e:bb:d1:32), Dst: Apple_d8:87:48 (f8:1e:df:d8:87:48)
▷ Internet Protocol Version 4, Src: 46.30.211.94 (46.30.211.94), Dst: 192.168.1.22 (192.168.1.22)
▷ Transmission Control Protocol, Src Port: imaps (993), Dst Port: 64400 (64400), Seq: 2965, Ack: 425, Len: 53
▽ Secure Sockets Layer
  ▽ TLSv1 Record Layer: Application Data Protocol: imap
      Content Type: Application Data (23)
      Version: TLS 1.0 (0x0301)
      Length: 48
      Encrypted Application Data: 74a980b1955b4f74c1be949df97da0f4a25f27704ed7b66a...
▽ Internet Message Access Protocol
  ▷ * OK IMAP4 ready\r\n
```

```
$ awk '$1~"Session-ID:" {printf("RSA %s%s ",$1,$2)} $1~"Master-Key:" {printf("%s%s\n",$1,$2)}' openssl-
s_client.txt > openssl-s_client.keys
$ cat openssl-s_client.keys
RSA Session-ID:5EF3E7EDCC46993E51935914ACC1CBE6723259121248F958BC223D54FA84CFA0 Master-Key:
0665121ADB266864CDEF89E32A6F1A39677D540DB5B362BC351D3B08EE3059800F9A218E6601710CE774AFB2CE3166C9
$
```



Export SSL Session Keys

- Export:
  - File -> Export -> SSL Session Keys (1.6.x)
  - File -> Export SSL Session Keys (1.8.x)
- Import:
  - SSL preferences: (Pre)-Master-Secret log filename

Provide SSL decryption in Wireshark to a 3rd party without having to share the private key!

Agenda

- Cryptology overview
- The SSL protocol
- Analyzing SSL with Wireshark
- Analyzing SSL with Tshark
- Common SSL connection problems
- Further reading
