TALLINN UNIVERSITY OF TECHNOLOGY

School of Information Technology

Department of Software Science

ITC70LT

Bolaji Ayoola Ladokun

156330IVCM

AN ANALYTICAL APPROACH TO CHARACTERIZATION OF TARGETED AND UNTARGETED ATTACKS IN CRITICAL INFRASTRUCTURE HONEYPOT

Master's Thesis

Supervisor: Hayretdin Bahsi

PhD
Senior Research Scientist

Tallinn, 2017

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia Teaduskond
Tarkvarateaduse Instituut

ITC70LT

Bolaji Ayoola Ladokun

156330IVCM

ANALÜÜTILINE KÄSITLUS, KUIDAS ERISTADA SIHIPÄRASEID JA JUHUSLIKKE RÜNNAKUID KRIITILISE INFRASTRUKTUURI MEEPOTI PIHTA

Magistritöö

Juhendaja: Hayretdin Bahsi
PhD
Vanemteadur

Tallinn, 2017

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Bolaji Ayoola Ladokun

02.05.2017

Abstract

National growth and economy largely depends on critical infrastructures. Examples of these infrastructures include oil and gas system, electric grids, water plants and banking. Attacks directed towards critical infrastructures are on the rise recently; this has caused perturbation for both governments and private providers. The growth of ICS components in critical infrastructures connected to the Internet is far too copious. States, private organizations and researchers use honeypots to gather information about the techniques and motives of attackers targeting their infrastructures.

However, on the Internet, data captured by research honeypots contains many untargeted attacks. Hacks and compromises in the past have shown indication that attackers clearly select the victim and tailored their exploits to the targeted systems. This paper discusses an analytical approach to characterization of targeted and untargeted attacks in critical infrastructure honeypot. To begin with, critical infrastructure honeypot for three sectors namely; power, oil and gas and bank was setup in collaboration with Nigerian Computer Emergency Response Team.

The experiment was in two phases in which the second phase has honeypot systems that are physically located in Nigeria and the public IP addresses were declared in dark web and ensure they were indexed by shodan and censys. Data was gathered from the honeypots in a thirty-day calendar window from the two different phases. Principal Component Analysis was introduced to reduce the dimension of the captured data and to observe the latent characteristics and projections of the data set.

This thesis is written in English language. It contains 69 pages, 20 tables and 15 figures.

Annotatsioon

Riigi ja majanduse kasv sõltuvad suuresti kriitilisest infrastruktuurist, nagu näiteks nafta ja gaasi süsteemid, elektriliinid, veepuhastus jaamad ning pangandus. Viimasel ajal on sihitud rünnakud selliste kriitilise infrastruktuuri osade vastu olnud tõusuteel, mis omakorda on tekitanud segadust, kuivõrd ühendatus Internetti on rikkalik. Riigid, eraorganisatsioonid ja uurijad kasutavad meepotte, et koguda informatsiooni nende ründajate tehnika ja motivatsiooni kohta, kes nende infrastruktuuri ründavad.

Meepottide abil kogutud informatsioon sisaldab ka arvukalt infot juhuslike rünnakute kohta, mille sihtmärgiks ei pruugi olla konkreetne infrastruktuur. Rünnakute ligiid ja süsteemi sisse pääsemised on varasemalt olnud indikatsiooniks, et ründajad on spetsiaalselt ohvri valinud ning oma rünnakutehnikat viimistlenud vastavalt ohvri süsteemidele. Käesolev magistritöö uurib analüütilist lähenemist eristamaks sihipäraseid ja juhuslikke rünnakuid kriitilise infrastruktuuri pihta läbi nende erinevate iseloomujoonte. Uurimuse tarvis seati üles meepott kriitilise infrastruktuuri imiteerimiseks koostöös Nigeeria CERT-iga – täpsemalt energia, nafta ja gaasi ning panganduse imiteerimiseks.

Eksperiment viidi läbi kahes osas, kusjuures teises osas oli meepoti füüsiliseks asukohaks Nigeeria ning avalikud IP addressid avalikustatud ka pimeveebis, et kindlustada IP aadressite indekseerimine *shodan*i ja *censys*i poolt. Andmeid koguti meepottidesse kolmekümnepäevastes perioodides kahes erinevas faasis. Selleks, et taandada andmestiku dimensioone ja leida peidetud tunnuseid ning projektsioone, kasutati peakomponentanalüüsi.

See magistritöö on kirjutatud inglise keeles. See sisaldab 69 lehekülge, 20 tabelit ning 15 joonist.

Table of abbreviations and terms

ICS	Industrial Control Systems
SCADA	Supervisory control and data acquisition
APT	Advanced Persistent Threats
CERT	Computer Emergency Response Team
RTU	Remote Terminal Unit
MMI	Man Machine Interface
HMI	Human Machine Interface
TCP/IP	Transfer Control Protocol/Internet Protocol
PLC	Programmable Logic Controllers
SSH	Secure Shell
PCA	Principal Component Analysis
DDoS	Distributed Denial of Service
MHN	Modern Honeypot Network
HTTP	HyperText Transfer Protocol
SMB	Server Message Block
SIP	Session Initiation Protocol
FTP	File Transfer Protocol
TFTP	Trivial File Transfer Protocol
SFTP	Secure Shell File Transfer Protocol
SCP	Secure Copy
LTS	Long Term Support
IDS	Intrusion Detection System
RPC	Remote Procedure Call

Table of contents

1 Introduction 11

1.1 Motivation 11

1.1.1 Problem Statement 12

1.1.2 Main Contribution 12

1.2 Scope 13

1.3 Chapter Summary 13

2 Background Information 14

2.1 Critical Infrastructures 14

2.2 SCADA and ICS Systems 14

2.2.1 Internet and The Third Generation SCADA System 15

2.3 Security Challenges in SCADA 16

2.4 Deception 17

2.5 Honeypots 17

2.5.1 Honeypots Categories 17

2.5.2 Honeypot Interaction Levels 18

2.5.3 Honeypots Platforms 19

2.5.4 Honeypots Descriptions 20

2.6 Targeted and Untargeted Attacks 24

2.7 Principal Component Analysis 25

2.8 Related Works 25

2.8.1 Attractiveness of Honeypots in Critical Infrastructures 25

2.8.2 Detecting Targeted Attacks in Honeypots 26

2.8.3 Malware Analysis in Targeted Attacks 26

2.8.4 Unsupervised Classification and Characterizations of Honeypot Attacks 27

3 Methodology and Implementation 29

3.1 Critical Threats to Critical Infrastructures 29

3.2 Honeypot Selection 30

3.3 Honeypot Deployment 30

3.4 CERT SCADA Honeynet 31

3.4.1 Device Specification and Emulated Services 31

3.5 Experiment Phase One and Two 32

Simulation of three Critical Infrastructures 32

3.5.1 Power 32

3.5.2 Oil and Gas 33

3.5.3 Banking 34

3.6 Customization of services 34

3.7 Monitoring and Data Collection 36

4 Analysis and Result 37

4.1 Data Preprocessing 37

4.2 Principal Component Analysis 37

4.2.1 Input Data 38

4.2.2 Eigen Vectors 40

4.2.3 Principal Components 41

4.2.4 Variables and Quality of Projections 43

4.2.5 Observations 45

4.3 Password Complexity 47

4.3.1 Phase One 47

4.3.2 Phase Two 49

4.3.3 Observations 51

4.4 Malwares 51

4.5 Web Attacks 52

5 Conclusion and Future Work 54

References 56

Appendix 1 61

Appendix 2 64

Appendix 3 67

List of figures

Figure 1: Architecture of Third Generation SCADA System [14] 15

Figure 2: Current SCADA System [15].. 16

Figure 3: Schematic Setup of Amun [29] 20

Figure 4: Argos: High-level overview [38] 22

Figure 5: Process of how Dionaea captures attack [41]. 23

Figure 6: Overview of MHN [58].. 30

Figure 7: Sample Bacnet Template Customization in Conpot 35

Figure 8: Sample HTTP Customization in Conpot 35

Figure 9: Scree Plot of Principal Components 42

Figure 10: Variable Correlation Circle.. 43

Figure 11: Biplot of variables and observations 46

Figure 12: Biplot of Observations of the First two principal Components.................... 46

Figure 13: Sample Phase 1 SQL Injection 52

Figure 14: Sample Phase 2 SQL Injection 53

Figure 15: Code snippet for attempt website defacement 53

List of tables

Table 1: Emulated Services on Dionaea 23

Table 2: 2016 Most Critical ICS Threats [57] 29

Table 3: Overview of Selected Honeypot 31

Table 4: Phase One Device Specification 31

Table 5: Honeypot System Specification for Sota PLC... 33

Table 6: Honeypot System Specification for Frobe Oil... 33

Table 7: Honeypot System Specification for GBT Bank 34

Table 8: Labels for PCA observations 39

Table 9: List of Variables used in PCA.. 39

Table 10: Resulting Eigen Vectors for PCA 40

Table 11: Resulting Principal Component 42

Table 12: Squared Cosines of Variables 44

Table 13: Well Projected Variables in Principal Components 45

Table 14: Phase 1 - Character Set of Password 48

Table 15: Phase 1 - Character Set Ordering 48

Table 16: Phase 1 - Password Length Variation 48

Table 17: Phase 2 - Character Set of Password 49

Table 18: Phase 2 - Character Set Ordering 50

Table 19: Phase 2 - Password Length Variation 50

Table 20: New malwares attracted by Phase two GBT bank honeypot 52

1 Introduction

National growth and economy largely depends on critical infrastructures. Examples of these infrastructures include oil and gas system, electric grids, water plants and banking. Attacks directed towards critical infrastructures are on the rise recently; this has caused perturbation for both governments and private providers. The case of Stuxnet has brought about growing concerns on the security of critical infrastructures and even became more evident with the attack on Ukrainian power grid [1] and Kemuri water company's Industrial Control Systems (ICS) infrastructures [2]. ICS allow operators to remotely operate a number of industrial systems from electrical power grids, oil and gas grids to water treatment plants [3]. Research interest in understanding the vulnerabilities in ICS, the threats and attack landscape has grown because of these attacks.

1.1 Motivation

The growth of ICS components in critical infrastructures connected to the Internet is far too copious. According to Kepersky 2016 online ICS availability report, 220,558 ICS components were discovered by the Shodan and Censys search engines [4]. Majority of cases, ICS components are design with the assumptions that the networks in which they will operate are isolated and not connected to the internet, so even built-in security are often not implemented [4]. Moreover, it is common knowledge that while most of these components are largely dependent on isolation for security; isolation is not sufficient to defend against the current threat landscape. It is important to note that both government and private owners are saddled with the responsibility of ensuring the safety of critical infrastructures.

The task is huge, considering the plethora of vulnerabilities that could be exploited by attackers. Also, ICS differs from conventional computer systems because there are inherent cybersecurity weaknesses and the extent the exploit of these vulnerabilities could lead caused [5]. Exploitation of these vulnerabilities could cause loss of lives and properties, economy loss and can even escalate to war. Perhaps the most dangerous attack threats that critical infrastructure currently face are targeted attacks. In targeted attacks, there is a clear indication that an attacker clearly select the victim that was attacked and tailored exploit(s) to the victim [6].

In defending these infrastructures, it is important to have a good understanding of the latent characteristics of these attacks by observing and analyzing the structure and data set of attacks directed towards them. This would help to improve the techniques used in defending critical infrastructures.

1.1.1 Problem Statement

States, private organizations and researchers use honeypots to gather information about the techniques and motives of attackers targeting their infrastructures. The true value of honeypot is in deception and being probed by attackers [7]. In the case of critical infrastructures, hacks and compromises in the past have shown indication that attackers clearly select the target and tailored the attacks to suit the target systems [6].

Previously, various researches have focused on honeypot to mimic different services on ICS/Supervisory control and data acquisition (SCADA) and critical infrastructures in general to know the culprit behind attacks and to understand the capabilities of attackers [8]. These researches have no proof that the data set collected contained data from targeted attacks. However, on the Internet, there are many attacks that are untargeted attacks. Targeted attacks remain rare in numbers when compared with untargeted ones [6]; However, it is this rarity that makes the detection more difficult [6].

Hence, the questions this research paper poses to answer are;

- How to identify possible indications of targeted attacks in critical infrastructure honeypots?
- Are research ICS honeypots capable of attracting targeted attacks?

1.1.2 Main Contribution

In previous researches, there have been no direct approach to characterizing targets and untargeted attacks in critical infrastructure honeypots. Identifying possible indications of targeted attacks in critical infrastructure honeypots would provide more insight to understanding the threat landscape critical infrastructures currently faces. To achieve this, it is important to simulate critical infrastructures.

Hence, the contributions this thesis makes are:

- The simulation of critical infrastructures honeypots for different sectors.
- Sets of insights to classification of attacks in honeypots as either targeted or untargeted attacks.

1.2 Scope

The main purpose of this thesis is to simulate different critical infrastructure honeypots for power, oil and gas and banking sectors and then analyze the attacks to identify clear indication of targeted attacks. The experiment environment is not suitable to attract and identify Advanced Persistent Threats (APT), therefore, detection of APT are out of scope. Also, the analysis method applied is exploratory data analysis technique.

The data captured were gathered from sensors to emulate critical infrastructures in Nigeria. The emulation environments were deployed outside of production ICS and critical infrastructures. These infrastructures were provided by the Nigerian Computer Emergency Response Team (CERT). The aim was to simulate critical infrastructures outside of real critical infrastructures.

1.3 Chapter Summary

This thesis consists of five chapters:

Chapter 1 provides the introduction and motivation behind the research

Chapter 2 discusses the technical theoretical background and reviewing existing literatures

Chapter 3 introduces the simulating of the critical infrastructure environment in two phases

Chapter 4 the evaluation of attacks and using principal component analysis

Chapter 5 discusses the conclusion and recommendation for future work

2 Background Information

The following sections discuss critical infrastructures, SCADA systems and architecture, security challenges in SCADA. Next is targeted and untargeted attacks and deception (review on honeypots). Lastly, related works and shortcomings would also be covered.

