A case study about the use and evaluation of cyber deceptive methods against highly targeted attacks

Alexandria Farar; Hayretdin Bahsi; Bernhards Blumbergs

A. Farar, H. Bahsi and B. Blumbergs, "A case study about the use and evaluation of cyber deceptive methods against highly targeted attacks," 2017 International Conference On Cyber Incident Response, Coordination, Containment & Control (Cyber Incident), London, UK, 2017, pp. 1-7, doi: 10.1109/CYBERINCIDENT.2017.8054640.

A Case Study About the Use and Evaluation of Cyber Deceptive Methods Against Highly Targeted Attacks

Alexandria Farar and Hayretdin Bahşi
Department of Computer Science
Tallinn University of Technology
Tallinn, Estonia

Bernhards Blumbergs
Technology Branch
NATO CCDCOE
Tallinn, Estonia

***Abstract*—Traditional defences such as intrusion detection systems, firewalls and antivirus software are not enough to prevent security breaches caused by highly targeted cyber threats. As many of these attacks go undetected, this paper shows the results of a case study which consists of implementation of a methodology that selects, maps, deploys, tests and monitors the deceptions for the purpose of early detection. Metrics are developed to validate the effectiveness of the deception implementation. Firstly, various deception mechanisms are mapped to the first three phases of the intrusion kill chain: reconnaissance, weaponization and delivery. Then, Red Teams were recruited to test the deceptions for two case scenarios. Applying metrics, it is shown that the deceptions in the case studies are effective in the detection of cyber threats before the target asset was exploited and successful in creating attacker confusion and uncertainty about the organization's network topology, services and resources.**

Keywords— deception; honeypots; highly targeted attack; cyber kill chain;

I. INTRODUCTION

Many cyber espionage and cyber crime incidents occur due to the highly targeted attacks that have been conducted against organizations as well as individuals [1]. Although an advanced persistent threat similar to Stuxnet can be perceived as an extreme case in a wide spectrum of highly targeted attacks, the common denominator of these attacks is the requirement of a sophisticated level of expertise and substantial resources. In order to accomplish their mission, attackers use a meticulous approach when planning and implementing a targeted attack. Objectives usually entail establishing a foothold within the information technology infrastructure of the targeted organization, with a primary end goal of data exfiltration; other possible aims include attacks against data integrity or availability of critical production systems [2]. Highly targeted attacks attempt to achieve their goals via multiple stages as characterized by the Intrusion Kill Chain model. This model was developed for advanced persistent threats [3], however, a similar model can characterize the targeted attacks [4]. The first three stages of the model are reconnaissance, weaponization and delivery. Reconnaissance stage covers the activities of attackers for gathering information about the target. The attackers prepare the attack payload in the weaponization stage, then transmit it to the target in the delivery stage.

On the defensive side, strong perimeter protection and traditional intrusion detection systems alone are not enough to deal with the targeted attacks as they can be bypassed through the use of advanced attack methods [4]. False positive results and lack of sufficient log management are other important obstacles that degrade the strength of protection in complex information system environments. Thus, many organizations do not realize that their network has been compromised until weeks, months or even years later. Using those systems is a good defense-in-depth strategy, however, there are still gaps that may be minimized by implementing non-traditional security defense measures such as deception, an active defense, which is designed to trick or confuse the attacker [5]. Deception systems can act as an important complementary component to existing protection mechanisms with the aim of attack detection. These systems can lure the attackers to them in order to create focus points for detection systems. Normal users should not access deceptions and any interaction with them is considered a violation – thus reducing the frequency of false positives as regularly experienced with traditional tools. Additionally, utilization of appropriate deception instruments at each stage of the intrusion kill chain may decrease the false negative detection ratio.

