UNIVERSIDADE DE LISBOA
Faculdade de Ciências
Departamento de Informática

ADVANCED PERSISTENT THREATS

António João Caldeira Lima

Dissertação orientada pelo Prof. Doutor Pedro Miguel Frazão Fernandes Ferreira
e co-orientada por João Manuel Gomes Moura

DISSERTAÇÃO

MESTRADO EM SEGURANÇA INFORMÁTICA
2015

Agradecimentos

Para os meus pais, António e Margarida, que fizeram os possíveis e os impossíveis para garantir os meus estudos, obrigado por estarem sempre presentes, obrigado por acreditarem e um muito obrigado pela força que me dão no dia-a-dia. Para as minhas irmãs, Catarina, Joana, Gabriela e Margarida, e o meu irmão Francisco, que sabiam sempre o que dizer e o que fazer para me motivar, um muito obrigado por me aturarem. Para o resto da minha família e amigos, pelas vossas questões, pelo vosso apoio e por saberem quando o que eu mais precisava era relaxar. À AnubisNetworks, por ter aceitado trabalhar comigo neste projeto, em especial à Salomé Viterbo, e o tempo que dedicou para me ajudar, e a todos os que me acompanhavam nos almoços na Escadinha, foram preciosos alicerces para o meu desenvolvimento pessoal e profissional, muito obrigado. Obrigado João Moura por todo o aconselhamento ao longo do projeto e por manteres uma cara séria ao ouvir as minhas ideias mais estranhas. Obrigado, Professor Pedro Ferreira, por ter aceitado este desafio na área da segurança e por se ter disponibilizado a aprender, ensinar e aconselhar durante este trabalho, a qualquer hora e em qualquer dia.

São todos a minha força, um muito obrigado!

”O Homem depende do seu pensamento.”
-Mokiti Okada

Resumo

Os sistemas computacionais tornaram-se uma parte importante da nossa sociedade, para além de estarmos intrinsecamente ligados a eles, a maioria da informação que utilizamos no nosso dia-a-dia está no seu formato digital. Ao contrário de um documento físico, um documento digital está exposto a uma maior variedade de ameaças, principalmente se estiver de alguma forma disponível à Internet. Informação é poder, por isso não é de admirar que alguém, algures esteja a tentar roubá-la, assim, é facto que os adversários já operam neste novo mundo. Ladrões, terroristas e mesmo a máfia começaram a utilizar a internet como um meio para alcançar os seus fins. A cibersegurança tenta proteger a informação e os sistemas contra estes e outros tipos de ameaças, utilizando anti-vírus, *firewalls* ou detetores de intrusões, entre outros. Infelizmente as notícias continuam a sair, milhões de euros roubados a bancos por via informática, empresas saqueadas da sua propriedade intelectual e governos envergonhados por os seus segredos serem expostos ao mundo. A questão coloca-se, porque é que os sistemas de segurança estão a falhar? Como está o adversário a ultrapassá-los? A verdade hoje em dia é que os atacantes não só adquiriram talentos avançados na área como também têm acesso a ferramentas extremamente sofisticadas e vão fazer uso delas para serem bem-sucedidos nos seus objetivos, sejam estes o roubo de informação, o objetivo mais comum e por isso o mais abordado neste trabalho, seja o ataque a infraestruturas críticas.

Advanced Persistent Threat(APT), ou ameaça avançada persistente, é um termo utilizado para caracterizar atacantes sofisticados, organizados e com recursos para concretizar ataques informáticos. Inventado pela força aérea Americana em 2006, o termo era uma forma de discutir intrusões informáticas com pessoal não militar. Nas suas origens, a palavra **Ameaça** indica que o adversário não é um pedaço de código automático, ou seja, o adversário é humano e é este humano que controla parte do ataque e contribui para o seu sucesso, **avançada** porque este humano é treinado e especializado na utilização de todo o espectro informático de forma a melhor conseguir atingir o seu objectivo e **persistente**, pois esse objectivo é formalmente definido, ou seja, o ataque só está concluído quando atingir o alvo em pleno. Infelizmente, o termo passou a ser utilizado para descrever qualquer ataque informático e a ter uma conotação extremamente comercial devido aos sistemas anti-APT que invadiram o mercado pouco tempo depois do ataque sofrido pela Google em 2010. Neste trabalho abordamos estes pressupostos, e explica-se o ver-

dadeiro significado do termo juntamente com uma forma mais científica, claramente mais útil do ponto das abordagens da engenharia. Nomeadamente, sugere-se uma visão mais abrangente da campanha de ataque, não se focando apenas no *software* utilizado pelo adversário, mas tentando olhar para a campanha como um todo; equipas, organização, manutenção e orçamento, entre outros. Mostramos também porque estes ataques são diferentes, relativamente às suas tácticas, técnicas e procedimentos, e porque merecem ser distinguidos com a sua própria designação e o seu próprio ciclo de vida. Para além de identificarmos vários ciclos de vida associados às APTs, o ciclo de vida mais utilizado para caracterizar estas campanhas de ataque foi analisado em detalhe, desde as primeiras etapas de reconhecimento até à conclusão dos objectivos. Discute-se também a essência de cada passo e porque são, ou não, importantes. De seguida realiza-se uma análise ao tipo de atacante por trás destas campanhas, quem são, quais as suas histórias e objectivos. Avalia-se também porque é que os mecanismos de defesa tradicionais continuam a ser ultrapassados e não conseguem acompanhar o passo rápido dos atacantes. Isto acontece principalmente devido à utilização de listas do que é malicioso e o bloqueio apenas do que se encontra nessa lista, chamado de *black listing*. Ainda que se tenha já realizado trabalho na área de detecção de anomalias, mostra-se também o porquê de esses sistemas continuarem a não ser suficientes, nomeadamente devido ao facto de definirem os seus pressupostos base erroneamente.

Durante a realização deste trabalho percebeu-se a falta de estatísticas que pudessem responder a algumas questões. E por isso foi realizado um estudo aos relatórios disponíveis relativos a este tipo de ataques e apresentados os resultados de uma forma simples, organizada e resumida. Este estudo veio ajudar a perceber quais os maiores objectivos neste tipo de ataque, nomeadamente a espionagem e o roubo de informação confidencial; quais os maiores vectores de ataque (sendo o e-mail o grande vencedor devido à facilidade de explorar o vector humano); quais as aplicações alvo e a utilização, ou não, de vulnerabilidades desconhecidas. Esperamos que esta recolha de informação seja útil para trabalhos futuros ou para interessados no tema.

Só depois de realizado este estudo foi possível pensar em formas de contribuir para a solução do problema imposto pelas APTs. Uma distinção ficou clara, existe não só a necessidade de detectar APTs, mas também a criticalidade da sua prevenção. A melhor forma de não ser vítima de infeção é a aplicação de boas práticas de segurança e, neste caso, a formação de todo o pessoal relativamente ao seu papel na segurança geral da organização. Aborda-se também a importância da preparação; segurança não é apenas proteger-se dos atacantes, mas principalmente saber como recuperar. Relativamente à deteção, foi realizado trabalho em duas vertentes, primeiramente e visto o trabalho ter sido realizado em ambiente de empresa, foi elaborado um plano para um sistema capaz de detectar campanhas de ataque que utilizassem o vetor de infeção do e-mail, fazendo uso dos sistemas já desenvolvidos pela AnubisNetworks que, sendo uma empresa de segurança

informática com fortes ligações ao e-mail, tinha o conhecimento e as ferramentas necessárias para a concretização do sistema. O sistema faz uso de uma caracterização de pessoas, chamado de *people mapping*, que visa a identificar os principais alvos dentro da empresa e quem exibe maiores comportamentos de risco. Esta caracterização possibilita a criação de uma lista de pessoal prioritário, que teria o seu e-mail (caso tivesse anexos ou endereços) analisado em ambiente de *sandbox*. Este sistema acabou por não ser construído e é apenas deixada aqui a sua esquematização, sendo que fica lançado o desafio para a sua realização. De forma a contribuir não só para a empresa, mas também para a comunidade científica de segurança, foi de seguida realizado trabalho de deteção em vários pontos de qualquer rede informática seguindo os quatro principais passos na execução de uma campanha APT. Decidimos então utilizar um ciclo de vida composto por quatro etapas, sendo elas, a fase de reconhecimento, a infeção inicial, o controlo e o roubo de informação. Neste modelo, procuraram-se possíveis sistemas para a deteção de eventos relacionados com APTs nos três principais pontos de qualquer rede: a Internet, a Intranet e a máquina cliente. Ao analisar cada fase em cada ponto da rede, foi possível perceber realmente quais as principais áreas de estudo e desenvolvimento para melhor detectar APTs. Mais concretamente, concluiu-se que a internet seria o ponto ideal de deteção das fases de reconhecimento, a intranet para detetar controlo e roubo de informação e a máquina cliente para detetar infeção inicial. Concluí-se o trabalho apresentando o nosso ponto de vista relativamente ao futuro, isto é, quem vai fazer uso das táticas utilizadas nas campanhas APT visto serem extremamente bem sucedidas, como vão os atacantes adaptar-se aos novos mecanismos de defesa e quais os novos possíveis vetores de infeção.

Palavras-chave: APT, Advanced Persistent Threat, Definição, Prevenção, Deteção, Futuro

Abstract

Computer systems have become a very important part of our society, most of the information we use in our everyday lives is in its digital form, and since information is power it only makes sense that someone, somewhere will try to steal it. Attackers are adapting and now have access to highly sophisticated tools and expertise to conduct highly targeted and very complex attack campaigns. Advanced Persistent Threat, or APT, is a term coined by the United States Air Force around 2006 as a way to talk about classified intrusions with uncleared personnel. It wrongly and quickly became the standard acronym to describe every sort of attack. This work tries to demystify the problem of APTs, why they are called as such, and what are the most common tactics, techniques and procedures. It also discusses previously proposed life-cycles, profile the most common adversaries and takes a look at why traditional defences will not stop them. A big problem encountered while developing this work was the lack of statistics regarding APT attacks. One of the big contributions here consists on the search for publicly available reports, its analysis, and presentation of relevant information gathered in a summarised fashion. From the most targeted applications to the most typical infection vector, insight is given on how and why the adversaries conduct these attacks. Only after a clear understanding of the problem is reached, prevention and detection schemes were discussed. Specifically, blueprints for a system to be used by AnubisNetworks are presented, capable of detecting these attacks at the e-mail level. It is based on sandboxing and people mapping, which is a way to better understand people, one of the weakest links in security. The work is concluded by trying to understand how the threat landscape will shape itself in upcoming years.

Keywords: APT, Advanced Persistent Threat, Definition, Detection, Prevention, Future

Contents

Lista de Figuras **xiii**

Lista de Tabelas **xv**

1 Introduction **1**

1.1 Motivation 2

1.2 Objectives 4

1.3 Contributions 4

1.4 Document Structure 5

2 Related Work **7**

2.1 General studies on Advanced Persistent Threats 7

2.2 Detection of Advanced Persistent Threats 8

2.3 Stopping the email before it is too late 12

2.4 Stopping data exfiltration 13

2.5 Protecting Critical Infrastructures 14

3 A study on Advanced Persistent Threats **17**

3.1 What is an Advanced Persistent Threat 17

3.1.1 How we see an Advanced Persistent Threat 18

3.2 What makes it different? 19

3.3 Analysing Advanced Persistent Threat Life cycles 21

3.4 A new APT model representation 27

3.5 Profiling attackers 31

3.6 Why traditional defences do not work? 32

4 A survey on Advanced Persistent Threat attacks **35**

5 What to do against Advanced Persistent Threats **41**

5.1 Preventing an Advanced Persistent Threat 41

5.2 Detecting an Advanced Persistent Threat 44

5.2.1 How a system would work at AnubisNetworks 45

5.2.2 Other detection schemes 49
5.3 What does the future hold? 56

6 Future work and conclusions 59
6.1 Discussion and future work 59
6.2 Conclusion 60

Bibliografia 76

List of Figures

1.1 Attack Sophistication vs. Intruder Technical Knowledge 3

2.1 Example of anomalies in a 2-dimensional data set 11

3.1 Example of infections 20
3.2 Example of ransom-ware 21
3.3 APT Lifecycle proposed by Mandiant 22
3.4 PaloAlto Networks APT Lifecycle 23
3.5 APT Lifecycle proposed by Wrightson 23
3.6 APT Lifecycle proposed by ZScaler 24
3.7 DELL SecureWorks APT Lifecycle 25
3.8 Example of a spear phishing email 26
3.9 APT Model Representation 29
3.10 APT Model Representation (part 2) 30

4.1 The number of APT reports per year 35
4.2 The most used infection vectors in the data set 36
4.3 The most targeted applications in the data set 37
4.4 Percentage of zero day vulnerabilities used in the data set 38
4.5 The attackers biggest objectives 39
4.6 Time line of APT actors 40

5.1 Proposed framework to implement in AnubisNetworks 46
5.2 Zoom of proposed layers 50
5.3 Example of host agent communicating with its Master 53

List of Tables

3.1 Example of different malware characteristics 21

5.1 Different uses of the framework depending on requirements 49

Chapter 1

Introduction

Computer systems have become a very important part of our society, most of the information we use in our everyday lives is in its digital form, and since information is power it only makes sense that someone, somewhere will try to steal it. Furthermore, critical systems like power grids, telecommunication networks and even dams are controlled by computer systems and a willing adversary can take advantage of that to inflict massive amounts of damage. Cyber-security tries to protect this information and these systems. Firewalls and anti-virus offer basic, very needed, but insufficient and inadequate protection. Intrusion detection and prevention work well for known types of attacks and intrusions, but still fail in face of sophisticated newer threats. Critical systems employing fault tolerant frameworks are still successfully exploited by skilled attackers. The big question here is: how are attackers still bypassing all of these defences? For starters, there is no such thing as perfect security. Breaches will happen. From defacements to database leaks, these attacks are the everyday of security professionals. But we will not be focusing on those.

Attackers are adapting, thieves, terrorists and even mobsters are now operating in the digital world. Nations are also conducting surveillance and attacks using computers. Attackers now have access to highly sophisticated tools and expertise to conduct highly targeted and very complex attack campaigns. In Figure 1.1[1] these will be at the top right corner. It is relatively straightforward to think that this type of adversaries will not be wasting their time with defacements or database dumps online. They want to be challenged and will go through great lengths to make sure their presence in the compromised system is not detected while controlling all of the data. But not all data is important or valuable. This is referred to as *cyber-espionage*, where highly skilled attackers with complex tools and a great deal of resources work their way into the target network to exfiltrate very specific confidential data. This can range from a competitor intellectual property to a nation's military defence strategies. It all depends on the adversary's objective. If their objective is to shut down another nation's power grid, that is referred to as *cyber-warfare*.

[1] Source: CERT Coordination Center, © 2002 by Carnegie Mellon University.

Why send the military to a nuclear facility and start a full out war when you can hide behind the anonymity of the internet and the deniability or the lack of attribution it provides, and launch a cyber-attack that slowly but surely brings that facility to a shutdown.

Advanced Persistent Threat, or APT, is a term coined by the United States Air Force around 2006, not only as a way to talk about classified intrusions with uncleared personnel, but also to describe the expertise of the attackers and complexity of their tools. It wrongly and quickly became the standard acronym to describe every sort of attack. APTs are a particular case of targeted cyber attack performed by organized groups with expertise and resources to accomplish their goals, that can target both critical infrastructures (such as telecommunications [149] or SCADA networks [76]) as well as information. Working together with AnubisNetworks, the focus will be on the e-mail infection vector, which is widely used by most APT groups.

Industrial Context Acquired by BitSight in late 2014, AnubisNetworks is a brand of NSEC, Sistemas Informáticos, SA. It was founded in 2006 and today is one of Europe's leading cyber security companies. Its focus is on anti-spam solutions, threat intelligence and botnet analysis. In the last twelve months, at the time of writing, AnubisNetworks processed more than one hundred and twenty million e-mails and discarded almost five million messages marked as spam or infected. The typical clients consist of telcos, service providers and large corporations. With companies understanding the risks of infection by e-mail, specially these advanced tools for espionage, and the big presence AnubisNetworks has in this market, it was only natural for AnubisNetworks to show interest on the topic and a partnership was created with Faculdade de Ciências, Universidade de Lisboa, to research this matter and develop this work.

1.1 Motivation

Citing Mandiant [93], "Everyone now knows what seasoned security professionals have long been aware of: there is no such thing as perfect security. Security breaches are inevitable, because determined threat actors will always find a way through the gap." And more often than not we see news about a new attack or a new vulnerability being found.

Cyber-crime is at an all time high with Norton reporting [102] that in 2013, costs related to cyber security incidents were as high as $113 Billion with the average cost per victim rising 50% compared to previous years. Cyber-crime is profitable and attackers will not back down. But who are these attackers? How are they able to be illegally inside a network for an average of 229 days before being found and why did only 33% of the companies approached by Mandiant discovered the breach by themselves [93]? Google's breach, called Operation Aurora, in 2010 [37, 138] and the RSA attack in 2011 [29, 115] got huge media attention on the issue of cyber-espionage. In the 2008 cyber-attacks against

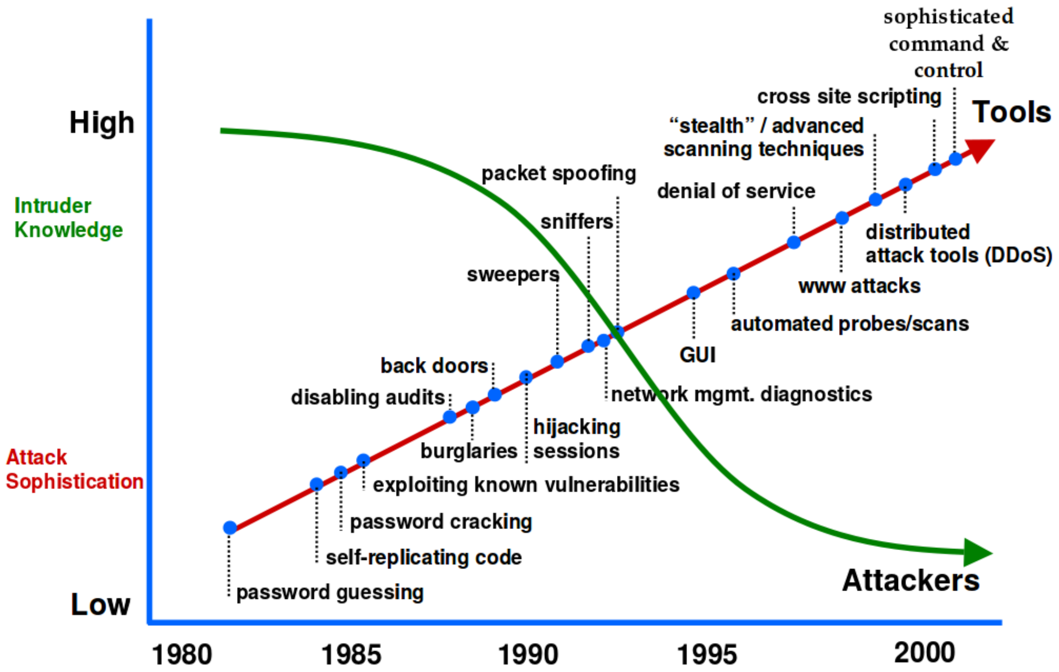


Figure 1.1: Attack Sophistication vs. Intruder Technical Knowledge

Georgia, attackers were able to disrupt news agencies, banks and telecommunication infrastructures, weeks before the Russian invasion. Together with STUXNET [151, 40], a weapon entirely made of code targeting Iran's nuclear centrifuges, Shamoon and others [148, 30], these attacks showed the world that malicious code can and will be used as a weapon. Cyber-insecurity is a serious problem and companies have started to increase their budget on information security and taking precautions with contracted third parties. FireEye [45] even reports on the fact that some companies are signing new third party contracts and are taking into account compensation if an attacker uses that third party to gain access to the company network. Information Security Media Group together with PaloAlto Networks [56] report very interesting findings, specifically, nearly 40% of their respondents have now an APT incident response plan, with more than 50% investing in tools for early detection. Unfortunately 43% still report a lack of budget to fight targeted attacks and a lot of companies still fall for the same old tricks, showing us that awareness sessions are still not a priority for these companies.

Like Sun Tzu said in *The Art of War* [137], ”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”. In this work we will gather the knowledge about adversaries, their tools, and their procedures. We achieve this by analysing the life cycle of an APT, profiling attackers and surveying attack reports from various sources. This will help the community better understand the problem as it stands currently. At the time of writing there was nothing similar in the literature. We will then try to see how the problem will evolve, i.e., how the attackers will work in the future, and propose new approaches on how to fight them.

In this modern age, mobile devices are the norm, and unfortunately Norton [102] reports that almost half of mobile device users do not use basic precautions such as passwords, security software or back up their files. These devices are everywhere, increasingly more powerful and ever more connected to each other, giving attackers a shiny new infection vector to exploit. For instance, the two-factor authentication used by many online services with the help of mobile devices, was recently broken with some clever social engineering techniques [31].

1.2 Objectives

With this work we intended to first and foremost understand Advanced Persistent Threats. From grasping the real concept of the name to discerning what are the usual tactics, techniques and procedures used in these attacks, we set ourselves to conduct a thorough investigation not only to make this information available to all, but especially in the hopes of helping AnubisNetworks in their future challenges. By understanding the problem, more specifically by understanding why the problem still persists, we will discuss ideas and related work on how one can prevent and/or detect an Advanced Persistent Threat. By working together with AnubisNetworks, a very active anti-spam company, we challenged ourselves to provide the blueprints of a system that could be used to detect and act on certain types of these threats. Mainly because of time constraints , the system was not developed in the scope of this work. Although the initial work plan was to build a forensic tool, it was soon verified that without a thorough research it would not be possible to plan an efficient forensic or prevention tool. Therefore we conducted a survey on available reports regarding these attacks with the intent of providing an agglomerate of data that will help future researchers understand the different problems persisting with APTs. And finally we tried to predict the future of these attacks. New defence frameworks are being developed, what will adapt when they are in use? How will attackers react? And how can we prepare for their adaptation?

