UNIVERSITY OF AMSTERDAM

Portable Passive Detection of Advanced Persistent Threats

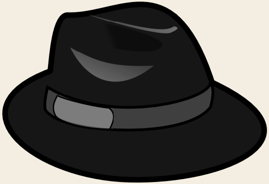
APT Catcher

Author: Guido Kroon

Supervisors: Marco Davids, Christian Hesselman (SIDN)

About Advanced Persistent Threats

- Advanced Persistent Threat (APT) [2];
- Highly skilled and well-resourced [17];
- Long duration of attack (months, years) [12][17];
- Specific motives, such as [12];
  - Intelligence gathering;
  - Financial enrichment;
- Not your average script kiddie.



Examples of Advanced Persistent Threats

- **Operation Aurora** (2010) - Source code theft of high profile targets, such as Google, Adobe and organisations in the defence and and financial sectors [19];
- **Stuxnet** (2010) - Israeli/United States joint effort, a computer worm specifically developed to attack the nuclear power programme in Iran [8];
- **Operation Shady RAT** (2011) - A large scale attack, targeted at more than 70 global companies, governments, and non-profit organisations for at least five years [1];
- **Belgacom breach** (2013) - The GCHQ breached Belgacom and had access to customer data, including encrypted and unencrypted streams of private communications [6].

Research questions

Main research question

Can a portable, passive Advanced Persistent Threat (APT) Catcher be designed to be easily deployed on the network which detects the presence of potential APTs?

Sub-questions

- What are the quantifiable characteristics of an APT?
- What methods are available to passively detect the presence of an APT?
- Can a prototype be designed to be deployed in an easy and feasible manner on the network to detect the presence of APTs?

Modus operandi I

	Kill Chain [12]	**Giura et al.** [7]	**Zero Entry Hacking** [5]
1	Reconnaissance	Reconnaissance	Reconnaissance
2	Development	Delivery	Scanning
3	Weaponisation	Exploitation	Exploitation
4	Delivery	Operation	Post exploitation and maintaining access
5	Exploitation	Data collection	
6	Installation	Exfiltration	
7	Command & Control		
8	Actions on objective		

Table: Several procedure models, which show a similar modus operandi.

Modus operandi II

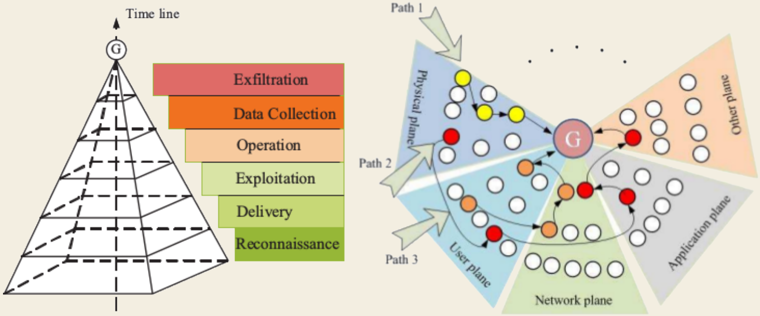


Figure: Attack pyramid [7].

Characteristics of the APT I

A typical APT has the following (non-exhaustive) characteristics [4][12][17][18]:

- **Inquisitive**: a strong desire to know as much as possible about the target. Lower hanging fruit would move to a new target when bored;
- **Stealthy approach**: circumventing all kinds of security controls to avoid detection. This also involves removing traces;
- **Preparation**: premeditated plan of execution by using newly acquired information;
- **Infiltration**: exploiting an asset to gain a foothold into the target. This may also involve social engineering (e.g. spear-phishing);

Characteristics of the APT II

- **Resourceful**: the APT is known for its sophisticated and custom designed attacks, such as self-built malware;
- **Exfiltration**: stealing as much confidential information as possible. The APT may use strong encryption to conceal the data being exfiltrated;

A natural born spy

The APT is a natural born spy that will stop at nothing to remain undetected, while carrying out its objective.

Detecting the APT I

- During active network scanning;
- During passive network scanning;
- During port scanning.

Detecting the APT II

- Host Intrusion Detection System (HIDS) (out of project scope);
  - OSSEC;
  - AIDE;
  - Samhain;
- Network Intrusion Detection System (NIDS);
  - Signature Based IDS (SBS);
  - Anomaly Based IDS (ABS).

Detecting the APT III

- Examples of NIDSs;
  - **Snort** - Most popular open source SBS NIDS, developed since 1998. Large community, with frequent signature updates [15];
  - **Suricata** - Open source SBS NIDS with multi-threading, hardware acceleration, IP reputation system, developed since 2009. Compatible with Snort rules[1], as well as their own rules[2] [14][16];
  - **Sagan** - Open source SBS NIDS / SIEM developed since 2011. Multi-threading support and has its own ruleset [13];
  - **Bro** - Advanced open source ABS NIDS, with behavioural network analysis, and its own script language to write detection parameters [3];
  - **PSAD** - Open source SBS NIDS. Scans `iptables` logs for suspicious behaviour [9].

