Analyzing Encrypted Traffic

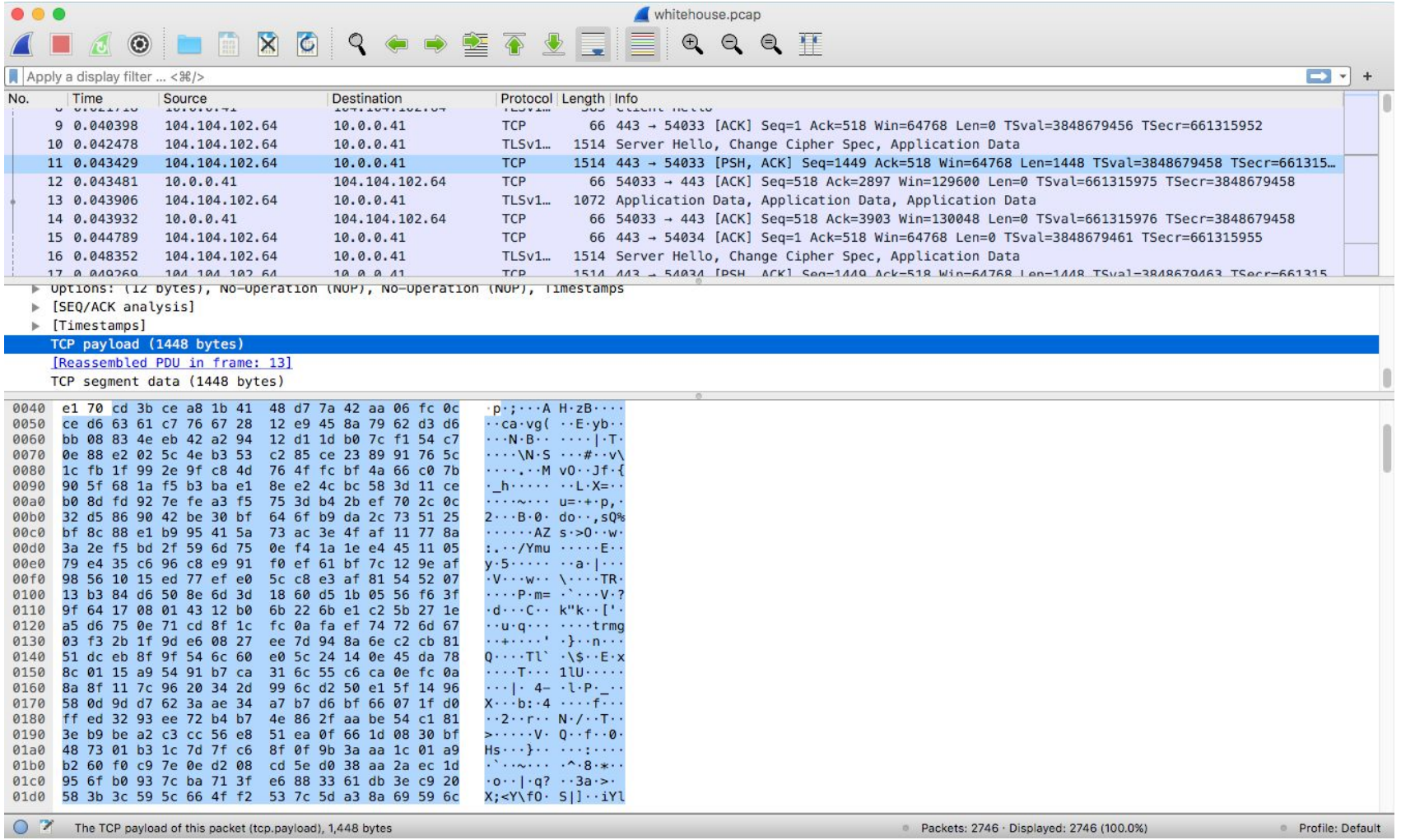
Corelight and Active Countermeasures

Welcome!

