ATHABASCA UNIVERSITY

A FEDERATED APPROACH TO IDENTIFYING AND REMOVING ADVANCED PERSISTENT SECURITY THREATS ON ENTERPRISE COMPUTER NETWORKS

BY

WADE W. WESOLOWSKY

A PROJECT PROPOSAL submitted in partial fulfillment

Of the requirements for the degree of

MASTER OF SCIENCE in INFORMATION SYSTEMS

Athabasca, Alberta

January, 2018

ABSTRACT

Computer Security is an intense flash point of concern in the modern computer landscape. Many threats to computer systems both known and unknown prey on the mind of computer professionals. Advanced Persistent Security Threats (APST) are a species of clandestine attack which infiltrate computer systems to exfiltrate data and struggle to maintain an everlasting foothold inside the target network. They are of particular concern thanks to their use and sponsorship by state actors in proxy battles and covert operations. In this landscape we chart the rise of the threat and search for free and open source software which can be used on an enterprise network to decrease the threat. Uncovering the threat will be a collaboration between the security tools cooperating together as components within a Federated Security Module (FSM). A federation between the security tools will offer a secure centralized data source from which their combined knowledge can be accessed. The efficacy of the federation will be tested on a simulated enterprise network to provide us with opportunities for improvement and lessons learned for future research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

DETAILED PROJECT PROPOSAL

1.1 The Problem/Opportunity

Curiosity surrounding ideal ways of protecting Information Technology (IT) assets from security threats is a growing area of study. The interconnected nature of computer systems in the modern world makes this problem even more difficult due to the myriad of ways devices can communicate and exchange data on a network. All the components in a computer network, including the human operators, introduce attack vectors for a determined adversary to exploit. The system administrator and other computer professionals tasked with protecting the network will often take baseline precautions to mitigate these threats. These precautions might take the form of frequent patches, password policies, user education, running anti-virus software, or other protection measures. Often baseline protections are adequate when trying to prevent damage from automated, routine security threats like spam and botnets. However, they often fall short when there is a human intelligence directing the attack - when the attacker takes the time to learn the network and tailor the attacks to the vulnerabilities therein. One exercise often undertaken is to hire a penetration tester to attempt a break-in of the network in order to expose and report on potential vulnerabilities [2]. Often, the attempt is successful at illustrating systemic problems that make the total compromise of the computer system possible. The custodians of the network often ask how this is possible when baseline security precautions were taken and studiously followed.

The inevitable conclusion must be that baseline network security is inadequate to the security challenges present in a modern computer network. Even when a minor system glitch is detected, it can be a warning indicator of a much larger and deeper issue [3]. Without the ability to police these security threats, there is little chance of the enterprise network remaining secure. While many threats exist, we would like to focus on one of the most salient threats which is the Advanced

Persistent Threat (APT). APT can be defined as 'a sophisticated attack composed of several steps and performed by experienced and highly skilled actors with a great determination to achieve their goal' [4, p. 1][5]. These threats are *advanced* due to their sophisticated nature and multi-staged methods used to attack the IT system. They are *persistent* because their foothold within the network is non-volatile and tends to be of a permanent nature. They are a *threat* due to their sneaky, often undetected, exploitation of vulnerabilities within the computer system. Within our terminology we have scoped the definition of APT as Advanced Persistent Security Threat (APST) to highlight our focus on computer security threats versus other threats which maybe be present on computer systems such as coding errors, bit rot, data loss, etcetera.

Since APST are so difficult to mitigate there is ample opportunity to explore available options for their detection and removal. This field is also exploding with the rise of malware and hacking tools created by state actors [6], and we see only expansion within this domain. Some writers even suggest the deployment of systems to monitor your employees to watch for insider threats [7]. There is no lack of innovation, and we hope that our Project will add further ideas to this endeavor.

1.2 The Goal

The research objective is the study of existing tools and methods which can be leveraged to detect and remove APST within enterprise computer networks. After a review of the literature, a federated security model (FSM) design will be proposed and implemented to integrate select tools and techniques together in a federated manner which will help to mitigate threats and vulnerabilities within an enterprise environment. The primary goal of the FSM will be detection of intrusions and vulnerabilities, while the subjugate goal will be removal or elimination of the identified threats or vulnerabilities.

Creating the software to perform the federation of various software security tools will be a key focus of our work. Due to the federation of components, there will be problems of overlapping, non-deterministic tool behaviour, and mismatch between the tools while still having the

requirement to maintain interoperability between separate components [8]. One would imagine that because our module will be active in the security realm that it would act in part like an identity management system [9] to ensure components can trust each other. We will doubtlessly have to use a federated identity management system like Shibboleth to authenticate our components before they pass information to the FSM.

Another goal of our research will be to prototype our FSM on a simulated enterprise network. This will require the setup of a lab environment running similar services and roles that might be present in an enterprise network environment. Then we would run the components of our FSM which will be used for the detection of APST. Since these threats are often directed by a human intelligence, the research will implant several types of malware within the enterprise environment to observe the efficacy of the FSM at finding the threats. An analysis of how the FSM performs during the simulation should suggest additional avenues for future research.

1.3 Impact or Significance of the Problem/Opportunity

There has been a consistent growth of security threats which target computer systems and networks over the years[10]. Each year many new and previously unknown threats are discovered by security researchers who studiously document them along with potential fixes for their exploited vulnerabilities. This happens with such a regular basis that anti-virus software must be updated daily and most software vendors have moved towards software patching models which allow for a consistent delivery of updates and upgrades to their software over the Internet[11].

Specifically in the realm of APST the sophistication of threats has slowly advanced as researchers find and analyze threats on a continual basis. When a new threat is discovered often it is reverse engineered by security researchers[12] in order to get a better idea of how it is constructed. This gives valuable insights into how the APST functions and sometimes even gives information about its source. However, it also allows researchers to learn new techniques of software exploitation which can be used to create new malware.

We would strongly believe that most corporate and government networks have already been compromised in some way and they are simply not aware of it. This is because one of the important features of APST is avoiding detection[13] in order to succeed at maximizing data exfiltration from the compromised network. Without tools and techniques to at least find this threat it will continue to plague the network [14] and most likely allow long term loss of corporate secrets and other confidential data.

A quick browse of WikiLeaks is all that should be required to justify the present research. It is quite clear that software exploitation has become a central focus within the Central Intelligence Agency (CIA), especially in relation to the Vault 7 leaks which have been ongoing [15]. There is an important section regarding how to evade forensics and anti-virus software [16][17][18][19]. This information demonstrates that state actors are well versed in evading detection while exploiting target systems and exfiltrating data. There can be no doubt that such practices are ongoing today.

We have argued that the risks to network security are only growing and shown how one state actor (CIA) may be developing exploitation tools. Within this space there is an urgent need to for better defense techniques which will help to alert security professional to potential risks within their enterprise networks. Furthermore, the arguments presented tend to suggest that this defense should be able to handle unknown threats. The impact of the research could be great depending on the programming skill of researcher, how astute they are to federate tools together, and the novelty of their ideas!