[1] The Talos ruleset (formerly VRT)

[2] Emerging Threats Suricata ruleset

Designing the APT Catcher I

- Client/server architecture;
  - Sensor (prototype);
  - Aggregator.

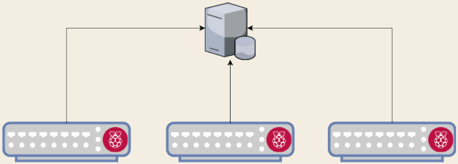


Figure: Client / server architecture.

Designing the APT Catcher II

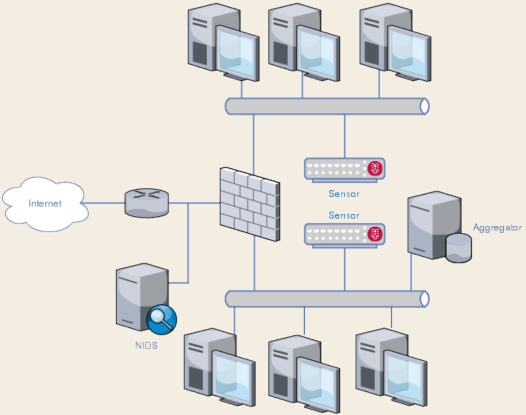


Figure: A more detailed overview of the APT Catcher within a network infrastructure.

Designing the APT Catcher III

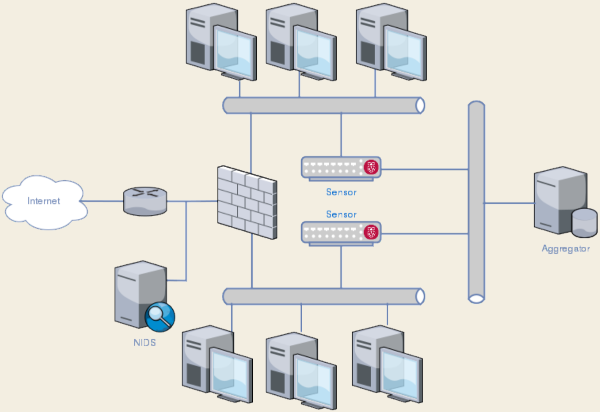


Figure: A new separate network for the sensors and the aggregator. Events are now sent exclusively over this network.

The sensor

- Portable;
- Heterogeneous detection with multiple sensors;
- Working prototype on a Raspberry Pi 3, using Docker.

Single board computer	Raspberry Pi 3
Processor	1.2 GHz 64-bit quad-core ARM Cortex-A53
Memory	1 GB (shared with GPU)
NIC	10/100 Mbit/s Ethernet
Operating System	Raspbian Jessie Lite [11]
Software	Docker v1.11, Unbound v1.5.9

Table: Raspberry Pi 3 prototype running Raspbian with Docker.

The sensor prototype

Docker container equipped with the following:

Base image	resin/rpi-raspbian [10]
Operating System	Raspbian Jessie Lite [11]
NIDS Software	Bro v2.4.1, PSAD v2.2.3, Snort v2.9.7.0 and Suricata v3.1.
Miscellaneous tools	netsniff-ng v0.6.1, Nmap v7.12, tcpdump v4.7.4 and TShark v2.0.4.

Table: Custom built Raspberry Pi 3 sensor container running Raspbian using Docker.

The aggregator

- Collects alarms of the sensors;
- Some dashboards already exist for several NIDSs;
- No dashboard exists which aggregates all alarms from all NIDSs.

Field testing

- Measurements taken with Monitorix;
- Measured performance of NIDSs running in the container;
- Measured performance of an attack simulation.

Field testing - Bro

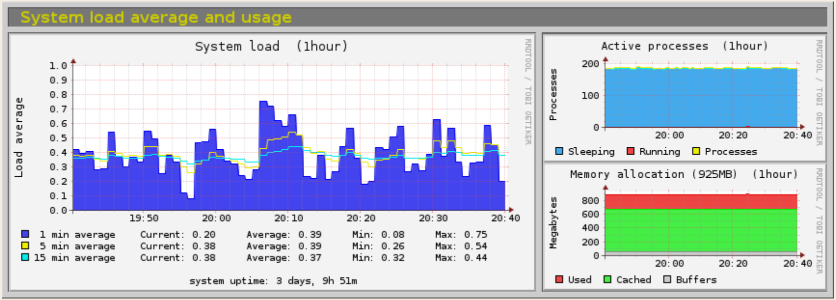


Figure: System load when Bro is running inside the APT Catcher sensor Docker container.

Field testing - Snort

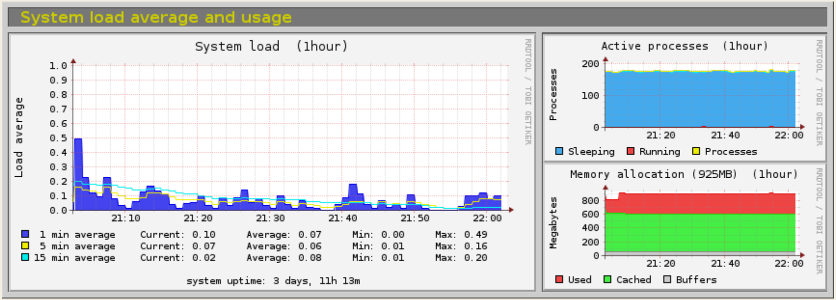


Figure: System load when Snort is running inside the APT Catcher sensor Docker container.

