



SharkFest '19 Europe



Plaintext Information in Network Protocols

Green Locks Are Not Enough

Simone Mainardi

mainardi@ntop.org



About me



- Simone Mainardi
- Engineer, PhD born in 1986
- Joined Luca Deri and ntop in late 2015
- Used to be a pure data scientist
- Now more close to a software developer





Agenda



- Introduction and motivation
 - Encrypted but not so encrypted
 - Secure but not so secure
- Plaintext information in network protocols
- Discussion and conclusion



Following Along



- Pcaps and docs available at
 - <https://bit.ly/328e5As>
- Screenshots shown during the presentation, look at the filename!

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.159	151.11.50.139	TCP	78	54902 → 4433 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1065084607 TSecr=0 ...
2	0.054020	151.11.50.139	192.168.1.159	TCP	74	4433 → 54902 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1400 SACK_PERM=1 TSval=14...
3	0.054120	192.168.1.159	151.11.50.139	TCP	66	54902 → 4433 [ACK] Seq=1 Ack=1 Win=131840 Len=0 TSval=1065084661 TSecr=14708486...



Encrypted Network Traffic



- Encryption is increasingly used in network protocols
- Fundamental to protect
 - Internet browsing
 - Online transactions
 - Instant messaging
 - Email
 - VoIP
 - ...



Cryptographic Protocols



- Cryptographic protocols necessary for the encryption of network communications
- Most popular is the Transport Layer Security (TLS)
- ~20 years since TLS 1.0

tools.ietf.org/html/rfc2246	January 1999
The TLS Protocol Version 1.0	
tools.ietf.org/html/rfc4346	April 2006
The Transport Layer Security (TLS) Protocol Version 1.1	
tools.ietf.org/html/rfc5246	August 2008
Category: Standards Track	
The Transport Layer Security (TLS) Protocol Version 1.2	
tools.ietf.org/html/rfc8446	August 2018
Obsoletes: 5077 , 5246 , 6961	
Updates: 5705 , 6066	
Category: Standards Track	
ISSN: 2070-1721	
The Transport Layer Security (TLS) Protocol Version 1.3	



What is the TLS



- Cryptographic protocol providing end-to-end communication security over the networks

← → ↻ datatracker.ietf.org/doc/rfc8446/

I E T F Datatracker Groups Documents Meetings Other User

Internet Engineering Task Force (IETF) E. Rescorla
Request for Comments: 8446 Mozilla
Obsoletes: 5077, 5246, 6961 August 2018
Updates: 5705, 6066
Category: Standards Track
ISSN: 2070-1721

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.



Where is the TLS



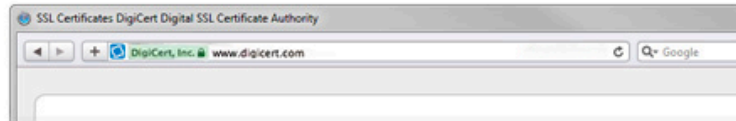
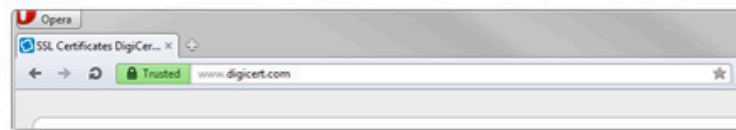
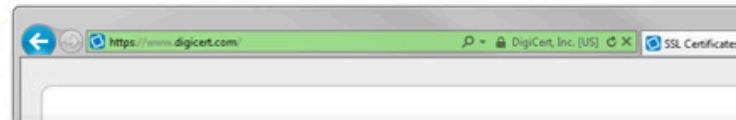
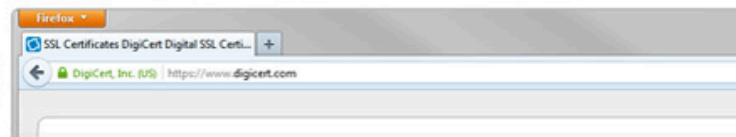
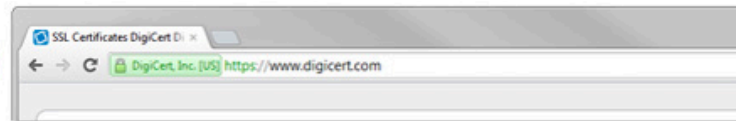
- Implemented in libraries and network applications
 - OpenVPN and other VPN tools
 - Quick UDP Internet Connections (QUIC)
 - Web Browsers (Chrome, FF, Opera, IE, ...)
 - Web Servers (Apache2, nginx, ...)
 - ...



TLS and HTTPS



- Probably everyone has experience with HTTPS
- HTTPS is HTTP transported over TLS
- Browsers and websites that use HTTPS are employing TLS encryption





Encryption Makes Us Feel Secure



- We feel secure when we know our traffic is encrypted
 - "No one can look at it!"
- We feel secure when we see the locks or a comfortable light-green while browsing the web
 - "It's something private just between me and the website!"
- But actually...



Fact #1: Encrypted \neq Secure [1/2]



- **Encrypted \neq Secure**
 - A secure communication must be encrypted
 - An encrypted communication is not necessarily secure
- Security depends on the cryptographic protocol (e.g., TLS), on its implementation (bugs?), on how cryptographic keys are managed, ...



Fact #1: Encrypted \neq Secure [2/2]



- Secure if...
 - ...the data being transferred is encrypted?
 - ...the parties exchanging information are who they claim to be?
 - ...the data has not been forged or tampered?
- TLS has vulnerabilities and is subject to attacks - as basically any other protocol



Fact #2: Plaintext Information in Encrypted Network Protocols [1/2]



- Cryptographic protocols or protocols that support encryption may carry certain **plaintext information**
- They will do that - almost surely - at least during the initial setup phase
 - Initial TLS handshake
 - Quick UDP Internet Connections (QUIC) or Google quick



Fact #2: Plaintext Information in Encrypted Protocols [2/2]



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.159	151.11.50.139	TCP	78	54902 → 4433 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1065084607 TSecr=0 ...
2	0.054020	151.11.50.139	192.168.1.159	TCP	74	4433 → 54902 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1400 SACK_PERM=1 TSval=14...
3	0.054120	192.168.1.159	151.11.50.139	TCP	66	54902 → 4433 [ACK] Seq=1 Ack=1 Win=131840 Len=0 TSval=1065084661 TSecr=14798486
4	0.056559	192.168.1.159	151.11.50.139	TLSv1...	583	Client Hello
5	0.112118	151.11.50.139	192.168.1.159	TCP	66	4433 → 54902 [ACK] Seq=1 Ack=518 Win=30080 Len=0 TSval=14798500 TSecr=1065084663
6	0.115210	151.11.50.139	192.168.1.159	TLSv1...	1454	Server Hello, Certificate
7	0.115215	151.11.50.139	192.168.1.159	TLSv1...	88	Server Key Exchange, Server Hello Done
8	0.115306	192.168.1.159	151.11.50.139	TCP	66	54902 → 4433 [ACK] Seq=518 Ack=1411 Win=130432 Len=0 TSval=1065084721 TSecr=14798...
9	0.117316	192.168.1.159	151.11.50.139	TLSv1...	192	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
10	0.169272	151.11.50.139	192.168.1.159	TLSv1...	117	Change Cipher Spec, Encrypted Handshake Message
11	0.169344	192.168.1.159	151.11.50.139	TCP	66	54902 → 4433 [ACK] Seq=644 Ack=1462 Win=131008 Len=0 TSval=1065084772 TSecr=14798...
12	0.172754	192.168.1.159	151.11.50.139	TCP	66	54902 → 4433 [ACK] Seq=644 Ack=1462 Win=131008 Len=0 TSval=1065084772 TSecr=14798...
13	0.225317	151.11.50.139	192.168.1.159	TCP	66	54902 → 4433 [ACK] Seq=644 Ack=1462 Win=131008 Len=0 TSval=1065084772 TSecr=14798...

```
Server Name list length: 21
Server Name Type: host_name (0)
Server Name length: 18
Server Name: ntopcloud.ntop.org
Extension: ec_point_formats (len=4)
Type: ec_point_formats (11)
Length: 4
EC point formats Length: 3
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.109	216.58.212.101	QUIC	1392	Client Hello, PKN: 1, CID: 16953050174146338482
2	0.046000	192.168.1.109	216.58.212.101	QUIC	478	Payload (Encrypted), PKN: 2, CID: 16953050174146338482
3	0.060057	216.58.212.101	192.168.1.109	QUIC	1392	Payload (Encrypted), PKN: 1

- ▶ Tag/value: PAD (Padding) (l=391)
- ▶ Tag/value: SNI (Server Name Indication) (l=15): mail.google.com
- ▶ Tag/value: STK (Source Address Token) (l=58)
- ▶ Tag/value: VER (Version) (l=4): 0024
- ▶ Tag/value: CCS (Common Certificate Sets) (l=16)
- ▶ Tag/value: NONC (Client Nonce) (l=32)
- ▶ Tag/value: MSPC (Max streams per connection) (l=4): 100
- ▶ Tag/value: AEAD (Authenticated encryption algorithms) (l=4), AES-GCM with a 12-byte tag and IV
- ▶ Tag/value: UAID (Client's User Agent ID) (l=24): beta Chrome/43.0.2357.45
- ▶ Tag/value: SCID (Server config ID) (l=16)



Fact #3: Plaintext Information in Network Protocols [1/2]



- Still a great deal of network protocols are **plaintext** or carry **plaintext information**
- Computers - and network protocols - have born and evolved when security was not an issue
 - Small, local networks (e.g., university labs) in which all the participants were trusted
 - Build something that 'just works'



Fact #3: Plaintext Information in Network Protocols [2/2]



- Even today when security is a main concern, certain network protocols didn't evolve in that sense
- Among the most common protocols which disseminate plaintext information there are
 - DHCP
 - DNS and mDNS
 - SSDP



This Talk is About...



- Fact #1: **Encrypted \neq Secure**
- Fact #2: Cryptographic protocols or protocols that support encryption may carry certain plaintext information
- Fact #3: Still a great deal of network protocols carry plaintext information



What is this Talk NOT About



- This talk is NOT about
 - Cryptographic protocols
 - TLS vulnerabilities / attacks / pitfalls
 - Network Encryption / Decryption
 - SSL Man-In-The-Middle



What is this Talk About



- This talk is about
 - Understanding how certain protocols disseminate plaintext information
 - Seeing which information is actually disseminated in plaintext
 - What it can be done to prevent it



Protocols Shown in this Talk



- Protocols
 - TLS
 - DNS
 - mDNS
 - DNS-SD
 - SSDP
 - DHCP
- ~10 minutes per protocol
 - Basic overview with real examples
 - No deep-dive



The Building Blocks of the TLS



- TLS actually consists of two protocols
- Only one actually carry encrypted application data
- TLS v 1.3, 1.2, 1.1, 1.0



tools.ietf.org/html/rfc8446

These properties should be true even in the face of an attacker who has complete control of the network, as described in [[RFC3552](#)]. See [Appendix E](#) for a more complete statement of the relevant security properties.

TLS consists of two primary components:

- A handshake protocol ([Section 4](#)) that authenticates the communicating parties, negotiates cryptographic modes and parameters, and establishes shared keying material. The handshake protocol is designed to resist tampering; an active attacker should not be able to force the peers to negotiate different parameters than they would if the connection were not under attack.
- A record protocol ([Section 5](#)) that uses the parameters established by the handshake protocol to protect traffic between the communicating peers. The record protocol divides traffic up into a series of records, each of which is independently protected using the traffic keys.



The TLS Handshake



- Before actually exchanging encrypted data, two parties willing to use TLS must perform an handshake
- Allows the server and client to
 - Authenticate each other
 - Negotiate an encryption algorithm and cryptographic keys
- Involves a series of back-and-forth packets between client and server



The TLS Handshake in Packets



- Shown TLS v 1.2, 1.3 fewer packets but still plaintext

sf_tls1.2.wikipedia.pcap

Apply a display filter ... <#/>

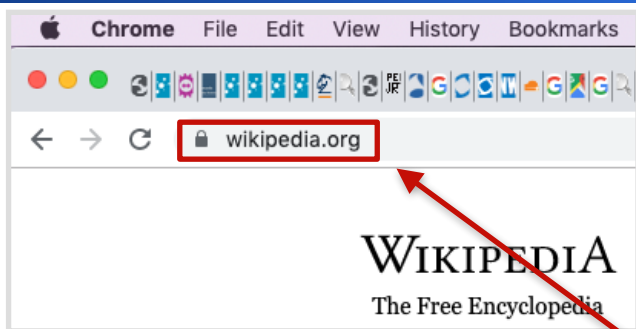
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.159	91.198.174.192	TCP	78	50241 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1003545409 TSecr=0 SACK_PERM=1
2	0.052954	91.198.174.192	192.168.1.159	TCP	74	443 → 50241 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1452 SACK_PERM=1 TSval=1494676261 TSecr=1003545409 WS=512
3	0.053037	192.168.1.159	91.198.174.192	TCP	66	50241 → 443 [ACK] Seq=1 Ack=1 Win=132480 Len=0 TSval=1003545461 TSecr=1494676261
4	0.053469	192.168.1.159	91.198.174.192	TLSv1.2	583	Client Hello
5	0.105248	91.198.174.192	192.168.1.159	TCP	66	443 → 50241 [ACK] Seq=1 Ack=518 Win=30208 Len=0 TSval=1494676274 TSecr=1003545461
6	0.106443	91.198.174.192	192.168.1.159	TLSv1.2	1506	Server Hello
7	0.106447	91.198.174.192	192.168.1.159	TCP	1506	443 → 50241 [ACK] Seq=1441 Ack=518 Win=30208 Len=1440 TSval=1494676274 TSecr=1003545461 [TCP segment of a reassembl...
8	0.106448	91.198.174.192	192.168.1.159	TLSv1.2	1282	Certificate [TCP segment of a reassembled PDU]
9	0.106552	91.198.174.192	192.168.1.159	TLSv1.2	1078	Certificate Status, Server Key Exchange, Server Hello Done
10	0.106566	192.168.1.159	91.198.174.192	TCP	66	50241 → 443 [ACK] Seq=518 Ack=2881 Win=129600 Len=0 TSval=1003545513 TSecr=1494676274
11	0.106566	192.168.1.159	91.198.174.192	TCP	66	50241 → 443 [ACK] Seq=518 Ack=4097 Win=128384 Len=0 TSval=1003545513 TSecr=1494676274
12	0.106626	192.168.1.159	91.198.174.192	TCP	66	50241 → 443 [ACK] Seq=518 Ack=5109 Win=130048 Len=0 TSval=1003545513 TSecr=1494676274
13	0.111389	192.168.1.159	91.198.174.192	TLSv1.2	151	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
14	0.111495	192.168.1.159	91.198.174.192	TLSv1.2	151	Application Data
15	0.111653	192.168.1.159	91.198.174.192	TLSv1.2	555	Application Data

Version: TLS 1.0 (0x0301)
Length: 512

Handshake Protocol: Client Hello
Handshake Type: Client Hello (1)
Length: 508
Version: TLS 1.2 (0x0303)
Random: 619ec107b2a4decca2952699e5bc5cce95839f79039c5ee3...
GMT Unix Time: Nov 24, 2021 23:47:35.000000000 CET



TLS Client Handshake: Example



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.159	91.198.174.192	TCP	78	50241 → 443 [SYN] Seq=0 Win=65535 Len=...
2	0.052954	91.198.174.192	192.168.1.159	TCP	74	443 → 50241 [SYN, ACK] Seq=0 Ack=1 Wi=...
3	0.053037	192.168.1.159	91.198.174.192	TCP	66	50241 → 443 [ACK] Seq=1 Ack=1 Win=132...
4	0.053469	192.168.1.159	91.198.174.192	TLSv1.2	583	Client Hello