1.4 Potential Causes of the Problem/Opportunity

In this section we explore some of the reasons why APST have become such a large nuisance in enterprise networks along with some of the reasons they are becoming an even more prevalent threat. The reasons explored include the expansion of network medium, the perpetual discovery of vulnerabilities, the ongoing value of data contained within the networks, the specter of cyberwarfare, complexity of enterprise networks, the evolution of countermeasures, and aging IT

infrastructure. All these areas collectively illuminate reasons for the proliferation of APST and highlight some of the problems and opportunities present in that space. We do not claim these potential causes are exhaustive or complete.

1.4.1 Network Medium

Computer networks were historically limited by physical wires which connected the computer systems. However, these days the network media could be cellular or wireless signals. There is less reliance on wired connections and an increasing focus on non-wired devices. This means that any computer system that is able to send and receive is at risk from nefarious signals. This risk can even extend to computer microphones and speakers which can be used to transmit and receive in a local area [20]. In general, an air gap between systems is no longer a deterrent to information leaving your computer [21]. Additionally, these types of attacks are difficult for the user to know about because humans cannot perceive radio frequency signals or hear the ranges that the malware transmits at and thus there is no warning signs that malware is working against them.

1.4.2 Vulnerabilities Everywhere

Each day software vulnerabilities are discovered and exploited. All one has to do is glance at the Bugtraq Mailing List or the Full Disclosure Mailing List to get a sense of how frequently this occurs. The onslaught of such notices gives even the most stoic reader pause, and highlights just how frequently software contains bugs, undocumented features, or other quirks which allow it to be compromised in so many ways. The sheer number of vulnerabilities shows the dangers of using an unpatched system, and even if you are using a patched system it shows that it would be impossible to trust your software to be bug free. The simple fact is that no matter which system you are running someone will have an exploit for it.

As a software consumer, there is even the risk that the software you have purchased contains back doors or other side-channels which allow covert access and surveillance. One long running debate is that your should have access to the source code of products which you run on your

computer and whether that will contribute to the security of the software [22]. Without auditing the source code, it is very difficult to trust that your computer is executing the code you would expect.

Even areas of the computer system which historically have been considered safe are now coming under attack. There are well known techniques for re-flashing the BIOS of a computer system and embedding malware inside it [23]. There are many places for malware to hide, "between Linux and the hardware are at least 2 1/2 kernels" [24]. A more recent scare has broken out regarding the use of the Minix operating system within Intel processors, "Thanks for putting a version of MINIX inside the ME-11 management engine chip used on almost all recent desktop and laptop computers in the world"[25].

1.4.3 The Value of Data

All organizations contain valuable data. This data often has technological and commercial value, which means that other people would like to gain access to this information. APST would be a great way for an actor to gain and maintain access to this data. The type of data which might be stolen can be almost anything from passwords to behavioural profiles.

There is an argument made that malware may be used in a stealing-reality attack [26]. This means that malware may be present in a system specifically to slowly spread and steal data for behavioural analysis. This type of attack is deadly because it can be used to build profiles on specific human targets on patterns of their behaviour that may not be easily changed.

Malware can even be used to gain political [27] advantage over opponents. We can imagine the computer accounts of a powerful people being compromised by an attack in order to track or blackmail these people. It is apparent from the Edward Snowden leaks [28] that this type of surveillance has already been wildly implemented by the National Security Agency (NSA). APST definitely plays a role in these types of attacks, as often computer systems must be compromised over the long term in order for reliable intelligence to be extracted over a long time horizon. Furthermore, often the adversary has an IT environment that is configured in such a way that generic attacks

would be unsuccessful, so custom advanced attacks are necessary.

1.4.4 Cyberwarfare

With nation states taking notice of the importance inherent in Information Technology infrastructures and their ability to subvert these infrastructure for various ends, there has been some dialogue about cyberwarfare. The digital frontier can be thought of as trench warfare in many respects, as it is still new and untested. The best defenses are multi-layered [29], however the visual of many trenches might be more apt. An aerial bombardment which bypasses the trenches, shows how inadequate this method of defense might be. We definitely see that superpowers [29] often blame each other for network intrusions and lack of cooperation in these issues.

While the specter of cyberwarfare is often maligned, the threat of other types of operations, like PsyOps [30] show that gaining control of a system might allow you to plant disinformation. So, while stealing information might be important - planting information might be just as effective. Therefore, securing systems often means that you need to make sure the information in them is not modified or altered in ways which benefit your adversary.

Cyberwarfare may not be limited to state groups, and many of the current threats originate from elsewhere [31]. The conflict between hacker groups (which happen to be different national groups) are in some cases international conflicts of ideology. This paper accepts that using patriotic groups within the country to advance the national agenda is possible.

Irregardless of the actors in question, it is clear that cyberwarfare is a discussed topic. While at this time there is scant evidence of a true cyber-war happening between nation states. There is evidence for skirmishes between long time rivals, like the USA and China. Any large company tied to a nation state would be wise to consider the possibility they may be targeted in the future and plan accordingly.

1.4.5 System Complexity

The enterprise network environment is always trying to push the boundaries to offer new services to its users [32]. We can see the shift in the last few years to bring-your-own devices and allowing workers to work from home using company provided computer systems. Within the network itself, systematic design of VLANs and other interconnectedness is often poorly understood [33]. This means that often the corporate network environment is in a constant state of redesign engineering flux where new threat avenues may be created from many different angles. Furthermore, the sheer complexity of the environment often means that no single individual has a full picture of how all the services work together to create a unified whole. Often there is many specialists which oversee their own small parts of the enterprise network. This is exacerbated with the often inadequate communication that exists within corporate structures. Should a security threat find its way into an enterprise network removing it will be a difficult challenge and it may be impossible to verify that it has been removed completely.

1.4.6 Countermeasures

APST have undergone many different evolutions over the years, and their writers have come up with more and more advanced countermeasures to prevent their detection and removal[34]. This has been briefly studied using such systems as BEAGLE[35] which show how malware authors refine their malware over time. It is important for us to realize during our research that new techniques will be created which may make our detection tools ineffective or obsolete and thus the current research cannot arrive at a perfect solution. Rather a balance will need to be supposed which allows adequate protection on the enterprise network.

1.4.7 Aging Infrastructure

No analysis would be complete without the mention of aging legacy IT systems present within any computing environment. From rumors about a forgotten server hiding in a back room or the

necessity of keeping an aging computer system operational to ensure business continuity, often the enterprise depends on legacy IT systems. These systems present a unique challenge and their modernization is a concern to computing professionals[36]. However, these systems present a unique target for malware as they are often poorly understood and often sparsely maintained. Furthermore, skill sets within industry for their operation are often sparse or difficult to find [37] so they may be maintained by system administrations who do not often understand their full complexities.

1.5 Literature Review

Our short literature review used online databases to search for topics related to "Advanced Persistent Threats" and "APT". The five main databases queried were: ScienceDirect, IEEE Xplore Digital Library, ACM Digital Library, Google Scholar, and SpringerLink. We found the the Google Scholar "related article" feature particularly helpful in tracking related articles which were indirectly related to our topic of interest.

