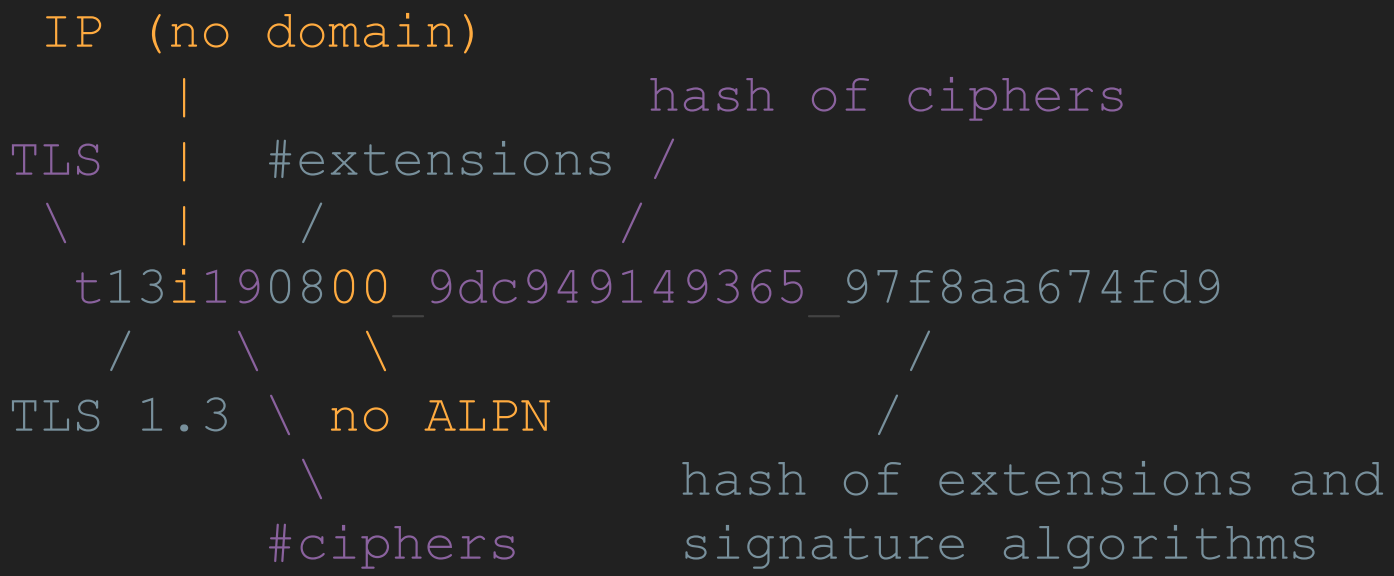
Erik Hjelmvik
Jonas Lejon

Netnod Tech Meeting
Oct 2024

What is this?

```
t13i190800_9dc949149365_97f8aa674fd9
```

It’s a JA4 fingerprint of Sliver!