2.1 Critical Infrastructures

Critical infrastructures provide essential services to the society and are mainstay to national economy, survival and health. Hence, the system assets used in critical infrastructures are valuables and should be defended from malfunction, disruption or destruction due to attacks through the Internet. In power and oil and gas sectors, Supervisory Control and Data Acquisition Systems (SCADA) are very essential to Critical Infrastructures because many of this systems are controlled by SCADA [9]. Contrary to this, banking sector uses conventional IT systems in its operations.

However, vulnerabilities in SCADA systems face similar threats as other networked computer systems together with threats associated with their legacy systems [10]. Considering how essential critical infrastructures are and the security challenges, defenders must understand how to effectively use deception and understand the characteristics of data gathered from honeypots.

2.2 SCADA and ICS Systems

Computer-based supervisory control systems were introduced in the 1960s and the first systems were based on mainframe computer technology available at that time. The systems were not yet called SCADA systems, as that particular acronym did not come into general use until 1980s [11]. Technology advancement has made SCADA systems to transcend from been limited to plants to being used for remote supervisory and control. The prototypical SCADA system is designed with four major parts: a central computer (host), Remote Terminal Units (RTUs), a wide-area telecommunications system and an operator who remotely interfaces to access the system [11]. “The operator interface is also referred to as the operator console, the Man Machine Interface (MMI) or Human Machine Interface (HMI)" [11].

Over the years, SCADA systems have experienced notable evolution. The third generation SCADA system has more functions, robustness and complexity. SCADA systems in the networked generation includes both serial and TCP/IP communication in its network design which breaks the isolation concept of previous designs [12]. TCP/IP implementation brought about migration and standardization in SCADA systems which makes SCADA systems to be able to understand TCP/IP connections [12]. An example of SCADA IP-based protocol is Modbus/TCP [12], [13].

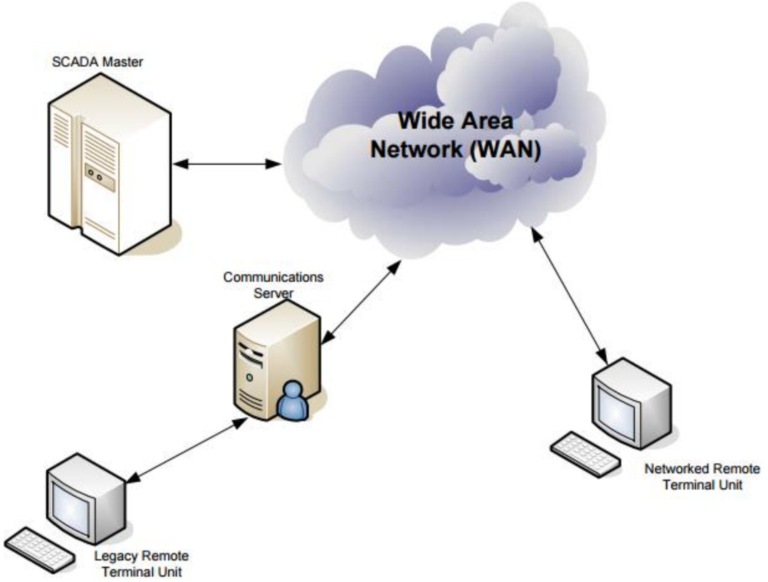


Figure 1: Architecture of Third Generation SCADA System [14]

2.2.1 Internet and The Third Generation SCADA System

As seen in figure 1 above, the third generation SCADA architecture connects different component via the Wide Area Network (WAN). In foreseeable future, the control Industry might be one of the main sectors that might be more demanding on the use of wireless technologies and the Internet for the control [15].

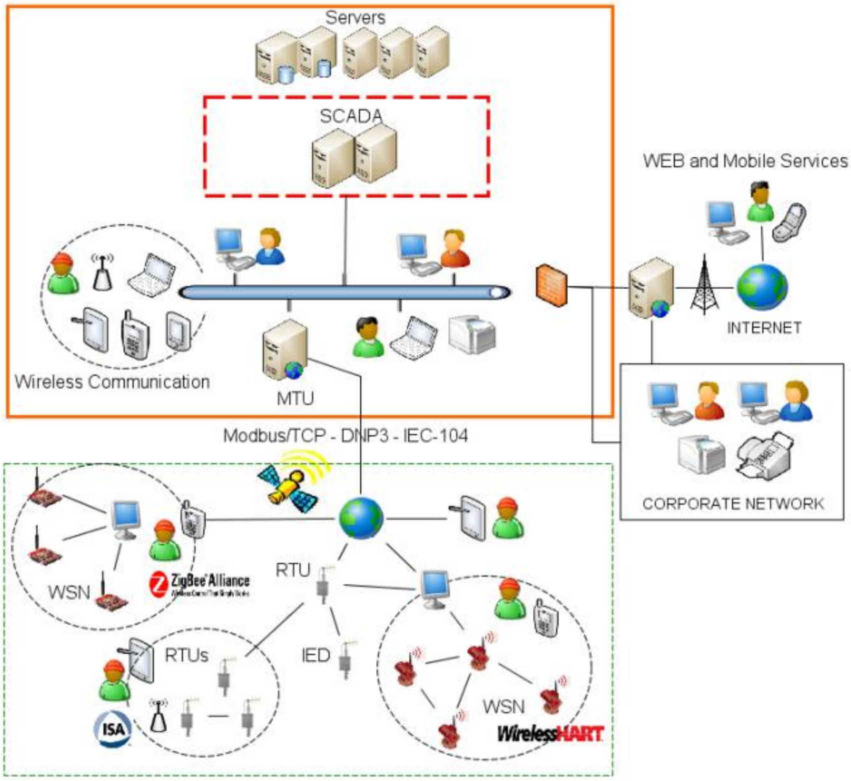


Figure 2: Current SCADA System [15]

With this it is safe to say, SCADA systems have inherited common vulnerabilities associated with TCP/IP. This could be considered more dangerous because SCADA systems are used in critical infrastructures as seen in modern SCADA architecture in Figure 2.

2.3 Security Challenges in SCADA

During the advent of SCADA systems in the 1960s, engineers and researchers focused more on building robustness, flexibility and safety [11]. The implementation of TCP/IP in SCADA and its ability to communicate over ethernet has opened it up to far too many possible attacks. The likelihood of exploiting both inbuilt and inherited vulnerability has increased exponentially and in predictable future, there is no likelihood this is going to change.

Many of the standard security procedures and best practices employed nowadays includes the use of antivirus, installation of patches and updates, strong password usage and encryption of data. These procedures are not generally and strictly implemented by system administrators. However, in security architecture, deception is considered to have the ability to give useful insights, methodologies and additional time to prepare adequate defenses [16]. This could be considered as a proactive security approach.

2.4 Deception

Deception is a deliberate act performed by a sender to cause a receiver to belief in contrary to what the sender believes is true to put the receiver's disadvantage [17]. Active deception only seeks to direct intruders from the real network environment to a fake environment which contains some data seemingly of great values to the attacker [18]. To achieve deception, an environment should be simulated which consists of three potential techniques [19]:

- The first simulation technique is providing fake information that a valuable resource or piece of information is present in a location [19].
- The second simulation technique is imitate the characteristics of a seemingly useful object [19].
- The third simulation technique is decoying which deflects attention from a real object to an irrelevant one [19].

A very common deceptive tool is honeypot, this would be discussed in the next section.

2.5 Honeypots

Honeypots are the best-known defensive deception in Cyberspace [20]. A honeypot is a computer designed to attract attackers by providing an environment with fake resources which looks valuable to the attacker [19]. It is an extra element in active defense system for network security usually used by network administrators [21]. It records attacks and intrusion information about techniques, tools and activities of the hacking process directed towards a device or system [21].

2.5.1 Honeypots Categories

Honeypot can be categorized into production and research honeypot. Production honeypot is used to protect network of corporation [21]. Production honeypots are used in companies with production systems. “They protect the target system by deceiving and detecting attacks, giving alert to administrator” [21].

1. Production Honeypot
2. Research Honeypot

Production Honeypot

Generally, this type of honeypot is organizational specific. They are deployed in an organization environment. Production honeypots usually to reflect the production network of the company (or specific services), inviting attackers to interact with them to expose vulnerabilities of the network [22]. Despite the fact that, they identify attack patterns, they give less information about the attackers than research honeypot [22].

Research Honeypot

This type of honeypot is largely used by universities and security research organizations. The focus is to gather information about tools and techniques used in launching attacks, identification of vulnerabilities exploitable by attacker and identification of attack trends and patterns. They are usually used in academics and research companies. “Research honeypot is primarily for learning new attacking methods and tools, gaining new information about attacks though it can be used for production honeypot” [23]. It provides more interactive chances for attackers and takes more risks of being controlled at the same time. Research honeypot take an effective data control mechanism to prevent from being a jump to attack other computer system [21] [24].

2.5.2 Honeypot Interaction Levels

Honeypot can be categorized based on interaction levels between the attacker and the system. The interaction levels of honeypots can be divided into three, namely:

- Low Interaction
- Medium Interaction
- High Interaction

Low Interaction Honeypot

The concept of low interaction client honeypot was first identified in taxonomy of honeypots be seen in [25][26]. A low interaction client honeypot is a client honeypot that uses simulated clients instead of using a real system to interact with servers [26]. Low interaction level corresponds to exposed functionality being limited. For example, a simulated SSH server of a honeypot is not able to authenticate against a valid login/password combination and allow for further interaction after successful authentication [25].

The interaction of attacker to the honeypot environment is very limited since it is a simulated environment. Low interaction honeypots usually lack the capacity to contain attacks. Experienced attackers could identify low interaction honeypots easily which may abend the attack far too early.

Medium Interaction Honeypot

The key feature of Medium Interaction Honeypots is application layer virtualization. These kinds of honeypots do not aim at fully simulating a fully operational system environment, nor do they implement all details of an application protocol. All that these kinds of honeypots do is to provide sufficient responses that known exploits await on certain ports that will trick them into sending their payload [27].

High Interaction Honeypot

High Interaction Honeypots are real, vulnerable systems, often running in a virtual machine environment and behind a rate limiting firewall. Due to the nature of High Interaction Honeypots, they can be used to detect 0day attack vectors and automatically adapt to any new command and control protocol [27].

2.5.3 Honeypots Platforms

1. Virtual Honeypot
2. Physical Honeypot

“A physical honeypot is a real machine with its own operating system and IP address, while a virtual honeypot is a machine which emulates system behavior and IP addresses”

[28]. It has the capability to respond to network traffic. There are lots of operating system available for use and are either built to work on Windows or Linux operating systems.

2.5.4 Honeypots Descriptions

Amun Honeypot

Amun is a lightweight low-interaction honeypot, designed to capture malware that spreads by exploiting server based vulnerabilities. Amun is made up of different modules which includes: vulnerability module, shellcode analyzer, request handle and amun kernel [29]. Amun is a powerful low interaction honeypot written in python and can be easily extended and adapt into different operating system. When deployed, it tries to emulate the required protocol of an application an attacker is trying to exploit, making it more successful at deceiving attackers [29]. Further information and capabilities of amun can be found at [29].

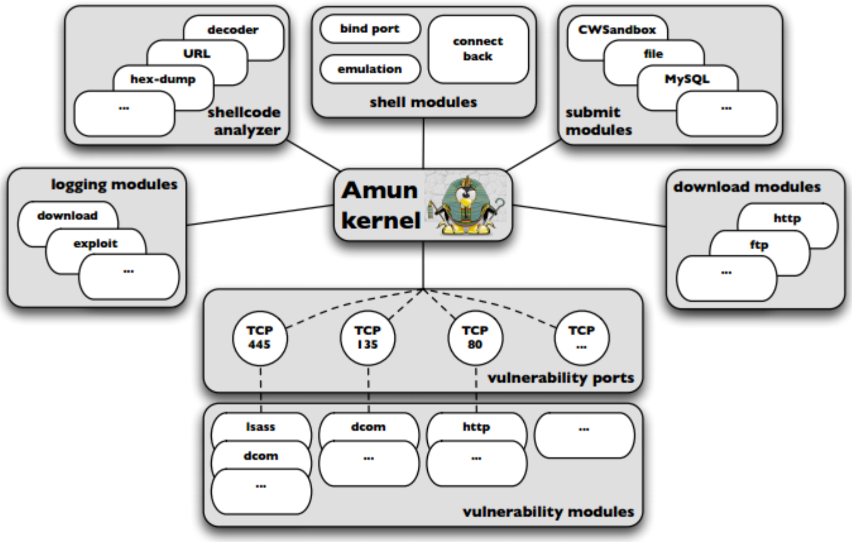


Figure 3: Schematic Setup of Amun [29]

Glastpof Honeypot

“Glastpof is a low-interaction web application honeypot capable of emulating thousands of vulnerabilities to gather data from attacks that target web applications” [30]. Glastpof provides responses to an attacker that is similar to what the attack expects from the exploits in the web application [30]. It handles both HyperText Transfer Protocol (HTTP) get and post request seamlessly.

Conpot Honeypot

Conpot is a low interaction honeypot used in SCADA/ICS network [31]. It simulates a SIMATIC S7-200PLC which includes the following protocols; Modbus (a serial communication protocol), Simple Network Management Protocol (SNMP) and Hypertext Transfer Protocol (HTTP) [31] [32]. The following are the emulated services by conpot.

- **Modbus**: “Modbus TCP/IP is the Modbus RTU protocol with a TCP interface that runs on Ethernet” [13]. More information about Modbus TCP/IP protocol can be found in [13].
- **S7 Communication (S7comm):** S7comm is a protocol owned by siemens which runs between programmable logic controllers (PLCs) of the Siemens S7-300/400 family [33]. It is used for PLC programming, exchanging data, accessing PLC data from SCADA systems and diagnostic purposes [33].
- **Hypertext Transfer Protocol (HTTP):** HTTP is the communication protocol on the Internet [34].
- **Simple Network Management Protocol (SNMP):** SNMP an application layer protocol which helps in the transfer of management information between network devices, such as workstations, nodes and routers [35]. It is a very good management to monitor, troubleshoot and fix problems on network devices.
- **Building Automation and Control networks (BACnet):** BACnet is a data communication protocol developed to create a standardize communication rules between building automation system components [36]. It allows systems to communicate with each other by providing standardized methods for transporting information [36].
- **Intelligent Platform Management Interface (IPMI):** IPMI is a series of specifications that provide standardized interfaces to platform management services [37]. It is used to monitor and control hardware.