APST have been documented to have several distinct phases in their life cycle [38][39][40][41]. While these stages may differ between published papers, it is widely accepted that humans are exploited in the early stages of attack[42]. The stages also provide differentiation for the strategies that can be deployed for detection as summarized in Table 1.1. Therefore, we will briefly look at the various stages as identified in the papers for completeness of the literature review and to make sure the reader is familiar with them. Also, depending at which stage an infection is detected suggests potential areas for removal or containment of the malware within the enterprise network. For a real-world analysis of APST stages of attack and some of the related actions taken by the malware at each stage please see [43].

The APT attack stage model adopted during our investigation will be the one contained in [38]. This model is selected because it is compact with four stages, each stage is clearly laid out in the paper presented. It is not as in depth as the phases identified in [41], but nevertheless we believe it is sufficient for our purposes and you can refer back to the comparison between the models for

comparisons. This model has four main phases: prepare stage, access stage, resident stage, and harvest stage. Within each of these main stages there are several sub-stages which the threat actors will take [44].

One of the high level methods of mitigating APST is to model the network and the various vulnerabilities that exist on it. One method which could be used by a security analyst is the Optimized Attribute Attack Graph which models the network and potential attack vectors and methods [45]. This graph can show the potential attack avenues for multiple hosts in the network, however the authors do not present an automated method of generating these graphs so usefulness would be limited for us. The idea of presenting a holistic view of the network lead us to another paper which presents an ontology based approach where various data providers can contribute to behaviour detection on the network. This approach known as TAON gives, "an OWL-based ontology offering a holistic view on actors, assets, and threat details, which are mapped to individual abstracted events and anomalies" [46]. This method will aggregate data from various data providers which are software components which provide information. This is a breed of behavioral detection systems which are posited as the next generation of APST detection methods because they model the stages of attack and try to use those as methods of detection. Unfortunately, this approach did not come with a corresponding software implementation so while the idea is novel we cannot implement it during our project. However, it does highlight the use of semantic web capabilities for the detection of threats. Also, it provides a assistant with data providers which could be used on a network for detection.

Most detection methods usually rely on network traffic analysis and inspection[47]. One paper which stood out was [48] which compared the network monitor implementations which were broken into packet capture, deep packet inspection, and flow-based observation. The paper brought to our attention Bro which is flexible and allows the creation of various detection methods. Another method of APST network detection is to detect traffic flows for Remote Access Tools running within internal enterprise networks [49]. The inspection of traffic by a host-based detection mod-

ule on each IP enabled device was an effective way of finding over 90% of the threats of infected systems. It is important to monitor internal hosts because often defensive strategies use an eggshell approach where the perimeter firewall is the only place where attacks might be stopped. The collaborative monitoring of hosts within the network is important to stop the lateral movement of APST within the network [50]. A final paper presents various real world recommendations for detection of threats on the network [51] however it is only a proposal and does not present real-world results for analysis.

Another proactive defense strategy is the use of honeypots which sit on the network and mimic real systems to lure APST to them[52]. The use of KFSensor was cited as the selection solution by this author, for our uses we will need to select more open source tools for our investigation. However, the main idea is the use of proactive notifications which are sent to the network administrator when the honeypot receives incoming network traffic.

The removal of APST was not an easy find in our investigation. When a network is compromised it would be prudent to simply take all infected hosts off line and rebuild them from known good backups. The detection components can be used to show when a host is compromised,"Across the first organisations estate of 5,000 plus machines, there were four that were infected with the malware" [53], and then a backup solution can be used to restore the host after forensic analysis. Removal becomes even more difficult because often the best malware authors build on the successes and mistakes of other malware authors when creating their malware designs [34].

The life cycle of the APST is important when considering different detection and removal methods for them. There is much research done into detection, but much less research was discovered for removal.

1.6 Potential Solutions to the Problem/Opportunity

There are many claimed solutions to the APST scourge, but many of them are difficult and costly to implement. We will attempt to secure our simulated enterprise network using free and open source tool. Therefore, while some proprietary security products perform well in this space we discount them from our analysis [54]. The main reason for this is due to funding constraints and to make the research more accessible to a wider audience.

In this section we will explore some solutions suggested by the literature review which might help to assist us. One of the most cited defense strategies is simply *user education about the dangers of cyberspace*[1]. However, we are going to focus on specific technologies we can implement within the enterprise to attempt to secure it better.

Some initial suggestions are good, and a first look presents some novel areas to consider. A PhD thesis gives voluminous suggestions without recommending any specific pieces of software [55]. A high level analysis provides many examples of signature based, anomaly based, and specification based detection methods [56]. An Intel presentation gives us some future considerations about how Intel Chips may be protected against these threats [57]. We also encounter general advice that all security appliances and software should be kept up-to-date [58], furthermore we should limit physical access to all facilities. These are excellent starting suggestions, but do not provide the concrete recommendations we need for our purposes.

Force the DNS servers on the local network to make use of fast-flux DNS domain [59]. Also, by limiting access that malware has to Command and Control servers prevents it from updating itself. The [59] paper gives us a list of several countermeasures which are highly recommended, which include: patch management, network segregation, white listing common Internet traffic, controlling access to software and hardware within the environment.

In [60] additional suggestions are given at the end of the paper. They include controlling all external media, making sure the endpoint security is kept up-to-date, and implementing network access control. Some of the tools recommended in this paper for network monitoring are OSSECC,

Snort, Sguil, and Splunk. However, some papers [61] only give general advice on how to monitor the network but do not provide us with a software tool recommendation for the endeavor. An additional paper expands on these tool recommendations and offers more granular suggestions [62].

Other papers [14] offer some suggestions about using the Microsoft Enhanced Mitigation Experience Toolkit to stop attacks, but this toolkit is currently end of life, so is not useful for us. This paper does suggest, "we propose to implement a detection approach that monitors if a process requests a handle to the LSASS process and performs suspicious function calls with the help of this handle.", however no source code is offered for this task. The final recommendation for Windows log analysis is indeed good, and we think having a tool which will be able to look at all the logs in a reasonable period of time could be useful.

The search for APST[1] must take into account event anomaly detection, data loss prevention. This paper presents an excellent summary of the defense strategies that can be employed at each stage of the APST life cycle, we have reproduced it in Table 1.1. This discussion illustrates the ongoing observation that different countermeasures should be used to stop APST depending on the attack stage it is currently in.

1.7 Research Methodology

1.7.1 Extended Literature Review

The literature review will be required in order to find components which can be selected for the FSM. The criteria for selection should include as many data points as possible to help in choosing the best products. A strong preference will be made for detection methods which have accompanying software projects. However, should ideas be adaptable with existing software products, those may be selected as well. The goal of the review will be to find as wide an array of software products that can be surmised to accomplish our goal of detection and removal of APST. A mind map will be created to connect various findings and hopefully identify appropriate tools.