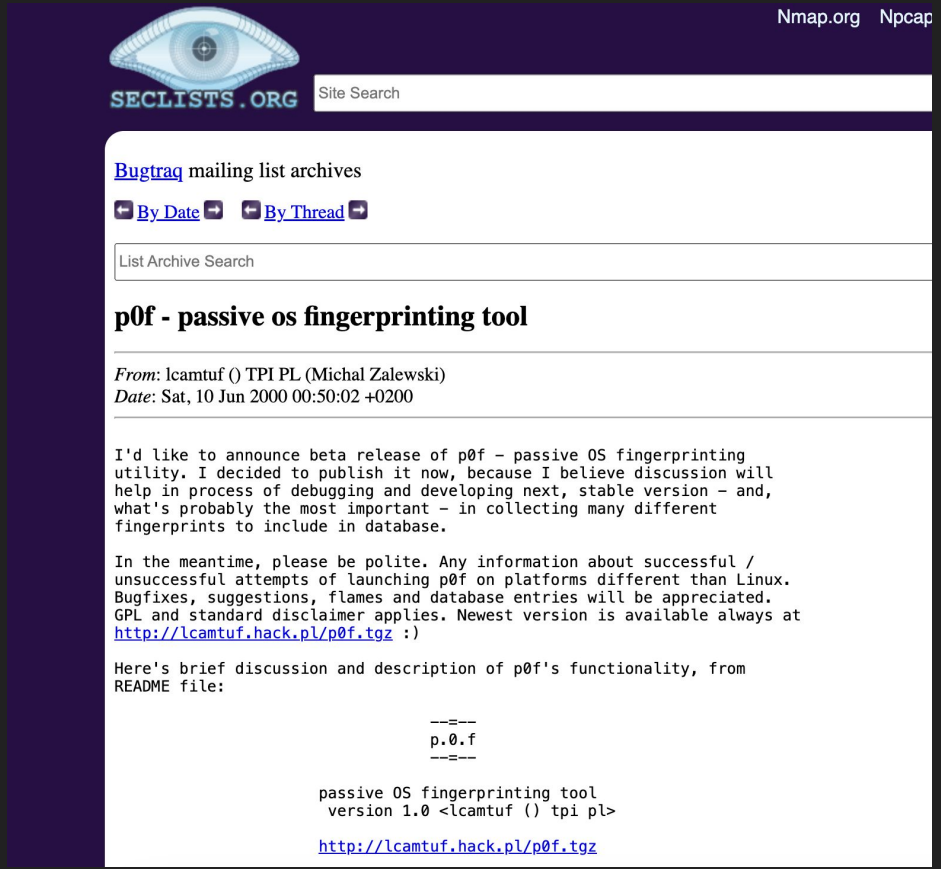


Abous us

- Erik Hjelmvik
  CTO at Netresec
- Jonas Lejon
  Cybersecurity specialist at Triop

History

- p0f - Passive
  - Announced by Michal Zalewski in 2000
- JA3/S
  - Announced in 2017
- Nmap - Active
  - OS detection since 1998
  - Protocol detection



What is Network Fingerprinting?

- Identifying applications
- Identifying operating systems
- Identifying communication libraries
- Devices

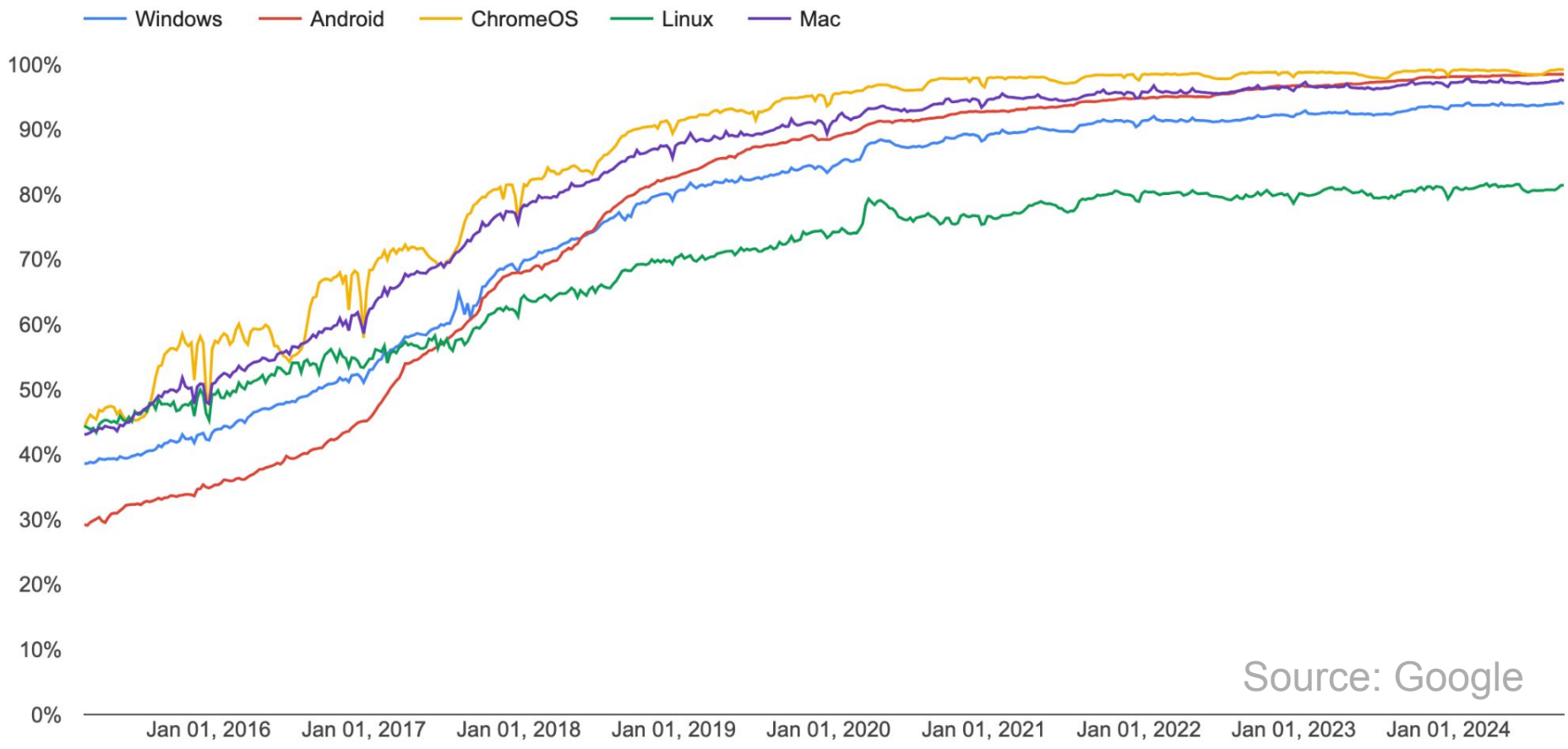
How?