```
    ▶ Cipher Suites (17 suites)
      Compression Methods Length: 1
    ▶ Compression Methods (1 method)
      Extensions Length: 401
    ▶ Extension: Reserved (GREASE) (len=0)
    ▼ Extension: server_name (len=22)
      Type: server_name (0)
      Length: 22
    ▼ Server Name Indication extension
      Server Name list length: 20
      Server Name Type: host_name (0)
      Server Name length: 17
      Server Name: www.wikipedia.org
```

```
0000 b8 27 eb 2b 90 f1 50 32 37 eb bb 40 08 00 45 00  '+'--+P2 7...@...E.
0010 02 39 00 00 40 00 40 06 6b f1 c0 a8 01 9f 5b c6  '9...@...k...[.
0020 ae c0 c4 41 01 bb 68 a1 9d 9d 75 8d 9f d3 80 18  '...A...h...W...|
0030 08 16 b4 4f 00 00 01 01 08 0a 3b d0 e3 75 59 16  '...0...;...;uY.
0040 f3 25 16 03 01 02 00 01 00 01 fc 03 03 61 9e c1  '%.....a...
0050 07 b2 a4 de cc a2 95 26 99 e5 bc 5c ce 95 83 9f  'y...^...i-<h...|
0060 79 03 9c 5e e3 98 0e 0b 69 b9 3c 68 7d 20 a4 7c  '4 d...W...K4
0070 e2 34 12 64 b0 eb aa b6 fb 57 8c bb c4 0a 4b 34  '...g...=y 3...
0080 82 a6 1e b0 2e 83 67 1a be a1 3d 79 ef 33 00 22  '...:...+ / , 0
0090 3a 3a 13 01 13 02 13 03 c0 2b c0 2f c0 2c c0 30  '...:/5
00a0 cc a9 cc a8 c0 13 c0 14 00 9c 00 9d 00 2f 00 35  '.....
00b0 00 0a 01 00 01 91 ba ba 00 00 00 00 00 16 00 14  '.....
00c0 00 00 11 77 77 77 2e 77 69 6b 69 70 65 64 69 61  'www.w ikipedia
00d0 2e 6f 72 67 00 17 00 00 ff 01 00 01 00 00 0a 00  '.org.....
```

- Open page <https://www.wikipedia.org>
- Host name is sent in plaintext, along with other information



TLS Server Handshake: Example



GlobalSign
GlobalSign Organization Validation CA - SHA256 - G2
*.wikipedia.org

***.wikipedia.org**
Issued by: GlobalSign Organization Validation CA - SHA256 - G2
Expires: Friday, 22 November 2019 at 08:59:59 Central European Standard Time
This certificate is valid

Details

Subject Name
Country or Region: US
County: California
Locality: San Francisco
Organisation: Wikimedia Foundation, Inc.
Common Name: *.wikipedia.org

Issuer Name
Country or Region: BE
Organisation: GlobalSign nv-sa
Common Name: GlobalSign Organization Validation CA - SHA256 - G2

OK

sf_tls1.2.wikipedia.pcap

Apply a display filter ... <#>

No.	Time	Source	Destination	Protocol	Length	Info
6	0.106443	91.198.174.192	192.168.1.159	TLSv1.2	1506	Server Hello
7	0.106447	91.198.174.192	192.168.1.159	TCP	1506	443 → 50241 [ACK] Seq=1441 Ack=518 Wi...
8	0.106448	91.198.174.192	192.168.1.159	TLSv1.2	1282	Certificate [TCP segment of a reassem...
9	0.106552	91.198.174.192	192.168.1.159	TLSv1.2	1078	Certificate Status, Server Key Exchan...

printableString: San Francisco
RDNSSequence item: 1 item (id-at-organizationName=Wikimedia Foundation, Inc.)
RelativeDistinguishedName item (id-at-organizationName=Wikimedia Foundation, Inc.)
Id: 2.5.4.10 (id-at-organizationName)
DirectoryString: printableString (1)
printableString: Wikimedia Foundation, Inc.
RDNSSequence item: 1 item (id-at-commonName=*.wikipedia.org)
RelativeDistinguishedName item (id-at-commonName=*.wikipedia.org)
Id: 2.5.4.3 (id-at-commonName)
DirectoryString: uTF8String (4)
uTF8String: *.wikipedia.org
subjectPublicKeyInfo

```
0120 63 2e 31 18 30 16 06 03 55 04 03 0c 0f 2a 2e 77 c.1.0...U...*.w
0130 69 6b 69 70 65 64 69 61 2e 6f 72 67 30 59 30 13 ikipedia.org0Y0
0140 06 07 2a 86 48 ce 3d 02 01 06 08 2a 86 48 ce 3d ...*H=...*H=
0150 03 01 07 03 42 00 04 67 75 ad 2e c6 6a e3 31 27 ...B.g u...j'1'
0160 5e 41 99 58 92 86 35 4c 8f 04 09 36 38 f0 f8 e5 ^A.X.5L...68...
0170 21 9c 86 aa 13 94 05 fe ae 9c fc b2 2f 56 1e 0d !.../...
0180 df 8e f7 6b b2 79 08 97 1f 9a 57 c2 ad 7b c3 b6 ...k.y...W...{...
0190 11 f3 69 93 44 9d e2 a3 82 05 95 30 82 05 91 30 ...i.D...0...0
01a0 0e 06 03 55 1d 0f 01 01 ff 04 04 03 02 03 88 30 ...U...0...0
01b0 81 a0 06 08 2b 06 01 05 05 07 01 01 04 81 93 30 ...+...0...0
01c0 81 90 30 4d 06 08 2b 06 01 05 05 07 30 02 86 41 ...0M...+...0...A
01d0 68 74 74 70 3a 2f 2f 73 65 63 75 72 65 2e 67 6c http://s.ecure.o
01e0 6f 62 61 6c 73 69 67 6e 2e 63 6f 6d 2f 63 61 63 obalsign.com/cac
```



TLS Handshake: Plaintext Information [1/2]



- Server Name Indication (SNI)
 - From the browser
 - Similar to the HTTP virtual hosts
- Cipher Suites
 - Sets of (more or less secure) algorithms to secure the communication

```
Server Name Indication extension
  Server Name list length: 20
  Server Name Type: host_name (0)
  Server Name length: 17
  Server Name: www.wikipedia.org
```

```
Cipher Suites (17 suites)
  Cipher Suite: Reserved (GREASE) (0x3a3a)
  Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
  Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
  Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
  Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0x1304)
  Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0x1305)
  Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0x1306)
  Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0x1307)
```



TLS Handshake: Plaintext Information [2/2]



- Server Certificate
 - Common Name
 - Alternative Names
 - Validity
 - Plaintext in TLS 1.2
 - Encrypted in TLS 1.3

```
▼ RDNSequence item: 1 item (id-at-commonName=*.wikipedia.org)
  ▼ RelativeDistinguishedName item (id-at-commonName=*.wikipedia.org)
    Id: 2.5.4.3 (id-at-commonName)
    ▼ DirectoryString: UTF8String (4)
      UTF8String: *.wikipedia.org
```

```
▼ Extension (id-ce-subjectAltName)
  Extension Id: 2.5.29.17 (id-ce-subjectAltName)
  ▼ GeneralNames: 39 items
    ▼ GeneralName: dNSName (2)
      dNSName: *.wikipedia.org
    ▼ GeneralName: dNSName (2)
      dNSName: wikimedia.org
    ▼ GeneralName: dNSName (2)
      dNSName: mediawiki.org
    ▼ GeneralName: dNSName (2)
      dNSName: wikibooks.org
```

```
▼ validity
  ▼ notBefore: utcTime (0)
    utcTime: 18-11-08 21:21:04 (UTC)
  ▼ notAfter: utcTime (0)
    utcTime: 19-11-22 07:59:59 (UTC)
```



How to Use TLS Handshake Data: SNI [1/2]



- SNI to profile users
 - *.facebook.com -> social media
 - *.bloomberg.com -> news
 - Services
 - SimilarWeb, Webshrinker, Symantec, Cyren
- Censorship in Korea

← → ↻ <https://www.wikipedia.com:443/>

Current categorization:

Reference
Last Time Rated/R

URL Category Check

Results for your request:
Full URL: <https://www.wikipedia.com>
Categories: **Education**
Alexa Rank: 1622878

[CHECK ANOTHER URL](#) [REPORT AS MISCLASSIFIED](#)

← → ↻ bleepingcomputer.com/news/security/south-korea-is-censoring-the-internet-by-snooping-on-sni-traffic/

South Korea is Censoring the Internet by Snooping on SNI Traffic

By [Sergiu Gatlan](#)

February 13, 2019 06:19 PM 1



How to Use TLS Handshake Data: SNI [2/2]



- SNI for HTTPS blocking / throttling
 - ntop's ntopng Edge
 - Trustwave's Web Filter
 - Sophos UTM
- SNI for Alerting
 - Suspicious or malicious host names

Past Alerts | Flow Alerts

Flow Alerts

10 ▾ | Type ▾ | Severity ▾

Date/Time ▾	Severity	Alert Type	Description	Actions
18:10:29	Warning	! Suspicious Activity	⚠ SSL Certificate Mismatch [Client Certificate: mydomain.es] [Server Certificate: mydomain.it] [Flow: 192.168.2.222:43794 ⇄ 194.247.56.15:443] [TCP] [Application: SSL] [Info: mydomain.es]	Explore Delete



How to Use TLS Handshake Data: Certificate and Cipher Suites



- Sever Certificate validity
- Cipher Suites to check if hosts in your network are using algorithms which are (deemed to be) secure
 - Entities maintain guidelines for TLS with regard to network security

← → ↻ csrc.nist.gov/publications/detail/sp/800-52/rev-2/final

SP 800-52 Rev. 2

Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations

Date Published: August 2019

Supersedes: SP 800-52 Rev. 1 (April 2014)

Author(s)
Kerry McKay (NIST), David Cooper (NIST)

DOCUMENTATION

Publication:
[SP 800-52 Rev. 2 \(DOI\)](#)
[Local Download](#)



How to Use TLS Handshake Data: Fingerprinting



- **Fingerprinting** to profile SSL/TLS Clients
 - Good, bad, expected, unexpected, unsecure
- A fingerprint (almost surely) identify a client
- **JA3** (<https://github.com/salesforce/ja3>)
 - Uses fields in the client hello

```
▼ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
Content Type: Handshake (22)
Version: TLS 1.0 (0x0301)
Length: 224
▼ Handshake Protocol: Client Hello
Handshake Type: Client Hello (1)
Length: 220
Version: TLS 1.2 (0x0303) ←
▶ Random
Session ID Length: 0
Cipher Suites Length: 38
▶ Cipher Suites (19 suites) ←
Compression Methods Length: 1
▶ Compression Methods (1 method)
Extensions Length: 141 ←
▶ Extension: server_name
▶ Extension: elliptic_curves ←
▶ Extension: ec_point_formats ←
▶ Extension: signature_algorithms
▶ Extension: next_protocol_negotiation
▶ Extension: Application Layer Protocol Negotiation
```

JA3 fingerprint for the standard Tor client:

e7d705a3286e19ea42f587b344ee6865

JA3 fingerprint for the Trickbot malware:

6734f37431670b3ab4292b8f60f29984

JA3 fingerprint for the Emotet malware:

4d7a28d6f2263ed61de88ca66eb011e3

← → ↻ raw.githubusercontent.com/salesforce/ja3/master/lists/osx-nix-ja3.csv