Table 1.1: Attack techniques and countermeasures in each stage of an APT attack taken from [1, Table. 3]

Stages	**Attack techniques/tools**	**Countermeasures**
Reconnaissance and Weaponization	OSINT, Social engineering Preparing malware	Security awareness training, Patch management, Firewall
Delivery	Spear phishing, Watering hole attack	Content filtering software, NIDS, Anti-virus software
Initial Intrusion	Zero-day exploits, Remote code execution	Patch management, HIDS, Advanced malware detection
Command and Control	Exploiting legitimate services, RAT, Encryption	NIDS, SIEM, Event Anomaly detection
Lateral Movement	Privilege Escalation, Collecting data	Access control, HIDS, NIDS, Event Anomaly detection
Data Exfiltration	Compression, Encryption, Intermediary Staging	Data Loss Prevention

Most likely we will provide broad recommendations for tools that should be used for detection and removal. The review will most likely identify a class of tools, but not necessarily a specific tool.

1.7.2 Selection of Components

The components used in the detection and removal of the APST will be free or open source software in order to facilitate easy modification and setup. The components selected should be configurable in a reasonable amount of time using available resources. Therefore, we will not select any components which require expensive licensing or have restricted functionality. This will limit the scope of our investigation because we will not be able to use many proprietary solutions. However, the components must be able to function well together and should be amenable to enterprise networks.

We will be limited to select a set of three to five components. These components should be as diverse and functional as possible without limiting the enterprise network.

1.7.3 Configuration of Components

A study of the best configuration for the components will be necessary to make sure they will function within the simulated enterprise network. The technical creation of the federation and its alignment between components must be well defined in order for the research to proceed. How components will share data with the federation will need to be studied, and we hope to find a simple way for the federation to take place.

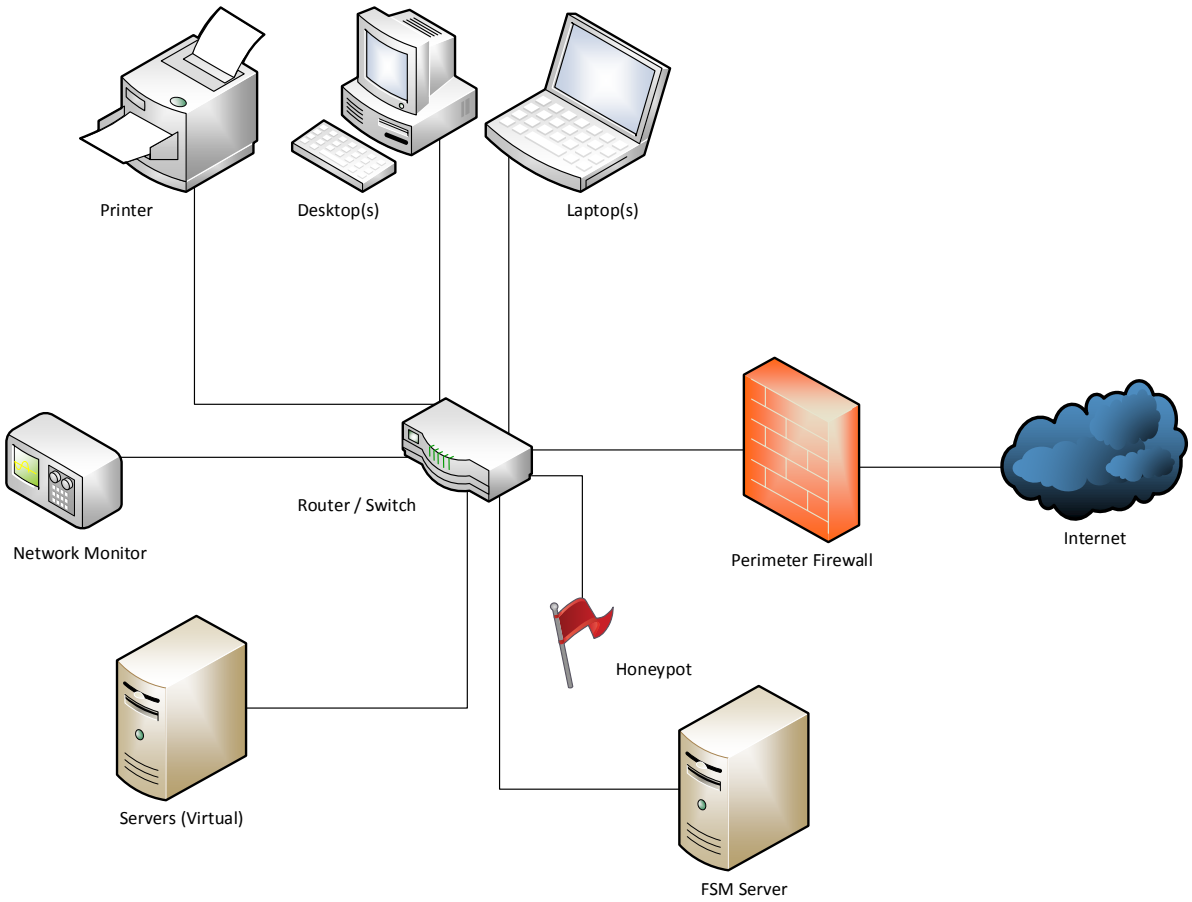
1.7.4 Programming of Federated Module

The federated module will need to be programming to allow communication between a central system and the various components selected for detection and removal of APST. The exact programming environment of the module at this time is unknown. It is surmised that standard programming practices and paradigms will be utilized along with appropriate programming environments. A GitHub repository will be setup for storing programming code produced as part of the project.

1.7.5 Penetration Testing Lab Setup

In order to test our tool we will need to setup a penetration testing lab similar to Figure 1.1 which we hope can be used to simulate a limited corporate network. Most likely this will include several computer systems running various operating systems, a domain controller, a file server, TCP/IP printing devices, and network equipment. We will draw from many sources to get a sense of the best way the lab should be built and configured, with a distinct emphasis on easy setup, configuration, and restoration of machines to a pristine state. One source of information would be chapter 4 and 5 in [63] which talks about setting up a penetration testing lab. Another source identified on setup of the lab environment for malware analysis is chapter 2 in [64]. Both these references give some guidance for the setup of a lab environment, but we will need to have an implementation specific for our needs.

Figure 1.1: Penetration Testing Lab Proposed Configuration



Within our setup there will be several different components:

- Desktop - one or more workstations running Windows end user operating systems. (May run as a guest virtual machine on a Linux host operating system.)
- FSM Server - the main analysis server where all data will be sent for processing as part of the FSM
- Honeypot - a fake high value host running on the network
- Laptop - one or more mobile workstations running Windows end user operating systems (May run as a guest virtual machine on a Linux host operating system.)
- Printer - a network device with print functionality
- Network Monitor - a computer system which is setup to use the monitoring port on the switch. (It will have the ability to watch all traffic on the internal network.)
- Perimeter Firewall - a firewall appliance which will separate the internal network from the Internet
- Router / Switch - a multi-port router / switch which will provide network connectivity for all the components in our network
- Servers - a group of Windows servers running in VMware Workstation on a single host computer. (This will allow creation of servers as required in a virtual environment.)