- Passive (sniffing packets)
- Active (sending packets)

Why?

- Almost all traffic on the internet is encrypted
- Malware identification
  - Backdoors, Command and Control traffic and servers
- Anti censorship identification
  - Tor
- Web Application Firewalls and Intrusion Detection Systems
- DoS/DDoS prevention
- QoS

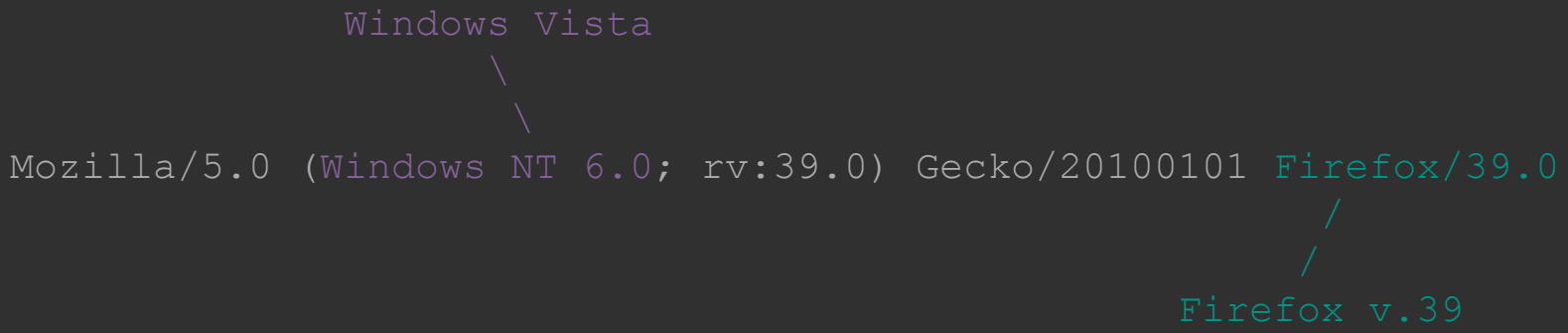
Percentage of pages loaded over HTTPS in Chrome by platform



What is this?

```
Mozilla/5.0 (Windows NT 6.0; rv:39.0) Gecko/20100101 Firefox/39.0
```

It’s a browser user-agent string!



Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅

Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅
Windows 2000	Windows NT 5.0	

Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅
Windows 2000	Windows NT 5.0	
Windows XP	Windows NT 5.1	
Windows Vista	Windows NT 6.0	
Windows 7		

Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅
Windows 2000	Windows NT 5.0	
Windows XP	Windows NT 5.1	
Windows Vista	Windows NT 6.0	
Windows 7	Windows NT 6.1	WTF?

Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅
Windows 2000	Windows NT 5.0	
Windows XP	Windows NT 5.1	
Windows Vista	Windows NT 6.0	
Windows 7	Windows NT 6.1	WTF?
Windows 8	Windows NT 6.2	

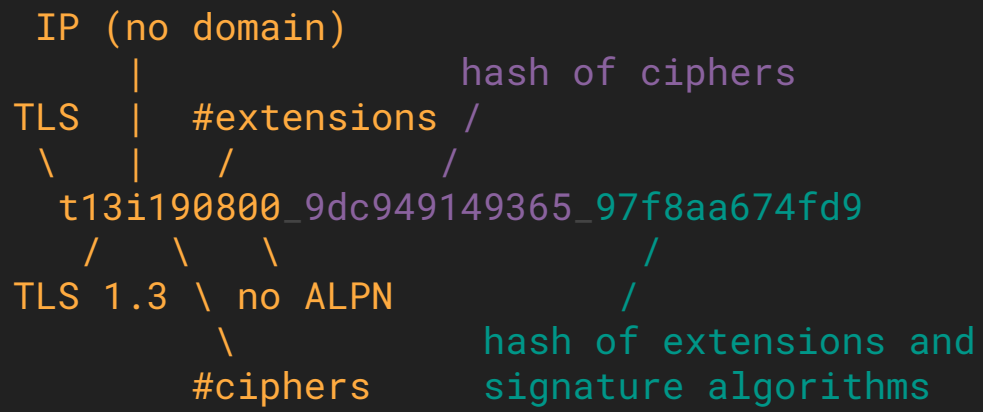
Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅
Windows 2000	Windows NT 5.0	
Windows XP	Windows NT 5.1	
Windows Vista	Windows NT 6.0	
Windows 7	Windows NT 6.1	WTF?
Windows 8	Windows NT 6.2	
Windows 10	Windows NT 10.0	✅

Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅
Windows 2000	Windows NT 5.0	
Windows XP	Windows NT 5.1	
Windows Vista	Windows NT 6.0	
Windows 7	Windows NT 6.1	WTF?
Windows 8	Windows NT 6.2	
Windows 10	Windows NT 10.0	✅
Windows 11	Windows NT 10.0	WTF?

Operating System	Platform Token	
Windows NT 4.0	Windows NT 4.0	✅
Windows 2000	Windows NT 5.0	
Windows XP	Windows NT 5.1	
Windows Vista	Windows NT 6.0	
Windows 7	Windows NT 6.1	WTF?
Windows 8	Windows NT 6.2	
Windows 10	Windows NT 10.0	✅
Windows 11	Windows NT 10.0	WTF?
Windows 12	(???)	

JA3 vs. JA4

	JA3	JA4
Input	TLS Version, Cipher Suites, Extensions, Elliptic Curves, EC Point Formats	Everything in JA3 plus: SNI flag, ALPN value, signature algorithms
Hash	MD5	Raw_SHA256_SHA256
Example	a85be79f7b569f1df5e6087b69deb493	t13i010400_0f2cb44170f4_5c4c70b73fa0



What to Fingerprint?

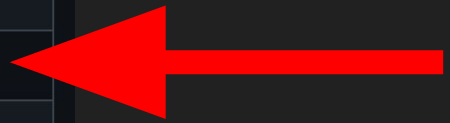
Input	Tools / Fingerprints	Output
L7 protocol	User-Agent, JA4HTTP, httprint, NIDS signatures	Application or Malware
TLS	JA3, JA4	TLS library Sometimes application / OS
IP / TCP / DHCP	p0f, Satori, JA4TCP	Operating System
Ethernet / WiFi	OUI	NIC Manufacturer / Hardware