```
83e04bc58d402f9633983cbf22724b02,"Charles,Google Play Music Desktop Player,Pos
424008725394c634a4616b8b1f2828a5,"Charles,java,eclipse"
be9f1360cf52dc1f61ae025252f192a3,"Chromium"
def8761e4bcaaf91d99801a22ac6f6d4,"Chromium"
fc5cb0985a5f5e295163cc8ffff8a6e1,"Chromium"
e7d46c98b078477c4324031e0d3b22f5,"Cisco AnyConnect Secure Mobility Client"
ed36017db541879619c399c95e22067d,"Cisco AnyConnect Secure Mobility Client"
```



Protection Against TLS Handshake Eavesdroppers



- Encrypted SNI as an extension of TLS v 1.3
 - The server publishes a public key on a well-known DNS record
 - The client then replaces the plaintext SNI with an encrypted SNI, encrypted using a symmetric encryption key derived using the server's public key



The Domain Name System (DNS)



- System to map symbolic names to IP addresses
 - e.g., wikipedia.com -> 1.2.3.4
- Hierarchical and distributed architecture
- Defines the **DNS protocol**
- Ultra-long history
 - Tens of RFCs

The image shows two screenshots of web browser pages from tools.ietf.org. The top screenshot is for RFC 883, titled "DOMAIN NAMES - IMPLEMENTATION and SPECIFICATION". It includes metadata such as "Obsoluted by: 1034, 1035", "Updated by: 973", "Network Working Group", and "Request for Comments: 883". The author is listed as "P. Mockapetris" from "ISI" in "November 1983". The main text of the RFC states: "This memo discusses the implementation of domain name servers and resolvers, specifies the format of transactions, and discusses the use of domain names in the context of existing mail systems and other network software." The bottom screenshot is for RFC 882, titled "DOMAIN NAMES - CONCEPTS and FACILITIES". It has similar metadata: "Obsoluted by: 1034, 1035", "Updated by: 973", "Network Working Group", and "Request for Comments: 882". The author is "P. Mockapetris" from "ISI" in "November 1983". The main text states: "This RFC introduces domain style names, their use for ARPA Internet mail and host address support, and the protocols and servers used to implement domain name facilities."



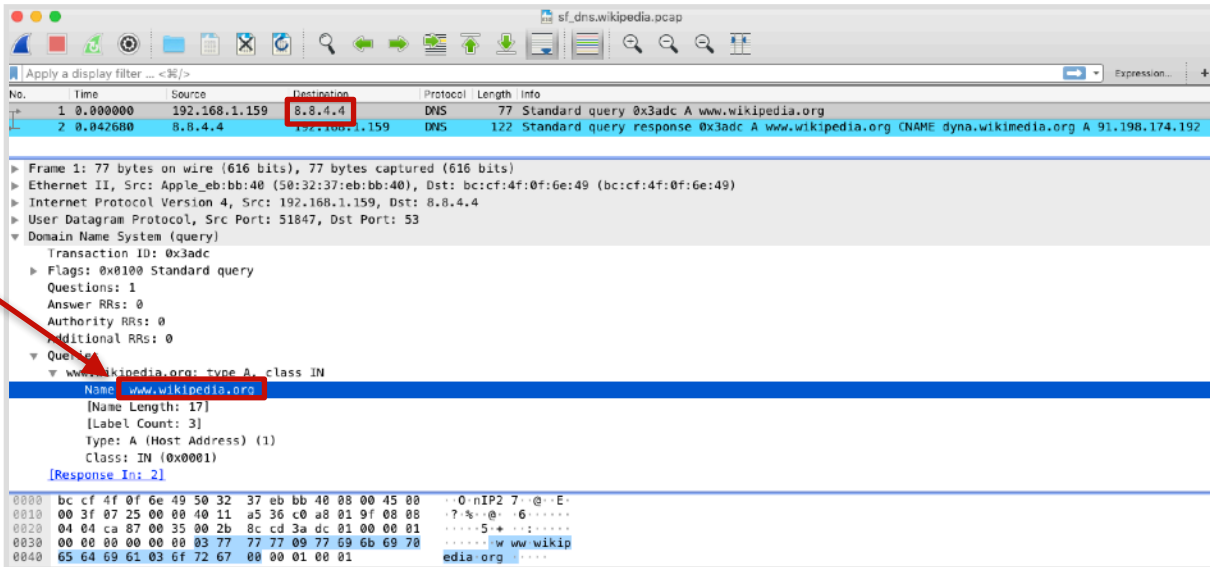
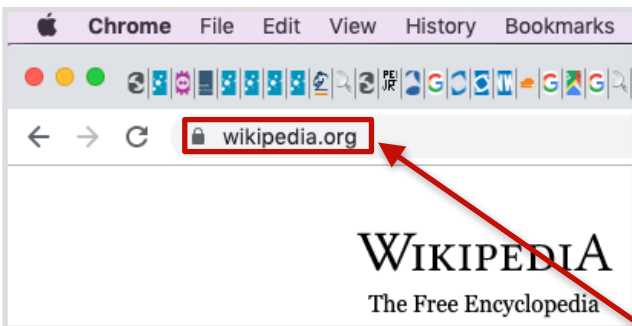
The DNS Protocol



- The DNS protocol (almost surely) kicks in every time a **name** is used to identify Internet resource
 - ping google.it
 - curl www.wikipedia.org
- To **resolve** a to an IP address, the **DNS client** queries a **DNS server**
 - The DNS client issues a **DNS query**
 - The DNS server responds with a **DNS query response**
- DNS server IP address is known to the DNS client



DNS Query: Example



- Open page <https://www.wikipedia.org>
- DNS query and query response before initiating the TLS connection



DNS Query Response: Example



sf_dns.wikipedia.pcap

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.159	8.8.4.4	DNS	77	Standard query 0x3adc A www.wikipedia.org
2	0.042680	8.8.4.4	192.168.1.159	DNS	122	Standard query response 0x3adc A www.wikipedia.org CNAME dyna.wikimedia.org A 91.198.174.192

Type: A (Host Address) (1)
Class: IN (0x0001)

Answers

- www.wikipedia.org: type CNAME, class IN, cname dyna.wikimedia.org
Name: www.wikipedia.org
Type: CNAME (Canonical NAME for an alias) (5)
Class: IN (0x0001)
Time to live: 13507
Data length: 17
CNAME: dyna.wikimedia.org
- dyna.wikimedia.org: type A, class IN, addr 91.198.174.192**

Name: dyna.wikimedia.org
Type: A (Host Address) (1)
Class: IN (0x0001)
Time to live: 118
Data length: 4
Address: 91.198.174.192

[Request In: 1]
[Time: 0.042680000 seconds]

```
0000 50 32 37 eb bb 40 bc cf 4f 0f 6e 49 08 00 45 80 P27...@... 0 nI...E...
0010 00 6c f1 bb 00 00 78 11 81 f2 08 08 04 04 c0 a8 .l...x...
0020 01 9f 00 35 ca 87 00 58 1b ba 3a dc 81 80 00 01 ...5...X...
0030 00 02 00 00 00 00 03 77 77 77 09 77 69 6b 69 70 .....w ww wikip
0040 65 64 69 61 03 6f 72 67 00 00 01 00 01 c0 0c 00 edia org .....
0050 05 00 01 00 00 34 c3 00 11 04 64 79 6e 61 09 77 .....4... dyna w
0060 69 6b 69 6d 65 64 69 61 c0 1a c0 2f 00 01 00 01 ikimedia /...
0070 00 00 00 76 00 04 5b c6 ae c0 .v...[...]
```

sf_tls1.2.wikipedia.pcap

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.159	91.198.174.192	TCP	78	50241 → 443 [SYN] Seq=0 Win=65535 Len...
2	0.052954	91.198.174.192	192.168.1.159	TCP	74	443 → 50241 [SYN, ACK] Seq=0 Ack=1 Wi...
3	0.053037	192.168.1.159	91.198.174.192	TCP	66	50241 → 443 [ACK] Seq=1 Ack=1 Win=132...
4	0.053469	192.168.1.159	91.198.174.192	TLSv1.2	583	Client Hello

- ▶ Cipher Suites (17 suites)
- ▶ Compression Methods Length: 1
- ▶ Compression Methods (1 method)
- ▶ Extensions Length: 401
- ▶ Extension: Reserved (GREASE) (len=0)
- ▶ Extension: server_name (len=22)
 - Type: server_name (0)
 - Length: 22
 - Server Name Indication extension
 - Server Name list length: 20
 - Server Name Type: host_name (0)
 - Server Name length: 17
 - Server Name: www.wikipedia.org**