Argos Honeypot

“Argos is an emulator for fingerprinting zero-day attacks for advertised honeypots with automatic signature generation” [38]. It is an environment containment for worms and human attacks [38]. Argos is a robust honeypot to capture zero-day attacks. It is important to have this honeypot because of the criticality nature of ICS.

“The emulator employ dynamic taint analysis to detect when a vulnerability is exploited to alter an application’s control flow” [38] . Argos used the following steps to achieve this [38]:

- tag data originating from an unsafe source as tainted;
- track tainted data during execution
- identify and prevent unsafe usage of tainted data.

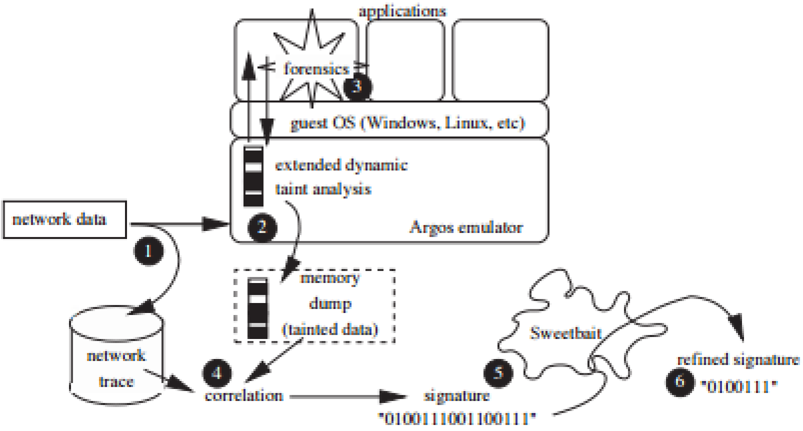
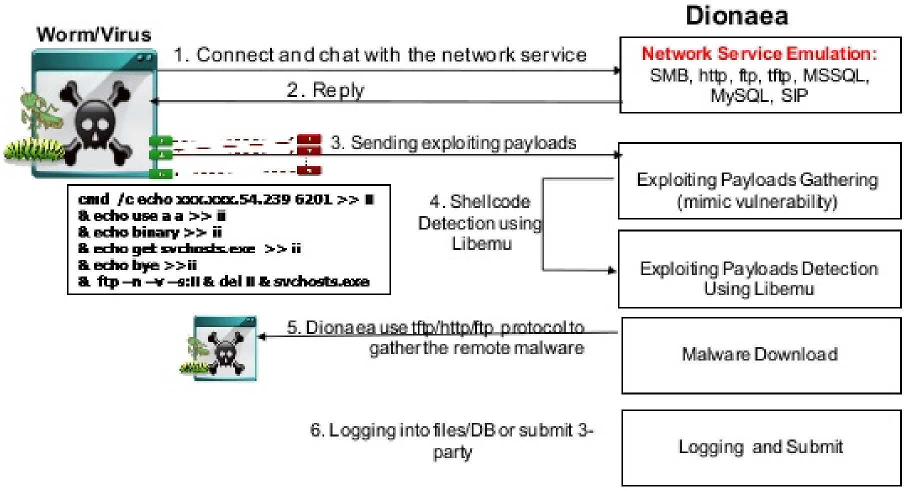


Figure 4: Argos: High-level overview [38]

Dionaea Honeypot

Dionaea honeypots embeds python as a scripting language, using libemu to detect shellcodes and it supports IPV6 and Transport Layer Security (TLS) [39]. The goal of Dionaea is to trap malware exploiting vulnerabilities exposed by services offered to a network. Tthe ultimate goal is gaining a copy malware [39]. Dionaea was configured to support all emulated services available (including Server Message Block (SMB), HTTP, File Transfer Protocol (FTP), Trivial Transfer File Protocol (TFTP), Session Initiation Protocol (SIP), and MySQL) [39] [40].


Dionaea
Worm/Virus
1. Connect and chat with the network service
2. Reply
3. Sending exploiting payloads
4. Shellcode Detection using Libemu
5. Dionaea use tftp/http/ftp protocol to gather the remote malware
6. Logging into files/DB or submit 3-party
Network Service Emulation:
SMB, http, ftp, tftp, MSSQL, MySQL, SIP
Exploiting Payloads Gathering (mimic vulnerability)
Exploiting Payloads Detection Using Libemu
Malware Download
Logging and Submit

```
cmd /c echo xxx.xxx.54.239 6201 >> ii
& echo use a a >> ii
& echo binary >> ii
& echo get svchosts.exe >> ii
& echo bye >>ii
& ftp –n –v –s:ii & del ii & svchosts.exe
```

Figure 5: Process of how Dionaea captures attack [41].

Table 1: Emulated Services on Dionaea

Service	Port	Description
SMB	80/TCP	Common Internet File System
FTP	21/TCP	Standard protocol for computer file transfer
TFTP	69/UDP	Lockstep file transfer protocol
SIP	5060/5061	Signaling protocol for controlling multimedia communication sessions
MYSQL	1433/TCP	Relational Database Management System
HTTP	80/TCP	Internet communication protocol

Elastic Honey Honeypot

“Elastichoney is a simple elasticsearch honeypot designed to catch attackers exploiting Remote code execution (RCE) vulnerabilities in elasticsearch” [42]. “RCE attacks are one of the most prominent security threats for web applications; it is a special kind of cross-site-scripting (XSS) attack that allows client inputs to be stored and executed as server side scripts” [43].

Cowrie Honeypot

Cowrie is a medium interaction SSH and Telnet honeypot designed to log brute force attack attempt and the shell interaction performed by the attacker [44]. It has a full fake filesystem of Debian 5.0 which allows an attacker to remove and add files [44]. It logs the entire session and interaction of an attacker. Cowrie supports the following features;

- Secure Shell File Transfer Protocol (SFTP) and Secure Copy (SCP).
- Supports SSH exec command.
- Logs SSH proxies.
- Save file uploaded via curl/wget [44].

2.6 Targeted and Untargeted Attacks

To achieve a profound defense against attacks, it is important for organizations to understand the landscape of attacks facing them. The task, however, is cumbersome because of the nature of the Internet. “Cyberspace is a factitious environment without solid boundary which makes it very different from the physical world” [45]. The concept of cyberspace boundaries is beyond the scope of this research.

Some researchers and companies use the term targeted attack and Advanced Persistent Threat (APT) interchangeably, however, this could be considered erroneous because targeted attacks are not always persistent in nature. APT is a situation whereby an attacker gain access to a system and manage some degree in the infrastructure while remaining undetected during this period [46]. In targeted attacks, there is a clear indication that an attacker select his potential victim and tailors his approach to suit the target environment [6]. “Spear phishing” is a subset of targeted attacks where malicious emails are sent to targeted individuals to compromise them to disclose sensitive information or credentials [6]. Attackers are more driven towards a system or individual when they have a pre-knowledge of what type or volume of information a system or an individual possesses; this in many cases serves as a motivation[6].

Untargeted attacks often referred to as non-targeted attacks can be referred to as “attack by opportunity”. In this type of attacks, there is no prove that the attack crafted his attack to the potential victim. The success of untargeted attacks depends largely on the negligence of the victim.

2.7 Principal Component Analysis

Principal component analysis (PCA) is well known of the techniques of multivariate analysis. It was first introduced by Pearson (1901), and developed independently by Hotelling (1933). Until development of electronic computers, PCA was not widely used, but it is now well ingrained in virtually every statistical computer package [47]. Principal component analysis is a multivariate statistical technique that has been widely used in multi-disciplinary research areas such as Internet traffic analysis, economics, image processing, and genetics, to name only a few[47] [48]. PCA is a very powerful exploratory data analysis research method. PCA is mainly used to reduce the dimensionality of a data set into a few uncorrelated variables, principal components (PCs), which retain most of the variation in the original data [47] [48].

2.8 Related Works

Honeypots have been used previously in many researches and a recent trend involves using honeypots on SCADA and critical infrastructures in general to identify attack patterns and trends. Despite this array of research, none has a central point of characterization of attacks directed towards research critical infrastructure honeypots as either targeted or untargeted. Previous research has shown that targeted attacks remain rare in numbers when compared with untargeted ones [6]. Therefore, in the following sections, related researches that undertakes detecting targeted attacks using honeypots, attractiveness of honeypots and characterization of attacks in honeypots will be discussed.

2.8.1 Attractiveness of Honeypots in Critical Infrastructures

S.M. Wade did a research on attractiveness of honeypots as critical infrastructure security tools for the detection and analysis of advanced threats [49]. The focal point in the research was to answer if control systems now attract more attention from hackers, organized crime, terrorists, and foreign intelligence services. The research concluded by indicating that while the SCADA Honeynet system received plenty of attention, not a single “visitor” to the honeynet attempted to take advantage of the SCADA specific services in spite of their well-known vulnerabilities [49]. In contrast to S.M. Wade, T.

Sochor and M. Zuzcak seek to detect the attractiveness of conventional IT systems honeypots sensors to attackers in [50].

These studies did not simulate any critical infrastructure sector. There was no attack directed towards the emulated ICS services in [49] which indicates that, it is difficult to measure the attractiveness to the honeypot environment. Besides that, the research does nothing to analyze if the collected data are targeted or untargeted data, hence it is impossible to determine if the honeypot in the research attracted targeted attacks or untargeted attacks only.

2.8.2 Detecting Targeted Attacks in Honeypots

K. G. Anagnostakis *et al.* implemented shadow honeypots to detect targeted attacks in [51]. The architecture provided in this research combines honeypot and anomaly detection to monitor network traffic. The architecture compares the differences between the previous state of a system with incoming traffic to detect anomalies [51]. The shadow honeypot provides protection by filtering incoming traffic which is obvious to the end-user. The study is end-user focused and seek to identify attacks directed to specific application on end-user's machine. R. S. Ramachandruni and P. Poornachandran used honeypots systems that mimics ICS services in detecting network attack vectors on SCADA systems [52]. N. Sayegh *et al.* provides a test-bed to reveal how easy it is to conduct an internal attack on SCADA components [53]. E. Vasilomanolakis *et al.* focused on automatically generating signatures for attacks directed towards ICS systems [54].

Theses researches only deals with attacks in general by premising on the fact that, any attack towards an application is targeted; there was no clear distinction in definition of what targeted attacks are. Moreover, no effort was made to identifying targeted and untargeted attacks. Also, K. G. Anagnostakis *et al.* provides a proof-of-concept on apache server application and not on critical infrastructures in general which is not characterization of targeted or untargeted attacks.

2.8.3 Malware Analysis in Targeted Attacks

The analysis of malwares to understand the mode of operation of attackers was demonstrated by O. Thonnard *et al.* in [6]. An in-depth analysis of email malwares identified by *Symantec.cloud* to pinpoint new insights to characteristics of emerging threats on the Internet. The methodology employed in this research was to use an advanced data analytics software framework called TRIAGE. It gives information on how attacks are engineered by extracting features such as email attachment, source IP, dates, email address, mail agent, anti-virus signatures and the mailer agent [6]. These selected features from attacks are clustered to identify similar characteristics found in attacks. O. Thonnard *et al*. assigned different weight to attack features by marking importance to certain features. The result of this experiment was a multi-dimensional attack clusters with a global statistics of attack campaigns [6]. M. Grottke *et al.* created a niche in [55] by presenting a novel approach to assessing the influence of cyber-security attacks in critical infrastructure networks. The research gives metrics and models for giving insight malware campaigns targeting critical infrastructure sectors [55].

Despite the insights provided by these studies, none appeal to research critical infrastructures honeypots. It focuses on the analysis of sophistication of malwares captured and there was no consideration for other indicators such as passwords, services that can give evidence of targeted attacks in research critical infrastructures honeypots.

2.8.4 Unsupervised Classification and Characterizations of Honeypot Attacks

P. Owezarski did a research on unsupervised classification and characterizations of honeypot attacks in [56]. In this research, the major aim is to characterize security exceptions and attacks occurring in honeypots. This is quite problematic because targeted attacks generally do not have to be an anomaly, which gives room for false negatives or false positives. Also, this study does not give any perception between targeted and untargeted attacks in research critical infrastructures honeypots but only provide an automatized way of characterizing honeypot traffic. PCA has been proposed for characterizing honeypot traffic and separating latent groups of activities in low interaction honeypots by S. Almotairi *et al.* in [48]. The research focused on summarizing honeypot traffic and showing interrelationships between group of activities which only seeks to find outliers by using network data. It is important for research honeypot to contain data from targeted attacks but in this study, PCA was not used to give any information or insight to targeted and untargeted attacks and was only applied to honeypot data that were from conventional IT systems not from critical infrastructures.

3 Methodology and Implementation

This section discusses the design and implementation of the honeypot environment setup together with Nigerian CERT. The aim here is to simulate critical infrastructure honeypots in two phases for a thirty-days calendar window in each of the phases. The goal is to gather data from the two phases and to analyze the collected data to identify clear indication of targeted attacks in critical infrastructure research honeypots.

Experiment was carried out to gather honeypot data from deceptive critical infrastructure simulating three sectors in each of the phases. The simulated critical infrastructures were setup outside of real critical infrastructures. The environment provides deceptive mechanism in such a way that it looks real to attackers. In this chapter, the honeypots that were selected and the two experiment phases would be discussed.

3.1 Critical Threats to Critical Infrastructures

Damage, malfunction or unavailability of critical infrastructure could result into economy loss and loss of lives. In identifying attacks and attack agents, threat description helps to have a good architecture and choosing appropriate honeypots. The table 2 below shows 2016 Most Critical ICS Threats.

Table 2: 2016 Most Critical ICS Threats [57]

No.	Threat
1	Social Engineering and Phishing
2	Infiltration of Malware via Removable Media and External Hardware
3	Malware Infection via Internet and Intranet
4	Intrusion via Remote Access
5	Human Error and Sabotage
6	Control Components Connected to the Internet
7	Technical Malfunctions and Force Majeure

8	Compromising of Extranet and Cloud Components
9	(D)DoS Attacks
10	Compromising of Smartphones in the Production Environment

3.2 Honeypot Selection

Modern Honeypot Network (MHN): MHN allows easy management and deployment of honeypots. The following honeypots are currently supported by MHN: Snort, Suricata, Dionaea, Conpot, Kippo, Amun, Glastopf, Wordpot, ShockPot, p0f, Elastichoney.

