Hunting Threats That Use Encrypted Network Traffic

June 30th, 2020

Introduction

Peter Manev -

QA/Trainer/Exec team - OISF

CSO - Stamus Networks

Twitter @pevma

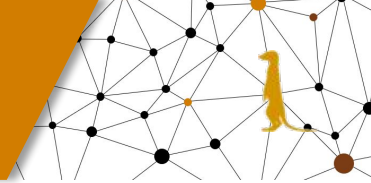
Eric Leblond -

Developer/Trainer/Exec team - OISF

CTO - Stamus Networks

Twitter @regiteric

Suricata



- Suricata is a high-performance network IDS, IPS and network security monitoring (NSM) engine
- Open-source software
- Owned and developed by a community run non-profit foundation - Open Information Security Foundation (OISF)
- Produces a high-level of situational awareness and detailed application layer transaction records

SURICATA

What We Will Cover

Outputs used for encrypted traffic hunting:

- TLS events (TLS 1, 1.2, 1.3)
- Anomaly events (new in 6.+)
- Alerts
- JA3/JA3s correlation

OSS tools used in this webinar for visualizing the outputs:

- ELK/SELKS6
- Scirius CE
- EveBox
- Moloch

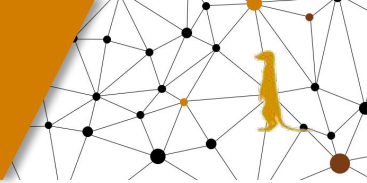
The TLS Handshake

- Begins with a handshake
  - Asymmetric encryption: two different keys are used
  - AKA public-key cryptography
- Public-key:
  - Server makes this available publicly
- Private-key
  - Secret, only used on the server side
- Data encrypted with public key can only be decrypted by private key

TLS support in Suricata

- TLS handshake analysis
  - Decode TLS message
  - Extract information
  - Output information in dedicated events
    - A JSON event for each connection
- And more features
  - Allow alerting on fields via dedicated keywords
  - Certificate chain extraction
- Additional methods:
  - JA3: algorithm to identify the client by its implementation
  - JA3S: algorithm to identify the server by its implementation

TLS log example



Tls: iecvlist.microsoft.com - CN=*.vo.msecnd.net


```
{
  "tls" : {
    "notbefore" : "2020-03-18T19:52:29"
    "issuerdn" : "C=US, ST=Washington, L=Redmond, O=Microsoft Corporation, OU=Microsoft IT, CN=Microsoft IT TLS CA 2"
    "ja3" : {
      "hash" : "1074895078955b2db60423ed2bf8ac23"
      "string" : "771,49192-49191-49172-49171-159-158-157-156-49196-49195-49188-49187-49162-49161-61-60-53-47-106-64-56-50-10-19-5-4,0-10-11-13-23-65281,23-24-25,0"
    }
    "notafter" : "2022-03-18T19:52:29"
    "version" : "TLS 1.2"
    "fingerprint" : "5e:aa:64:b5:7d:5b:63:50:03:84:f0:e6:55:a8:a5:d8:dd:8c:ff:33"
    "serial" : "1C:00:14:5F:23:03:6B:BC:E6:2F:3F:2C:56:00:00:00:14:5F:23"
    "sni" : "iecvlist.microsoft.com"
    "ja3s" : {
      "hash" : "0cac51a1efd65f5b6c047f539d24313e"
      "string" : "771,49192,65281-0-11-23"
    }
    "subject" : "CN=*.vo.msecnd.net"
  }
```

TLS keywords

- Match on fields in the certificate
  - tls.cert_issuer
  - tls.cert_subject
  - tls.cert_fingerprint
  - tls.sni
  - ...
- Examples
  - Check your usage of internal PKI
    - alert tls any any -> $SERVERS any (tls.cert_issuer; content:!"Cn=my,OU=awesome,O=company"; sid:1; rev:1;)
  - Pin your main server fingerprint
    - Alert tls any any -> $AUTH_SERVER any (tls.cert_fingerprint; content:!"22:33:44:55:66"; sid:2; rev:1;)

What do you get ? (demo)

- Demo in SELKS hunting interface
- TLS alerts
  - Certificate information
  - JA3 information
- TLS events
  - Pivot by flow_id

