Issues in Information Systems
Volume 22, Issue 3, pp. 241-254, 2021

DOI: https://doi.org/10.48009/3_iis_2021_259-272

Honeypots and knowledge for network defense

Ping Wang, *Robert Morris University, wangp@rmu.edu*
Hubert D'Cruze, *University of Maryland, hubert.dcruze@yahoo.com*

Abstract

Computer networks as part of critical infrastructure facilities and assets for most organizations are facing increasing challenges in defending against various and sophisticated cyber threats, intrusions, and attacks. Knowledge is a key factor in cyber defense, and honeypots could be an effective tool for gaining knowledge for cyber defense. This research paper draws upon a cyber defense knowledge model based on the classic of *The Art of War* and focuses on the use of honeypots for network intrusion detection. The cyber defense model highlights the role of knowledge (and the lack of knowledge) of strengths and vulnerabilities of yourself and your opponent in cyber defense. This study illustrates the dynamics of the knowledge and its network security benefits using honeypots in a simulation of detection of intrusions and distributed denial of service (DDoS) attacks on a virtual network.

Keywords: cyber defense, network defense, vulnerabilities, knowledge, intrusion detection, DDoS

Introduction

Organizations and individuals are paying more and more attention to cyber defense as cyber threats and attacks have been on the rise. According to a recent data breach investigation report, denial of service and system intrusions are among the major attacks targeting enterprise networks (Verizon, 2021). Meanwhile, cyber attack methods are becoming more sophisticated. For example, traditional denial of service attacks have become more distributed and are launched from massive botnets, and system intrusions involve increasing leverage of malware and deployment of ransomware. Unfortunately, we often do not know the attacks until after they happen and damage is done.

The statement that knowledge is power attributed to Francis Bacon during the English Renaissance still rings true in modern cyber defense. Fast-growing and more and more sophisticated cyber threats and attacks make it increasingly challenging for modern organizations to defend their computer networks, cyber space, and critical assets due to a lack of knowledge of the vulnerabilities and threats (Booker & Musman, 2019; Wang & D'Cruze, 2020). As a result of such lack of knowledge for cyber defense, cybersecurity analysts and responders in practice have to depend on inadequate and questionable information for decision making and response as there is too much "diverse and noisy" data for them to assess the cyber threat adequately (Booker & Musman, 2019). In contrast, organizations equipped with the best knowledge of cyber threats will be in the best position to defend against increasing cyberattacks (Booz Allen Hamilton, 2019). Thus, knowledge of cybersecurity vulnerabilities and threats is a critical factor in defending against cyber attacks.

Knowledge is relative to ignorance or the lack of knowledge, and knowledge or the lack of knowledge may involve yourself and your adversary. Knowledge for cyber defense includes knowing the strengths and weaknesses of your opponent and yourself. Knowing the adversary and oneself is a highly important strategic and tactical concept in *The Art of War*, a classic of military strategies and tactics by Sun Tzu in the 5th century B.C. For example, it is found that knowledge of the common techniques and tools of ransomware used by attackers can be a powerful defense against ransomware attacks (Dunn, 2019). On the contrary, lack of knowledge may cause and add to difficulties for sound decision making under risk and uncertain conditions (M'manga et al., 2019; Wang, 2013). Therefore, vulnerability assessment and penetration testing are voluntarily used by many organizations for gaining knowledge and discovering and addressing security vulnerabilities and risks.

Honeypots can be an effective cyber defense mechanism to lure and monitor attackers and gather valuable knowledge about system intruders for better defense (Kelly, Pitropakis, Mylonas, McKeown, & Buchanan, 2021; Paliwal, 2017; Priyanka, 2018). Unlike intrusion detection systems (IDS), honeypots willingly provide intruders access to restricted systems with the intention to monitor their activities and analyze their goals. This study will adopt the cyber defense knowledge model from a recent study and focus on the knowledge discovery process in detecting and monitoring network intrusions and DDoS attacks using a honeypot simulation on a virtual network. The goal of the study is to reveal and illustrate how the deceptive honeypot strategy and tactics affect the knowledge dynamics between the defender and offender. The study also contributes a network simulation method and data for network intrusion detection and penetration testing. The following sections will review the relevant theoretical background, explain the adopted cyber defense knowledge model, describe the simulation method, and report and discuss the findings from the simulation and tests.

Background

There are valuable lessons of wisdom from classics on ancient warfare that can benefit modern cyber warfare and cyber defense. Knowledge is a strategic factor critical to the outcome of a warfare in *The Art of War.* The best-known statement in *The Art of War* on the role of knowledge is "Know the enemy and know yourself, and you can fight a hundred battles with no danger of defeat" (Michaelson & Michaelson, 2003). The concept of knowledge in *The Art of War* includes the following three types of awareness and assessment of the strengths and weaknesses of yourself and the enemy:

- If you know the enemy and know yourself, you need not fear the result of a hundred battles.
- If you know yourself but not the enemy, for every victory gained you will also suffer a defeat.
- If you know neither the enemy nor yourself, you will succumb in every battle.
  (Sun, trans. 1910, p.45)

Knowing the strengths and weaknesses of both yourself and the adversary is the best scenario which makes you win every battle. The concepts of knowledge and lack of knowledge in *The Art of War* apply to the domain of modern cyber warfare and cyber defense as well. Having knowledge of the cyber warfare capabilities of both yourself and the adversary is essential to victory in the cyber domain (Wilson, 2018). For example, the MITRE Corporation provides the following useful community knowledge resources on known cyber attacks and vulnerabilities:

- The Common Attack Pattern Enumeration and Classification (CAPEC) that lists and categorizes cyber attack patterns and methods used for exploiting hardware and software vulnerabilities;
- The Common Vulnerabilities and Exposures (CVE) that lists the CVE character, identifier, and description of known cybersecurity vulnerabilities (MITRE, 2020).