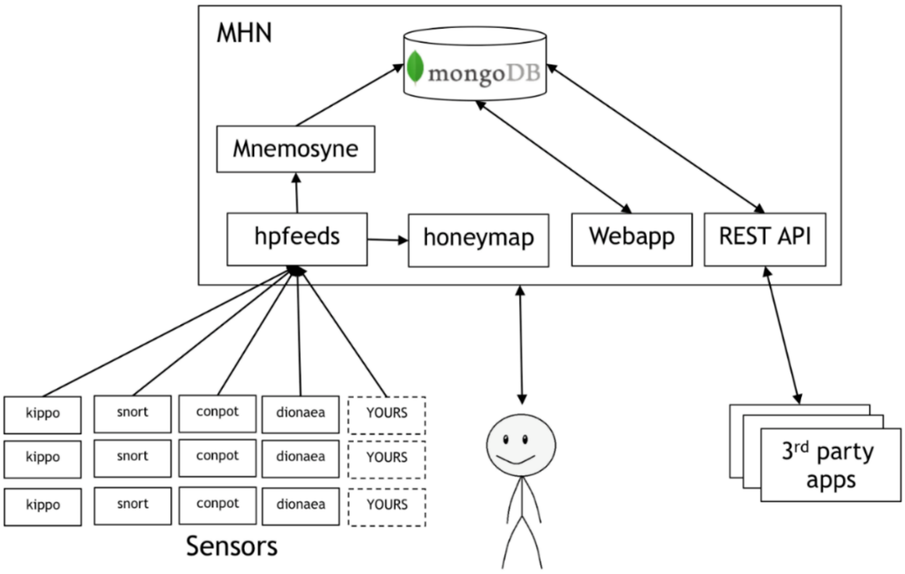


Figure 6: Overview of MHN [58]

3.3 Honeypot Deployment

To have a good basis for analysis of the data to be collected, the experiment was conducted into two phases. Also, data was collected from Nigerian CERT from a SCADA Honeynet setup since 2015. The honeypot is a research honeypot used to gather fashionable information about trends and patterns of attacks on SCADA systems. The next section will discuss the overview of the honeypots previously deployed by Nigerian CERT and the two experiment phases deployed in collaboration with Nigerian CERT; their capabilities and the emulated services.

Table 3: Overview of Selected Honeypot

Honeypot	Type	License	Version
Amun	Lo	GNU General Public License	0.1.7
Glastpof	Lo	GNU General Public License	3.1.2
Conpot	Lo	GNU General Public License	0.5.1
Argos	Lo	Open Source Distribution	0.7
Dionaea	Lo	GNU General Public License	2.0
P0f	Lo	Open Source Distribution	2.0
Elastichoney	Lo	MIT License	0.0.1
Kippo	Me	Open Source Distribution	1.1.5
Cowrie	Me	Open Source Distribution	

3.4 CERT SCADA Honeynet

SCADA honeynet sensors was deployed with 2 public Internet protocols (IPs) facing the Internet. The IP addresses belong to Nigerian CERT. The target environment is another Internet facing device to attackers. However, since the classification of honeypots are based on the interaction of adversaries, it is important to select honeypot that is suitable for the experiment.

3.4.1 Device Specification and Emulated Services

On each of the machines, Ubuntu Server 14.04.5 (LTS) Long Term Support was installed as the operating system. The table below shows the hardware specification and the deployed honeypots.

Table 4: Phase One Device Specification

Operating System	**Ubuntu Server 14.04.5 x64 (LTS)**
Hardware	
- RAM	256MB
- Disc	20GB
Network Interface Cards	2
Honeypots	
- SSH Honeypot	Kippo

- Web Honeypot	Glastpof
- Amun	Vulnerability Emulation Honeypot
- Argos	Emulator for capturing zero day attacks
- Dionaea	Malware Honeypot
- Conpot	SCADA/ICS Honeypot
Intrusion Detection System	Snort

3.5 Experiment Phase One and Two

This section will discuss the deployment of honeypot in two phases.

Simulation of three Critical Infrastructures

Three major sectors of interest were selected to be simulated, namely: Oil and gas, power and banking sectors. Of the three critical infrastructures, banking system is different because it does not make use of ICS; this was selected in other to be able to see the differences in measuring the targeted attack landscape facing different critical infrastructures. The first phase of the experiment focuses on deploying honeypot in each sector without customization of default honeypot service templates. The second phase was more specific in that, templates were customized to look real as much as possible.

After setting up honeypots on the machines, the IPs were declared to a hacking community in darknet and was also crawled by shodan and censys; this declaration was only done in the second phase of the experiment. To browse dark web, tor browser was used. A fake account was created on two different hacker’s platforms on dark web. In these two communities, hackers do share resources and information.

The IP addresses were provided by Nigerian CERT. The target environment is another Internet facing device to attackers. Also, in the second phase of the experiment, the honeypot systems were physically located in Lagos Nigeria.

3.5.1 Power

Recently , the energy sector has become has attracted attacks more and is now among the top five most targeted sectors worldwide [59]. In the power sector, the device of interest is the SCADA device use for distribution. A fake company (called sota plc) was used. MHN was used to deploy honeypots services on the server.

Table 5: Honeypot System Specification for Sota PLC

Hostname	**SotaPLC**
Operating System	Ubuntu Server 14.04.5 x64 (LTS)
Hardware	
- RAM	1GB
- Disc	25GB
Network Interface Cards	2
Honeypots	
- Cowrie	SSH Honeypot
- P0f	OS fingerprinting tool
- Elastic Honey	Elasticsearch
- Dionaea	Malware Honeypot
- Conpot	SCADA/ICS Honeypot
Intrusion Detection System	Snort
No of Public IPs	1

3.5.2 Oil and Gas

The oil and gas honeypot was setup under the fake company name frobe oil. Like power sector a server which is physically located in Lagos, Nigeria was deployed in the second phase as well.

Table 6: Honeypot System Specification for Frobe Oil

Hostname	**Frobe Oil**
Operating System	Ubuntu Server 14.04.5 x64 (LTS)
Hardware	
- RAM	1GB
- Disc	25GB
Network Interface Cards	2
Honeypots	
- Cowrie	SSH Honeypot

- P0f	OS fingerprinting tool
- Elastic Honey	Elasticsearch
- Dionaea	Malware Honeypot
- Conpot	SCADA/ICS Honeypot
Intrusion Detection System	Snort
No of Public IPs	1

3.5.3 Banking

The Banking System is a very critical sector under critical infrastructures. A fake company name GBT Bank was setup and honeypot was deployed under this name. The peculiarity of this is that, it does not make use of conventional ICS infrastructures. The table below shows the honeypots and specification of the machine used in deploying the honeypots.

Table 7: Honeypot System Specification for GBT Bank

Hostname	**Gbt Bank**
Operating System	Ubuntu Server 14.04.5 x64 (LTS)
Hardware	
- RAM	1GB
- Disc	25GB
Network Interface Cards	2
Honeypots	
- Cowrie	SSH Honeypot
- Argos	Zero-day attack capture honeypot
- Dionaea	Malware Honeypot
- Glastpof	Web Honeypot
Intrusion Detection System	Snort
No of Public IPs	2

3.6 Customization of services

In the second phase of the honeypot deployment, default templates in the honeypots were customized after fresh deployment of the honeypots on the machines for the three-simulation environment. It is important to customize service because hackers might know what default honeypot templates looks like. Figure 8 and 9 shows some examples of the customized templates variables such as vendor name, vendor identifier and host IP.

```
<device_info>
    <device_name>Main Distro</device_name>
    <device_identifier>36113</device_identifier>
    <vendor_name>Sota PLC</vendor_name>
    <vendor_identifier>20</vendor_identifier>
    <max_apdu_length_accepted>2048</max_apdu_length_accepted>
  <segmentation_supported>segmentedBoth</segmentation_supported>
    <model_name>VAV-DD Controller</model_name>
    <protocol_version>1</protocol_version>
</device_info>
```

Figure 7: Sample Bacnet Template Customization in Conpot

```
<http enabled="True" host="169.255.57.42" port="80">
  <global>
    <config>
      <entity name="protocol_version">HTTP/1.1</entity>
      <entity name="tarpit">0</entity>
    </config>
    <headers>
      <!-- this date header will be updated, if enabled above -->
      <entity name="Date">Thu, 16 Mar 2017 07:30:00 GMT</entity>
    </headers>
  </global>
 </http>
```

Figure 8: Sample HTTP Customization in Conpot

3.7 Monitoring and Data Collection

To successfully deploy a honeypot, we must correctly deploy the honeypot architecture [60]. In these experiment, monitoring, control and data capture is very important. Majorly, we have three major essential characteristics in honeypot architecture which are as follows; [60], [61], [62]:

- Data capture – monitoring and logging attacks in honeypot systems.
- Data control – this involves controlling and containment of attacker's activities
- Data collection – involves capturing data and storing it in specific locations. [60]

The honeypots systems are remotely managed and controlled via SSH. Data were captured from honeypot logs, IDS logs (in form of pcaps) and system logs. The discussion of data collection and monitoring tools is beyond the scope of this study.

4 Analysis and Result

In this chapter, the analysis of collected data from the experiments would be discussed. Principal component analysis would be used to analyze the data set, password complexities and malwares in each sector would be analyzed.

4.1 Data Preprocessing

In each of the experiment phases, data logs were exported from mongo database and stored JSON file format and was later converted to csv. The log file contains the following information; source IP address, destination IP, source port, destination port, timestamp of attack, the attacked sensor, protocol/service, username, password and payloads. For analysis, missing information in any data instance or raw are discarded.

For each experiment, the entered logs are combined in an excel file for each experiment. The honeynet data from Nigeria CERT contains 1,048,576 records were collected over the course of 21 days. In the first and second experiment, Sota PLC honeypot has a total of 13,252 and 24,160 records, GBT Bank a total of 21,142 and 40,907 records and Frobe Oil has 9,002 and 12,330 records respectively.

4.2 Principal Component Analysis

In reducing high dimensional data, principal component analysis is suitable. It reduces high dimensional variables *p* to smaller number q, which are referred to as principal components [63]. Often, it is possible to retain most of the variability in the original variables with *q* very much smaller than *p* [63]. Principal component premise on the following assumptions [64];

- It does not require any distributional assumptions and can be used with many types of data.
- The extracted principal components are uncorrelated.
- The first few principal components retain most of the variation in the original data.

Since the data set collected is continuous, PCA is suitable to reduce the dimension and the first few components would be analyzed to understand the latent characteristics of the data set.

Considering the volume of the data collected, the goals of PCA are [65];

- extract the most important information from the data;
- compress the size of the data set by keeping only this important information;
- simplify the description of the data set; and
- analyze the structure of the observations and the variables.

The following sections would discuss the input data, the principal components and the characteristics of the observations and variables

4.2.1 Input Data

Principal component analysis requires a square matrix as an input data. The input data used in the analysis are data that gives indication of attacks on the emulated services in the honeypot and was used to form a 24 by 7 matrix. This indicates a total of 24 variables and 7 labels. Each of the labels represents the honeypot environment in the critical infrastructures. The labels are GBTB_1, SPLC_1, FOIL_1, GBTB_2, SPLC_2, FOIL_2 and HN; this represents the observations.

Let A represent the 24 by 7 matrix.

$$A = \begin{pmatrix} x1_1 & x1_2 & \dots x1_{24} \\ \\ x2_1 & x2_2 & \dots x2_{24} \\ . & . & . \\ . & . & . \\ . & . & . \\ x7_1 & x7_2 & \dots x7_{24} \end{pmatrix} \qquad (1)$$

Where the element x_{ij} represents the variable associated to each label; hence we have *i* observations and *j* variables. The tables below show the variables represented by the matric A.

Table 8: Labels for PCA observations

Label	Description
GBTB_1	GBT Bank Honeypot in Phase 1
SPLC_1	Sota PLC Honeypot Phase 1
FOIL_1	Frobe Oil Honeypot Phase 1
GBTB_2	GBT Bank Honeypot in Phase 2
SPLC_2	Sota PLC Honeypot Phase 2
FOIL_2	Frobe Oil Honeypot Phase 2
HN	Pre-existing Honeypot from CERT

Table 9: List of Variables used in PCA

No.	Variables
1	Average number of daily attacks directed towards the honeypot systems
2	Number of malwares captured by dionanea honeypot
3	Total number of SSH Tries
4	Total number of unique SSH tries
5	Number of BACNET Sessions
6	Average Number of Attacked ports
7	Number of IPMI sessions
8	Number of Glastpof Sessions
9	Number of shell emulation offered to attackers
10	Number of MS RPC EndPoint Mapper (EP Mapper)
11	Number of FTPD
12	Number of HTTP session
13	Number of FTPDataListen
14	Number of Microsoft DS
15	Number of Mirrorc
16	Number of Mirrord
17	Number of rejected connection attempt (represented by pcap)
18	Number of Rtp Udp Stream
19	Number of SipCall

20	Number of SipSession
21	Number of smbd
22	Number of Tftp Server Handler Request
23	Number of Passwords greater than 8 characters
24	Number of Unique Passwords
25	Number of Unique IPs

4.2.2 Eigen Vectors

Overall, variables would be preprocessed before analyzing them. This would be done by making sure that the center of gravity of the data is 0. the columns of A (from equation 1) will be centered so that the mean of each column is equal to 0. This is achieved by finding the covariance. “The analysis is referred to as a covariance PCA” [65]. The we can say; $A^T1 = 0$.

Let the standardized matrix be represented by Z, each element in the vector variable, z_i, $i = 1, \ldots, p$. Then, the linear function y for each vector variable in z_i is;

$$y = zv' \qquad (2)$$

where V is the known eigen vector matrix.