JA4+

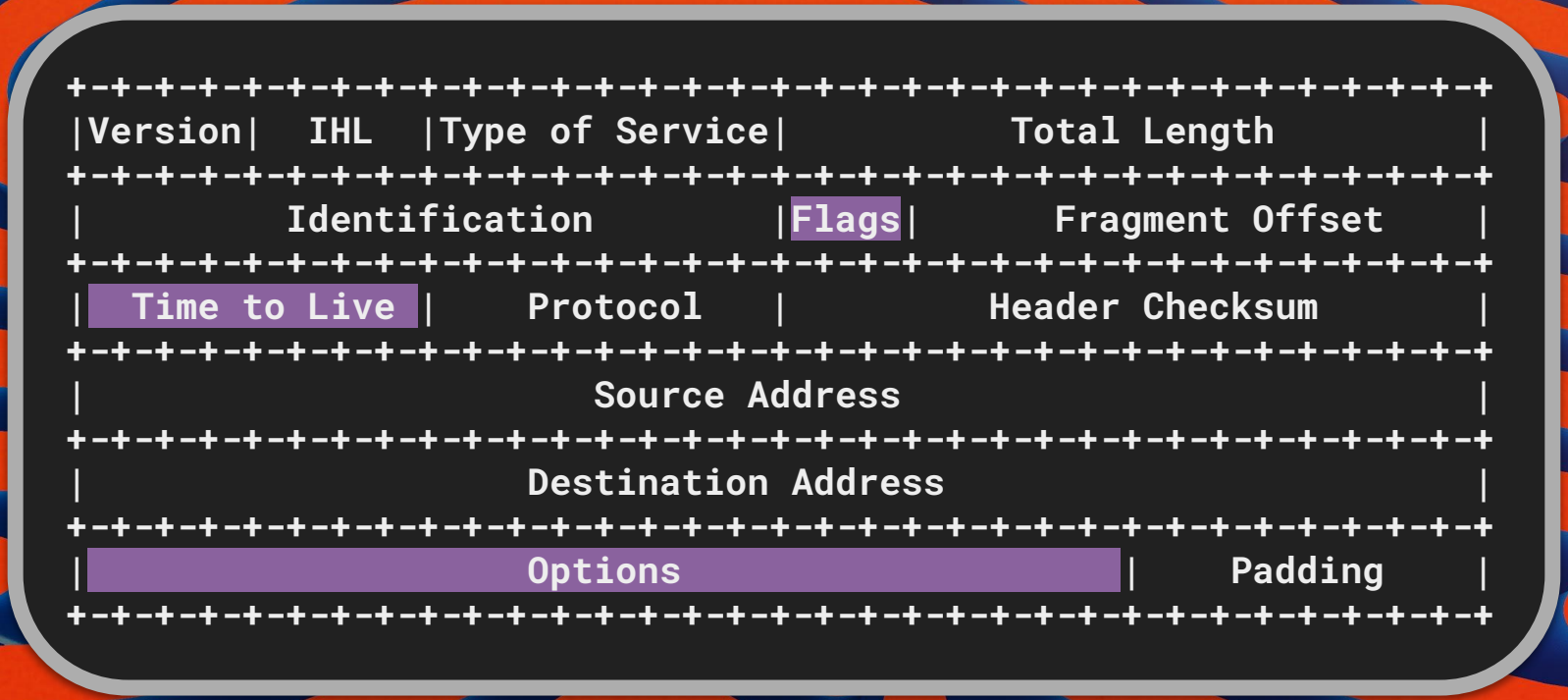
Full Name	Short Name	Description
JA4	JA4	TLS Client Fingerprinting
JA4Server	JA4S	TLS Server Response / Session Fingerprinting
JA4HTTP	JA4H	HTTP Client Fingerprinting
JA4Latency	JA4L	Client to Server Latency Measurment / Light Distance
JA4LatencyServer	JA4LS	Server to Client Latency Measurement / Light Distance
JA4X509	JA4X	X509 TLS Certificate Fingerprinting
JA4SSH	JA4SSH	SSH Traffic Fingerprinting
JA4TCP	JA4T	TCP Client Fingerprinting
JA4TCPServer	JA4TS	TCP Server Response Fingerprinting
JA4TCPScan	JA4TScan	Active TCP Fingerprint Scanner