1.7.6 Hacking the Enterprise Network

The researcher will attempt to infiltrate the network in order to plant malware as a test of the effectiveness of the tools. Unfortunately, it is not possible to have a real APST infecting the network because we would need to have a real threat actor attacking the network. Thus the

Table 1.2: Project Deliverables

Deliverable	Description
FSM Source Code	The source code for the FSM will be uploaded to the Git repository.
Tool Source Code	The source code for the various open source tools that will make up the selected detection tools.
Final Project Report	The final project report which will be submitted to the committee at the conclusion of the project.
User Documentation	Limited documentation will be created to document for the user how the FSM can be deployed into an enterprise network. Additionally, we will collect and document the setup of our simulated enterprise network environment.
Google Sites Website	All documentation and project files will be made available through a Google Site. This website will contain links to the source code and other resources to assist anyone interested in contributing to the project further.
Google Slides Presentation	Create a short Slide presentation to present findings of the research.

researcher will function as a ”threat actor” for this exercise and will perform simulated penetration tests on the network.

1.7.7 Results and Future Work

All findings of the research will be summarized. Including some discussion about the efficacy of the various components to detect and remove APST. The overall success of the detection and removal of APST will be discussed in a qualitative manner with an overall assessment of success or failure of the methods used. Recommendations on how to fix the shortcomings of the researcher will be used to motivate future investigations.

1.8 Project Deliverables

There will be many different deliverables developed as part of this project. The main deliverables have been outlined in Table 1.2.

1.9 Project Milestones/Schedule

The first two weeks of research will be an expanded literature review and synthesis of literature to identify important trends and avenues of investigation. The mind-mapping tool CmapTools will be used to visualize connections between disparate techniques. The milestone for this phase will be the completed mind-map.

The next week will be dedicated to collecting case studies of existing APST analyses in order to compile a meta-analysis of source documents. We will obtain source code if possible, but mostly we will gather the papers and technical data relating to these APST. The milestone for this phase will be the technical data of these threats.

The next investigation will be the proposal of methods of detection, which will focus on currently available tools and existing techniques of detection. The detection tools will be subjectively ranked for efficacy, to discover which tools will be used for form the FSM. The time for this investigation will be two weeks. The milestone for this phase will be the ranking of tools, most likely we will select the top three tools be believe will be best suited for our purposes. We will of course reference relevant section in this work 1.6.

The subsequent investigation will be the proposal of methods of removal, which will focus on currently available tools and existing techniques for removal. We do not expect removal to be an easy task, so should no removal tools be identified, we will move to strengthen methods of detection. The time allowed for this will be one week, and we will recommend at least one tool for removal.

Once the best techniques have been selected we will investigate the best architecture for the FSM. This will require selecting specific language and operating systems for the implementation of this tool. We would budget around two months for setup and coding of the FSM module. The milestone for this section will be the finalized setup for the FSM module.

The FSM module will be setup on a prototype simulated enterprise network to ensure it is fit for purpose. We will allot two weeks setup of this network, and then two weeks for testing of

various malware attacks within this environment to see how the various components of the FSM module respond. Two milestones will exist at this phase, the setup of the finished environment and the testing results from the various malware samples. Some user documentation will be generated regarding the setup of this environment, but we will use pre-existing documentation whenever possible.

The final milestone will be the creation of the project report which will include a summary of all technical and project work done during the project. This project report will be prepared using LaTeX and will closely follow the standards found in the Athabasca University project template.

A timeline has been set out in Table 1.3.

1.10 Project Participants

The following primary participants will be the individuals involved in the completion of this project. Each individual has specific duties which have been outlined in the Handbook for MSc IS Project [65]. Table 1.4 briefly outlines each project participant and their role in the project.

1.11 Project Resource Requirements

Table 1.5 and Table 1.6 summarize materials that will be required during the execution of the project. We will require several different computers running an array of operating systems. Some of this hardware will be donated from various sources, but we will still need to purchase some computer components to make the hardware suitable for our purposes.

1.12 Project Budget

The budget contained in Table 1.7 will list any known costs which may be involved in the project implementation. Costs are estimates only, and are used to assist in budgeting for project expenses.

Table 1.3: Project Timeline

Task Name	Duration	Start	Finish
Literature Review and Synthesis	14 days	Mon 2/12/18	Thu 3/1/18
Review of collected literature	7 days	Mon 2/12/18	Tue 2/20/18
Extract useful tools and techniques to MindMap	7 days	Mon 2/12/18	Tue 2/20/18
Create a library of papers	7 days	Mon 2/12/18	Tue 2/20/18
Complete CmapTools MindMap	3 days	Wed 2/21/18	Fri 2/23/18
Share MindMap with Supervisor	1 day	Mon 2/26/18	Mon 2/26/18
Case Studies of existing APST Analyses	7 days	Fri 3/2/18	Mon 3/12/18
Technical Data of APST	7 days	Fri 3/2/18	Mon 3/12/18
Proposed Detection Methods	14 days	Tue 3/13/18	Fri 3/30/18
Detection Tool Candidates	14 days	Tue 3/13/18	Fri 3/30/18
Proposed Removal Methods	7 days	Tue 3/13/18	Wed 3/21/18
Removal Tool Candidates	7 days	Tue 3/13/18	Wed 3/21/18
Final Tool Selection	4 days	Mon 4/2/18	Thu 4/5/18
Supervisor Approval	1 day	Fri 4/6/18	Fri 4/6/18
Setup Enterprise Network	14 days	Fri 4/6/18	Wed 4/25/18
Procure Hardware	3 days	Fri 4/6/18	Tue 4/10/18
Prepare Donated/Personal Hardware	2 days	Fri 4/6/18	Mon 4/9/18
Install Operating Systems	4 days	Wed 4/11/18	Mon 4/16/18
Install VMware Workstation	1 day	Tue 4/17/18	Tue 4/17/18
Setup Virtual Servers	4 days	Wed 4/18/18	Mon 4/23/18
Setup Active Directory Domain	2 days	Tue 4/24/18	Wed 4/25/18
Setup Exchange Server	2 days	Tue 4/24/18	Wed 4/25/18
Setup File Server	2 days	Tue 4/24/18	Wed 4/25/18
Setup Print Server	2 days	Tue 4/24/18	Wed 4/25/18
Connect Workstations to Network	5 days	Fri 4/6/18	Thu 4/12/18
Setup Network Tap	2 days	Fri 4/6/18	Mon 4/9/18
Setup HoneyPot	2 days	Fri 4/6/18	Mon 4/9/18
Setup Perimeter Firewall	2 days	Fri 4/6/18	Mon 4/9/18
Setup Internet Connection	2 days	Fri 4/6/18	Mon 4/9/18
Setup Clients	3 days	Fri 4/6/18	Tue 4/10/18
Implementation of the Federated Security Module (FSM)	48 days	Thu 4/26/18	Mon 7/2/18
Setup FSM Server	28 days	Thu 4/26/18	Mon 6/4/18
Setup Tool 1,2,3,4	7 days each	Thu 4/26/18	Mon 6/4/18
Setup Tool Cooperation	20 days	Tue 6/5/18	Mon 7/2/18
Simulated Attacks	12 days	Mon 7/2/18	Wed 7/18/18
Phase 1,2,3,4 Attacks	3 days each	Mon 7/2/18	Wed 7/18/18
Analysis of Detection and Removal Responses from FSM	7 days	Mon 7/23/18	Tue 7/31/18
List of Recommendations	14 days	Mon 7/23/18	Thu 8/9/18
Project Report	60 days	Wed 8/1/18	Tue 10/23/18
Oral Presentation	1 day	Wed 10/24/18	Wed 10/24/18