Table 10: Resulting Eigen Vectors for PCA

Variables	F1	F2	F3	F4	F5	F6	F7
Average number of daily attacks	0.028	-0.737	0.189	0.539	-0.148	0.047	-0.007
Number of Malwares	0.719	-0.377	0.221	-0.471	0.112	-0.036	0.019
Unique SSH tries	0.005	-0.002	-0.008	0.041	0.059	-0.258	-0.059
Total SSH Session	0.085	-0.042	-0.039	0.020	0.475	-0.191	-0.099
Number of Bacnet Attacks	0.000	0.000	0.001	0.000	-0.001	-0.016	-0.004
Average number of attacked ports	0.086	0.406	0.846	0.154	0.082	-0.136	0.001
Number of IPMI sessions	0.001	0.001	0.000	0.002	-0.017	-0.036	-0.005
Number of Modbus sessions	0.001	0.000	0.001	0.002	-0.020	-0.059	-0.008
Web Attack	0.015	-0.008	-0.025	-0.040	0.136	0.148	0.021
emulation	0.000	0.000	0.001	0.000	0.001	0.006	0.001
epmapper	0.000	-0.001	0.001	-0.001	0.000	0.004	0.000

ftpdatalisten	0.000	0.000	0.000	0.000	0.000	0.000	0.000
ftpd	0.001	0.000	-0.005	-0.004	0.014	-0.013	-0.003
http	0.012	0.004	-0.039	-0.061	0.166	-0.224	-0.066
microsoft-ds	0.000	0.000	0.000	0.000	-0.001	0.000	0.000
mirrorc	0.000	0.000	0.000	0.000	0.001	-0.001	0.000
mirrord	0.000	0.000	0.000	0.000	0.001	0.000	0.001
mssqld	0.001	0.001	0.000	-0.003	0.002	0.007	0.002
mysqld	0.030	0.054	-0.072	-0.114	0.413	0.638	0.076
pcap	0.677	0.380	-0.303	0.431	-0.267	0.120	-0.017
RtpUdpStream	0.002	0.002	-0.001	0.002	0.014	-0.047	-0.014
SipCall	-0.005	0.006	-0.110	-0.460	-0.445	-0.268	-0.031
SipSession	0.088	0.013	-0.293	0.208	0.439	-0.531	0.041
smbd	0.002	0.002	-0.008	-0.005	-0.004	-0.024	0.103
TftpServerHandler	0.001	0.000	-0.002	-0.005	0.003	-0.014	0.393
Password > 8	0.000	-0.002	-0.003	0.006	0.009	-0.074	0.894
Number of Unique Passwords	0.006	0.017	0.038	0.047	-0.223	-0.119	0.092

4.2.3 Principal Components

The linear combination of the reduced data set can be seen from equation (1), hence we have the matrix notation.

$$Y = ZV \qquad (3)$$

Principal components F1-F7 represents the linear combination of the reduced data as seen in the table below. 100% of the entire data have been represented by 7 principal components. The principal components F1 and F2 represents 95.901% of the total variables; this indicates that, F1 and F2 depicts a 95.901% of the latent characteristics of the data.

Table 11: Resulting Principal Component

	F1	F2	F3	F4	F5	F6	F7
Eigenvalue	200373985.971	73326670.597	5901781.271	5285735.479	408757.948	103547.238	0.000
Variability (%)	70.208	25.693	2.068	1.852	0.143	0.036	0.000
Cumulative %	70.208	95.901	97.968	99.820	99.964	100.000	100.000

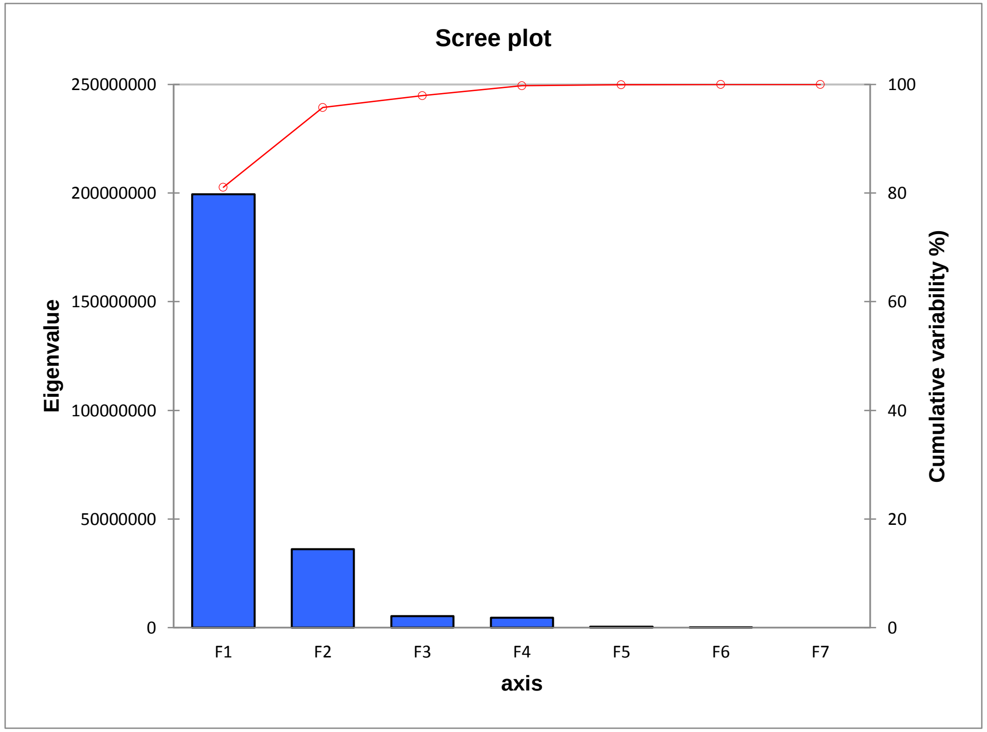


Figure 9: Scree Plot of Principal Components

4.2.4 Variables and Quality of Projections

To understand the relationships between variables, the first two components would be used. The first two principal components (F1 and F2) that represents 95.774% of the data are plotted on a correlation circle. Correlation is measured between the range -1 and 1. It is important to interpret variables that are well projected, that is, variables that a projected far from the origin. There are 3 possibilities in measuring the correlation between the variables [66];

- Significantly positively correlated variables (r close to 1)
- Uncorrelated (r close to 0)
- Significantly negatively variables (r close to -1).

The correlation circle figure below shows the correlation between variables. The average number of attacks and number of unique IPs are significantly positively correlated. Total Number of SSH connections and number of malwares are significantly positively correlated as well.

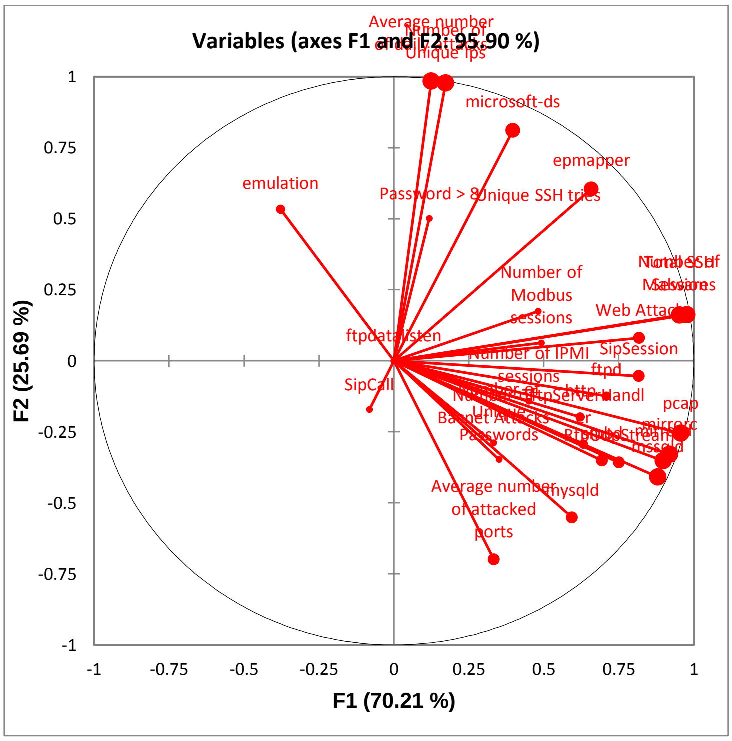


Figure 10: Variable Correlation Circle

To better understand the variables' projections on the scale of -1 and 1, the squared cosines of variables provides a better insight in interpreting the variable projections in the correlation circle. The table below shows the squared cosine of variables.

Table 12: Squared Cosines of Variables

Variables	**F1**	**F2**	**F3**	**F4**	**F5**
Average number of daily attacks	0.015	**0.968**	0.015	0.000	0.000
Number of Malwares	**0.955**	0.026	0.010	0.008	0.000
Unique SSH tries	0.232	0.031	**0.249**	0.134	0.068
Total SSH Session	**0.908**	0.026	0.001	0.005	0.058
Number of Bacnet Attacks	**0.110**	0.083	0.074	0.031	0.025
Average number of attacked ports	0.111	**0.489**	0.182	0.218	0.000
Number of IPMI sessions	0.203	0.020	0.045	0.019	**0.370**
Number of Modbus sessions	0.242	0.004	0.055	0.000	**0.260**
Web Attack	**0.670**	0.006	0.182	0.003	0.113
emulation	0.143	**0.284**	0.074	0.225	0.061
epmapper	**0.434**	0.366	0.049	0.129	0.000
ftpdatalisten	**0.000**	0.000	0.000	0.000	0.000
ftpd	**0.502**	0.015	0.230	0.086	0.136
http	**0.387**	0.039	0.328	0.015	0.153
microsoft-ds	0.157	**0.658**	0.108	0.002	0.075
mirrorc	**0.846**	0.108	0.003	0.030	0.009
mirrord	**0.806**	0.124	0.002	0.059	0.008
mssqld	**0.774**	0.167	0.042	0.006	0.004
mysqld	**0.354**	0.304	0.128	0.008	0.142
pcap	**0.915**	0.065	0.007	0.012	0.000
RtpUdpStream	**0.564**	0.129	0.022	0.018	0.071
SipCall	0.007	0.030	**0.853**	0.023	0.079
SipSession	**0.668**	0.003	0.008	0.277	0.034
smbd	**0.483**	0.124	0.191	0.141	0.009
TftpServerHandler	0.400	0.086	**0.431**	0.002	0.012
Password > 8	0.014	**0.251**	0.042	0.135	0.030
Number of Unique Passwords	0.123	0.121	**0.362**	0.024	0.354

Number of Unique IPs	0.030	**0.955**	0.014	0.001	0.000

Note: Values in bold correspond for each variable to the factor for which the squared cosine is the largest.

The resulting principal components and observation can be interpreted by considering the table 12 of squared cosines of variables. In PCA, it is important to understand that, the greater the squared of cosine, the greater the link to the corresponding axis. Well projected variables can be seen in the correlation circle in figure 10. Therefore, we will consider variables with squared cosines value greater than 0.5 which is seen in table 13 below.

Table 13: Well Projected Variables in Principal Components

First Component (F1)	Second Component (F2)	Third Component (F3)
Number of Malwares	Number of Unique IPs	SipCalls
SSH Sessions	Microsoftds	
Web Attack	Attacks Ports	
FTPD	Number of attacks	
Mirrorc		
Mirrord		
Mssqld		
Pcap		
RtpUdpStream		
SipSession		

4.2.5 Observations

As PCA is exploratory data analysis method which reveals the latent structure of the data and better give us the variance in the data set. An insight to the behavior of each honeypot systems could be seen in the biplot of variables and observation in figure 11 below.

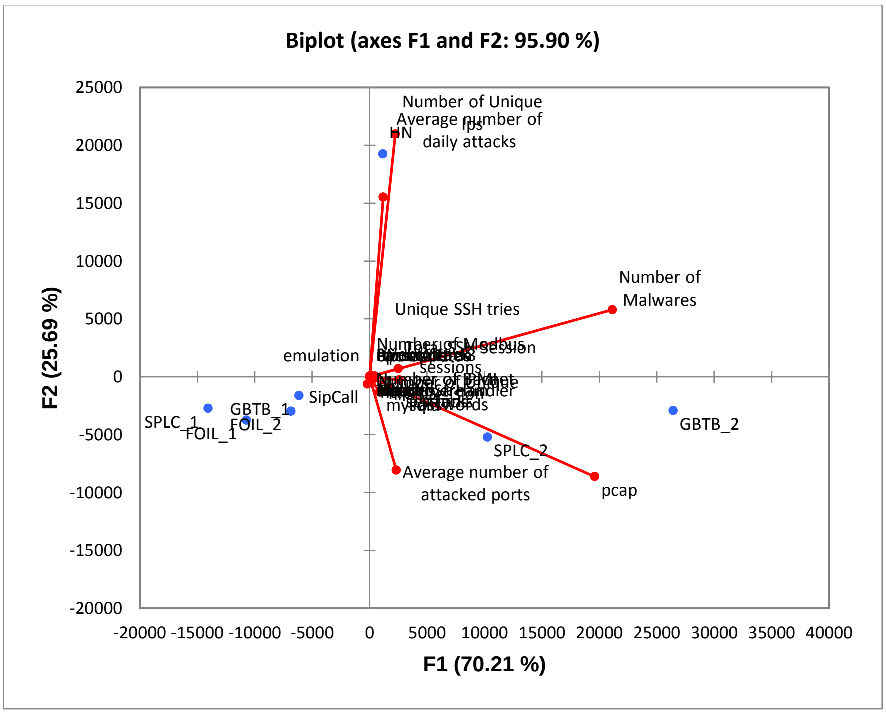


Figure 11: Biplot of variables and observations

The observation plot shows the interrelationship between the variables and the observation.