JA4+

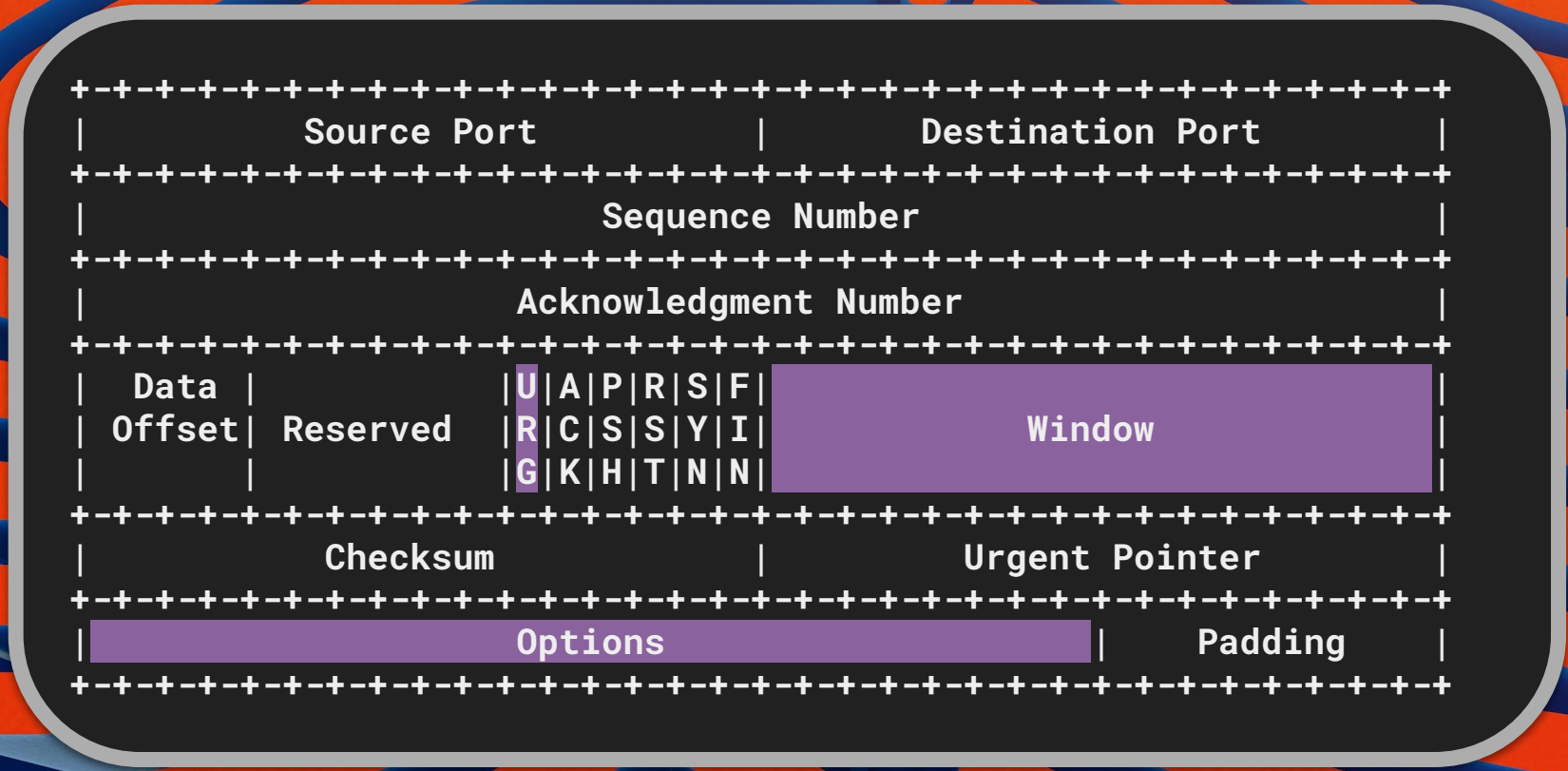
Full Name	Short Name	Description
JA4	JA4	TLS Client Fingerprinting
JA4Server	JA4S	TLS Server Response / Session Fingerprinting
JA4HTTP	JA4H	HTTP Client Fingerprinting
JA4Latency	JA4L	Client to Server Latency Measurment / Light Distance
JA4LatencyServer	JA4LS	Server to Client Latency Measurement / Light Distance
JA4X509	JA4X	X509 TLS Certificate Fingerprinting
JA4SSH	JA4SSH	SSH Traffic Fingerprinting
JA4TCP	JA4T	TCP Client Fingerprinting
JA4TCPServer	JA4TS	TCP Server Response Fingerprinting
JA4TCPScan	JA4TScan	Active TCP Fingerprint Scanner