- Questions & Discussion

Analyzing SSL with Tshark (1)

- -V to show whole tree (and decrypted application data)
- tshark -G fields | fgrep "ssl."

  tshark -R ssl.alert_message
- tshark -G currentprefs | egrep "^#?ssl."

  tshark -o ssl.keys_list:<ip>,<port>,<proto>,<keyfile> \

  -o ssl.debug_file:<log-file>

Analyzing SSL with Tshark (2)

```
tshark -r file.cap -o ssl.keys_list:192.168.3.3,443,http,"c:\key.pem" \
                   -o ssl.debug_file:"c:\ssl-debug.log" -V -R http
```

```
$ tshark -o ssl.keys_list:192.168.3.3,443,http,"c:\tmp.key" \
     -r session-reuse.cap -R ssl.alert_message
 17  27.530927  192.168.3.3 -> 192.168.3.1  TLSv1 Alert (Level: Warning, Description: Close Notify)
 20  32.811207  192.168.3.1 -> 192.168.3.3  TLSv1 Alert (Level: Warning, Description: Close Notify)
 32  54.756406  192.168.3.3 -> 192.168.3.1  TLSv1 Alert (Level: Warning, Description: Close Notify)
 35  62.809496  192.168.3.1 -> 192.168.3.3  TLSv1 Alert (Level: Warning, Description: Close Notify)
 51 126.272833  192.168.3.3 -> 192.168.3.1  TLSv1 Alert (Level: Warning, Description: Close Notify)
 54 137.815000  192.168.3.1 -> 192.168.3.3  TLSv1 Alert (Level: Warning, Description: Close Notify)
$
```

Agenda

- Cryptology overview
- The SSL protocol
- Analyzing SSL with Wireshark
- Analyzing SSL with Tshark
- Common SSL connection problems
- Further reading
- Questions & Discussion