Table 1.4: Duties of Project Participants

Project Role	**Name**	**Institution**	**Role in Project**
Committee Chair	Dr. Larbi Esmahi	Athabasca University	Oversees the project committee and provides final approval on project reports.
Project Supervisor and Project Sponsor	Dr. Harris Wang	Athabasca University	Mentors graduate student through all phases of project, provides project recommendations and corrections as necessary.
Additional Reader	Dr. Qing Tan	Athabasca University	Reviews final project deliverables and provides input to project committee.
Graduate Student	Wade Wesolowsky	Athabasca University	Performs research into topic and produces final report.

Table 1.5: Hardware

Description	**Supplier / Availability**
Server for Virtual Machines (16 GB RAM, 500 GB SSD hard drive, Intel processor)	Re-purposed hardware
FSM Server (16 GB RAM, 500 GB SSD hard drive, Intel processor)	Donated hardware
Network Monitor (Raspberry Pi 3)	Purchased from research funds
Cisco SG350-10PP 10-Port Switch	Purchased from research funds
Perimeter Firewall (Dual Nics Router pc Kettop-Mi18C)	Purchased from research funds
Client Hosts	Donated hardware

Table 1.6: Software

Description	**Supplier / Availability**
Software Repository (GitHub.com) for storing source code	Researcher from online sources
Overleaf.com for creating LaTeX documentation	Researcher from online sources
Google Apps for Business for storing papers and creating a project website	Researcher from online sources
Windows Desktop Operating Systems for simulation testing exercises	Researcher from OntheHub software
Windows Server Operating Systems for simulation testing exercises	Researcher from OnTheHub software
Open Source Tools for creating APST federated solution	Researcher from online sources
CmapTools will be used to create mind-maps for further analysis	Researcher from online sources
VMware Workstation 14 for Windows	Researcher from online sources
Microsoft Project 2013	Researcher from OntheHub Software
Microsoft Visio 2013	Researcher from OntheHub software

Table 1.7: Cost of Materials

Material	**Description**	**Cost ($)**
Software	Git Software Repository (GitHub.com)	0
Software	Overleaf.com	12 / month
Software	Google Apps for Business	5 / month
Hardware	500 GB SSD	250
Hardware	500 GB SSD	250
Hardware	Cisco SG350-10PP 10-Port Switch	250
Hardware	Raspberry Pi 3 Complete Starter Kit - 32 GB Edition	100
Hardware	Dual Nics Router pc Kettop-Mi18C - Celron MiniPC	150

The SSDs will be put into the two server systems, as they both will be donated without hard drives.

We will apply for $1000 of project funding through grants available at Athabasca University. This research funding will cover the cost of various hardware components necessary to carry out the research.

REFERENCES

[1] P. Chen, L. Desmet, and C. Huygens, "A study on advanced persistent threats," in *IFIP International Conference on Communications and Multimedia Security*. Springer, 2014, pp. 63–72. [Online]. Available: https://lirias.kuleuven.be/bitstream/123456789/461050/1/2014-apt-study.pdf

[2] Y. Wang and J. Yang, "Ethical hacking and network defense: Choose your best network vulnerability scanning tool," in *2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, March 2017, pp. 110–113. [Online]. Available: https://doi.org/10.1109/WAINA.2017.39

[3] C. Stoll, *The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage*. Gallery Books, 2005. [Online]. Available: https://books.google.ca/books?id=9B1RfCAar2cC

[4] B. I. D. Messaoud, K. Guennoun, M. Wahbi, and M. Sadik, "Advanced persistent threat: New analysis driven by life cycle phases and their challenges," in *2016 International Conference on Advanced Communication Systems and Information Security (ACOSIS)*, Oct 2016, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ACOSIS.2016.7843932

[5] C. Tankard, "Advanced persistent threats and how to monitor and deter them," *Network Security*, vol. 2011, no. 8, pp. 16 – 19, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1353485811700861

[6] M. Courtney, "States of cyber warfare," *Engineering Technology*, vol. 12, no. 3, pp. 22–25, April 2017. [Online]. Available: https://doi.org/10.1049/et.2017.0300

[7] N. Shalev, I. Keidar, Y. Weinsberg, Y. Moatti, and E. Ben-Yehuda, "Watchit: Who watches your it guy?" in *Proceedings of the 26th Symposium on Operating Systems Principles*, ser. SOSP '17. New York, NY, USA: ACM, 2017, pp. 515–530. [Online]. Available: https://doi.org/10.1145/3132747.3132752

[8] J. Estublier, H. Verjus, and P.-Y. Cunin, "Building software federation," in *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA 2001, Las Vegas, USA*, 2001. [Online]. Available: http://www-adele.imag.fr/Les.Publications/intConferences/PDPTA2001Est.pdf

[9] M. Ates, J. Fayolle, C. Gravier, and J. Lardon, "Complex federation architectures: stakes, tricks & issues," in *Proceedings of the 5th international conference on Soft computing as transdisciplinary science and technology*. ACM, 2008, pp. 152–157. [Online]. Available: https://doi.org/10.1145/1456223.1456258

[10] T. R. Jackson, J. G. Levine, J. B. Grizzard, and H. L. Owen, "An investigation of a compromised host on a honeynet being used to increase the security of a large enterprise network," in *Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004.*, June 2004, pp. 9–14. [Online]. Available: https://doi.org/10.1109/IAW.2004.1437791

[11] R. Souza, C. Chavez, and R. A. Bittencourt, "Rapid releases and patch backouts: A software analytics approach," *IEEE Software*, vol. 32, no. 2, pp. 89–96, Mar 2015. [Online]. Available: https://doi.org/10.1109/MS.2015.30

[12] S. Burji, K. J. Liszka, and C. C. Chan, "Malware analysis using reverse engineering and data mining tools," in *2010 International Conference on System Science and Engineering*, July 2010, pp. 619–624. [Online]. Available: https://doi.org/10.1109/ICSSE.2010.5551719

[13] P. OKane, S. Sezer, and K. McLaughlin, "Obfuscation: The hidden malware," *IEEE Security Privacy*, vol. 9, no. 5, pp. 41–47, Sept 2011. [Online]. Available: https://doi.org/10.1109/MSP.2011.98

[14] M. Ussath, D. Jaeger, F. Cheng, and C. Meinel, "Advanced persistent threats: Behind the scenes," in *2016 Annual Conference on Information Science and Systems (CISS)*, March 2016, pp. 181–186. [Online]. Available: https://doi.org/10.1109/CISS.2016.7460498