1.3 Contributions

With this work we hope not only to better inform the information security community about the APT acronym, what it represents and what makes it different but also to provide our own, clear definition of Advanced Persistent Threat and hope that it will improve future discussions and research on the topic. The main contributions of this work are accomplished in several parts, all of them tackling the APT problem from a different angle. After analysing the typical APT life cycle in detail and understanding what each step represents, we profile the attackers and explain why traditional defences are so easily bypassed. We then explain what was clear to us amidst this research: fighting APTs can, and should, be divided in two sub-categories, prevention and detection. Prevention not only includes what to do before the attack, but also what to do after a breach. For the detection objective we first take a more specific approach and a framework is proposed for use at AnubisNetworks in order to detect these type of attacks at the e-mail vector. Then, we take a more broad look on how to detect and block APTs, by analysing possible solutions and related work at the different levels, specifically the Internet, Intranet and Host levels of abstraction. We hope not only to aid future researchers with the study here presented but also to provide interested readers with relevant data as a result of a survey conducted on APT attacks. Finally we share our thoughts on the possible directions attackers will take with their techniques and procedures for the future.

1.4 Document Structure

This work is divided in six chapters. Chapter one introduces the problem, the motivation, our objectives and our contributions. In Chapter two a review is presented of related work and the state of the art regarding not only APTs but also different ways of increasing infrastructure and systems protection against attacks. Chapter three, four and five hold the main contributions of this work. Chapter three will focus on the study of the APT problem, from definition of the acronym to the understanding of why traditional defence mechanisms do not stop it. Chapter four presents the survey conducted on Advanced Persistent Threat attacks and chapter five is where we express our thoughts on both prevention and detection of APTs, together with what might be the future of such threats. Finally, chapter six is kept for discussion and future work as well as conclusions.

Chapter 2

Related Work

2.1 General studies on Advanced Persistent Threats

Before getting into specifics, one must first fully understand the problem at hand. Because our work follows and complements this line of study we will now take a look at previous work done on APT definition, typical attack phases and adversaries as well as some general notions related to these threats. Levine [84] did an exceptionally thorough study on the topic and although he provided the usual APT definition, steps and actors, where his work really stands out is in the study of losses caused by the attacks. Not only the impact on corporations but, more importantly, collateral or secondary damages. In particular Google's loss of market share in China as a result of Operation Aurora and the suffering of the people of Iran as a consequence of international sanctions regarding the country's Nuclear program, which is now delayed because of STUXNET. It is not so much a technical work but a look at the APT problem from a social and economical point of view. Similar to this work, Marczak et al. [94] wrote about the use of APT tactics by oppressive governments on their citizens, with shocking aftermaths such as imprisonment, kidnaps and even missing citizens. Le Blond et al. [83] studied the problem through the lens of a non-governmental organization and concluded that the regular phishing emails targeting said organizations make use of remarkably tenacious and highly developed social engineering tactics.

Symantec's report [134] concentrates on the industrial espionage side of APTs, showing us some real attack examples, APT actor profiles and modus operandi. They also introduce a new term, MOTA, or Massive Organizational Targeted Attacks. The idea behind this term is simple, if the adversary has a zero day vulnerability he might want to use it to infect several organizations simultaneously for maximum information control. Without going into detail, the authors explain a system in use by the company to detect these attacks. If a certain email meets a specific set of criteria, it is analysed in a semi-manual process that will identify false positives, leaving only the remaining emails to be manually screened.

Chen et al. [23] did a high-quality study on APTs as well, from definition to attack model, and possible countermeasures. The authors do not go into great detail but give the overall ideas for defences against APTs, such as awareness training, advanced malware detection and anomaly detection. The research is quite similar to ours and we hope to add even more information, especially the survey, predictions and possible detection techniques.

Vries [32] did very interesting work on this topic. From the simulation of typical attack steps to the proposal of a new APT framework. This framework links high level attack structures to low level attack methods. It also takes into account attack steps, features, and detection location, as well as analysis and business aspects. One of the most challenging research topics, also acknowledged by this author, is the detection of the starting point of infection. Nowadays people can receive malicious files or be directed to phishing websites from several different sources, including by phone. We can try to block the attack in the email but many other infection vectors remain open.

2.2 Detection of Advanced Persistent Threats

Now that we have seen related work about APTs, we present research done on fighting them. Any good defence against APTs will involve several components working together. Recently Moon et al. [97] did research on a Multi-Layer Defence System (MLDS) that can defend against APTs using the same techniques employed by security information and event management systems, that is, by collecting and analysing log information from devices, servers and end-users. The framework makes use of eight different modules, from classifiers and storage to analysers and managers, all working together, communicating with one another to try and detect the attacks. Although theoretically feasible and in the correct path to protect organizations against APTs, in practice a system like this would need to merge thousands of logs from different hardware and software, and show those logs to an administrator that would still need to go through them manually. More research needs to be done on this topic, i.e., how to efficiently combine different system logs, detect false positives and semi-automatically, or automatically, block these attacks on the fly.

Security Information and Event Management (SIEM) systems follow a similar approach. By collecting logs and other security related events from servers, networks, firewalls, anti virus or intrusion prevention systems, which are then forwarded to a management unit, the system provides a centralized tool for administrators to inspect and flag anomalies. These systems can be profiled with what normal event conditions are and become an unsupervised anomaly detector. Gabriel et al. [49] did work in data mining to better feed SIEM systems with hidden patterns in malware. By finding associations between malware attributes and grouping them by similarity into clusters, they show how native data mining methods are applicable for the analysis of relevant security log data, such as policy violations.

Bianchi et al. [12] worked on Blacksheep, a distributed system for detecting a compromised host amongst similar machines by comparing memory dumps from different hosts. The system works relatively well as long as systems are homogeneous and a viable memory dump can be obtained, i.e., at a comparable ”checkpoint” across machines. The authors also claim that virtualization or cloud-based systems offer ideal settings for the collection of memory dumps. Therefore a company hosting virtualized workstations could take precise system snapshots and feed those to the Blacksheep to obtain the best possible results. Some of the problems that remain include the malware making little to no use of memory, the timing of the snapshots and the possibility of the malware to escape from the guest to the host machine, as it has happened in the past with other similar attacks [99].

Giura and Wang [52] not only did research on APTs, but also proposed a new model called attack pyramid and a new detection framework, which makes use of that model, to detect APTs. The goal is to differentiate normal, abnormal and known malicious behaviours by correlating the different planes in the pyramid. This can be achieved with the use of a powerful large scale distributed system. Similar to an anomaly detector, if the analysed behaviour is not normal, not malicious, and if the attack confidence indicator together with the risk level rise above a specific threshold, an APT alarm will be triggered, and just like the anomaly detector problem, we first need to understand what behaviours are considered abnormal and how to map them in the system.

Oprea et al. [105] proposed a graph framework to identify small communities of early-stage malware infections. Besides providing us with a new Command and Control[1] (C&C) detector tailored to an enterprise setting, what differentiates this work from others is the fact that the system does not require malware binaries for training. Connections to uncommon destinations, specifically, new or unpopular, are called rare destinations and are the starting point of the detection scheme. A total of eight features were used in the scoring algorithm. Although this system would not be able to detect waterhole attacks from frequently visited websites, compromised third parties that work close with the company or communications using other protocols, it managed to achieve a 98.33% of true positives with a false negative rate at 6.35%.

Since highly sophisticated and targeted attacks make use of zero-day exploits, and zero-day attacks are difficult to prevent because they exploit unknown vulnerabilities, Bilge and Dumitras [13] did an interesting study and developed a tool for automatic detection of *past* zero-day attacks. If the system detects the presence of an executable that was unknown before but now its behaviour is related to a disclosed vulnerability, that

[1] Designation given to the machine, or set of machines, controlling the attack operations by direct communication with the infected machine(s).

executable is classified as a zero-day attack. Not only this tool helps with the systematic study of these threats and provides data for the debate on the full disclosure policies, they had some interesting findings. For instance, the public disclosure of vulnerabilities was followed by an increase of up to five orders of magnitude in the volume of attacks exploiting those vulnerabilities compared to previous numbers. They could also notice that, on average, a zero-day exploit had three hundred and twelve days of use until the vulnerability was disclosed. Unfortunately, because the binary reputation data only reports on executable files and on machines running their product, they were unable to detect attacks from the web, exploits using polymorphic malware or non-executable exploits (PDF, XLSX, DOC, etc), which still are the preferred APT delivery method.

Chandola et al. [22] did a very detailed survey on anomaly detection and although one of the biggest challenges in this area are the assumptions regarding what are the normal and anomalous behaviours, the overall idea remains the same. An example is shown in Figure 2.1, taken from the same work. N1 and N2 represent the normal behaviour, in line with our work this would be the typical websites visited by our network hosts, typical emails and even typical host events, while O1, O2 and O3 are anomalies, which could be, for example, a new FTP connection to an unknown IP address. These systems and frameworks are of great help, and we will later see how they are not enough.

Anagnostakis et al. [7] presented a novel hybrid architecture of honeypots and anomaly detectors, called Shadow Honeypots. The idea is simple, traffic that is thought anomalous is processed by the shadow honeypot, this shadow is a copy of the system in production, configured to detect possible attacks. The outcome is then used to filter future attacks and update the anomaly detector. Two anomaly detection heuristics were used in the proof-of-concept, fingerprints of spreading worms and buffer overflow detection. The system cannot handle polymorphic attacks and induced a 20-50% overhead.

Friedberg et al. [47] also recently contributed with research regarding anomaly detectors to combat APTs. In their work they not only explained why current security solutions are insufficient, they also discuss the model definition of a novel anomaly detection approach based on log analysis. Contrary to the existing mechanisms, this approach constructs a model while processing the input. By combining multiple rules and analysing multiple logs, the system is able to detect anomalies that would otherwise go undetected, e.g. an administrator login with several failed attempts in a short time followed by the copy of large amounts of data. Obviously a system like this will use a lot of resources and the administrator will still have to spend valuable time performing root-cause analysis.

What we see in this research area and on these proposed frameworks is a need for event correlation and anomaly detection. That happens because, and we will come back to this later on, traditional defence systems lack the ability to detect unknown malware, they are mostly based on blacklisting, that is, a list of known malicious executables to block. Adversaries know this and exploit it by creating custom malware for their attack

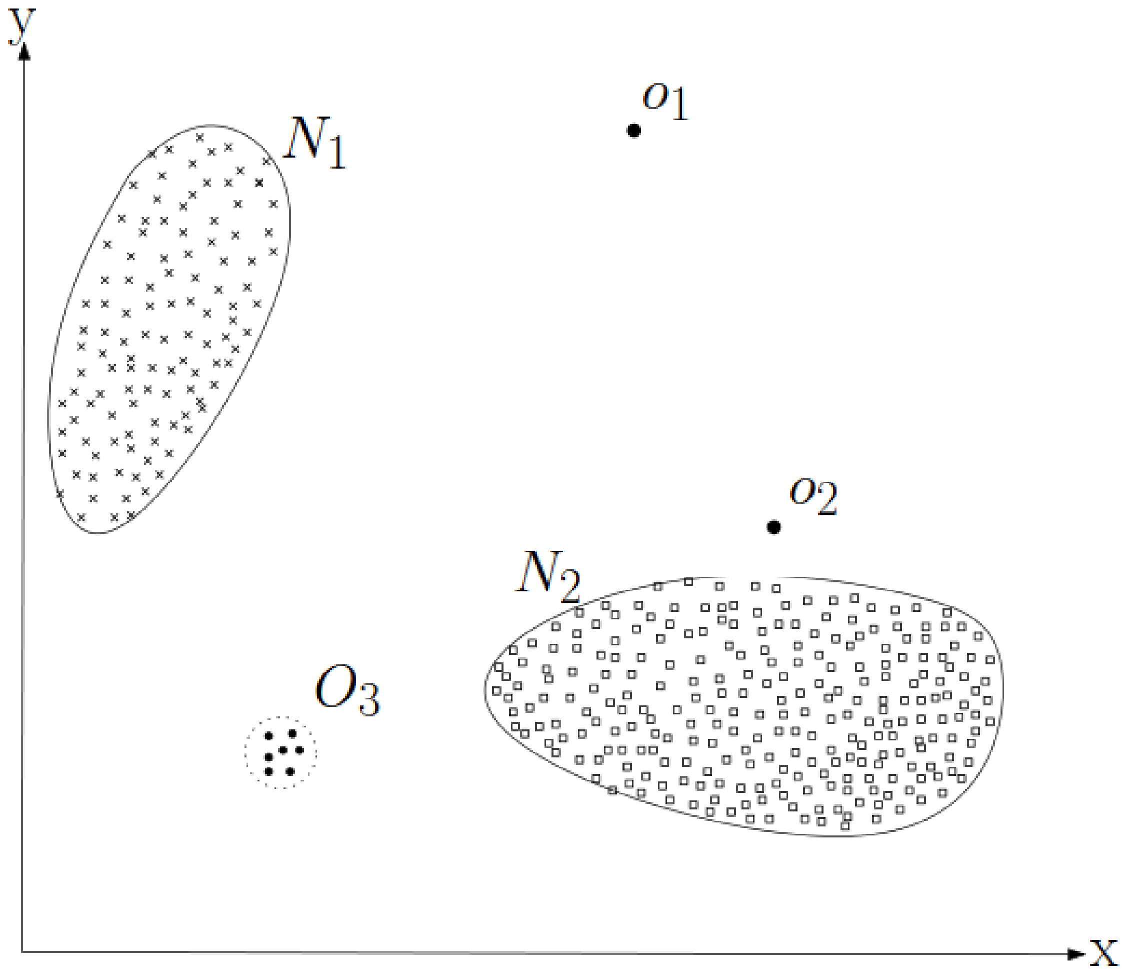


Figure 2.1: Example of anomalies in a 2-dimensional data set

campaigns, making it extremely successful. Modern schemes need to work around this and employ strategies that will detect unknown malware [11], they look to achieve this by searching for abnormal behaviour in the network, hosts or connections as well as logging every event and try to filter out security related ones that could possibly be a sign of infection. Although in the right direction, anomaly detectors still have a great number of false positives. In this ever changing online world what was abnormal yesterday may be normal today, as companies change the way they work. Additionally they are also flawed in their most basic assumptions [125], specifically, the inappropriate use of tools borrowed from the machine learning community. Anomaly detection makes great use of machine learning to find abnormal behaviour, something unseen before, but the strength of machine learning is in finding activity similar to something previously seen, something known, and because of that, machine learning components are very rarely used in real world cyber security settings. As for event correlation, looking at the security logs of a large corporation is similar to looking for a needle in a haystack, systems will still miss relevant actions and administrators will still need to manually inspect them.

2.3 Stopping the email before it is too late

If delivery is so successful, some researchers try to tackle the problem at its roots, specifically the email attack vector. Amin et al. [6] proposed a new email-filtering technique focused on persistent threat and recipient-oriented features. Taking as input the email's specific data, like the role that a specific person has in the organization or how many hits that person's email has on Google, the random forest classifier either classifies emails as targeted malicious email or non-targeted malicious email. Despite its low false positive rate, the framework does not consider file attachments. The researchers proposed that addition for future work which would make the framework much more effective in combating APTs, which still make great use of the email vector and their attachments to infect their targets. Dewan et al. [35] did similar work but focusing on the differences between normal spam, or phishing, and spear-phishing email. According to them, the additional information, or context, used together with social engineering tactics are what separate a regular phishing attack from a spear-phishing one. This type of extra information can be extracted from social media like LinkedIn. Features like job level, email subject, attachment name and most frequently occurring words in the message are used to differentiate benign, spam and spear-phishing messages. Success rates were as high as 98.28% in spear versus spam email classification, but overall accuracy dropped versus benign email because of the absence of attachment features in the data set, related to privacy issues. The addition of social media and, despite being very light, analysis of the attachment are already a very big step towards blocking APT delivery and an improvement to previous work.

Stringhini and Thonnard [130] propose a new approach to detect spear-phishing emails sent from compromised accounts, i.e., by admitting the compromise of an inside host they try to prevent further compromise by blocking malicious emails sent from that host. Writing, composition, and interaction habits were used as attributes for profiling sender behaviours. They had a detection rate above 90% for users with more than a thousand emails sent and false positives could be dealt with the use of two-factor authentication, a confirmation of some sort informing the user of an email that is about to be sent and if it was the user who sent it. An attacker could wait and learn the normal behaviour of the infected target, however, the authors prove how it is difficult for an attacker to figure out which specific features are the most representative of a user's writing style, since they vary from user to user.

Focusing on a more proactive approach Wendt [143] introduced Omen, a simple tool to help administrators in identifying potential spear-phishing targets before the email is sent. Using logs of abnormal visits to a company's website, the idea is to help the analyst find suspicious patterns faster than the attacker can craft a spear-phishing message. First they attempt to distinguish browser from crawler traffic, effectively separating and decreasing the amount of data that analysts have to look at. Then, with an interactive user interface, an administrator can see different user's visit patterns. Off course this is still very dependent on the skill of the person inspecting the logs and on them being able to distinguish between normal visits to the website and adversaries gathering information.

Just as we approached the problem in this work, other researchers also saw the criticality of the email infection vector and the problems people bring to the overall security of a system. It does not matter how good and secure your doors are if someone opens the windows. It is a challenging topic because of the inherent privacy issues, but also due to the requirement of categorizing benign and malicious emails. Will the system have access to the full email, or just meta data? Can the system open attachments or just check the hashes? Unfortunately, to be effective, the system would need to be intrusive. As we will show later in this work, email infection vector is a big part of APTs and if systems were able to detect and neutralize those emails, the adversaries would need to re-adapt.

2.4 Stopping data exfiltration

More oriented towards preventing cyber-espionage, other researchers accept the attacker's breach and try to detect or block data leaks. Parno et al. [110] propose CLAMP, an architecture that adds data confidentiality to the LAMP[2] model, preventing data leaks even in the presence of compromise, by enforcing strong access control while making few changes to application code. The architecture focuses on ensuring that private information can only be accessed by the respective owner, focusing on strong authentication rather than in confidentiality, and on isolating code ran by different users. By assigning an entire virtual web server to each user, all activity from that web server can be attributed to that particular user and isolated from others. Using virtualization requires increased hardware capabilities, but with only a 5-10 ms delay, the increased confidential data security justifies the costs.

Borders and Prakash [16] admit the impossibility of sensitive data leak detection and focus on measuring and constraining its maximum volume. By understanding that a large portion of legitimate traffic is repeated or constrained by protocol specifications, and ignoring this data, real information leaving the network can be isolated regardless of data hiding techniques. With a focus on the Hypertext Transfer Protocol (HTTP) and its interaction with Hypertext Markup Language (HTML) and Javascript, by computing the expected content of HTTP requests, the amount of unconstrained outbound bandwidth is the difference between actual and expected requests. Despite its limitations the scheme was able to contain the maximum volume of data leak to approximately 1.5% of the raw

[2] Linux, Apache, MySQL, Perl/PHP

request size. The leak would still be possible, but in very small portions. This might help other systems that would capture this type of behaviour as malicious and report on it.

This is another interesting way of looking at the APT problem. By acknowledging that the adversary will be successful in his initial infection, and that we will not be able to detect his presence or steps inside the network, we might be able to stop him at the exit, that is, by blocking data exfiltration, the most common APT objective. The adversaries will always adapt and if the normal exfiltration techniques stop working they might move to cloud services, peer-2-peer or physical methods. Nevertheless, data exfiltration blocking and/or detection is still a very interesting research topic with a lot of work still to do.

2.5 Protecting Critical Infrastructures

Although it is not the focal point of this work, we talk next about some work done on protecting critical infrastructures, defined by governments as essential assets for the correct functioning of society and economy. APTs most common objective is espionage, but different groups, such as governments, may use the same techniques to get a military advantage over their targets or simply just to cause chaos and destruction. Because of that, protection of critical infrastructures has seen an increase in research, specifically smart grids, thanks to their inherent importance in society and latest work developed on creating a more modern, connected grid. Lu et al. [91] presented a classification and evaluation of security threats on smart grids. Following the CIA[3] model, their categorization targets three types of attacks in terms of their goals, information confidentiality, data integrity and network availability. Since the highest priority in the smart grid is availability, the authors conducted an experiment to understand the feasibility of a denial of service attack and its consequences. The studied protocol proved to be quite resilient, the authors found that the performance does not degrade until the attack intensity index approaches one, i.e. the attacker is flooding more than 90% of the network bandwidth. It would be interesting to see similar research, but instead of brute-force flooding, injecting specifically crafted messages to try and exploit weak protocol definitions.

Skopik et al. [124] also did research on the smart grids, particularly how they deal with APTs. By checking the logs and using a white-list based anomaly detection approach in the SCADA backends, due to their restrictive and predictable behaviour, the authors believe that APT attacks can be detected. The system successfully detected an anomaly in a SCADA operation, since an event occurred without a corresponding firewall entry. The framework looks promising, although it would be curious to see it preform under several different circumstances and network layouts.

Finally, Bessani et al. [10] presented CRUTIAL, an intrusion-tolerant and self-healing

[3] The triad of Information Security: Confidentiality, Integrity and Availability.

information switch designed to protect critical infrastructures, while preserving legacy systems. By building a WAN of LANs, it becomes possible to define realms with distinct trustworthiness levels, focusing the problem on protecting realms from one another. The switch itself uses a rich access control model, is intrusion-tolerant and can be installed to resemble a distributed firewall. The authors propose four different deployment types depending on the criticality of the service and the corresponding performance requirements. Some open problems remain, one of them is how to keep the communication infrastructure resilient at lower costs.

As we can see from previous research, Advanced Persistent Threats are a never ending topic, from understanding the groups behind them and what reasons move them to develop such offensive capabilities, to learning the true consequences, both socially and economically, of successful attacks. This has driven researchers to work on blocking such threats and their most prevalent infection vectors by developing frameworks and systems capable of dealing with previously unknown malware, tactics and techniques. Later on in this work, after we have understood the APT problem, we will add to this discussion our ideas of standalone systems and systems that working together will help on the prevention and detection of APTs.

Chapter 3

A study on Advanced Persistent Threats

3.1 What is an Advanced Persistent Threat

Advanced Persistent Threat is agreed in the security community to be a term coined by the United States Air Force (USAF) around 2006, in a small meeting room, as a way to talk about classified intrusions with uncleared personnel [142, 46]. Explained by a former USAF Intelligence Officer, Richard Bejtlich[1], ”**Advanced** means the adversary can operate in the full spectrum of computer intrusion. They can use the most pedestrian publicly available exploit against a well-known vulnerability, or they can elevate their game to research new vulnerabilities and develop custom exploits, depending on the target's posture. **Persistent** means the adversary is formally tasked to accomplish a mission. They are not opportunistic intruders. Like an intelligence unit they receive directives and work to satisfy their masters. Persistent does not necessarily mean they need to constantly execute malicious code on victim computers. Rather, they maintain the level of interaction needed to execute their objectives. **Threat** means the adversary is not a piece of mindless code. This point is crucial. Some people throw around the term ”threat” with reference to malware. If malware had no human attached to it (someone to control the victim, read the stolen data, etcetera), then most malware would be of little worry (as long as it didn't degrade or deny data). Rather, the adversary here is a threat because it is organized and funded and motivated. Some people speak of multiple groups consisting of dedicated crews with various missions.”