Real life detection on TLS

- Certificate by default, because bad guy ...likes lazy

```
alert tls $EXTERNAL_NET any -> $HOME_NET any (msg:"ET POLICY OpenSSL Demo CA - Internet Widgits Pty (O)"; flow:established,to_client; tls_cert_subje
ct; content:"O=Internet Widgits Pty Ltd"; metadata: former_category POLICY; classtype:not-suspicious; sid:2011540; rev:6; metadata:created_at 2010_0
9_27, updated_at 2017_11_27;)
```

- JA3 phishing

ET JA3 Hash - Possible Malware - Banking Phish

```
alert tls $HOME_NET any -> $EXTERNAL_NET any (msg:"ET JA3 Hash - Possible Malware - Banking Phish"; ja3_hash; conten
t:"10ee8d30a5d01c042afd7b2b205facc4"; metadata: former_category JA3; reference:url,github.com/trisulnsm/trisul-script
s/blob/master/lua/frontend_scripts/reassembly/ja3/prints/ja3fingerprint.json; reference:url,www.malware-traffic-analys
is.net; classtype:unknown; sid:2028362; rev:2; metadata:created_at 2019_09_10, updated_at 2019_10_29;)
```

SURICATA

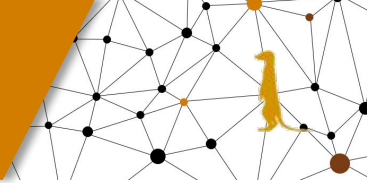
Some existing sig list

- Abuse.ch
  - Fingerprint
  - ja3
- ET Open/Pro using TLS
  - TlS rules
  - JA3 hash rules
  - SSL Blacklist rules

TLS JA3 algorithm

- Client sends TLS Client Hello after TCP session established
  - Packet and the way in which it is generated is dependent on packages and methods used when building the client application
- Server responds with TLS Server Hello
  - Similar to client, respond depends on how software was built and data sent from client
- Negotiations are sent in the clear and allow for fingerprinting
  - Still compatible with TLS 1.3

JA3 example



Tls: iecvlist.microsoft.com - CN=*.vo.msecnd.net


```
{
  "tls" : {
    "notbefore" : "2020-03-18T19:52:29"
    "issuerdn" : "C=US, ST=Washington, L=Redmond, O=Microsoft Corporation, OU=Microsoft IT, CN=Microsoft IT TLS CA 2"
    "ja3" : {
      "hash" : "1074895078955b2db60423ed2bf8ac23"
      "string" : "771,49192-49191-49172-49171-159-158-157-156-49196-49195-49188-49187-49162-49161-61-60-53-47-106-64-56-50-10-19-5-4,0-10-11-13-23-65281,23-24-25,0"
    }
    "notafter" : "2022-03-18T19:52:29"
    "version" : "TLS 1.2"
    "fingerprint" : "5e:aa:64:b5:7d:5b:63:50:03:84:f0:e6:55:a8:a5:d8:dd:8c:ff:33"
    "serial" : "1C:00:14:5F:23:03:6B:BC:E6:2F:3F:2C:56:00:00:00:14:5F:23"
    "sni" : "iecvlist.microsoft.com"
    "ja3s" : {
      "hash" : "0cac51a1efd65f5b6c047f539d24313e"
      "string" : "771,49192,65281-0-11-23"
    }
    "subject" : "CN=*.vo.msecnd.net"
  }
```

Interesting JA3 ressources

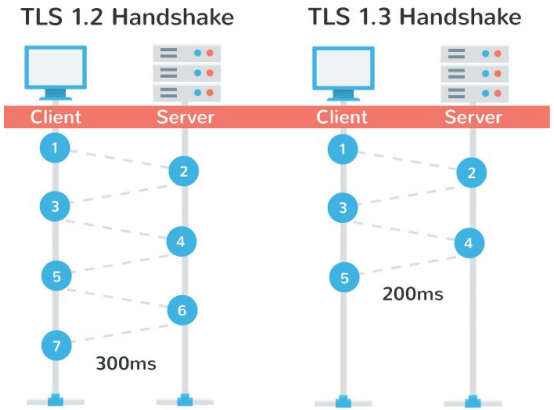
- Mapping to TLS user agent
  - Get a name of user agent behind the hash
  - Building a list of hashes to TLS user agent
    - By experiment
  - Example: https://ja3er.com/downloads.html
- Abuse.ch JA3 list
- Use your Suricata to generate (cleanlist/alertlist) hashes