Common SSL problems I (1)

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.3	TCP	24269 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000667	192.168.3.3	192.168.3.1	TCP	https > 24269 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=4
3	0.000716	192.168.3.1	192.168.3.3	TCP	24269 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.020817	192.168.3.1	192.168.3.3	SSLv3	Client Hello
5	0.021173	192.168.3.3	192.168.3.1	TCP	https > 24269 [ACK] Seq=1 Ack=65 Win=5840 Len=0
6	0.024816	192.168.3.3	192.168.3.1	SSLv3	Alert (Level: Fatal, Description: Handshake Failure)
7	0.025488	192.168.3.3	192.168.3.1	TCP	https > 24269 [FIN, ACK] Seq=8 Ack=65 Win=5840 Len=0
8	0.025536	192.168.3.1	192.168.3.3	TCP	24269 > https [ACK] Seq=65 Ack=9 Win=127992 Len=0
9	0.031750	192.168.3.1	192.168.3.3	TCP	24269 > https [FIN, ACK] Seq=65 Ack=9 Win=127992 Len=0
10	0.032001	192.168.3.3	192.168.3.1	TCP	https > 24269 [ACK] Seq=9 Ack=66 Win=5840 Len=0

Common SSL problems I (2)

In apache2:

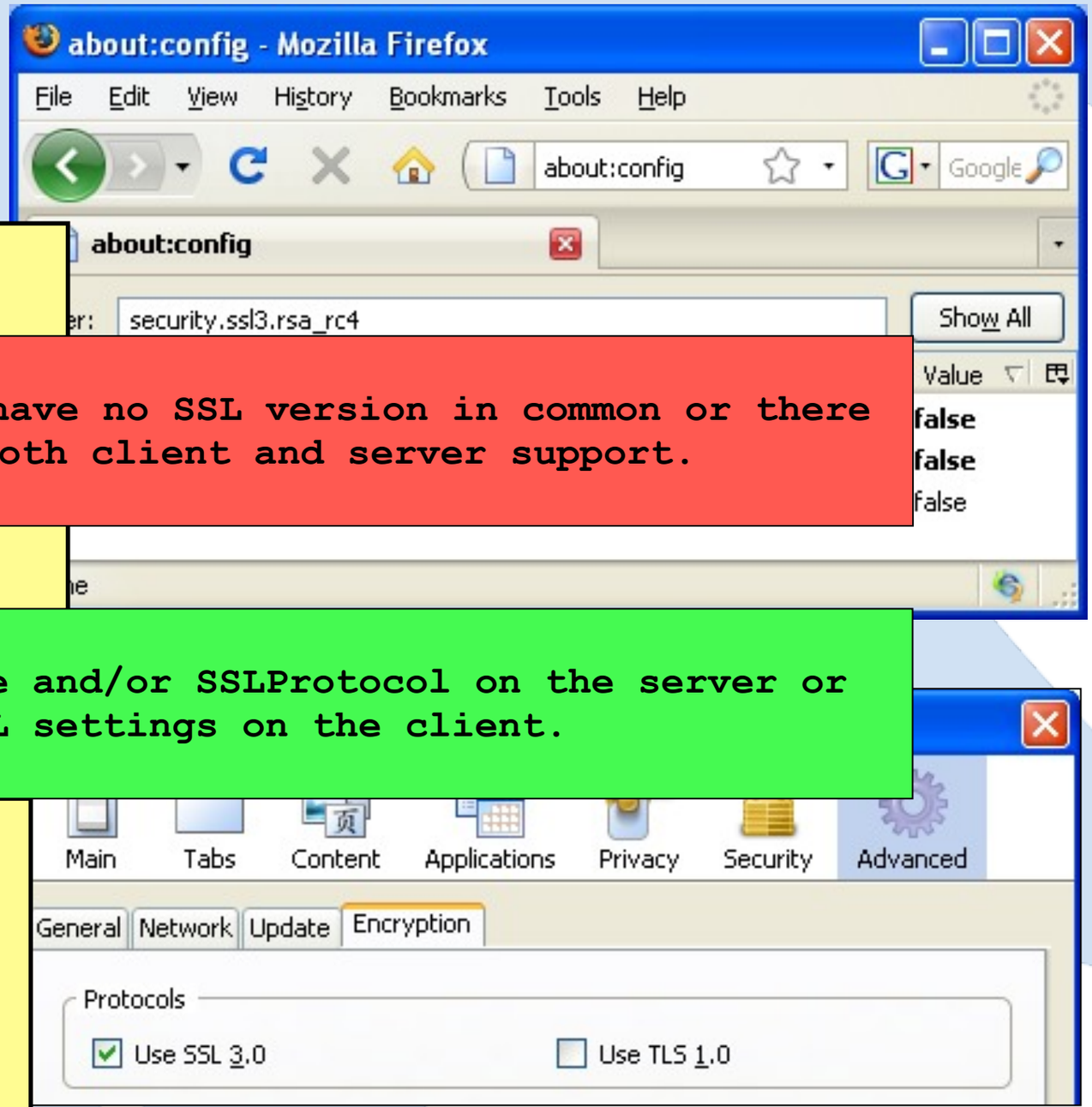
SSLCipherSuite RC4+RSA

The client and the server have no SSL version in common or there is no cipher that both client and server support.

Reconfigure SSLCipherSuite and/or SSLProtocol on the server or adjust the SSL settings on the client.

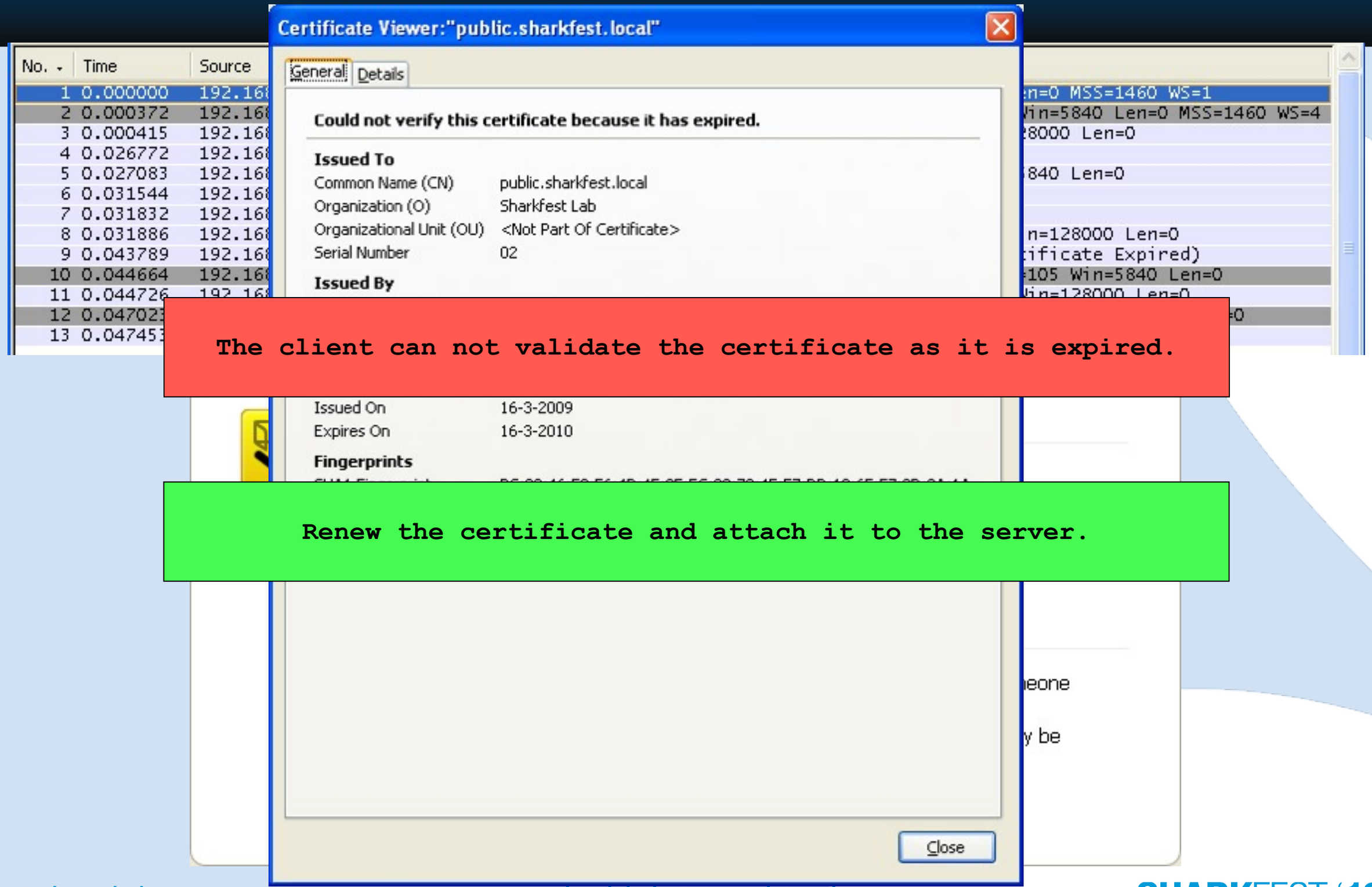
In apache2:

SSLProtocol TLSv1

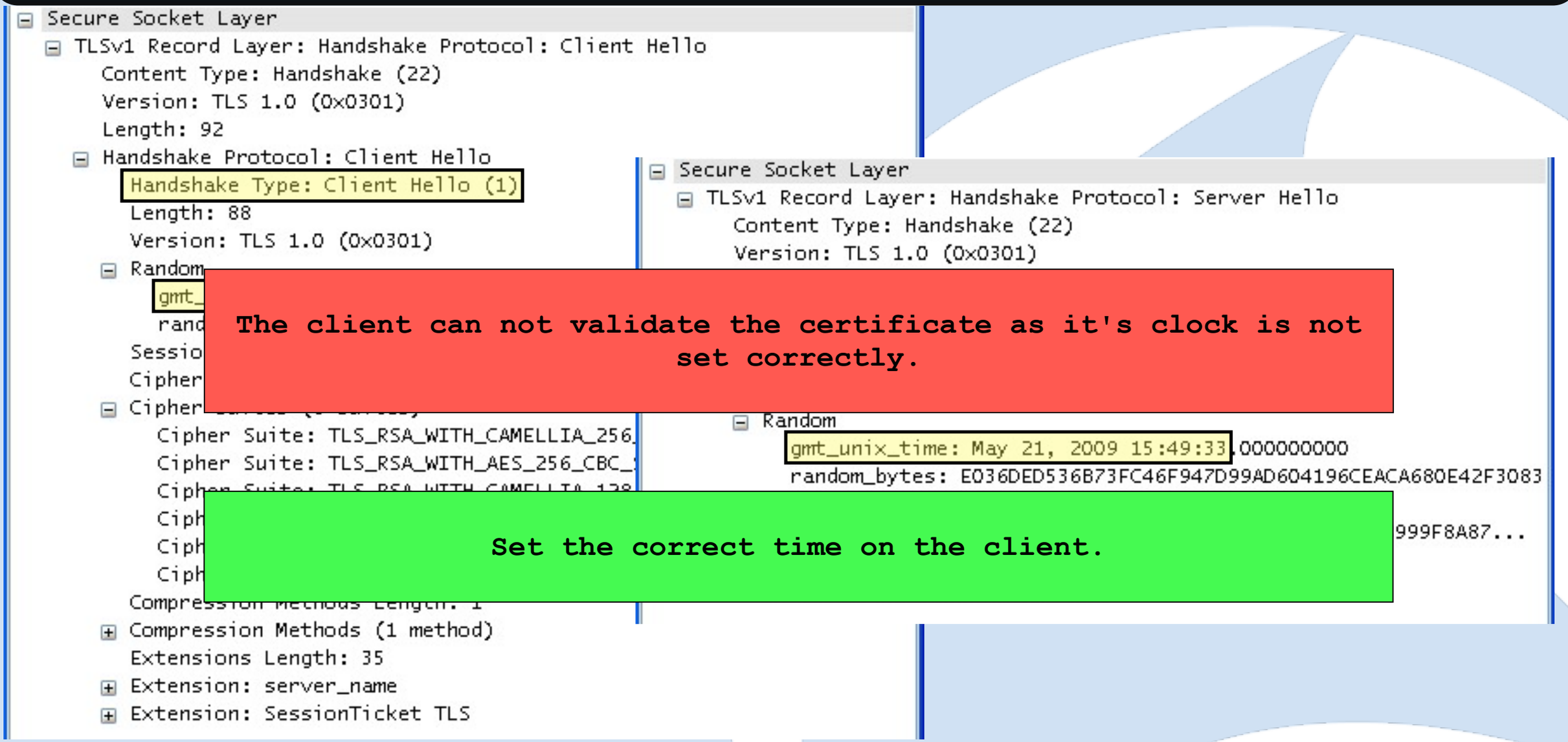