The cyber attack scenario modelling proposed by Ahn, Kim, and Lee (2020) also includes identifiers, descriptions, and attributes of attacks and vulnerabilities. The Goal and Effect modelling method provides network simulation and analysis of the goals and damage effects of various cyber attacks, including social engineering, reconnaissance, privilege escalation, forgery, denial of service, command and control, exfiltration, destroy device, spreading, resource consumption, and unknown attacks (Ahn, Kim, & Lee, 2020).

The concept of knowledge for cyber defense is dynamic as knowledge and lack of knowledge (or ignorance) are relative to each other. Your knowledge will increase if your opponent's knowledge decreases and ignorance grows, and obscuring or hiding your strengths to the adversary serves to drive the knowledge dynamic to your advantage (Wang & D'Cruze, 2020). In addition, the strategy of deception is emphasized in the following quotes from *The Art of War* so as to disguise yourself and mislead your opponent to decrease and minimize your opponent's true knowledge of your strengths and weaknesses:

> 18. All warfare is based on deception.
>
> 19. Hence, when able to attack, we must seem unable; when using our forces, we must seem inactive; when we are near, we must make the enemy believe we are far away; when far away, we must make him believe we are near.
>
> 20. Hold out baits to entice the enemy. Feign disorder, and crush him.
>
> 21. … Pretend to be weak, that he may grow arrogant.
>
> (Sun, trans. 1910, p.62).

The use of honeypots exemplifies the strategy of deception in the cyber defense domain. Honeypots are intentional deceptions in nature to lure, trap, and monitor intruders using a fake system as a bait. Examples of deception strategies and tactics in network defense include broadcasting of fake SSIDs for easy WiFi access or intentional exposure of deceptive passwords or CAPTCHA mechanisms to allow attackers access to deceptive services (Faveri, Moreira, & Amaral, 2018). Deployment of deceptive strategies and tactics in cyber defense helps to strengthen your cyber defense by misleading your adversaries and minimizing their true knowledge of your systems, assets, strengths, and vulnerabilities (Wang & D'Cruze, 2020).

A honeypot is a "decoy" system to attract intruders and monitor and learn from their activities and tactics in order to examine the network and system vulnerabilities and improve the system security (Paliwal, 2017). Unlike intrusion detection systems (IDS) that reply on known attack signatures, a honeypot is an active network defense system that can monitor and record unknown attacks to detect, prevent and react to active intrusions without disturbing the asset of value (Priyanka, 2018). Honeypots consist of research and production honeypots with different purposes: (1) A research honeypot is to monitor the activities of malicious intruders to gather data for analysis of the tactics, techniques, and procedures of threat actors; (2) Production honeypots are more specifically configured and deployed to emulate current production systems or infrastructure and protect them from current or potential attacks (Kelly, Pitropakis, Mylonas, McKeown,

& Buchanan, 2021). Honeypots may offer different levels of interaction between the intruders and the target system: low, high, and medium:

- A low interaction honeypot emulates minimum system services and features to lure the intruders and log their activities but has very limited responses and system protection;
- A high interaction honeypot offers the most realistic and attractive target with real services and operating systems most similar to the real production infrastructure and provides most accurate data about intruders and their activities;
- A medium interaction honeypot combines the strengths of the low interaction and high interaction honeypots and usually runs on a virtual application layer with more interaction and data collection than low interaction honeypots but less emulation and system exposure risks than high interaction honeypots (Kelly, Pitropakis, Mylonas, McKeown, & Buchanan, 2021).

Cyber defense knowledge model

This study adopts the knowledge model for cyber defense proposed by Wang and D'Cruze (2020) based on the defense knowledge principles laid out in *The Art of War*. The proposed model consists of a Knowledge and Goals Matrix and a Knowledge Discovery Process. Table 1 below shows the Knowledge and Goals Matrix.

The Knowledge and Goals Matrix depicts the relationships of knowledge and goals between yourself and your opponent. In cyber defense, knowledge and goals of yourself mean the following:

- Discover and know the vulnerabilities of your own systems as well as the measures and actions to take to mitigate and minimize the vulnerabilities and risks for yourself;
- Know how to hide your critical assets and vulnerabilities from the opponent for the goal of minimizing your opponent's knowledge of your assets and weaknesses;
- Know how to set up attractive fake targets or decoys such as a honeypot with intentional vulnerabilities to lure intruders for the goal of deceiving and misleading your opponents (Wang & D'Cruze, 2020).

Knowledge of your opponent(s) means the following:

- Discover and know your opponent's strengths adequately to guide the improvement of your defense and counter-attack strategies;
- Discover and know your opponent's critical assets and vulnerabilities to be well prepared for necessary exploitation of them (Wang & D'Cruze, 2020).

Table 1: Knowledge and Goals Matrix (Wang & D'Cruze, 2020)

	Knowledge	Goals
Yourself	● Know your own vulnerabilities ● Know how to mitigate your own vulnerabilities ● Know how to hide your assets and vulnerabilities from your opponent ● Know how to set up fake vulnerabilities	● To minimize your vulnerabilities ● To assess and manage your vulnerabilities and risks ● To minimize opponent's knowledge of your vulnerabilities ● To mislead, misinform, distract, and deceive your opponent
Opponent	● Know your opponent's strengths ● Know your opponent's assets and vulnerabilities ● Know how to discover your opponent's vulnerabilities	● To be aware of threats and avoid striking the strong spots of your opponent ● To exploit the vulnerabilities of your opponent ● To maximize your knowledge of your opponent

The other component of the cyber defense model is the knowledge discovery process for gaining the necessary knowledge of yourself and your opponent and reaching your cyber defense goals. The knowledge discovery process includes the steps of footprinting, port scanning, enumeration, penetration testing, and vulnerability analysis. Footprinting is the initial reconnaissance step used by black-hat, white-hat, and gray-hat hackers for passive information gathering about a target using various tools and techniques (Hassan & Hijazi, 2018; Wang & D'Cruze, 2020). Port scanning is used to identify open ports and services available on a network host such as a server to seek vulnerabilities for exploitation. Enumeration is more aggressive information gathering through active connections to extract more detailed information such as usernames and services from a system. Penetration testing is more aggressive knowledge discovery to test and determine the vulnerabilities of the target system and their exploitability. Vulnerability analysis is the final step of the knowledge discovery process to analyze and assess the potential impact of the discovered vulnerabilities and their relevant threats and risks in order to properly manage the cyber risks (Muckin & Fitch, 2019; Wang & D'Cruze, 2020).

