Transport Layer Security (TLS1.3)

NALINI ELKINS

INDUSTRY NETWORK TECHNOLOGY COUNCIL

PRESIDENT@INDUSTRYNETCOUNCIL.ORG

A few words about me

- President: Industry Network Technology Council
- Founder & CEO: Inside Products, Inc.
- Advisory Board: India Internet Engineering Society
- RFCs: RFC8250 (Embedded performance and diagnostics for IPv6) and others
- Product developer (OEMed by IBM and others)
- Working with IPv6 for 15 years
- Working with network management, diagnostic, performance issues at large brick-and-mortar enterprises for over 30 years

Agenda

- Introduction
- TLS handshake (1.2 vs. 1.3)
- TLS 1.3 implications
- Interesting Internet Drafts
- Post-quantum implications

TLS1.3

Internet Engineering Task Force (IETF)
Request for Comments: 8446
Obsoletes: 5077, 5246, 6961

E. Rescorla
Mozilla

August 2018

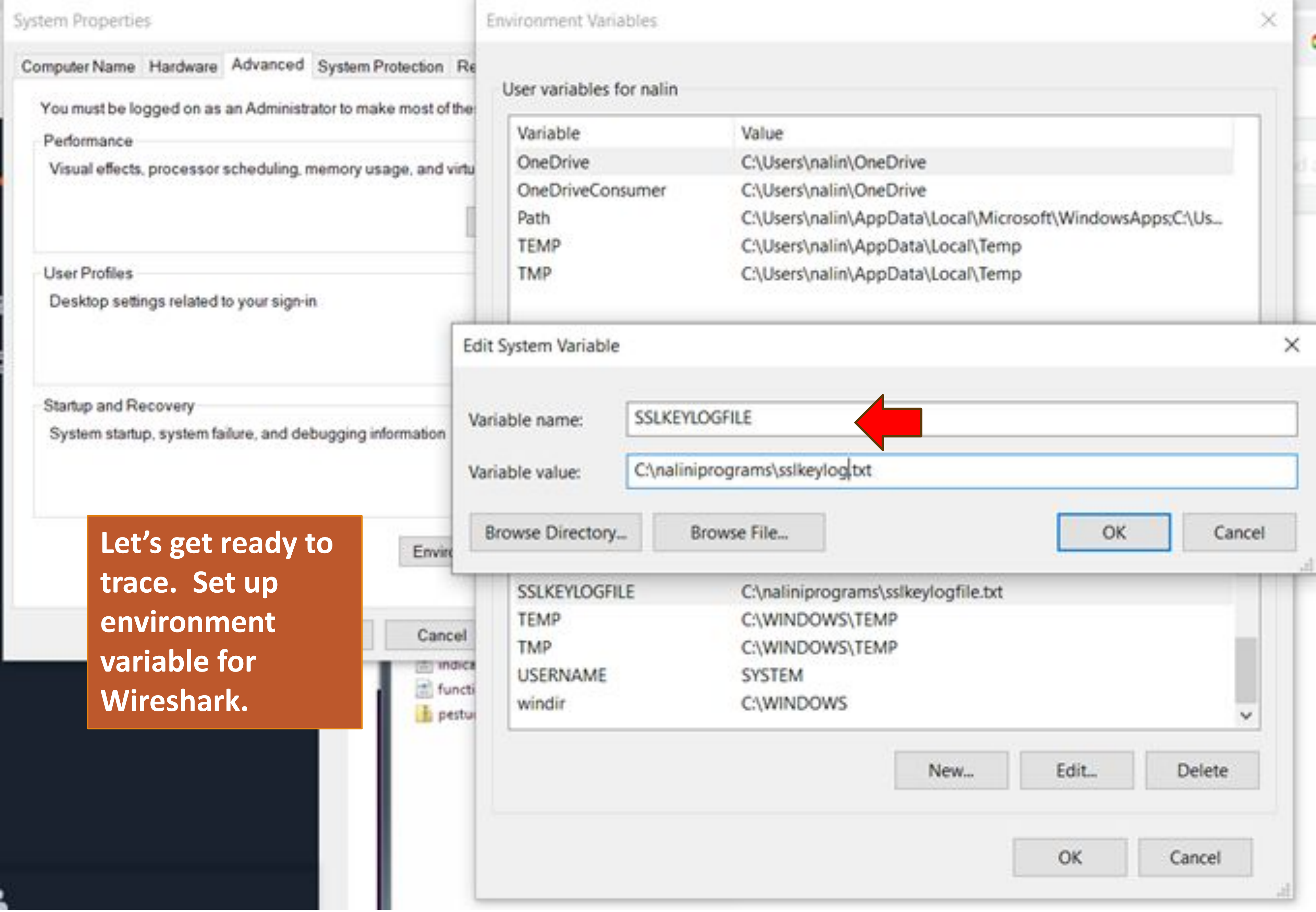
The Transport Layer Security (TLS) Protocol Version 1.3

Abstract

This document specifies version 1.3 of the Transport Layer Security (TLS) protocol. TLS allows client/server applications to communicate over the Internet in a way that is designed to prevent eavesdropping, tampering, and message forgery.

TLS1.3: Important Features

- More of the handshake encrypted
- Improved performance
- Number of deprecations
- Future changes will be made to TLS1.3 only (most likely)



System Properties
Computer Name
Hardware
Advanced
System Protection
Re
You must be logged on as an Administrator to make most of the
Performance
Visual effects, processor scheduling, memory usage, and virtu
User Profiles
Desktop settings related to your sign-in
Startup and Recovery
System startup, system failure, and debugging information
Envir
Cancel
Environment Variables

User variables for nalin

Variable	Value
OneDrive	C:\Users\nalin\OneDrive
OneDriveConsumer	C:\Users\nalin\OneDrive
Path	C:\Users\nalin\AppData\Local\Microsoft\WindowsApps;C:\Us...
TEMP	C:\Users\nalin\AppData\Local\Temp
TMP	C:\Users\nalin\AppData\Local\Temp

Edit System Variable

Variable name: SSLKEYLOGFILE

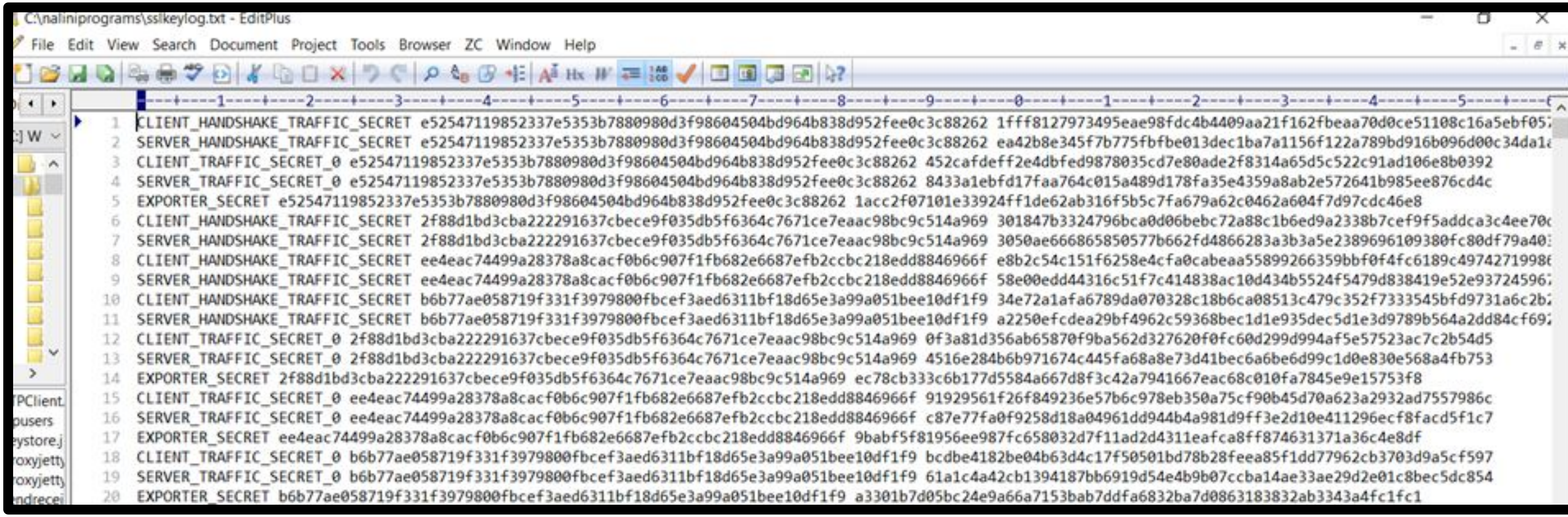
Variable value: C:\naliniprograms\sslkeylog.txt

Browse Directory...
Browse File...
OK
Cancel

SSLKEYLOGFILE	C:\naliniprograms\sslkeylogfile.txt
TEMP	C:\WINDOWS\TEMP
TMP	C:\WINDOWS\TEMP
USERNAME	SYSTEM
windir	C:\WINDOWS

Let’s get ready to trace. Set up environment variable for Wireshark.

