Packets don't lie - Threat Hunting with Zeek

Swapneel Patnekar
APNIC Community Trainer

Housekeeping

- If you wish to ask a question
  - Type your question in the Zoom Q&A
  - Unmute and ask!
- Webinar is being recorded

$about

- APNIC Community Trainer
- Founder & CEO of Shreshta IT
- Member of the Forum of Incident Response and Security Teams (FIRST) DNS Abuse SIG
- Program Committee member Indian Network Operators Group(INNOG) and APNIC conferences
- Past - Board member of India Internet Engineering Society (IIESoC)
- Investigate and report malicious domain names for takedowns

What is Network Security Monitoring?

- Network Security Monitoring is the collection, detection and analysis of network security data
- Three cycles - Collection, Detection and Analysis
- NSM does not prevent intrusions
- NSM data types
  - Full content - PCAP
  - Extracted content
  - Transaction data
  - Alert data - NIDS

Value of NSM

- Is data being exfiltrated from my network ?
- Is there a system in the network communicating/communicated with a C2C ?
- Profiling the network - Visibility
- What is happening on my network?

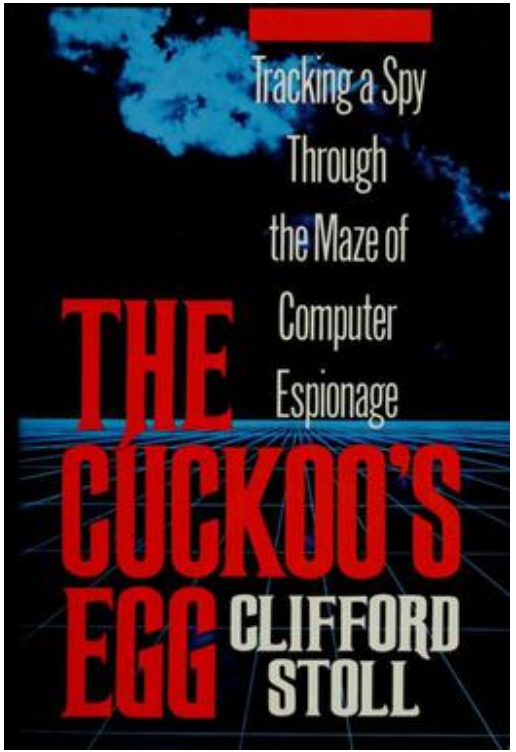
Date flow start	Len	Proto	Src IP Addr:Port		Dst IP Addr:Port	Packets	Bytes
Apr 13 2005 14:59:56	0	TCP	213.161.64.210:80	->	211.99.1.218:34156	5	828 B
Apr 13 2005 14:59:56	0	TCP	64.62.154.4:80	->	162.139.189.158:4527	3	140 B
Apr 13 2005 14:59:56	2	TCP	131.132.112.21:1138	->	64.18.47.234:80	5	637 B
Apr 13 2005 14:59:56	1	TCP	64.62.191.95:80	->	172.212.81.18:4390	5	493 B
Apr 13 2005 14:59:56	0	TCP	216.109.117.206:80	->	211.223.204.230:1132	3	266 B
Apr 13 2005 14:59:56	1	TCP	83.141.49.51:80	->	211.92.9.56:37157	42	57.0 KB
Apr 13 2005 14:59:48	5	TCP	191.210.93.172:80	->	149.194.8.73:3530	20	16.6 KB
Apr 13 2005 14:59:56	0	TCP	191.101.94.201:80	->	199.53.250.100:30267	5	633 B
Apr 13 2005 14:59:56	0	TCP	199.81.104.90:60553	->	213.161.61.209:80	6	803 B
Apr 13 2005 14:59:48	10	TCP	9.4.223.185:1433	->	168.150.251.37:22520	3	140 B

```
Sep 27 15:00:57 shreshtait sshd[2229376]: Invalid user user from 217.94.221.254 port 49483
Sep 27 15:02:11 shreshtait sshd[2229405]: Invalid user ngrc from 103.99.203.103 port 35526
Sep 27 15:12:25 shreshtait sshd[2229638]: Invalid user jesus from 167.172.207.63 port 43488
Sep 27 15:14:17 shreshtait sshd[2229667]: Invalid user deddy from 206.189.157.139 port 46082
Sep 27 15:45:50 shreshtait sshd[2230159]: Invalid user ts3bot from 103.70.144.140 port 34024
Sep 27 16:14:26 shreshtait sshd[2230621]: Invalid user james from 160.251.73.96 port 40224
Sep 27 16:35:07 shreshtait sshd[2230884]: Invalid user oracle from 134.209.50.147 port 49768
Sep 27 17:00:38 shreshtait sshd[2231297]: Invalid user oracle from 134.209.151.21 port 53640
Sep 27 17:00:38 shreshtait sshd[2231301]: Invalid user testuser from 134.209.151.21 port 53652
Sep 27 17:00:38 shreshtait sshd[2231295]: Invalid user admin from 134.209.151.21 port 53628
Sep 27 17:00:38 shreshtait sshd[2231302]: Invalid user admin from 134.209.151.21 port 53614
Sep 27 17:00:38 shreshtait sshd[2231296]: Invalid user admin from 134.209.151.21 port 53618
Sep 27 17:00:38 shreshtait sshd[2231303]: Invalid user admin from 134.209.151.21 port 53656
Sep 27 17:00:38 shreshtait sshd[2231304]: Invalid user ansible from 134.209.151.21 port 53662
Sep 27 17:00:38 shreshtait sshd[2231305]: Invalid user ubuntu from 134.209.151.21 port 53620
Sep 27 17:00:38 shreshtait sshd[2231307]: Invalid user admin from 134.209.151.21 port 53668
Sep 27 17:00:38 shreshtait sshd[2231300]: Invalid user deploy from 134.209.151.21 port 53644
Sep 27 17:00:38 shreshtait sshd[2231308]: Invalid user web from 134.209.151.21 port 53682
Sep 27 17:00:38 shreshtait sshd[2231313]: Invalid user minecraft from 134.209.151.21 port 53728
Sep 27 17:00:38 shreshtait sshd[2231309]: Invalid user admin from 134.209.151.21 port 53716
Sep 27 17:00:38 shreshtait sshd[2231316]: Invalid user admin from 134.209.151.21 port 53756
Sep 27 17:00:38 shreshtait sshd[2231317]: Invalid user devops from 134.209.151.21 port 53762
Sep 27 17:13:58 shreshtait sshd[2231629]: Invalid user Admin from 43.130.3.44 port 50500
```