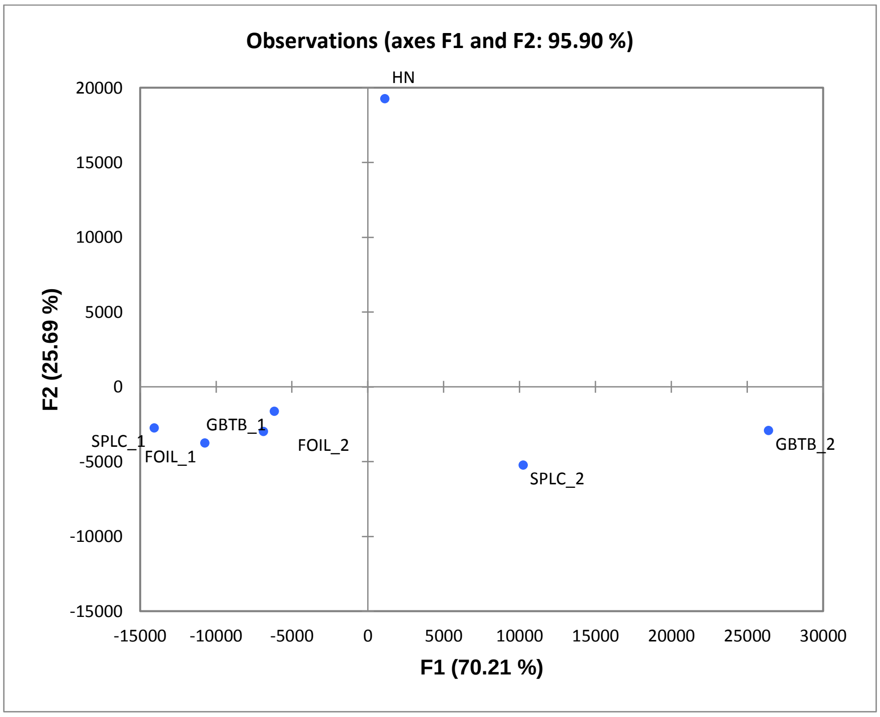


Figure 12: Biplot of Observations of the First two principal Components

In general, considering the honeypot GBTB_1, SPLC_1, FOIL_1, GBTB_2, SPLC_2, FOIL_2 and HN. The GBT Bank honeypot in the second phase attracted the most number of attacks as seen in the correlation circle and biplot of observation in figure 11 and 13; it is well projected.

Variables and Observations

- Extreme malware activity on GBTB_2
- High number of rejected malware connections towards SPLC_2
- SipCall projection towards FOIL_2
- High volume of Unique SSH tries towards HN
- Password greater than 8 directed towards HN

Base on the observation of the well projected variables in the PCA, further investigation of web attacks, passwords and malwares was conducted. Section 4.3, 4.4 and 4.5 would discuss the analysis and observations

4.3 Password Complexity

Although passwords greater that eight characters are well projected towards HN, this variable behavior would be analyzed for each system in each phase. In this section, we analyze the passwords used in attempts to gain access to the honeypot systems. Attackers tried a range of password combinations which includes; lower case alphabets, upper case alphabet, numbers, alphanumeric and mixed alphanumeric. The complexity of passwords can give indication of targeted attacks. Passwords that contains only numbers or only alphabets are of low complexity and passwords whose length is less than eight characters long. Also, another consideration is the character set ordering of the passwords in terms of strings, digit and special characters' combinations. The following subsections would discuss the password complexities in the two experiment phases and the observations.

4.3.1 Phase One

In the first phase of the experiment, the power honeypot 403 passwords were analyzed. 332 (82.38%) were one to six characters long. 390 (96.77%) were one to eight characters while more than eight characters long. For GBT Bank, 248 passwords were analyzed with 74 unique entries. 94 (37.9%) were one to six characters, 150 (60.48%) were one to eight characters and 98 (39.52%) were more than eight characters. In oil and gas, 228 passwords were analyzed with 35 unique passwords, 193 (84.65%) were one to six characters long. 217 (95.18%) were one to eight characters long and 11 (4.82%) were more than eight characters.

The tables below show the character complexities of the passwords.

Table 14: Phase 1 - Character Set of Password

Character Sets	Sota PLC	GBT Bank	Frobe Oil
Lower Alpha	376 (93.3%)	119 (47.98%)	144 (63.16%)
Lower Alphanumeric	22 (5.46%)	72 (29.03%)	10 (4.39%)
Mixed Alphanumeric	2 (0.5%)	2 (0.81%)	1 (0.44%)
Numeric	3 (0.74%)	41 (16.53%)	69 (30.26%)
Mixed Alpha	-	10 (4.03%)	2 (0.88%)
Upper Alpha	-	1 (0.4%)	-
Upper Alpha Special	-	1 (0.4%)	-
Lower Alpha Special	-	1 (0.4%)	-

Table 15: Phase 1 - Character Set Ordering

Character Set Ordering	Sota PLC	GBT Bank	Frobe Oil
All Strings	376 (93.3%)	130 (52.42%)	146 (64.04%)
String Digit	15 (3.72%)	72 (29.03%)	10 (4.39%)
Other Mask	6 (1.49%)	2 (0.81%)	3 (1.32%)
All Digit	3 (0.74%)	41 (16.53%)	69 (30.26%)
String-Digit-String	3 (0.74%)	-	-
Digit-String-Digit	-	2 (0.81%)	-
String-Special-String	-	1 (0.4%)	-

Table 16: Phase 1 - Password Length Variation

Password Length	Sota PLC	Password Length	GBT Bank	Password Length	Frobe Oil
1	0.25%	1	3.23%	1	(7.46%)
2	3.47%	2	2.42%	2	(0.88%)
3	31.02%	3	4.44%	3	(3.07%)

4	14.89%	4	12.1%	4	(25.0%)
5	19.85%	5	8.87%	5	(39.91%)
6	12.9%	6	6.85%	6	(8.33%)
7	8.93%	7	8.06%	7	(1.75%)
8	5.46%	8	14.52%	8	(8.77%)
9	0.99%	9	7.26%	9	(1.75%)
10	0.99%	10	16.53%	10	(2.63%)
11	0.99%	11	13.31%	-	-
12	0.25%	13	0.4%	12	(0.44%)
-	-	14	0.4%		
-	-	15	0.4%		
-	-	16	0.4%		
-	-	19	0.4%		
-	-	30	0.4%		

4.3.2 Phase Two

In the power honeypot, a total number of 825 passwords were analyzed. 691 of the 825 were unique entries. 664 (80.48%) of those passwords are one to six characters long, 798 (96.73%) are one to eight characters long and 27 (3.27%) are more than eight characters long. For GBT Bank honeypot, 294 passwords were analyzed which has 94 unique entries. 217 (87.15%) of these passwords are one to eight characters long and 32 (12.85%) are more than eight characters long. In the case of Frobe oil, 712 passwords were examined which has 119 unique entries. 628 (88.2%) of those passwords are one to eight characters long while 84 (11.8%) are longer than eight characters. The tables below show the character complexities of the passwords.

Table 17: Phase 2 - Character Set of Password

Character Sets	Sota PLC	GBT Bank	Frobe Oil
Lower Alpha	767 (92.97%)	157 (63.05%)	401 (56.32%)
Lower Alphanumeric	38 (4.61%)	26 (10.44%)	89 (12.5%)
Mixed Alphanumeric	9 (1.09%)	3 (1.2%)	7 (0.98%)
Numeric	7 (0.85%)	45 (18.07%)	160 (22.47%)

Mixed Alpha	4 (0.48%)	10 (4.02%)	23 (3.23%)

Table 18: Phase 2 - Character Set Ordering

Character Set Ordering	Sota PLC	GBT Bank	Frobe Oil
All Strings	767 (92.97%)	78 (31.33%)	425 (59.69%)
String Digit	27 (3.27%)	25 (10.04%)	82 (11.52%)
Other Mask	16 (1.94%)	8 (3.21%)	39 (5.48%)
All Digit	7 (0.85%)	45 (18.07%)	160 (22.47%)
String-Digit-String	4 (0.48%)	90 (36.14%)	-
Digit-String-Digit	4(0.48%)	2 (0.8%)	6 (0.84%)
String-Special-String	-	1 (0.4%)	-

Table 19: Phase 2 - Password Length Variation

Password Length	Sota PLC	Password Length	GBT Bank	Password Length	Frobe Oil
1	0.36%	1	3.21%	1	4.35%
2	2.91%	2	2.41%	2	0.84%
3	29.21%	3	5.62%	3	3.09%
4	14.79%	4	3.61%	4	24.3%
5	17.7%	5	14.86%	5	18.68%
6	15.52%	6	0.4%	6	8.29%
7	8.36%	7	10.44%	7	8.85%
8	7.88%	8	16.06%	8	19.8%
9	0.97%	9	14.46%	9	4.35%
10	1.7%	10	2.81%	10	2.53%
11	0.48%	11	3.21%	11	2.81%
12	0.12%	12	20.08%	12	0.14%
-	-	13	0.4%	13	0.42%
-	-	14	0.4 %	14	0.28%
-	-	15	0.4 %	15	0.28%
-	-	-	-	16	0.28%

-	-	-	-	18	0.14%
-	-	-	-	19	0.28%
-	-	-	-	32	0.14%
	-	-	-	35	0.14%

4.3.3 Observations

In general, majority of the passwords that was captured are not complex. However, phase 2 honeypots attracted more number of passwords and the passwords are more complex (in terms of length, character sets and character set ordering). Passwords with mixed alpha and alphanumeric in GBT Bank, Sota PLC and Frobe Oil increased in phase two from 4.84% to 5.22%, 0.5% to 2.38% and 1.32 to 4.21% respectively.

Passwords character set ordering in terms of String-Digit, String-Digit-String, Digit-String and String-Special-String in GBT Bank, Sota PLC and Frobe oil increased from 31.05 to 50.59, 5.95% to 7.02% and 4.39% to 17.84% respectively. In terms of password length that are greater than eight characters, in GBT Bank, Sota PLC and Frobe Oil attracted more passwords in phase two from 54.02 to 57.82, 8.68% to 11.15% and 13.59 to 31.55 respectively.

4.4 Malwares

In both phases of the experiments, the dionaea honeypot captured malwares with the same hashes across simulated honeypots, however, in the GBT Bank honeypot during the phase two experiment, the dionaea honeypot captured four malwares with hashes of malware that were not found on other honeypots. This indicate that the GBT attracted a targeted malware. The following are the hashes of malwares:

- bb18c488bafbc2b5d5d01f6abfdcb3dc
- ab30cb38efe604bf4a96df8e879bcdb8
- 80547f054bdf134c69b11fd0ee9339dd
- cbfc90a3a4359950ecdde594a4a0e149

The hashes of these malwares were submitted on virus total (https://www.virustotal.com/) for analysis, these malwares are mostly trojans. See Appendix for analysis result from virus total.

Table 20: New malwares attracted by Phase two GBT bank honeypot

MD5 Hash	First Submission on Virus Total
bb18c488bafbc2b5d5d01f6abfdcb3dc	2017-04-05 08:43:24 UTC
ab30cb38efe604bf4a96df8e879bcdb8	2017-04-05 08:43:24 UTC
80547f054bdf134c69b11fd0ee9339dd	2017-04-05 08:43:24 UTC
cbfc90a3a4359950ecdde594a4a0e149	2017-03-25 11:04:06 UTC

4.5 Web Attacks

The GBT Bank web environment emulated SQL injection vulnerabilities using glastpof as discussed earlier. The SQL injection in phase two looks more sophisticated compared to the one in phase two. The average number of SQL attack directed in the first phase is 15 characters long while the one directed in the second phase is 21 characters long. Sample SQL injection attacks in the first and second phase can be seen below

```
1' OR '1'='1'
1 AND USER_NAME() = 'dbo'
1' AND 1=(SELECT COUNT(*)
FROM tablenames)
1 AND 1=1
1 EXEC XP_
1'1
```

Figure 13: Sample Phase 1 SQL Injection

```
1 UNI/**/ON SELECT ALL FROM WHERE
1 AND ASCII(LOWER(SUBSTRING((SELECT TOP 1 name
FROM sysobjects WHERE xtype='U'), 1,1))) > 116
1 UNION ALL SELECT 1,2,3,4,5,6,name FROM
sysObjects WHERE xtype = 'U' --
' OR username IS NOT NULL OR username = '1' AND
non_existant_table = '1

1'1
1' AND 1=(SELECT COUNT(*) FROM tablenames)
```

Figure 14: Sample Phase 2 SQL Injection

In addition to the SQL injections, a malicious activity of an international hacker (Hmei7) was detected in the phase two [67]. The attacker tried to deface the website. Hmei7 used the code snippet below.

```
<?php echo
'<b><br><br>'.php_uname().'<br></b>'; echo
'<form action="" method="post"
enctype="multipart/form-data" name="uploader"
id="uploader">'; echo '<input type="file"
name="file" size="50"><input name="_upl"
type="submit" id="_upl"
value="Upload"></form>'; if( $_POST['_upl'] ==
"Upload" ) { if(@copy($_FILES['file']['tmp_name'],
$_FILES['file']['name'])) { echo '<b>Upload SUKSES
!!!</b><br><br>'; } else { echo '<b>0wnerd by
Hmei7</b><br><br>'; } } ?>
```

Figure 15: Code snippet for attempt website defacement

5 Conclusion and Future Work

In this thesis, background study in chapter 2 was done to show the need for identifying targeted and untargeted attacks in research critical infrastructures honeypot. The major contribution is the simulation of three critical infrastructure honeypots and the announcement of those honeypots in dark web. Chapter 3 discusses the two phases experiments carried out to gather data for analysis and the data set gotten from CERT Nigeria. In chapter 4, exploratory data analysis technique was used to analyze the data set; several variables provide insight to understanding how to identify targeted and untargeted attacks was analyzed using principal component analysis. The result shows a variation in the two experiment phases.

In research honeypots, it can be concluded that a high percentage of data collected from these honeypots are from untargeted attacks as there are no indication that an attacker clearly selected a victim to attack. However, for research honeypots to attract targeted attacks, researchers need to go through the overhead needed in customizing honeypots to suit simulated sector to attract targeted attacks. PCA proves to be exploratory data analysis tool to analyze huge data set from honeypots. In conclusion, from phase 2 of the experiment, Nigerians banks seems to attract more targeted attacks which gives an indication of targeted attacks from malwares, web exploits and password complexity than other critical infrastructures.