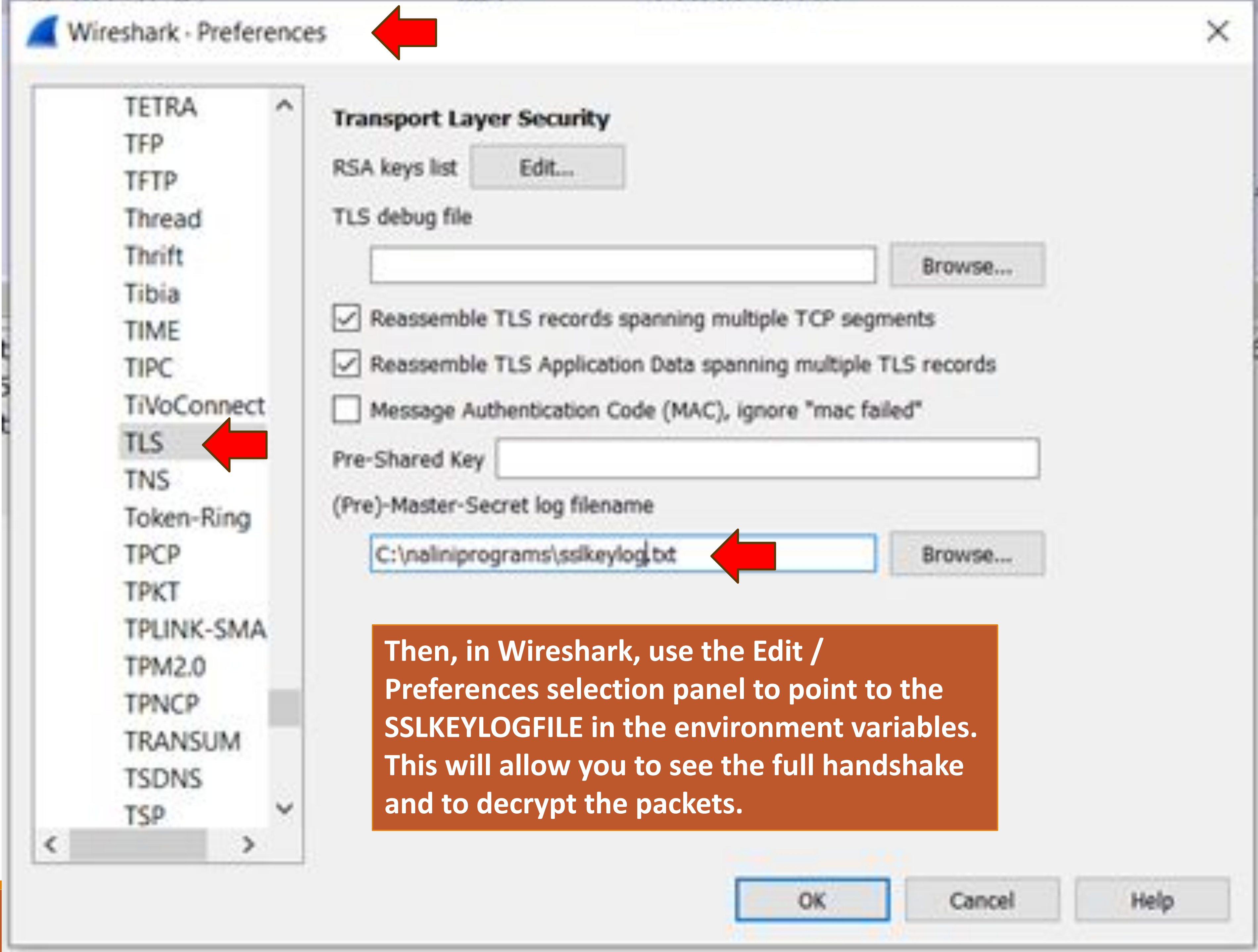
SSLKEYLOGFILE



C:\naliniprograms\sslkeylog.txt - EditPlus
File Edit View Search Document Project Tools Browser ZC Window Help

```
CLIENT_HANDSHAKE_TRAFFIC_SECRET e52547119852337e5353b7880980d3f98604504bd964b838d952fee0c3c88262 1fff8127973495eae98fdc4b4409aa21f162fbeaa70d0ce51108c16a5ebf05
SERVER_HANDSHAKE_TRAFFIC_SECRET e52547119852337e5353b7880980d3f98604504bd964b838d952fee0c3c88262 ea42b8e345f7b775fbfbe013dec1ba7a1156f122a789bd916b096d00c34da1
CLIENT_TRAFFIC_SECRET_0 e52547119852337e5353b7880980d3f98604504bd964b838d952fee0c3c88262 452cafdeff2e4dbfed9878035cd7e80ade2f8314a65d5c522c91ad106e8b0392
SERVER_TRAFFIC_SECRET_0 e52547119852337e5353b7880980d3f98604504bd964b838d952fee0c3c88262 8433a1ebfd17faa764c015a489d178fa35e4359a8ab2e572641b985ee876cd4c
EXPORTER_SECRET e52547119852337e5353b7880980d3f98604504bd964b838d952fee0c3c88262 1acc2f07101e33924ff1de62ab316f5b5c7fa679a62c0462a604f7d97cdc46e8
CLIENT_HANDSHAKE_TRAFFIC_SECRET 2f88d1bd3cba222291637cbece9f035db5f6364c7671ce7eaac98bc9c514a969 301847b3324796bca0d06bebc72a88c1b6ed9a2338b7cef9f5addca3c4ee70
SERVER_HANDSHAKE_TRAFFIC_SECRET 2f88d1bd3cba222291637cbece9f035db5f6364c7671ce7eaac98bc9c514a969 3050ae666865850577b662fd4866283a3b3a5e2389696109380fc80df79a40
CLIENT_HANDSHAKE_TRAFFIC_SECRET ee4eac74499a28378a8cacf0b6c907f1fb682e6687efb2ccbc218edd8846966f e8b2c54c151f6258e4cfa0cabeaa55899266359bbf0f4fc6189c4974271998
SERVER_HANDSHAKE_TRAFFIC_SECRET ee4eac74499a28378a8cacf0b6c907f1fb682e6687efb2ccbc218edd8846966f 58e00edd44316c51f7c414838ac10d434b5524f5479d838419e52e93724596
CLIENT_HANDSHAKE_TRAFFIC_SECRET b6b77ae058719f331f3979800fbcef3aed6311bf18d65e3a99a051bee10df1f9 34e72a1afa6789da070328c18b6ca08513c479c352f7333545bfd9731a6c2b
SERVER_HANDSHAKE_TRAFFIC_SECRET b6b77ae058719f331f3979800fbcef3aed6311bf18d65e3a99a051bee10df1f9 a2250efcdea29bf4962c59368bec1d1e935dec5d1e3d9789b564a2dd84cf69
CLIENT_TRAFFIC_SECRET_0 2f88d1bd3cba222291637cbece9f035db5f6364c7671ce7eaac98bc9c514a969 0f3a81d356ab65870f9ba562d327620f0fc60d299d994af5e57523ac7c2b54d5
SERVER_TRAFFIC_SECRET_0 2f88d1bd3cba222291637cbece9f035db5f6364c7671ce7eaac98bc9c514a969 4516e284b6b971674c445fa68a8e73d41bec6a6be6d99c1d0e830e568a4fb753
EXPORTER_SECRET 2f88d1bd3cba222291637cbece9f035db5f6364c7671ce7eaac98bc9c514a969 ec78cb333c6b177d5584a667d8f3c42a7941667eac68c010fa7845e9e15753f8
CLIENT_TRAFFIC_SECRET_0 ee4eac74499a28378a8cacf0b6c907f1fb682e6687efb2ccbc218edd8846966f 91929561f26f849236e57b6c978eb350a75cf90b45d70a623a2932ad7557986c
SERVER_TRAFFIC_SECRET_0 ee4eac74499a28378a8cacf0b6c907f1fb682e6687efb2ccbc218edd8846966f c87e77fa0f9258d18a04961dd944b4a981d9ff3e2d10e411296ecf8facd5f1c7
EXPORTER_SECRET ee4eac74499a28378a8cacf0b6c907f1fb682e6687efb2ccbc218edd8846966f 9babf5f81956ee987fc658032d7f11ad2d4311eafca8ff874631371a36c4e8df
CLIENT_TRAFFIC_SECRET_0 b6b77ae058719f331f3979800fbcef3aed6311bf18d65e3a99a051bee10df1f9 bcdbe4182be04b63d4c17f50501bd78b28feea85f1dd77962cb3703d9a5cf597
SERVER_TRAFFIC_SECRET_0 b6b77ae058719f331f3979800fbcef3aed6311bf18d65e3a99a051bee10df1f9 61a1c4a42cb1394187bb6919d54e4b9b07ccba14ae33ae29d2e01c8bec5dc854
EXPORTER_SECRET b6b77ae058719f331f3979800fbcef3aed6311bf18d65e3a99a051bee10df1f9 a3301b7d05bc24e9a66a7153bab7ddfa6832ba7d0863183832ab3343a4fc1fc1
```

- Has the secrets for TLS1.x (1.0-1.3)
- File can get big!



Wireshark · Preferences

Transport Layer Security

RSA keys list [Edit...]

TLS debug file

☑ Reassemble TLS records spanning multiple TCP segments

☑ Reassemble TLS Application Data spanning multiple TLS records

□ Message Authentication Code (MAC), ignore "mac failed"

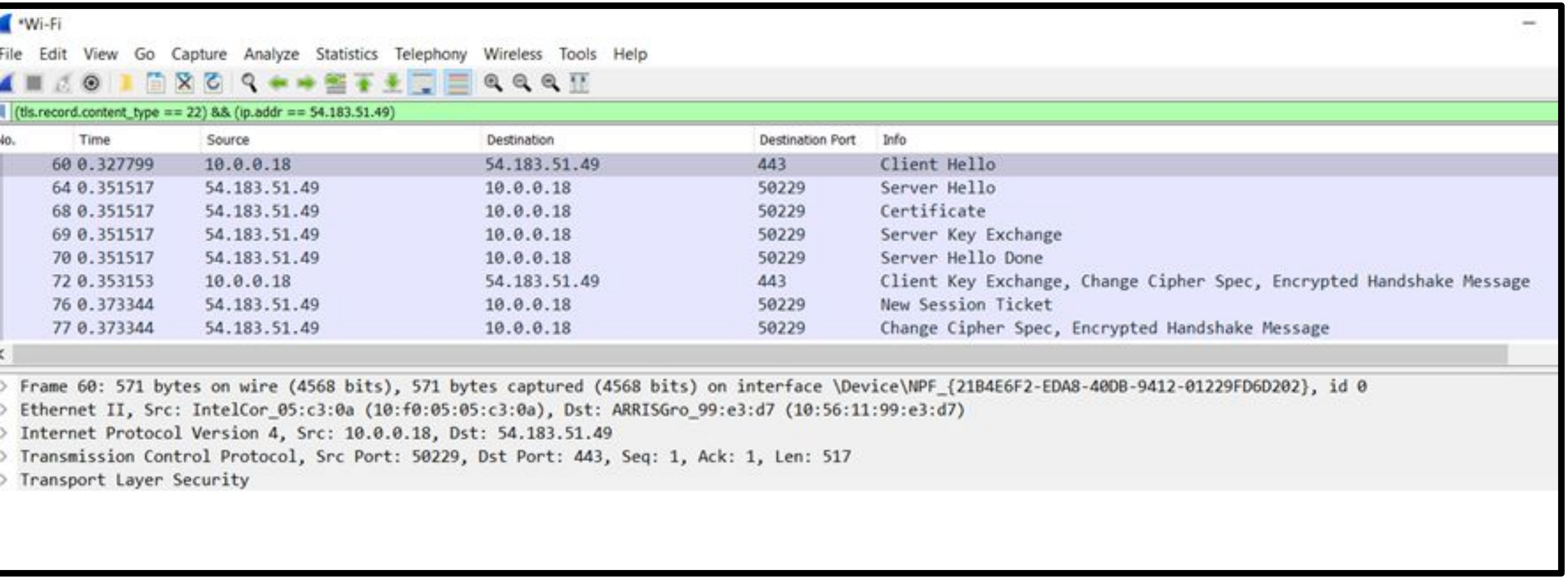
Pre-Shared Key

(Pre)-Master-Secret log filename

C:\naliniprograms\sslkeylog.txt

Then, in Wireshark, use the Edit / Preferences selection panel to point to the SSLKEYLOGFILE in the environment variables. This will allow you to see the full handshake and to decrypt the packets.

Let’s do some traces!