TLS JA3S

- Method of creating a fingerprint from the server side of the TLS handshake – TLS Server Hello
- Decimal values of the bytes for the following fields:
  - Version, Accepted Cipher, and List of Extensions
  - Concatenated and delimited as JA3
- Resulting value is hashed with MD5
- Server doesn't always respond the same to all clients
  - But responds the same to the same client

TLS 1.3

- Better security
  - Improved privacy
    - Hide as much data as possible
    - Prevent interception
  - Remove deprecated algorithms (SHA1, RC4, …)
  - Encrypt most of the negotiation
- Faster



(image source: https://kinsta.com/blog/tls-1-3/)

TLS 1.3 event example


```
  ],
  "in_iface": "eth1",
  "tls": {
    "sni": "app.pendo.io",
    "ja3s": {
      "string": "771,4865,51-43",
      "hash": "eb1d94daa7e0344597e756a1fb6e7054"
    },
    "version": "TLS 1.3",
    "ja3": {
      "string": "771,4865-4866-4867-49195-49199-49196-49200-52393-52392-49171-49172-156-157-47-53-10,0-23-65281-10-11-35-16-5-13-18-51-45-43-27-21,29-23-24,0",
      "hash": "66918128f1b9b03303d77c6f2eefd128"
    }
  },
```

```
  "in_iface": "eth1",
  "tls": {
    "sni": "app.pendo.io",
    "ja3s": {
      "string": "771,4865,51-43",
      "hash": "eb1d94daa7e0344597e756a1fb6e7054"
    },
    "version": "TLS 1.3",
    "ja3": {
      "string": "771,4865-4866-4867-49195-49199-49196-49200-523
      "hash": "66918128f1b9b03303d77c6f2eefd128"
    }
  },
  "see_name": "SSPLAB",
  "@version": "1",
  "offset": 499967217,
```

What's remaining for Suricata in TLS 1.3

TLS data

- JA3
- JA3S
- TLS Server Name Indication
  - But there is draft on encryption…

Flow entries

- Data à la Netflow
  - In and Out volume and packets count
  - Enriched with
    - Application layer identification
    - Tunnel information

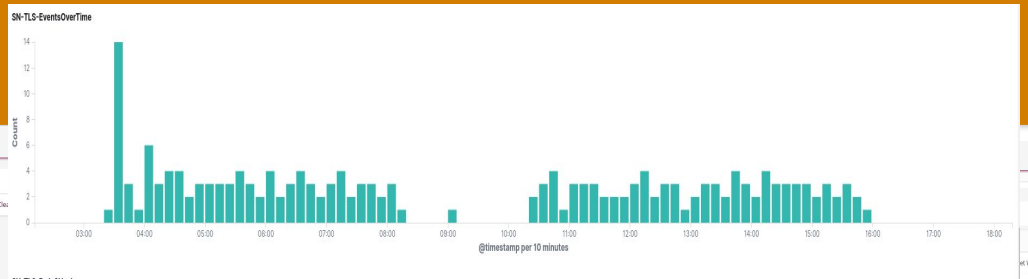
Flow event in Suricata

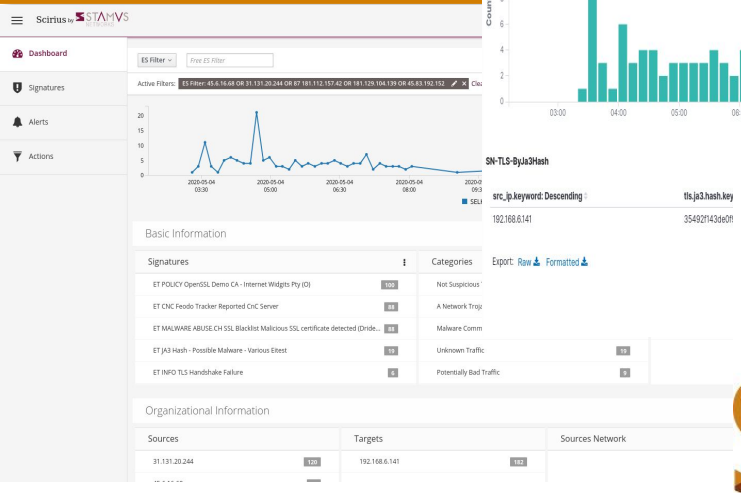
- Complete data
  - To server side
  - To client side
- Duration and timestamps
  - Start timestamp
  - End timestamp
  - Duration
- TCP flags
- Flow Identification
  - Flow_id for suricata cross event
  - Community_id for cross system correlation


```
{
  "timestamp": "2018-07-23T21:21:08.915073+0200",
  "thread_id": 1,
  "flow_id": 1550404452389229,
  "event_type": "flow",
  "src_ip": "172.16.1.117",
  "src_port": 34152,
  "dest_ip": "172.16.1.130",
  "dest_port": 4433,
  "proto": "TCP",
  "app_proto": "tls",
  "flow": {
    "pkts_toserver": 7,
    "pkts_toclient": 6,
    "bytes_toserver": 843,
    "bytes_toclient": 1987,
    "start": "2018-07-23T21:21:08.889197+0200",
    "end": "2018-07-23T21:21:08.915073+0200",
    "age": 0,
    "state": "closed",
    "reason": "shutdown",
    "alerted": false,
    "wrong_thread": true
  },
  "community_id": "1:O1+CVbJkgApsa3kcHF/ZFvgQmW0=",
  "tcp": {
    "tcp_flags": "1b",
    "tcp_flags_ts": "1b",
    "tcp_flags_tc": "1b",
    "syn": true,
    "fin": true,
    "psh": true,
    "ack": true,
    "state": "closed"
  }
}
```