- Who we are
- Today's goals
  - Quick coverage of encrypted traffic
  - How this affects Zeek, RITA, and AC-Hunter
  - How to handle encrypted traffic
- Threat Hunter Community Discord
  - https://discord.gg/6tHmJCtc
  - #live-webcast-chat
  - #acm-webcast-content
    - PDF of these slides

Encrypted traffic on the wire

- Can see headers
  - IP addresses
  - Protocol, ports, flags
- Can't see payload
  - Request placed or destination URL
  - Malware, watermarks, spam
  - Hurts signature IDS


whitehouse.pcap
Apply a display filter ... <⌘/>
No. Time Source Destination Protocol Length Info
9 0.040398 104.104.102.64 10.0.0.41 TCP 66 443 → 54033 [ACK] Seq=1 Ack=518 Win=64768 Len=0 TSval=3848679456 TSecr=661315952
10 0.042478 104.104.102.64 10.0.0.41 TLSv1… 1514 Server Hello, Change Cipher Spec, Application Data
11 0.043429 104.104.102.64 10.0.0.41 TCP 1514 443 → 54033 [PSH, ACK] Seq=1449 Ack=518 Win=64768 Len=1448 TSval=3848679458 TSecr=661315…
12 0.043481 10.0.0.41 104.104.102.64 TCP 66 54033 → 443 [ACK] Seq=518 Ack=2897 Win=129600 Len=0 TSval=661315975 TSecr=3848679458
13 0.043906 104.104.102.64 10.0.0.41 TLSv1… 1072 Application Data, Application Data, Application Data
14 0.043932 10.0.0.41 104.104.102.64 TCP 66 54033 → 443 [ACK] Seq=518 Ack=3903 Win=130048 Len=0 TSval=661315976 TSecr=3848679458
15 0.044789 104.104.102.64 10.0.0.41 TCP 66 443 → 54034 [ACK] Seq=1 Ack=518 Win=64768 Len=0 TSval=3848679461 TSecr=661315955
16 0.048352 104.104.102.64 10.0.0.41 TLSv1… 1514 Server Hello, Change Cipher Spec, Application Data
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
[SEQ/ACK analysis]
[Timestamps]
TCP payload (1448 bytes)
[Reassembled PDU in frame: 13]
TCP segment data (1448 bytes)
0040 e1 70 cd 3b ce a8 1b 41 48 d7 7a 42 aa 06 fc 0c
0050 ce d6 63 61 c7 76 67 28 12 e9 45 8a 79 62 d3 d6
0060 bb 08 83 4e eb 42 a2 94 12 d1 1d b0 7c f1 54 c7
0070 0e 88 e2 02 5c 4e b3 53 c2 85 ce 23 89 91 76 5c
0080 1c fb 1f 99 2e 9f c8 4d 76 4f fc bf 4a 66 c0 7b
0090 90 5f 68 1a f5 b3 ba e1 8e e2 4c bc 58 3d 11 ce
00a0 b0 8d fd 92 7e fe a3 f5 75 3d b4 2b ef 70 2c 0c
00b0 32 d5 86 90 42 be 30 bf 64 6f b9 da 2c 73 51 25
00c0 bf 8c 88 e1 b9 95 41 5a 73 ac 3e 4f af 11 77 8a
00d0 3a 2e f5 bd 2f 59 6d 75 0e f4 1a 1e e4 45 11 05
00e0 79 e4 35 c6 96 c8 e9 91 f0 ef 61 bf 7c 12 9e af
00f0 98 56 10 15 ed 77 ef e0 5c c8 e3 af 81 54 52 07
0100 13 b3 84 d6 50 8e 6d 3d 18 60 d5 1b 05 56 f6 3f
0110 9f 64 17 08 01 43 12 b0 6b 22 6b e1 c2 5b 27 1e
0120 a5 d6 75 0e 71 cd 8f 1c fc 0a fa ef 74 72 6d 67
0130 03 f3 2b 1f 9d e6 08 27 ee 7d 94 8a 6e c2 cb 81
0140 51 dc eb 8f 9f 54 6c 60 e0 5c 24 14 0e 45 da 78
0150 8c 01 15 a9 54 91 b7 ca 31 6c 55 c6 ca 0e fc 0a
0160 8a 8f 11 7c 96 20 34 2d 99 6c d2 50 e1 5f 14 96
0170 58 0d 9d d7 62 3a ae 34 a7 b7 d6 bf 66 07 1f d0
0180 ff ed 32 93 ee 72 b4 b7 4e 86 2f aa be 54 c1 81
0190 3e b9 be a2 c3 cc 56 e8 51 ea 0f 66 1d 08 30 bf
01a0 48 73 01 b3 1c 7d 7f c6 8f 0f 9b 3a aa 1c 01 a9
01b0 b2 60 f0 c9 7e 0e d2 08 cd 5e d0 38 aa 2a ec 1d
01c0 95 6f b0 93 7c ba 71 3f e6 88 33 61 db 3e c9 20
01d0 58 3b 3c 59 5c 66 4f f2 53 7c 5d a3 8a 69 59 6c
The TCP payload of this packet (tcp.payload), 1,448 bytes
Packets: 2746 · Displayed: 2746 (100.0%)
Profile: Default