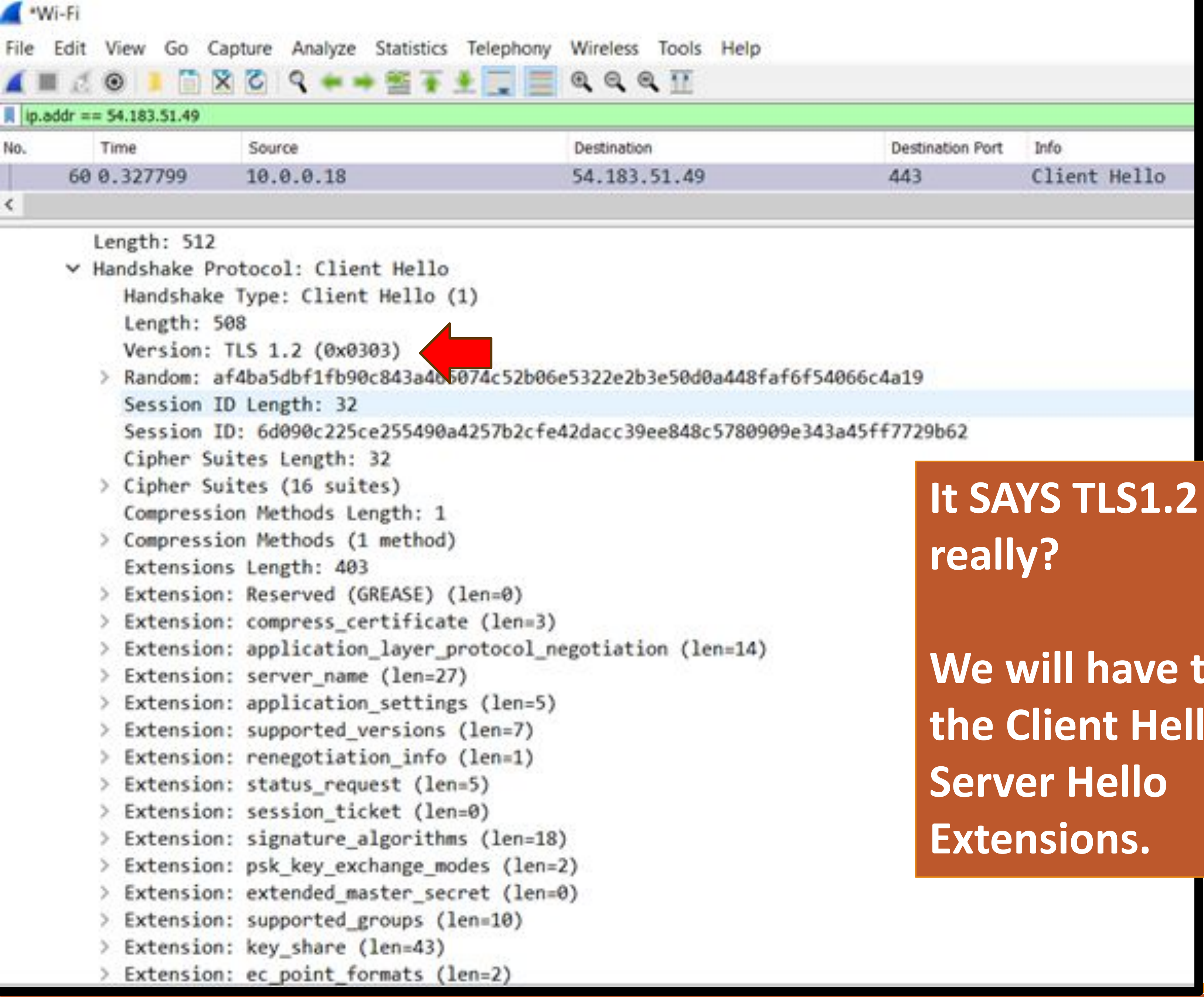

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

(tls.record.content_type == 22) && (ip.addr == 54.183.51.49)

No.	Time	Source	Destination	Destination Port	Info
60	0.327799	10.0.0.18	54.183.51.49	443	Client Hello
64	0.351517	54.183.51.49	10.0.0.18	50229	Server Hello
68	0.351517	54.183.51.49	10.0.0.18	50229	Certificate
69	0.351517	54.183.51.49	10.0.0.18	50229	Server Key Exchange
70	0.351517	54.183.51.49	10.0.0.18	50229	Server Hello Done
72	0.353153	10.0.0.18	54.183.51.49	443	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
76	0.373344	54.183.51.49	10.0.0.18	50229	New Session Ticket
77	0.373344	54.183.51.49	10.0.0.18	50229	Change Cipher Spec, Encrypted Handshake Message

```
> Frame 60: 571 bytes on wire (4568 bits), 571 bytes captured (4568 bits) on interface \Device\NPF_{21B4E6F2-EDA8-40DB-9412-01229FD6D202}, id 0
> Ethernet II, Src: IntelCor_05:c3:0a (10:f0:05:05:c3:0a), Dst: ARRISGro_99:e3:d7 (10:56:11:99:e3:d7)
> Internet Protocol Version 4, Src: 10.0.0.18, Dst: 54.183.51.49
> Transmission Control Protocol, Src Port: 50229, Dst Port: 443, Seq: 1, Ack: 1, Len: 517
> Transport Layer Security
```

- We can see TLS handshakes
- Are they TLS1.2 or TLS1.3?


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

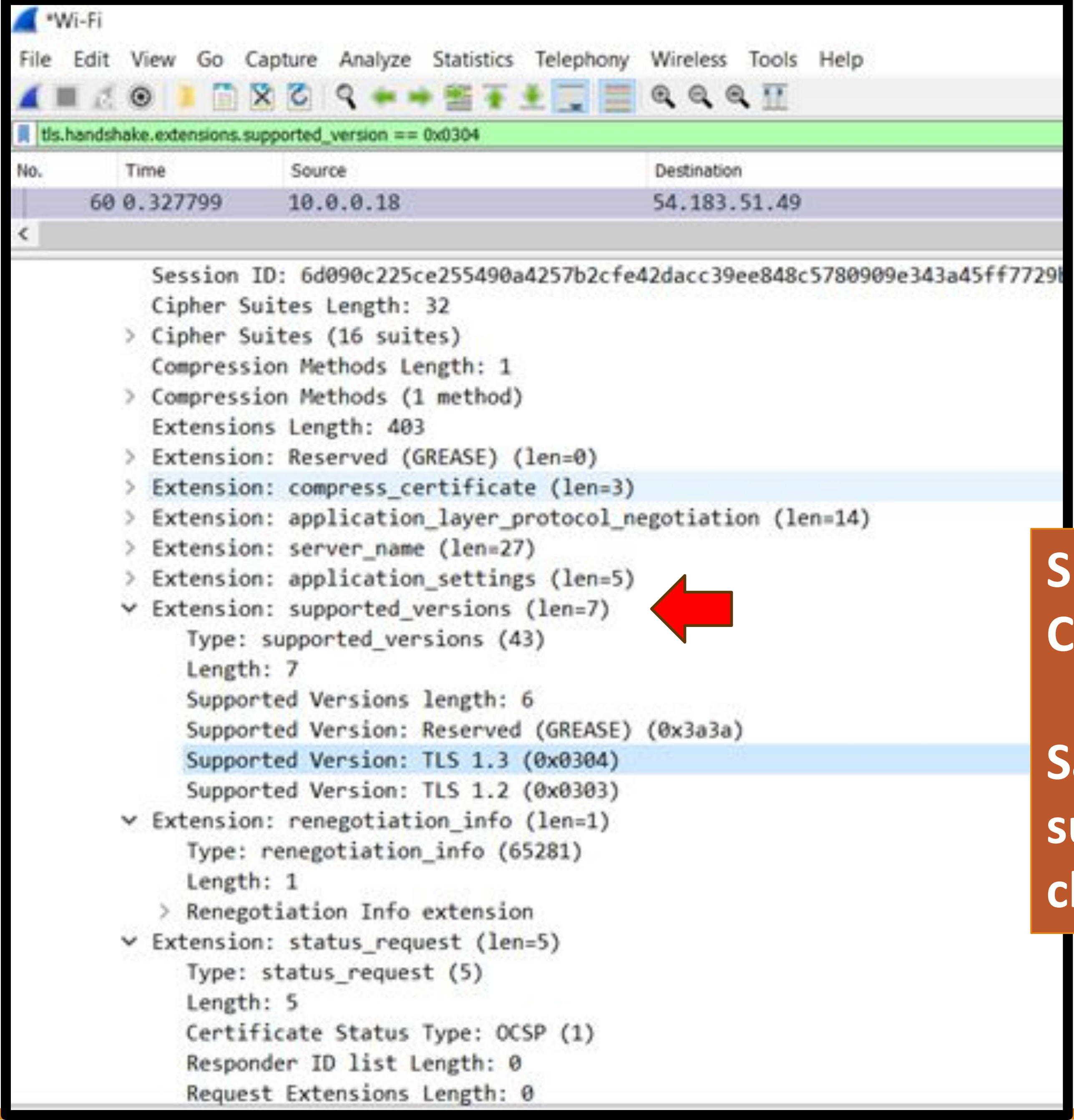
```
ip.addr == 54.183.51.49
```

No.	Time	Source	Destination	Destination Port	Info
60	0.327799	10.0.0.18	54.183.51.49	443	Client Hello

```
      Length: 512
    ⌄ Handshake Protocol: Client Hello
        Handshake Type: Client Hello (1)
        Length: 508
        Version: TLS 1.2 (0x0303)
      > Random: af4ba5dbf1fb90c843a465074c52b06e5322e2b3e50d0a448faf6f54066c4a19
        Session ID Length: 32
        Session ID: 6d090c225ce255490a4257b2cfe42dacc39ee848c5780909e343a45ff7729b62
        Cipher Suites Length: 32
      > Cipher Suites (16 suites)
        Compression Methods Length: 1
      > Compression Methods (1 method)
        Extensions Length: 403
      > Extension: Reserved (GREASE) (len=0)
      > Extension: compress_certificate (len=3)
      > Extension: application_layer_protocol_negotiation (len=14)
      > Extension: server_name (len=27)
      > Extension: application_settings (len=5)
      > Extension: supported_versions (len=7)
      > Extension: renegotiation_info (len=1)
      > Extension: status_request (len=5)
      > Extension: session_ticket (len=0)
      > Extension: signature_algorithms (len=18)
      > Extension: psk_key_exchange_modes (len=2)
      > Extension: extended_master_secret (len=0)
      > Extension: supported_groups (len=10)
      > Extension: key_share (len=43)
      > Extension: ec_point_formats (len=2)
```

It SAYS TLS1.2 but is it really?

We will have to look in the Client Hello and Server Hello Extensions.