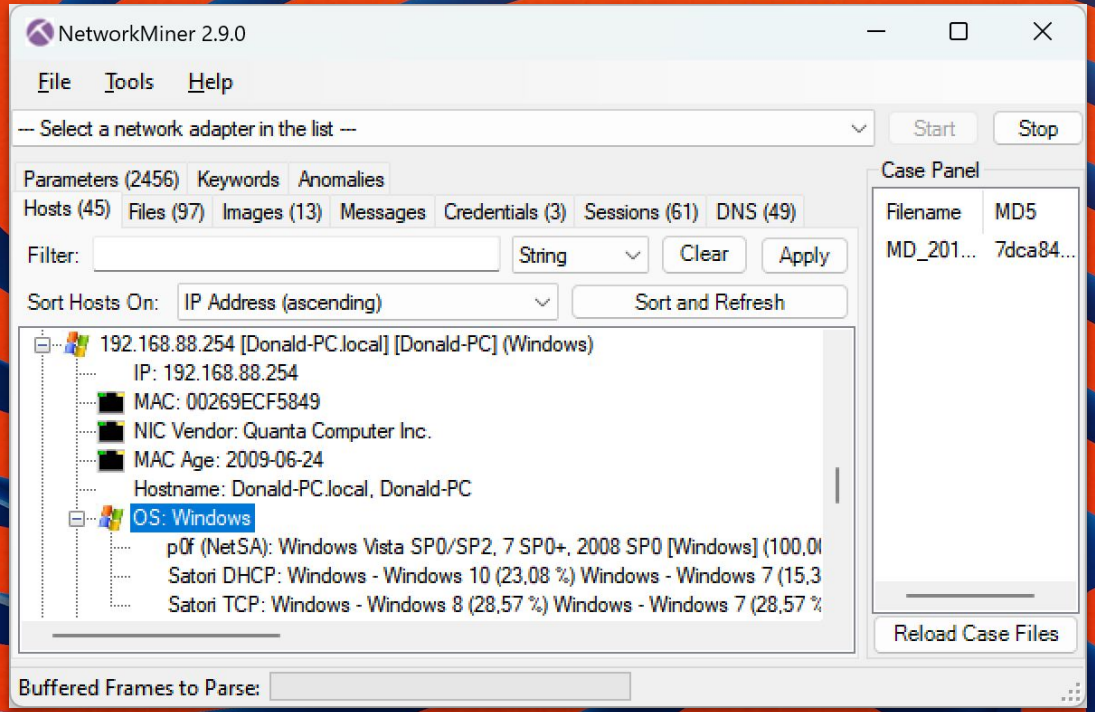
IP (RFC 791)


Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options
Padding

TCP (RFC 793)


Source Port
Destination Port
Sequence Number
Acknowledgment Number
Data Offset
Reserved
U R G
A C K
P S H
R S T
S Y N
F I N
Window
Checksum
Urgent Pointer
Options
Padding

OS fingerprinting with TCP/IP



p0f:
https://lcamtuf.coredump.cx/p0f3/

Carnegie Mellon CERT's p0f:
https://tools.netsa.cert.org/p0f/index.html

Satori TCP/DHCP:
https://github.com/xnih/satori

Active Fingerprinting

- JARM - TLS Fingerprinting
  - Sends 10 TLS Client Hello packets
  - 62 character fingerprint
- Looks something like this:

00000000000000000043d43d00043de2a97eabb398317329f027c66e4c1b01

(Sliver gRPC admin listening on default TCP port 31337)

```
➜  ~ git clone https://github.com/salesforce/jarm.git
Cloning into 'jarm'...
remote: Enumerating objects: 102, done.
remote: Counting objects: 100% (62/62), done.
remote: Compressing objects: 100% (23/23), done.
remote: Total 102 (delta 53), reused 39 (delta 39), pack-reused 40 (from 1)
Receiving objects: 100% (102/102), 38.17 KiB | 640.00 KiB/s, done.
Resolving deltas: 100% (54/54), done.
➜  ~ cd jarm
➜  jarm git:(master) python3 jarm.py -p 31337 [redacted]
Domain: [redacted]
Resolved IP: [redacted]
JARM: 00000000000000000043d43d00043de2a97eabb398317329f027c66e4c1b01
➜  jarm git:(master) python3 jarm.py -p 8834 [redacted]
Domain: [redacted]
Resolved IP: [redacted]
JARM: 2ad2ad0002ad2ad00042d42d000000020120996177a65431cde640fa58d2e8
➜  jarm git:(master)
```