Methodology

This study uses a virtual network simulation of intrusion detection via a honeypot to test the knowledge model for cyber defense. The virtual network for simulation includes two virtual machines running Kali Linux and another virtual machine running Windows 10 on the VirtualBox platform, a free cross-platform virtualization application provided by Oracle. VirtualBox is widely used for simulation, testing, and disaster recovery as it allows easy switch to the saved snapshots of a previous virtual machine state if necessary. Kali Linux is an enterprise-ready security auditing Linux distribution bundled with many tools for security knowledge discovery and analysis; and it primarily serves cybersecurity professionals and IT administrators for advanced reconnaissance and penetration testing, forensic analysis, and security auditing (Wang & D'Cruze, 2020).

On the simulation network, three virtual machines are set up to generate and detect unauthorized network intrusions and DDoS attacks and discover the attack sources and their IP addresses via a honeypot. An Apache web server, a firewall, and a PentBox honeypot are installed on one of the Kali virtual machines at the IP address of 10.0.0.102. The other Kali virtual machine at the IP of 10.0.0.101 is used to test if the web server target at 10.0.0.102 is working properly to lure and monitor the intruders.

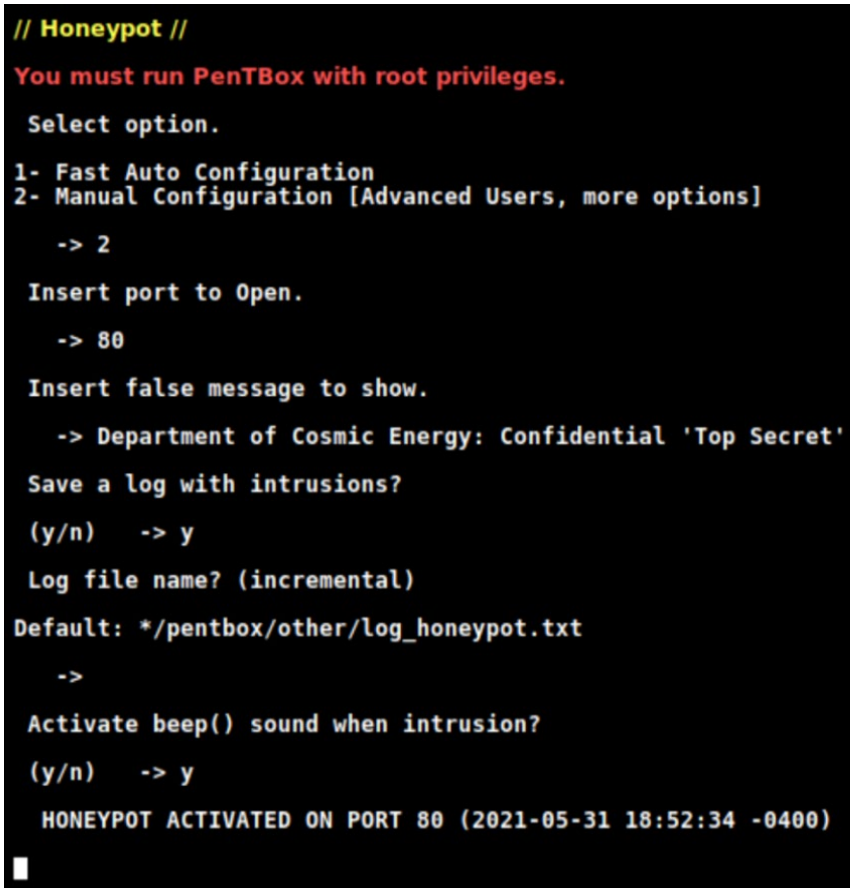
Intrusions and DDoS attacks are launched from the Windows 10 virtual machine at the IP address of 10.0.0.103 using the Low Orbit Cannon (LOIC) application. LOIC is a popular open-source network stress testing and denial-of-service attack application written in C#. LOIC is able to send multiple simultaneous requests to flood the target web server with TCP or UDP packets to disrupt the normal service of the target host (Asri & Pranggono, 2015; Murugan, Ganesan, & Thiyagu, 2018). LOIC has also been effectively tested as a denial-of-service attack tool for exploiting the MITRE's CVE vulnerabilities (Cheepborisuttikul & Teng-Amnuay, 2013).

The honeypot software tool used in the simulation for deception and detection of network intruders is PentBox installed on the 10.0.0.102 virtual machine. The PentBox honeypot is a security suite that can be used for penetration testing to perform various operations. The PentBox kit allows you to listen for connections for intrusion detection and contains multiple tools to perform and monitor network attack activities, including cracking hashes, stress testing, DNS enumeration, etc., and is able provide valuable information about the attacker (Murugan, Ganesan, & Thiyagu, 2018; Priyanka, 2018). The Wireshark sniffer is also used on this host to capture network data for further analysis.

The following section reports and discusses the implementation and findings of the simulation of the honeypot detection and monitoring of network intrusions and DDoS attacks to illustrate the knowledge discovery model for cyber defense.