*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

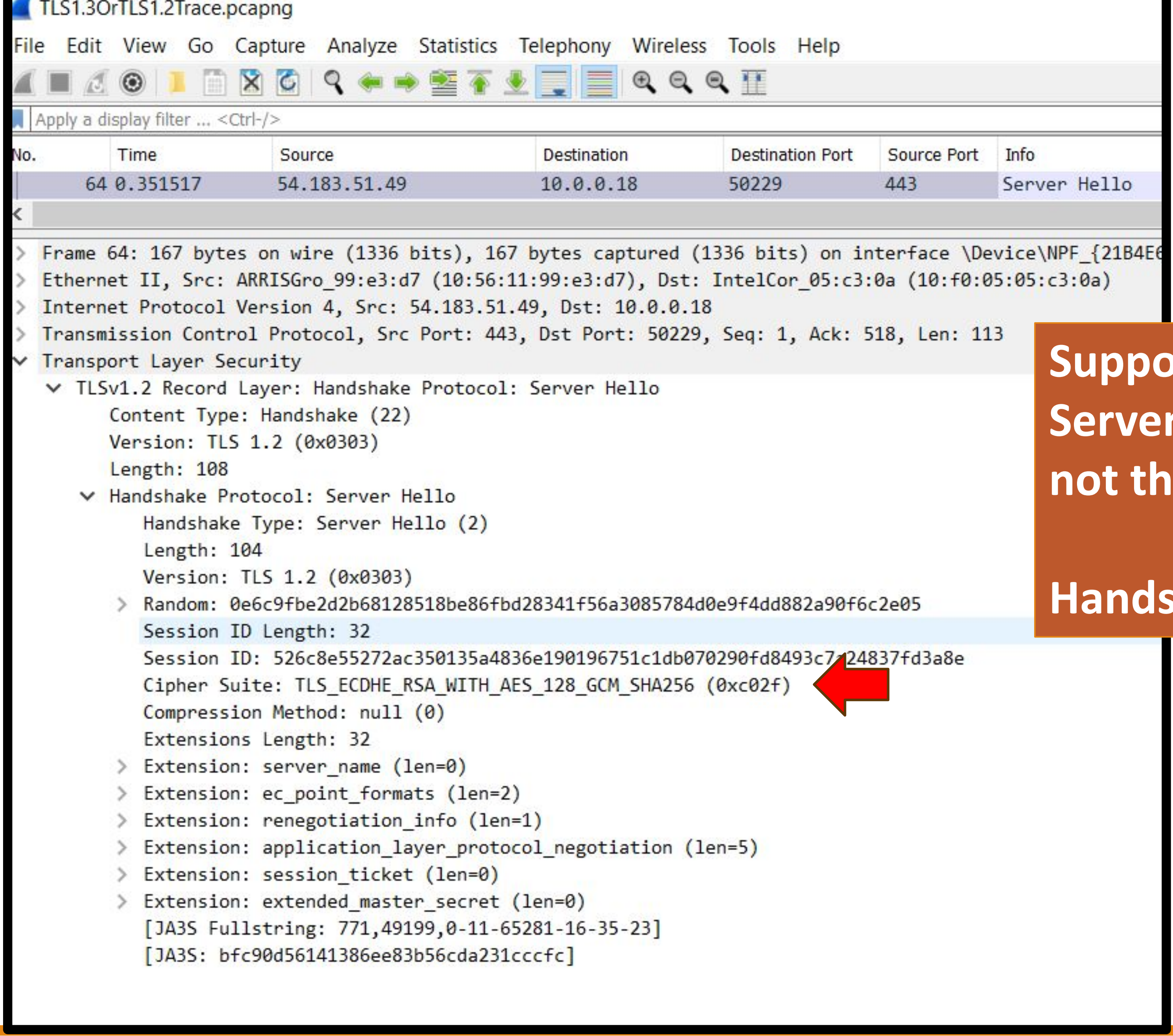
tls.handshake.extensions.supported_version == 0x0304

No.	Time	Source	Destination
60	0.327799	10.0.0.18	54.183.51.49

```
    Session ID: 6d090c225ce255490a4257b2cfe42dacc39ee848c5780909e343a45ff7729
    Cipher Suites Length: 32
  > Cipher Suites (16 suites)
    Compression Methods Length: 1
  > Compression Methods (1 method)
    Extensions Length: 403
  > Extension: Reserved (GREASE) (len=0)
  > Extension: compress_certificate (len=3)
  > Extension: application_layer_protocol_negotiation (len=14)
  > Extension: server_name (len=27)
  > Extension: application_settings (len=5)
  v Extension: supported_versions (len=7)
      Type: supported_versions (43)
      Length: 7
      Supported Versions length: 6
      Supported Version: Reserved (GREASE) (0x3a3a)
      Supported Version: TLS 1.3 (0x0304)
      Supported Version: TLS 1.2 (0x0303)
  v Extension: renegotiation_info (len=1)
      Type: renegotiation_info (65281)
      Length: 1
    > Renegotiation Info extension
  v Extension: status_request (len=5)
      Type: status_request (5)
      Length: 5
      Certificate Status Type: OCSP (1)
      Responder ID list Length: 0
      Request Extensions Length: 0
```

Supported Versions Client Hello Extension.

Says what versions are supported by this client.


TLS1.3OrTLS1.2Trace.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

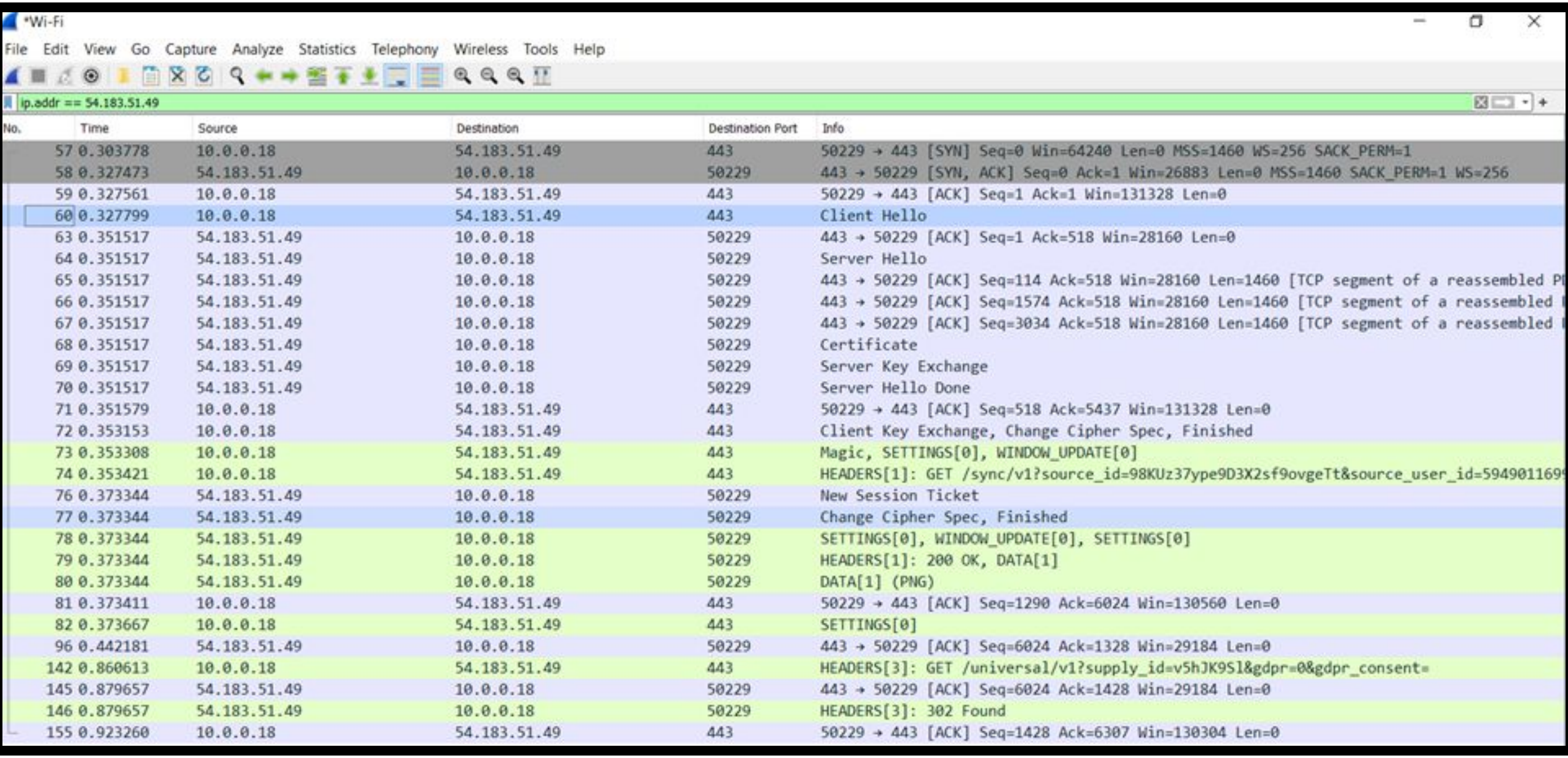
No.	Time	Source	Destination	Destination Port	Source Port	Info
64	0.351517	54.183.51.49	10.0.0.18	50229	443	Server Hello

```
> Frame 64: 167 bytes on wire (1336 bits), 167 bytes captured (1336 bits) on interface \Device\NPF_{21B4E6
> Ethernet II, Src: ARRISGro_99:e3:d7 (10:56:11:99:e3:d7), Dst: IntelCor_05:c3:0a (10:f0:05:05:c3:0a)
> Internet Protocol Version 4, Src: 54.183.51.49, Dst: 10.0.0.18
> Transmission Control Protocol, Src Port: 443, Dst Port: 50229, Seq: 1, Ack: 518, Len: 113
v Transport Layer Security
    v TLSv1.2 Record Layer: Handshake Protocol: Server Hello
        Content Type: Handshake (22)
        Version: TLS 1.2 (0x0303)
        Length: 108
      v Handshake Protocol: Server Hello
          Handshake Type: Server Hello (2)
          Length: 104
          Version: TLS 1.2 (0x0303)
        > Random: 0e6c9fbe2d2b68128518be86fbd28341f56a3085784d0e9f4dd882a90f6c2e05
          Session ID Length: 32
          Session ID: 526c8e55272ac350135a4836e190196751c1db070290fd8493c7[illegible]24837fd3a8e
          Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
          Compression Method: null (0)
          Extensions Length: 32
        > Extension: server_name (len=0)
        > Extension: ec_point_formats (len=2)
        > Extension: renegotiation_info (len=1)
        > Extension: application_layer_protocol_negotiation (len=5)
        > Extension: session_ticket (len=0)
        > Extension: extended_master_secret (len=0)
          [JA3S Fullstring: 771,49199,0-11-65281-16-35-23]
          [JA3S: bfc90d56141386ee83b56cda231cccfc]
```

Supported Versions Server Hello Extensio not there.

Handshake is TLS1.2.

Let's decrypt!