Common SSL problems II

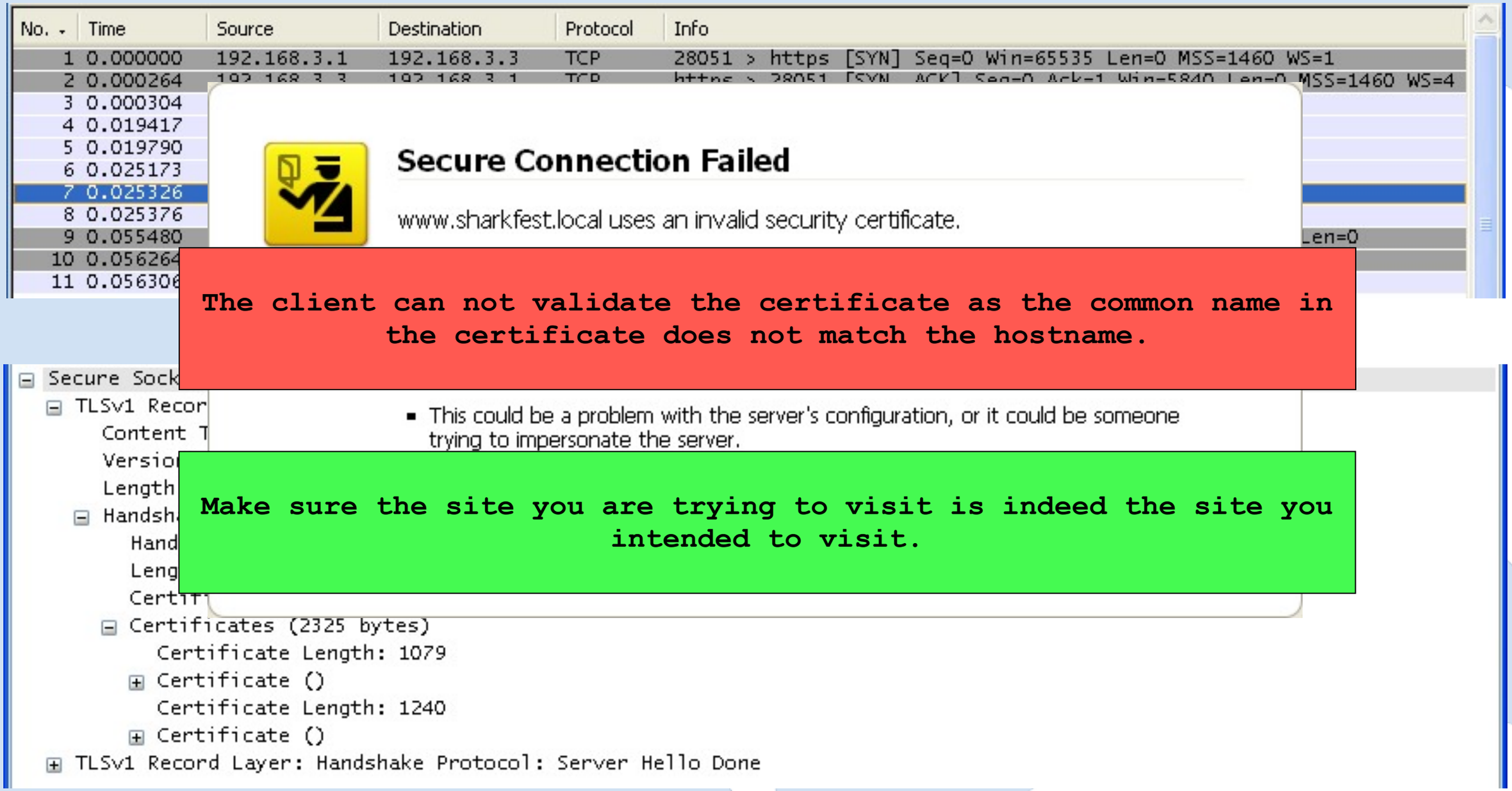
Common SSL problems III (1)



Common SSL problems III (2)



Common SSL problems IV



Common SSL problems V (1)

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.4	TCP	30245 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000289	192.168.3.4	192.168.3.1	TCP	https > 30245 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=4
3	0.000314	192.168.3.1	192.168.3.4	TCP	30245 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.017233	192.168.3.1	192.168.3.4	SSL	Client Hello
5	0.017657	192.168.3.4	192.168.3.1	TCP	https > 30245 [ACK] Seq=1 Ack=99 Win=5840 Len=0
6	0.019863	192.168.3.4	192.168.3.1	TLSv1	Server Hello,
7	0.019939	192.168.3.4	192.168.3.1	TLSv1	Certificate, Certificate Request, Server Hello Done
8	0.019966	192.168.3.1	192.168.3.4	TCP	30245 > https [ACK] Seq=99 Ack=2572 Win=128000 Len=0
9	3.299274	192.168.3.1	192.168.3.4	TLSv1	Certificate, Client Key Exchange, Certificate Verify, Change Ciph
10	3.300415	192.168.3.4	192.168.3.1	TLSv1	Alert (Level: Fatal, Description: Unknown CA)
11	3.300763	192.168.3.4	192.168.3.1	TCP	https > 30245 [FIN, ACK] Seq=2579 Ack=1501 Win=8768 Len=0
12	3.300791	192.168.3.1	192.168.3.4	TCP	30245 > https [ACK] Seq=1501 Ack=2580 Win=127992 Len=0
13	3.310232	192.168.3.1	192.168.3.4	TCP	30245 > https [FIN, ACK] Seq=1501 Ack=2580 Win=127992 Len=0