Common types

- HTTPS (TLS/SSL)
  - and smtps, imaps, pop3s….
- VPN
  - ipsec, openvpn, others
- SSH
  - including all tunneled traffic
- DNS over TLS, DNS over HTTPS...

Unencrypted DNS

- Originally unencrypted
  - UDP and TCP port 53
  - Multicast DNS (UDP port 5353)
  - LLMNR (UDP and TCP port 5355)
- Can identify *who* even if we can't see *what*
- Netbios

Encrypted DNS

- Encrypted options
  - DNS over TLS (TCP port 853)
  - DOH (DNS over HTTPS (TCP port 443)
  - In construction: Oblivious DOH, DNS over QUIC
- Tougher to see *who*
- Blog on Name lookup
  - https://www.activecountermeasures.com/alternative-dns-techniques/

Options

- Not causing problems for security tools
  - Leave traffic as is
- Causing visibility problems
  - Block entirely
    - Must be able to identify with a firewall/IDS
  - Force through a proxy
    - Traffic unmodified, to log connections
    - Decrypt, inspect, encrypt, send on
      - Needs client trust
      - Privacy issues
- Provide service internally

Identifying Encrypted Traffic

- SSH
  - "SSH-"
- TLS
  - Can identify encryption types
- Zeek can find these and flag them

Analysis (Threat Hunting)

- Can't see *what was said*
  - Payload and metadata
- Can still see:
  - Beacons/Strobes
  - Long connections
  - Connections to Threat Intel hosts
  - TLS
    - Ja3 encryption negotiation (client signature)
    - certificate
- DNS - depends

Encrypted Traffic in Zeek logs


```
{ [-]
  _path: x509
  _system_name: corelight-suricata-demo
  _write_ts: 2020-10-30T02:48:57.615642Z
  basic_constraints.ca: false
  certificate.curve: prime256v1
  certificate.issuer: CN=GTS CA 1O1,O=Google Trust Services,C=US
  certificate.key_alg: id-ecPublicKey
  certificate.key_length: 256
  certificate.key_type: ecdsa
  certificate.not_valid_after: 2020-09-23T03:47:22.000000Z
  certificate.not_valid_before: 2020-07-01T03:47:22.000000Z
  certificate.serial: 0CF54C4F0C98EA9C08000000004AAB73
  certificate.sig_alg: sha256WithRSAEncryption
  certificate.subject: CN=upload.video.google.com,O=Google LLC,L=Mountain View,ST=California,C=US
  certificate.version: 3
  id: FfYNRg39iyaCYKxVq5
  san.dns: [ [-]
    upload.video.google.com
    *.clients.google.com
    *.docs.google.com
    *.drive.google.com
    *.gdata.youtube.com
    *.googleapis.com
    *.photos.google.com
    *.upload.google.com
    *.upload.youtube.com
    *.youtube-3rd-party.com
    upload.google.com
    upload.youtube.com
    uploads.stage.gdata.youtube.com
  ]
  ts: 2020-10-30T02:48:57.615642Z
}
```