The research that predicates this paper is based on the case studies where appropriate deception mechanisms are mapped to the first three phases of the intrusion kill chain: reconnaissance, weaponization and delivery. This method is chosen because it is imperative that highly targeted attacks be detected at the earliest possible stages in the kill chain – effectively breaking the chain before the target asset is exploited. Additionally, a Red Team Engagement Plan which details the penetration test activity against the target systems is developed and executed to test the

effectiveness of the deceptions, along with two metrics, Dwell Time which is the detection time of the threats and Attacker Deception-Perception Survey that evaluates the perceptions of penetration testers who take part in the execution of engagement plan. The threat model assumes that the attackers are considerably sophisticated and substantially resourced which can be enough to cover an average level of highly targeted attack. Thus, advanced persistent threats which are mostly conducted by state-sponsored actors are out of the scope. The main contribution of this study is the case studies implemented on a cyber range facility and evaluation of deceptions in terms of detection time and attacker perceptions.

II. Related Works

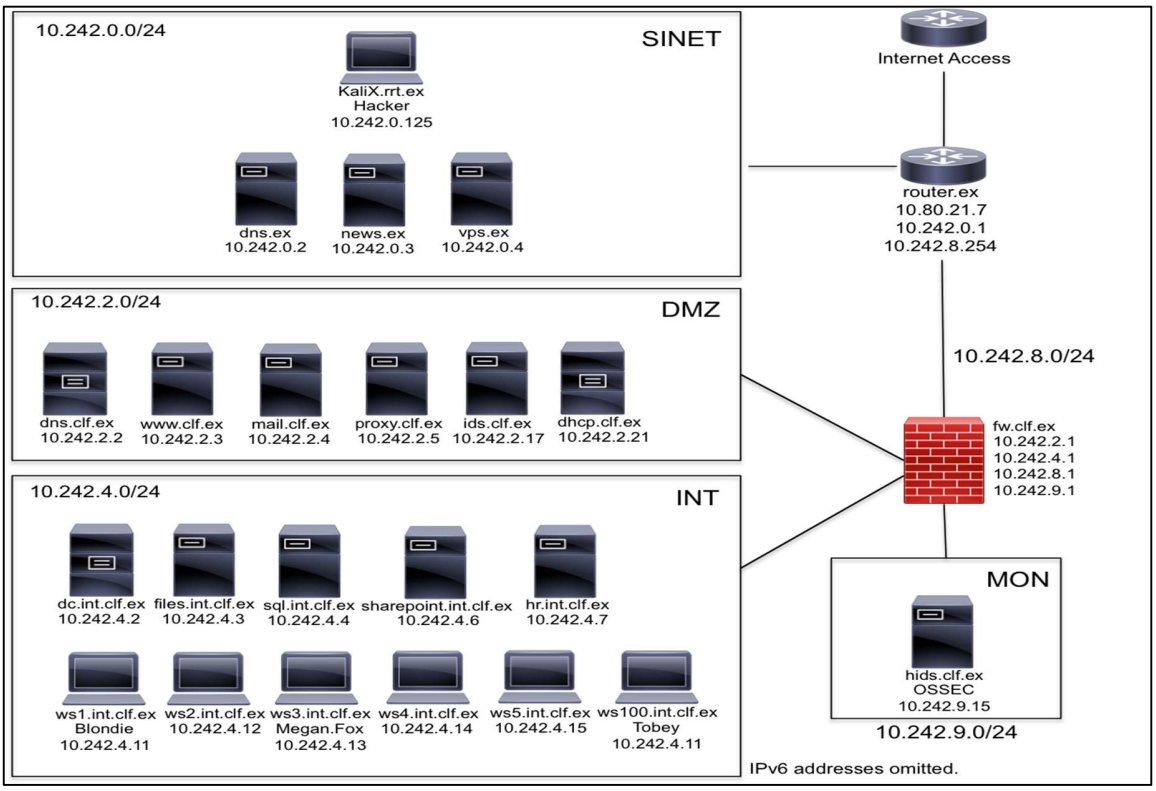
Although there is a multitude of research relating to deception mechanisms, most are focused on only one type of deception, such as a fake network topology [6], a defense against Reconnaissance; or mimicking a web site, a defense against Delivery [7]. Studies show that as the number of deception mechanisms deployed on a network increases, the likelihood of detection also increases [7] [2]. Mapping of existing cyber deception techniques to the different phases of intrusion kill chain is discussed in the studies of Heckman et al. and Briskin et al. [8] [9]. These studies also provide guidance for the planning, preparation and execution of deception systems. However, they lack practical implementations and do not provide a validation method for the effectiveness of deception design. Two fictional case studies about Stuxnet and an APT espionage campaign are given in order to explore the operational aspects of offensive and defensive deception techniques [8].