[15] (2017) Vault 7: Cia hacking tools revealed. WikiLeaks. Accessed 2017-12-06. [Online]. Available: https://wikileaks.org/ciav7p1/

[16] Network operations division cryptographic requirements. Central Intelligence Agency. Accessed 2017-12-06. [Online]. Available: https://wikileaks.org/ciav7p1/cms/files/NOD%20Cryptographic%20Requirements%20v1.1%20TOP%20SECRET.pdf

[17] Network operations division cne operational data exchange format (codex) specification. Central Intelligence Agency. Accessed 2017-12-06. [Online]. Available: https://wikileaks.org/ciav7p1/cms/files/Codex-Spec-v1-SECRET.pdf

[18] Network operations division persisted dll specification. Central Intelligence Agency. Accessed 2017-12-06. [Online]. Available: https://wikileaks.org/ciav7p1/cms/files/ICE-Spec-v3-final-SECRET.pdf

[19] Network operations division in-memory code execution specification. Central Intelligence Agency. Accessed 2017-12-06. [Online]. Available: https://wikileaks.org/ciav7p1/cms/files/ICE-Spec-v3-final-SECRET.pdf

[20] M. Hanspach and M. Goetz, "On covert acoustical mesh networks in air," *arXiv preprint arXiv:1406.1213*, 2014. [Online]. Available: https://arxiv.org/pdf/1406.1213.pdf

[21] L. Shing, J. Astacio, A. Figueroa, and C. C. Shing, "Vulnerabilities of radio frequencies," in *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Aug 2015, pp. 2682–2686. [Online]. Available: https://doi.org/10.1109/FSKD.2015.7382381

[22] G. Schryen and R. Kadura, "Open source vs. closed source software: Towards measuring security," in *Proceedings of the 2009 ACM Symposium on Applied Computing*, ser. SAC '09. New York, NY, USA: ACM, 2009, pp. 2016–2023. [Online]. Available: https://doi.org/10.1145/1529282.1529731

[23] R. Wojtczuk and A. Tereshkin, "Attacking intel bios," *BlackHat, Las Vegas, USA*,

2009. [Online]. Available: http://www.blackhat.com/presentations/bh-usa-09/WOJTCZUK/BHUSA09-Wojtczuk-AtkIntelBios-SLIDES.pdf

[24] R. Minnich. Replace your exploit-ridden firmware with linux. YouTube. Accessed 2017-12-06. [Online]. Available: https://youtu.be/iffTJ1vPCSo

[25] A. Tanenbaum. An open letter to intel. Accessed 2017-12-06. [Online]. Available: http://www.cs.vu.nl/~ast/intel/

[26] Y. Altshuler, N. Aharony, A. Pentland, Y. Elovici, and M. Cebrian, "Stealing reality: When criminals become data scientists (or vice versa)," *IEEE Intelligent Systems*, vol. 26, no. 6, pp. 22–30, Nov 2011. [Online]. Available: https://doi.org/10.1109/MIS.2011.78

[27] F. Li, A. Lai, and D. Ddl, "Evidence of advanced persistent threat: A case study of malware for political espionage," in *2011 6th International Conference on Malicious and Unwanted Software*, Oct 2011, pp. 102–109. [Online]. Available: https://doi.org/10.1109/MALWARE.2011.6112333

[28] A. A. Adams, "Report of a debate on snowden's actions by acm members," *SIGCAS Comput. Soc.*, vol. 44, no. 3, pp. 5–7, Oct. 2014. [Online]. Available: https://doi.org/10.1145/2684097.2684099

[29] N. Kshetri, "Cyberwarfare: Western and chinese allegations," *IT Professional*, vol. 16, no. 1, pp. 16–19, Jan 2014. [Online]. Available: https://doi.org/10.1109/MITP.2014.4

[30] S. Bazan, "A new way to win the war," *IEEE Internet Computing*, vol. 21, no. 4, pp. 92–97, 2017. [Online]. Available: https://doi.org/10.1109/MIC.2017.2911419

[31] T. A. Berson and D. E. Denning, "Cyberwarfare," *IEEE Security Privacy*, vol. 9, no. 5, pp. 13–15, Sept 2011. [Online]. Available: https://doi.org/10.1109/MSP.2011.132

[32] J. L. G. Dietz and J. A. P. Hoogervorst, "Enterprise ontology in enterprise engineering," in *Proceedings of the 2008 ACM Symposium on Applied Computing*, ser. SAC

'08. New York, NY, USA: ACM, 2008, pp. 572–579. [Online]. Available: https://doi.org/10.1145/1363686.1363824

[33] Y.-W. E. Sung, X. Sun, S. G. Rao, G. G. Xie, and D. A. Maltz, "Towards systematic design of enterprise networks," *IEEE/ACM Trans. Netw.*, vol. 19, no. 3, pp. 695–708, Jun. 2011. [Online]. Available: https://doi.org/10.1109/TNET.2010.2089640

[34] J. Moubarak, M. Chamoun, and E. Filiol, "Comparative study of recent mea malware phylogeny," in *2017 2nd International Conference on Computer and Communication Systems (ICCCS)*, July 2017, pp. 16–20. [Online]. Available: https://doi.org/10.1109/CCOMS.2017.8075178

[35] M. Lindorfer, A. Di Federico, F. Maggi, P. M. Comparetti, and S. Zanero, "Lines of malicious code: Insights into the malicious software industry," in *Proceedings of the 28th Annual Computer Security Applications Conference*, ser. ACSAC '12. New York, NY, USA: ACM, 2012, pp. 349–358. [Online]. Available: https://doi.org/10.1145/2420950.2421001

[36] R. Khadka, B. V. Batlajery, A. M. Saeidi, S. Jansen, and J. Hage, "How do professionals perceive legacy systems and software modernization?" in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014. New York, NY, USA: ACM, 2014, pp. 36–47. [Online]. Available: https://doi.org/10.1109/10.1145/2568225.2568318

[37] P. A. Sandborn and V. J. Prabhakar, "The forecasting and impact of the loss of critical human skills necessary for supporting legacy systems," *IEEE Transactions on Engineering Management*, vol. 62, no. 3, pp. 361–371, Aug 2015. [Online]. Available: https://doi.org/10.1109/TEM.2015.2438820

[38] M. Li, W. Huang, Y. Wang, W. Fan, and J. Li, "The study of apt attack stage model," in *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, June 2016, pp. 1–5. [Online]. Available: https://doi.org/10.1109/ICIS.2016.7550947

[39] R. Brewer, "Advanced persistent threats: minimising the damage," *Network Security*, vol. 2014, no. 4, pp. 5 – 9, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1353485814700406

[40] J. Chen, C. Su, K.-H. Yeh, and M. Yung, "Special issue on advanced persistent threat," *Future Generation Computer Systems*, vol. 79, no. Part 1, pp. 243 – 246, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17324913