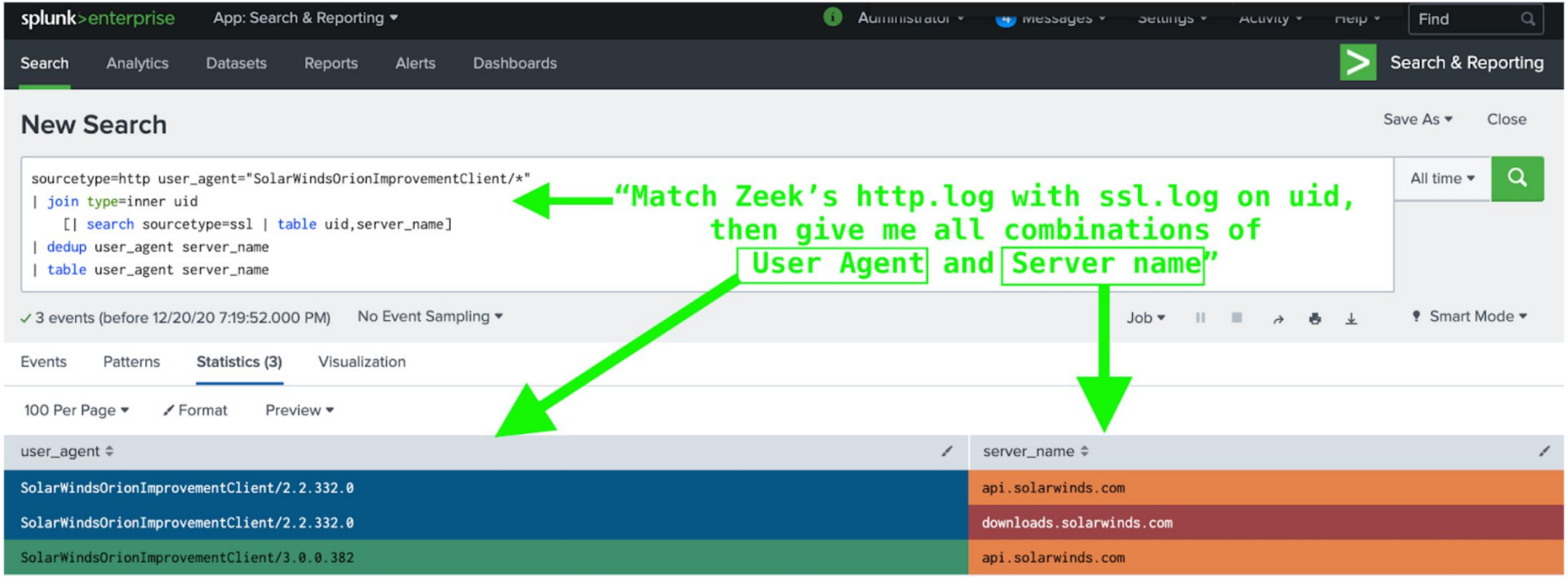
```
{ [-]
  _path: ssl
  _system_name: corelight-suricata-demo
  _write_ts: 2020-10-30T02:48:57.615649Z
  cert_chain_fuids: [ [-]
    FfYNRg39iyaCYKxVq5
    F65RLd2WJouvSnWLrj
  ]
  cipher: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
  client_cert_chain_fuids: [ [-]
  ]
  curve: secp256r1
  encrypted_dns_resp_h: false
  established: true
  id.orig_h: 10.7.17.103
  id.orig_p: 49573
  id.resp_h: 172.217.164.131
  id.resp_p: 443
  issuer: CN=GTS CA 1O1,O=Google Trust Services,C=US
  ja3: 205200cdaac61b110838556b834070d1
  ja3s: 84aaf6d03fc8c5bfb56d1d188735b268
  resumed: false
  server_name: update.googleapis.com
  subject: CN=upload.video.google.com,O=Google LLC,L=Mountain View,ST=California,C=US
  ts: 2020-10-30T02:48:57.615566Z
  uid: CxQ2Dv4ZL9tqKfBlac
  validation_status: certificate has expired
  version: TLSv12
}
```