Wang et al. first introduced the notion of multilayer deception in [10]. Similar to the intrusion kill chain, they modelled a multi-stage attack with three layers of penetration: a human layer (employee information), local asset layer (employee's local machine) and a global asset layer (shared server assets), with deceptions being mapped at these three layers. Additionally, Wang et al. formulated an optimization model that chooses the best location of deceptive components at human and local asset layers and minimizes the total loss in case of an attack. However, this model does not cover all deception layers and the study lacks practical implementation of the proposed system. The idea of early detection of cyber attacks using deception was also demonstrated by Almeshekah et al. in [11], where they used the intrusion kill chain as a framework to show the effectiveness of mapping deception mechanisms at multiple levels in the chain. This paper was mainly theoretical in nature, with no experiment having been performed or metrics developed to test the effectiveness of the deceptions. However, in a dissertation by Almeshekah, in lieu of traditional honeypot scheme, he introduced a centralized deceptive fake server called Deceptiver [12]. The server hooks into a company's internet facing servers and injects deceit when it detects malicious interaction, thus creating a fake view of an organization's resources to either confuse and/or lead attackers astray. Deceptiver was a proof of concept prototype, and in the implementation it was integrated with an Apache Web Server to test it. However, they only measured the performance of the integration of the web server with Deceptiver, as opposed to the actual effectiveness of the deception itself.

III. Methodology

The experiment was implemented by applying a systematic approach to deploying deception for the early detection of advanced cyber threats. The deceptive methodology entails first selecting the evaluation environment and designing a network topology diagram to model the network infrastructure. Relevant environment can be a cyber range facility where virtual environment is utilized for cyber security trainings or operational environment where the actual business or mission processes take place. Next, an attacker profile is created by defining a Threat Model and Threat Scenario, followed by the selection, mapping and deployment of deceptions, based on the Intrusion Kill Chain. A Penetration testing scheme is also developed, which outlines a strategic deception test plan, and a Red Team Engagement plan is executed. The attacks on the deceptions are monitored and finally, the effectiveness of the deceptions is measured and validated through metrics.

Figure 1 Network Topology Diagram



A. Evaluation Environment and Network Infrastructure

The experiment is conducted at the NATO CCDOE Cyber Range facility in Tallinn, Estonia. It is an experimental environment used for Red Team exercises. The virtual environment was hosted on the VMWare ESXi 6.0 virtualization platform.

The devices set up and configured for the exercise experiment to make up the network are represented in a network topology diagram as shown in Figure 1. The overall network consisted of the Internal (INT), demilitarized zone (DMZ), simulated Internet (SINET) and monitor (MON) networks. Servers and workstations were both Windows and Linux-based. Two routers, a DHCP server, IDS, HIDS and a single firewall were also configured.

B. Threat Model and Threat Scenario

The model is based on highly skilled attackers who have substantial resources for the accomplishment of the mission. Advanced persistent threats which require very advanced technical capabilities and huge amount of resources are out of scope. The attackers are assumed to have skills in conducting spear phishing campaigns, compromising the known vulnerabilities and achieving lateral movements between different network segments. The motivation of the attackers is obtaining valuable information which can be used for monetary gain or espionage purposes.

In the threat scenario for this experiment the target is Company Z, a research firm that sells zero-day vulnerabilities to governments, with the average flaw going for $45,000-180,000. Thus, the firm is susceptible to highly targeted cyber threats. Company Z stores a catalog of zero-day exploits and their high profile client list on a file server located in the internal network.