*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

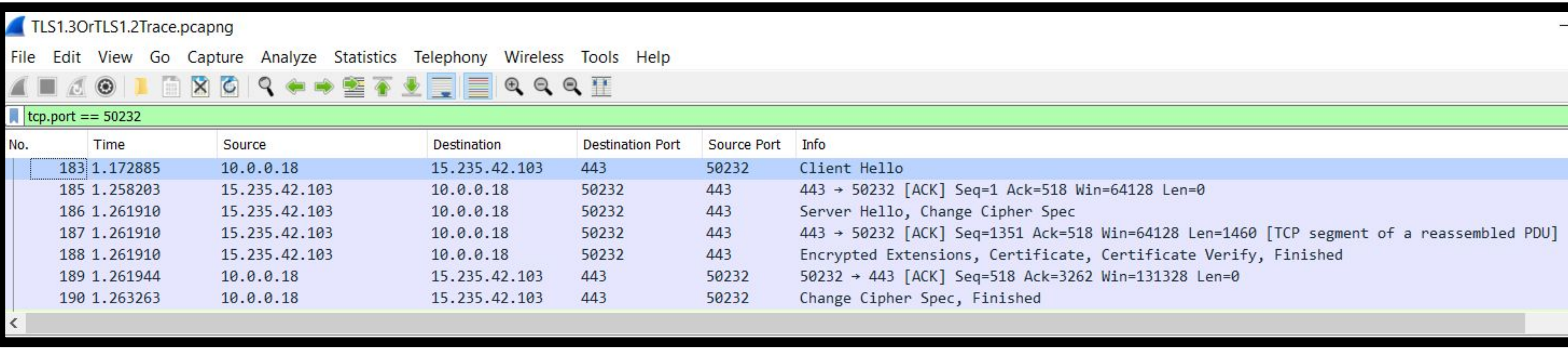
ip.addr == 54.183.51.49

No.	Time	Source	Destination	Destination Port	Info
57	0.303778	10.0.0.18	54.183.51.49	443	50229 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
58	0.327473	54.183.51.49	10.0.0.18	50229	443 → 50229 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0 MSS=1460 SACK_PERM=1 WS=256
59	0.327561	10.0.0.18	54.183.51.49	443	50229 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0
60	0.327799	10.0.0.18	54.183.51.49	443	Client Hello
63	0.351517	54.183.51.49	10.0.0.18	50229	443 → 50229 [ACK] Seq=1 Ack=518 Win=28160 Len=0
64	0.351517	54.183.51.49	10.0.0.18	50229	Server Hello
65	0.351517	54.183.51.49	10.0.0.18	50229	443 → 50229 [ACK] Seq=114 Ack=518 Win=28160 Len=1460 [TCP segment of a reassembled P
66	0.351517	54.183.51.49	10.0.0.18	50229	443 → 50229 [ACK] Seq=1574 Ack=518 Win=28160 Len=1460 [TCP segment of a reassembled
67	0.351517	54.183.51.49	10.0.0.18	50229	443 → 50229 [ACK] Seq=3034 Ack=518 Win=28160 Len=1460 [TCP segment of a reassembled
68	0.351517	54.183.51.49	10.0.0.18	50229	Certificate
69	0.351517	54.183.51.49	10.0.0.18	50229	Server Key Exchange
70	0.351517	54.183.51.49	10.0.0.18	50229	Server Hello Done
71	0.351579	10.0.0.18	54.183.51.49	443	50229 → 443 [ACK] Seq=518 Ack=5437 Win=131328 Len=0
72	0.353153	10.0.0.18	54.183.51.49	443	Client Key Exchange, Change Cipher Spec, Finished
73	0.353308	10.0.0.18	54.183.51.49	443	Magic, SETTINGS[0], WINDOW_UPDATE[0]
74	0.353421	10.0.0.18	54.183.51.49	443	HEADERS[1]: GET /sync/v1?source_id=98KUz37ype9D3X2sf9ovgeTt&source_user_id=594901169
76	0.373344	54.183.51.49	10.0.0.18	50229	New Session Ticket
77	0.373344	54.183.51.49	10.0.0.18	50229	Change Cipher Spec, Finished
78	0.373344	54.183.51.49	10.0.0.18	50229	SETTINGS[0], WINDOW_UPDATE[0], SETTINGS[0]
79	0.373344	54.183.51.49	10.0.0.18	50229	HEADERS[1]: 200 OK, DATA[1]
80	0.373344	54.183.51.49	10.0.0.18	50229	DATA[1] (PNG)
81	0.373411	10.0.0.18	54.183.51.49	443	50229 → 443 [ACK] Seq=1290 Ack=6024 Win=130560 Len=0
82	0.373667	10.0.0.18	54.183.51.49	443	SETTINGS[0]
96	0.442181	54.183.51.49	10.0.0.18	50229	443 → 50229 [ACK] Seq=6024 Ack=1328 Win=29184 Len=0
142	0.860613	10.0.0.18	54.183.51.49	443	HEADERS[3]: GET /universal/v1?supply_id=v5hJK9Sl&gdpr=0&gdpr_consent=
145	0.879657	54.183.51.49	10.0.0.18	50229	443 → 50229 [ACK] Seq=6024 Ack=1428 Win=29184 Len=0
146	0.879657	54.183.51.49	10.0.0.18	50229	HEADERS[3]: 302 Found
155	0.923260	10.0.0.18	54.183.51.49	443	50229 → 443 [ACK] Seq=1428 Ack=6307 Win=130304 Len=0

- Full TLS1.2 handshake

- Can also see the packet payload because of SSLKEYLOGFILE

Let’s look at another handshake

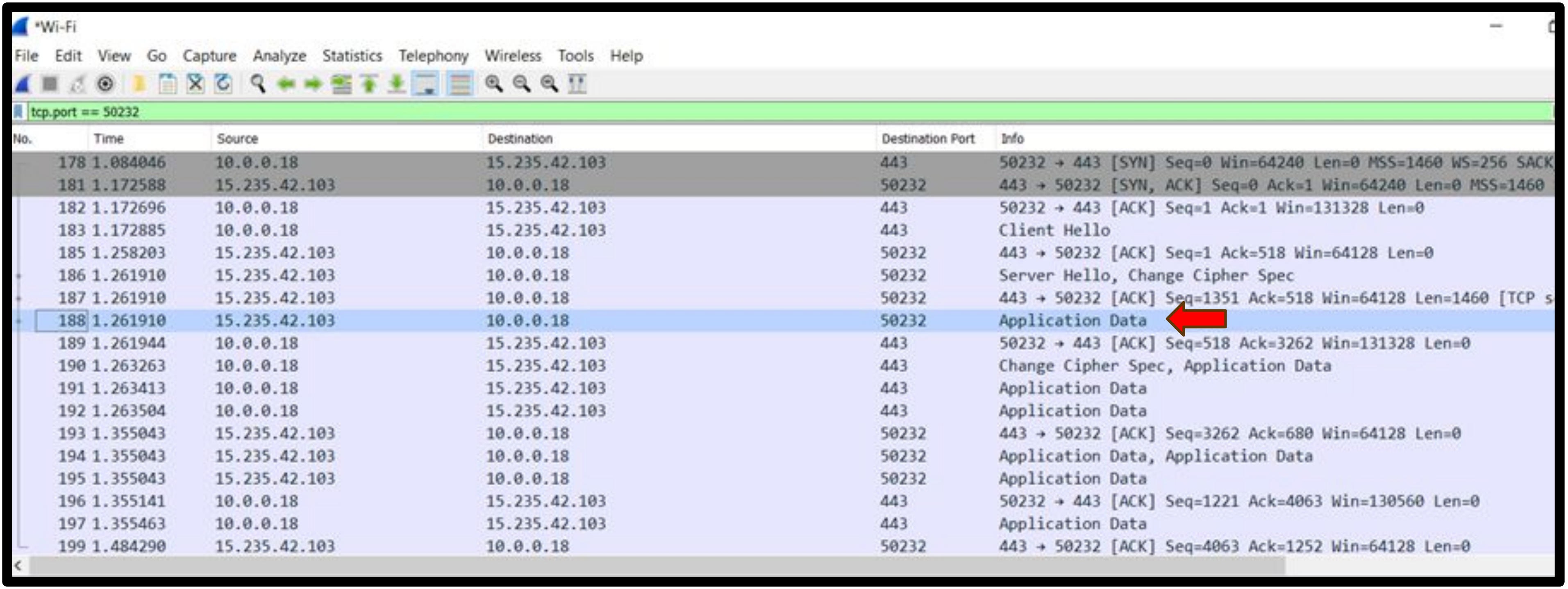

TLS1.3OrTLS1.2Trace.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.port == 50232

No.	Time	Source	Destination	Destination Port	Source Port	Info
183	1.172885	10.0.0.18	15.235.42.103	443	50232	Client Hello
185	1.258203	15.235.42.103	10.0.0.18	50232	443	443 → 50232 [ACK] Seq=1 Ack=518 Win=64128 Len=0
186	1.261910	15.235.42.103	10.0.0.18	50232	443	Server Hello, Change Cipher Spec
187	1.261910	15.235.42.103	10.0.0.18	50232	443	443 → 50232 [ACK] Seq=1351 Ack=518 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
188	1.261910	15.235.42.103	10.0.0.18	50232	443	Encrypted Extensions, Certificate, Certificate Verify, Finished
189	1.261944	10.0.0.18	15.235.42.103	443	50232	50232 → 443 [ACK] Seq=518 Ack=3262 Win=131328 Len=0
190	1.263263	10.0.0.18	15.235.42.103	443	50232	Change Cipher Spec, Finished

- TLS1.3 handshake
- Can see the handshake because of SSLKEYLOGFILE
- Otherwise more of the handshake is encrypted.
- Let’s look at next slide.