14	3.310386	192.168.3.4	192.168.3.1	TCP	https > 30245 [ACK] Seq=2580 Ack=1502 Win=8768 Len=0

Common SSL problems V (2)

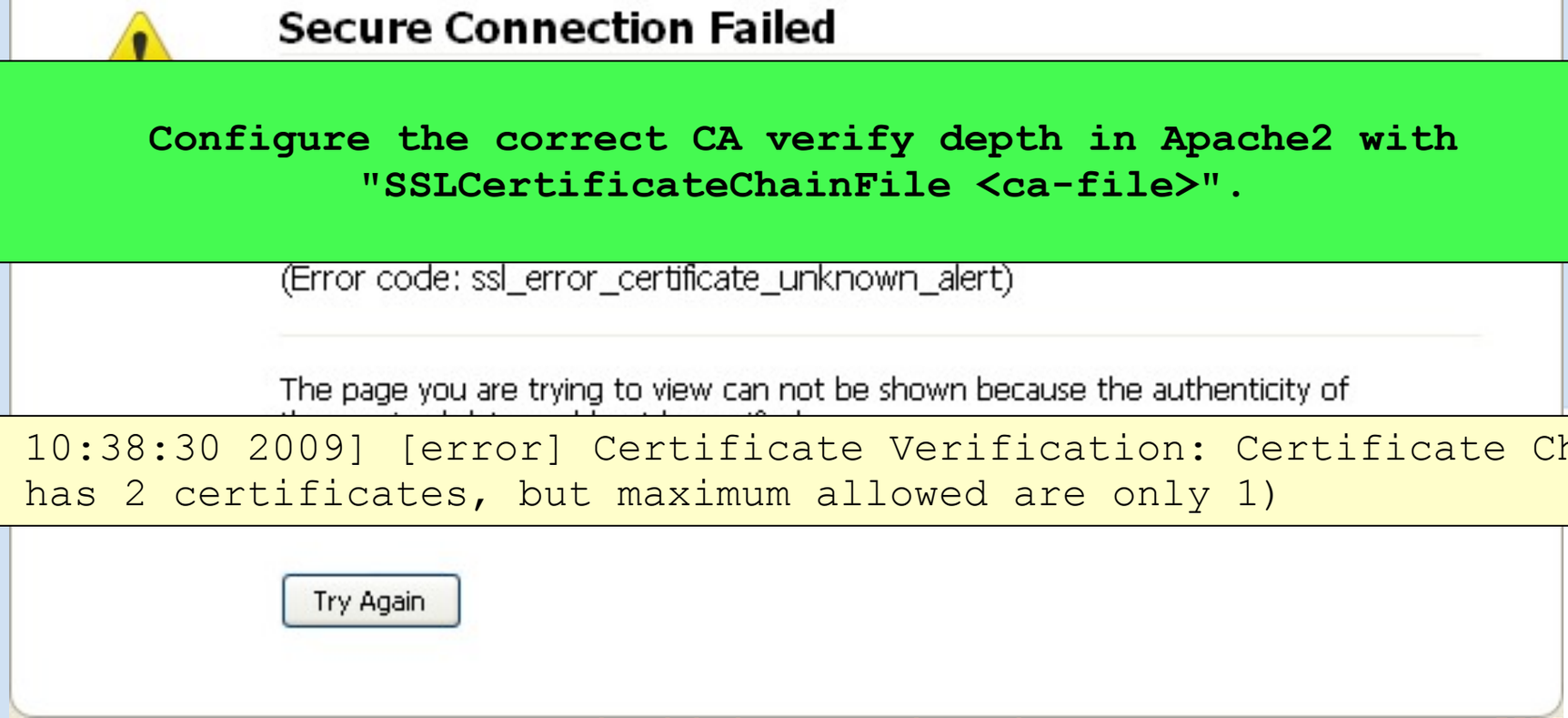

```
[Thu May 21 10:29:45 2009] [error] Certificate Verification: Error (2): unable to get
issuer certificate
```

Common SSL problems VI

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.4	TCP	30824 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000178	192.168.3.4	192.168.3.1	TCP	https > 30824 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=4
3	0.000214	192.168.3.1	192.168.3.4	TCP	30824 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.008474	192.168.3.1	192.168.3.4	TLSv1	Client Hello
5	0.008656	192.168.3.4	192.168.3.1	TCP	https > 30824 [ACK] Seq=1 Ack=99 Win=5840 Len=0
6	0.010907	192.168.3.4	192.168.3.1	TLSv1	Server Hello,
7	0.011001	192.168.3.4	192.168.3.1	TLSv1	Certificate, Certificate Request, Server Hello Done
8	0.011040	192.168.3.1	192.168.3.4	TCP	30824 > https [ACK] Seq=99 Ack=2726 Win=128000 Len=0
9	3.441257	192.168.3.1	192.168.3.4	TLSv1	Certificate, Client Key Exchange, Certificate Verify, Change Ciph
10	3.443320	192.168.3.4	192.168.3.1	TLSv1	Alert (Level: Fatal, Description: Certificate Unknown)
11	3.44376				
12	3.44379				
13	3.44583				=0
14	3.44598				

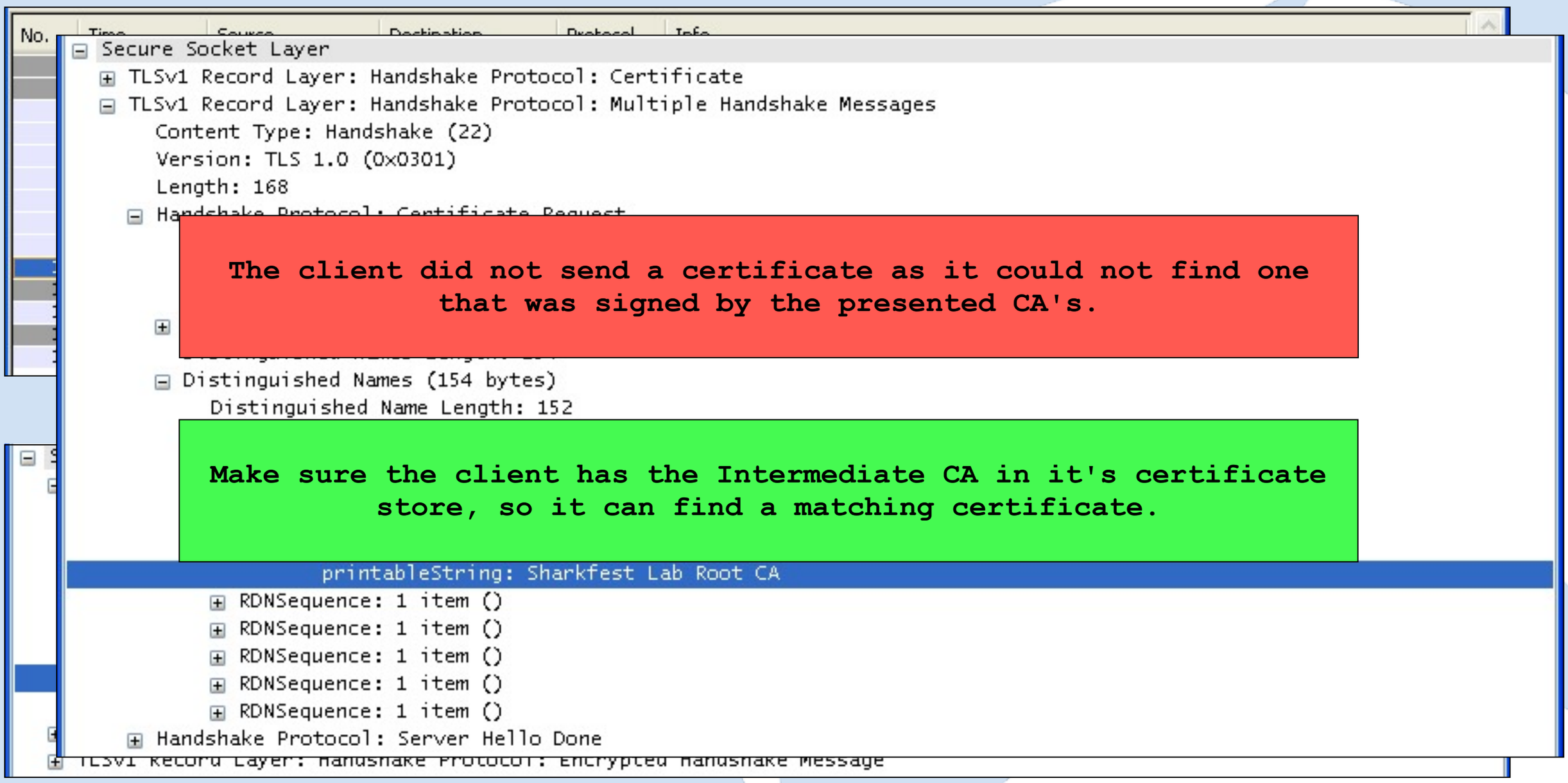
The server can not validate the client certificate as the CA chain used is larger than the allowed depth.