Findings and discussions

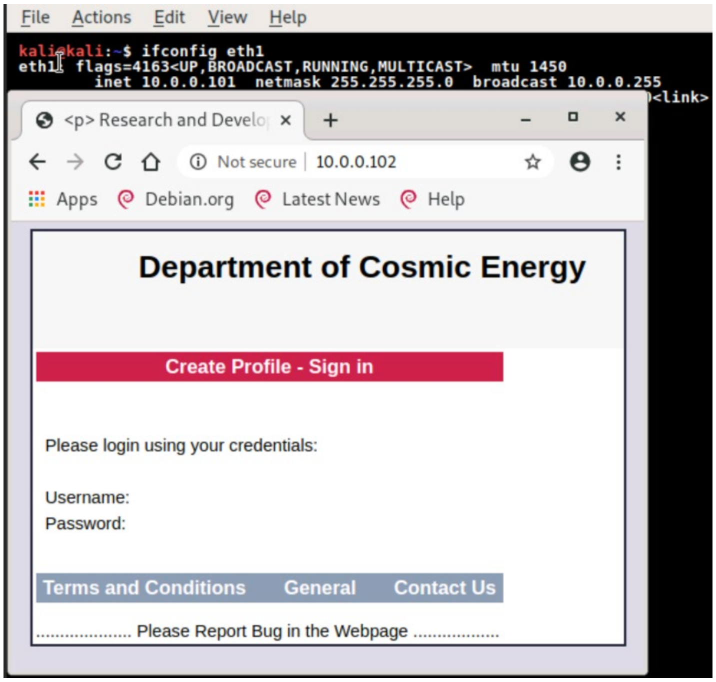
The honeypot deployment is the key to the network simulation. The PentBox version 1.8 honeypot is activated and configured on the Kali virtual machine at the IP of 10.0.0.102 to monitor the fake web service "Department of Cosmic Energy" on port 80 with logging enabled. Figure 1 below shows the honeypot activation and configuration.



```
// Honeypot //
You must run PenTBox with root privileges.
 Select option.
1- Fast Auto Configuration
2- Manual Configuration [Advanced Users, more options]
   -> 2
 Insert port to Open.
   -> 80
 Insert false message to show.
   -> Department of Cosmic Energy: Confidential 'Top Secret'
 Save a log with intrusions?
 (y/n)   -> y
 Log file name? (incremental)
Default: */pentbox/other/log_honeypot.txt
   ->
 Activate beep() sound when intrusion?
 (y/n)   -> y
  HONEYPOT ACTIVATED ON PORT 80 (2021-05-31 18:52:34 -0400)
```

Figure 1: PentBox honeypot activated and configured on port 80

Figure 2 below shows the fake website “Department of Cosmic Energy” running unsecured on port 80 as the bait for the honeypot on the Kali virtual machine of 10.0.0.102. A firewall is also deployed on the same virtual machine hosting the fake web service and the honeypot as a control variable to test the effectiveness of the honeypot. GUFW (Graphical Uncomplicated Firewall) is the firewall used for this simulation. GUFW is a graphical utility for managing the Uncomplicated Firewall (UFW), which uses iptables rules available on major Linux distributions.



```
kali@kali:~$ ifconfig eth1
eth1  flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1450
        inet 10.0.0.101  netmask 255.255.255.0  broadcast 10.0.0.255
                                                            )<link>
```

Department of Cosmic Energy

Create Profile - Sign in

Please login using your credentials:

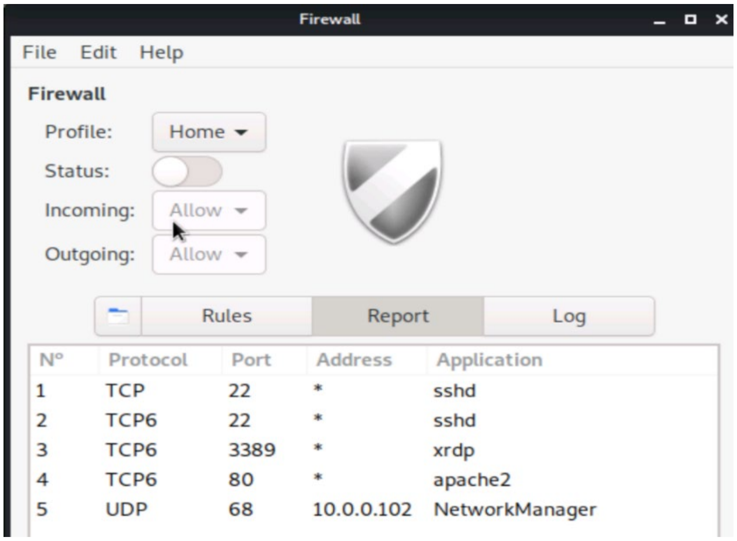
Username:
Password:

Terms and Conditions General Contact Us

.... Please Report Bug in the Webpage ....

Figure 2: Fake website running as the bait for the honeypot

The control test shows when the GUFW firewall is configured to "Deny" all incoming and outgoing packets, the honeypot detects no intrusion activity as all packets are already blocked by the firewall. Therefore, the firewall should be and is set to "Allow" all traffic in all directions to make the "bait" more attractive to intruders as intended for the honeypot to work. Figure 3 below shows the firewall setting.