C. Penetration Testing Schema

The traditional goal of penetration testing is to identify the exploits and vulnerabilities that exist within an organization's IT infrastructure and to help confirm the effectiveness of the security measures that have been implemented [13]. In this experiment, the penetration test is designed to test how effective the deceptions are in detecting the cyber threats according to intrusion kill chain steps.

Two professional penetration testers (RedTeam1, RedTeam2) are recruited with each having at least three years of experience. They have been involved in several professional penetration test projects. The duration for the penetration test is three days due to limited availability of testers. Black box penetration test methodology is selected where the adversaries have no knowledge of the network. However, passive reconnaissance information was provided due to the limited time of three days to complete attacker goals.

The Penetration Testing Scheme consists of four parts to include a Red Team Exercise Briefing, Red Team Rules of Engagement, Red Team Diary (RTD) and Red Team Exercise Debriefing. The Red Team Exercise Briefing details key aspects of the exercise. It is provided to RT participants and includes the dates of execution, exercise objectives, exercise outcomes, type

of exercise, exercise environment, and simplified threat model and threat scenarios to preserve results integrity in the case of the black box test. In the Red Team Rules of Engagement, the Red Team is provided general guidelines on how to conduct the penetration test. It consists of attack time limitations (i.e. 3 days), reporting requirements, type of penetration test and general attack guidelines. The Red Team Diary consists of a daily log that the penetration tester uses to document activities performed on the network. It includes such information as timestamps, source IPs, IPs of machines compromised, exploits executed on machines and other details. After the Red Team exercise is complete and the Red Team Diary has been reviewed, a debriefing takes place. The Red Team Debriefing is an interview that takes place between the exercise leader and the Red Team participants in order to ask direct questions regarding the tools, tactics and techniques used as well as why they made the decisions that they made during the attack.

D. Deployment of Deceptions and Exercise Execution

Deception mechanisms selected are T-Pot, Spam Honeypot with Intelligent Virtual Analyzer (SHIVA), Yet Another Low Interaction Honeypot (YALIH), KFSensor and Active Defense Harbinger Distribution (ADHD) [14][15][16][17]. All solutions are free and open source except for KFSensor.

In the case of reconnaissance, T-pot, a honeynet is chosen to deceive the attacker regarding the topology and contents of the target organization's network. It also defeats the weaponization phase of the intrusion kill chain, causing the attacker to develop exploits that are ineffectual, as he will fashion them based on a false network topology and non-existent services. Portspoof and KFSensor are also selected to further create a fake topology by emulating services that are non-existence on the network. In particular, Portspoof has the ability to slow down reconnaissance that uses port scanning, while KFSensor implements service emulation and has a built in IDS engine that captures these attacks in real time.