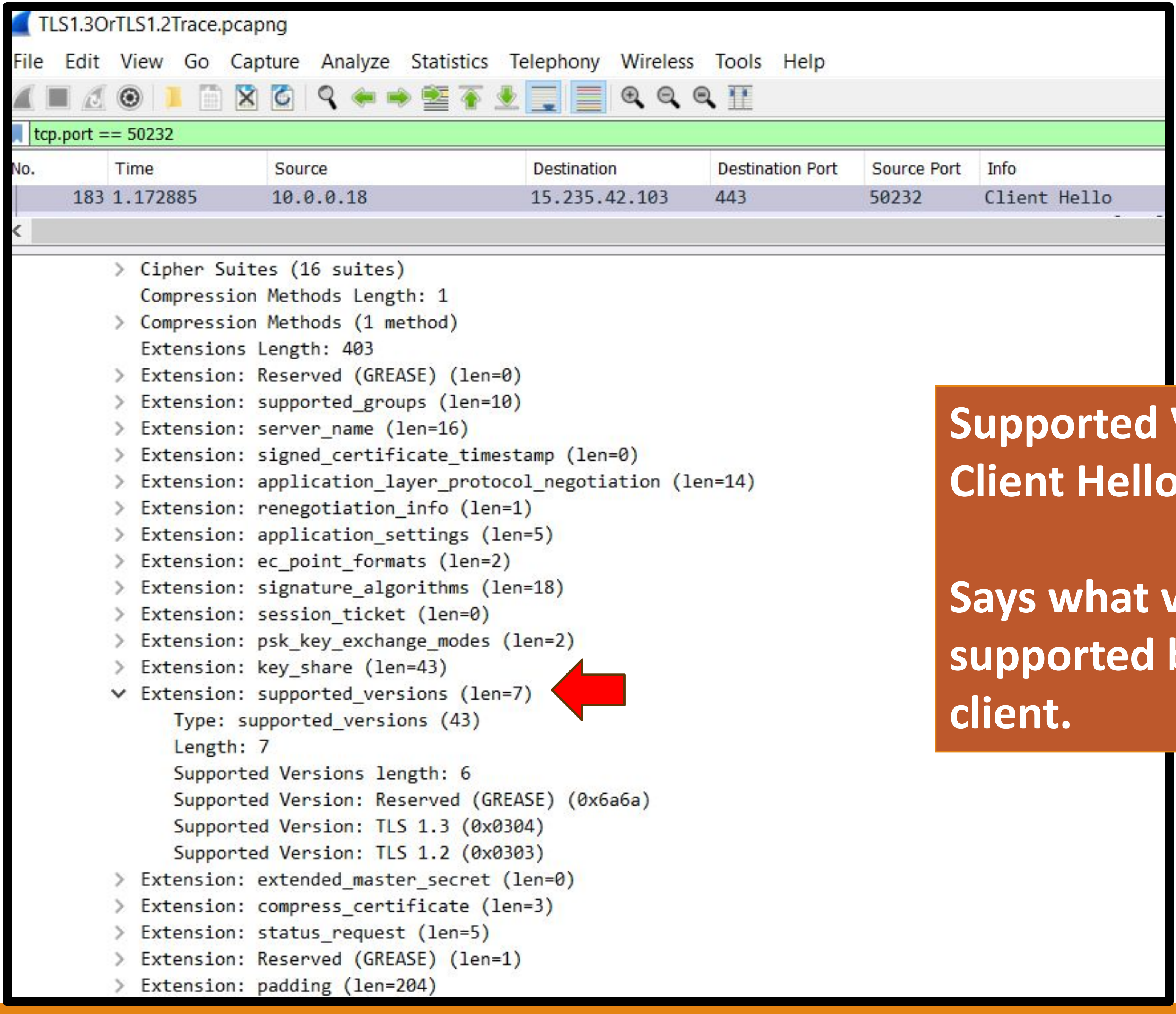
Same handshake w/out SSLKEYLOGFILE


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.port == 50232

No.	Time	Source	Destination	Destination Port	Info
178	1.084046	10.0.0.18	15.235.42.103	443	50232 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK
181	1.172588	15.235.42.103	10.0.0.18	50232	443 → 50232 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
182	1.172696	10.0.0.18	15.235.42.103	443	50232 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0
183	1.172885	10.0.0.18	15.235.42.103	443	Client Hello
185	1.258203	15.235.42.103	10.0.0.18	50232	443 → 50232 [ACK] Seq=1 Ack=518 Win=64128 Len=0
186	1.261910	15.235.42.103	10.0.0.18	50232	Server Hello, Change Cipher Spec
187	1.261910	15.235.42.103	10.0.0.18	50232	443 → 50232 [ACK] Seq=1351 Ack=518 Win=64128 Len=1460 [TCP s
188	1.261910	15.235.42.103	10.0.0.18	50232	Application Data
189	1.261944	10.0.0.18	15.235.42.103	443	50232 → 443 [ACK] Seq=518 Ack=3262 Win=131328 Len=0
190	1.263263	10.0.0.18	15.235.42.103	443	Change Cipher Spec, Application Data
191	1.263413	10.0.0.18	15.235.42.103	443	Application Data
192	1.263504	10.0.0.18	15.235.42.103	443	Application Data
193	1.355043	15.235.42.103	10.0.0.18	50232	443 → 50232 [ACK] Seq=3262 Ack=680 Win=64128 Len=0
194	1.355043	15.235.42.103	10.0.0.18	50232	Application Data, Application Data
195	1.355043	15.235.42.103	10.0.0.18	50232	Application Data
196	1.355141	10.0.0.18	15.235.42.103	443	50232 → 443 [ACK] Seq=1221 Ack=4063 Win=130560 Len=0
197	1.355463	10.0.0.18	15.235.42.103	443	Application Data
199	1.484290	15.235.42.103	10.0.0.18	50232	443 → 50232 [ACK] Seq=4063 Ack=1252 Win=64128 Len=0

- Notice only Client Hello and Server Hello are sent unencrypted.
- Application data packet before the 2nd Change Cipher Spec from the Client is handshake packet.
- Can’t see Server Certificate



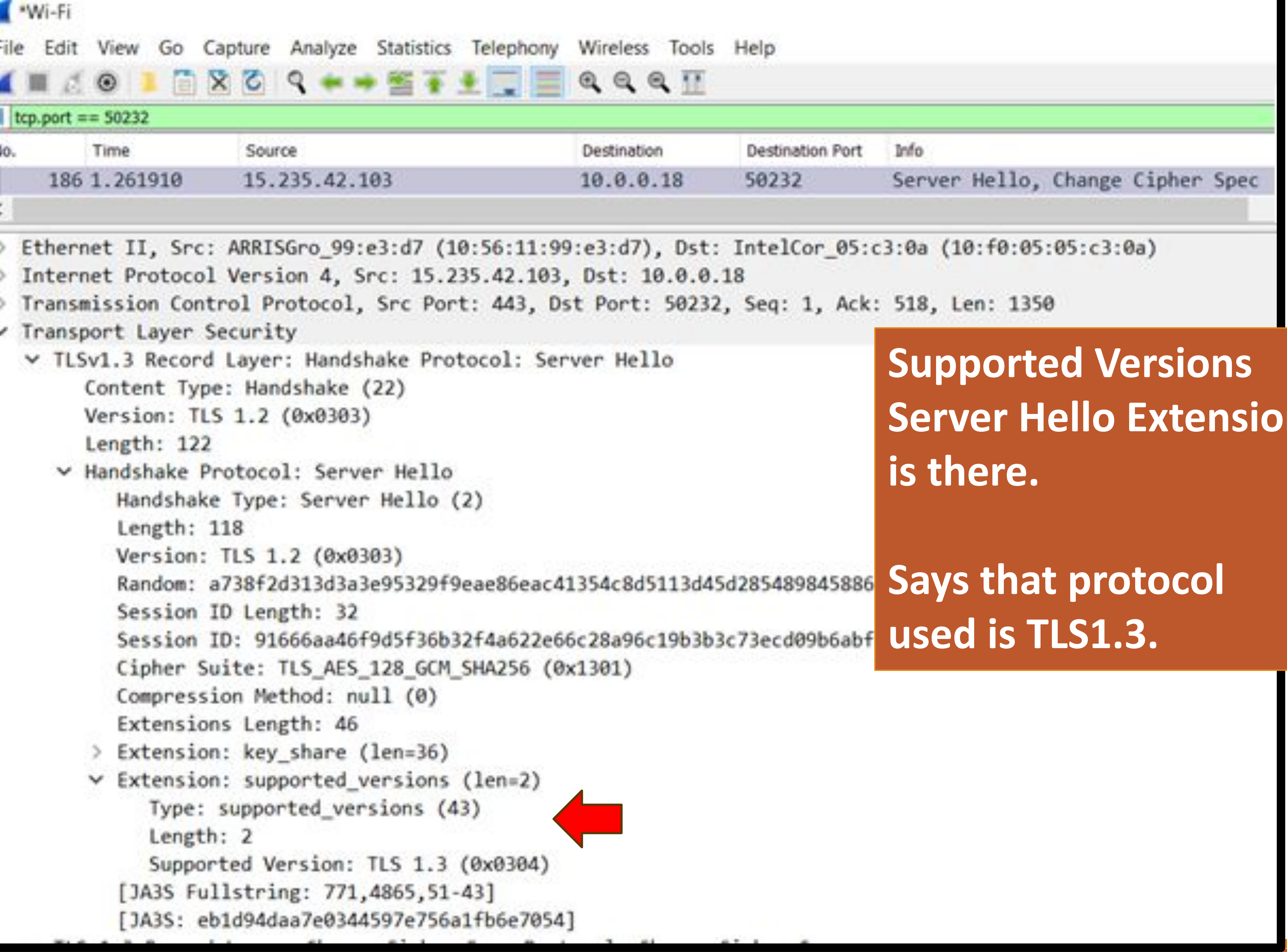
```
tcp.port == 50232
```

```
No.      Time       Source       Destination     Destination Port   Source Port   Info
    183 1.172885   10.0.0.18    15.235.42.103   443                50232         Client Hello
```

```
  > Cipher Suites (16 suites)
    Compression Methods Length: 1
  > Compression Methods (1 method)
    Extensions Length: 403
  > Extension: Reserved (GREASE) (len=0)
  > Extension: supported_groups (len=10)
  > Extension: server_name (len=16)
  > Extension: signed_certificate_timestamp (len=0)
  > Extension: application_layer_protocol_negotiation (len=14)
  > Extension: renegotiation_info (len=1)
  > Extension: application_settings (len=5)
  > Extension: ec_point_formats (len=2)
  > Extension: signature_algorithms (len=18)
  > Extension: session_ticket (len=0)
  > Extension: psk_key_exchange_modes (len=2)
  > Extension: key_share (len=43)
  v Extension: supported_versions (len=7)
        Type: supported_versions (43)
        Length: 7
        Supported Versions length: 6
        Supported Version: Reserved (GREASE) (0x6a6a)
        Supported Version: TLS 1.3 (0x0304)
        Supported Version: TLS 1.2 (0x0303)
  > Extension: extended_master_secret (len=0)
  > Extension: compress_certificate (len=3)
  > Extension: status_request (len=5)
  > Extension: Reserved (GREASE) (len=1)
  > Extension: padding (len=204)
```

Supported Versions Client Hello Extension.

Says what versions are supported by this client.


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

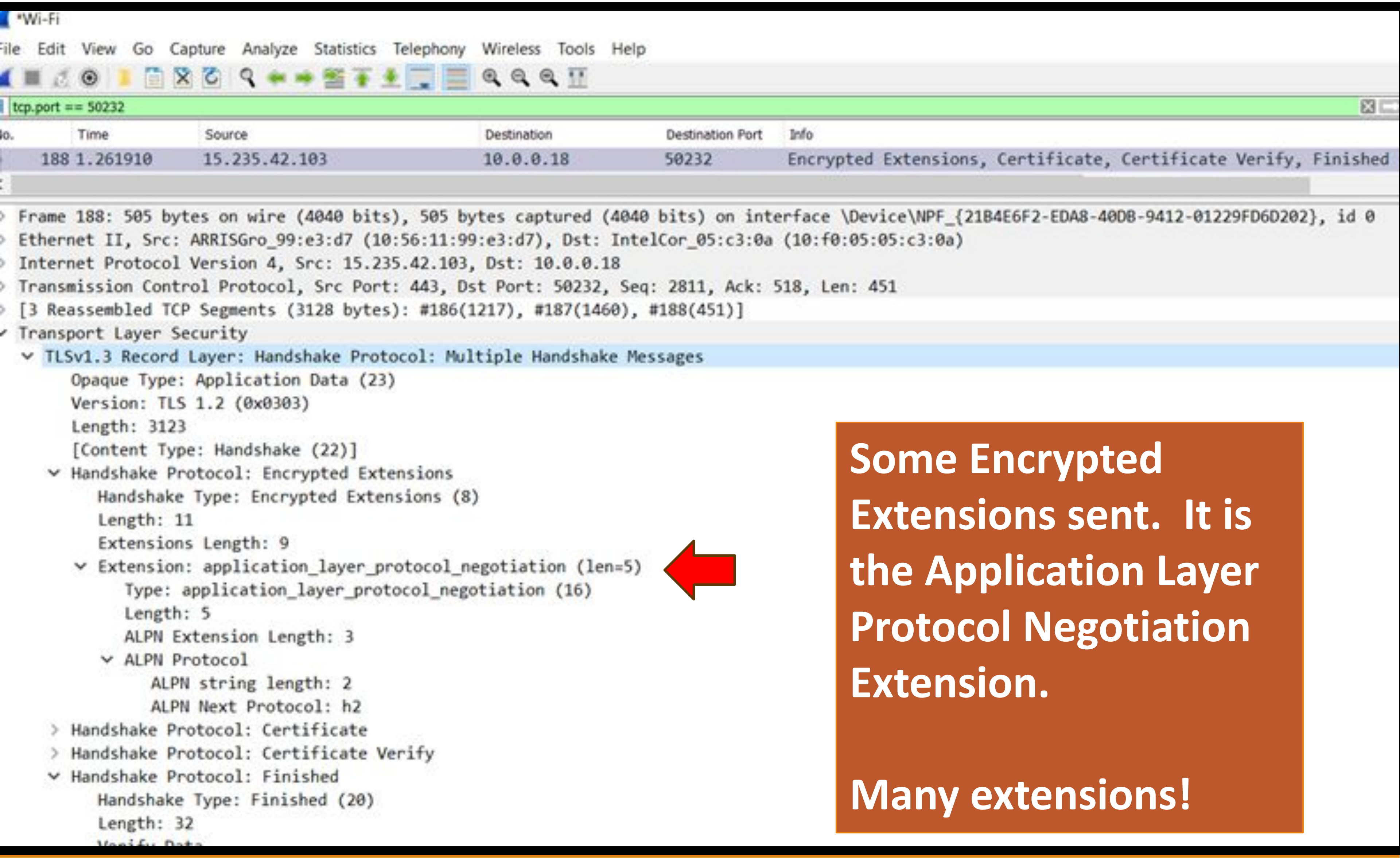
```
tcp.port == 50232

No.   Time        Source          Destination   Destination Port   Info
  186 1.261910    15.235.42.103   10.0.0.18     50232              Server Hello, Change Cipher Spec
```

```
> Ethernet II, Src: ARRISGro_99:e3:d7 (10:56:11:99:e3:d7), Dst: IntelCor_05:c3:0a (10:f0:05:05:c3:0a)
> Internet Protocol Version 4, Src: 15.235.42.103, Dst: 10.0.0.18
> Transmission Control Protocol, Src Port: 443, Dst Port: 50232, Seq: 1, Ack: 518, Len: 1350
v Transport Layer Security
  v TLSv1.3 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 122
    v Handshake Protocol: Server Hello
        Handshake Type: Server Hello (2)
        Length: 118
        Version: TLS 1.2 (0x0303)
        Random: a738f2d313d3a3e95329f9eae86eac41354c8d5113d45d285489845886
        Session ID Length: 32
        Session ID: 91666aa46f9d5f36b32f4a622e66c28a96c19b3b3c73ecd09b6abf
        Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
        Compression Method: null (0)
        Extensions Length: 46
      > Extension: key_share (len=36)
      v Extension: supported_versions (len=2)
          Type: supported_versions (43)
          Length: 2
          Supported Version: TLS 1.3 (0x0304)
        [JA3S Fullstring: 771,4865,51-43]
        [JA3S: eb1d94daa7e0344597e756a1fb6e7054]
```

Supported Versions Server Hello Extension is there.