SURICATA

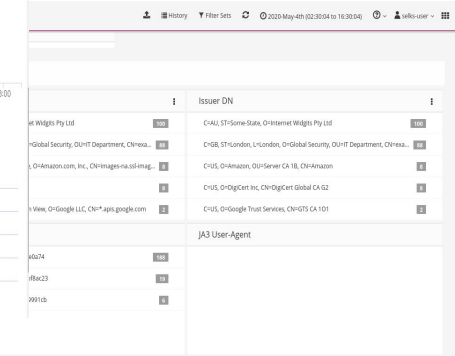
Trickbot

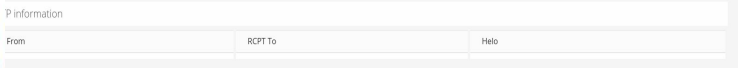

SN-TLS-EventsOverTime
Count
@timestamp per 10 minutes

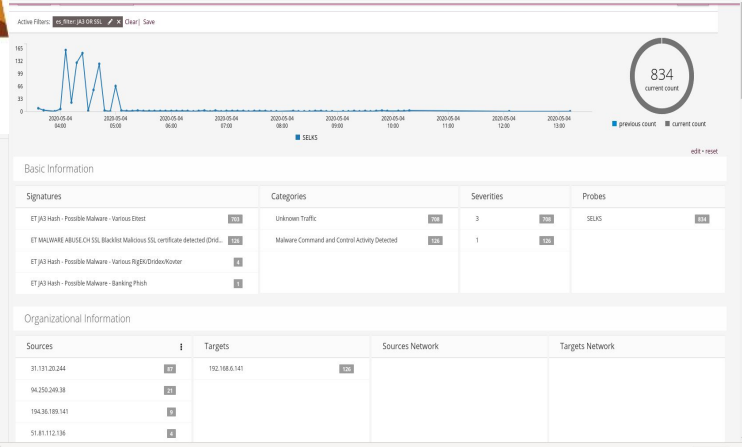

Scirius by STAMUS
Dashboard
Signatures
Alerts
Actions
Basic Information
Signatures
Categories
Organizational Information
Sources
Targets
Sources Network

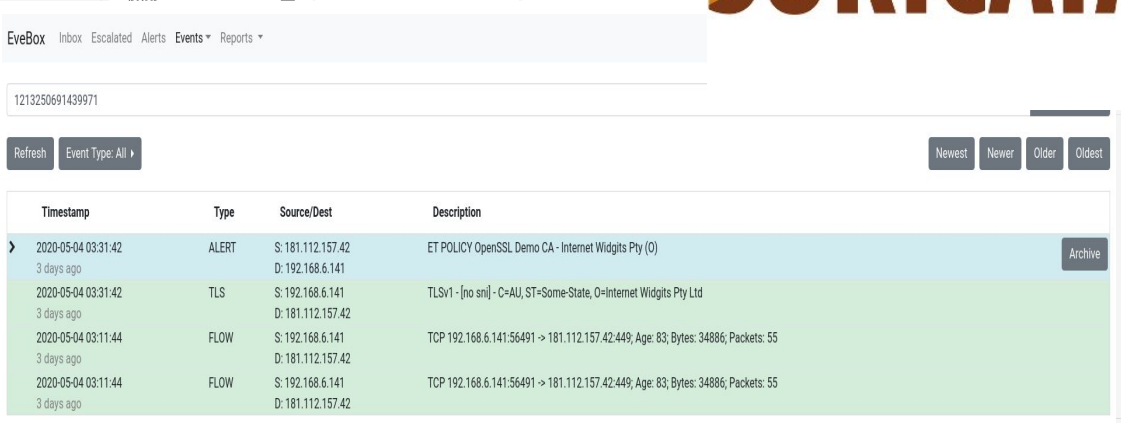
SN-TLS-ByJa3Hash
src_ip.keyword: Descending
tls.ja3.hash.key
192.168.6.141
Export: Raw Formatted