```
0000 b8 27 eb 2b 90 f1 50 32 37 eb bb 40 08 00 45 00 ...+...P2 7...@...E...
0010 02 39 00 00 40 00 40 06 6b f1 c0 a8 01 9f 5b c6 ...9...@...k...f...
```



DNS Protocol: Plaintext Information



- All the resolved names are plaintext
 - Even if all the subsequent communications are encrypted



How to use DNS Data



- Names to passively profile users similar to what has been seen with the TLS SNI
- The ISP or even a Free-Wifi bar can easily get their hands into the DNS traffic



DNS Queries Interception [1/2]



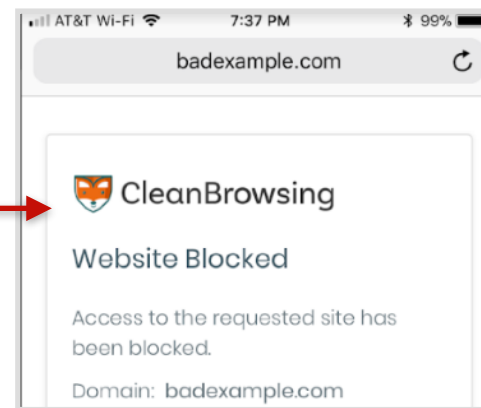
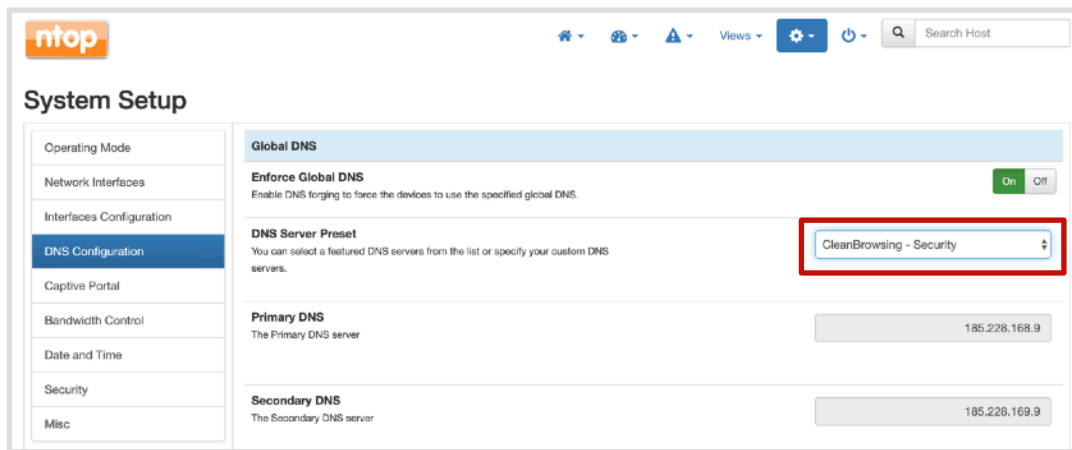
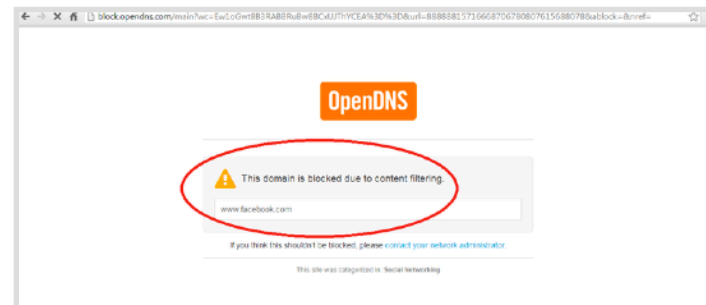
- As there is no encryption / authentication, queries can be intercepted
 - Transparently redirect the DNS queries to a DNS server chosen by the ISP (or an attacker)
 - The DNS server can respond with arbitrary IP addresses
- Interceptions can be made for various purposes
 - Censorship
 - Displaying ads
 - Collecting statistics
 - Blocking malware
- The point is that they are not authorized by users and are difficult to spot



DNS Queries Interception [2/2]



- DNS-based content filtering
 - OpenDNS, CleanBrowsing DNS and other services





Protection Against DNS Eavesdroppers: DoH



- DNS over HTTPS (DoH)
- TCP port 443
- Third-party observers can't look at DNS requests
- Supported by recent browsers (FF, Chrome)

The screenshot shows a web browser window with the address bar displaying `tools.ietf.org/html/rfc8484`. Below the address bar, there are navigation links: `[Docs]`, `[txt|pdf]`, `[draft-ietf-doh-...]`, `[Tracker]`, `[Diff1]`, `[Diff2]`, and `[Errata]`. The main content area displays the following information:

PROPOSED STANDARD
Errata Exist

Internet Engineering Task Force (IETF) P. Hoffman
Request for Comments: 8484 ICANN
Category: Standards Track P. McManus
ISSN: 2070-1721 Mozilla
October 2018

DNS Queries over HTTPS (DoH)

Abstract

This document defines a protocol for sending DNS queries and getting DNS responses over HTTPS. Each DNS query-response pair is mapped into an HTTP exchange.



Protection Against DNS Eavesdroppers: Firefox DoH



Firefox 1.1.1.1 — the internet's Fastest, X +
https://1.1.1.1/help

1.1.1.1

Connection Information

Please include this URL when you create a post in the community forum.

`https://1.1.1.1/#!/help`

Click to copy

Debug Information

Connected to 1.1.1.1	Yes
Using DNS over HTTPS (DoH)	Yes

Enable DNS over HTTPS
Use Provider: Cloudflare (Default)

Help Cancel OK

ntop Active DNSOverHTTPS Flows

Application	Protocol	Client	Server	Duration	Breakdown	Actual Thpt	Total Bytes	Info
TLS.DNSOverHTTPS	TCP	172.20.10.4	mozilla.cloudflare-dns.c...	01:59	Client: Server	0 bit/s	21.77 KB	mozilla.cloudflare-dns.c...
DNS.DNSOverHTTPS	UDP	172.20.10.4	1.1.1.1 :domain	< 1 sec	Client: Server	0 bit/s	204 Bytes	mozilla.cloudflare-dns.c...

sf_dns_doh.pcap

Apply a display filter ... < %>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.20.10.4	104.16.248.249	TCP	78	49877 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=496349789 TSecr=0 SACK_P...
2	0.087116	104.16.248.249	172.20.10.4	TCP	66	443 → 49877 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1300 SACK_PERM=1 WS=1024
3	0.087208	172.20.10.4	104.16.248.249	TCP	54	49877 → 443 [ACK] Seq=1 Ack=1 Win=262144 Len=0
4	0.089016	172.20.10.4	104.16.248.249	TLSv1.3	571	Client Hello
5	0.179334	104.16.248.249	172.20.10.4	TCP	54	443 → 49877 [ACK] Seq=1 Ack=518 Win=30720 Len=0
6	0.179339	104.16.248.249	172.20.10.4	TLSv1.3	1354	Server Hello, Change Cipher Spec
7	0.179341	104.16.248.249	172.20.10.4	TCP	1354	443 → 49877 [ACK] Seq=1301 Ack=518 Win=30720 Len=1300 [TCP segment of a reassembled P...
8	0.179442	172.20.10.4	104.16.248.249	TCP	54	49877 → 443 [ACK] Seq=518 Ack=2601 Win=259520 Len=0
9	0.179953	172.20.10.4	104.16.248.249	TCP	54	[TCP Window Update] 49877 → 443 [ACK] Seq=518 Ack=2601 Win=262144 Len=0
10	0.180826	104.16.248.249	172.20.10.4	TLSv1.3	503	Application Data
11	0.180883	172.20.10.4	104.16.248.249	TCP	54	49877 → 443 [ACK] Seq=518 Ack=3050 Win=261632 Len=0
12	0.206957	172.20.10.4	104.16.248.249	TLSv1.3	118	Change Cipher Spec, Application Data
13	0.207815	172.20.10.4	104.16.248.249	TLSv1.3	224	Application Data
14	0.207925	172.20.10.4	104.16.248.249	TLSv1.3	297	Application Data
15	0.208016	172.20.10.4	104.16.248.249	TLSv1.3	133	Application Data
16	0.283559	104.16.248.249	172.20.10.4	TCP	54	443 → 49877 [ACK] Seq=3050 Ack=1074 Win=32768 Len=0



Protection Against DNS Eavesdroppers: DoT



- DNS over TLS (DoT)
- TCP port **853**
- System-wide
- Linux: systemd-resolved
(systemd >= 239)
- Linux/Win/OS X: DNS Privacy Daemon -
stubby

→ ↻ 🔒 tools.ietf.org/html/rfc7858

[Docs] [txt|pdf] [draft-ietf-dpri...] [Tracker] [Diff1] [Diff2] [Errata]

Updated by: [8310](#) PROPOSED STANDARD
Errata Exist

Internet Engineering Task Force (IETF)
Request for Comments: 7858
Category: Standards Track
ISSN: 2070-1721

Z. Hu
L. Zhu
J. Heidemann
USC/ISI
A. Mankin
Independent
D. Wessels
Verisign Labs
P. Hoffman
ICANN
May 2016

Specification for DNS over Transport Layer Security (TLS)

Abstract

This document describes the use of Transport Layer Security (TLS) to provide privacy for DNS. Encryption provided by TLS eliminates opportunities for eavesdropping and on-path tampering with DNS queries in the network, such as discussed in [RFC 7626](#). In addition, this document specifies two usage profiles for DNS over TLS and provides advice on performance considerations to minimize overhead from using TCP and TLS with DNS.



Protection Against DNS Eavesdroppers: systemd-resolved DoT



sf_dns_dot.pcap

Apply a display filter ... <=>/> Expression...

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.185	8.8.8.8	TCP	74	58290 → 853 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=707761427 TSecr=0 ...
2	0.034926	8.8.8.8	192.168.1.185	TCP	74	853 → 58290 [SYN, ACK] Seq=0 Ack=1 Win=60192 Len=0 MSS=1380 SACK_PERM=1 TSval=2386231...
3	0.034971	192.168.1.185	8.8.8.8	TCP	66	58290 → 853 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=707761462 TSecr=2386231312
4	0.035180	192.168.1.185	8.8.8.8	TLSv1.2	264	Client Hello
5	0.067922	8.8.8.8	192.168.1.185	TCP	66	853 → 58290 [ACK] Seq=1 Ack=199 Win=61440 Len=0 TSval=2386231346 TSecr=707761462
6	0.085177	8.8.8.8	192.168.1.185	TLSv1.2	3135	Server Hello, Certificate, Server Key Exchange, Server Hello Done
7	0.085210	192.168.1.185	8.8.8.8	TCP	66	58290 → 853 [ACK] Seq=199 Ack=3070 Win=62592 Len=0 TSval=707761512 TSecr=2386231362
8	0.086210	192.168.1.185	8.8.8.8	TLSv1.2	151	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
9	0.086307	192.168.1.185	8.8.8.8	TLSv1.2	89	Application Data
10	0.086419	192.168.1.185	8.8.8.8	TLSv1.2	128	Application Data
11	0.117127	8.8.8.8	192.168.1.185	TLSv1.2	342	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
12	0.117158	8.8.8.8	192.168.1.185	TCP	66	853 → 58290 [ACK] Seq=3346 Ack=369 Win=61440 Len=0 TSval=2386231395 TSecr=707761513
13	0.128189	8.8.8.8	192.168.1.185	TLSv1.2	178	Application Data
14	0.128316	192.168.1.185	8.8.8.8	TCP	66	58290 → 853 [ACK] Seq=199 Ack=3070 Win=62592 Len=0 TSval=707761512 TSecr=2386231362
15	1.288536	192.168.1.185	8.8.8.8	TLSv1.2	89	Application Data
16	1.288620	192.168.1.185	8.8.8.8	TLSv1.2	133	Application Data
17	1.320299	8.8.8.8	192.168.1.185	TCP	66	853 → 58290 [ACK] Seq=3346 Ack=369 Win=61440 Len=0 TSval=2386231395 TSecr=707761513
18	1.331239	8.8.8.8	192.168.1.185	TLSv1.2	229	Application Data
19	1.373326	192.168.1.185	8.8.8.8	TCP	66	58290 → 853 [ACK] Seq=199 Ack=3070 Win=62592 Len=0 TSval=707761512 TSecr=2386231362
20	2.970121	192.168.1.185	8.8.8.8	TLSv1.2	89	Application Data
21	2.970205	192.168.1.185	8.8.8.8	TLSv1.2	133	Application Data
22	3.001785	192.168.1.185	8.8.8.8	TCP	66	853 → 58290 [ACK] Seq=3346 Ack=369 Win=61440 Len=0 TSval=2386231395 TSecr=707761513
23	3.011624	8.8.8.8	192.168.1.185	TLSv1.2	167	Application Data

```
ubuntu@ubuntu:~$ systemd --version
systemd 240
+PAM +AUDIT +SELINUX +IMA +APPARMOR +SMACK +SYSVINIT +UTMP +LIBCRYPTSETUP +GCRY
PT +GNUTLS +ACL +XZ +LZ4 +SECCOMP +BLKID +ELFUTILS +KMOD -IDN2 +IDN -PCRE2 defa
ult-hierarchy=hybrid
ubuntu@ubuntu:~$ cat /etc/systemd/resolved.conf | grep -v \#
[Resolve]
DNS=8.8.8.8
Domains=~.
DNSOverTLS=opportunistic
ubuntu@ubuntu:~$ systemctl restart systemd-resolved
```



The multicast DNS (mDNS)



- Resolve host names to IP addresses in (small) networks
- No need for a DNS server
- IP UDP multicast packets
- Only resolves host names ending with **.local**

tools.ietf.org/html/rfc6762

ISSN: 2070-1721 February 2013

Multicast DNS

Abstract

As networked devices become smaller, more portable, and more ubiquitous, the ability to operate with less configured infrastructure is increasingly important. In particular, the ability to look up DNS resource record data types (including, but not limited to, host names) in the absence of a conventional managed DNS server is useful.

```
simone@devel:~$ avahi-resolve-host-name -4 Simones-MacBook-Pro.local
Simones-MacBook-Pro.local      192.168.2.126
simone@devel:~$ ping -c1 Simones-MacBook-Pro.local
PING Simones-MacBook-Pro.local (192.168.2.126) 56(84) bytes of data:
64 bytes from 192.168.2.126: icmp_seq=1 ttl=64 time=1.00 ms

--- Simones-MacBook-Pro.local ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.009/1.009/1.009/0.000 ms
simone@devel:~$
```



mDNS Implementations



- Apple
 - Bonjour (mDNSResponder)
- Linux & BSDs
 - Avahi (avahi-daemon)
 - systemd-resolved
- Windows
 - Bonjour for Windows (mDNSResponder.exe)
 - Link-local Multicast Name Resolution (LLMNR) - not actually mDNS but similar



mDNS Query: Example



```
simone@devel:~$ avahi-resolve-host-name -4 Simones-MacBook-Pro.local
Simones-MacBook-Pro.local      192.168.2.126
```

- Query sent to 224.0.0.251
- UDP with src/dst ports 5353
- Name is carried in plaintext in a standard DNS packet

sf_mdns_avahi_name_lookup.pcap

Apply a display filter ... <=>/> Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.2.222	224.0.0.251	MDNS	85	Standard query 0x0000 A Simones-MacBook-Pro.local, "QM" question
2	0.000198	192.168.2.126	224.0.0.251	MDNS	143	Standard query response 0x0000 A, cache flush 192.168.2.126 AAAA, cache flush fe80::bb:af48:f82e:a77a NSEC,...
3	0.000275	fe80::bb:af48:f82e:a77a	ff02::fb	MDNS	163	Standard query response 0x0000 A, cache flush 192.168.2.126 AAAA, cache flush fe80::bb:af48:f82e:a77a NSEC,...

► Ethernet II, Src: SuperMic_d4:cc:f9 (00:25:90:d4:cc:f9), Dst: IPv4mcast_fb (01:00:5e:00:00:fb)

► Internet Protocol Version 4, Src: 192.168.2.222, Dst: 224.0.0.251

► User Datagram Protocol, Src Port: 5353, Dst Port: 5353

▼ Multicast Domain Name System (query)

Transaction ID: 0x0000

► Flags: 0x0000 Standard query

Questions: 1

Answer RRs: 0

Authority RRs: 0

Additional RRs: 0

▼ Queries

▼ Simones-MacBook-Pro.local: type A, class IN, "QM" question

Name: Simones-MacBook-Pro.local

[Name Length: 25]

```
0000  01 00 5e 00 00 fb 00 25 90 d4 cc f9 08 00 45 00  ..^...%.....E
0010  00 47 67 d3 40 00 ff 11 6f 50 c0 a8 02 de e0 00  ..Gg@...oP....
0020  00 fb 14 e9 14 e9 00 33 27 0a 00 00 00 00 01    .....3.....
0030  00 00 00 00 00 00 13 53 69 6d 6f 6e 65 73 2d 4d  .....S imones-M
0040  61 63 42 6f 6f 6b 2d 50 72 6f 05 6c 6f 63 61 6c  acBook-P ro local
0050  00 00 01 00 01  .....
```



mDNS Query Response: Example



```
simone@devel:~$ avahi-resolve-host-name -4 Simones-MacBook-Pro.local
Simones-MacBook-Pro.local      192.168.2.126
```

- Response sent to 224.0.0.251
- UDP with src/dst ports 5353
- Name and IP address carried in plaintext in a standard DNS packet

The screenshot shows a network packet capture analysis tool window titled "sf_mdns_avahi_name_lookup.pcap". The main display area shows a table of captured packets and a detailed view of the selected packet (No. 3).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.2.222	224.0.0.251	MDNS	85	Standard query 0x0000 A Simones-MacBook-Pro.local, "QM" question
2	0.000198	192.168.2.126	224.0.0.251	MDNS	143	Standard query response 0x0000 A, cache flush 192.168.2.126 AAAA, cache flush fe80::bb:af48:f82e:a77a NSEC,...
3	0.000275	fe80::bb:af48:f82e:a77a	ff02::fb	MDNS	163	Standard query response 0x0000 A, cache flush 192.168.2.126 AAAA, cache flush fe80::bb:af48:f82e:a77a NSEC,...

The detailed view of packet 3 shows the following structure:

- Ethernet II, Src: Apple_bd:5e:24 (78:31:c1:bd:5e:24), Dst: IPv4mcast_fb (01:00:5e:00:00:fb)
- Internet Protocol Version 4, Src: 192.168.2.126, Dst: 224.0.0.251
- User Datagram Protocol, Src Port: 5353, Dst Port: 5353
- Multicast Domain Name System (response)
 - Transaction ID: 0x0000
 - Flags: 0x8400 Standard query response, No error
 - Questions: 0
 - Answer RRs: 1
 - Authority RRs: 0
 - Additional RRs: 2
 - Answers
 - Simones-MacBook-Pro.local: type A, class IN, cache flush, addr 192.168.2.126
 - Name: Simones-MacBook-Pro.local
 - Type: A (Host Address) (1)
 - .000 0000 0000 0001 = Class: IN (0x0001)
 - 1... = Cache flush: True
 - Time to live: 120
 - Data length: 4
 - Address: 192.168.2.126



mDNS: Tips & Tricks to Play with It