Another interesting remark from Bejtlich: ”Too many critics focus on malware, ignoring (or being unaware) of the impressive management and administration applied to repeatedly attempting to access, or preserving access to target organizations. APT incidents are not hit-and-run, smash-and-grab affairs.”

It was only around 2010, when Google revealed that it was a victim of a so called APT attack and security companies started selling anti-APT systems, that the term got a

[1] `http://taosecurity.blogspot.pt/2010/01/what-is-apt-and-what-does-it-want.html` Accessed 02-February-2015

commercial connotation. As Kaspersky puts it[2], ”There are two ways to look at it: APT as a thing and APT as people. On the one hand, an advanced persistent threat refers to a highly precise sort of cyberattack. On the other hand, advanced persistent threat can also refer to the groups, often state sponsored or well-funded in other ways, that are responsible for launching such precision attacks.” As for Symantec[3],”An APT is a type of targeted attack. Targeted attacks use a wide variety of techniques[...]. APTs can and often do use many of these same techniques. An APT is always a targeted attack, but a targeted attack is not necessarily an APT.” Mandiant[4] on the other hand ”defines the APT as a group of sophisticated, determined and coordinated attackers that have been systematically compromising U.S. government and commercial computer networks for years.” And finally, the definition according to the National Institute of Standards and Technology [101], ”an adversary that possesses sophisticated levels of expertise and significant resources which allow it to create opportunities to achieve its objectives by using multiple attack vectors (e.g., cyber, physical, and deception). These objectives typically include establishing and extending footholds within the information technology infrastructure of the targeted organizations for purposes of exfiltrating information, undermining or impeding critical aspects of a mission, program, or organization; or positioning itself to carry out these objectives in the future. The advanced persistent threat: (i) pursues its objectives repeatedly over an extended period of time; (ii) adapts to defenders' efforts to resist it; and (iii) is determined to maintain the level of interaction needed to execute its objectives.” For the interested reader, [23] has a very good, up-to-date research on the topic, [126] explains the different components and techniques in a successful attack and [1] has a selection of APT reports which is regularly updated.

3.1.1 How we see an Advanced Persistent Threat

It is easy to see how an APT is a sophisticated type of targeted attack. They can either be highly social engineered or highly technical. NSA, for example, makes use of sophisticated tools that have a success rate as high as 80% [129] with little to no use of social engineering. This includes zero day vulnerabilities, exploiting operating systems or applications and attacks on hardware, such as hard drive firmware or Basic Input Output System infection [144, 21]. Unfortunately we now know things do not stop here, we are now aware of attacks on global communications and the weakening of cryptographic algorithms for espionage reasons. We will not be tackling these kind of attacks in our proposed frameworks for lack of time and capacity to approach the problem in the detail it deserves. Instead we will focus our efforts in the less technical attacks that make bigger

[2] `http://blog.kaspersky.com/apt/` Accessed 02-June-2015

[3] `http://www.symantec.com/content/en/us/enterprise/white_papers/b-advanced_persistent_threats_WP_21215957.en-us.pdf` Accessed 02-June-2015

[4] `https://dl.mandiant.com/EE/assets/PDF_MTrends_2010.pdf` Accessed 02-June-2015

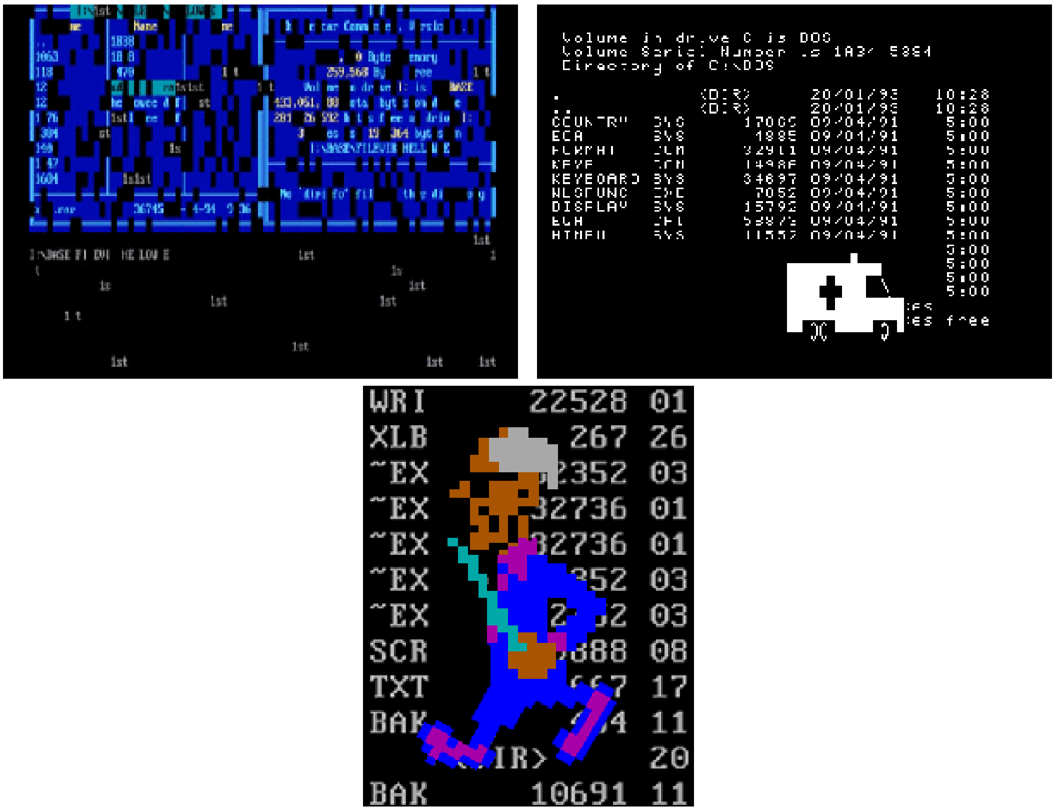
use of the social engineering vector. Social engineering is still a very effective intrusion method and Mitnick has a very interesting book on the topic [96]. The attacks might still make use of sophisticated techniques but are prevalently accompanied by some sort of social engineering to trick the victim into doing something to help the attacker.

In this work we will try to approach the problem from a different perspective, our definition of an APT is similar but not equal to its original term. We agree with Bejtlich when he says that many people focus on the malware part, so we will look at an APT as a campaign, as a process, a continuous action in time. What makes it advanced, is not only the spectrum of computer intrusion or the sophistication of the malware, but what goes behind the scenes, i.e., the infrastructure, management, teams, budget, and eventual government involvement, before, during and after the objective is completed. As for the persistence, we agree with the original definition. The objective is divided into smaller problems, each assigned to their specific team. No team will stop until they succeed with their mission. What makes it a threat is the human factor. The tools used by these groups are not your typical worm or virus, they rely on their masters input and this human interaction is what makes these attacks a real threat.

A recent cyberattack platform called Regin [72] is a great example of such a threat. Discovered in 2012 with samples dating as far back as 2003, it is still unclear how initial compromise was conducted. Undetected for almost ten years and affecting only 27 different victims such as telecom operators, government institutions and advanced mathematical/cryptographic researchers, the adversaries objectives were intelligence gathering and facilitation of other types of, highly targeted, attacks. The implementation, coding methods, hiding techniques and flexibility make Regin one of the most sophisticated attack platforms of recent times. The amount of resources, organization and expertise depicted in such an attack made researchers point at a possible nation-state sponsor. A few years later, the tool was linked to the National Security Agency [117].

3.2 What makes it different?

How are these attacks different from what we see and hear on the news every now and then? Old virus are not a problem any more, you used to know you were infected because you could actually see what the virus did, such as all of the console letters falling down the screen or an ambulance passing across the desktop, as shown in Figure 3.1, these were viruses made by hobbyist motivated by curiosity and intellectual challenges. Unfortunately things changed for the worse, attackers started to go after fame and recognition, these are the attackers that will show you a big *you have been hacked* message, deface your website and leak your database. Together with this development, a new form of organized crime emerged, focused on the digital world. Motivated to make easy and quick money, cyber-criminals infected millions of machines and continued, to this day, working



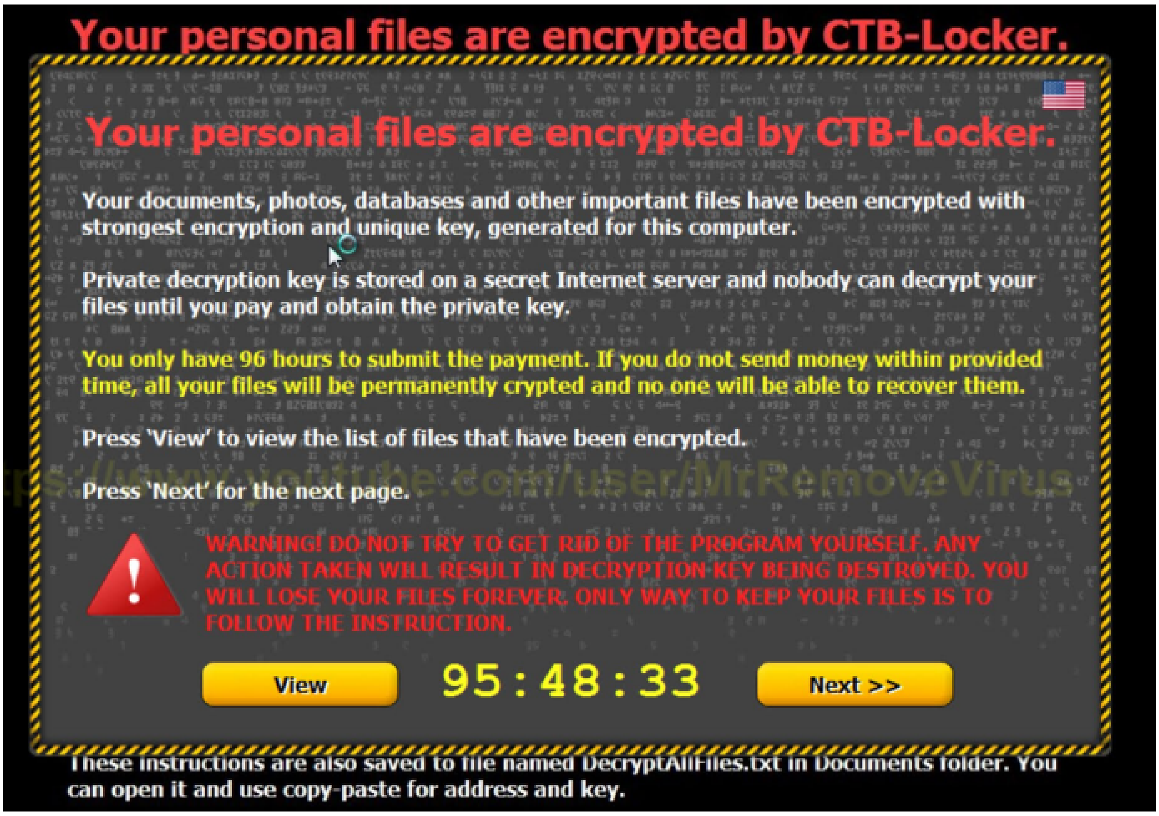
```
Volume in dr.ve C is DOS
Volume Seric: Number .s 1A3/ 5864
Directory of C:\DOS

.              <DIR>       20/01/93   10:28
..             <DIR>       20/01/93   10:28
COUNTRY   SYS      17065  09/04/91    5:00
EGA       SYS       4885  09/04/91    5:00
FORMAT    COM      32911  09/04/91    5:00
KEYB      COM      14986  09/04/91    5:00
KEYBOARD  SYS      34697  09/04/91    5:00
NLSFUNC   EXE       7052  09/04/91    5:00
DISPLAY   SYS      15792  09/04/91    5:00
EGA       CPI      58873  09/04/91    5:00
HIMEM     SYS      11552  09/04/91    5:00
                                      5:00
                                      5:00
                                      5:00
                                      5:00
                            es
                            es free
```

```
WRI      22528 01
XLB        267 26
~EX       2352 03
~EX      32736 01
~EX      32736 01
~EX        352 03
~EX      2  2 03
SCR       888 08
TXT       7 17
BAK        4 11
   IR>       20
BAK      10691 11
```

Figure 3.1: Examples of infections

on new forms of making cash. From ransom-ware [26], like the one seen in Figure 3.2, to creating and selling malware toolkits [27], building your own botnet [3], and off course the typical credit card frauds [80] and personal information harvesting.

Things are not quite the same for an APT hacker. He has evolved. For starters he does not want to be known or found, he does not want media attention or fame, he wants to remain in the shadows for as long as possible. He is not looking for quick monetary gains, he is looking for information, he is looking to control the targeted infrastructure, making sure he always has another way in. He will study your organization, your employees and your assets, he will make such use of social engineering tactics that the email you receive will look just like any benign day-to-day email. Even compared against the typical botnet masters, the malware used here is relatively different. A zombie would not be looking for SSH credentials, valid root certificates or specific supervisory control and data acquisition systems. An example of such differences can be observed in Table 3.1. A message by the "Honker Union of China", a very famous hacker group in the country, in 2010 to its members says: "What benefit can hacking a Web page bring our country and the people? It is only a form of emotional catharsis, please do not launch any pointless attacks, the real attack is to fatally damage their network or gain access to their sensitive information" [73],



Your personal files are encrypted by CTB-Locker.

Your personal files are encrypted by CTB-Locker.

Your documents, photos, databases and other important files have been encrypted with strongest encryption and unique key, generated for this computer.

Private decryption key is stored on a secret Internet server and nobody can decrypt your files until you pay and obtain the private key.

You only have 96 hours to submit the payment. If you do not send money within provided time, all your files will be permanently crypted and no one will be able to recover them.

Press 'View' to view the list of files that have been encrypted.

Press 'Next' for the next page.

WARNING! DO NOT TRY TO GET RID OF THE PROGRAM YOURSELF. ANY ACTION TAKEN WILL RESULT IN DECRYPTION KEY BEING DESTROYED. YOU WILL LOSE YOUR FILES FOREVER. ONLY WAY TO KEEP YOUR FILES IS TO FOLLOW THE INSTRUCTION.

These instructions are also saved to file named DecryptAllFiles.txt in Documents folder. You can open it and use copy-paste for address and key.

Figure 3.2: Example of ransom-ware

it is clear the shift in tactics, procedures and objectives even in the hacking community.

Typical Botnet Malware objectives	APT specific objectives
Capture e-mails	Read local data
Capture passwords	Record audio
Capture credit card numbers	Record video
Use machine for spamming	Safe and stealthy exfiltration
Use machine for DDoS	Self-termination

Table 3.1: Example of different malware characteristics

3.3 Analysing Advanced Persistent Threat Life cycles

There are several proposed attack life cycles that enumerate each and every step of a campaign. Mandiant, in Figure 3.3, Palo Alto Networks, in Figure 3.4, as well as Wrightson [147] in Figure 3.5 and ZScaler in Figure 3.6 proposed new and different cycles specifically for APT campaign categorization.

Although data exfiltration continues to be the most common APT objective, and is used in all the life cycles, both Palo Alto and Mandiant include a different phase, called destruction or complete mission, to better help us understand that the attacker might be

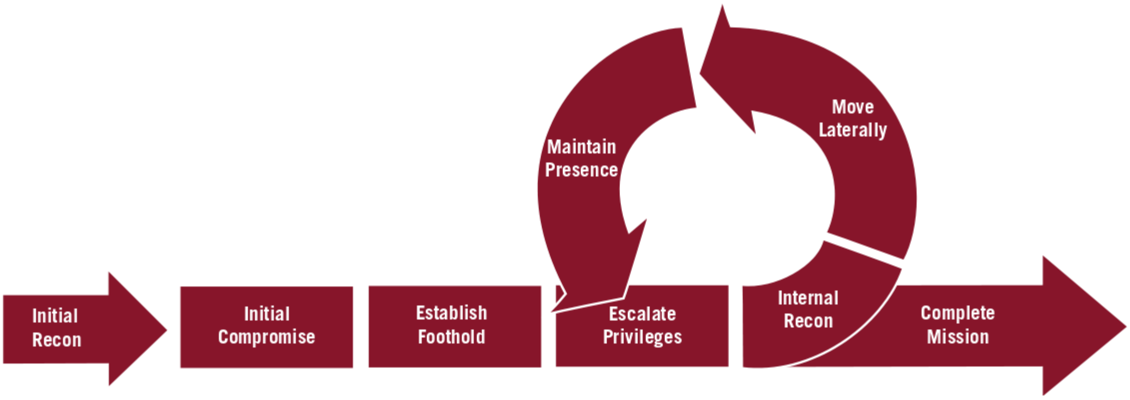


Figure 3.3: APT Lifecycle proposed by Mandiant

more interested in shutting down or disrupting a system, like data wiping, disabling critical infrastructures or flooding of communication channels.

These life cycles are quite generic and use different names for steps representing the same actions, for example, exploitation and initial compromise or progression and lateral compromise. The idea stays the same, the adversary will study your organization, will target specific people and/or services, exploit them and complete its objective. With that in mind, we will tackle the cycle proposed by DELL SecureWorks, depicted in Figure 3.7, since it is built with several more phases, giving us the chance to understand them better as a whole and individually. Using the chosen life cycle as baseline, a longer description for each phase and some critiques are given next.

Define objectives and targets

First, the attackers will have to define, or receive from their superiors, an objective. This could be a short term objective or long term goal. What exactly do they need? Usually they want access to information (software code, government strategies, academic research and others), but they might also want to shut down a country's electrical grid or communication channels. With clear objectives defined, now they need to choose their targets. By targets we mean people, organizations or governments holding the necessary information for the adversary to be successful. This can be a low hanging fruit, like an unprotected third party working for a highly secured company, or a nation electricity provider.

Find and organize accomplices

The sophistication and work necessary to successfully carry out an attack like this make it impossible for one person only. Although this type of attackers usually already have organized teams of skilled developers, experienced testers, and are sometimes backed up by states [44, 92], different hacking groups can merge to tackle more sophisticated targets. And if a team can not be created, there is always the possibility of a water hole [146] attack, where the attackers would only need to

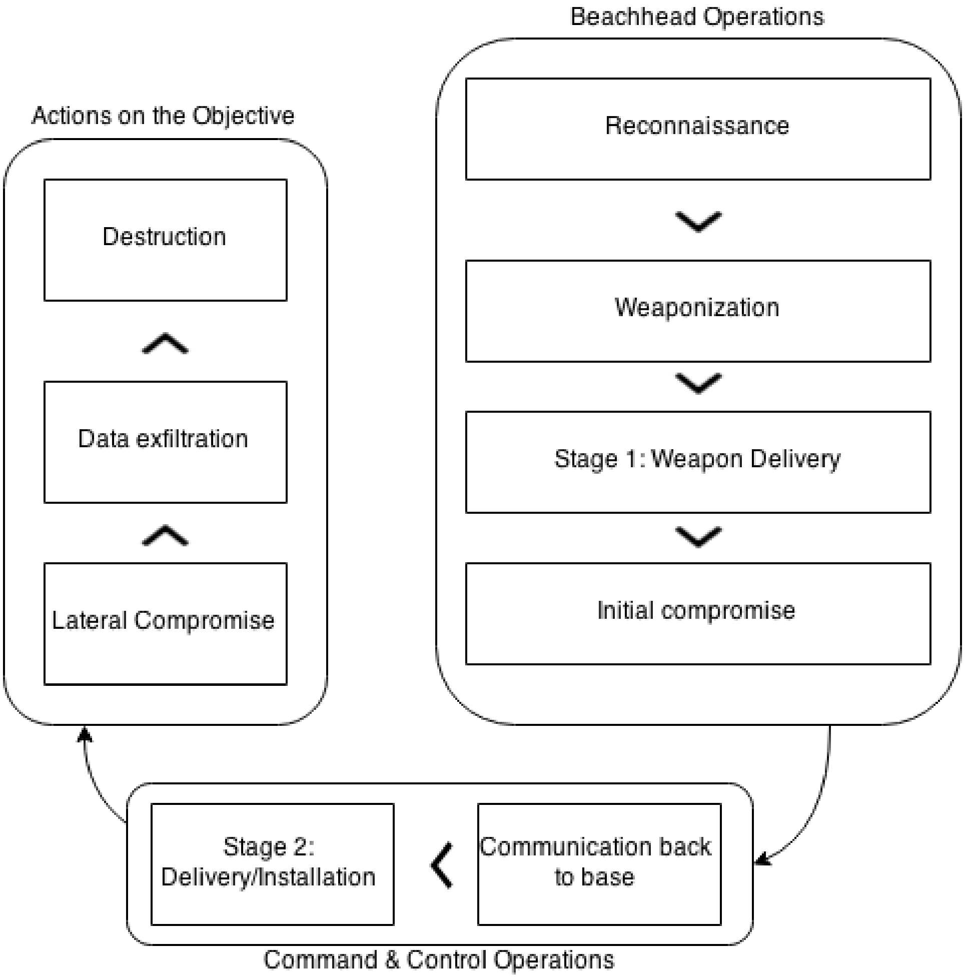


Figure 3.4: PaloAlto Networks APT Lifecycle

choose which website to target, infect it and wait for the target to visit it. If physical penetration is required, insider attacks [48] are a viable option, such as paying a soon to be ex-employee to release the malware in the target network. Throughout our work we will be referring to insider threats or attacks following the definition in Silowash et al. work [122]:

”...is a current or former employee, contractor, or business partner who meets the following criteria:

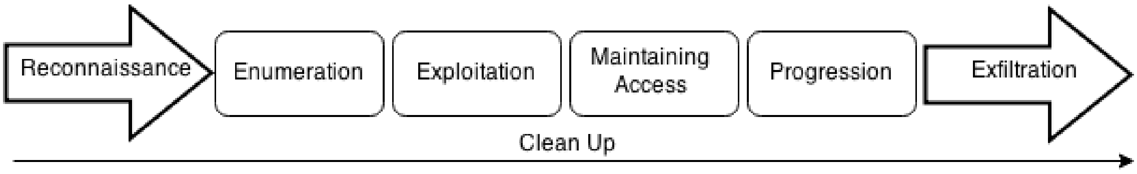


Figure 3.5: APT Lifecycle proposed by Wrightson

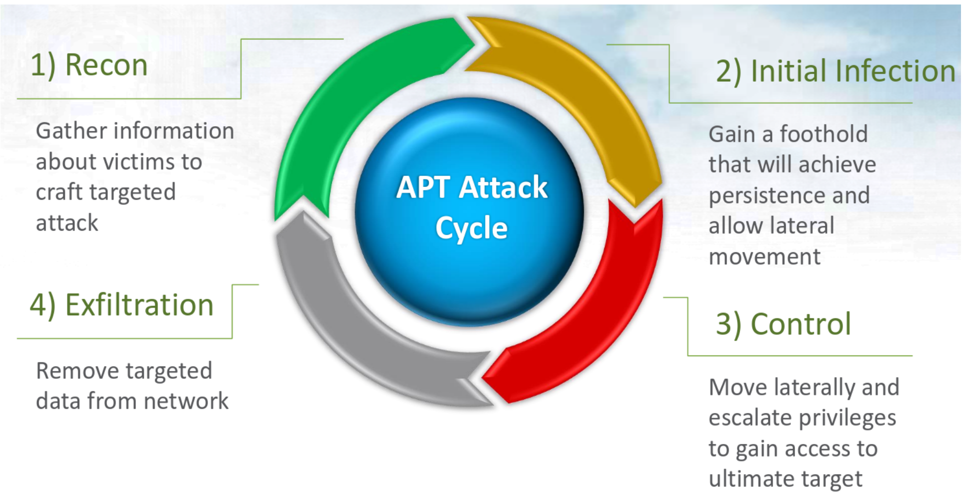

1) Recon
Gather information about victims to craft targeted attack
2) Initial Infection
Gain a foothold that will achieve persistence and allow lateral movement
APT Attack Cycle
4) Exfiltration
Remove targeted data from network
3) Control
Move laterally and escalate privileges to gain access to ultimate target

Figure 3.6: APT Lifecycle proposed by ZScaler

- has or had authorized access to an organization's network, system, or data;
- has intentionally exceeded or intentionally used that access in a manner that negatively affected the confidentiality, integrity, or availability of the organization's information or information systems."