Exploit Detection in Certificates

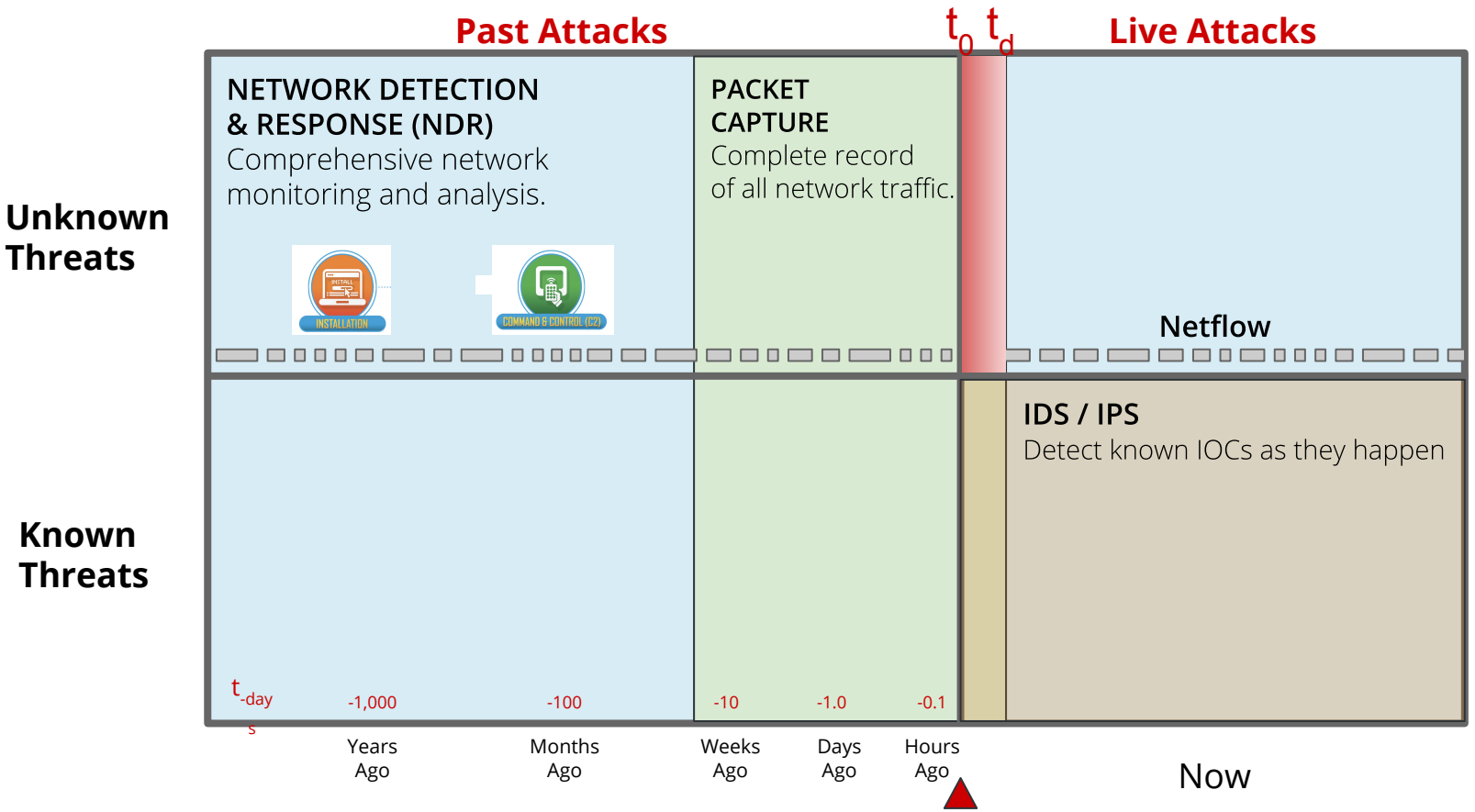
```
35  event x509_certificate(f: fa_file, cert_ref: opaque of x509, cert: X509::Certificate)
36          {
37          if ( cert?$key_alg && cert$key_alg == "id-ecPublicKey" && ! cert?$curve )
38                  {
39                  NOTICE([$note=Unknown_X509_Curve, $f=f, $msg="ECC certificate with unknown curve; potential CVE-2020-0601 exploit attempt"]);
40
41                  if ( log_certs )
42                          Log::write(CVE_2020_0601::LOG, Info($ts=network_time(), $fuid=f$id, $certificate=encode_base64(x509_get_certificate_string(cert_ref, F))));
43                  }
44          }
```

```
17  event ssl_session_ticket_handshake(c: connection, ticket_lifetime_hint: count, ticket: string)
18          {
19          if ( /^..\x00{16}.../ in ticket && |ticket| > 56 && bytestring_to_count(sub_bytes(ticket, 35, 2))+56 == |ticket| )
20                  NOTICE([$note=CVE_2020_13777_Server, $conn=c, $msg="Server potentially vulnerable to CVE-2020-13777 detected", $identifier=cat(c$id$orig_h)]);
21          }
22
23  event ssl_change_cipher_spec(c: connection, is_orig: bool) &priority=-5
24          {
25          if ( ! is_orig || ! c$ssl$resumed )
26                  return;
27
28          if ( c$ssl$gnutls_ch )
29                  NOTICE([$note=CVE_2020_13777_Resumed, $conn=c, $msg="Server potentially vulnerable to CVE-2020-13777 detected; client resumed with suspicious
```