2017-03-13 23:23:24 15 minutes ago	**S:** 151.65.181.96 **D:** 50.116.50.93	ET TELNET SUSPICIOUS busybox enable
2017-03-13 23:23:24 15 minutes ago	**S:** 151.65.181.96 **D:** 50.116.50.93	ET TELNET SUSPICIOUS busybox shell
2017-03-13 23:23:11 16 minutes ago	**S:** 183.214.141.101 **D:** 50.116.50.93	ET SCAN Potential SSH Scan
2017-03-13 23:22:56 16 minutes ago	**S:** 50.116.50.93 **D:** 125.212.203.45	GPL TELNET TELNET access
2017-03-13 23:22:33 16 minutes ago	**S:** 125.212.203.45 **D:** 50.116.50.93	ET TELNET busybox MIRAI hackers - Possible Brute Force Attack
2017-03-13 23:22:33 16 minutes ago	**S:** 125.212.203.45 **D:** 50.116.50.93	ET TELNET SUSPICIOUS Path to BusyBox
2017-03-13 23:22:33 16 minutes ago	**S:** 125.212.203.45 **D:** 50.116.50.93	ET TELNET SUSPICIOUS busybox enable
2017-03-13 23:22:33 16 minutes ago	**S:** 125.212.203.45 **D:** 50.116.50.93	ET TELNET SUSPICIOUS busybox shell

Origins of Zeek (Previously known as Bro IDS)

- Lawrence Berkeley National Laboratory
- The Cuckoo’s Egg - Clifford Stoll
- traceroute, tcpdump, libpcap



What is Zeek?

- Zeek is a passive, open-source network traffic analyzer
- Used as a network security monitor (out-of -band) to support investigations of suspicious or malicious activity
- Also supports a wide range of traffic analysis tasks, such as performance measurement and troubleshooting
- Runs on commodity hardware and provides a low-cost alternative to expensive proprietary solutions

Why Zeek?

- Transforms traffic into logs
  - Store Zeek logs for months or years
- Built for lighting-fast search and insight
  - Zeek's interlinked logs allow for instantaneous pivots
- Extracts all major files types
- Program custom behavioural detections via Zeek scripting language
- PCAP-like fidelity at 1% the data size
- Track attacker movement across ports and protocols and time

Zeek Architecture

- At a very high level, Zeek is architecturally layered into two major components
- Event engine (or core) - reduces the incoming packet stream into a series of higher-level events
- These events reflect network activity in policy-neutral terms, i.e., they describe what has been seen, but not why, or whether it is significant
- Event engine component comprises a number of subcomponents
  - Input sources
  - Packet analysis
  - Session analysis, and
  - File analysis

Zeek Log Formats

- Zeek creates a variety of logs when run in its default configuration.
  - Zeek TSV format logs
  - Zeek JSON format logs
- When we Zeek on any .pcap file, it creates different log files
- Run zeek -C -r file.pcap
  - Zeek completes its task without reporting anything to the command line
- Similarly, we can zeek logs in JSON format using the command below
  - zeek -C -r file.pcap LogAscii::use_json=T

conn.log

- The connection log, or conn.log, is one of the most important logs Zeek creates
- Zeek's conn.log, tracks both TCP and UDP protocols
- The conn.log primarily captures so-called "layer 3" and "layer 4" elements of network activity
- This is essentially who is talking to whom, when, for how long, and with what protocol
- cat conn.log
  or
  cat conn.log | jq . (JSON log format)

- The conn.log primarily captures so-called “layer 3” and “layer 4” elements of network activity.