Build or acquire tools

Unless the tools the attackers are building/acquiring are going to be used in the reconnaissance phase, weaponizing without knowing the target capabilities would be a waste of resources, so this should be done after the research phase. Custom tools have a lower chance to be detected, and are still the preferred choice of attackers. Add to that a zero-day vulnerability and the attack will most likely be very successful [40, 81, 66]. It is worrying that some systems still fall to the use of openly available remote access trojans [43, 92] exploiting known vulnerabilities [114, 136].

Research target infrastructure/employees

This is the well known reconnaissance/information gathering phase, which can take months. This can be passive, like open-source intelligence (search engines, social media, etcetera), which involves no direct interaction between the attacker and the target, or active, like phone-calls, which will involve direct contact and leaving traces. It will be done before and after intrusion. Things like infrastructure, services and their software versions, employed security, vulnerable systems or people, and emails, are what the attackers will be looking for. Tyler Wrightson in his book [147] has several chapters dedicated to the techniques used by hackers during reconnaissance, so interested readers are encouraged to check the book. In a nutshell, Wrightson organizes reconnaissance like the following (with examples from the book):

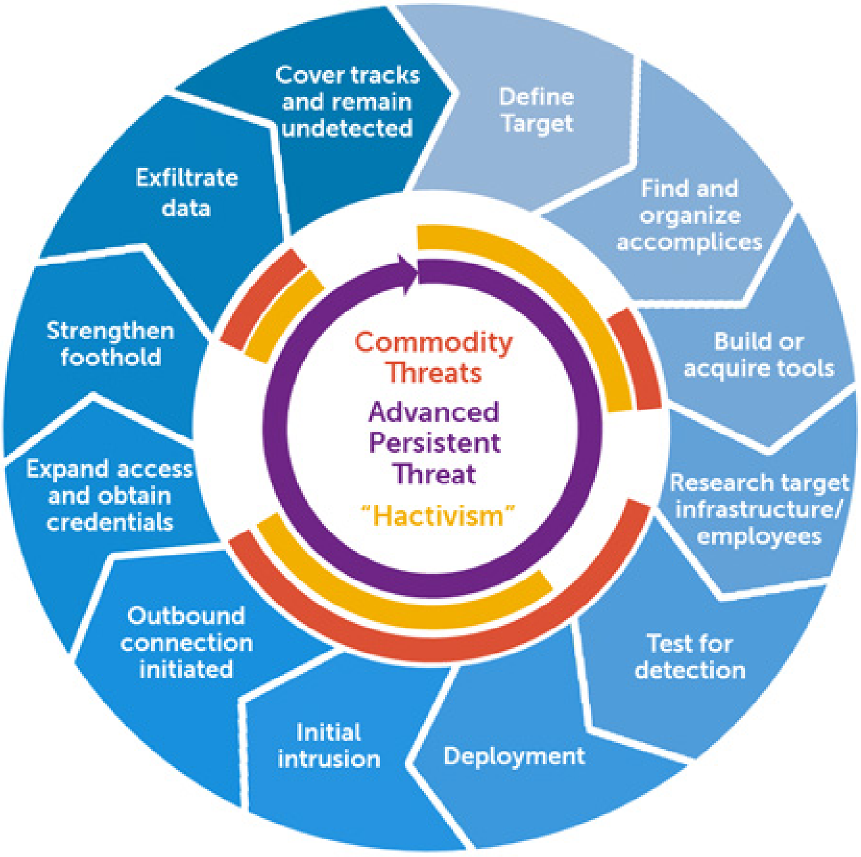

Define Target
Find and organize accomplices
Build or acquire tools
Research target infrastructure/ employees
Test for detection
Deployment
Initial intrusion
Outbound connection initiated
Expand access and obtain credentials
Strengthen foothold
Exfiltrate data
Cover tracks and remain undetected
Commodity Threats
Advanced Persistent Threat
"Hactivism"

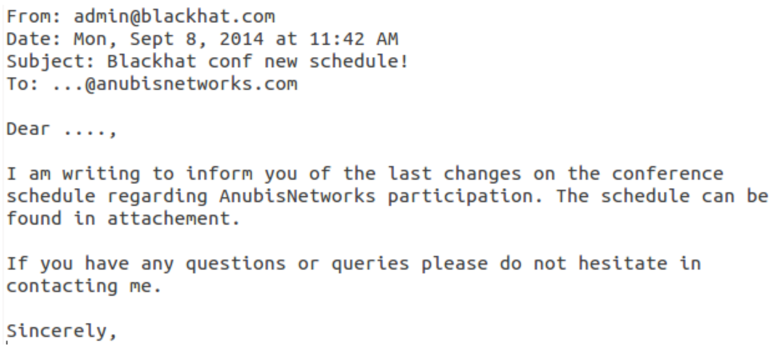
Figure 3.7: DELL SecureWorks APT Lifecycle

1. Technical reconnaissance
   (a) Anti-virus software;
   (b) Internet-routable subnets.
2. Non-technical reconnaissance
   (a) Geographical locations;
   (b) Important personnel.

Test for detection

Although this type of testing can be done online, attackers do not want to raise any alarms and make their target suspicious. Attackers are known to mimic their victims environments and test their tools offline. Several APT C&C servers were found with virtual machines set up like a mirror of their respective victims. STUXNET [40] is believed to have mimicked their target with meticulous detail, things like software and their versions, hardware and PLC's were acquired and installed for live testing.

Deployment/Delivery and Initial intrusion



```
From: admin@blackhat.com
Date: Mon, Sept 8, 2014 at 11:42 AM
Subject: Blackhat conf new schedule!
To: ...@anubisnetworks.com

Dear ....,

I am writing to inform you of the last changes on the conference
schedule regarding AnubisNetworks participation. The schedule can be
found in attachement.

If you have any questions or queries please do not hesitate in
contacting me.

Sincerely,
```

Figure 3.8: Example of a spear phishing email

The problem here is how to deliver the tools. Usually, giving use to the extensive recon phase, attackers like to use spear-phishing, or targeted malicious emails [133, 109, 71] to deliver malicious payload. These are not your typical spam mail, they are highly personalized, make use of topics of interest and may include the victims name while pretending to come from a friend. An example of such an email is given in Figure 3.8. A targeted email was sent after AnubisNetworks publicized their presence at the BlackHat[5] conference, and together with the email was a malicious payload ready to be delivered. This payload will most likely exploit a vulnerability, maybe a zero-day, and in nature needs to be of limited size to be able to inject the necessary code. These are usually generic and will download, after exploitation, the custom built tools from the C&C and are therefore called droppers. The emails are personally tailored to entice users into opening attachments or visiting websites. These emails are so personal that even security conscious users sometimes fall for them[6]. Even if this approach does not work, candy drops (USB drives, DVD's and even mail packages sent to the target address) or physical access (impersonating cleaning crew or bribing employees) are other possible options. Not surprisingly, and especially thanks to water hole attacks, [45] reports that in 2013 the average number of Web-derived attacks was over three times higher than email-derived attacks. You may be reluctant to follow a link to an unknown website, but what if it is a trusted one? Initial intrusion happens if the delivery is successful. At this point the malware is executed and installed, the attacker has control of the infected machine.

Outbound connection initiated

This is the typical connection check. Most malware will check first for a legitimate online connection (e.g. GET google.com), and only after will initiate C&C conver-

[5] `https://www.blackhat.com/` Accessed 02-June-2015

[6] In Mandiant APT1 report [92]: In one case a person replied, "I'm not sure if this is legit, so I didn't open it." Within 20 minutes, someone in APT1 responded with a terse email back: "It's legit." The person then opened the attachment.

sation. What they do here differs from malware to malware, some will only try to get updates, others will sleep and wait until a specific time or until they have orders, and some will exfiltrate machines/network information. This is the main communication channel between attacker and target organization, therefore, the most vulnerable phase to detection. The attacker has to be very careful about what monitoring systems are in place, and use protocols, transmission times and message sizes that avoid detection.

Obtain credentials, expand access and strengthen foothold

According to [2], 90% of binaries morph within one hour. With access to the network, attackers will want to obtain new tools, gather credentials and expand their access to other machines using techniques like pass-the-hash [61]. Why? Well, maybe the machine they got infected is not the one they wanted, at least not yet. They will also want to have other doors to get inside the network in case one closes. And they need to do all this while covering their tracks and avoid detection.

Exfiltrate data/Complete objective

Finally, attackers complete their goals, may it be data exfiltration, money theft or the destruction of infrastructures. This is not usually done in a one-time-hit, but overtime, with low impact to avoid suspicion and, most importantly, detection.

Cover tracks and remain undetected

In reality this is not a one time phase, this is an every-phase phase, i.e., attackers will want to remain undetected for the whole campaign, making sure to delete old files and logs. On one hand, some code will try to make their traffic look as legitimate as possible, if no flags are raised there is no need to hide anything, and if there is nothing being hidden then it must be legitimate. On the other hand, some malware will automatically delete itself after initial infection, wipe the whole machine [132] and slow down or stop itself if forensic tools are detected.

3.4 A new APT model representation

The previous life cycles have the problem of not grasping the complexity inherent to an advanced persistent threat attack campaign. Therefore, in order to improve this, we propose a new way of representing the teams, objectives and usual procedures of an APT group. This new model, based on all the available literature and knowledge of APT groups and techniques merged in two simple pictures is shown in figures 3.9 and 3.10. With this representation we hope to help the security community in understanding better the complexity in orchestrating, managing and executing a campaign like this. One of the biggest differences is that previous life cycles focused too much on the cycle and how a step leads to another. In this model the focus is on how different teams work on different phases that evolve in **parallel** towards a common goal. By showing what kind of teams are working in these groups and what type of tasks they are assigned, we hope to better fundament our view of how APTs are more than a piece of sophisticated malware.

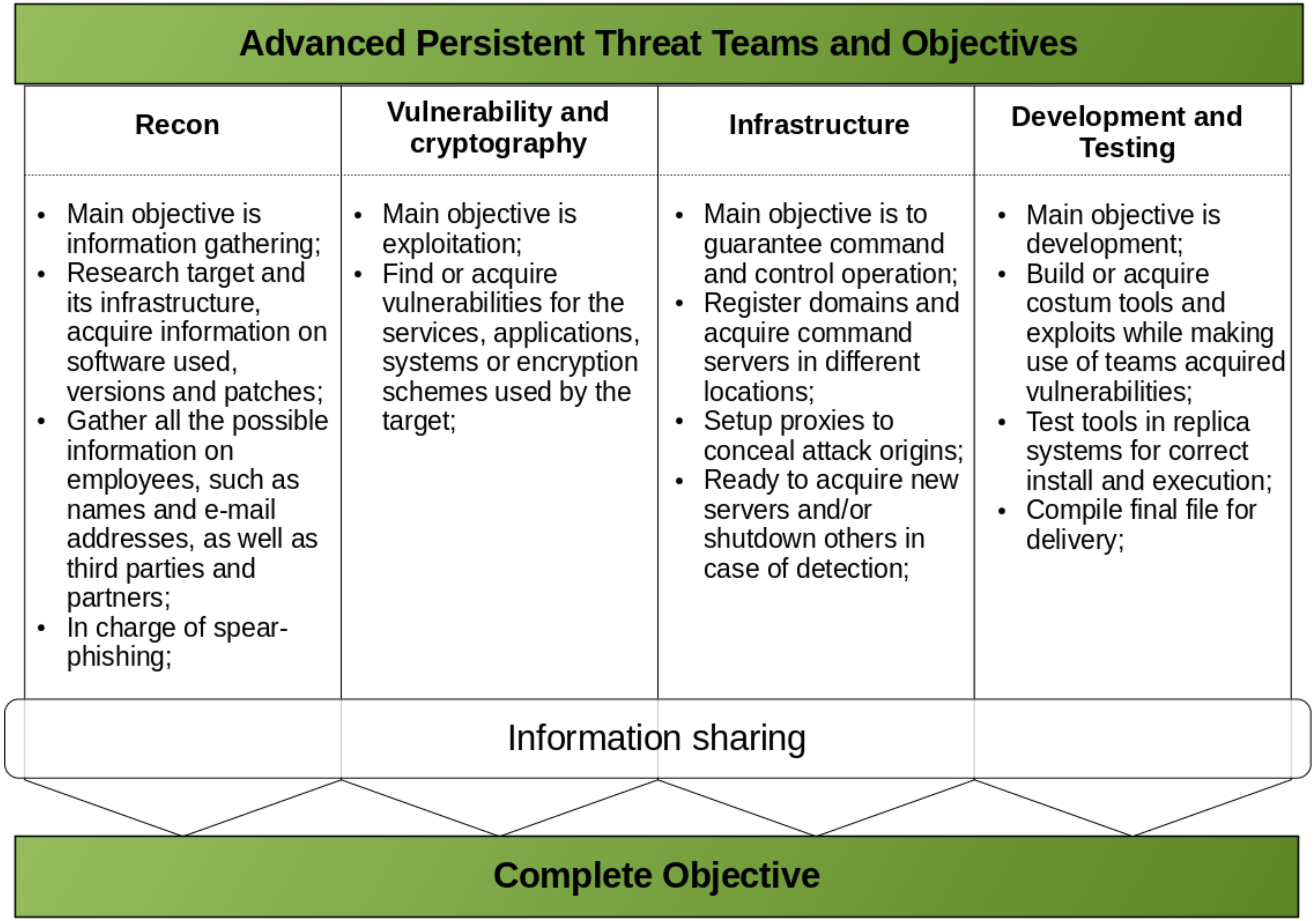
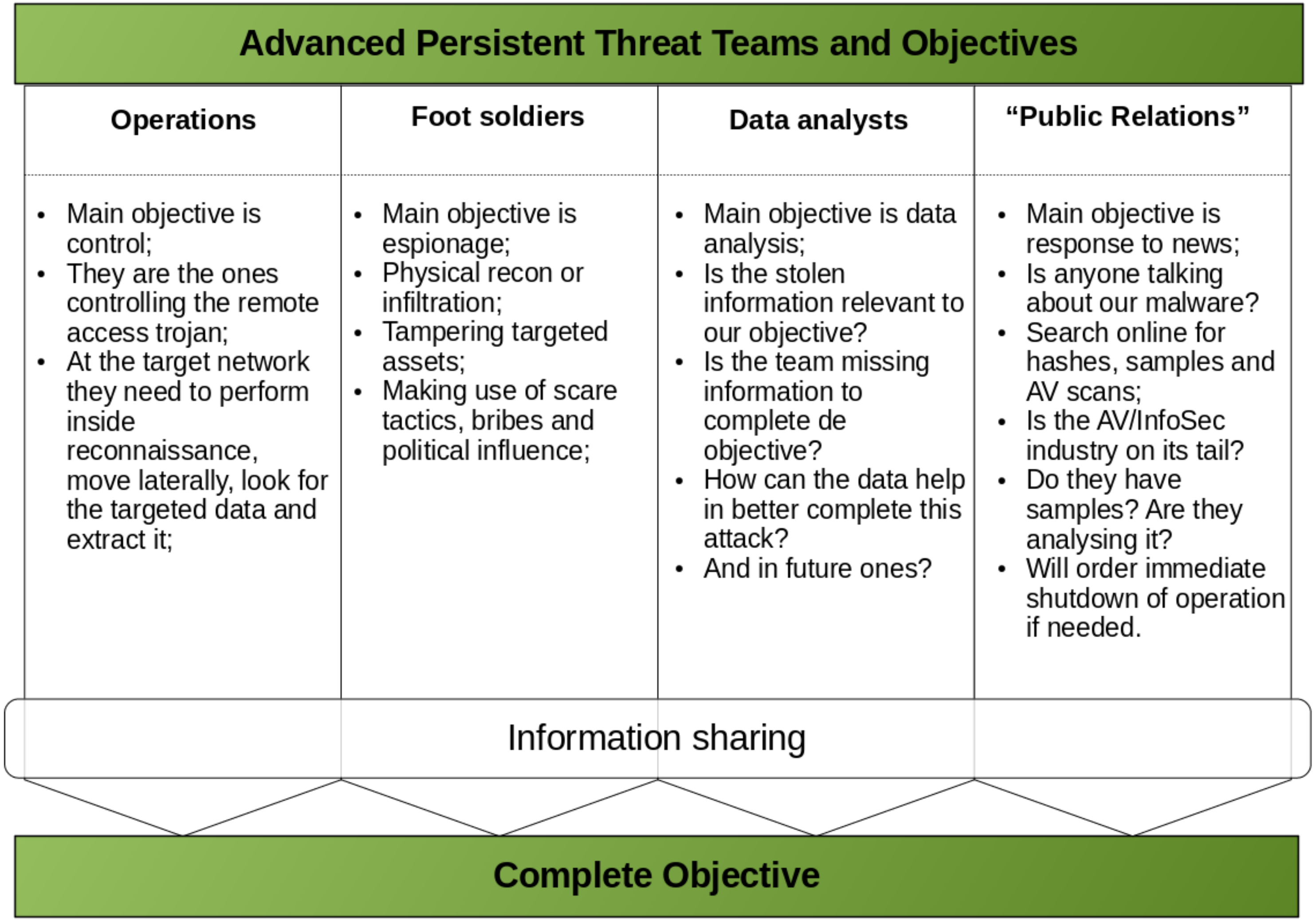

Advanced Persistent Threat Teams and Objectives
Recon
Main objective is information gathering;
Research target and its infrastructure, acquire information on software used, versions and patches;
Gather all the possible information on employees, such as names and e-mail addresses, as well as third parties and partners;
In charge of spear-phishing;
Vulnerability and cryptography
Main objective is exploitation;
Find or acquire vulnerabilities for the services, applications, systems or encryption schemes used by the target;
Infrastructure
Main objective is to guarantee command and control operation;
Register domains and acquire command servers in different locations;
Setup proxies to conceal attack origins;
Ready to acquire new servers and/or shutdown others in case of detection;
Development and Testing
Main objective is development;
Build or acquire costum tools and exploits while making use of teams acquired vulnerabilities;
Test tools in replica systems for correct install and execution;
Compile final file for delivery;
Information sharing
Complete Objective

Figure 3.9: APT Model Representation



Advanced Persistent Threat Teams and Objectives

Operations

- Main objective is control;
- They are the ones controlling the remote access trojan;
- At the target network they need to perform inside reconnaissance, move laterally, look for the targeted data and extract it;

Foot soldiers

- Main objective is espionage;
- Physical recon or infiltration;
- Tampering targeted assets;
- Making use of scare tactics, bribes and political influence;

Data analysts

- Main objective is data analysis;
- Is the stolen information relevant to our objective?
- Is the team missing information to complete de objective?
- How can the data help in better complete this attack?
- And in future ones?

"Public Relations"

- Main objective is response to news;
- Is anyone talking about our malware?
- Search online for hashes, samples and AV scans;
- Is the AV/InfoSec industry on its tail?
- Do they have samples? Are they analysing it?
- Will order immediate shutdown of operation if needed.

Information sharing

Complete Objective

Figure 3.10: APT Model Representation (part 2)

3.5 Profiling attackers

Understanding who is doing these attacks and how they are doing it is crucial. Although the lack of written work is not surprising, it is certain that both parts of the world are in a constant cyber-battle. Unfortunately most of the literature is about Eastern groups attacking Western countries. A good example of such a group is Hidden Lynx [36]. Based in China, with fifty to one hundred operatives and active since at least 2009, they have used three zero day attacks since 2011 and are known for their organization, agility, resourcefulness and pioneering ”watering hole” attacks to ambush their targets. The attacks are so advanced and targeted that they would certainly need research and intelligence gathering prior to any infection.

Filippoupolitis et al. [42] provide us with a good list of features to profile a cyber-attacker and we will make use of this list to profile a typical APT adversary following the information gathered related to the groups and their operatives:

Skill Highly skilled attackers, typically had access to computers from an early age and developed an interest in hacking or similar activities. Some are expert developers while others are cryptography geniuses.

Education Usually with higher education and proficient English, these attackers have the technical and theoretical skills needed to make part of a larger attack team. Most recently, the groups have started recruiting at the Universities.

Risk As they are usually nation sponsored they are able to operate free from government laws. These activities have a very low risk for attackers and states since they also offer plausible deniability. If they are not state sponsored, they know how to minimize their risks and mitigate possible repercussions.