```
simone@devel: ~$ avahi-resolve-host-name -4 Simone's-MacBook-Pro.local
Simone's-MacBook-Pro.local 192.168.2.126
```

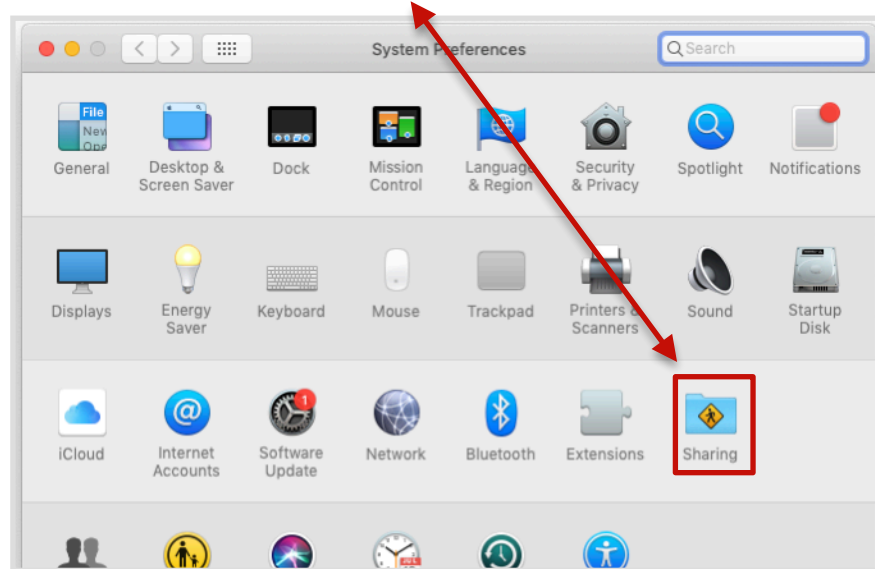
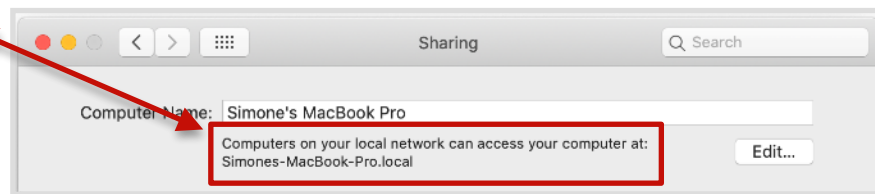
- Setting the name on OS X
 - System Prefs->Sharing

- Can use dig

```
$ dig @224.0.0.251 -p5353 +short \
"Simone's-MacBook-Pro.local"
192.168.2.126
```

- Can reverse lookup

```
$ avahi-resolve-address 192.168.2.126
192.168.2.126 Simone's-MacBook-Pro.local
```





Advertising Services with mDNS



- mDNS per-se does not provide information device types and services
- **Advertise** information about **network services** that a device offers
- DNS Service-Discovery (**DNS-SD**) - RFC 6763
 - Allows clients to discover services, and to resolve those services to host names using standard DNS queries



Discovering Services With DNS-SD



```
Simones-Mac-mini:~ simone$ dns-sd -B _services._dns-sd._udp
```

```
Browsing for _services._dns-sd._udp
```

```
DATE: ---Sun 15 Sep 2019---
```

```
17:34:30.426 ...STARTING...
```

Timestamp	A/R	Flags	if	Domain	Service Type	Instance Name
17:34:30.426	Add	3 6 .	.	.	_tcp.local.	_nfw
17:34:30.426	Add	3 6 .	.	.	_tcp.local.	_airplay
17:34:30.426	Add	3 6 .	.	.	_tcp.local.	_raop
17:34:30.426	Add	3 6 .	.	.	_tcp.local.	_touch-able
17:34:30.426	Add	3 6 .	.	.	_tcp.local.	_appletv-v2
17:34:30.426	Add	3 6 .	.	.	_udp.local.	_sleep-proxy
17:34:30.426	Add	3 6 .	.	.	_tcp.local.	_ssh
17:34:30.426	Add	3 6 .	.	.	_tcp.local.	_sftp-ssh
17:34:30.426	Add	2 6 .	.	.	_tcp.local.	_companion-link

```
Simones-Mac-mini:~ simone$ dns-sd -B _ssh._tcp
```

```
Browsing for _ssh._tcp
```

```
DATE: ---Sun 15 Sep 2019---
```

```
17:34:37.338 ...STARTING...
```

Timestamp	A/R	Flags	if	Domain	Service Type	Instance Name
17:34:37.339	Add	3 6 local.	local.	local.	_ssh._tcp.	Simone's Mac mini
17:34:37.339	Add	2 6 local.	local.	local.	_ssh._tcp.	Simone's MacBook Pro

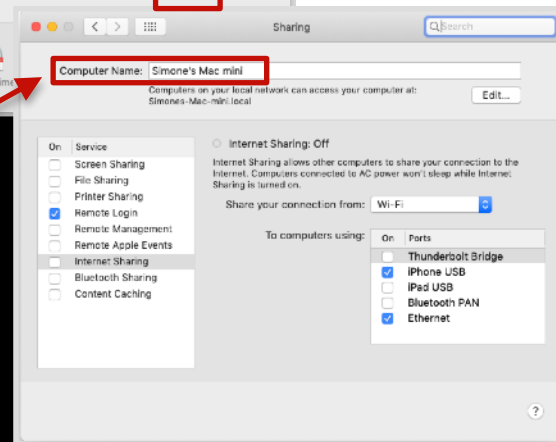
```
Simones-Mac-mini:~ simone$ dns-sd -L "Simone's MacBook Pro" _ssh._tcp
```

```
Lookup Simone's MacBook Pro._ssh._tcp.local
```

```
DATE: ---Sun 15 Sep 2019---
```

```
17:34:49.436 ...STARTING...
```

```
17:34:49.437 Simone's\032MacBook\032Pro._ssh._tcp.local. can be reached at Simones-MacBook-Pro.local.:22 (interface 6)
```





DNS-SD Service Discovery: Example [1/3]



- Queries
- Responses
- Known-Answer
Suppression to avoid wasting network capacity with repeated transmission of those answers

```
Simones-Mac-mini:~ simone$ dns-sd -B _services._dns-sd._udp
Browsing for _services._dns-sd._udp
DATE: --Sun 15 Sep 2019--
17:34:30.426 ...STARTING...
Timestamp  A/R    Flags if Domain                Service Type      Instance Name
17:34:30.426 Add    3 6 .          _tcp.local.      _nfw
17:34:30.426 Add    3 6 .          _tcp.local.      _airplay
17:34:30.426 Add    3 6 .          _tcp.local.      _raop
17:34:30.426 Add    3 6 .          _tcp.local.      _touch-able
17:34:30.426 Add    3 6 .          _tcp.local.      _appletv-v2
17:34:30.426 Add    3 6 .          _udp.local.      _sleep-proxy
17:34:30.426 Add    3 6 .          _tcp.local.      _ssh
17:34:30.426 Add    3 6 .          _tcp.local.      _sftp-ssh
17:34:30.426 Add    2 6 .          _tcp.local.      _companion-link
```

No.	Time	Source	Destination	Protocol	Length	Info
4.835305	192.168.1.159	224.0.0.251	ff02::fb	MDNS	88	Standard query 0x0000 PTR _services._dns-sd._udp.local, "QU" question
4.836013	fe80::1432:96d0:d46c:82e3	ff02::fb	ff02::fb	MDNS	108	Standard query response 0x0000 PTR _services._dns-sd._udp.local, "QU" question
4.895118	192.168.1.122	192.168.1.159	192.168.1.159	MDNS	160	Standard query response 0x0000 PTR _ssh._tcp.local PTR _sftp-ssh._tcp.local PTR _companion-link._tcp.local
4.896890	192.168.1.160	192.168.1.159	192.168.1.159	MDNS	209	Standard query response 0x0000 PTR _airplay._tcp.local PTR _raop._tcp.local PTR _touch-able._tcp.local PTR _appletv-v2._tcp.local PTR _sleep-proxy._udp.local
4.901520	192.168.1.100	224.0.0.251	ff02::fb	MDNS	106	Standard query response 0x0000 PTR _nfw._tcp.local
4.903034	fe80::ba27:ebff:fe2b:90f1	ff02::fb	ff02::fb	MDNS	126	Standard query response 0x0000 PTR _nfw._tcp.local
5.840141	192.168.1.159	224.0.0.251	ff02::fb	MDNS	307	Standard query 0x0000 PTR _services._dns-sd._udp.local, "QM" question PTR _ssh._tcp.local PTR _sftp-ssh._tcp.local PTR _companion-link._tcp.local PTR _airplay._tcp.local
5.840778	fe80::1432:96d0:d46c:82e3	ff02::fb	ff02::fb	MDNS	327	Standard query response 0x0000 PTR _services._dns-sd._udp.local, "QM" question PTR _ssh._tcp.local PTR _sftp-ssh._tcp.local PTR _companion-link._tcp.local PTR _airplay._tcp.local
8.849890	192.168.1.159	224.0.0.251	ff02::fb	MDNS	307	Standard query 0x0000 PTR _services._dns-sd._udp.local, "QM" question PTR _ssh._tcp.local PTR _sftp-ssh._tcp.local PTR _companion-link._tcp.local PTR _airplay._tcp.local
8.850581	fe80::1432:96d0:d46c:82e3	ff02::fb	ff02::fb	MDNS	327	Standard query response 0x0000 PTR _services._dns-sd._udp.local, "QM" question PTR _ssh._tcp.local PTR _sftp-ssh._tcp.local PTR _companion-link._tcp.local PTR _airplay._tcp.local
17.8694...	192.168.1.159	224.0.0.251	ff02::fb	MDNS	307	Standard query 0x0000 PTR _services._dns-sd._udp.local, "QM" question PTR _ssh._tcp.local PTR _sftp-ssh._tcp.local PTR _companion-link._tcp.local PTR _airplay._tcp.local
17.8701...	fe80::1432:96d0:d46c:82e3	ff02::fb	ff02::fb	MDNS	327	Standard query response 0x0000 PTR _services._dns-sd._udp.local, "QM" question PTR _ssh._tcp.local PTR _sftp-ssh._tcp.local PTR _companion-link._tcp.local PTR _airplay._tcp.local



DNS-SD Service Discovery: Example [2/3]



sharkfest_mdns-dns-sd_ssh_tcp.pcap

mdns

No.	Time	Source	Destination	Protocol	Length	Info
2	0.217678	192.168.1.159	224.0.0.251	MDNS	75	Standard query 0x0000 PTR _ssh_tcp.local, "OU" question
3	0.218347	fe80::1432:96d0:d46c:82e3	ff02::fb	MDNS	95	Standard query 0x0000 PTR _ssh_tcp.local, "OU" question
4	0.373955	192.168.1.122	192.168.1.159	MDNS	279	Standard query response 0x0000 PTR Simone's MacBook Pro._ssh_tcp.local SRV, cache flush 0 0 22 Simone's MacBook Pro.local TXT, cache flush TXT AAAA, cache flush fe...
5	1.220678	192.168.1.159	224.0.0.251	MDNS	144	Standard query 0x0000 PTR _ssh_tcp.local, "OU" question PTR Simone's MacBook Pro._ssh_tcp.local PTR Simone's MacBook Pro._ssh_tcp.local
6	1.221325	fe80::1432:96d0:d46c:82e3	ff02::fb	MDNS	164	Standard query 0x0000 PTR _ssh_tcp.local, "OU" question PTR Simone's MacBook Pro._ssh_tcp.local PTR Simone's MacBook Pro._ssh_tcp.local
4	2.32298	192.168.1.159	224.0.0.251	MDNS	144	Standard query 0x0000 PTR _ssh_tcp.local, "OU" question PTR Simone's MacBook Pro._ssh_tcp.local PTR Simone's MacBook Pro._ssh_tcp.local
4	2.32957	fe80::1432:96d0:d46c:82e3	ff02::fb	MDNS	164	Standard query 0x0000 PTR _ssh_tcp.local, "OU" question PTR Simone's MacBook Pro._ssh_tcp.local PTR Simone's MacBook Pro._ssh_tcp.local

▼ Simone's MacBook Pro._ssh_tcp.local: type SRV, class IN, cache flush, priority 0, weight 0, port 22, target Simone's MacBook Pro.local

Service: Simone's MacBook Pro

Protocol: _ssh

Name: _tcp

Type: SRV (Server Selection) (33)

.000 0000 0000 0001 = Class: IN (0x0001)

1... = Cache flush: True

Time to live: 120

Data length: 28

Priority: 0

Weight: 0

Port: 22

Target: Simone's MacBook Pro.local

► Simone's MacBook Pro._ssh_tcp.local: type TXT, class IN, cache flush

▼ Simone's MacBook Pro._device-info_tcp.local: type TXT, class IN

Name: Simone's MacBook Pro._device-info_tcp.local

Type: TXT (Text strings) (16)

.000 0000 0000 0001 = Class: IN (0x0001)

0... = Cache flush: False

Time to live: 4500

Data length: 32

TXT Length: 20

TXT: model=MacBookPro11,1

TXT Length: 19

TXT: osxvers=18