Configure the correct CA verify depth in Apache2 with "SSLCertificateChainFile <ca-file>".




```
[Thu May 21 10:38:30 2009] [error] Certificate Verification: Certificate Chain too
long (chain has 2 certificates, but maximum allowed are only 1)
```

Common SSL problems VII



Common SSL problems VIII

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.3.1	192.168.3.4	TCP	32123 > https [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=1
2	0.000085	192.168.3.4	192.168.3.1	TCP	https > 32123 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=4
3	0.000108	192.168.3.1	192.168.3.4	TCP	32123 > https [ACK] Seq=1 Ack=1 Win=128000 Len=0
4	0.009474	192.168.3.1	192.168.3.4	SSL	Client Hello
5	0.009680	192.168.3.4	192.168.3.1	TCP	https > 32123 [ACK] Seq=1 Ack=99 Win=5840 Len=0
6	0.011632	192.168.3.4	192.168.3.1	TLSv1	Server Hello,
7	0.011719	192.168.3.4	192.168.3.1	TLSv1	Certificate, Certificate Request, Server Hello Done
8	0.011744	192.168.3.1	192.168.3.4	TCP	32123 > https [ACK] Seq=99 Ack=2591 Win=128000 Len=0
9	5.275850	192.168.3.1	192.168.3.4	TCP	[TCP segment of a reassembled PDU]
10	5.275889	192.168.3.1	192.168.3.4	TLSv1	Certificate, Client Key Exchange, Certificate Verify, Change Ciph
11	5.276312				
12	5.283642				
13	5.2844(				
14	5.2844				
15	5.2874				
16	5.2880				

The server rejected the client certificate because it has been revoked by the signing CA.