To improve classification of targeted and untargeted attacks, machine learning approaches can be used to improve on the results from an exploratory data analysis technique such as PCA. A suitable machine learning technique is the k-nearest neighbor algorithm for continuous data set, which is a non-parametric method for classification and regression. This would help to understand data collected from critical infrastructure honeypots are from targeted sources or not.

This study has also demonstrated that exploratory data analysis technique is suitable for formal modeling, and has shown some insights to identifying of targeted and untargeted attacks in critical infrastructure especially when observing the projection of variables and the variation in the observation labels in section 4.2.3 and 4.2.4. This is because PCA reduced the dimension of the data collected represented with principal components without losing the latent characteristics of the original data collected. Also, a consideration for future work is to run these experiments in a controlled environment setup. In a controlled environment, more set of insights would be seen which could also be applied in production honeypots.

References

[1] M. J. Assante, "Confirmation of a Coordinated Attack on the Ukrainian Power Grid," *Confirmation of a Coordinated Attack on the Ukrainian Power Grid*, 2016. [Online]. Available: https://ics.sans.org/blog/2016/01/09/confirmation-of-a-coordinated-attack-on-the-ukrainian-power-grid#. [Accessed: 16-Mar-2017].

[2] E. Kovacs, "Attackers Alter Water Treatment Systems in Utility Hack: Report," 2016. [Online]. Available: http://www.securityweek.com/attackers-alter-water-treatment-systems-utility-hack-report. [Accessed: 16-Mar-2016].

[3] J. Lemay, Antoine Fernandez and S. Knight, "An isolated virtual cluster for SCADA network research," *ICS-CSR 2013 Proc. 1st Int. Symp. ICS SCADA Cyber Secur. Res. 2013*, p. Pages 88-96, 2013.

[4] O. Andreeva, S. Gordeychik, G. Gritsai, and O. Kochetova, "Industrial Control Systems and Their Online Availability," pp. 1–16, 2016.

[5] N. Chaffin, May;Trent, "Common Cybersecurity Vulnerabilities in Industrial Control Systems," *Dhs*, no. May, p. 88, 2011.

[6] O. Thonnard, L. Bilge, G. O'Gorman, S. Kiernan, and M. Lee, "Industrial espionage and targeted attacks: Understanding the characteristics of an escalating threat," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7462 LNCS, pp. 64–85, 2012.

[7] B. Smith and J. Boyle, *Lecture notes in computer science*. 2009.

[8] D. R. V. A. S. Kumar, A. Taranum, and M. S. Goud, "Technique for Migration to IPV6 for a Secure SCADA Architecture," vol. 4, no. 4, pp. 128–133, 2014.

[9] R. J. Robles and M.-K. Choi, "Assessment of the Vulnerabilities of SCADA , Control Systems and Critical Infrastructure Systems," *Int. J. Grid Distrib. Comput.*, vol. 2, no. 2, pp. 27–34, 2009.

[10] V. Urias, B. Van Leeuwen, and B. Richardson, "Supervisory Command and Data Acquisition (SCADA) system cyber security analysis using a live, virtual, and constructive (LVC) testbed," *Proc. - IEEE Mil. Commun. Conf. MILCOM*, no. Lvc, pp. 1–8, 2012.

[11] W. T. Shaw, *Cyber Security for SCADA Systems*. Penwell, 2006.

[12] C. Xenakis, *Critical Information Infrastructure Security*, 2010th ed. New York, 2010.

[13] I. Manual, "Anr-Lan," pp. 1–36.

[14] Office of the Manager National Communications System, “Supervisory Control and Data Acquisition (SCADA) Systems,” *Tech. Inf. Bull. 04-1*, no. October, p. 76, 2004.

[15] C. Alcaraz, G. Fernandez, and F. Carvajal, “Security aspects of SCADA and DCS environments,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7130, pp. 120–149, 2012.

[16] N. Quist, “Active Defense through Deceptive Configuration Techniques,” *Inf. Secur.*, 2009.

[17] J. K. Burgoon and D. B. Buller, “Interpersonal deception: III. Effects of deceit on perceived communication and nonverbal behavior dynamics,” *J. Nonverbal Behav.*, vol. 18, no. 2, pp. 155–184, 1994.

[18] U. Eric J. Holdaway, Major, “Active Computer & Network Defense,” no. April, p. 43, 2011.

[19] M. A. McQueen and W. F. Boyer, “Deception used for cyber defense of control systems,” *Proc. - 2009 2nd Conf. Hum. Syst. Interact. HSI ’09*, pp. 624–631, 2009.

[20] L. Spitzner, *Honeypots: Tracking Hackers”*. Addislon-Wesley, 2002.

[21] F. Zhang, S. Zhou, Z. Qin, and J. Liu, “Honeypot: a supplemented active defense system for network security,” *Parallel Distrib. Comput. Appl. Technol. 2003. PDCAT’2003. Proc. Fourth Int. Conf.*, pp. 231--235, 2003.

[22] I. Mokube and M. Adams, “Honeypots: Concepts, Approaches, and Challenges,” *Proc. 45th Annu. southeast Reg. Conf. - ACM-SE 45*, pp. 321–326, 2007.

[23] B. ALEKHYA, “Virtual CMS Honey pot capturing threats,” vol. 4, no. 4, pp. 1492–1497, 2013.

[24] Honeynet Project, “Know Your Enemy: Honeynets.” [Online]. Available: http://old.honeynet.org/papers/honeynet/. [Accessed: 06-Apr-2017].

[25] P. K. Christian Seifert, Ian Welch, “Taxonomy of Honeypots,” 2006.

[26] C. Seifert, I. Welch, and P. Komisarczuk, “HoneyC - The Low-Interaction Client Honeypot Client honeypots are necessary because they are,” no. August, pp. 1–9, 2006.

[27] G. Wicherski, “Medium interaction honeypots,” *Ger. Honeynet Proj.*, 2006.

[28] X. Fu, W. Yu, D. Cheng, X. Tan, K. Streff, and S. Graham, “On recognizing virtual honeypots and countermeasures,” *Proc. - 2nd IEEE Int. Symp. Dependable, Auton. Secur. Comput. DASC 2006*, pp. 211–218, 2006.

[29] J. Göbel, “Amun: Automatic Capturing of Malicious Software.,” *Sicherheit*, 2010.

[30] C. S. Vetsch, M. Koßin, and M. Mauer, “Know Your Tools : Glastopf,” *Honeynet Proj.*, pp. 1–29, 2010.

[31] D. Rist, L., Vestergaard, J., Haslinger, “Conpot,” 2014. [Online]. Available:

http://conpot.org/.

[32] C. Blended, M. Threat, and A. F. A. Software, “InfoSec Reading Room,” *Inf. Secur.*, 2009.

[33] T. Wiens, “S7 Communication (S7comm),” 2016. [Online]. Available: https://wiki.wireshark.org/S7comm. [Accessed: 04-Apr-2017].

[34] R. Fielding, U. C. Irvine, and J. Gettys, “Hypertext Transfer Protocol -- HTTP/1.1,” pp. 1–176, 1999.

[35] Cisco, “Simple Network Management Protocol,” pp. 1–8, 2013.

[36] C. Corporation, U. T. Corporation, and C. No, “BACnet Basics User’s Guide,” no. 11, 2013.

[37] W. Fischer, “IPMI Basics,” 2010. [Online]. Available: https://www.thomas-krenn.com/en/wiki/IPMI_Basics. [Accessed: 05-Apr-2017].

[38] G. Portokalidis, A. Slowinska, and H. Bos, “Argos: An Emulator for Fingerprinting Zero-Day Attacks for Advertised Honeypots with Automatic Signature Generation,” *Proc. 1st SIGOPS/EuroSys Eur. Conf. Comput. Syst.*, vol. 40, pp. 15–27, 2006.

[39] M. Schloesser, “Dionaea Honeypot,” 2013. [Online]. Available: https://github.com/rep/dionaea.

[40] T. Sochor and M. Zuzcak, “Study of Internet Threats and Attack Methods Using Honeypots and Honeynets,” *Commun. Comput. Inf. Sci.*, vol. 431, pp. 118–127, 2014.

[41] J. Y.-C. Cheng, “Updates and highlight from recent honeypot tools development,” 2016.

[42] J. Wright, “Elastichoney,” 2015. [Online]. Available: https://github.com/jordan-wright/elastichoney. [Accessed: 01-Jul-2016].

[43] Y. Zheng and X. Zhang, “Path sensitive static analysis of web applications for remote code execution vulnerability detection,” *Proc. - Int. Conf. Softw. Eng.*, pp. 652–661, 2013.

[44] M. Oosterhof, “Cowrie Honeypot,” 2017. [Online]. Available: https://github.com/micheloosterhof/cowrie. [Accessed: 02-Apr-2017].

[45] F. C. Cheng and W. H. Lai, “The prospects of jurisdictional issues in cyberspace,” *Proc.- 2011 IEEE Int. Conf. HPCC 2011 - 2011 IEEE Int. Work. FTDCS 2011 -Workshops 2011 Int. Conf. UIC 2011- Work. 2011 Int. Conf. ATC 2011*, pp. 916–923, 2011.

[46] R. BEJTLICH, “Understanding the advanced persistent threat.” [Online]. Available: http://searchsecurity.techtarget.com/magazineContent/Understanding-the-advanced-persistent-threat. [Accessed: 07-Apr-2017].

[47] I. T. Jolliffe, “Principal Component Analysis, Second Edition,” *Encycl. Stat. Behav. Sci.*, vol. 30, no. 3, p. 487, 2002.

[48] S. Almotairi, A. Clark, G. Mohay, and J. Zimmermann, “Characterization of attackers’ activities in honeypot traffic using principal component analysis,” *Proc. - 2008 IFIP Int. Conf. Netw. Parallel Comput. NPC 2008*, pp. 147–154, 2008.

[49] S. Wade, “SCADA Honeynets: The attractiveness of honeypots as critical infrastructure security tools for the detection and analysis of advanced threats,” *Iowa State Univ.*, p. 67, 2011.

[50] T. Sochor and M. Zuzcak, “Attractiveness Study of Honeypots and Honeynets in Internet Threat Detection,” vol. 522, pp. 69–81, 2015.

[51] S. Sexton and D. Zilberman, “Detecting Targeted Attacks Using Shadow Honeypots,” *Biotechnology*, pp. 1–40, 2010.

[52] R. S. Ramachandruni and P. Poornachandran, “Detecting the network attack vectors on SCADA systems,” *2015 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2015*, pp. 707–712, 2015.

[53] N. Sayegh, A. Chehab, I. H. Elhajj, and A. Kayssi, “Internal security attacks on SCADA systems,” *2013 3rd Int. Conf. Commun. Inf. Technol. ICCIT 2013*, pp. 22–27, 2013.

[54] E. Vasilomanolakis, S. Srinivasa, and C. G. Cordero, “Multi-stage attack detection and signature generation with ICS honeypots,” *Proc. NOMS 2016 - 2016 IEEE/IFIP Netw. Oper. Manag. Symp.*, pp. 1227–1232, 2016.

[55] M. Grottke, A. Avritzer, D. S. Menasch??, J. Alonso, L. Aguiar, and S. G. Alvarez, “WAP: Models and metrics for the assessment of critical-infrastructure-targeted malware campaigns,” *2015 IEEE 26th Int. Symp. Softw. Reliab. Eng. ISSRE 2015*, pp. 330–335, 2016.

[56] P. Owezarski, “Unsupervised classification and characterization of honeypot attacks,” *Proc. 10th Int. Conf. Netw. Serv. Manag. CNSM 2014*, pp. 10–18, 2015.

[57] I. C. Systems and T. Ics, “BSI Publications on Cyber-Security,” pp. 1–20, 2014.

[58] M. Wollenweber, “Modern Honeypot Network,” 2015. [Online]. Available: https://github.com/threatstream/mhn.

[59] C. Wueest, “Targeted Attacks Against the Energy Sector,” *Symantec Corp.*, pp. 1–29, 2014.

[60] P. Sokol, P. Pekarčík, and T. Bajtoš, “Data Collection and Data Analysis in Honeypots and Honeynets 1 Introduction.”

[61] F. H. Abbasi and R. J. Harris, “Experiences with a generation III virtual honeynet,” *2009 Australas. Telecommun. Networks Appl. Conf. ATNAC 2009 - Proc.*, no. July 2014, 2009.

[62] L. Spitzner, *Honeypots: Tracking Hackers*. 2002.

[63] I. Jolliffe, “Principal Component Analysis,” *Wiley StatsRef Stat. Ref. Online*, pp. 1–5, 2014.

[64] F. A. G. Sharma and S. B. M. Kaur, “Verification of Detection of Principal Components in Low Interaction Honeypots using StatistiXL Tool,” vol. 4, no. 2, pp. 11–14, 2010.

[65] H. Abdi and L. J. Williams, “Principal component analysis,” *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 2, no. 4, pp. 433–459, 2010.

[66] XLSTAT, “Principal component analysis (PCA) in Excel,” 2017. [Online]. Available: https://help.xlstat.com/customer/en/portal/articles/2062222-running-a-principal-component-analysis-pca-with-xlstat?b_id=9283. [Accessed: 28-Apr-2017].

[67] E. Kovacs, “Hackers Around the World: Hmei7, Indonesian Defacer,” 2013. [Online]. Available: http://news.softpedia.com/news/Hackers-Around-the-World-Hmei7-Indonesian-Defacer-361176.shtml. [Accessed: 27-Apr-2017].