```
Simone's-Mac-mini:~ simone$ dns-sd -B _ssh._tcp
Browsing for _ssh._tcp
DATE: ---Sun 15 Sep 2019---
17:34:37.338 ...STARTING...
Timestamp   A/R   Flags  if Domain           Service Type      Instance Name
17:34:37.339 Add    3   6 local.           _ssh._tcp.       Simone's Mac mini
17:34:37.339 Add    2   6 local.           _ssh._tcp.       Simone's MacBook Pro
```



DNS-SD Service Discovery: Example [3/3]



Apply a display filter ... <36/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.2.126	224.0.0.251	MDNS	194	Standard query 0x0000 PTR _universal._sub._ipp._tcp.local, "00" question PTR _unive
2	0.000694	fe80::bba:f48:f82e:a77a	ff02::fb	MDNS	214	Standard query 0x0000 PTR _universal._sub._ipp._tcp.local, "00" question PTR _unive
3	0.101480	192.168.2.125	224.0.0.251	MDNS	707	Standard query response 0x0000 PTR OKI-MC342-361BF5._ipp._tcp.local PTR OKI-MC342-3

User Datagram Protocol, Src Port: 5353, Dst Port: 5353

Multicast Domain Name System (response)

Transaction ID: 0x0000

Flags: 0xB400 Standard query response, No error

Questions: 0

Answer RRs: 2

Authority RRs: 0

Additional RRs: 3

Answers

Additional records

- OKI-MC342-361BF5._ipp._tcp.local: type SRV, class IN, cache flush, priority 60, weight 0, port 631, target oki-mc342-361bf5.local
 - Service: OKI-MC342-361BF5
 - Protocol: **_ipp**
 - Name: _tcp
 - Type: SRV (Server Selection) (33)
 - .000 0000 0000 0001 = Class: IN (0x0001)
 - 1... .. = Cache flush: True
 - Time to live: 7200
 - Data length: 25
 - Priority: 60
 - Weight: 0
 - Port: **631**
 - Target: oki-mc342-361bf5.local
- OKI-MC342-361BF5._ipp._tcp.local: type TXT, class IN, cache flush
 - Name: OKI-MC342-361BF5._ipp._tcp.local
 - Type: TXT (Text strings) (16)
 - .000 0000 0000 0001 = Class: IN (0x0001)
 - 1... .. = Cache flush: True
 - Time to live: 7200
 - Data length: 512
 - TXT Length: 9
 - TXT: txtvers=1
 - TXT Length: 8
 - TXT: qtotal=1
 - TXT Length: 111
 - TXT: pd=application/octet-stream,application/vnd.hp-PCL,application/postscript,application/pdf,image/jpeg,image/urf
 - TXT Length: 12
 - TXT: rp=ipp/print
 - TXT Length: 12
 - TXT: ty=OKI MC342

Printers & Scanners

Printers

- EPSON WF-2630 Series
 - Offline
- Samsung CLP-310 Se...
 - Offline, Last Used

EPSON WF-2630 Series

Open Print Queue...

Options & Supplies...

Add

Name: OKI-MC342-361BF5 2

Location:

Use: AirPrint

Add



DNS-SD: Tips & Tricks to Play with It



- Can use avahi-browse

```
$ avahi-browse --all
+ docker0 IPv4 apt-cacher-ng proxy on devel      _apt_proxy._tcp      local
+   eno1 IPv4 apt-cacher-ng proxy on devel      _apt_proxy._tcp      local
+   eno1 IPv4 Simone's MacBook Pro             _companion-link._tcp local
+   eno1 IPv4 Simone's MacBook Pro             SFTP File Transfer   local
+   eno1 IPv4 Simone's MacBook Pro             SSH Remote Terminal  local
```

- Can use dig

```
$ dig @224.0.0.251 -p 5353 -t ptr _ssh._tcp.local
[...]
;; ANSWER SECTION:
_ssh._tcp.local.      10          IN          PTR         Simone's\032MacBook\032Pro._ssh._tcp.local.

;; ADDITIONAL SECTION:
Simone's\032MacBook\032Pro._ssh._tcp.local. 10 IN SRV 0 0 22 Simones-MacBook-Pro.local.
Simone's\032MacBook\032Pro._ssh._tcp.local. 10 IN TXT ""
Simone's\032MacBook\032Pro._device-info._tcp.local. 10 IN TXT "model=MacBookPro11,1" "osxvers=18"
Simones-MacBook-Pro.local. 10 IN AAAA fe80::bb:af48:f82e:a77a
Simones-MacBook-Pro.local. 10 IN A 192.168.2.126
```



How to Use mDNS Data



- **Names to passively profile users**
 - Apple devices are particularly open in their default hostname choice of the users' first and last names



How to Use DNS-SD Data



- **Port scanning**
 - `_ssh._tcp`
- **Service type enumeration**
 - Meta-query: `"_services._dns-sd._udp.<domain>"`
 - `$ dns-sd -B _services._dns-sd._udp`
- **OS versions, details, information**
 - Sent in TXT and SRV records



mDNS and DNS-SD: Secure? [1/2]



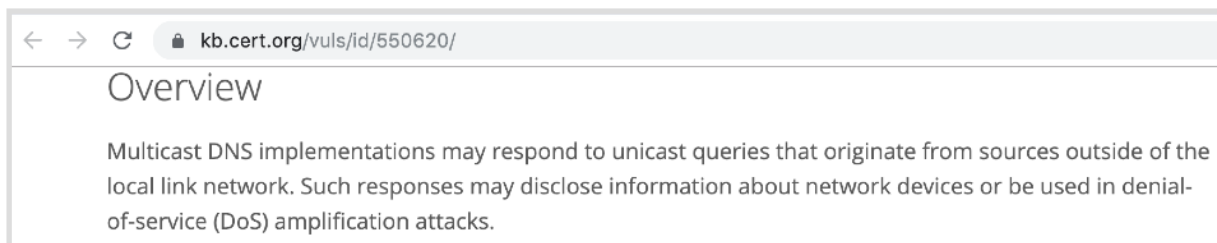
- mDNS and DNS-SD are just specifications for how to name and use records in the existing DNS system, it has no specific additional security requirements over and above those that already apply to DNS queries and DNS updates



mDNS and DNS-SD: Secure? [2/2]



- An **attacker** can respond to typo-ed domains, race against valid domains, and advertise services that don't really exist
- If not properly configured, mDNS may reply to queries from outside the link local network!
 - Publicly (Internet!) disclose software and services, as well as other potentially sensitive information, suchlike hostname, internal network configuration settings, model number, etc
 - Amplification attacks: requests for all services with a spoofed source IP address





The Simple Service Discovery Protocol (SSDP) [1/2]



- Similar in spirit to mDNS-SD, SSDP is used for the advertisement/discovery of network devices and services
 - Step 1 (**Discovery**) in the Universal Plug and Play (UPnP) technology which enables "seamless proximity networking in addition to control and data transfer among networked devices"
- Likely that home devices support UPnP and hence SSDP
 - They can be easily discovered by your computer or phone
- Devices, for example when they join the network, can query for specific devices and their services
 - Internet gateways, audio systems, TVs, or printers



The Simple Service Discovery Protocol (SSDP) [2/2]



- IP UDP (port 1900) multicast packets carrying HTTP
- Discovery
 - Advertisement
 - For example when a device is newly connected to the network
 - Search
 - Look for available devices and offered services



SSDP Advertisement: Example



```
Apply a display filter ... <=>/>
No. Time Source Destination Protocol Length Info Location
46 52.805808 192.168.2.6 239.255.255.250 SSDP 406 NOTIFY * HTTP/1.1 http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
47 52.830459 192.168.2.6 239.255.255.250 SSDP 408 NOTIFY * HTTP/1.1 http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
48 52.852971 192.168.2.6 239.255.255.250 SSDP 396 NOTIFY * HTTP/1.1 http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
49 52.876770 192.168.2.6 239.255.255.250 SSDP 350 NOTIFY * HTTP/1.1 http://192.168.2.6:52235/rccr/RemoteControlReceiver.xml
50 52.101178 192.168.2.6 239.255.255.250 SSDP 359 NOTIFY * HTTP/1.1 http://192.168.2.6:52235/rccr/RemoteControlReceiver.xml

Frame 47: 408 bytes on wire (3264 bits), 408 bytes captured (3264 bits)
Ethernet II, Src: Samsung_E0:dd:be: (d0:66:7b:0e:dd:be), Dst: IPv4cast_7f:ff:fa (01:00:5e:7f:ff:fa)
Internet Protocol Version 4, Src: 192.168.2.6, Dst: 239.255.255.250
User Datagram Protocol, Src Port: 1020, Dst Port: 1900
Simple Service Discovery Protocol
  NOTIFY * HTTP/1.1\r\n
  | [Expert Info [Chat/Sequence]: NOTIFY * HTTP/1.1\r\n
  | Request Method: NOTIFY
  | Request URI:
  | Request Version: HTTP/1.1
  | HOST: 239.255.255.250:1900\r\n
  | CACHE-CONTROL: max-age=1800\r\n
  | LOCATION: http://192.168.2.6:52235/dmr/SamsungMRDesc.xml\r\n
  | NT: urn:schemas-upnp-org:service:ConnectionManager:1\r\n
  | NTS: sddp:alive\r\n
  | WWW-URL: http://192.168.2.6:52235/dmr/SamsungMRDesc.xml;urn:schemas-upnp-org:service:ConnectionManager:1\r\n
  | SERVER: Linux/9.0 UPnP/1.0 PROTOTYPE/1.0\r\n
  | CONTENT-LENGTH: 0\r\n
  | \r\n
  | [Full request URI: http://239.255.255.250:1900]
```

- Multicast/Unicast NOTIFY message
- Notification type and subtype (NT and NTS), Unique Service Name (USN), Server, ...

```
192.168.2.6:52235/dmr/SamsungMRDesc.xml
Not Secure 192.168.2.6:52235/dmr/SamsungMRDesc.xml
<root xmlns="urn:schemas-upnp-org:device-1-0" xmlns:df="http://schemas.microsoft.com/windows" >
  <script/>
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <device>
    <deviceType>urn:schemas-upnp-org:device:MediaRenderer:1</deviceType>
    <df:X_deviceCategory>Display.TV.LCD Multimedia.DMR</df:X_deviceCategory>
    <dlina:X_DLNAOC xmlns:dlina="urn:schemas-dlna-org:device-1-0">DMR-1.50</dlina:X_DLNAOC>
    <friendlyName>Mainardi's LED TV</friendlyName>
    <manufacturer>Samsung Electronics</manufacturer>
    <manufacturerURL>http://www.samsung.com/sec</manufacturerURL>
    <modelDescription>Samsung TV DMR</modelDescription>
    <modelName>UE40D6500</modelName>
    <modelNumber>AllShare1.0</modelNumber>
    <modelURL>http://www.samsung.com/sec</modelURL>
    <serialNumber>20081224DMR</serialNumber>
    <UDN>uuid:937746b0-6777-c90a-8328-e7817dc2926e</UDN>
    <sec:deviceID>KLCFP7UYUVAGO</sec:deviceID>
```

**UPnP Step 2:
Description**



SSDP Search: Example



- M-SEARCH HTTP multicast request
- Namespace (fixed, MAN), Search Target (ST), User Agent
- Example is an iPhone looking for remotely-controllable TVs

sf_ssdp_samsung_remote_control.pcap

No.	Time	Source	Destination	Protocol	Length	Info	User-Agent
175	117.052496	192.168.2.7	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1	10S/13.1.3 UPnP/1.1 ConnectSDK/1.6.0
176	117.058389	192.168.2.7	239.255.255.250	SSDP	221	M-SEARCH * HTTP/1.1	10S/13.1.3 UPnP/1.1 ConnectSDK/1.6.0
177	117.058396	192.168.2.7	239.255.255.250	SSDP	186	M-SEARCH * HTTP/1.1	10S/13.1.3 UPnP/1.1 ConnectSDK/1.6.0
178	117.058397	192.168.2.7	239.255.255.250	SSDP	219	M-SEARCH * HTTP/1.1	10S/13.1.3 UPnP/1.1 ConnectSDK/1.6.0

Frame 176: 221 bytes on wire (1768 bits), 221 bytes captured (1768 bits)

Ethernet II, Src: Apple_ce:28:53 (58:40:4e:ce:28:53), Dst: IPv4mcast_7:ff:fa (01:00:5e:7f:ff:fa)

Internet Protocol Version 4, Src: 192.168.2.7, Dst: 239.255.255.250

User Datagram Protocol, Src Port: 58825, Dst Port: 1900

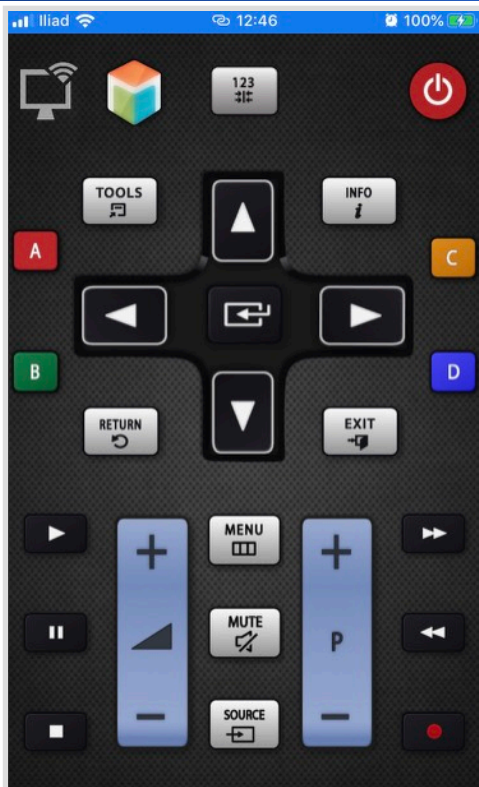
Simple Service Discovery Protocol

- M-SEARCH * HTTP/1.1\r\n
 - [Expert Info (Chat/Sequence): M-SEARCH * HTTP/1.1\r\n]Request Method: M-SEARCHRequest URI: *Request Version: HTTP/1.1Host: 239.255.255.250:1900\r\nST: urn:schemas-upnp-org:device:MediaRenderer:1\r\nUser-Agent: 10S/13.1.3 UPnP/1.1 ConnectSDK/1.6.0\r\nMAN: \"ssdp:discover\"\r\n