Says that protocol used is TLS1.3.

w/SSLKEYLOGFILE: Encrypted Extensions


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tcp.port == 50232

No.   Time        Source          Destination   Destination Port   Info
188   1.261910    15.235.42.103   10.0.0.18     50232              Encrypted Extensions, Certificate, Certificate Verify, Finished

> Frame 188: 505 bytes on wire (4040 bits), 505 bytes captured (4040 bits) on interface \Device\NPF_{21B4E6F2-EDA8-40DB-9412-01229FD6D202}, id 0
> Ethernet II, Src: ARRISGro_99:e3:d7 (10:56:11:99:e3:d7), Dst: IntelCor_05:c3:0a (10:f0:05:05:c3:0a)
> Internet Protocol Version 4, Src: 15.235.42.103, Dst: 10.0.0.18
> Transmission Control Protocol, Src Port: 443, Dst Port: 50232, Seq: 2811, Ack: 518, Len: 451
> [3 Reassembled TCP Segments (3128 bytes): #186(1217), #187(1460), #188(451)]
v Transport Layer Security
  v TLSv1.3 Record Layer: Handshake Protocol: Multiple Handshake Messages
      Opaque Type: Application Data (23)
      Version: TLS 1.2 (0x0303)
      Length: 3123
      [Content Type: Handshake (22)]
    v Handshake Protocol: Encrypted Extensions
        Handshake Type: Encrypted Extensions (8)
        Length: 11
        Extensions Length: 9
      v Extension: application_layer_protocol_negotiation (len=5)
          Type: application_layer_protocol_negotiation (16)
          Length: 5
          ALPN Extension Length: 3
        v ALPN Protocol
            ALPN string length: 2
            ALPN Next Protocol: h2
    > Handshake Protocol: Certificate
    > Handshake Protocol: Certificate Verify
    v Handshake Protocol: Finished
        Handshake Type: Finished (20)
        Length: 32
        Verify Data
```

Some Encrypted Extensions sent. It is the Application Layer Protocol Negotiation Extension.

Many extensions!

TLS Encrypted Client Hello

Although TLS 1.3 [RFC8446] encrypts most of the handshake, including the server certificate, there are several ways in which an on-path attacker can learn private information about the connection. The plaintext Server Name Indication (SNI) extension in ClientHello messages, which leaks the target domain for a given connection, is perhaps the most sensitive, unencrypted information in TLS 1.3.

https://datatracker.ietf.org/doc/draft-ietf-tls-esni/

Take 2 points of view

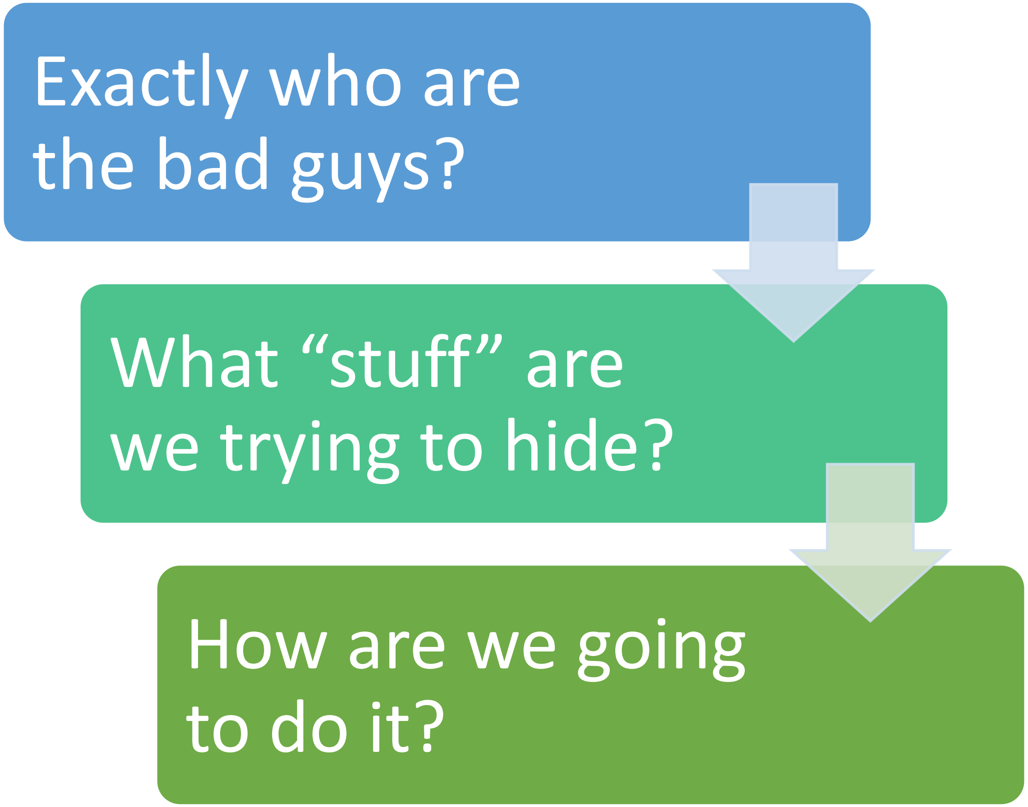
Privacy advocates

Large “brick-and-mortar” enterprises

What we agree on …

- 10,000 foot view of encryption
- Encryption is about hiding things from the bad guys

Where we start to get problems



Exactly who are the bad guys?

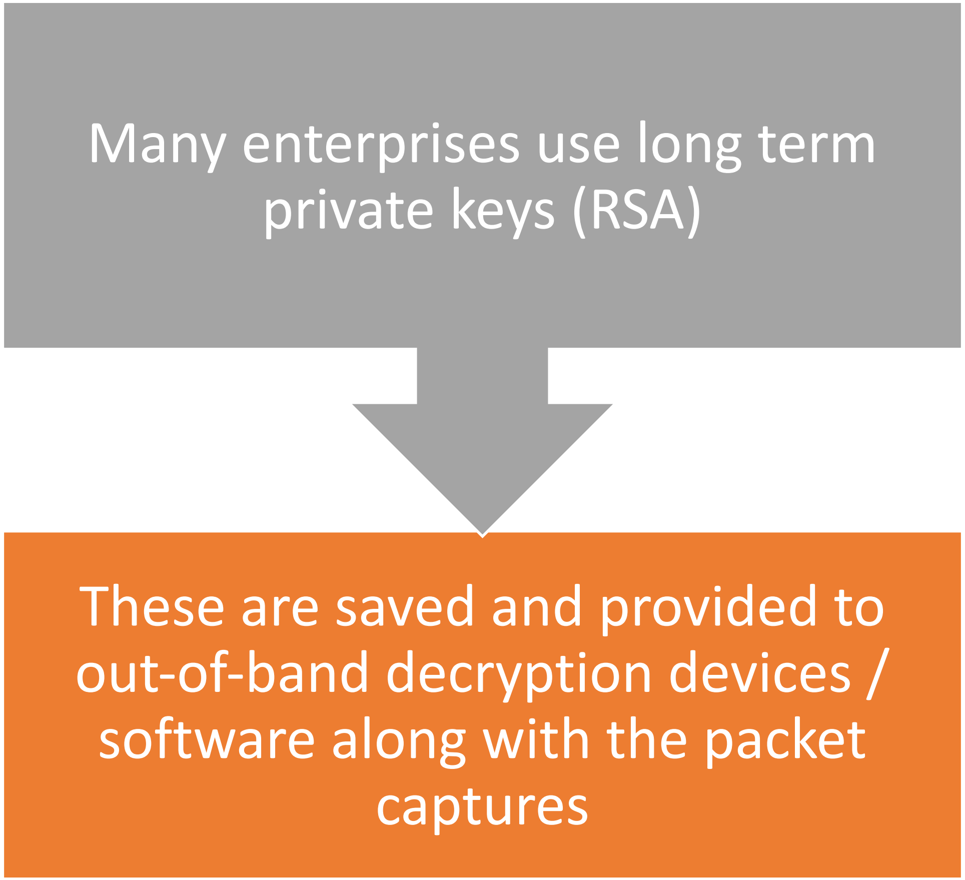
What “stuff” are we trying to hide?

How are we going to do it?

Take a one line change

Transport Layer Security (TLS) Protocol Version 1.3: draft-02 : Remove support for static RSA and DH key exchange.

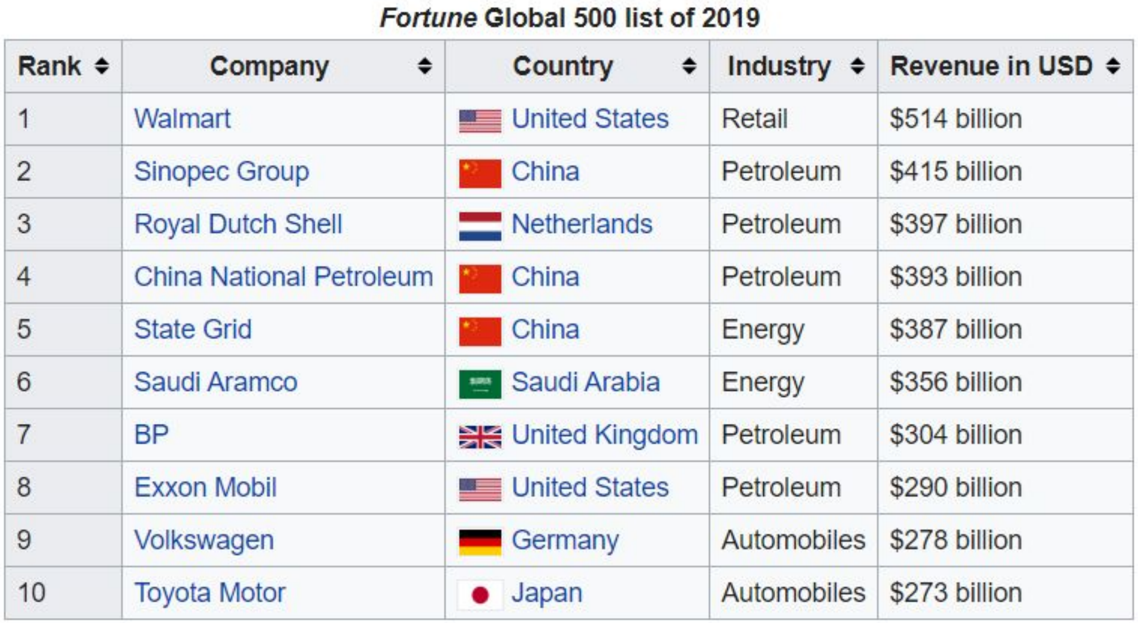
But the problem is the reality of enterprises today



Many enterprises use long term private keys (RSA)

These are saved and provided to out-of-band decryption devices / software along with the packet captures

Who are these "enterprises"



Fortune Global 500 list of 2019

Rank	Company	Country	Industry	Revenue in USD
1	Walmart	United States	Retail	$514 billion
2	Sinopec Group	China	Petroleum	$415 billion
3	Royal Dutch Shell	Netherlands	Petroleum	$397 billion
4	China National Petroleum	China	Petroleum	$393 billion
5	State Grid	China	Energy	$387 billion
6	Saudi Aramco	Saudi Arabia	Energy	$356 billion
7	BP	United Kingdom	Petroleum	$304 billion
8	Exxon Mobil	United States	Petroleum	$290 billion
9	Volkswagen	Germany	Automobiles	$278 billion