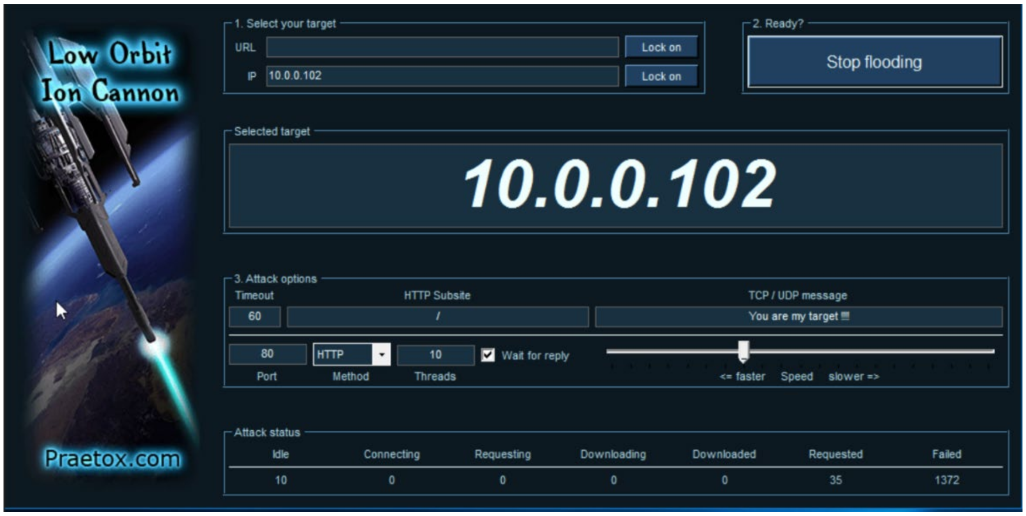
Firewall

Profile: Home
Status:
Incoming: Allow
Outgoing: Allow

Nº	Protocol	Port	Address	Application
1	TCP	22	*	sshd
2	TCP6	22	*	sshd
3	TCP6	3389	*	xrdp
4	TCP6	80	*	apache2
5	UDP	68	10.0.0.102	NetworkManager

Figure 3: GUFW firewall settings

For this experiment, the Low Orbit Ion Cannon (LOIC) application is used for launching DDoS attacks from the Windows virtual machine at the IP of 10.0.0.103. LOIC is set to attack the target web server on the virtual machine at 10.0.0.102. The DDoS attacks launched from LOIC generates enough traffic to the target server with TCP, UDP and HTTP packets to disrupt the web service. Figure 4 below shows the DDoS attacks in action launched from the LOIC application.



1. Select your target
URL
IP 10.0.0.102

2. Ready?

Selected target
10.0.0.102

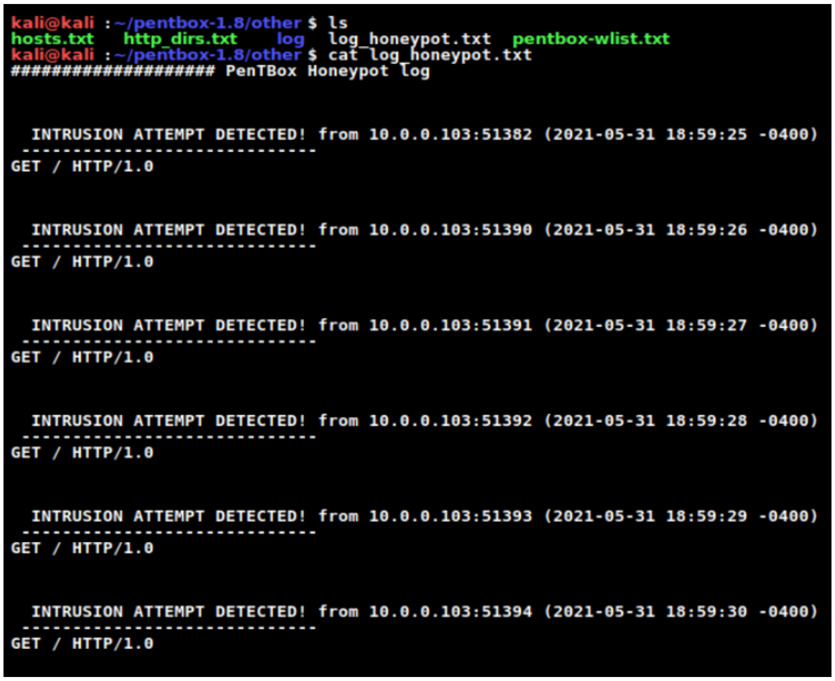
3. Attack options
Timeout 60
HTTP Subsite /
TCP / UDP message You are my target !!!
Port 80
Method HTTP
Threads 10
☑ Wait for reply

Attack status

Idle	Connecting	Requesting	Downloading	Downloaded	Requested	Failed
10	0	0	0	0	35	1372