Field testing - Suricata

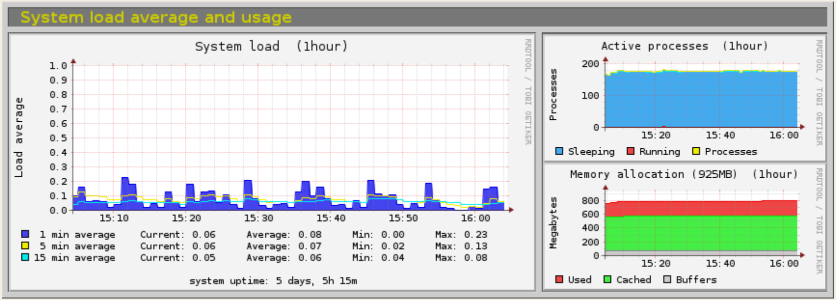


Figure: System load when Suricata is running inside the APT Catcher sensor Docker container.

Demonstration

Conclusion

- The APT is increasingly sophisticated, patient and stealthy;
- Detection of the APT causes a paradigm shift in defence strategies;
  - Don't just expect the threat at your door;
  - Expect them already in your home;
- The portable APT Catcher helps to detect such threats, in your home, continuously.

Questions?

?

Bibliography I

[1] Dmitri Alperovitch et al.
Revealed: operation shady RAT, volume 3.
McAfee, 2011.

[2] Beth Binde, Russ McRee, and Terrence J O'Connor.
Assessing outbound traffic to uncover advanced persistent threat.
SANS Institute. Whitepaper, 2011.

[3] Bro.
About Bro and the Bro Project.
`https://www.bro.org/documentation/faq.html`, 2016.
[Online; accessed 25-July-2016].

Bibliography II

[4] Terry Cutler.
The Anatomy of an Advanced Persistent Threat.
http://www.securityweek.com/anatomy-advanced-persistent-threat, 2010.
[Online; accessed 5-July-2016].

[5] Patrick Engebretson.
The basics of hacking and penetration testing: ethical hacking and penetration testing made easy, chapter 1, pages 14–18.
Elsevier, 2013.

Bibliography III

[6] Ryan Gallagher.
The Inside Story of How British Spies Hacked Belgium's Largest Telco.
https://theintercept.com/2014/12/13/belgacom-hack-gchq-inside-story/, 2016.
[Online; accessed 01-August-2016].

[7] Paul Giura and Wei Wang.
A context-based detection framework for advanced persistent threats.
In *Cyber Security (CyberSecurity), 2012 International Conference on*, pages 69–74. IEEE, 2012.

Bibliography IV

[8] Ralph Langner.
Stuxnet: Dissecting a cyberwarfare weapon.
IEEE Security & Privacy, 9(3):49–51, 2011.

[9] Michael Rash.
psad: Intrusion Detection and Log Analysis with iptables.
`http://cipherdyne.org/psad/`, 2016.
[Online; accessed 25-July-2016].

[10] Raspbian.
Base image for the Raspberry Pi.
`https://hub.docker.com/r/resin/rpi-raspbian/`, 2016.
[Online; accessed 25-July-2016].

Bibliography V

[11] Raspbian.
Download Raspbian for Raspberry Pi.
https://www.raspberrypi.org/downloads/raspbian/, 2016.
[Online; accessed 25-July-2016].

[12] Dell SecureWorks.
Advanced Threat Protection with Dell SecureWorks Security Services.
http://www.secureworks.com/assets/pdf-store/white-papers/wp-advanced-threat-protection.pdf, 2016.
[Online; accessed 27-June-2016].

Bibliography VI

[13] Quadrant Information Security.
Sagan.
https://quadrantsec.com/sagan_log_analysis_engine/, 2016.
[Online; accessed 25-July-2016].

[14] Eric Smith.
Snort vs Suricata.
http://wiki.aanval.com/wiki/Snort_vs_Suricata, 2016.
[Online; accessed 25-July-2016].

Bibliography VII

[15] Snort.
Snort FAQ/Wiki.
`https://www.snort.org/faq`, 2016.
[Online; accessed 25-July-2016].

[16] Suricata.
Complete list of Suricata Features.
`https://suricata-ids.org/features/all-features/`, 2016.
[Online; accessed 25-July-2016].

[17] Colin Tankard.
Advanced persistent threats and how to monitor and deter them.
Network security, 2011(8):16–19, 2011.

Bibliography VIII

[18] J Vukalović and D Delija.
Advanced persistent threats-detection and defense.
In *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015 38th International Convention on*, pages 1324–1330. IEEE, 2015.

[19] Kim Zetter.
Google hack attack was ultra sophisticated, new details show.
Wired Magazine, 14, 2010.