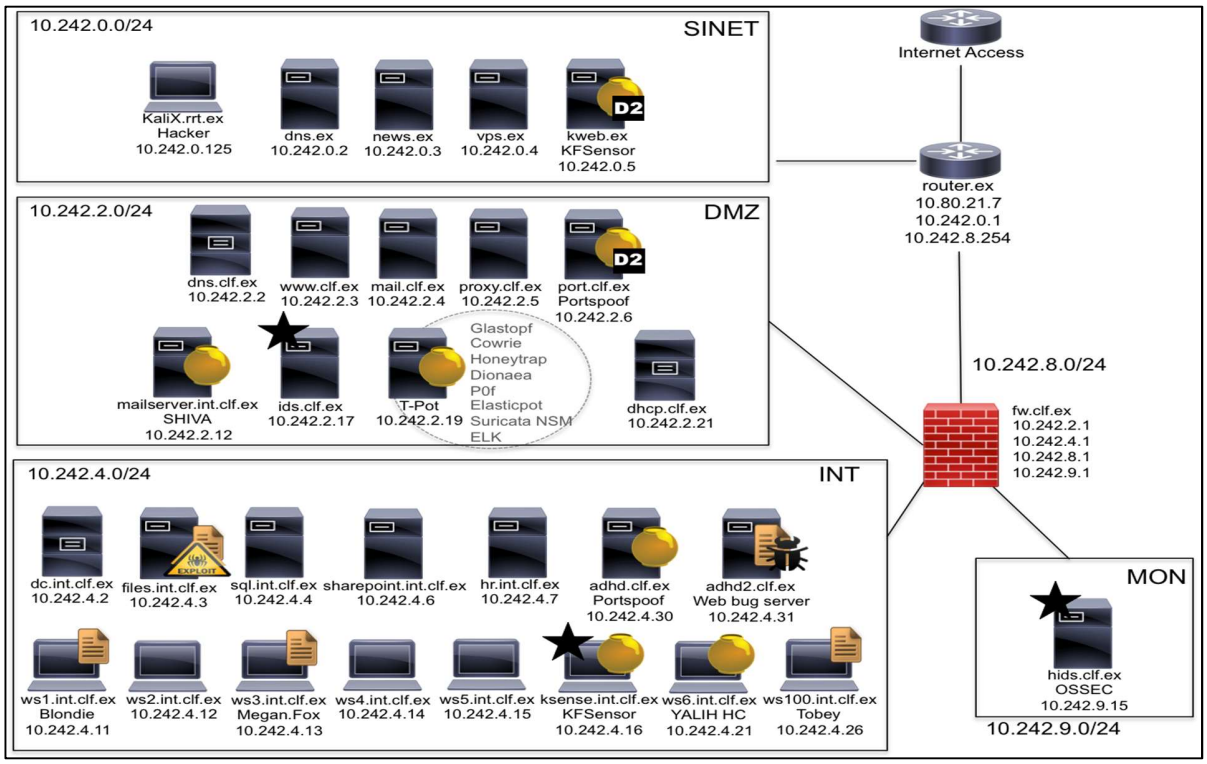
For the delivery phase, T-Pot, YALIH and SHIVA were mapped. T-pot contains vulnerable web (Glastopf), SSH (Cowrie) and malware (Dionaea) server honeypots that the attacker may interact with and be detected. SHIVA is a high interaction SPAM / Open Relay honeypot that analyses SPAM and acts as an Open Relay. YALIH is a honeyclient that retrieves email attachments and URLs and scans them to assess if they are malicious or not. In this experiment, the YALIH email honeyclient was configured to retrieve the user Blondie's email for analysis.

As depicted in Figure 2, two case scenarios are implemented: Deployment1 (D1) and Deployment2 (D2). The deployments are identical, except in the case of D2 where two additional deceptions are added for enhanced deception in-depth.

D1 deceptions are placed in the DMZ and the Internal Network. T-Pot and SHIVA are placed in the DMZ. KFSensor, YALIH honeyclient and ADHD (Web Bug Server and Portspoof) active defenses are placed in the Internal network. The honeytokens were strategically placed in the Documents directory and/or Desktop of three Windows 7 workstations: ws1(10.242.4.11) ws3(10.242.4.13) and ws100(10.242.4.26). Additionally, Honeydocs were placed on the files server, in and around the directory containing the real client list and zero day exploits. In D2, to add more complexity to the network and increase number of deceptions, ADHD Portspoof was also placed in the DMZ, port.clf.ex (10.242.2.6). KFSensor, was added to the SINET, kweb.ex (10.242.0.5). This was done in an effort to detect the ACT before it reaches the DMZ, and confuse the attacker earlier in the attack chain. Portspoof was deployed in the DMZ, further adding to network complexity perception.

The monitoring devices were located in the DMZ (Suricata/T-Pot), INT (KFSensor/IDS) and MON (OSSEC/HIDS) [18]. Log collection and analysis was done both manually and using the Elastic stack (ELK) [19]. RedTeam1 and RedTeam2 were assigned to attack the network for D1 and D2 respectively.