An error occurred during a connection to private.sharkfest.local.

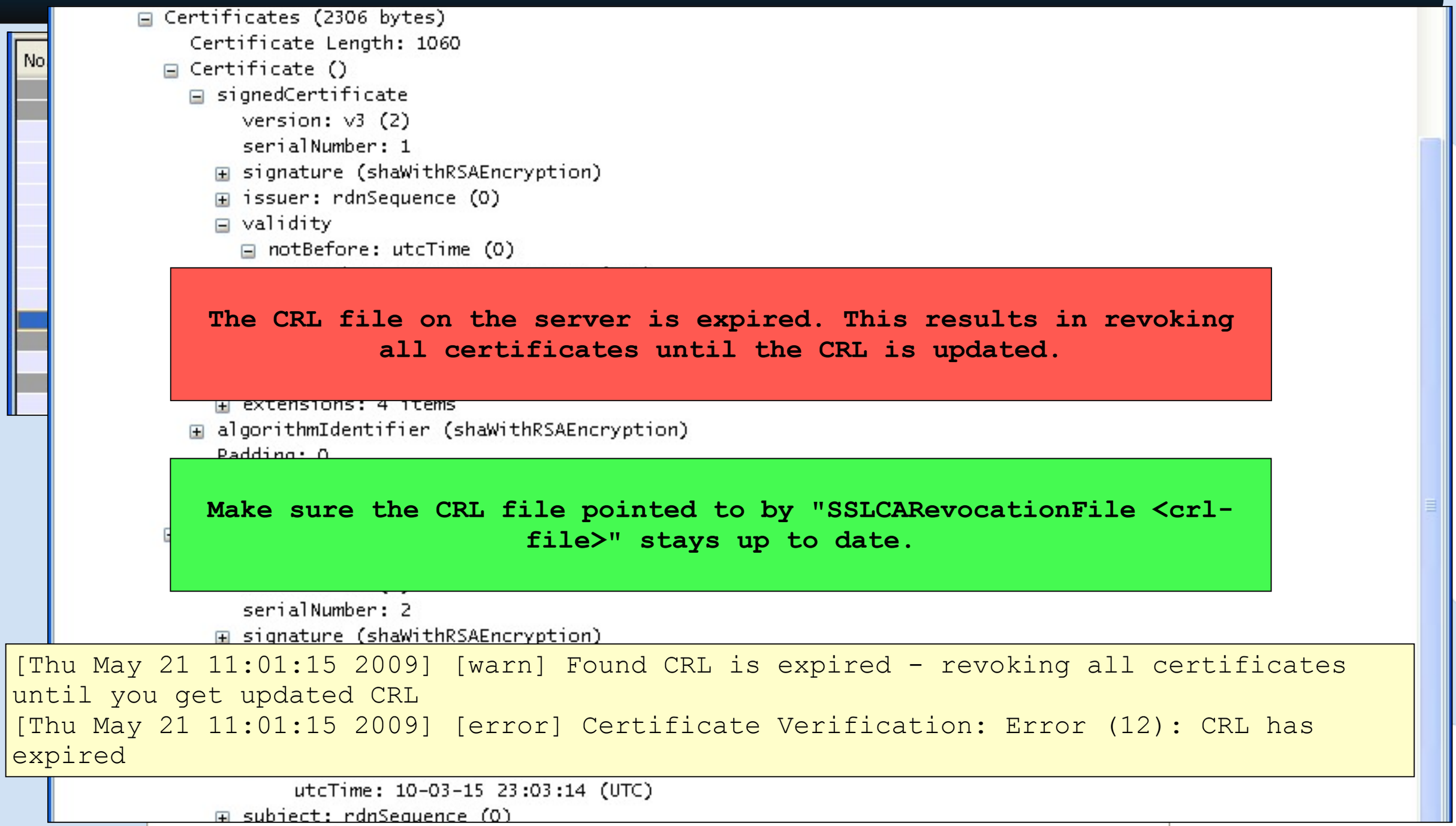
The client needs to request a new certificate.

The page you are trying to view can not be shown because the authenticity of the received data could not be verified.