[41] B. I. D. Messaoud, K. Guennoun, M. Wahbi, and M. Sadik, "Advanced persistent threat: New analysis driven by life cycle phases and their challenges," in *2016 International Conference on Advanced Communication Systems and Information Security (ACOSIS)*, Oct 2016, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ACOSIS.2016.7843932

[42] M. Bere, F. Bhunu-Shava, A. Gamundani, and I. Nhamu, "How advanced persistent threats exploit humans," *International Journal of Computer Science Issues (IJCSI)*, vol. 12, no. 6, p. 170, 2015. [Online]. Available: https://www.researchgate.net/profile/Attlee_Gamundani/publication/301689293_How_Advanced_Persistent_Threats_Exploit_Humans/links/57223ec208ae586b21d3e6c6.pdf

[43] (2017) Advanced persistent threat activity targeting energy and other critical infrastructure sectors. United States Computer Emergency Readiness Team. Accessed 2017-12-09. [Online]. Available: https://www.us-cert.gov/ncas/alerts/TA17-293A

[44] A. Lemay, J. Calvet, F. Menet, and J. M. Fernandez, "Survey of publicly available reports on advanced persistent threat actors," *Computers and Security*, vol. 72, no. Supplement C, pp. 26 – 59, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404817301608

[45] M. Li, W. Huang, Y. Wang, and W. Fan, "The optimized attribute attack graph based on apt attack stage model," in *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, Oct 2016, pp. 2781–2785. [Online]. Available: https://doi.org/10.1109/CompComm.2016.7925204

[46] R. Luh, S. Schrittwieser, and S. Marschalek, "Taon: An ontology-based approach to mitigating targeted attacks," in *Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services*, ser. iiWAS '16. New York, NY, USA: ACM, 2016, pp. 303–312. [Online]. Available: https://doi.org/10.1145/3011141.3011157

[47] M. Marchetti, F. Pierazzi, M. Colajanni, and A. Guido, "Analysis of high volumes of network traffic for advanced persistent threat detection," *Computer Networks*, vol. 109, no. Part 2, pp. 127 – 141, 2016, traffic and Performance in the Big Data Era. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128616301633

[48] I. Ghafir, V. Prenosil, J. Svoboda, and M. Hammoudeh, "A survey on network security monitoring systems," in *2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, Aug 2016, pp. 77–82. [Online]. Available: https://doi.org/10.1109/W-FiCloud.2016.30

[49] M. Yamada, M. Morinaga, Y. Unno, S. Torii, and M. Takenaka, "Rat-based malicious activities detection on enterprise internal networks," in *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*, Dec 2015, pp. 321–325. [Online]. Available: https://doi.org/10.1109/ICITST.2015.7412113

[50] A. Greco, A. Caponi, and G. Bianchi, "Facing lateral movements using widespread behavioral probes," in *2016 11th International Conference for Internet Technology and Secured Transactions (ICITST)*, Dec 2016, pp. 159–160. [Online]. Available: https://doi.org/10.1109/ICITST.2016.7856688

[51] I. Ghafir and V. Prenosil, *Proposed Approach for Targeted Attacks Detection*. Cham: Springer International Publishing, 2016, pp. 73–80. [Online]. Available: https://doi.org/10.1007/978-3-319-24584-3_7

[52] Z. Saud and M. H. Islam, "Towards proactive detection of advanced persistent threat (apt) attacks using honeypots," in *Proceedings of the 8th International Conference on Security of*

Information and Networks, ser. SIN '15. New York, NY, USA: ACM, 2015, pp. 154–157. [Online]. Available: https://doi.org/10.1145/2799979.2800042

[53] M. Auty, "Anatomy of an advanced persistent threat," *Network Security*, vol. 2015, no. 4, pp. 13 – 16, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1353485815300283

[54] G. G. Granadillo, J. Garcia-Alfaro, H. Debar, C. Ponchel, and L. R. Martin, "Considering technical and financial impact in the selection of security countermeasures against advanced persistent threats (apts)," in *2015 7th International Conference on New Technologies, Mobility and Security (NTMS)*, July 2015, pp. 1–6. [Online]. Available: https://doi.org/10.1109/NTMS.2015.7266480

[55] R. Mehresh, "Schemes for surviving advanced persistent threats," Ph.D. dissertation, PhD thesis, Faculty of the Graduate School of the University at Buffalo, State University of New York, 2013. [Online]. Available: https://www.cse.buffalo.edu/caeiae/documents/pdf/ruchika-mehresh-2013-dissertation.pdf

[56] H. M. Deylami, R. C. Muniyandi, I. T. Ardekani, and A. Sarrafzadeh, "Taxonomy of malware detection techniques: A systematic literature review," in *2016 14th Annual Conference on Privacy, Security and Trust (PST)*, Dec 2016, pp. 629–636. [Online]. Available: https://doi.org/10.1109/PST.2016.7906998

[57] D. Durham, "Mitigating exploits, rootkits and advanced persistent threats," in *2014 IEEE Hot Chips 26 Symposium (HCS)*, Aug 2014, pp. 1–39. [Online]. Available: https://doi.org/10.1109/HOTCHIPS.2014.7478798

[58] A. K. Sood and R. J. Enbody, "Targeted cyberattacks: A superset of advanced persistent threats," *IEEE Security Privacy*, vol. 11, no. 1, pp. 54–61, Jan 2013. [Online]. Available: https://doi.org/10.1109/MSP.2012.90

[59] N. Virvilis, D. Gritzalis, and T. Apostolopoulos, "Trusted computing vs. advanced persistent

threats: Can a defender win this game?" in *2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing*, Dec 2013, pp. 396–403. [Online]. Available: https://doi.org/10.1109/UIC-ATC.2013.80

[60] J. Vukalovi and D. Delija, "Advanced persistent threats - detection and defense," in *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, May 2015, pp. 1324–1330. [Online]. Available: https://doi.org/10.1109/MIPRO.2015.7160480

[61] S. Torii, M. Morinaga, T. Yoshioka, T. Terada, and Y. Unno, "Multi-layered defense against advanced persistent threats (apt)," *Fujitsu Sci. Tech. J*, vol. 50, no. 1, pp. 52–59, 2014. [Online]. Available: http://www.fujitsu.com/global/documents/about/resources/publications/fstj/archives/vol50-1/paper09.pdf

[62] B. Binde, R. McRee, and T. J. OConnor, "Assessing outbound traffic to uncover advanced persistent threat," *SANS Institute. Whitepaper*, 2011. [Online]. Available: https://www.sans.edu/student-files/projects/JWP-Binde-McRee-OConnor.pdf

[63] "Professional penetration testing," in *Professional Penetration Testing*, T. Wilhelm, Ed. Boston: Syngress, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9781597494250000014

[64] M. Sikorski and A. Honig, *Practical Malware Analysis: A Hands-On Guide to Dissecting Malicious Software*. No Starch Press, 2012. [Online]. Available: https://books.google.ca/books?id=FQC8EPYy834C

[65] (2017) Handbook for msc is projects. Athabasca University. Accessed 2017-12-03. [Online]. Available: http://mscis.athabascau.ca/resources/handbook-mscis-projects.php