Figure 4: DDoS attacks launched from LOIC

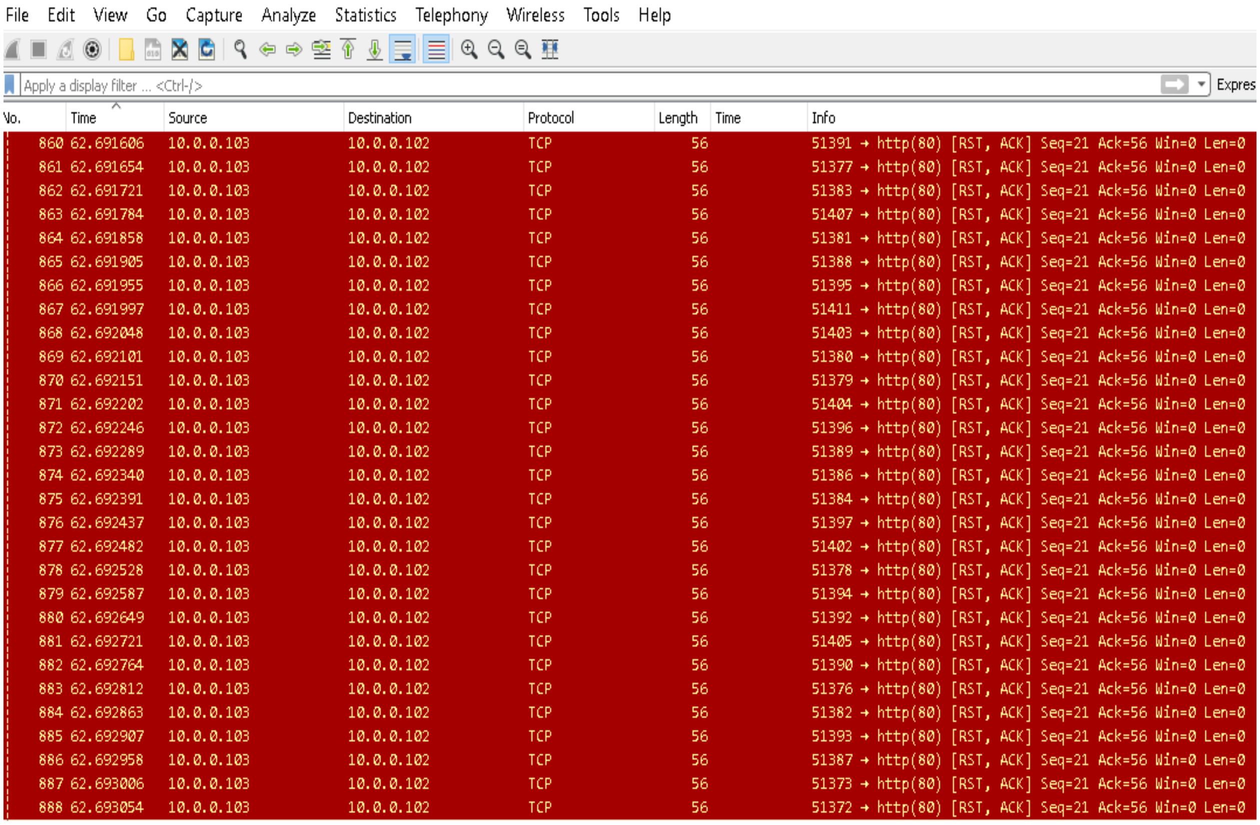
LOIC was only running for a few seconds for this simulation, and PentBox was able to detect hundreds of intrusion attempts including over a hundred HTTP requests to the web service. The honeypot detections of the intrusions were recorded in the log file. Figure 5 below is a partial display of the log file.



```
kali@kali :~/pentbox-1.8/other $ ls
hosts.txt   http_dirs.txt    log   log_honeypot.txt  pentbox-wlist.txt
kali@kali :~/pentbox-1.8/other $ cat log_honeypot.txt
#################### PenTBox Honeypot log


  INTRUSION ATTEMPT DETECTED! from 10.0.0.103:51382 (2021-05-31 18:59:25 -0400)
 -----------------------------
GET / HTTP/1.0


  INTRUSION ATTEMPT DETECTED! from 10.0.0.103:51390 (2021-05-31 18:59:26 -0400)
 -----------------------------
GET / HTTP/1.0


  INTRUSION ATTEMPT DETECTED! from 10.0.0.103:51391 (2021-05-31 18:59:27 -0400)
 -----------------------------
GET / HTTP/1.0


  INTRUSION ATTEMPT DETECTED! from 10.0.0.103:51392 (2021-05-31 18:59:28 -0400)
 -----------------------------
GET / HTTP/1.0


  INTRUSION ATTEMPT DETECTED! from 10.0.0.103:51393 (2021-05-31 18:59:29 -0400)
 -----------------------------
GET / HTTP/1.0


  INTRUSION ATTEMPT DETECTED! from 10.0.0.103:51394 (2021-05-31 18:59:30 -0400)
 -----------------------------
GET / HTTP/1.0
```

Figure 5: Intrusion detections recorded in the honeypot log file

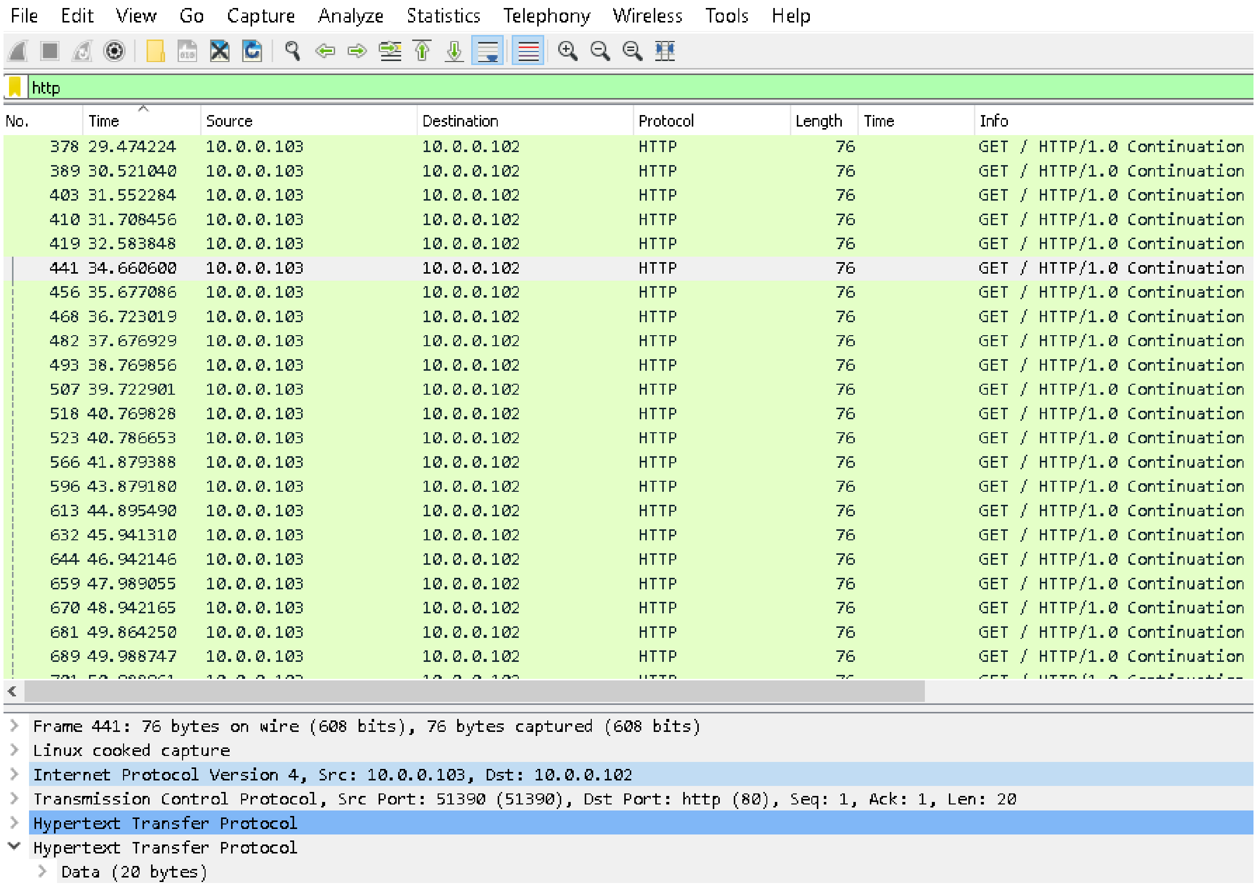
The PentBox honeypot provides useful knowledge about the attacker that will benefit subsequent security analysis and defense of critical assets. It not only detected and monitored the intrusions but also recorded valuable information about the flooding DDoS attacks from the source IP address 10.0.0.103 and the ephemeral ports where the attacks were generated. It also recorded the date and times of the attacks. The honeypot used for this simulation is of low interaction for research purpose without impacting any production network or system. The DDoS attacks consist of TCP, UDP, and HTTP packets. The Wireshark network sniffer was run during the simulation to capture the packets attracted by the honeypot for further analysis. Figure 6 below shows a flagged part of a Wireshark capture of the TCP requests from the attack machine at 10.0.0.103 to the target machine at 10.0.0.102. The capture shows continuous HTTP requests to the web server at port 80, which is typical signature of a flooding DDoS attack. The HTTP packets sent from the attack machine are nearly parallel with repeated resets and persistent requests to reach and flood the target web server.