Try Again

```
[Thu May 21 10:57:57 2009] [error] Certificate Verification: Error (23): certificate
revoked
```

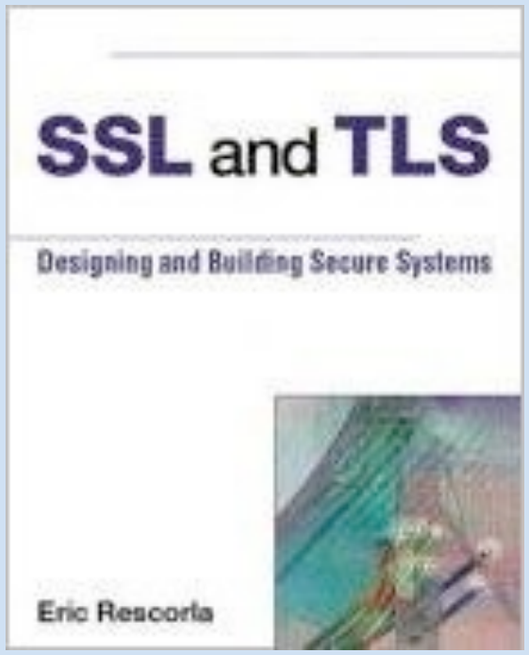
Common SSL problems IX



Agenda

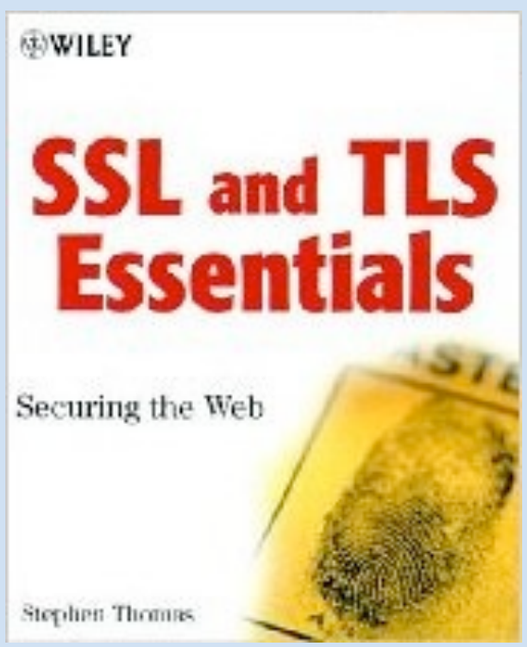
- Cryptology overview
- The SSL protocol
- Analyzing SSL with Wireshark
- Analyzing SSL with Tshark
- Common SSL connection problems
- Further reading
- Questions & Discussion

Further Reading about SSL



SSL and TLS: Designing and Building Secure Systems

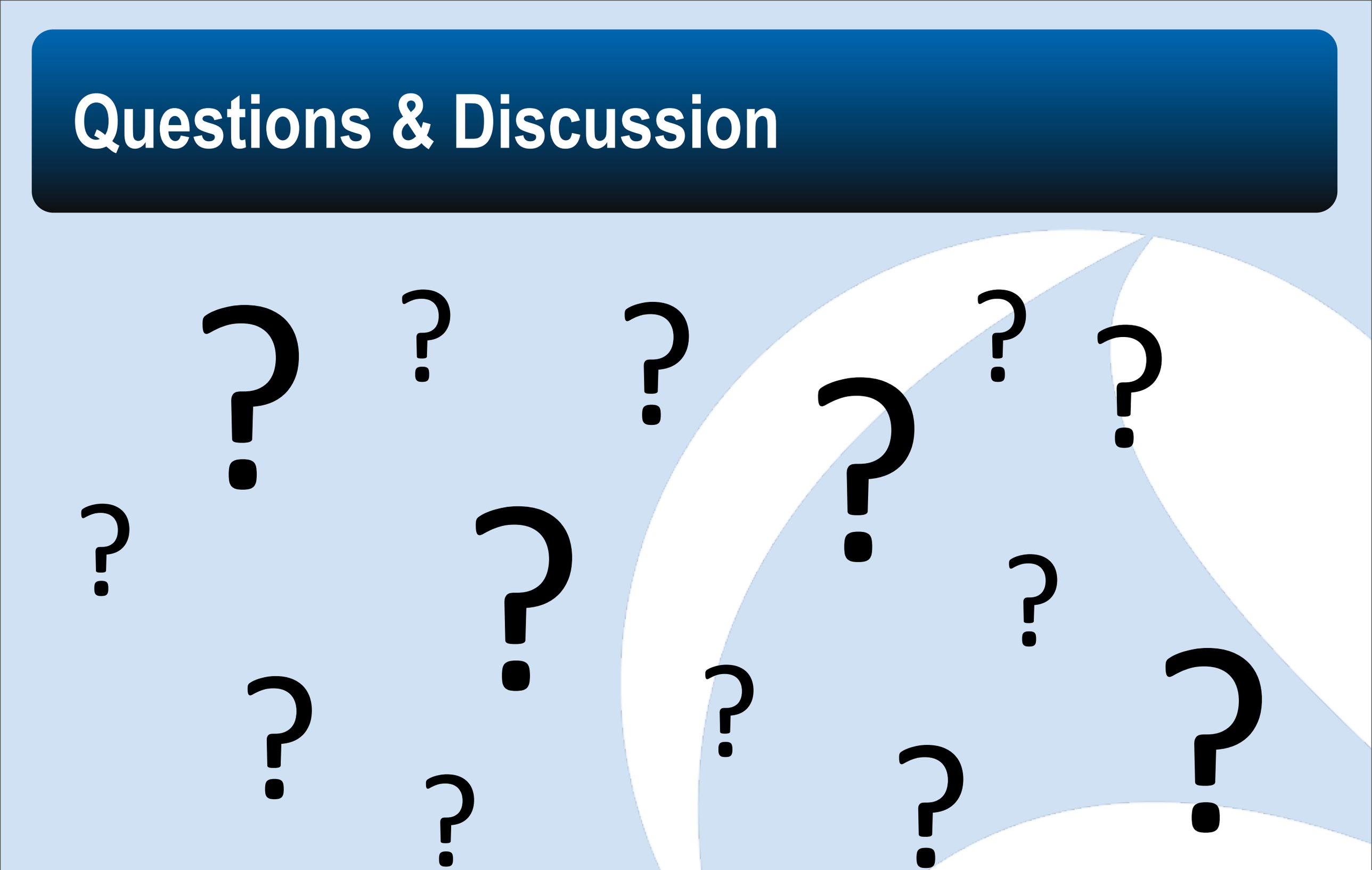
by Eric Rescorla



SSL and TLS Essentials: Securing the Web

by Stephen A. Thomas

Questions & Discussion



FIN/ACK, ACK, FIN/ACK, ACK!

Thank You!

If you would like to receive the tracefiles (and keys!)

that I used, please mail me: sake.blok@SYN-bit.nl