Goal The goal, usually, is to exfiltrate data. Some are more cyber-warfare oriented and might want to disrupt or destroy targeted systems, with very few examples of actual money theft.

Speed As we have seen before, 90% of binaries morph within one hour of infection. These attackers are ready and they are fast. As soon as the infection is successful, they take control.

Mistakes They do make mistakes, typos/bugs in the code and they might leave traces. After all, they are human. Mandiant reported that one APT1 hacker was lost for a long while trying to figure out a console command[7]. It is usually these mistakes that lead security companies to their attributions.

[7] `http://www.youtube.com/watch?v=6p7FqSav6Ho` Accessed 15-October-2014

Anti-forensics Some of the malware reported employs anti-forensics techniques. Although some make use of the usual dummy functions, variables and loops in the code, others try to detect virtualization and even use run-time decoding in memory.

Success These attacks are very successful and we, the general public, just know about the reported ones, caught months or even years after initial intrusion. It is not so hard to believe that there are a lot of undiscovered attacks, or undisclosed ones, and new ones happening every day.

3.6 Why traditional defences do not work?

As Osorio et al. show in their work [106], ”the old idea of measuring the number of infected files detected within end-point device as a good measure of their effectiveness has become obsolete. Instead, measures such as time to detect, time to countermeasure issuance, and ability to identify short-lived C&C sites seem more relevant to determining the ’goodness’ of security products”. As we will see next, current security systems make heavy use of lists with known malware or attacks and only work on detecting those, and that is why they fail to stop more advanced, custom and modern threats.

Firewalls. Most of this type of attacks make use of the HTTP/HTTPS protocols, on ports 80 and 443, respectively, which are usually open in the firewall. Even if there is any sort of inspection, encryption will render it useless. At the application level, WAF’s or Web Application Firewalls, have been proven to fail even against known attacks [60].

Intrusion Detection Systems (IDS). There are three different approaches: *a*) signature-based or misuse or rule based detection, that compare data against a set of attack signatures. Because of this a lot of work is spent on updating these systems, and although known attacks can be detected reliably with a low false positive rate, it will not detect custom/new attacks with some research showing that they can even be bypassed [24]; *b*) anomaly detection, that, like the name says, looks for non-normal behaviours and therefore does not need to be kept up-to-date on malware signatures. This type of systems are better at detecting novel attacks and *may* be a contribution in the fight against APTs, although, as we have seen before, a lot of work is still needed in this field. One of the challenges regarding APTs is that the adversary will study its target and will learn what is normal behaviour in the target. If the attacker knows the company uses cloud service *x* to upload documentation, he might pass undetected and accomplish extraction making use of the same service. Another major setback in this technology is that they still suffer from large numbers of false positives, large computational need and the lack of datasets for testing [50]; and *c*) hybrid, which combines the best of both worlds. Although intrusion detection systems are not in the scope of our study, interested readers are advised to read [86] which has a good IDS review, covering signature based, anomaly based and stateful analysis, as well as [119] which has a good survey on the techniques used on these systems, and [125] which shows how flawed in their most basic assumptions anomaly detectors are.

Anti-virus. Also signature based and therefore need to be kept up-to-date to be effective. In 2014 there was on average one hundred and sixty thousand new malware samples per day [53], with anti virus engines needing to update their rules several times a day. A late update might dictate a successful infection, as such, anti virus offer needed protection against known malware and some vendors make use of heuristics and emulation to detect unknown malware. Nevertheless, custom code and the use of any sort of morphism might render this defence mechanism useless.

People. Insider attacks, unintentional or not, are still a big problem. People are always the weakest link, it doesn't matter how good and secure your system is if your Chief Security Specialist will gladly give you SSH access and kindly reset your password [18]. Companies are still not investing enough in security awareness and training for their personnel which makes them vulnerable to social engineering attacks.

Chapter 4

A survey on Advanced Persistent Threat attacks

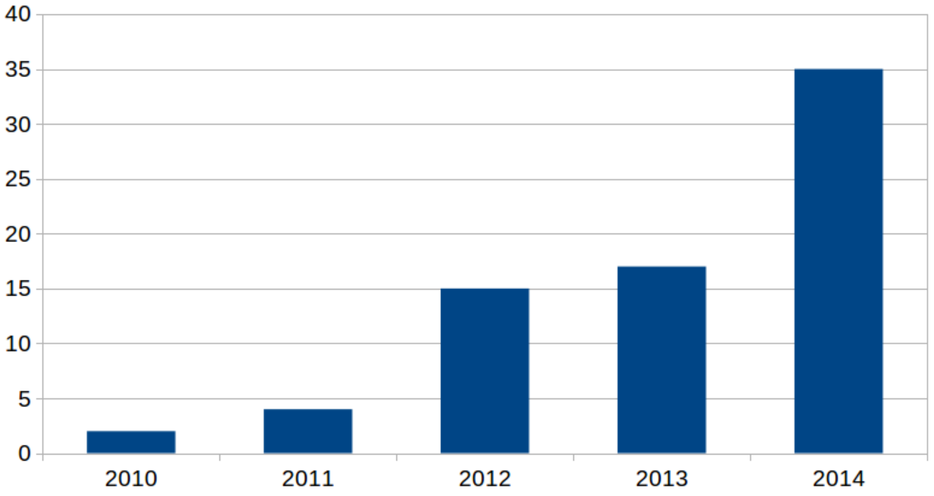


Figure 4.1: The number of APT reports per year

When researching on the topic of APTs a lack of meaningful statistics turned out to be a problem. If there was ever a need to represent how much of these type of attacks made use of phishing or how many used exploits abusing zero-day vulnerabilities, it was not possible, the data was just not there, at least in an aggregated way. The purpose of this survey is not only to fix that, but also to answer other emerging questions, such as, what the most common infection vector is, which are the most targeted applications and who are the most active actors.

A survey on the available reports from 2010 to 2014, inclusive, was conducted. Seventy three reports were analysed and they were all gathered from open-source intelligence. The reports were first selected on the basis of what the companies classified as an APT attack and were then re-evaluated to our new definition of APT, that is, reports that did not approach the adversary infrastructure, or that did not look at the ”behind the scenes”, were discarded.

By looking at the number of reports per year, there is a clear increase in APT attack awareness. As it is shown in Figure 4.1 the number of available reports has increased significantly over the past years, with more than double the reports in 2014 compared to 2012. Although it is no surprise that the attacks using these techniques are becoming more prevalent, since they are extremely successful, the increase is also a consequence of the rise in better prepared professionals and the understanding of the importance information sharing has on this field. Security companies are now more open to share their clients breaches as well as enterprises are losing the fear in announcing theirs. This is an important step to the security community because the more we understand who, why and how are they performing these attacks the better we can search for ways to prevent ourselves from becoming just another victim.

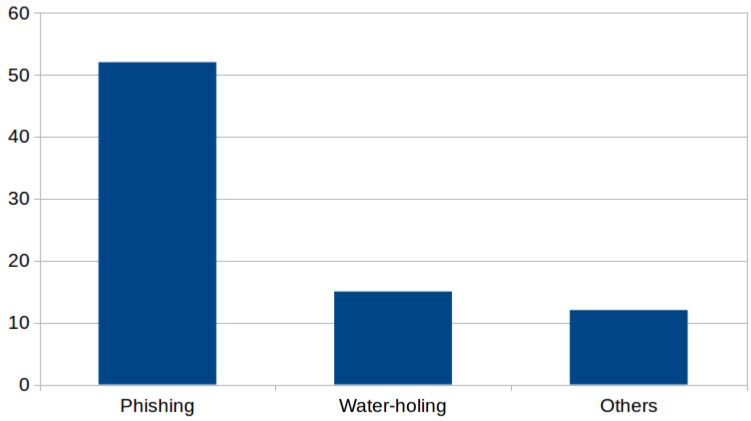


Figure 4.2: The most used infection vectors in the data set

Spear-phishing techniques are still the preferred choice of attackers, with two out of three attacks using this vector as seen in Figure 4.2. Highly social engineered traps, from e-mail to instant messaging, are used by the adversary to manipulate the victim into performing certain actions, like visiting a website controlled by the attacker or opening a malicious file, all via purely digital methods. Instead of targeting the systems, secured by all types of software and hardware making them hard to breach, the attackers target people. A well crafted e-mail, exploiting the human factor, is likely to succeed with very little effort, while on the other hand, a very technical attack will take more effort and cost more to the adversary. Related to this choice are the most targeted applications, a targeted user might be suspicious of an executable file, but will have no problem in opening a Microsoft Office Word or a PDF file, which they most likely already do on a daily basis. Figure 4.3 shows us how exactly this distribution was observed, with the Office suite taking the larger piece of the cake and Adobe products, like Reader and Flash, in close second. Some attacks targeted several applications at once while others were company specific and targeted in-house built applications, which again shows us how targeted these attacks can be.

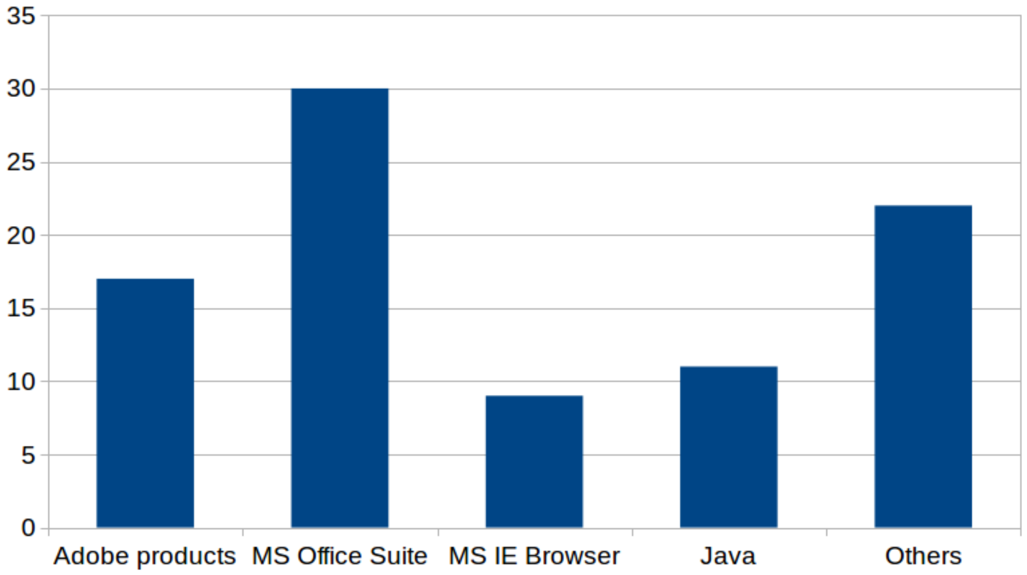


Figure 4.3: The most targeted applications in the data set

Zero-day vulnerabilities are expensive to buy and extremely difficult to find, so it is with no surprise that most of these advanced attacks do not really make use of advanced malware. In this data set, as depicted in Figure 4.4, one in three attacks exploited a previously unknown vulnerability with their tools. If a known trojan exploring a patched vulnerability works, there is no need to waste resources on complex malware exploiting zero-days, the attackers focus on being efficient and target the weakest link. Still zero-day vulnerabilities have their advantages, for instance, they not only give the attacker a much higher success rate, they also increase the time the attacker remains in the system, adding on average one more year until detection.

Attackers behind APTs are more interested in information, or cyber-espionage. As seen in Figure 4.5, this was clear in the data set. Ninety-two percent of the attacks reported as final objective information gathering, the attackers know the value of information and the power it brings, no matter how small or insignificant. From recording microphone audio to accessing Bluetooth enabled devices and downloading contact information, from getting fighter jet blueprints [74] to stealing insider information for an advantage in stock market trading [33]. Cyber-weapons were used very rarely, and probably just as a last resort. These are complex pieces of software that would require a sizeable budget and, if traceable back to their origins, could lead to diplomatic incidents. Finally, some instances of theft were noted, specifically some reports show that malware was being created to steal certificates [55] that were later sold to other malware creators. By signing their code with a valid certificate, attackers add another layer of trust to their files, preventing them from detection and they also might be able to get access to new infection vectors, like a legitimate software update.

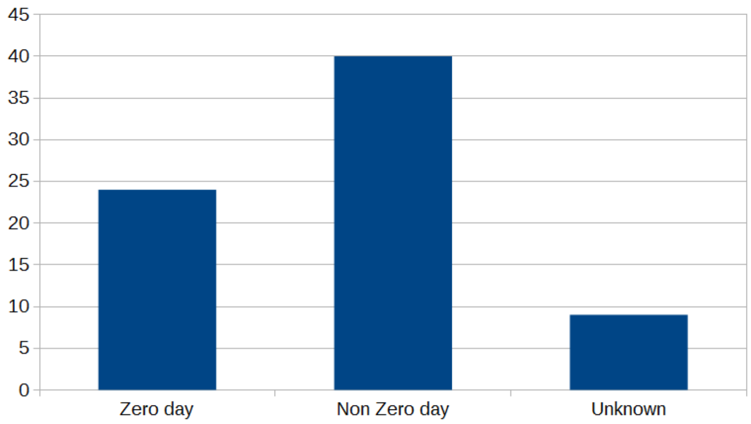


Figure 4.4: Percentage of zero day vulnerabilities used in the data set

Attribution in the digital world is always difficult, from proxy to proxy, C&C to C&C, it is always hard to pinpoint the actual origin of the attack. But it is not impossible. The reports consulted were very thorough and besides looking at the infrastructure, domains and IP addresses to pinpoint groups or nationalities, they also took into consideration other factors, such as operating system language and working hours. It is also safe to assume that these are not your typical cyber-criminals, these are well funded groups with very specific orders of espionage that would benefit governments the most. Let us take those assumptions and create a time line of threat actors, depicted in Figure 4.6. Keep in mind that this time line shows us the predicted year of the earliest malware samples from the reports dated 2010-2014. It is clear, and known, that cyber-espionage has been in use since at least the nineties [15]. Especially the United States of America, this is their playground, they have been studying how to break into systems even before the system is installed. From the very strict export restrictions of cryptographic material until 1992 [113] to the recent leaks from the ex-NSA contractor Edward Snowden [67], we have more then enough evidence showing what security professionals knew for a long time, the United States strive for global cyber dominance and they are not alone. What is happening is the sort of *arms race* in the digital world. We can see that in the last ten years at least eight nations have actively been performing cyber-espionage campaigns on other nations, or specific individuals. As an interesting fact most of them are also part of the nuclear-power list, and although no report talked about them, it would be no surprise that the United Kingdom [85], France and Israel (known to have cooperated with USA in the makings of STUXNET and Flame [98, 150]), are also making use of code to accomplish their surveillance and espionage programs. China is one of the biggest actors, and the loudest one, since every few weeks a new report of a Chinese malware intended for espionage is released, this got five Chinese military hackers charged with cyber espionage against the United States [103]. Some authors believe that Russia was losing ground in this race [51], but as we can see in the time line Russia was quick to join the race and had already developed highly sophisticated espionage oriented software by 2004. A big problem nowadays is that nations don't even need to produce this type of software themselves, they can "legitimately" acquire it from specific vendors[1][2]. Not only that, an increasing number of commercial services offer zero-day vulnerabilities for their clients[3][4][5][6]. This not only opens the field to a lot more, not so technically advanced, actors, but ultimately can have serious consequences to the people [64]. And every nation wants to try and level the information play field not only on possible aggressors, but also on their allies [95].

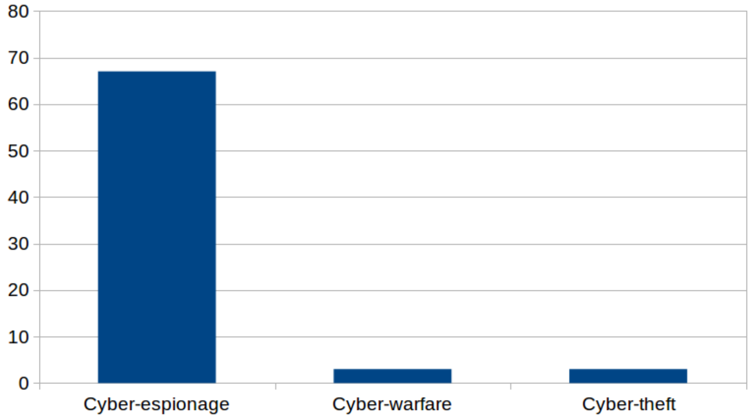


Figure 4.5: The attackers biggest objectives

Conclusions from this survey include how APTs do not necessarily translate to sophisticated code exploiting unknown vulnerabilities. Attackers focus their efforts on the low hanging fruit and if exploiting a patched vulnerability works, there is no need to expend more efforts on other alternatives, this means companies will have to follow better practises regarding software updates and therefore block known infection vectors. We also found that adversaries are targeting people and their inherent vulnerabilities instead of systems, which makes investing in security training and awareness key to protect companies. By updating the software in use and preparing the people using it, we are reducing the adversary attack surface and making them expend more efforts in order to succeed in their mission. E-mail continues to be a very active attack vector and therefore systems that are able to inspect messages and quarantine possible infections before they get on the target inbox will be essential. Not only that, enterprises will need to change the way they think security and apply mature philosophy, people, processes and technology in their

[1] http://www.finfisher.com/FinFisher/index.html Accessed 15-October-2014

[2] http://www.hackingteam.it/ Accessed 15-October-2014

[3] http://www.vupen.com/english/services/lea-index.php Accessed 15-October-2014

[4] http://www.revuln.com/ Accessed 15-October-2014

[5] https://www.endgame.com/ Accessed 15-October-2014

[6] https://www.exodusintel.com/ Accessed 15-October-2014

defence plan [104]. Finally, it was noticeable the increase of threat actors throughout the years, and they are not limited to governments or countries with the capability of performing such attacks. The spectrum of adversaries is increasing and systems need to take that into account in their early development stages, as well as people who need to be informed about the dangers in order to be better prepared.

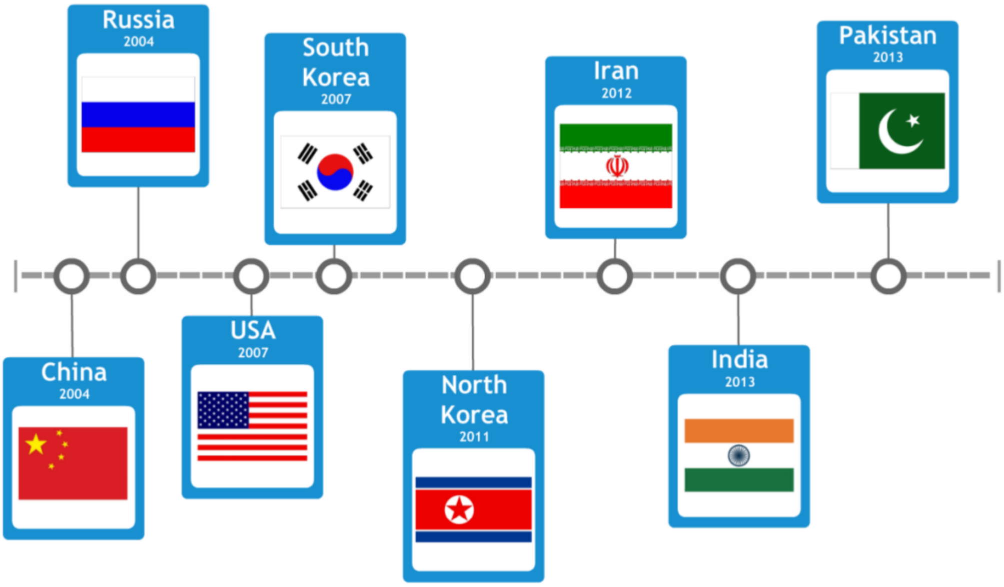


Figure 4.6: Time line of APT actors

Chapter 5

What to do against Advanced Persistent Threats

5.1 Preventing an Advanced Persistent Threat

A lot of literature has been tackling the problem of fighting these threats [131, 135, 141, 52, 47, 89], and it is clear that there are at least two ways to look at defence, prevention and detection. We will first look at the importance of prevention and how being prepared and ready can change the outcome of an attack.

One should plan on being hacked. What will you do when you are a target? No one is unhackable, being good at security is being able to deal with a hack [17]. What policies, procedures and guidelines are available to follow before, during and after an attack? Understand what data and assets are most vital to your operations. What happens if they are successfully attacked? How are they protected? Be hard to breach! [41] propose a model making use of game theory to compute the optimal attack path for the adversary, which then leaves the defender with the best possible response strategies, such as patching the system or setting up new access control mechanisms. [65] follows a similar approach with the use of a Markov belief model, by monitoring the current malicious activities and predicting future actions, that allows monitoring of adversary activity as they progress through the kill chain. The AVI composite fault model (Attack, Vulnerability and Intrusion) goes a step further, proposed in [139] it shows the importance of prevention and how vital it is to reduce the adversaries attack spectrum. By reducing our systems Vulnerabilities (prevention) and blocking attacks (detection) we are significantly reducing the adversary chances of success.

Finally, think like an attacker. Ask yourself the 5ws [145]:

- **Who** would attack us?

  It is really important to distinguish the different types of threats. The defences we employ if we are facing hacktivists will be different from the ones we set-up against cyber-criminals or nation states. All of them will use different tactics, procedures and have different motives, expertise and money to spend for the success of the attack.

- **What** do they want?

  We should look at the services we provide, the information we hold and understand how our assets could be important to the different adversaries described in the previous point. Client private information, government secrets or desired intellectual property, are some examples of information that might need increased protection.

- **When** would they attack us?

  Is the company publicising their appearance in a future conference? Did the company just made a very important announcement? Are we looking to hire new personnel? All these events can trigger the start of an attack campaign, as well as be used for spear-phishing mails. If an attacker knows that a company is hiring a developer, he might send an infected curriculum vitae to the human resources department and achieve delivery followed by initial infection.

- **Where** would they strike?

  Would they go as far as to physically infiltrate the company? Maybe strike a different branch? How good are the security policies in the third parties working for the company? Are they going for the servers or the people? Should the company be cautious with e-mails or vulnerable application running at the servers?

- **Why** would they?

  Are they looking to make a quick buck? Is the profit from the attack worth the time and money spent on it? Will the the company be the latest target of a hacktivist group? Are they looking to spy on us for the long term? Are they using our network to target another?