Figure 2. D1 & D2 Deployments



The Red Team exercise schedule indicates that each Red Team had three days to accomplish the mission of stealing Company Z's zero-day exploits and the high profile government client list. RedTeam1 and RedTeam2 executed their attacks between May 4th-6th 2016 and May 16th-18th respectively.

E. Evaluation Metrics

The two metrics used in the experiment were Dwell Time (DT) and the Deception-Perception Survey. Dwell Time measures how long the adversary is inside your network prior to being detected and the Attacker Deception Perception Survey measures the success of the deceptions in creating confusion and uncertainty on the part of the hacker [20].

Dwell Time is measured by using forensic data (i.e. logs, netflow or pcaps) to trace threats back to their origin (IP Address). In this experiment, Dwell Time is calculated by subtracting the Attack Start Time (AST) from the Time Attack Detected (TAD). These measurements (timestamps) were derived from the RTD and conducting forensic analysis of the captured data (honeypot logs) using the Elastic Stack (ELK) for T-Pot, and manual log analysis for the standalone deceptions.

The Time to Detection (TTD) specifies the maximum amount of time that the attack can remain undetected; and is selected purely based on perceived risk tolerance. If the DT is within the TTD, then the deception is effective. In this scenario, the risk tolerance is low; therefore TTD is set at less than or equal to 60 minutes, and may be adjusted as needed. The Time for Mission Execution (TME) is three days. TME describes thenumber of days allowed for the attacker to accomplish the mission.

The Attacker Deception-Perception measurement is derived from the Red Team Diary Debriefing, and is based on the Likert-type Scale to measure attacker perceptions [21]. Figure 3 gives the list of questions asked to penetration testers.

Figure 3. Attacker Deception-Perception Survey

Attacker Deception-Perception Survey
What was your overall perception of the network, as far as level of difficulty in navigation? ☐ 1-Extremely Not Complex ☐ 2-Not Complex ☐ 3- Neutral ☐ 4-Complex ☐ 5-Extremely Complex
1. How likely is it that the machines were decoys and not real? ☐ 1-Extremely Unlikely ☐ 2-Unlikely ☐ 3- Neutral ☐ 4-Likely ☐ 5-Extremely Likely
2. How likely is it that you were confused about identifying services or resources? ☐ 1-Extremely Unlikely ☐ 2-Unlikely ☐ 3- Neutral ☐ 4-Likely ☐ 5-Extremely Likely
3. How likely is it that you were interacting with honeypots? ☐ 1-Extremely Unlikely ☐ 2-Unlikely ☐ 3- Neutral ☐ 4-Likely ☐ 5-Extremely Likely
4. How likely is it that you became frustrated as a result of the complexity of the network, and not being able to locate the client list and exploits? ☐ 1-Extremely Unlikely ☐ 2-Unlikely ☐ 3- Neutral ☐ 4-Likely ☐ 5-Extremely Likely
5. How likely is it that your failure to complete the mission due to confusion about the network topology? ☐ 1-Extremely Unlikely ☐ 2-Unlikely ☐ 3- Neutral ☐ 4-Likely ☐ 5-Extremely Likely

The Debriefing consists of two sessions: direct, open-ended questions that the exercise leader asks of the Red Team participants and an Attacker Deception-Perception Survey. The open-ended questions asked are formulated based on the analysis of the Red Team Diary, and are geared toward the attacker's perception of the network, and why certain actions

were taken; but also gives insight into what tools the attacker used and the motivation behind it. The Attacker Deception-Perception Survey makes an assessment of the attacker's view of network complexity and effectiveness of deceptions.

For D1, RedTeam1 successfully exploited and compromised many vulnerable systems, however, the Nmap scanning activity was detected by T-Pot (p0F), and subsequently by the T-Pot (Glastopf) web server honeypot. Table 1 summarizes the detection times and dates of D1. Additionally, almost all of the systems in the SINET and DMZ were exploited. The internal network was breached and the sql.int.clf.ex server was exploited, however, the other servers, workstations and/or deceptions were not. The allowable time to detection is less than or equal to 60 minutes. P0f detected the Nmap scan within seven minutes, while the Glastopf web server honeypot detected the attack at the 15 minute mark.