- This is essentially who is talking to whom, when, for how long, and with what protocol.

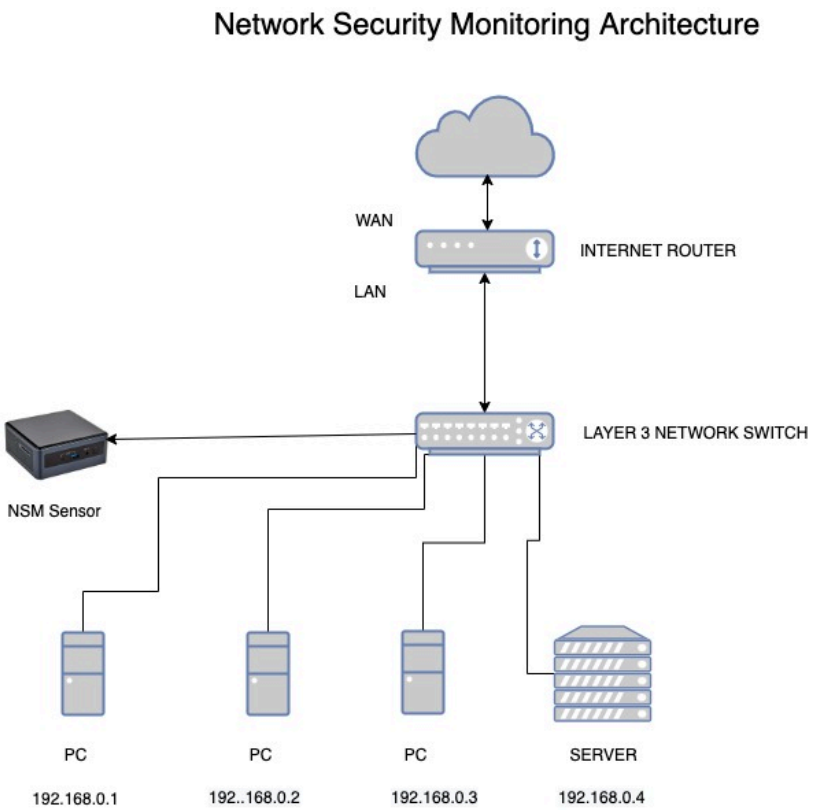

```
{
 "ts": 1591367999.305988,
 "uid": "CMdzit1AMNsmfAliQc",
 "id.orig_h": "192.168.4.76",
 "id.orig_p": 36844,
 "id.resp_h": "192.168.4.1",
 "id.resp_p": 53,
 "proto": "udp",
 "service": "dns",
 "duration": 0.06685185432434082,
 "orig_bytes": 62,
 "resp_bytes": 141,
 "conn_state": "SF",
 "missed_bytes": 0,
 "history": "Dd",
 "orig_pkts": 2,
 "orig_ip_bytes": 118,
 "resp_pkts": 2,
 "resp_ip_bytes": 197
}
```

dns.log

- dns.log captures application-level name resolution activity
- Applications mainly use DNS to resolve names to IP addresses, IP addresses to names, and certain other functions
- Even intruders use DNS for the same purposes
- Primary functions like who is asking a question, what is the nature of the question, who answered the question, and how was the question answered

Installation

- Binary Packages
- docker image - https://hub.docker.com/r/zeekurity/zeek
- Installation instructions
  - https://github.com/zeek/zeek-training/tree/master/Intro-to-Zeek22


Network Security Monitoring Architecture
WAN
INTERNET ROUTER
LAN
LAYER 3 NETWORK SWITCH
NSM Sensor
PC
192.168.0.1
PC
192..168.0.2
PC
192.168.0.3
SERVER
192.168.0.4

Let’s hunt for badness

- Huge props to these folks for making the PCAPs available to the community
- Stratosphere IPS Project
  - https://mcfp.felk.cvut.cz/publicDatasets/CTU-Malware-Capture-Botnet-25-3/
  - https://mcfp.felk.cvut.cz/publicDatasets/CTU-Malware-Capture-Botnet-347-1/
- Josh Stroschein
  - https://github.com/jstrosch/malware-samples/blob/master/binaries/vidar/2020/April/samples_pcap.zip
- Brad Duncan
  - https://www.malware-traffic-analysis.net/2018/CTF/index.html

Additional Resources

- Zeek Packages
  - https://packages.zeek.org
- Zeek Documentation
  - https://docs.zeek.org/en/master/logs/index.html
- PCAPs
  - https://www.netresec.com/?page=PcapFiles
- Brim - A desktop application with Zeek and Suricata builtin
  - https://www.brimdata.io

Questions?

- swapneel@brainattic.in
- @pswapneel
- https://www.linkedin.com/in/swapneel-patnekar/