```
0000 01 00 5e 7f ff fa 58 40 4e ce 28 53 08 00 45 00  ..@..X@N-(S..E-
0010 00 cf fe 1c 00 00 01 11 08 58 c0 a8 02 07 ef ff  ..X.....
0020 ff fa e5 c9 07 6c 00 bb 5f 55 4d 2d 53 45 41 52  ..-..UM-SEAR
0030 43 48 20 2a 20 48 54 54 50 2f 31 2e 31 0d 0a 48  CH * HTT P/1.1..H
0040 6f 73 74 3a 20 32 33 39 2e 32 35 35 2e 32 35 35  ost: 239 .255.255
0050 2e 32 35 30 3a 31 39 30 30 0d 0a 53 54 3a 20 75  .250:190 0-ST: u
0060 72 6e 3a 73 63 68 65 6d 61 73 2d 75 70 6e 70 2d  rrn:schem as-upnp-
0070 6f 72 67 3a 64 65 76 69 63 65 3a 4d 65 64 69 61  org:devi ce:Media
0080 52 65 6e 64 65 72 65 72 3a 31 0d 0a 55 73 65 72  Renderer :1-User
0090 2d 41 67 65 6e 74 3a 20 69 4f 53 2f 31 33 2e 31  -Agent: 10S/13.1
00a0 2e 33 20 55 50 6e 50 2f 31 2e 31 20 43 6f 6e 6e  .3 UPnP/ 1.1 Conn
00b0 65 63 74 53 44 4b 2f 31 2e 36 2e 30 0d 0a 4d 41  ectSDK/1 .6.0..MA
00c0 4e 3a 20 22 73 73 64 70 3a 64 69 73 63 6f 76 65  N: \"ssdp :discove
00d0 72 22 0d 0a 4d 58 3a 20 35 0d 0a 0d 0a          r\"..MX: 5.....
```



Discover a Remotely-Controllable TV with SSDP: Example



1: Advertisement

sf_ssdp_samsung_remote_control.pcap

Apply a display filter ... <%%/>

No.	Time	Source	Destination	Protocol	Length	Info	Location
164	105.936030	192.168.2.6	239.255.255.250	SSDP	342	NOTIFY * HTTP/1.1	http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
165	105.963407	192.168.2.6	239.255.255.250	SSDP	351	NOTIFY * HTTP/1.1	http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
166	105.984359	192.168.2.6	239.255.255.250	SSDP	398	NOTIFY * HTTP/1.1	http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
167	106.008052	192.168.2.6	239.255.255.250	SSDP	406	NOTIFY * HTTP/1.1	http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
168	106.031944	192.168.2.6	239.255.255.250	SSDP	408	NOTIFY * HTTP/1.1	http://192.168.2.6:52235/dmr/SamsungMRDesc.xml
169	106.056051	192.168.2.6	239.255.255.250	SSDP	396	NOTIFY * HTTP/1.1	http://192.168.2.6:52235/dmr/SamsungMRDesc.xml

2: Description

sf_ssdp_samsung_remote_control.pcap

Apply a display filter ... <%%/>

No.	Time	Source	Destination	Protocol	Length	Info	Location
188	118.718209	192.168.2.7	192.168.2.6	HTTP	272	GET /dmr/SamsungMRDesc.xml HTTP/1.1	
189	118.723542	192.168.2.6	192.168.2.7	TCP	66	52235 → 62458 [ACK] Seq=1 Ack=207 Wi...	
190	118.726718	192.168.2.6	192.168.2.7	TCP	204	52235 → 62458 [PSH, ACK] Seq=1 Ack=2...	
191	118.727930	192.168.2.6	192.168.2.7	TCP	1514	52235 → 62458 [ACK] Seq=139 Ack=207 ...	
192	118.727937	192.168.2.6	192.168.2.7	TCP	1514	52235 → 62458 [ACK] Seq=1587 Ack=207...	
193	118.728038	192.168.2.7	192.168.2.6	TCP	66	62458 → 52235 [ACK] Seq=207 Ack=139 ...	
194	118.729342	192.168.2.7	192.168.2.6	TCP	66	62458 → 52235 [ACK] Seq=207 Ack=3035...	
195	118.730040	192.168.2.6	192.168.2.7	HTTP/X...	138	HTTP/1.1 200 OK	

3: Control

sf_ssdp_samsung_remote_control.p

Apply a display filter ... <%%/>

No.	Time	Source	Destination	Protocol	Length	Info	Location
309	155.827724	192.168.2.7	192.168.2.6	TCP	99	50260 → 55000 [PSH, ACK] Seq=108 Ack...	
310	155.832219	192.168.2.6	192.168.2.7	TCP	66	55000 → 50260 [ACK] Seq=66 Ack=141 W...	
311	155.863519	192.168.2.6	192.168.2.7	TCP	87	55000 → 50260 [PSH, ACK] Seq=66 Ack=...	



SSDP: Plaintext Information



- Plaintext information which can unveil devices types, characteristics and software version
 - User Agents
 - iOS/13.1.3 UPnP/1.1 ConnectSDK/1.6.0
 - Servers
 - Linux/9.0 UPnP/1.0 PROTOTYPE/1.0
 - USNs
 - 937746b0-6777-c90a-8328-e7817dc2926e::upnp:rootdevice



How to Use SSDP Data



- **Services Enumeration**
 - Advertised in NOTIFY messages
 - Perform queries with M-SEARCH
- **OS and other applications versions, details, information**
 - Advertised both in M-SEARCH and NOTIFY messages



SSDP: Secure?



- “To be found by a network search, a device shall send a unicast UDP response to the source IP address and port that sent the request to the multicast address.”
- Amplification attacks: requests for all services with a spoofed source IP address





The Dynamic Host Configuration Protocol (DHCP)



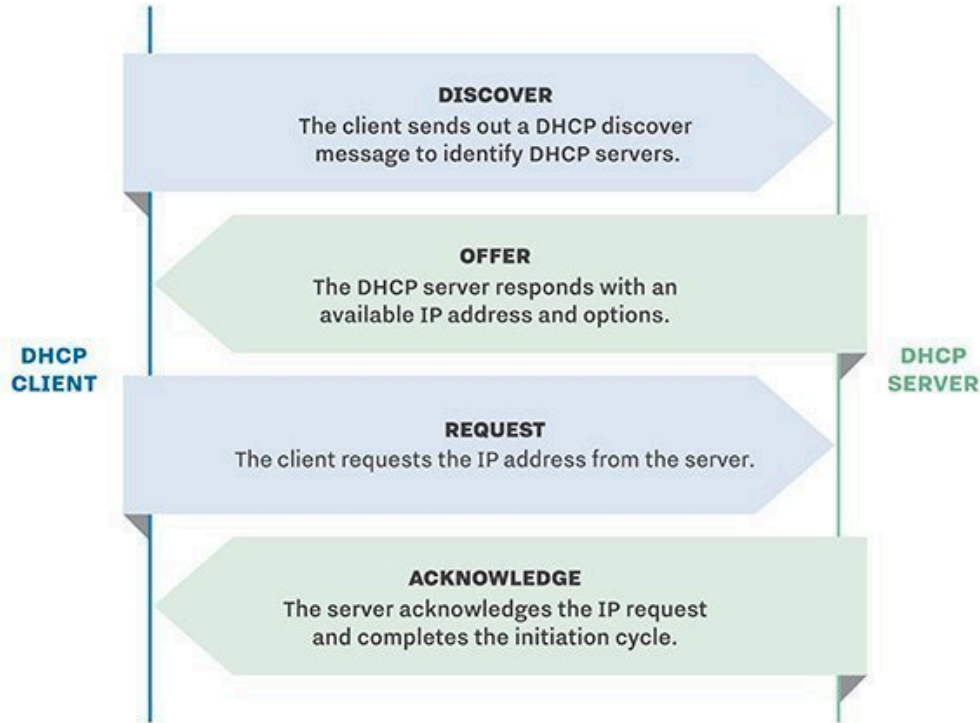
- What happens right after a host has connected to the network?
 - After the **ethernet** cable has been plugged
 - After the **WiFi** has been successfully joined
- To use the network an host typically needs at least to
 - Have an IP address
 - Know the IP address of someone who is in charge of carrying its traffic to the internet (i.e., the **gateway**)
 - Know the IP address of the **DNS server**
- The Dynamic Host Configuration Protocol (**DHCP**) is used to tell the newly connected host all the necessary information to use the joined network



DHCP: Basics



- DHCP client on the host
- DHCP server on the network
- UDP, 4 phases (**DORA**)
 - **D**iscover
 - **O**ffer
 - **R**equest
 - **A**cknowledgement





DHCP Discover: Example



The screenshot displays a network capture in Wireshark for the file 'sf_dhcp.pcap'. The filter is 'udp.port==67 || udp.port==68'. The packet list shows a DHCP Discover packet (No. 13) from source 192.168.1.100 to destination 255.255.255.255. The packet details pane shows the following options:

- Magic cookie: DHCP
- Option: (53) DHCP Message Type (Discover)
 - Length: 1
 - DHCP: Discover (1)
- Option: (61) Client identifier
 - Length: 7
 - Hardware type: Ethernet (0x01)
 - Client MAC address: SamsungE_aa:d3:9d (cc:21:19:aa:d3:9d)
- Option: (57) Maximum DHCP Message Size
 - Length: 2
 - Maximum DHCP Message Size: 1500
- Option: (60) Vendor class identifier
 - Length: 14
 - Vendor class identifier: android-dhcp-9
- Option: (12) Host Name
 - Length: 14
 - Host Name: Galaxy-A7-2018
- Option: (55) Parameter Request List
 - Length: 10
 - Parameter Request List Item: (1) Subnet Mask
 - Parameter Request List Item: (3) Router
 - Parameter Request List Item: (6) Domain Name Server
 - Parameter Request List Item: (15) Domain Name
 - Parameter Request List Item: (26) Interface MTU
 - Parameter Request List Item: (28) Broadcast Address
 - Parameter Request List Item: (51) IP Address Lease Time
 - Parameter Request List Item: (58) Renewal Time Value
 - Parameter Request List Item: (59) Rebinding Time Value
 - Parameter Request List Item: (43) Vendor-Specific Information
- Option: (255) End
 - Option End: 255
 - Padding: 00

On the right, a screenshot of an Android phone's 'About phone' screen is shown. A red box highlights 'Galaxy A7 (2018)', and a blue arrow points from the 'Host Name' field in the Wireshark details pane to this box.



DHCP Offer: Example



sf_dhcp.pcap

udp.port==67 || udp.port == 68

No.	Time	Source	Destination	Source	Destination	Dest Sour	Protocol	Length	Host Name	Vendor class identifier	Info
12	35.930227	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67 68	DHCP	344	Galaxy-A7-2018	android-dhcp-9	DHCP Discover - Transaction ID 0xf035f4...
13	35.930970	192.168.1.100	192.168.1.121	b8:27:eb:2b:90:f1	cc:21:19:aa:d3:9d	68 67	DHCP	342			DHCP Offer - Transaction ID 0xf035f4...
14	35.966873	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67 68	DHCP	356	Galaxy-A7-2018	android-dhcp-9	DHCP Request - Transaction ID 0xf035f4...
15	35.975489	192.168.1.100	192.168.1.121	b8:27:eb:2b:90:f1	cc:21:19:aa:d3:9d	68 67	DHCP	342			DHCP ACK - Transaction ID 0xf035f4...

Your (client) IP address: 192.168.1.121

Next server IP address: 0.0.0.0

Relay agent IP address: 0.0.0.0

Client MAC address: SamsungE_aa:d3:9d (cc:21:19:aa:d3:9d)

Client hardware address padding: 00000000000000000000

Server host name not given

Boot file name not given

Magic cookie: DHCP

- Option: (53) DHCP Message Type (Offer)
 - Length: 1
 - DHCP: Offer (2)
- Option: (54) DHCP Server Identifier
 - Length: 4
 - DHCP Server Identifier: 192.168.1.100
- Option: (51) IP Address Lease Time
 - Length: 4
 - IP Address Lease Time: (600s) 10 minutes
- Option: (1) Subnet Mask
 - Length: 4
 - Subnet Mask: 255.255.255.0
- Option: (3) Router
 - Length: 4
 - Router: 192.168.1.100
- Option: (6) Domain Name Server
 - Length: 4
 - Domain Name Server: 208.67.222.222
- Option: (15) Domain Name
 - Length: 10
 - Domain Name: ntop.local



DHCP Request: Example



sf_dhcp.pcap

Apply a display filter ... <3%/>

No.	Time	Source	Destination	Source	Destination	Dest	Sour	Protocol	Length	Host Name	Vendor class identifier	Info
12	35.930227	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67	68	DHCP	344	Galaxy-A7-2018	android-dhcp-9	DHCP Discover - Transaction ID 0xf035f4...
13	35.930970	192.168.1.100	192.168.1.121	b8:27:eb:2b:90:f1	cc:21:19:aa:d3:9d	68	67	DHCP	342			DHCP Offer - Transaction ID 0xf035f4...
...	35.966873	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67	68	DHCP	356	Galaxy-A7-2018	android-dhcp-9	DHCP Request - Transaction ID 0xf035f4...
15	35.975489	192.168.1.100	192.168.1.121	b8:27:eb:2b:90:f1	cc:21:19:aa:d3:9d	68	67	DHCP	342			DHCP ACK - Transaction ID 0xf035f4...

▼ Option: (61) Client identifier
Length: 7
Hardware type: Ethernet (0x01)
Client MAC address: SamsungE_aa:d3:9d (cc:21:19:aa:d3:9d)

▼ Option: (50) Requested IP Address
Length: 4
Requested IP Address: 192.168.1.121

▼ Option: (54) DHCP Server Identifier
Length: 4
DHCP Server Identifier: 192.168.1.100

▼ Option: (57) Maximum DHCP Message Size
Length: 2
Maximum DHCP Message Size: 1500

▼ Option: (60) Vendor class identifier
Length: 14
Vendor class identifier: android-dhcp-9

▼ Option: (12) Host Name
Length: 14
Host Name: Galaxy-A7-2018

▼ Option: (55) Parameter Request List
Length: 10
Parameter Request List Item: (1) Subnet Mask
Parameter Request List Item: (3) Router
Parameter Request List Item: (6) Domain Name Server
Parameter Request List Item: (15) Domain Name



Broadcasted DHCP in a Small Network



sf_dhcp.pcap

eth.dst == ff:ff:ff:ff:ff:ff

No.	Time	Source	Destination	Source	Destination	Dest	Sour	Protocol	Length	Host Name	Vendor class identifier	Info
1	0.000000	0.0.0.0	255.255.255.255	88:bd:45:d9:6d:a6	ff:ff:ff:ff:ff:ff	67	68	DHCP	348	Galaxy-A3-2017	android-dhcp-8.0.0	DHCP Discover - Transaction ID 0x29404c1c
3	1.053107	0.0.0.0	255.255.255.255	88:bd:45:d9:6d:a6	ff:ff:ff:ff:ff:ff	67	68	DHCP	360	Galaxy-A3-2017	android-dhcp-8.0.0	DHCP Request - Transaction ID 0x29404c1c
5	10.279607	0.0.0.0	255.255.255.255	58:40:4e:ce:28:53	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	iPhone		DHCP Discover - Transaction ID 0x1f4b7e8b
7	12.290459	0.0.0.0	255.255.255.255	58:40:4e:ce:28:53	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	iPhone		DHCP Request - Transaction ID 0x1f4b7e8b
9	34.880545	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67	68	DHCP	344	Galaxy-A7-2018	android-dhcp-9	DHCP Discover - Transaction ID 0xf035f406
10	35.380496	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67	68	DHCP	344	Galaxy-A7-2018	android-dhcp-9	DHCP Discover - Transaction ID 0xf035f406
12	35.930227	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67	68	DHCP	344	Galaxy-A7-2018	android-dhcp-9	DHCP Discover - Transaction ID 0xf035f406
14	35.966873	0.0.0.0	255.255.255.255	cc:21:19:aa:d3:9d	ff:ff:ff:ff:ff:ff	67	68	DHCP	356	Galaxy-A7-2018	android-dhcp-9	DHCP Request - Transaction ID 0xf035f406
16	48.912781	0.0.0.0	255.255.255.255	78:31:c1:bd:5e:24	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	Simones-MBP		DHCP Request - Transaction ID 0xc2a7c233
18	60.037611	0.0.0.0	255.255.255.255	74:e1:b6:c6:da:a9	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	iPaddiLabriella		DHCP Request - Transaction ID 0x90396f27
20	117.781577	0.0.0.0	255.255.255.255	00:24:e4:74:f0:ee	ff:ff:ff:ff:ff:ff	67	68	DHCP	342			DHCP Discover - Transaction ID 0x14b38c73
22	118.797668	0.0.0.0	255.255.255.255	00:24:e4:74:f0:ee	ff:ff:ff:ff:ff:ff	67	68	DHCP	342			DHCP Request - Transaction ID 0x14b38c73
24	122.805210	192.168.1.161	0.0.0.0	00:24:e4:74:f0:ee	ff:ff:ff:ff:ff:ff	67	68	DHCP	342			DHCP Release - Transaction ID 0x34d8125e
25	140.051067	0.0.0.0	255.255.255.255	b0:ee:7b:fd:f5:fd	ff:ff:ff:ff:ff:ff	67	68	DHCP	590	TV Box - 140		DHCP Request - Transaction ID 0x55c5946f
27	153.966058	0.0.0.0	255.255.255.255	60:03:08:d5:56:38	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	Mainas-Apple-TV		DHCP Discover - Transaction ID 0x7822a88
28	153.966712	0.0.0.0	255.255.255.255	60:03:08:d5:56:38	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	Mainas-Apple-TV		DHCP Discover - Transaction ID 0x7822a89
30	155.362404	0.0.0.0	255.255.255.255	60:03:08:d5:56:38	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	Mainas-Apple-TV		DHCP Discover - Transaction ID 0x7822a89
32	156.401173	0.0.0.0	255.255.255.255	60:03:08:d5:56:38	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	Mainas-Apple-TV		DHCP Request - Transaction ID 0x7822a89
34	202.334896	0.0.0.0	255.255.255.255	d0:66:7b:0e:dd:be	ff:ff:ff:ff:ff:ff	67	68	DHCP	590		udhcp 1.14.3-VD Linu...	DHCP Discover - Transaction ID 0x19f46911
36	203.346550	0.0.0.0	255.255.255.255	d0:66:7b:0e:dd:be	ff:ff:ff:ff:ff:ff	67	68	DHCP	590		udhcp 1.14.3-VD Linu...	DHCP Request - Transaction ID 0x19f46911
40	252.553526	0.0.0.0	255.255.255.255	38:9d:92:17:f5:39	ff:ff:ff:ff:ff:ff	67	68	DHCP	590	EPSON17F539	udhcp	DHCP Discover - Transaction ID 0x36b1a56d
41	252.559651	0.0.0.0	255.255.255.255	38:9d:92:17:f5:39	ff:ff:ff:ff:ff:ff	67	68	BOOTP	342			Boot Request from 38:9d:92:17:f5:39 (SeikoEps_17:f5:39)
43	253.560168	0.0.0.0	255.255.255.255	38:9d:92:17:f5:39	ff:ff:ff:ff:ff:ff	67	68	DHCP	590	EPSON17F539	udhcp	DHCP Request - Transaction ID 0x36b1a56d
47	323.181935	0.0.0.0	255.255.255.255	58:40:4e:ce:28:53	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	iPhone		DHCP Request - Transaction ID 0x1f4b7e8d
51	331.462211	0.0.0.0	255.255.255.255	50:32:37:eb:bb:40	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	Simones-Mini		DHCP Request - Transaction ID 0x4a23af3d
53	377.282160	0.0.0.0	255.255.255.255	78:31:c1:bd:5e:24	ff:ff:ff:ff:ff:ff	67	68	DHCP	343	DESKTOP-E7D8H40	MSFT 5.0	DHCP Discover - Transaction ID 0x8b64e948
54	378.284014	192.168.1.100	255.255.255.255	b8:27:eb:2b:90:f1	ff:ff:ff:ff:ff:ff	68	67	DHCP	342			DHCP Offer - Transaction ID 0x8b64e948
55	378.351718	0.0.0.0	255.255.255.255	78:31:c1:bd:5e:24	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	DESKTOP-E7D8H40	MSFT 5.0	DHCP Request - Transaction ID 0x8b64e948
56	378.374263	192.168.1.100	255.255.255.255	b8:27:eb:2b:90:f1	ff:ff:ff:ff:ff:ff	68	67	DHCP	369			DHCP ACK - Transaction ID 0x8b64e948
57	385.284609	192.168.1.165	255.255.255.255	78:31:c1:bd:5e:24	ff:ff:ff:ff:ff:ff	67	68	DHCP	342	DESKTOP-E7D8H40	MSFT 5.0	DHCP Inform - Transaction ID 0x63739331



How to Use DHCP Data: Discovers and Request [1/2]



- **DHCP Discovers** and **Requests** are sent in **broadcast**
- Every host on the same subnet sees all the DHCP discovers and requests
- Passively determine
 - All the MAC addresses connected to the network
 - All the host names of all the devices connected to the network
 - Associations between IP and MAC addresses



How to Use DHCP Data: Discovers and Request [2/2]



- Host Name / MAC address
 - Associate devices to people (Simones-Mini: the Mac Mini of Simone)
 - Determine device types (Galaxy A7)
- Vendor class identifier
 - Determine the DHCP client and, thus, the operating system

Dest	Sour	Protocol	Length	Host Name	Vendor class identifier
67	68	DHCP	348	Galaxy-A3-2017	android-dhcp-8.0.0
67	68	DHCP	360	Galaxy-A3-2017	android-dhcp-8.0.0
67	68	DHCP	342	iPhone	
67	68	DHCP	342	iPhone	
67	68	DHCP	344	Galaxy-A7-2018	android-dhcp-9
67	68	DHCP	344	Galaxy-A7-2018	android-dhcp-9
67	68	DHCP	344	Galaxy-A7-2018	android-dhcp-9
67	68	DHCP	356	Galaxy-A7-2018	android-dhcp-9
67	68	DHCP	342	Simones-MBP	
67	68	DHCP	342	iPaddiLabriella	
67	68	DHCP	342		
67	68	DHCP	342		
67	68	DHCP	342		
67	68	DHCP	590	TV Box - 140	
67	68	DHCP	342	Mainas-Apple-TV	
67	68	DHCP	342	Mainas-Apple-TV	
67	68	DHCP	342	Mainas-Apple-TV	
67	68	DHCP	342	Mainas-Apple-TV	
67	68	DHCP	590		udhcp 1.14.3-VD Linu...
67	68	DHCP	590		udhcp 1.14.3-VD Linu...
67	68	DHCP	590	EPSON17F539	udhcp
67	68	BOOTP	342		
67	68	DHCP	590	EPSON17F539	udhcp
67	68	DHCP	342	iPhone	
67	68	DHCP	342	Simones-Mini	
67	68	DHCP	343	DESKTOP-E7D8H40	MSFT 5.0
68	67	DHCP	342		
67	68	DHCP	369	DESKTOP-E7D8H40	MSFT 5.0
68	67	DHCP	342		
67	68	DHCP	342	DESKTOP-E7D8H40	MSFT 5.0



How to Use DHCP Data: Fingerprinting



- **Fingerprinting** to guess the OS
 - The order in which the DHCP client asks for certain options is relatively unique and identifies the specific operating system version