What about systems that prevent these attacks? Anti-virus, firewalls and even extra preventive software like anti-exploit[1] are a needed layer of defence, but as we have seen before they are not enough. Next we will see how proposed systems and frameworks could help.

The teams behind an APT campaign usually have one objective, data exfiltration, but cyber-destruction might also be a goal. To protect critical systems and infrastructures against destruction we need resilient intrusion tolerance systems and architectures, i.e., providing assurance of system operation in the presence of compromise [34, 139, 128]. These are the type of measures governments need to take to increase their countries overall cyber-security. The focus here is to protect the system from faults, malicious or not, and

[1] `http://www.malwarebytes.org/antiexploit/`

guarantee that the system will continue to work properly. Even with these systems the problem persists, because the most common objective for the APT team is espionage. There is a need for completely different defence mechanisms which should focus on protecting people and data instead of the system as a whole. Data protection and loss prevention can be achieved with techniques like fragmentation and scattering [39], information dispersal [112] and secret sharing [79]. Keep in mind that all these frameworks are designed so that the attacker cannot acquire the data by targeting what is keeping it, but if an authorized person legitimately downloads that information and the APT hacker has already compromised this person's system, the problem will still persist, the attacker will have access to the information. Although those frameworks would definitely help by protecting the servers that are storing the information, awareness training [121] is extremely important in the fight against APTs. Some researchers have worked on this topic, i.e., by developing spear-phishing prevention frameworks, it would be possible to protect the system by preventing people from clicking on or even downloading anything malicious [78, 143, 130, 75, 35]. Spear-phishing is a huge infection vector, and since attackers still make use of infected PDF files, [100] has an up-to-date survey on state-of-the-art malicious PDF detection.

Despite their huge rate of false positives and the never ending amount of logs, anomaly based intrusion detection systems could be of help in preventing novel attacks [111], with some researchers using them to try and predict a compromise [77] but as we have seen before, they have problems in their basic assumptions. Another problem that remains even with all this technology is insider attacks, specifically intentional ones (unintentional insider threats happen when people unknowingly fall for a phishing attack). Some frameworks have been proposed to detect insider attacks [70, 120, 87, 108], but employers should be alert, conduct proper interviews, both in hiring and on contract termination, as well as background checks. Administrators should also make sure that the principle of least privilege[2] is enforced by security policies and they should have continued training, especially in log analysis. It is important, against APT attacks, that administrators are experienced in analysing logs and do it regularly since one unknown infection in the network can propagate and cause more damage, [88] proposed a search engine to discover other victims inside a network during an APT investigation based on attributes acquired from a known APT victim. Finally, and the best overall defence strategy, pioneered by Lockheed Martin in 2011, the *Intelligence driven defence* [62], a defence framework informed by adversary campaigns and intrusion kill-chains. As stated by the Intelligence and National Security Alliance [9]: "A kill chain is a sequence of activities and overall operations that a threat vector must traverse in order to cause an effect. If the sequence can be interrupted or defeated at any point, the threat actor cannot inflict the effect that he intends", i.e., not only we should think like an attacker and try to predict what they will do next, we should

[2] Having only the necessary permissions to do a job correctly.

have several security layers, incrementally more strict, capable of disrupting the overall attack campaign. [20] extends this concept by allowing the adversary to remain inside the network in order to learn and gather counter-intelligence for future attacks. By learning how the different groups conduct their reconnaissance phase, we may be able to predict better which systems or people are most vulnerable, and therefore, prepare them better. If the adversary uses spear-phishing, we can use anti-spear-phishing systems or simply have awareness training sessions to better prepare everyone. If we know that the attackers will try to use HTTPS to upload files to a C&C we can be vigilant about network traffic and try to catch it there [140, 14]. It all boils down to understanding the threat and having an active and prepared security team.

5.2 Detecting an Advanced Persistent Threat

As we have seen, APTs are a mix of complex processes and sophisticated techniques. Nevertheless, Anti-APT systems are being developed and sold by well known security companies, including but not limited to FireEye[3], DELL[4], PaloAltoNetworks[5] and ZScaler[6]. Most, if not all of them, make use of sand-boxing techniques in some sort of appliance or by using cloud services, together with information gathered from threat intelligence resources, which is the correct way to go regarding APTs. Nevertheless, when an independent test was conducted on the most recent, sophisticated and expensive systems that promise to detect APTs, it was proven that they fail to block even an attack made with moderate effort [59]. The systems are proprietary so we can not say what they are doing wrong, or right, and how they are being bypassed.

In order to propose a system built for detection, we must first define the scope of the detector. Following the data surveyed in Chapter 4, we can see that phishing, more specifically spear-phishing, is the most common vector of attack. The typical APT delivery is accomplished by an e-mail with either a link or an attachment containing a malicious exploit that will install a Remote Access Trojan. We will focus on detecting this type of attacks here, but leave some guidelines for other types of detection in a later section. For a thorough survey on the techniques and tools used in malware analysis refer to [38], specifically section five. If the spear-phishing e-mail is not blocked by any other security measure, one of the most common proposed techniques is the Sandbox. The Sandbox is a really critical layer of security because it is not signature based. Unfortunately this industry is just like the Anti-Virus one, a cat and mouse game, with attackers trying to detect if their malware is running on a simulated environment and making use of techniques to

[3] https://www.fireeye.com/

[4] http://www.secureworks.com/

[5] https://www.paloaltonetworks.com/

[6] https://www.zscaler.com/

escape it [123], and defenders trying to hide that fact while simulating user input[7] and system calls. By developing custom sandboxes we are increasing the work required by the adversary. By selling the same sandbox as an appliance we lose that advantage, unless we carefully control the appliance usage as attackers might just acquire those systems and study how to bypass them. Both options are viable, some companies will prefer to have the appliance on their side, while others will be happy to let the computation be done in the cloud. This brings us to another problem in using this solution, scalability. The amount of e-mails with attachments and URLs is significant, even for smaller companies, so which of the e-mails should the system check? A map of the companies personnel showing who has access to what, and which assets are of vital importance, could help us identify priority targets for the adversary. Nevertheless, it is important and would be better to check all e-mails since an attacker might target a ”low priority” employee and wait for some administrator login on that machine later on.

5.2.1 How a system would work at AnubisNetworks

AnubisNetworks[8], a BitSight Company[9], is in a prime position to implement a security system capable of protecting its clients at the e-mail level. Being an anti-spam company, the attachments and links present in an e-mail can be checked, without violating the privacy of the users. By building a custom sandbox that would not leave the AnubisNetworks facilities and execute on it suspected attachments as well as visiting URLs, it would be possible to provide an even better and safer e-mail protection system. A sample framework is proposed making use of current systems at AnubisNetworks, such as Maltracker and Mail Protection System (MPS), which is a security e-mail gateway with carrier grade features that protects and controls networks and users from spam and malware. This is possible, in part, thanks to the filtering technologies in use, such as, but not limited to, real time reputation analysis, heuristic analysis, anti-spoofing, and anti-phishing. Before going into detail on how the system would work, first a short definition of Maltracker and some recommendations in regards to APT detection are given, together with a **new** approach proposed here for people mapping.

[7] For example: `http://www.autoitscript.com/`

[8] `https://www.anubisnetworks.com/`

[9] `https://www.bitsighttech.com/`

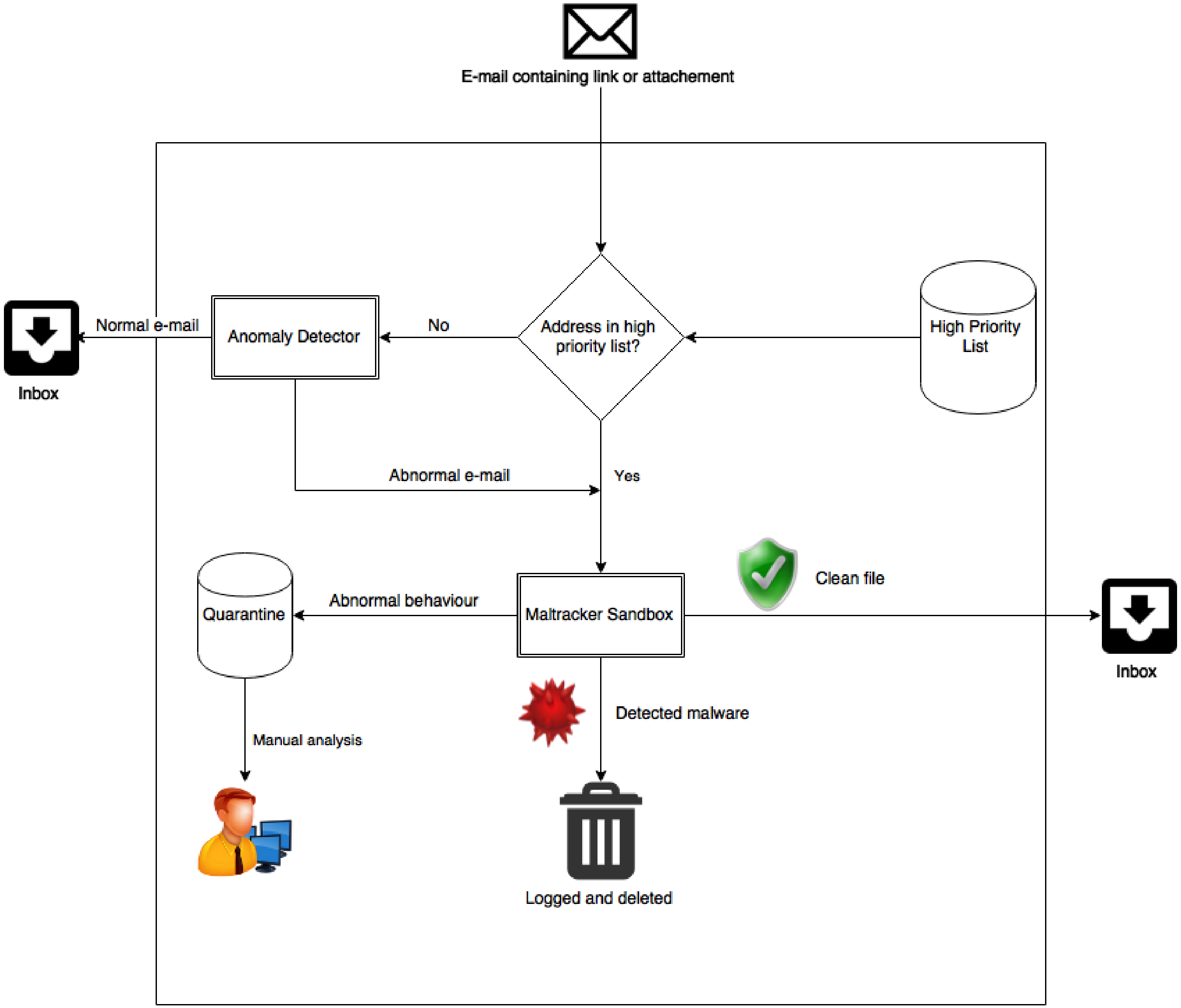


Figure 5.1: Proposed framework to implement in AnubisNetworks

Maltracker

Maltracker is a sandboxing platform for malware analysis, it is scalable and therefore suited to perform hundreds of analyses in parallel. The features include, but are not limited to, static and behavioural analysis of files and URLs, anti-virus scanning with third party software and detection of C&C communications. Although a system like this is great in detecting the typical malware, like crypto blockers or keyloggers, it would need changes to be able to detect the APT specific infection vectors, such as extraction of password protected compressed files or the simulation of user input at malicious webpages, like pressing a download button. One of the problems consists on how the network is setup and how the analysis would be made at the sandbox, since an exploit on application x version y running on operating system z may not work in any other combination. There are, at least, two approaches to this problem for heterogeneous or homogeneous networks. For the former, a brute force approach would be taken, consisting on having the sandbox with several configurations, operating systems, applications and respective versions, and executing the analysis on all of them. Although this approach puts more strain on the overall system and delays the analysis, the administration will not need to employ strict policies on systems or control user hosts, which in today's ever changing landscape, makes it the most viable option. Alternatively, an homogeneous network would allow for a smarter analysis on the sandbox by maintaining it and keeping it updated just like the hosts in the network. A problem with this approach would be that if the sandbox does not detect the attack, the attacker will have a way into any host in the network.

People mapping

The outcome of this mapping is a list of pairs defined by a person and his or hers correspondent mapping score. This mapping score will be obtained by understanding:

a) what is their role and how important is it,

b) how long have they been in the company,

c) what information and/or systems they have access to,

d) how close are they to C-Levels (such as Chief Executive Officer),

e) how visible are they to the public (e.g. email address available on company website or very detailed social network profiles; elaborate reconnaissance like an adversary would),

f) how many e-mails they receive per day, and

g) how susceptible are they to phishing (perform mock phishing attacks and evaluate results).

These are some examples and depending on the company or its objectives, different questions may be asked, but the idea remains the same, different people behave differently online and by performing this kind of evaluation we improve the overall system security performance. Instead of checking every single e-mail, we only perform this detailed analysis on the addresses belonging to the people with the highest scores, called the *high priority list* or HPL.

E-mail anomaly detector

Just like some of the literature reviewed in Chapter 2, this system would benefit from email categorization. Specifically, after a learning phase for each user, the system could categorize normal email traffic and raise alerts for abnormal behaviours, inbound or outbound. Besides looking at where the email came from it would also be possible to look at the headers and understand if the mail is traceable and if the source is trusted, or to whom it is destined. For example, while it is normal for someone in human resources to receive emails with attached files from unknown addresses, maybe the same is not applicable to other employees. The addition of this system increases the overall security, since the adversary may target personalities not present in the high priority list.

The proposed framework is shown in Figure 5.1. After deciding on who makes it to the priority list and what defines an email as abnormal, the system is ready to go. We propose three different ways of setting up the framework, detailed in Table 5.1. The alternatives are based on which emails to check: every email (level 1), HPL personnel together with suspicious emails (level 2) or just HPL personnel (level 3). We will detail the Level 2 alternative for its balance of effectiveness and performance requirements.

The system would work as follows. After passing by MPS, if the email contains a link or attachment, that was not already blacklisted, it is passed to our system:

- The recipient would then be checked against the *High Priority List*, and if a match occurs the attachment and/or link are executed in the sandbox environment;

- If no match in the *HPL* occurs, the email is forwarded to the anomaly detector, and if the message is categorized as abnormal it is sent to be thoroughly analysed in the sandbox.

- This sandbox environment will then complete the analysis with one of the three evaluations, clean, infected or abnormal;

- If the analysis is not able to conclude neither clean nor infected, the email is quarantined and will await manual analysis (only the sandbox, when the email is evaluated as clean, or the human analyst can authorize the final delivery of email);

A system like this, just like all security mechanisms one can put in place, is not full-proof. False-positives and negatives will happen, but we hope that with manual analysis those numbers may remain low enough, so the system still remains a viable option. APTs as we have seen, might represent one in a thousand emails, but if with this system we can stop that email, stop an infection that would later lead to espionage or destruction, it is worth the extra computational, personnel and timely costs.

	E-mails checked	Effectiveness	Performance requirements
Level 1	High Priority List	Average	Low
Level 2	HPL+Abnormal emails	Above average	Medium
Level 3	All emails	Maximum	High

Table 5.1: Different uses of the framework depending on requirements

5.2.2 Other detection schemes

In this section we will take a closer look to the detection of an APT attack by focusing on the four major attack steps proposed by ZScaler:

1) Reconnaissance,

2) Initial infection,

3) Control, and

4) Data exfiltration.

At each phase we will study how different detection schemes and mechanisms could be used at the three proposed ”zoom-layers”, depicted in Figure 5.2, namely:

a) Internet,

b) Intranet, and

c) Host.

When we refer to the host we are referring to a machine, or set of machines, inside the intranet, to which we have total access and control. The intranet is a controlled and secure environment, in which we have access to all the information that enters and leaves the network, and finally the demilitarized zone represents a set of machines open to the internet, we still have control over them and access to the information but they are not as secure since they are exposed to the dangers of the world wide web. At each layer we will have access to different types of information, which allows us to predict better ongoing attacks, or act upon them. We will discuss what could be done by different companies with different services if they worked together as one, as well as some ideas that could be applied in standalone.

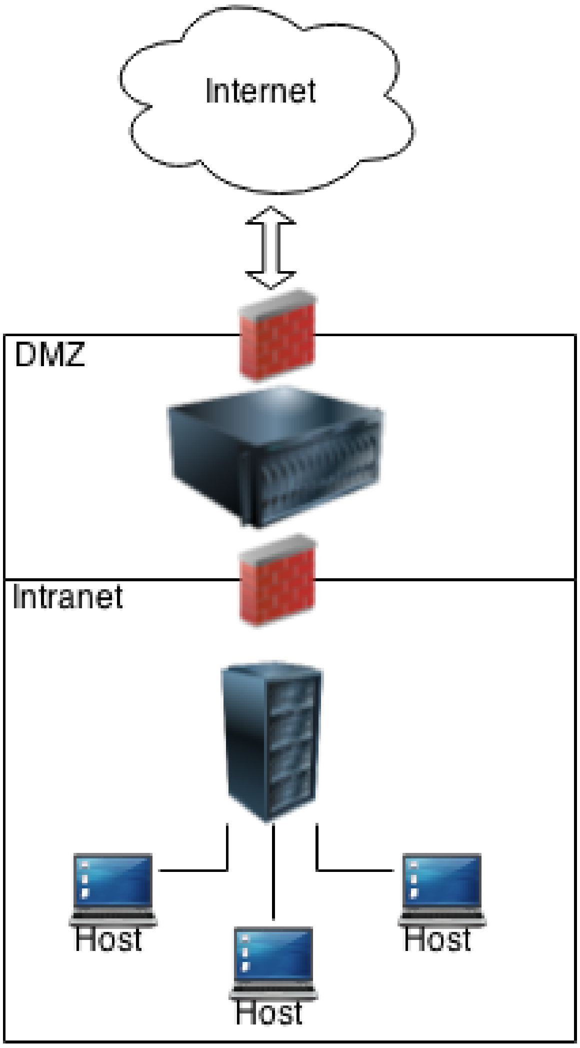


Figure 5.2: Zoom of proposed layers

1. **Reconnaissance**

   (a) Internet: In this particular phase we can only look at the internet layer since the adversary can gather all the information needed to carry out the attack with no direct interaction with our infrastructure, which is called passive reconnaissance. We include in this layer servers and services in the demilitarized zone of the network. Although we could be on the lookout for port scans, those have become part of the internet and alone do not mean much. What we can do is gather information from different events and correlate them within a certain time frame. The events and correlations may then be used to predict malicious activity. If social networking companies, search engines and e-mail providers worked together with potential targets, a new range of options would become available. For instance:

- Company *x* detects a port scan within its servers,
- Search engine records exhaustive queries for a specific employee,
- Employee social networks are meticulously analysed, which is also recorded,
- Employee receives an uncommon e-mail with an attachment or link.

These steps would have been some of the events which triggered an alert in order to prevent the next phase of the attack. This might seem like a big stretch, but work has already been done in some of the points we expressed. For instance, Allen et al. [5] provides us with a good example of what can be done to detect active reconnaissance. Active reconnaissance makes use of techniques which have direct contact with our servers, leaving us with some information of the attackers intentions. By manually identifying characteristics in the generated traffic of well known reconnaissance tools, one can create specific rules to block and/or alert administrators of an ongoing, or possible future, cyber attack. Paradise et al. [107] did work on understanding machine operated social network profiles, called social bots, used to gather information on the targeted organization. They propose and evaluate monitoring strategies that focus on detecting friend requests from social bots. To identify intruders, the authors propose the use of manual inspection to determine if a friend request is legitimate, together with counter-intelligence investigation, such as a background check on the requester profile. Because such approach would be infeasible on every profile in the organization, a subset is chosen. The challenge is selecting a good subset. Following our proposed *people mapping* in the previous section, one could use the list of high priority personnel as the chosen subset or, as the authors proposed for future work, the use of honeypot profiles. We can see how both works can be merged together with our proposed sandbox email analysis to protect infrastructures even at the earliest phases of attacks. A problem remains with the good identification of *time-frames*, that is, correlating these events in a certain window of time making them relevant to our defence. An email coming six months after a scan and a friend request in a social network might not be as relevant as an email delivered in the same week. Other researchers look for alternatives, such as deception [118] by dropping, delaying or modifying server and services responses, in order to stall the adversaries reconnaissance.

(b) Intranet: It is important to note that we will not include insider threat reconnaissance at this stage as we consider that to be part of the control phase, since the attacker is already inside the network and has already acquired inside knowledge.

(c) Host: Since there is no direct contact with internal hosts from the outside world for reconnaissance purposes, we add nothing at this stage.

2. **Initial infection**

    (a) Internet: At this stage, two things can happen, either the victim clicks on a malicious link or opens a malicious attachment. If the link is known to be malicious it would be possible to re-direct the request to a safe location, for instance at the Domain Name System level which is already done by some Internet Service Providers and Public DNS servers. Since APT adversaries know about these lists they register new unknown domains, sometimes target specific, for their campaigns. As for the attachment, after successful infection, the event to which we could pay attention is the initial communication to the C&C, also called beaconing home. Unfortunately the traffic generated by these events looks just as normal as any other traffic, whether it is encrypted or not.

    (b) Intranet: Besides being able to see the C&C beacon going out, at this stage we actually have the attacker's code. One of the most employed ways to prevent initial infection is to run that code on controlled systems, such as the proposed sandbox framework, which are usually appliances installed inside company networks. The objective of this sandbox is to detect the malware before it is delivered to the target and executed. Different malware might be able to detect its analysis and halt or change the execution path, nevertheless the sandbox has to be able to detect most of the malicious behaviours and block the infection.