In D2, RedTeam2 was successful in executing many exploits and conducting ARP spoofs on the DMZ and INT networks. However, the ARP spoofs failed to produce any worthwhile information. The INT was not penetrated before the end of the exercise. Emails sent to user Blondie were not related to the scenario, and although the YALIH honeyclient retrieved the emails and scanned the URL that was delivered by the attacker, neither the suspicious email attachment nor URL was flagged as malicious. As shown in Table 2, the attack began on the network using an Nmap scan on May 16th at 21:00 and was detected by T-Pot's P0f, Dionaea and Glastopf at 21:30. Honeytrap detected the attacker at 00:51 on May 17th, when the attacker probed port 25. KFSensor (kweb) also logged attacker activity on May18, including both source IP addresses the attacker used: 10.242.0.125 and 10.80.100.89 (own machine). Both T-Pot and KFSensor logged over hundreds of attempts by the attacker to find vulnerabilities and exploit them. However, they are too numerous to list in this paper. The allowable time to detection is less than or equal to 60 minutes.

The results of the Attacker Deception-Perception Survey for both deployments revealed that overall, the attackers felt that the network was complex, and were confused at times regarding the identification of services or resources. They felt it was unlikely that the machines were honeypots, although possible, due to unexpected responses to the network probe. Even though the attackers were often times confused and was not confident about network topology, the inability to reach to the target information was stated as "not enough time."

Table 1. RedTeam1 Attack Timeline

Date	Attack Start Time (AST)	Time Attack Detected (TAD)	Deception	Dwell Time (MIN)	IP	Comments
May 4	13:40	13:47	T-Pot	7	10.242.2.19	P0f – Nmap SYN scan
		13:55		15		Glastopf – POST Request
		16:55		-		Suricata – Port 80 - SQLi

Table 2. RedTeam2 Attack Timeline

Date	Attack Start Time (AST)	Time Attack Detected (TAD)	Deception	Dwell Time (MIN)	IP	Comments
May 16	21:00	21:30	T-Pot	30	10.242.2.19	P0f; Nmap Scan Port:443, Dionaea; Port 21
		21:31	T-Pot	31	10.242.2.19	Glastopf; Port 80
May 17		00:51 01:08	T-Pot T-Pot	- -	10.242.2.19 10.242.2.19	Honeytrap; Port 25 Port 587
May 18		01:56	KFSensor	-	10.242.0.5	ICMP ECHO REQ ARP Spoof 10.242.0.1
		2:11	KFSensor	-	10.242.0.5	NBT SMB SYN Scan Port 445

IV. Conclusion and Future Work

This research presents case studies for the implementation and evaluation of deceptive methodology that selects, maps, deploys, tests and monitors various deceptions against highly targeted attacks. The relevant deceptions address the detection of attacks at the reconnaissance, weaponization and delivery stages of intrusion kill chain. Recruited Red Teams attacked the systems after the deployment of deception systems. Additionally, overall system is equipped with many system monitoring tools. In order to evaluate the effectiveness of the deceptions, two metrics, Dwell Time and the Attacker Deception-Perception Survey, are utilized. Dwell Time is obtained from the timestamp of intrusion detection logs generated by the system monitoring tools. The perception of the penetration testers are collected through surveys in order to perceive the effectiveness of deceptions. The results of case studies show that the deception methodology is effective in early detection of highly targeted attacks and creates confusion and uncertainty for the penetration testers.

Future work that would be beneficial to this research would be the development of additional qualitative metrics to test the effectiveness of active defences. Comparing the outcome of white box testing vs. black box testing would also be interesting. Testing the deceptions ability to detect insider threats, both malicious and accidental is much needed research, as they represent the weakest link of the security.