```
simone@Mac-mini:Downloads$ curl -XSET -V-H 'Content-Type: application/json' 'https://api.fingerbank.org/api/v2/combinations/interrogate?pretty=true&key=1,15,3,6,4,45,4,31,33,121,249,252,43' | python -m json.tool
{
  "total": 100,
  "received": 100,
  "xtra": 100,
  "average": 80,
  "sp": 141,
  "time": 785,
  "time": 785,
  "time": 785,
  "current": 785,
  "device": {
    "can_be_more_precise": true,
    "child_devices_count": 12,
    "child_virtual_devices_count": 1,
    "created_at": "2019-09-09T15:09:50.000Z",
    "id": 1,
    "name": "Windows OS",
    "parent_id": 16879,
    "parents": [
      {
        "created_at": "2017-09-14T18:41:06.000Z",
        "id": 16879,
        "name": "Operating System",
        "parent_id": null,
        "updated_at": "2017-09-10T16:33:10.000Z",
        "virtual_parent_id": null
      }
    ],
    "updated_at": "2018-11-09T14:52:53.000Z",
    "virtual_parent_id": null
  },
  "device_name": "Operating System/Windows OS",
  "score": 50,
  "version": ""
}
```

No.	Time	Source	Destination	Source	Destination	Dest Sour	Protocol	Length	Host Name	Vendor class identifier
53	377.282168	0.0.0.0	255.255.255.255	78:31:c1:bd:5e:24	ff:ff:ff:ff:ff:ff	67 68	DHCP	343	DESKTOP-E7D8H40	MSFT 5.0
54	378.284614	192.168.1.100	255.255.255.255	b8:27:eb:2b:90:f1	ff:ff:ff:ff:ff:ff	67 68	DHCP	342	DESKTOP-E7D8H40	MSFT 5.0
55	378.351718	0.0.0.0	255.255.255.255	78:31:c1:bd:5e:24	ff:ff:ff:ff:ff:ff	67 68	DHCP	369	DESKTOP-E7D8H40	MSFT 5.0
56	378.374263	192.168.1.100	255.255.255.255	b8:27:eb:2b:90:f1	ff:ff:ff:ff:ff:ff	67 68	DHCP	342	DESKTOP-E7D8H40	MSFT 5.0
57	385.284609	192.168.1.165	255.255.255.255	78:31:c1:bd:5e:24	ff:ff:ff:ff:ff:ff	67 68	DHCP	342	DESKTOP-E7D8H40	MSFT 5.0
58	385.285249	192.168.1.100	192.168.1.165	b8:27:eb:2b:90:f1	78:31:c1:bd:5e:24	68 67	DHCP	342		

Options: 253 Parameter Request List
Length: 13
Parameter Request List Item (1) Subnet Mask
Parameter Request List Item (15) Domain Name
Parameter Request List Item (3) Router
Parameter Request List Item (6) Domain Name Server
Parameter Request List Item (44) NetBIOS over TCP/IP Name Server
Parameter Request List Item (46) NetBIOS over TCP/IP Node Type
Parameter Request List Item (47) NetBIOS over TCP/IP Scope
Parameter Request List Item (31) Perform Router Discover
Parameter Request List Item (33) Static Route
Parameter Request List Item (121) Classless Static Route
Parameter Request List Item (249) Private/Classless Static Route (Microsoft)
Parameter Request List Item (252) Private/Proxy autodiscovery
Parameter Request List Item (43) Vendor-Specific Information





DHCP: Secure?



- DHCP **does not include** any mechanism for **authentication**
- Vulnerable to attacks
 - Cannot really trust the response (a 'rouge' DHCP server could respond and tell hosts malicious information such as a DNS server or gateway)
 - Malicious clients can easily exhaust DHCP server resources such as the pool of available IP addresses



Take-Home [1/2]



- Facts
 - Cryptographic protocols or protocols that support encryption may carry certain plaintext information
 - Still a great deal of network protocols carry plaintext information
- Plaintext information can expose information about you, your habits, the devices you use, their features and software



Take-Home [2/2]



- TLS, DNS, mDNS, DNS-SD, SSDP, DHCP are just a few examples
- Make sure you trust the networks you connect to, and you trust those who connect to your networks
- Remove personal information from your devices (e.g., Simone's MacBook Pro)
- Use of VPN and DoH/DoT at minimum