tls.ja3s.hash.keyword: Descending
Count
f2e1706526fe0692ee36be58110ffc83
f2e1706526fe0692ee36be58110ffc83
f2e1706526fe0692ee36be58110ffc83
f2e1706526fe0692ee36be58110ffc83
f2e1706526fe0692ee36be58110ffc83


Issuer DN
JA3 User-Agent


IP information
From
RCPT To
Helo


834
current count
Basic Information
Signatures
Categories
Severities
Probes
Organizational Information
Sources
Targets
Sources Network
Targets Network

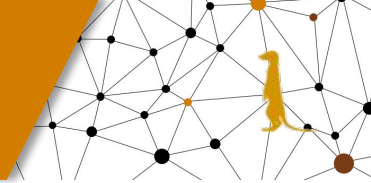

EveBox
Inbox
Escalated
Alerts
Events
Reports
1213250691439971
Refresh
Event Type: All
Newest
Newer
Older
Oldest
Timestamp
Type
Source/Dest
Description
2020-05-04 03:31:42
3 days ago
ALERT
S: 181.112.157.42
D: 192.168.6.141
ET POLICY OpenSSL Demo CA - Internet Widgits Pty (O)
Archive
2020-05-04 03:31:42
3 days ago
TLS
S: 192.168.6.141
D: 181.112.157.42
TLSv1 - [no sni] - C=AU, ST=Some-State, O=Internet Widgits Pty Ltd
2020-05-04 03:11:44
3 days ago
FLOW
S: 192.168.6.141
D: 181.112.157.42
TCP 192.168.6.141:56491 -> 181.112.157.42:449; Age: 83; Bytes: 34886; Packets: 55
2020-05-04 03:11:44
3 days ago
FLOW
S: 192.168.6.141
D: 181.112.157.42
TCP 192.168.6.141:56491 -> 181.112.157.42:449; Age: 83; Bytes: 34886; Packets: 55

Conclusion

- Who said IDS is dead again ?
  - TLS is a serious challenge
  - Visibility is decreasing
  - BUT Suricata can still do efficient analysis
- Come out and play
  - Pcap: https://github.com/jstrosch/malware-samples/tree/master/binaries/trickbot/2020/May
  - Suricata forums/help/discussions: https://forum.suricata.io/
  - Suricata trainings/webinars: https://suricata-ids.org/training/
  - OISF: https://oisf.net/
  - SELKS 6: https://www.stamus-networks.com/selks-6

Annex

JA3 & JA3S with Suricata



Fingerprint	94:84:7b:d6:25:6f:3c:d4:df:95:08:95:d9:fd:c2:8f:7e:14:b4:31
Issuerdn	C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
Ja3.Hash	1d095e68489d3c535297cd8dffb06cb9
Ja3.String	769,47-53-5-10-49171-49172-49161-49162-50-56-19-4,65281-0-10-11,23-24,0
Notafter	2020-04-15T23:05:24
Notbefore	2020-01-16T23:05:24
Serial	03:FA:C2:98:B0:E4:16:CB:E2:66:27:7C:63:CC:03:5B:3C:E9

SURICATA

Using decryption

Where to put Suricata

- Behind SSL load balancers
- FWs/GWs can decrypt traffic and mirror it to a port
  - McAfee, Cisco, Palo Alto, Juniper...
- Behind/next to proxies
- Most important is to be able to see the traffic as end clients are

The JA3 Hash

- Decimal values of the byte values of the following fields are concatenated from client hello
  - Version, Accepted Ciphers, List of Extensions, Elliptic Curves, and Elliptic Curve Formats
- Concatenated in order using a “,” and a “-” to delimit values in fields
  - If no values the fields are left empty
- Result is then hashed using MD5