    (c) Host: This is possibly the best position for a defender to detect and neutralize remote access trojans because the malware is built to remain persistent in the host while having a numerous set of capabilities that either leave traces in the logs or expose themselves with ”noisy” events. By using an agent at each and every host in the network one could look for possible application exploits, such as memory corruption or overflows, and malicious activity, like the typical creation of registry key and start up process. These agents could be: hardware based, small and simple so it would be possible to guarantee its security and correctness, but could also be software based, following the techniques employed by anti-virus, rootkit and exploit detectors. The major advantage of a framework like this, is that the agent would be transparent to the attacker and all the communication between the administrator node and the agent would go unnoticed. This is important so that the attacker would not ”panic-stop” the attack or delete its tracks. This agent would then be responsible for reporting those malicious events to the administrator node. An example is shown in Figure 5.3, the idea is to block any further communication between the infected machine and the attacker while still keeping the infection for analysis, classification and even attribution. Christodorescu et al. [25] showed us how virus scanners are susceptible to obfuscation employed by malware writers since they are purely syntactic and ignore the semantics of instructions, making them unreliable in detecting initial infection at the host. They presented a malware detection algorithm that incorporates instruction semantics and proved its resilience to common obfuscation techniques. Specifically, the authors contribute with a definition to determine malicious behaviour, and although undecidable, it is capable of handling specific sets of malware transformations. Future work should be done on defining APT specific behaviours that could be used by semantic based systems or help normal virus scanners in detecting this specific threat. Josse [69] presents a framework based on virtual environments and manual analysis to understand how malware installs itself and maintains hooks within the target system. The tool is designed for security analysts and the information is obtained by observing the whole infected host from outside. At the same time, the analyst has the possibility to interact with the virtual machine and dynamically drive the execution of the malware, which is crucial for detecting custom and advanced APT malware initial infection, obfuscation and encryption techniques.

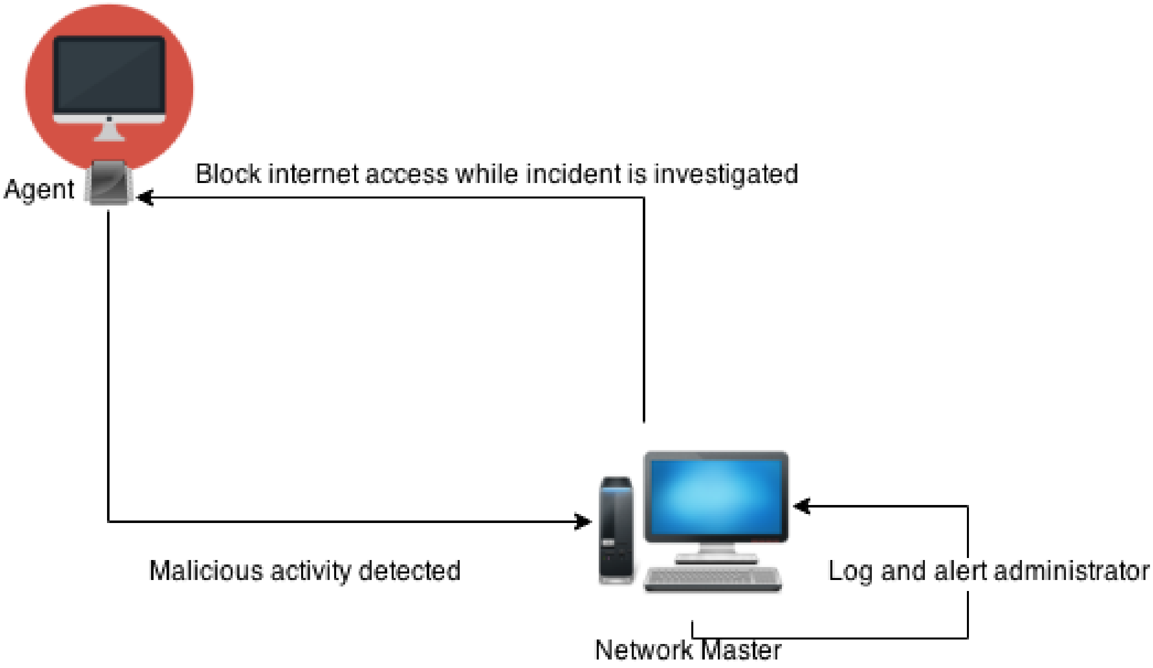

Agent
Block internet access while incident is investigated
Malicious activity detected
Log and alert administrator
Network Master

Figure 5.3: Example of host agent communicating with its Master

3. **Control**

   (a) Internet: This is where the botnet hunter companies shine by sink-holing domains generated by a domain generation algorithm (DGA) and looking for botnet traffic. The difference is that the typical APT will not have hundreds of thousands of infections. Instead, it will have relatively legitimate looking traffic and will most likely not use DGAs. Some events one could look at would be new domains with no reputation getting small amounts of traffic from only one particular set of IP addresses, or the domain receiving several continuous HTTP requests without a corresponding accessible web page.

(b) Intranet: By white listing domains, blocking certain protocols and making use of detection systems, one could detect the communication between the infected host and the adversary, usually by making use of the same techniques employed by botnet masters. Although not APT specific, Gu et al. [57] gave us an example of such detector. By focusing on the IDS-driven correlation with the tracking of the two-way communication between internal hosts and external entities, the system develops an evidence trail that can later be matched with an infection sequence model, including stages of the process such as malware download and outbound bot dialogue, and evaluated on their indicative of a successful local host infection. Not only the system becomes ineffective if encrypted communication channels for C&C are used, the authors admit the impossibility to accurately detecting all steps. One remaining problem relates to predicting the order and time-window in which the events are recorded (a user may get infected and the first communication with the C&C happens only after three months), which the system would no longer be able to correlate. Gu continued his work [58] but now with a study on detecting botnet C&C channels, at the network level, based on IRC and HTTP protocols without any prior knowledge, such as signatures or binaries. Some of the problems with detecting these behaviours include:

- Malware following normal protocol usage;
- Communication with low volume traffic;
- Only one infection in the network;
- The communication may be encrypted.

Because APT communications still make great use of the HTTP protocol, this work has a particular interest to us. The authors observed that for HTTP-based C&C, bots had a strong periodical visit pattern, but the detection system was not as robust, or evasion-resilient, for a single client infection. For group infections however, encrypting messages in HTTP was not a problem for the correlation analysis, an improvement from previous work although the framework is still vulnerable to exploitation of time windows. As we can see with some of the work done in this field, the intranet is the best available choice to detect C&C communication that could indicate possible infections. If it would also be possible to detect malicious activity inside the network, such as scans or peer-to-peer communication, we would have the opportunity to pinpoint the infected machine and act on it. By detecting the adversary lateral movement we might be able to contain the infection and stop it from spreading.

(c) Host: It is also possible to detect at this level the unusual communication between the attacker and the host itself by looking at new connections and open sockets, as well as outgoing scans, but by being at the host, one level deeper, we get access to new events. We are now able to detect administrator abuse, such as log file deletion, execution of unknown software or the creation of hidden files.

4. **Data exfiltration**

   (a) Internet: Just like at the control phase, here we can look for connections being established with unknown or abnormal addresses. The particularity of this phase relies in the difference of the data size. Data exfiltration will send larger communication packets to the C&C. This could be used to our advantage and help us detect and block the exfiltration.

   (b) Intranet: The main goal here is to detect confidential data exfiltration, which is the most common objective in this type of attack. By looking at which information is leaving our network, how and to where is it going, we can stop the exfiltration and alert administrators about the leak, who can then investigate the infected hosts. If the attacker does not encrypt or compress the information, a digital watermark would be enough to detect the leak, the watermarking should be detectable even if the leak is parted in clear format. If encryption is used, one possible solution would be to use a proxy that would preform a Man-in-the-Middle ”attack”, that is, either the information is deciphered at the proxy or it will not be allowed to leave the network. The same rule would be applied to compressed files. This is possibly the best option towards detecting encrypted exfiltration. Each host would have a cryptographic key to encrypt data, and the proxy would then be responsible of decrypting the information, check its legitimacy, encrypt it again, and send it to the final destination with the corresponding agreed key. Liu et al. [90] did some work in this field, particularly, they developed a multilevel framework capable of application identification, content signature and covert channel detection. The main challenge with such a framework is the detection of the information leak despite the transformations used on the content, such as encryption, compression or slight modifications. By relying on signatures representing confidential information, a scalability problem arises, as the system is built to deeply inspect packets, identify applications used, block clear information leaks and look for covert channels. The system can be used to detect most of clear format data leaks, but results for compressed or encrypted data would be slightly more interesting in regards to APTs. Al-Bataineh and White [4] did work on detecting malicious data exfiltration making use of compression and encryption. By using information entropy and byte frequency distribution on the HTTP POST request contents together with information theory concepts which measure randomness in content, the classifier is capable of detecting encryption or compression where it is not expected to exist, making this event an anomaly and a possible cause of further inspection. Although a step in the right direction, the adversary would only need to use HTTPS communication to evade such detection.

(c) Host: Since we could not be sure to detect this phase at the Intranet level, at the host level, the agent would be able to detect the compression/encryption of classified files, looking for example at unusually high CPU usages, and corresponding exfiltration channels, such as external drives, cloud storage or peer-to-peer connections, and act by blocking internet access or alerting the administrators. This could be accomplished with the watermarking techniques described above.

As we have seen, some of the proposed ideas already had some work done by other researchers, but all of them are still open for large improvement. One of the conclusions of this study relates to where and how one can better defend against the adversaries; one can look at the internet to detect steps taken in the reconnaissance phase, at the host to detect initial infection and at the Intranet to detect control and data exfiltration. This is no surprise since each layer of defence has its own access to different events which can lead to a better infection detection. As we have said previously, a system to detect advanced threats would need to be a mixture of different components at different parts of the network. Therefore, one could gather all of the work and ideas, improve them, and create a system capable of better protecting the infrastructure by analysing information from several layers.

5.3 What does the future hold?

Let us start by clearly stating, **APT is the new normal**. What was studied and presented about these threats, such as tactics, techniques and procedures, is currently being used by criminal groups and nation states in their operations. This means that the industry needs to be prepared for such threats starting from development [8], i.e., the basic assumptions regarding security need to be well defined and take into account the most advanced adversaries. Although information and money are strong motivators behind these attacks, looking at the future one particular objective becomes a priority: the need for valid certificates. As anti-virus vendors and operating system creators consider a file to be secure if it is signed with a valid digital certificate, attackers have everything to gain from compromising either a certificate authority [63] or the algorithms used during signing [54]. Both of these scenarios are worrying and certificate authorities need to be very careful with their security, as well as with the certificates they sell. A certificate in the digital world represents trust, so a certificate authority is basically selling trust, and the moment they show they can not be trusted, they will surely suffer great losses, if not bankruptcy. More on the topic, including methods of protection against illegitimately-correctly signed software on [82].

Another evolving trend is the cyber-mercenaries for hire. These are the typical hackers, that formed organized groups, learned the APT tactics and started selling them as a service. When hired, they are given the target and objective. After conclusion they move on to the next client and target. These are very skilled, lethal and low profile hackers. As reported on [68]: ”they are rarely discovered [which] is due in part to their skill level and in part to being misidentified as a state actor instead of a non-state actor if they are discovered. The low risk of discovery, frequent misattribution to a nation state, and growing demand of their services ensures that the EaaS[10] threat actor will flourish”.

Following the previous problem, another interesting trend is related to specific infections being sold by botnet owners. If the worm used by the botnet master infects a certain internet address owned by a high profile company, this infection can then be sold. After purchase, they replace the malware with code of their own, technically more advanced and stealthier. This will alter the behaviour of the machine and it might be perceptible in the network logs. Nevertheless it is interesting how a common botnet infection can become a successfully targeted espionage campaign. Hence the importance of detecting even the most ”basic” botnet infections.

As the global number of mobile devices increases and their respective processing power rises, these ”small PCs” that everyone carries in their pockets become an enticing target for cyber-criminals. Specifically, with 283 million units shipped and over 84% of the market share in the third quarter of 2014 [28], Android is the most wanted target. Users are still not really interested in their mobile security. Symantec [116] reports on the increasing trade-off between privacy and free *apps*. The average user will not even worry about installing anti-virus software, while using the same device for home banking, e-mail and surfing the web. Their inherent mobility and the user need to connect to any available free Wi-Fi provides the adversary with a door into the target corporate networks. Mobile application developers themselves may be targeted as a way to reach other specific victims [127]. By infecting a developer of a popular application attackers not only get access to source code and costumer data, they also gain the ability to issue an update

[10] Espionage-as-a-Service

targeting specific individuals.

Going even further, to a distant future where e-mail protection systems are so perfect that no malware gets by. With the ever increasing number of features in social networking services combined with the relatively simple task of setting up a legitimately looking profile and a little help of social engineering, attackers can easily focus on sending malicious files via those services (Facebook, LinkedIn, Skype and others). And of course, old espionage techniques are still valid [19].

Chapter 6

Future work and conclusions

6.1 Discussion and future work

With this work, we provide interested readers with a literature survey from which we derived a definition of Advanced Persistent Threat, and we highlighted the reasons that make this type of attacks different. We thoroughly examined its life-cycle, profiled a typical adversary and discussed why current defence mechanisms fail to stop them. Other researchers, or companies, can make use of this information when working on new detection schemes, or on their new products. Another main contribution of this work is a set of guidelines on what to do against these threats. More specifically, we leave the blueprints on how a system could be implemented at AnubisNetworks to improve detection of this type of attack campaigns, focusing on the e-mail infection vector, which would result on an overall improvement of the service provided. We also discuss how other companies or researchers can plan their products, or work, to face this threat at the different layers of the network. One could follow up on the proposed ideas and base their research on the related work we reviewed, or start a completely new work from scratch. An important feature of such a system in order to detect or block these threats, is the need to be built like a LEGO$^{®}$ set, that is, the success of the system will depend on all the little pieces put together at the different layers of the network. From detecting reconnaissance steps and stopping initial infection to blocking data exfiltration, different phases of attack require different systems to detect them.

As a consequence of the literature survey and the proposals within this work, some topics for suture work arise naturally:

- Obviously the most obvious path to follow after this work would be to actually implement the proposed blueprints at AnubisNetworks, and although we are sure that a system like this would find success at the company, it is still a complex framework with details that would take a lot of time and work to complete. Once the system is online, we would then have to test it against real threats, specifically, evaluate its performance in real scenarios, how many e-mails could the system handle versus the amount of e-mails a high priority person receives, and based on those facts refine the system;

- Understanding of what the network is doing that is normal and what is it doing that can be said abnormal to help anomaly detection schemes in better characterizing what an APT attack looks like in the network;

- Improving other current defence mechanisms, such as anti-virus or firewall solutions, to better adapt in this threat landscape;

- Working on reducing the threat of the people vector with security awareness frameworks or tools;

- Taking advantage of what we wrote about future attacks, one could also pick any of the ideas proposed for APT detection and, together with the presented frameworks in related literature, work on analysing, improving and setting them up for real scenario testing.

6.2 Conclusion

What is an Advanced Persistent Threat? We looked at the problem following a scientific perspective by first finding the source of the acronym and what it meant by then. We saw how that meaning changed because of misinformation and we discussed an approach, closer to its origin, to look at the APT from a birds eye-view, not just by looking at the malware perspective, but by looking at the attack campaign as a whole. Security companies grabbed the term to describe sophisticated pieces of malware, but the truth is not quite that simple. Highly organized groups with available resources were able to successfully infiltrate high profile corporations with trivial, if not free and open source, pieces of software.

We later showed how APTs are different from other threats and why they deserve to be studied as such. We examined and critiqued one of the most used life-cycles when describing the typical APT steps and we also presented other life-cycles from different sources. We concluded that a more representative model was necessary, hence we contributed with a novel APT representation that improves the understanding of how complex, organized, resourceful and tenacious are these adversaries. These were later characterized using a list of features, together with the information we gathered, that will help readers understand with greater depth who is behind these attacks, what are their backgrounds and what is their level of expertise. Another problem we encountered and showed why it still persists, is the lack of functional and effective defences against these threats. As most security schemes still rely on detecting known threats, and anomaly detection lacks on consistency, by knowing this adversaries exploit current frameworks at their most basic assumptions, making use of encryption, polymorphism and other techniques. One conclusion is clear, security systems need to adapt and change their assumptions overtime on what a successful attack is.

A big problem we encountered while developing this work was the lack of data regarding APT attacks. Statistics were usually focused on the companies targeted and on the information stolen. We went looking for information regarding the actual attacks, that is, how was the payload delivered, what applications did they targeted and what was the adversaries goal. To fix this lack of literature, we set out to look at the available reports, unfortunately not that many, on discovered APT attacks, and evaluated if they could be classified as APTs. The results were presented here in a summarised fashion with critiques of our own.

Throughout the work we realized that being ready for APT attacks is not all about detection and reaction, but also about prevention and being proactive. We shared the knowledge we built on how one should prepare itself and its systems for these kind of attack before moving on to detection frameworks. The main proposed framework, to be used at AnubisNetworks, would be a great line of defence against APTs, specifically because attackers still make great use of the e-mail vector to deliver their malicious content, which is also the main focus of the company. Obviously, the system will never be perfect, but it will be a step in the right direction to block the threat before any person has a chance to download a malicious payload. The framework focuses on two points:

- First, and for performance reasons, the need to select which e-mails will be checked, a process we called people mapping. The idea is to really understand which people have access to certain resources, how much of a danger they are to the company, and reflect that into the mapping score;
- The second point consists in the integration of the people mapping, the e-mail filtering already in place at AnubisNetworks and a recently developed sandbox system called Maltracker. By presenting a framework capable of merging all these tools together, after implementation and refinement, one could expect to have higher probability of blocking both commodity threats, such as ransom ware or new botnet instances, and more stealthy ones, typical of APTs.

We later took a more scientific approach, hoping to help future research and engineering projects in the field, to the problem of APT detection, and discussed how one could build systems at the different levels of abstraction in the several layers of a typical enterprise network by using some of the related work. From the reconnaissance phase to the data exfiltration, by the internet or at the host, we studied what could be done and which work has already been conducted that could be improved. Finally, we concluded our work by trying to predict how the threat landscape will shape itself up in the upcoming years, how attackers will adapt, how their tools will evolve and what their goals will be.

Bibliography

[1] Aptnotes. `https://github.com/kbandla/APTnotes`. Accessed: 2014-10-31.

[2] Why don't traditional cyber security defenses work? `http://www.fireeye.com/adaptive-defense/why-dont-traditional-defenses-work.html`. Accessed: 2014-10-31.

[3] Elise Ackerman. Confessions of a botnet herder. `http://www.forbes.com/sites/eliseackerman/2012/05/19/i-run-a-small-botnet-and-sell-stolen-information-ask-me-anything/`. Accessed: 2015-01-22.

[4] Areej Al-Bataineh and Gregory White. Analysis and detection of malicious data exfiltration in web traffic. In *Malicious and Unwanted Software (MALWARE), 2012 7th International Conference on*, pages 26–31. IEEE, 2012.

[5] William H Allen, Gerald A Marin, and Luis A Rivera. Automated detection of malicious reconnaissance to enhance network security. In *SoutheastCon, 2005. Proceedings. IEEE*, pages 450–454. IEEE, 2005.

[6] Rohan Amin, JJC Ryan, and Johan Rene van Dorp. Detecting targeted malicious email. *Security & Privacy, IEEE*, 10(3):64–71, 2012.

[7] Kostas G Anagnostakis, Stelios Sidiroglou, Periklis Akritidis, Konstantinos Xinidis, Evangelos P Markatos, and Angelos D Keromytis. Detecting targeted attacks using shadow honeypots. In *Usenix Security*, 2005.

[8] Eric Baize. Developing secure products in the age of advanced persistent threats. *IEEE Security & Privacy*, (3):88–92, 2012.

[9] George Bamford, John Felker, and Troy Mattern. Operational levels of cyber intelligence. *Intelligence and National Security Alliance*, 2013.

[10] Alysson Neves Bessani, Paulo Sousa, Miguel Correia, Nuno Ferreira Neves, and Paulo Verissimo. The crutial way of critical infrastructure protection. *Security & Privacy, IEEE*, 6(6):44–51, 2008.

[11] Aaron Beuhring and Kyle Salous. Beyond blacklisting: Cyberdefense in the era of advanced persistent threats. *Security & Privacy, IEEE*, 12(5):90–93, 2014.

[12] Antonio Bianchi, Yan Shoshitaishvili, Christopher Kruegel, and Giovanni Vigna. Blacksheep: Detecting compromised hosts in homogeneous crowds. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 341–352. ACM, 2012.

[13] Leyla Bilge and Tudor Dumitras. Before we knew it: an empirical study of zero-day attacks in the real world. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 833–844. ACM, 2012.

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

[15] Andrew Bomford. Echelon spy network revealed. `http://news.bbc.co.uk/2/hi/503224.stm`. Accessed: 2015-01-22.

[16] Kevin Borders and Atul Prakash. Quantifying information leaks in outbound web traffic. In *Security and Privacy, 2009 30th IEEE Symposium on*, pages 129–140. IEEE, 2009.

[17] Ross Brewer. Advanced persistent threats: minimising the damage. *Network Security*, 2014(4):5–9, 2014.

[18] Peter Bright. Anonymous speaks: the inside story of the hbgary hack. `http://arstechnica.com/tech-policy/2011/02/anonymous-speaks-the-inside-story-of-the-hbgary-hack/3/`. Accessed: 2014-10-31.

[19] David Brunnstrom and Lisa Shumaker. Dissident warns china sending spies to u.s. in scholarly guise. `http://www.aol.com/article/2014/02/27/dissident-warns-china-sending-spies-to-u-s-in-scholarly-guise/20840193/`. Accessed: 2015-02-02.

[20] Vit Bukac, Vaclav Lorenc, and Vashek Matyáš. Red queen's race: Apt win-win game. In *Security Protocols XXII*, pages 55–61. Springer, 2014.

[21] Muhammad Irfan Afzal Butt. Bios integrity an advanced persistent threat. In *Information Assurance and Cyber Security (CIACS), 2014 Conference on*, pages 47–50. IEEE, 2014.

[22] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3):15, 2009.

[23] Ping Chen, Lieven Desmet, and Christophe Huygens. A study on advanced persistent threats. In *Communications and Multimedia Security*, pages 63–72. Springer, 2014.

[24] Tsung-Huan Cheng, Ying-Dar Lin, Yuan-Cheng Lai, and Po-Ching Lin. Evasion techniques: Sneaking through your intrusion detection/prevention systems. *Communications Surveys & Tutorials, IEEE*, 14(4):1011–1020, 2012.

[25] Mihai Christodorescu, Somesh Jha, Sanjit A Seshia, Dawn Song, and Randal E Bryant. Semantics-aware malware detection. In *Security and Privacy, 2005 IEEE Symposium on*, pages 32–46. IEEE, 2005.

[26] Lucian Constantin. Cryptowall ransomware held over 600k computers hostage, encrypted 5 billion files. `http://www.pcworld.com/article/2600543/cryptowall-held-over-halfamillion-computers-hostage-encrypted-5-billion-files.html`. Accessed: 2015-01-22.