Sunburst Anomaly



Retrospective Detection Made Easy



SSH Visibility Starts In Open Source


```
   auth_attempts: 0
   cipher_alg: aes128-ctr
   client: SSH-2.0-libssh2_1.4.3
   compression_alg: none
   cshka: ssh-rsa,ssh-dss
   direction: INBOUND
   hassh: 92674389fa1e47a27ddd8d9b63ecd42b
   hasshAlgorithms: diffie-hellman-group14-sha1,diffie-hellman
cbc,rijndael-cbc@lysator.liu.se,aes192-cbc,aes128-cbc,blowfish
96,hmac-ripemd160,hmac-ripemd160@openssh.com;none
   hasshServer: cca34b641961a75a15b91d1f1a13a3fb
   hasshServerAlgorithms: ecdh-sha2-nistp256,ecdh-sha2-nistp38
sha1,diffie-hellman-group14-sha1,diffie-hellman-group1-sha1;ae
gcm@openssh.com,aes128-cbc,3des-cbc,blowfish-cbc,cast128-cbc,a
sha1-etm@openssh.com,umac-64-etm@openssh.com,umac-128-etm@open
etm@openssh.com,hmac-sha1-96-etm@openssh.com,hmac-md5-96-etm@o
256,hmac-sha2-512,hmac-ripemd160,hmac-ripemd160@openssh.com,hm
   hasshVersion: 1.1
   host_key: 24:ca:ee:e1:84:b3:0f:1a:17:86:c0:72:0a:8c:61:f6
   host_key_alg: ssh-rsa
```


Home » **Corelight Labs** » Detecting OpenBSD CVE-2019-19521 SSH exploit attempts