10	Toyota Motor	Japan	Automobiles	$273 billion

Also includes…

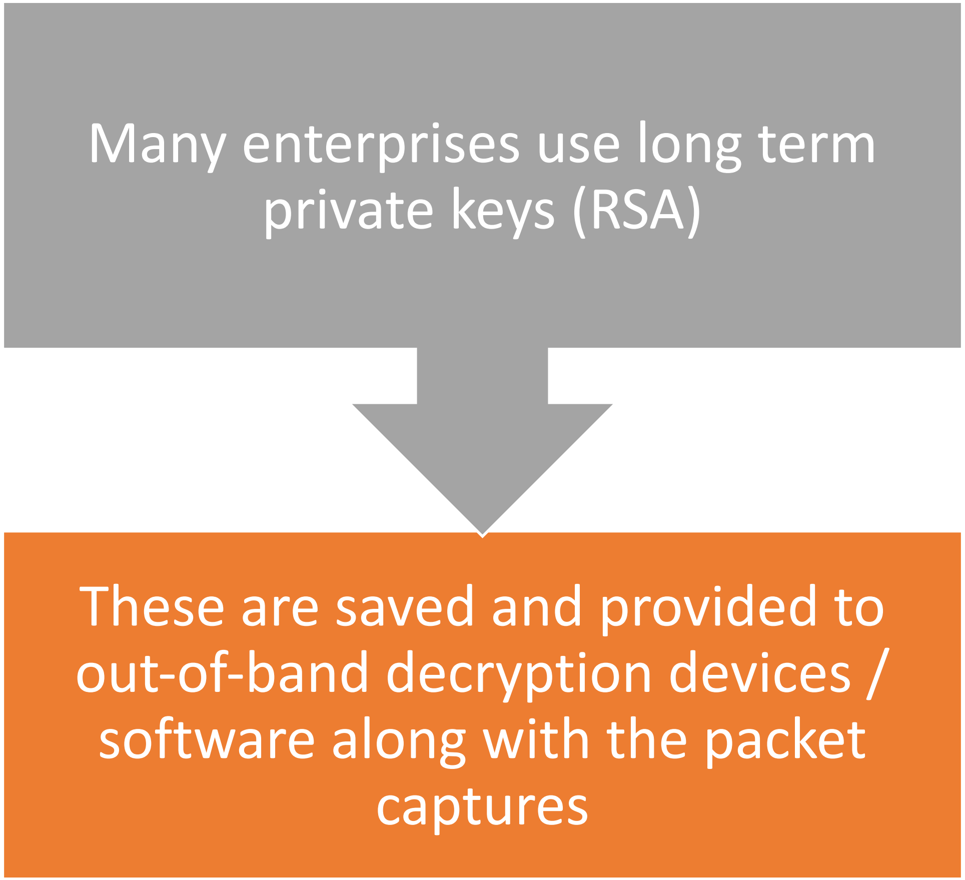
- Health care insurance in the US
- Government (federal, regional, local)
- Schools, hospitals, etc

How do they solve problems?

Packet decryption

DoD and Deep Packet Inspection (DPI)

How do enterprises do DPI today?



Many enterprises use long term private keys (RSA)

These are saved and provided to out-of-band decryption devices / software along with the packet captures

Deprecating Obsolete Key Exchange Methods in TLS1.2

- This document deprecates the use of RSA key exchange and Diffie Hellman over a finite field in TLS 1.2, and discourages the use of static elliptic curve Diffie Hellman cipher suites.
- Note that these prescriptions apply only to TLS 1.2 since TLS 1.0 and   1.1 are deprecated by [RFC8996] and TLS 1.3 either does not use the affected algorithm or does not share the relevant configuration options.

- https://datatracker.ietf.org/doc/draft-ietf-tls-deprecate-obsolete-kex/

Freeze TLS1.2

Abstract

TLS 1.2 is in widespread use and can be configured such that it provides good security properties. TLS 1.3 is also in widespread use and fixes some known deficiencies with TLS 1.2, such as removing error-prone cryptographic primitives and encrypting more of the traffic so that it is not readable by outsiders.

Both versions have several extension points, so items like new cryptographic algorithms, new supported groups (formerly "named curves"), etc., can be added without defining a new protocol. This document specifies that TLS 1.2 is frozen: no new algorithms or extensions will be approved.

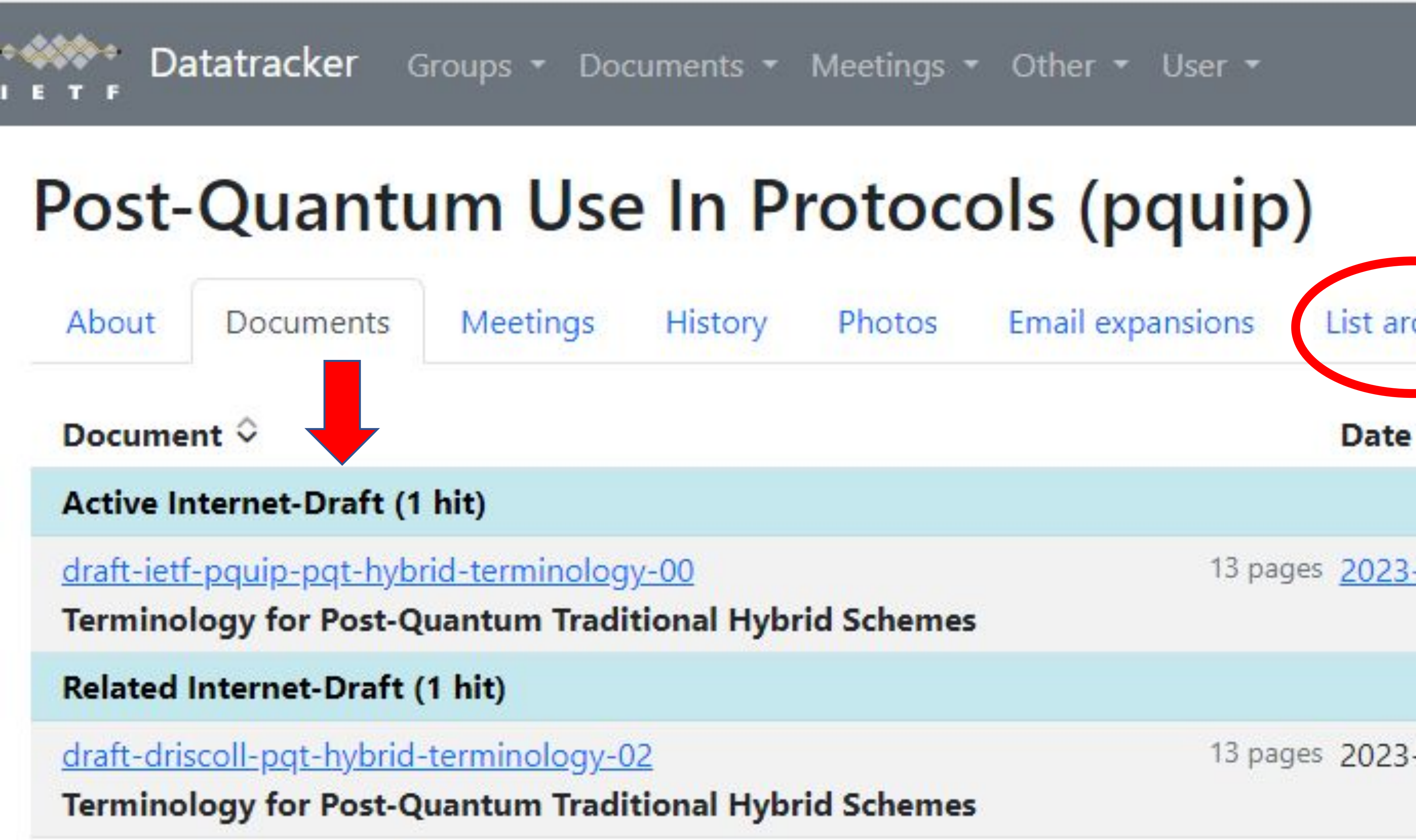
Further, TLS 1.3 use is widespread, and new protocols should require and assume its existence.

https://datatracker.ietf.org/doc/draft-rsalz-tls-tls12-frozen/

Hybrid Key Exchange in TLS1.3

- Hybrid key exchange refers to using multiple key exchange algorithms simultaneously and combining the result with the goal of providing security even if all but one of the component algorithms is broken.
- It is motivated by transition to post-quantum cryptography. This document provides a construction for hybrid key exchange in the Transport Layer Security (TLS) protocol version 1.3.

https://datatracker.ietf.org/doc/draft-ietf-tls-hybrid-design/



Post-Quantum Use In Protocols (pquip)

Document		Date
Active Internet-Draft (1 hit)		
draft-ietf-pquip-pqt-hybrid-terminology-00 **Terminology for Post-Quantum Traditional Hybrid Schemes**	13 pages	2023-
Related Internet-Draft (1 hit)		
draft-driscoll-pqt-hybrid-terminology-02 **Terminology for Post-Quantum Traditional Hybrid Schemes**	13 pages	2023-

Terminology for Post Quantum

One aspect of the transition to post-quantum algorithms in cryptographic protocols is the development of hybrid schemes that incorporate both post-quantum and traditional asymmetric algorithms. This document defines terminology for such schemes. It is intended to be used as a reference and, hopefully, to ensure consistency and clarity across different protocols, standards, and organisations.

https://datatracker.ietf.org/doc/draft-ietf-pquip-pqt-hybrid-terminology/

Changes for Post Quantum

- Need cipher suites
- Need TLS changes (key exchange)
- Need certificate / signing changes
- Need implementation in crypto libraries (OpenSSL, etc)
- Need changes to compilers (Java, C++, etc)
- Need changes to web servers (Apache), data base servers, etc.
- Need to change application programs

Questions?

Contact:

info@iiesoc.in

president@industrynetcouncil.org