V. References

[1] Danielle Anne Veluz. (2011, May) Trend Micro. [Online]. http://www.trendmicro.com/vinfo/us/threat-encyclopedia/web-attack/95/understanding-highly-targeted-attacks

[2] Ivan Dimov. (2013, June) Infosec Institute. [Online]. http://resources.infosecinstitute.com/guiding-principles-in-information-security/

[3] Eric M Hutchins, Michael J Cloppert, and Rohan M Amin, "Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains," *Leading Issues in Information Warfare & Security Research*, vol. 1, p. 80, 2011.

[4] A.K. Sood and R.J. Enbody, "Targeted cyberattacks; a superset of advanced persistent threats," in *IEEE Security & Privacy*, vol. 11, 2013, pp. 54-61.

[5] Bryce Galbraith. (2015, October) Info Security. [Online]. http://www.infosecurity-magazine.com/opinions/apts-anticipatory-active-defenses/

[6] Trend Micro. (ND) Trend Micro. [Online]. http://www.trendmicro.com/vinfo/us/security/definition/targeted-attacks

[7] PwC et al. (2015) PwC. [Online]. http://www.pwc.com/us/en/increasing-it-effectiveness/publications/us-cybercrime-survey-2015.html

[8] K.E. Heckman, F.J. Stech, R.K. Thomas, B. Schmoker, and A.W. Tsow, "Intrusions, Deception and Campaigns," in *Cyber Denial, Deception and Counter Deception*.: Springer International Publishing, 2015, pp. 31-82.

[9] G. Briskin et al., "Design Considerations for Building Cyber Deception Systems," in *Cyber Deception*.: Springer International Publishing, 2016, pp. 71-97.

[10] W. Wang et al., "Detecting targeted attacks by multilayer deception," *Journal of Cyber Security and Mobility*, vol. 2, no. 2, pp. 175-199, 2013.

[11] Mohammed Almeshekah, Eugene Spafford, and Mikhail Atallah, "Improving Security Using Deception," *Center for Education and Research Information Assurance and Security, Purdue University, Tech. Rep. CERIAS Tech Repor*, vol. 13, 2013.

[12] Mohammed H Almeshekah, "Using Deception to Enhance Security," Purdue University West Lafayette, PhD Dissertation 2015.

[13] Rahmat et. al BudLarto, "Development Of Penetration Testing Model For Increasing Network Security," Network Research Group ,School of Computer Sciences, Pulau Pinang, White Paper 2004.

[14] Deutsche Telekom AG Honeypot Project. (2015, March) T-Pot: A Multi-Honeypot Platform. [Online]. http://dtag-dev-sec.github.io/mediator/feature/2015/03/17/concept.html#concept

[15] Sumit Sharma and Rahul Binjve. (2015, November) Shiva-Spampot. [Online]. https://github.com/shiva-spampot/shiva

[16] Masood Mansoori, Ian Welch, and Qiang Fu, "YALIH, Yet Another Low Interaction Honeyclient," in *Proceedings of the Twelfth Australasian Information Security Conference*, Auckland, 2014, pp. 7-15.

[17] (2016) Black Hills Infomation Security. [Online]. http://www.blackhillsinfosec.com/?page_id=4419

[18] OSSEC. [Online]. http://ossec.github.io

[19] Abdelkader Lahmadi and Frederic Beck, "Powering Monitoring Analytics with ELK Stack," in *International Conference on Autonomous Infrastructure, Management and Security (AIMS 2015)*. , 2015.

[20] John N. Stewart, "Advanced Technologies/Tactics Techniques, Procedures: Closing the Attack Window, and Thresholds for Reporting and Containment," *Best Practices in Computer Network Defense: Incident Detection and Response*, vol. 35, pp. 30-42, 2014.

[21] Wade M. Vagias. (2006) Clemson University. [Online]. https://www.clemson.edu/centers-institutes/tourism/documents/sample-scales.pdf