No.	Time	Source	Destination	Protocol	Length	Time	Info
860	62.691606	10.0.0.103	10.0.0.102	TCP	56		51391 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
861	62.691654	10.0.0.103	10.0.0.102	TCP	56		51377 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
862	62.691721	10.0.0.103	10.0.0.102	TCP	56		51383 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
863	62.691784	10.0.0.103	10.0.0.102	TCP	56		51407 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
864	62.691858	10.0.0.103	10.0.0.102	TCP	56		51381 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
865	62.691905	10.0.0.103	10.0.0.102	TCP	56		51388 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
866	62.691955	10.0.0.103	10.0.0.102	TCP	56		51395 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
867	62.691997	10.0.0.103	10.0.0.102	TCP	56		51411 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
868	62.692048	10.0.0.103	10.0.0.102	TCP	56		51403 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
869	62.692101	10.0.0.103	10.0.0.102	TCP	56		51380 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
870	62.692151	10.0.0.103	10.0.0.102	TCP	56		51379 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
871	62.692202	10.0.0.103	10.0.0.102	TCP	56		51404 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
872	62.692246	10.0.0.103	10.0.0.102	TCP	56		51396 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
873	62.692289	10.0.0.103	10.0.0.102	TCP	56		51389 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
874	62.692340	10.0.0.103	10.0.0.102	TCP	56		51386 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
875	62.692391	10.0.0.103	10.0.0.102	TCP	56		51384 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
876	62.692437	10.0.0.103	10.0.0.102	TCP	56		51397 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
877	62.692482	10.0.0.103	10.0.0.102	TCP	56		51402 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
878	62.692528	10.0.0.103	10.0.0.102	TCP	56		51378 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
879	62.692587	10.0.0.103	10.0.0.102	TCP	56		51394 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
880	62.692649	10.0.0.103	10.0.0.102	TCP	56		51392 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
881	62.692721	10.0.0.103	10.0.0.102	TCP	56		51405 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
882	62.692764	10.0.0.103	10.0.0.102	TCP	56		51390 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
883	62.692812	10.0.0.103	10.0.0.102	TCP	56		51376 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
884	62.692863	10.0.0.103	10.0.0.102	TCP	56		51382 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
885	62.692907	10.0.0.103	10.0.0.102	TCP	56		51393 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
886	62.692958	10.0.0.103	10.0.0.102	TCP	56		51387 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
887	62.693006	10.0.0.103	10.0.0.102	TCP	56		51373 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0
888	62.693054	10.0.0.103	10.0.0.102	TCP	56		51372 → http(80) [RST, ACK] Seq=21 Ack=56 Win=0 Len=0

Figure 6: Wireshark capture of flooding TCP requests

Thanks to the honeypot attraction on the simulation network, a large number of packets including HTTP packets were detected and captured by Wireshark within a few seconds. We can also learn useful knowledge about the intrusions and attacks from the HTTP packets. There was a total of 104 HTTP packets captured by Wireshark in the honeypot simulation. The GET method is used to request data from the target resource. GET is the primary mechanism of information retrieval and the focus of almost all web service performance optimizations. Therefore, GET request packets are often used as a flooding tool to overwhelm the web server in DDoS attacks so that legitimate requests from other users and hosts cannot be handled. Following the TCP stream in Wireshark will also provide useful knowledge about the details of the web requests, such as the source host, timestamps, the payload, metadata in the form of header fields, sequence, and coding. In addition, we can have the opportunity to examine and learn more about the source from the different layers of the HTTP packets captured in Wireshark. Figure 7 below shows some HTTP packets captured and their network layers.



```
http

No.   Time        Source       Destination   Protocol   Length   Time   Info
  378 29.474224   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  389 30.521040   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  403 31.552284   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  410 31.708456   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  419 32.583848   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  441 34.660600   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  456 35.677086   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  468 36.723019   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  482 37.676929   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  493 38.769856   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  507 39.722901   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  518 40.769828   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  523 40.786653   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  566 41.879388   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  596 43.879180   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  613 44.895490   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  632 45.941310   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  644 46.942146   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  659 47.989055   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  670 48.942165   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  681 49.864250   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation
  689 49.988747   10.0.0.103   10.0.0.102    HTTP           76          GET / HTTP/1.0 Continuation

> Frame 441: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 10.0.0.103, Dst: 10.0.0.102
> Transmission Control Protocol, Src Port: 51390 (51390), Dst Port: http (80), Seq: 1, Ack: 1, Len: 20
> Hypertext Transfer Protocol
v Hypertext Transfer Protocol
    > Data (20 bytes)
```