Appendix 1

```
[ file data ]

* name..: ab30cb38efe604bf4a96df8e879bcdb8

* size..: 74752

* md5...: ab30cb38efe604bf4a96df8e879bcdb8

* sha1..: 972e7429cac8b4b43d8b2789ea973bd8f044e230


[ scan result ]

ALYac   1.0.1.9/20170508       found DeepScan:Generic.Sdbot.7216E179

AVG     16.0.0.4776/20170508    found Generic_r.RUD

AVware  1.5.0.42/20170508      found Trojan.Win32.Generic!BT

Ad-Aware       3.0.3.1010/20170508     found DeepScan:Generic.Sdbot.7216E179

AegisLab       4.2/20170508    found Troj.W32.Generic!c

AhnLab-V3      3.9.0.17440/20170508    found nothing

Antiy-AVL      1.0.0.1/20170508       found Worm/Win32.AGeneric

Arcabit 1.0.0.802/20170508      found DeepScan:Generic.Sdbot.7216E179

Avast   8.0.1489.320/20170508   found Win32:Malware-gen

Avira   8.3.3.4/20170507       found TR/Dropper.Gen

Baidu   1.0.0.2/20170503       found Win32.Trojan.WisdomEyes.16070401.9500.9985

BitDefender     7.2/20170508    found DeepScan:Generic.Sdbot.7216E179

Bkav    1.3.0.8876/20170506     found nothing

CAT-QuickHeal   14.00/20170508  found Trojan.Generic

CMC     1.1.0.977/20170507      found nothing

ClamAV  0.99.2.0/20170508      found nothing

Comodo  27052/20170508  found nothing

CrowdStrike     1.0/20170130    found malicious_confidence_99% (W)

Cyren   5.4.30.7/20170508      found nothing

DrWeb   7.0.28.2020/20170508      found Trojan.DownLoader24.27110

ESET-NOD32      15378/20170508      found a variant of Win32/Poebot.NCA

Endgame 0.4.2/20170503  found malicious (high confidence)

F-Prot  4.7.1.166/20170508      found nothing

F-Secure       11.0.19100.45/20170508      found DeepScan:Generic.Sdbot.7216E179

Fortinet       5.4.233.0/20170508      found W32/Injector.LSH!tr
```

```
GData   A:25.12280B:25.9474/20170508    found DeepScan:Generic.Sdbot.7216E179
Ikarus  0.1.5.2/20170507       found Trojan.Win32.Poebot
Invincea       6.3.0.25213/20170413    found virus.win32.ramnit.a
Jiangmin       16.0.100/20170508      found Worm.Generic.cvo
K7AntiVirus     10.10.23255/20170508    found Backdoor ( 000353841 )
K7GW    10.10.23254/20170508    found Backdoor ( 000353841 )
Kaspersky      15.0.1.13/20170508     found HEUR:Trojan.Win32.Generic
Kingsoft       2013.8.14.323/20170508  found nothing
Malwarebytes    2.1.1.1115/20170508     found nothing
McAfee  6.0.6.653/20170508      found GenericRXBF-DZ!AB30CB38EFE6
McAfee-GW-Edition      v2015/20170507  found BehavesLike.Win32.PWSZbot.lc
MicroWorld-eScan       12.0.250.0/20170508     found DeepScan:Generic.Sdbot.7216E179
Microsoft      1.1.13701.0/20170508    found Trojan:Win32/Dynamer!ac
NANO-Antivirus  1.0.76.16894/20170507   found Trojan.Win32.Poebot.enibzi
Paloalto       1.0/20170508    found generic.ml
Panda   4.6.4.2/20170507       found Trj/CI.A
Qihoo-360      1.0.0.1120/20170508     found Win32/Trojan.97a
Rising  28.0.0.1/20170506      found nothing
SUPERAntiSpyware       5.6.0.1032/20170507     found nothing
SentinelOne     1.0.0.154/20170330      found static engine - malicious
Sophos  4.98.0/20170508 found Mal/Generic-S
Symantec       1.3.1.0/20170507       found W32.IRCBot
Tencent 1.0.0.1/20170508       found Win32.Trojan.Generic.Lmua
TheHacker      6.8.0.5.1509/20170508   found nothing
TotalDefense    37.1.62.1/20170508      found nothing
TrendMicro      9.740.0.1012/20170508   found TROJ_FORUCON.BMC
TrendMicro-HouseCall    9.900.0.1004/20170508   found WORM_SDBOT.SMA
VBA32   3.12.26.4/20170506      found SScope.Injector.MY
VIPRE   57926/20170508  found Trojan.Win32.Generic!BT
ViRobot 2014.3.20.0/20170508    found nothing
Webroot 1.0.0.207/20170508      found W32.Bot.Gen
Yandex  5.5.1.3/20170504       found Trojan.Poebot!txy+Wy80ZI4
Zillya  2.0.0.3273/20170505     found Trojan.Poebot.Win32.128
```

ZoneAlarm 1.0/20170508 found HEUR:Trojan.Win32.Generic

Zoner 1.0/20170508 found nothing

nProtect 2017-05-08.01/20170508 found nothing

Appendix 2

```
[ file data ]
* name..: cbfc90a3a4359950ecdde594a4a0e149
* size..: 74752
* md5...: cbfc90a3a4359950ecdde594a4a0e149
* sha1..: 01bc83b9b1db799215c46b6b97f4e8cb0f62240a
[ scan result ]
ALYac   1.0.1.9/20170508       found DeepScan:Generic.Sdbot.E60AF8E8
AVG     16.0.0.4776/20170508    found Worm/Agobot.JJL
AVware  1.5.0.42/20170508      found Trojan.Win32.Generic!BT
Ad-Aware       3.0.3.1010/20170508     found DeepScan:Generic.Sdbot.E60AF8E8
AegisLab       4.2/20170508    found Troj.W32.Generic!c
AhnLab-V3      3.9.0.17440/20170508    found Trojan/Win32.Generic.C1887479
Antiy-AVL      1.0.0.1/20170508       found Worm/Win32.AGeneric
Arcabit 1.0.0.802/20170508      found DeepScan:Generic.Sdbot.E60AF8E8
Avast   8.0.1489.320/20170508      found Win32:Malware-gen
Avira   8.3.3.4/20170507       found TR/Dropper.Gen
Baidu   1.0.0.2/20170503       found Win32.Trojan.WisdomEyes.16070401.9500.9954
BitDefender     7.2/20170508    found DeepScan:Generic.Sdbot.E60AF8E8
Bkav    1.3.0.8876/20170506     found W32.DustonerZSC.Trojan
CAT-QuickHeal   14.00/20170508  found Trojan.Generic
CMC     1.1.0.977/20170507     found nothing
ClamAV  0.99.2.0/20170508      found nothing
Comodo  27052/20170508  found UnclassifiedMalware
CrowdStrike     1.0/20170130    found malicious_confidence_100% (W)
Cyren   5.4.30.7/20170508      found nothing
DrWeb   7.0.28.2020/20170508      found BackDoor.IRC.Sdbot.34130
ESET-NOD32     15378/20170508      found a variant of Win32/Poebot.NCA
Endgame 0.4.2/20170503  found malicious (high confidence)
F-Prot  4.7.1.166/20170508     found nothing
F-Secure       11.0.19100.45/20170508      found DeepScan:Generic.Sdbot.E60AF8E8
Fortinet       5.4.233.0/20170508     found W32/Injector.LSH!tr
```

```
GData   A:25.12280B:25.9474/20170508    found DeepScan:Generic.Sdbot.E60AF8E8

Ikarus  0.1.5.2/20170507       found Trojan.Win32.Poebot

Invincea       6.3.0.25213/20170413    found virus.win32.virut.bn

Jiangmin       16.0.100/20170508      found Worm.Generic.cuy

K7AntiVirus     10.10.23255/20170508    found Backdoor ( 000353841 )

K7GW    10.10.23254/20170508    found Backdoor ( 000353841 )

Kaspersky      15.0.1.13/20170508      found HEUR:Trojan.Win32.Generic

Kingsoft       2013.8.14.323/20170508  found nothing

Malwarebytes    2.1.1.1115/20170508     found Backdoor.Bot

McAfee  6.0.6.653/20170508      found Generic.ays

McAfee-GW-Edition      v2015/20170507  found BehavesLike.Win32.PWSZbot.lc

MicroWorld-eScan       12.0.250.0/20170508     found DeepScan:Generic.Sdbot.E60AF8E8

Microsoft      1.1.13701.0/20170508    found Trojan:Win32/Dynamer!ac

NANO-Antivirus  1.0.76.16894/20170507   found Trojan.Win32.Poebot.ebvzna

Paloalto       1.0/20170508    found generic.ml

Panda   4.6.4.2/20170507       found Trj/CI.A

Qihoo-360      1.0.0.1120/20170508     found Win32/Trojan.97a

Rising  28.0.0.1/20170506      found nothing

SUPERAntiSpyware       5.6.0.1032/20170507     found Backdoor.Bot/Variant

SentinelOne     1.0.0.154/20170330      found static engine - malicious

Sophos  4.98.0/20170508 found Mal/Generic-S

Symantec       1.3.1.0/20170507       found W32.IRCBot

Tencent 1.0.0.1/20170508       found Win32.Trojan.Generic.Tdfw

TheHacker      6.8.0.5.1509/20170508   found nothing

TotalDefense    37.1.62.1/20170508      found nothing

TrendMicro      9.740.0.1012/20170508   found WORM_SDBOT.SMA

TrendMicro-HouseCall    9.900.0.1004/20170508   found WORM_SDBOT.SMA

VBA32   3.12.26.4/20170506      found SScope.Injector.MY

VIPRE   57926/20170508  found Trojan.Win32.Generic!BT

ViRobot 2014.3.20.0/20170508    found Trojan.Win32.Z.Sdbot.74752[h]

Webroot 1.0.0.207/20170508      found nothing

Yandex  5.5.1.3/20170504       found Trojan.Poebot!DGbT8/2vmho
```

```
Zillya  2.0.0.3273/20170505     found Trojan.Poebot.Win32.126
```

Appendix 3

```
[ file data ]

* name..: 80547f054bdf134c69b11fd0ee9339dd

* size..: 74752

* md5...: 80547f054bdf134c69b11fd0ee9339dd

* sha1..: 5486b313569c7a18763925c087cc3de071bdea5e


[ scan result ]

ALYac   1.0.1.9/20170508       found DeepScan:Generic.Sdbot.B4AE9E03

AVG     16.0.0.4776/20170508    found Worm/Agobot.JJL

AVware  1.5.0.42/20170508      found Trojan.Win32.Generic!BT

Ad-Aware       3.0.3.1010/20170508     found DeepScan:Generic.Sdbot.B4AE9E03

AegisLab       4.2/20170508    found Troj.W32.Generic!c

AhnLab-V3      3.9.0.17440/20170508    found Trojan/Win32.Generic.C1898246

Antiy-AVL      1.0.0.1/20170508       found Worm/Win32.AGeneric

Arcabit 1.0.0.802/20170508      found DeepScan:Generic.Sdbot.B4AE9E03

Avast   8.0.1489.320/20170508       found Win32:Malware-gen

Avira   8.3.3.4/20170507       found TR/Dropper.Gen

Baidu   1.0.0.2/20170503       found Win32.Trojan.WisdomEyes.16070401.9500.9955

BitDefender     7.2/20170508    found DeepScan:Generic.Sdbot.B4AE9E03

Bkav    1.3.0.8876/20170506     found W32.DustonerZSC.Trojan

CAT-QuickHeal   14.00/20170508  found Trojan.Generic

CMC     1.1.0.977/20170507      found nothing

ClamAV  0.99.2.0/20170508       found nothing

Comodo  27052/20170508  found UnclassifiedMalware

CrowdStrike     1.0/20170130    found malicious_confidence_100% (W)

Cyren   5.4.30.7/20170508       found W32/Trojan.ZBSI-1633

DrWeb   7.0.28.2020/20170508       found BackDoor.IRC.Sdbot.34130

ESET-NOD32      15378/20170508       found a variant of Win32/Poebot.NCA

Endgame 0.4.2/20170503  found malicious (high confidence)

F-Prot  4.7.1.166/20170508      found nothing
```

```
F-Secure	11.0.19100.45/20170508	found DeepScan:Generic.Sdbot.B4AE9E03
Fortinet	5.4.233.0/20170508	found W32/Injector.LSH!tr
GData	A:25.12280B:25.9474/20170508	found DeepScan:Generic.Sdbot.B4AE9E03
Ikarus	0.1.5.2/20170507	found Trojan.Win32.Poebot
Invincea	6.3.0.25213/20170413	found virus.win32.virut.bn
Jiangmin	16.0.100/20170508	found Worm.Generic.cvb
K7AntiVirus	10.10.23255/20170508	found Backdoor ( 000353841 )
K7GW	10.10.23254/20170508	found Backdoor ( 000353841 )
Kaspersky	15.0.1.13/20170508	found HEUR:Trojan.Win32.Generic
Kingsoft	2013.8.14.323/20170508	found nothing
Malwarebytes	2.1.1.1115/20170508	found Backdoor.Bot
McAfee	6.0.6.653/20170508	found RDN/Sdbot.worm
McAfee-GW-Edition	v2015/20170507	found BehavesLike.Win32.PWSZbot.lc
MicroWorld-eScan	12.0.250.0/20170508	found DeepScan:Generic.Sdbot.B4AE9E03
Microsoft	1.1.13701.0/20170508	found Trojan:Win32/Dynamer!ac
NANO-Antivirus	1.0.76.16894/20170507	found Trojan.Win32.Poebot.ebvzna
Paloalto	1.0/20170508	found generic.ml
Panda	4.6.4.2/20170507	found Trj/CI.A
Qihoo-360	1.0.0.1120/20170508	found Win32/Trojan.97a
Rising	28.0.0.1/20170506	found nothing
SUPERAntiSpyware	5.6.0.1032/20170507	found Backdoor.Bot/Variant
SentinelOne	1.0.0.154/20170330	found static engine - malicious
Sophos	4.98.0/20170508 found Mal/Generic-S
Symantec	1.3.1.0/20170507	found W32.IRCBot
Tencent	1.0.0.1/20170508	found Win32.Trojan.Generic.Dxmi
TheHacker	6.8.0.5.1509/20170508	found nothing
TotalDefense	37.1.62.1/20170508	found nothing
TrendMicro	9.740.0.1012/20170508	found TROJ_FORUCON.BMC
TrendMicro-HouseCall	9.900.0.1004/20170508	found WORM_SDBOT.SMA
VBA32	3.12.26.4/20170506	found SScope.Injector.MY
VIPRE	57926/20170508	found Trojan.Win32.Generic!BT
ViRobot	2014.3.20.0/20170508	found Trojan.Win32.Z.Sdbot.74752.A[h]
```

Webroot 1.0.0.207/20170508	found W32.Bot.Gen
Yandex 5.5.1.3/20170504	found BackDoor.Sdbot!
Zillya 2.0.0.3273/20170505	found nothing