Active Fingerprinting

Scan 1

```
JARM: 00000000000000000043d43d00043de2a97eabb398317329f027c66e4c1b01
Scan 1: |||,
Scan 2: |||,
Scan 3: |||,
Scan 4: |||,
Scan 5: |||,
Scan 6: |||,
Scan 7: 1303|0303||002b-0033,
Scan 8: 1303|0303||002b-0033,
Scan 9: |||,
Scan 10: 1303|0303||002b-0033
```

hash of extensions

Active Fingerprinting

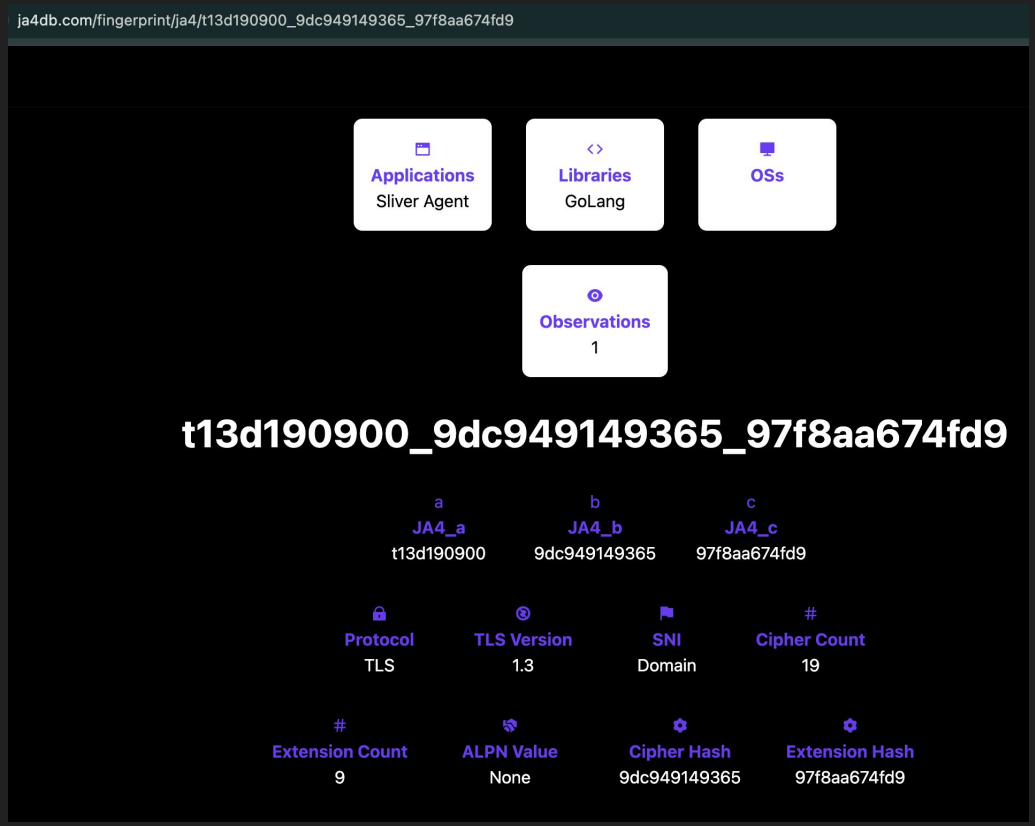
Scan 10

```
JARM: 00000000000000000043d43d00043de2a97eabb398317329f027c66e4c1b01
Scan 1: |||,
Scan 2: |||,
Scan 3: |||,
Scan 4: |||,
Scan 5: |||,
Scan 6: |||,
Scan 7: 1303|0303||002b-0033,
Scan 8: 1303|0303||002b-0033,
Scan 9: |||,
Scan 10: 1303|0303||002b-0033
```

hash of extensions

Databases

- On Github
- ja4db.com
- https://ja3.zone/
- Censys, BinadyEdge Shodan
  - JARM
- VirusTotal - JA3/JA3S
- sslbl.abuse.ch/ja3-fingerprints/



Final Words

- Most traffic is encrypted
- Passive fingerprinting often gives false positives
  - Combine with active fingerprinting
  - And other intel such as X509 attributes, IP, DNS etc..
- The Internet is full of “middleboxes”
- TLS 1.3 = more fingerprints due to more extensions

Anti fingerprinting

- Chrome sends extensions in random order
  - Works with JA4 but not JA3
- Cobalt Strike (C&C) TLS stack matches Java (server side)
  - Client uses Windows TLS stack
- Sliver C&C https implant listener
  - -E, --disable-randomized-jarm disable randomized jarm fingerprints
- Traffic analysis
  - Maybenot, a framework for traffic analysis defenses (Pulls and Witwer)
    - Tor Circuit Padding Framework
  - OBFS4, Snowflake, meek-azure