Figure 7: Wireshark capture of HTTP GET requests

Conclusion

This study draws upon the knowledge model for cyber defense based on the knowledge and defense principles in *The Art of War* and focuses on the use of deception via honeypots to mislead and increase the ignorance of your opponents. As knowledge and ignorance are dynamic and relative to each other, one's knowledge for cyber defense strengthens as the opponent's ignorance grows. Honeypots in network defense provide attractive baits to lure intruders into a trap, monitor their activities, and gather useful knowledge about the intruders and their attacks to gain cyber defense knowledge and to distract attackers away from real assets of value. The virtualized simulation of network intrusion and DDoS attack detection via the PentBox honeypot in this study illustrates that valuable knowledge can be gained from detection, logging and packet captures in the honeypot environment. The network defender using the honeypot and the fake web service as a bait in this simulation not only misleads the intruders and attacks to a wrong target away from valuable assets but also gains useful knowledge about the attacker while the attacker remains ignorant. However, the simulation in this study is limited to a simplified low-interaction type honeypot and may not represent more complex honeypots used in large organizations.

It is important to reiterate the central point of this study that this is not a research on how to use a honeypot or how to conduct penetration testing. Instead, we should focus on the big picture here which is to gain knowledge for cyber defense for yourself while deceiving the adversaries and increasing their ignorance.

Creative thinking is essential to the future of cyber defense as cyber threats are dynamic with new and unforeseeable threats and risks emerging from time to time (Wang & D'Cruze, 2020). Cyber defense is not just about the use of tools and techniques but rather dependent upon creative thinking with adequate knowledge of the opponents and risks. Classics on warfare do provide valuable food for thoughts and wisdom for cyber defense regardless of the tools and techniques used at different times. However, this study is limited to a knowledge model based on certain knowledge principles in *The Art of War*. Future and follow-up studies will continue to explore and discover inspirations and wisdom for cyber defense from other and various sources of literature. It is also important to think creatively outside the box and learn the lessons and apply the wisdom from other fields to solving the problems in cybersecurity. So future research in cyber defense should incorporate more interdisciplinary approaches.

References

Ahn, M.K., Kim, Y.H., & Lee, J. (2020). Hierarchical multi-stage cyber attack scenario modeling based on G&E model for cyber risk simulation analysis. *Applied Sciences, 10*(1426), 1-16.

Asri, S., & Pranggono, B. (2015). Impact of distributed denial-of-service attack on advanced metering infrastructure. *Wireless Personal Communications, 83* (3), 2211-2223.

Booker, L.B., & Musman, S.A. (2019). A model-based, decision-theoretic perspective on automated cyber response. Retrieved from https://arxiv.org/abs/2002.08957

Booz Allen Hamilton. (2019). 2020 Cybersecurity Threat Trends Outlook. Retrieved from www.boozallen.com

Cheepborisuttikul, T., & Teng-Amnuay, Y. (2013). Using Low Orbit Ion Cannon for denial of service attack based on CVE. *Proceedings of the Second International Conference on Advances in Information Technology - AIT2013,* 145-149.

Dunn, J.E. (2019). How ransomware attacks. Retrieved from https://nakedsecurity.sophos.com/2019/11/15/how-ransomware-attacks

Faveri, C.D., Moreira, A., & Amaral, V. (2018). Multi-paradigm deception modeling for cyber defense. *The Journal of Systems and Software, 141*(2018), 32-51.

Hassan, N. A., & Hijazi, R. (2018). Technical footprinting. *Open Source Intelligence Methods and Tools,* 313-339. doi:10.1007/978-1-4842-3213-2_8

Kelly, C., Pitropakis, N., Mylonas, A., McKeown, S., & Buchanan, W.J. (2021). A comparative analysis of honeypots on different cloud platforms. *Sensors, 21*(2433), 1-19.

Michaelson, G., & Michaelson, S. (2003). *Sun Tzu for success.* Avon, MA: Adams Media Corporation.

MITRE. (2020). Common Vulnerabilities and Exposures (CVE). Retrieved from http://cve.mitre.org/

M'manga,A., Faily,S., McAlaney, J., Williams, C., Kadobayashi, Y., & Miyamoto, D. (2019). A normative decision-making model for cyber security. *Information & Computer Security, 27*(5), 636-646.

Muckin, M., & Fitch, S.C. (2019). A threat-driven approach to cyber security. Retrieved from https://www.lockheedmartin.com/

Murugan,S.G., Ganesan, T., & Thiyagu, S. (2018). Detecting & isolating anonymous nodes using honeypot in networks. *International Journal of Scientific Research in CS, Eng, & IT, 3*(3), 926-931.

Paliwal, S. (2017). Honeypot: A trap for attackers. *International Journal of Advanced Research in Computer and Communication Engineering, 6*(3), 842-845.

Priyanka, E.K. (2018). Intrusion detection in meta search engine through heuristics: Honeypot. *International Journal of Scientific & Engineering Research, 9*(6), 1654-1660.

Sun, T. (1910). *The art of war* (L. Giles, Trans.). BookYards.com.

Verizon. (2021). DBIR: 2021 Data Breach Investigations Report. Retrieved from https://www.verizon.com/

Wang, P. (2013). Decision under uncertainties of online phishing. In S. Ao & L. Gelman (Eds.), *Electrical engineering and intelligent systems* (pp. 207-218). New York, NY: Springer Science+Business Media, LLC.

Wang, P., & D'Cruze, H. (2020). Lessons on the power of knowledge for cyber defense from Sun Tzu's *the Art of War*. *Issues in Information Systems, 21* (3), 105-116.

Wilson, R. (2018). Sun Tzu and the art of cyberwar. *Defense AT&L, January-February 2018,* 30-34.