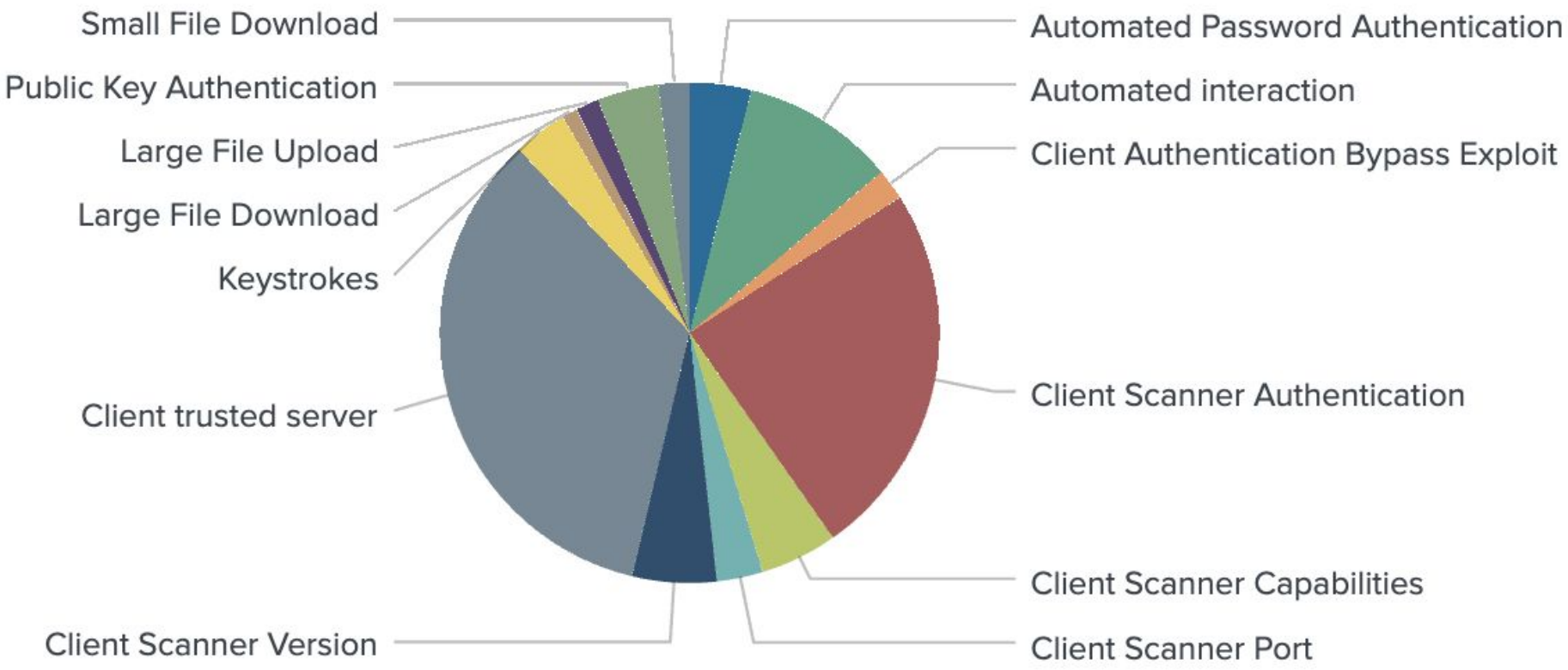
Detecting OpenBSD CVE-2019-19521 SSH exploit attempts

December 6, 2019 by Anthony Kasza

GitHub

master ▾ zeek / scripts / policy / protocols / ssh /

And Continues With Corelight's ETC



Encrypted Traffic: RITA, AC-Hunter

- Most Threat Hunt techniques still work
  - Beacons, Strobes, Long connections, Threat Intel, Client Signature, Certificate
- Potential issues
  - DNS: encrypted or not?
  - Unexpected Protocol

References and Questions

- https://assets.corelight.com/portals/jsnbhy7x/zeekposter
- https://corelight.blog/2020/12/22/detecting-sunburst-solarigate-activity-in-retrospect-with-zeek-a-practical-example/
- https://corelight.blog/2020/06/18/dns-over-tls-and-dns-over-https/
- https://corelight.blog/2020/06/16/the-light-shines-even-brighter-updates-to-corelights-encrypted-traffic-collection/
- https://corelight.blog/2020/05/13/analyzing-encrypted-rdp-connections/
- https://corelight.blog/2020/01/17/day-1-detection-cve-2020-0601/
- https://www.activecountermeasures.com/free-tools/rita/
- https://www.activecountermeasures.com/ac-hunter-features/
- https://www.corelight.com/about-zeek/how-zeek-works
- https://corelight.blog/2019/12/06/detecting-openbsd-cve-2019-19521-ssh-exploit-attempts/
- https://www.activecountermeasures.com/alternative-dns-techniques/
- https://corelight.blog/2019/11/20/introducing-corelight-encrypted-traffic-collection/

Thanks!

- Ed, Vijit, John, and Roger from Corelight
- Shelby from Active Countermeasures
- Presenters Alex and Bill