[27] Lucian Constantin. Zeus builder service spotted on the underground market. `http://news.softpedia.com/news/ZeuS-Builder-Services-Spotted-on-the-Underground-Market-177422.shtml`. Accessed: 2015-01-22.

[28] International Data Corporation. Smartphone os market share, q3 2014. `http://www.idc.com/prodserv/smartphone-os-market-share.jsp`. Accessed: 2015-02-02.

[29] Art Coviello. Open letter to rsa customers. `http://www.sec.gov/Archives/edgar/data/790070/000119312511070159/dex991.htm`. Accessed: 2014-11-24.

[30] Jamie Crawford. The u.s. government thinks china could take down the power grid. *CNN*, 2014.

[31] Feike Hacquebord David Sancho and Rainer Link. Finding holes: Operation emmental. 2014.

[32] JA De Vries, J van den Berg, ME Warnier, and H Hoogstraaten. Towards a roadmap for development of intelligent data analysis based cyber attack detection systems. 2012.

[33] Kristen Dennesen, Barry Vengerik, Jonathan Wrolstad, and Jordan Berry. Fin4: Stealing insider information for an advantage in stock trading? `https://www.fireeye.com/blog/threat-research/2014/11/fin4_stealing_insid.html`. Accessed: 2015-01-22.

[34] Yves Deswarte, Laurent Blain, and J-C Fabre. Intrusion tolerance in distributed computing systems. In *Research in Security and Privacy, 1991. Proceedings., 1991 IEEE Computer Society Symposium on*, pages 110–121. IEEE, 1991.

[35] Prateek Dewan, Anand Kashyap, and Ponnurangam Kumaraguru. Analyzing social and stylometric features to identify spear phishing emails. *arXiv preprint arXiv:1406.3692*, 2014.

[36] Stephen Doherty, Jozsef Gegeny, Branko Spasojevic, and Jonell Baltazar. Hidden lynx–professional hackers for hire, 2013.

[37] David Drummond. A new approach to china. *The official Google blog*, 12, 2010.

[38] Manuel Egele, Theodoor Scholte, Engin Kirda, and Christopher Kruegel. A survey on automated dynamic malware-analysis techniques and tools. *ACM Computing Surveys (CSUR)*, 44(2):6, 2012.

[39] Jean-Charles Fabre, Yves Deswarte, and Brian Randell. *Designing secure and reliable applications using fragmentation-redundancy-scattering: an object-oriented approach*. Springer, 1994.

[40] Nicolas Falliere, Liam O Murchu, and Eric Chien. W32. stuxnet dossier. *White paper, Symantec Corp., Security Response*, 2011.

[41] Xupeng Fang, Lidong Zhai, Zhaopeng Jia, and Wenyan Bai. A game model for predicting the attack path of apt. In *Dependable, Autonomic and Secure Computing (DASC), 2014 IEEE 12th International Conference on*, pages 491–495. IEEE, 2014.

[42] Avgoustinos Filippoupolitis, George Loukas, and Stelios Kapetanakis. Towards real-time profiling of human attackers and bot detection.

[43] FireEye. Poison ivy: Assessing damage and extracting intelligence. *Special Report*, 2013.

[44] FireEye. Apt28: A window into russia's cyber espionage operations? *Special Report*, 2014.

[45] FireEye. Fireeye advanced threat report:2013, 2014.

[46] Fortinet. Threats on the horizon: The rise of the advanced persistent threat. 2013.

[47] Ivo Friedberg, Florian Skopik, Giuseppe Settanni, and Roman Fiedler. Combating advanced persistent threats: From network event correlation to incident detection. *Computers & Security*, 2014.

[48] Brian Fung. Why insiders, not hackers, are the biggest threat to cybersecurity. *National Journal Available at:¡ http://www.nextgov.com/cybersecurity/2013/06/why-insiders-not-hackers-are-biggest-threat-cybersecurity/64595/¿[Accessed 06 November 2014]*, 2013.

[49] Roland Gabriel, Tobias Hoppe, Alexander Pastwa, and Sebastian Sowa. Analyzing malware log data to support security information and event management: Some research results. In *Advances in Databases, Knowledge, and Data Applications, 2009. DBKDA'09. First International Conference on*, pages 108–113. IEEE, 2009.

[50] Pedro Garcia-Teodoro, J Diaz-Verdejo, Gabriel Maciá-Fernández, and Enrique Vázquez. Anomaly-based network intrusion detection: Techniques, systems and challenges. *computers & security*, 28(1):18–28, 2009.

[51] Keir Giles. "information troops"-a russian cyber command? In *Cyber Conflict (ICCC), 2011 3rd International Conference on*, pages 1–16. IEEE, 2011.

[52] Paul Giura and Wei Wang. Using large scale distributed computing to unveil advanced persistent threats. *SCIENCE*, 1(3):pp–93, 2013.

[53] Steve Gold. 160,000 new malware samples arriving every day. `http://www.scmagazineuk.com/160000-new-malware-samples-arriving-every-day/article/349235/`. Accessed: 2015-02-18.

[54] Dan Goodin. Crypto breakthrough shows flame was designed by world-class scientists. `http://arstechnica.com/security/2012/06/flame-crypto-breakthrough/`. Accessed: 2015-02-02.

[55] Kaspersky Labs' Global Research & Analysis Team GReAT. Winnti. more than just a game. `securelist.com/analysis/internal-threats-reports/37029/winnti-more-than-just-a-game/`. Accessed: 2015-01-22.

[56] Information Security Media Group. Ismg advanced persistent threats: Survey. 2014.

[57] Guofei Gu, Phillip A Porras, Vinod Yegneswaran, Martin W Fong, and Wenke Lee. Bothunter: Detecting malware infection through ids-driven dialog correlation. In *Usenix Security*, volume 7, pages 1–16, 2007.

[58] Guofei Gu, Junjie Zhang, and Wenke Lee. Botsniffer: Detecting botnet command and control channels in network traffic. 2008.

[59] Boldizsár Bencsáth Levente Buttyán Roland Kamarás Gábor Molnár Gábor Vaspöri Gábor Ács Kurucz, Zoltán Balázs. An independent test of apt attack detection appliances. *Technical Report of MRG Effitas and CrySyS Lab*, 2014.

[60] Hannes Holm and Mathias Ekstedt. Estimates on the effectiveness of web application firewalls against targeted attacks. *Information Management & Computer Security*, 21(4):250–265, 2013.

[61] C Hummel. Why crack when you can pass the hash. *Retrieved November*, 21:2009, 2009.

[62] Eric M Hutchins, Michael J Cloppert, and Rohan M Amin. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. *Leading Issues in Information Warfare & Security Research*, 1:80, 2011.

[63] Mikko Hypponen. Diginotar hacked by black.spook and iranian hackers. `https://www.f-secure.com/weblog/archives/00002228.html`. Accessed: 2015-02-02.

[64] Ionut Ilascu. Finfisher used to spy on democracy protesters in bahrain. `http://news.softpedia.com/news/FinFisher-Used-To-Spy-on-Democracy-Protesters-in-Bahrain-454200.shtml`. Accessed: 2015-01-22.

[65] Georgios Ioannou, Panos Louvieris, Natalie Clewley, and Gavin Powell. A markov multi-phase transferable belief model: an application for predicting data exfiltration apts. In *Information Fusion (FUSION), 2013 16th International Conference on*, pages 842–849. IEEE, 2013.

[66] iSIGHT. Cyber espionage operators sandworm team leverage cve-2014-4114 zero-day. 2014.

[67] Al Jazeera. Timeline of edward snowden's revelations. `http://america.aljazeera.com/articles/multimedia/timeline-edward-snowden-revelations.html`. Accessed: 2015-01-22.

[68] INC. Jeffrey Carr, TAIA GLOBAL. The tries framework: Counter-reconnaissance against eaas threat actors. 2015.

[69] Sébastien Josse. Rootkit detection from outside the matrix. *Journal in Computer Virology*, 3(2):113–123, 2007.

[70] Miltiadis Kandias, Alexios Mylonas, Nikos Virvilis, Marianthi Theoharidou, and Dimitris Gritzalis. An insider threat prediction model. In *Trust, Privacy and Security in Digital Business*, pages 26–37. Springer, 2010.

[71] Kaspersky. The icefog apt: A tale of cloak and three daggers (2013).

[72] Kaspersky. The regin platform: Nation-state ownage of gsm networks. 2014.

[73] Key. Honker union of china to launch network attacks against japan is a rumor. `http://www.chinahush.com/2010/09/15/honker-union-of-china-to-launch-network-attack-against-japan-is-a-rumor/`. Accessed: 2015-01-22.

[74] Swati Khandelwal. Chinese spies stole australia's new f-35 lightning-ii fighter jet design, snowden reveals. `http://thehackernews.com/2015/01/F-35-Lightning-II-fighter-Jet-Design.html`. Accessed: 2015-01-22.

[75] Mahmoud Khonji, Youssef Iraqi, and Andrew Jones. Mitigation of spear phishing attacks: A content-based authorship identification framework. In *Internet Technology and Secured Transactions (ICITST), 2011 International Conference for*, pages 416–421. IEEE, 2011.

[76] Si-Jung Kim, Do-Eun Cho, and Sang-Soo Yeo. Secure model against apt in m-connected scada network. *International Journal of Distributed Sensor Networks*, 2014, 2014.

[77] Yong-Ho Kim and Won Hyung Park. A study on cyber threat prediction based on intrusion detection event for apt attack detection. *Multimedia Tools and Applications*, 71(2):685–698, 2014.

[78] Engin Kirda and Christopher Kruegel. Protecting users against phishing attacks with antiphish. In *Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International*, volume 1, pages 517–524. IEEE, 2005.

[79] Hugo Krawczyk. Secret sharing made short. In *Advances in Cryptology—CRYPTO'93*, pages 136–146. Springer, 1994.

[80] Brian Krebs. Cards stolen in target breach flood underground markets. `http://krebsonsecurity.com/2013/12/cards-stolen-in-target-breach-flood-underground-markets/`. Accessed: 2015-01-22.

[81] CrySyS Lab. skywiper (a.k.a. flame a.k.a. flamer): A complex malware for targeted attacks. 2012.

[82] Andrey Ladikov. Why you shouldn't completely trust files signed with digital certificates. http://securelist.com/blog/security-policies/68593/why-you-shouldnt-completely-trust-files-signed-with-digital-certificates/. Accessed: 2015-02-02.

[83] Stevens Le Blond, Adina Uritesc, Cédric Gilbert, Zheng Leong Chua, Prateek Saxena, and Engin Kirda. A look at targeted attacks through the lense of an ngo. In *Proceedings of the 23rd USENIX conference on Security Symposium*, pages 543–558. USENIX Association, 2014.

[84] Christopher Levine. Conceptualizing financial loses as a result of advanced persistent threats. *Honors College Theses, Paper 122*, 2013.

[85] John Leyden. Latest snowden reveal: It was gchq that hacked belgian telco giant. http://www.theregister.co.uk/2013/09/20/gchq_belgacom_hack_link/. Accessed: 2015-01-29.

[86] Hung-Jen Liao, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-Yuan Tung. Intrusion detection system: A comprehensive review. *Journal of Network and Computer Applications*, 36(1):16 – 24, 2013.

[87] Alexander Liu, Cheryl Martin, Tom Hetherington, and Sara Matzner. A comparison of system call feature representations for insider threat detection. In *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC*, pages 340–347. IEEE, 2005.

[88] Shun-Te Liu, Yi-Ming Chen, and Shiou-Jing Lin. A novel search engine to uncover potential victims for apt investigations. In *Network and Parallel Computing*, pages 405–416. Springer, 2013.

[89] Xiaomei Liu. Research on prevention solution of advanced persistent threat. In *2014 2nd International Conference on Software Engineering, Knowledge Engineering and Information Engineering (SEKEIE 2014))*. Atlantis Press, 2014.

[90] Yali Liu, Cherita Corbett, Ken Chiang, Rennie Archibald, Biswanath Mukherjee, and Dipak Ghosal. Sidd: A framework for detecting sensitive data exfiltration by an insider attack. In *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on*, pages 1–10. IEEE, 2009.

[91] Zhuo Lu, Xiang Lu, Wenye Wang, and Cliff Wang. Review and evaluation of security threats on the communication networks in the smart grid. In *MILITARY COMMUNICATIONS CONFERENCE, 2010-MILCOM 2010*, pages 1830–1835. IEEE, 2010.

[92] APT Mandiant. Exposing one of china's cyber espionage units (feb. 2013).

[93] M Mandiant. Trends. 2014 threat report.

[94] William R Marczak, John Scott-Railton, Morgan MARQUISBOIRE, and Vern Paxson. When governments hack opponents: A look at actors and technology. In *Proceedings of the 23rd USENIX Security Symposium*, 2014.

[95] Michael. CNN Martinez. Allies spy on allies because a friend today may not be one tomorrow. `http://edition.cnn.com/2013/10/30/us/spying-on-allies-everybody-does-it/index.html`. Accessed: 2015-01-22.

[96] Kevin D Mitnick and William L Simon. *The art of deception: Controlling the human element of security*. John Wiley & Sons, 2011.

[97] Daesung Moon, Hyungjin Im, Jae Dong Lee, and Jong Hyuk Park. Mlds: Multi-layer defense system for preventing advanced persistent threats. *Symmetry*, 6(4):997–1010, 2014.

[98] Ellen Nakashima, Greg Miller, and Julie Tate. U.s., israel developed flame computer virus to slow iranian nuclear efforts, officials say. `http://www.washingtonpost.com/world/national-security/us-israel-developed-computer-virus-to-slow-iranian-nuclear-efforts-officials-say/2012/06/19/gJQA6xBPoV_story.html`. Accessed: 2015-01-29.

[99] Ryan Naraine. Us-cert warns of guest-to-host vm escape vulnerability. `http://www.zdnet.com/article/us-cert-warns-of-guest-to-host-vm-escape-vulnerability/`. Accessed: 2015-01-29.

[100] Nir Nissim, Aviad Cohen, Chanan Glezer, and Yuval Elovici. Detection of malicious pdf files and directions for enhancements: A state of the art survey. *Computers & Security*, 2014.

[101] NIST. Managing information security risk: Organization, mission, and information system view. *SP 800-39*, 2011.

[102] Norton. 2013 report.

[103] The Federal Bureau of Investigation. Five chinese military hackers charged with cyber espionage against u.s. `http://www.fbi.gov/news/news_blog/five-chinese-military-hackers-charged-with-cyber-espionage-against-u.s`. Accessed: 2015-01-29.

[104] Jon Oltsik. The esg cybersecurity maturity model. 2014.

[105] Alina Oprea, Zhou Li, Ting-Fang Yen, Sang Chin, and Sumayah Alrwais. Detection of early-stage enterprise infection by mining large-scale log data. *arXiv preprint arXiv:1411.5005*, 2014.

[106] Fernando C Colón Osorio, Ferenc Leitold, Dorottya Mike, Chris Pickard, Sveta Miladinov, and Anthony Arrott. Measuring the effectiveness of modern security products to detect and contain emerging threats—a consensus-based approach. In *Malicious and Unwanted Software:" The Americas"(MALWARE), 2013 8th International Conference on*, pages 27–34. IEEE, 2013.

[107] Abigail Paradise, Rami Puzis, and Asaf Shabtai. Anti-reconnaissance tools: Detecting targeted socialbots. 2014.

[108] Joon S Park and Joseph Giordano. Role-based profile analysis for scalable and accurate insider-anomaly detection. In *Performance, Computing, and Communications Conference, 2006. IPCCC 2006. 25th IEEE International*, pages 7–pp. IEEE, 2006.

[109] Bimal Parmar. Protecting against spear-phishing. *Computer Fraud & Security*, 2012(1):8–11, 2012.

[110] Bryan Parno, Jonathan M McCune, Dan Wendlandt, David G Andersen, and Adrian Perrig. Clamp: Practical prevention of large-scale data leaks. In *Security and Privacy, 2009 30th IEEE Symposium on*, pages 154–169. IEEE, 2009.

[111] Animesh Patcha and Jung-Min Park. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks*, 51(12):3448–3470, 2007.

[112] Michael O Rabin. Efficient dispersal of information for security, load balancing, and fault tolerance. *Journal of the ACM (JACM)*, 36(2):335–348, 1989.

[113] Federal Regulations. International traffic in arms regulations. `https://epic.org/crypto/export_controls/itar.html`. Accessed: 2015-01-22.

[114] Global Research and Analysis Team. The nettraveler (aka 'travnet'). 2013.

[115] Uri Rivner. Anatomy of an attack. *RSA [database online]*, 2011.

[116] Fran Rosch. Study finds mobile privacy concerns often traded for free apps. `https://community.norton.com/en/blogs/norton-protection-blog/study-finds-mobile-privacy-concerns-often-traded-free-apps`. Accessed: 2015-02-02.

[117] Marcel Rosenbach, Hilmar Schmundt, and Christian Stöcker. Source code similarities: Experts unmask 'regin' trojan as nsa tool. `http://www.spiegel.de/international/world/regin-malware-unmasked-as-nsa-tool-after-spiegel-publishes-source-code-a-1015255.html`. Accessed: 2015-02-18.

[118] Neil C Rowe and Han C Goh. Thwarting cyber-attack reconnaissance with inconsistency and deception. In *Information Assurance and Security Workshop, 2007. IAW'07. IEEE SMC*, pages 151–158. IEEE, 2007.

[119] Usman Asghar Sandhu, Sajjad Haider, Salman Naseer, and Obaid Ullah Ateeb. A survey of intrusion detection & prevention techniques. In *Shaheed Zulfiqar Ali Bhutto Institute of Science and Technology, Islamabad.(SZABIST), University of the Punjab Gujranwala Campus, 2011 International Conference on Information Communication and Management IPCSIT*, volume 16, 2011.

[120] E Eugene Schultz. A framework for understanding and predicting insider attacks. *Computers & Security*, 21(6):526–531, 2002.

[121] ISACA Cyber SecurityNexus. Advanced persistent threat awareness. `http://www.isaca.org/knowledge-center/research/researchdeliverables/pages/advanced-persistent-threats-awareness-study-results.aspx`, 2014. Online; accessed 15-October-2014.

[122] George Silowash, Dawn Cappelli, Andrew Moore, Randall Trzeciak, Timothy J Shimeall, and Lori Flynn. Common sense guide to mitigating insider threats 4th edition. Technical report, DTIC Document, 2012.

[123] Abhishek Singh and Zheng Bu. Hot knives through butter: Evading file-based sandboxes. *FireEye*, 2013.

[124] Florian Skopik, Ivo Friedberg, and Roman Fiedler. Dealing with advanced persistent threats in smart grid ict networks. In *Innovative Smart Grid Technologies Conference (ISGT), 2014 IEEE PES*, pages 1–5. IEEE, 2014.

[125] Robin Sommer and Vern Paxson. Outside the closed world: On using machine learning for network intrusion detection. In *Security and Privacy (SP), 2010 IEEE Symposium on*, pages 305–316. IEEE, 2010.

[126] Aditya K Sood and Richard J Enbody. Targeted cyberattacks: a superset of advanced persistent threats. *IEEE security & privacy*, 11(1):54–61, 2013.

[127] T.C. Sottek. Apple says it was attacked by hackers, will issue malware removal tool today. `http://www.theverge.com/2013/2/19/4005460/apples-computers-attacked-by-hackers-says-reuters`. Accessed: 2015-02-02.

[128] Paulo Sousa, Alysson Neves Bessani, Miguel Correia, Nuno Ferreira Neves, and Paulo Verissimo. Resilient intrusion tolerance through proactive and reactive recovery. In *Dependable Computing, 2007. PRDC 2007. 13th Pacific Rim International Symposium on*, pages 373–380. IEEE, 2007.

[129] SPIEGEL. Inside tao: Documents reveal top nsa hacking unit. `http://www.spiegel.de/international/world/the-nsa-uses-powerful-toolbox-in-effort-to-spy-on-global-networks-a-940969.html`. Accessed: 2015-01-27.

[130] Gianluca Stringhini and Olivier Thonnard. That ain't you: Detecting spearphishing emails before they are sent. *arXiv preprint arXiv:1410.6629*, 2014.

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

[132] Kaspersky Labs' Global Research & Analysis Team. What was that wiper thing? `http://securelist.com/blog/incidents/34088/what-was-that-wiper-thing-48/`. Accessed: 2015-01-29.

[133] TrendLabs APT Research Team et al. Spear-phishing email: Most favored apt attack bait. *Last accessed September*, 2:2013, 2012.

[134] Olivier Thonnard, Leyla Bilge, Gavin O'Gorman, Seán Kiernan, and Martin Lee. Industrial espionage and targeted attacks: Understanding the characteristics of an escalating threat. In *Research in Attacks, Intrusions, and Defenses*, pages 64–85. Springer, 2012.

[135] Satoru Torii, Masanobu Morinaga, Takashi Yoshioka, Takeaki Terada, and Yuki Unno. Multi-layered defense against advanced persistent threats (apt). *FUJITSU Sci. Tech. J*, 50(1):52–59, 2014.

[136] Forward-Looking Threat Research Team TrendMicro. Inside an apt campaign with multiple targets in india and japan. 2012.

[137] Sun Tzu. *The art of war.* Orange Publishing, 2013.

[138] Rohit Varma. Mcafee labs: Combating aurora. Technical report, 2010.

[139] Paulo Esteves Veríssimo, Nuno Ferreira Neves, and Miguel Pupo Correia. Intrusion-tolerant architectures: Concepts and design. In *Architecting Dependable Systems*, pages 3–36. Springer, 2003.

[140] Nart Villeneuve and James Bennett. Detecting apt activity with network traffic analysis. *Trend Micro Incorporated [pdf] Available at:¡ http://www. trend-micro. com/cloud-content/us/pdfs/securityintelligence/white-papers/wp-detecting-apt-activity-with-network-traffic-analysis. pdf¿ [Accessed 31 October 2014]*, 2012.

[141] Nikos Virvilis, Dimitris Gritzalis, and Theodoros Apostolopoulos. Trusted computing vs. advanced persistent threats: Can a defender win this game? In *Ubiquitous Intelligence and Computing, 2013 IEEE 10th International Conference on and 10th International Conference on Autonomic and Trusted Computing (UIC/ATC)*, pages 396–403. IEEE, 2013.

[142] WebSense. Advanced persistent threats and other advanced attacks: Threat analysis and defense strategies for smb, mid-size, and enterprise organizations. 2011.

[143] Jeremy Daniel Wendt. Omen: Identifying potential spear-phishing tar-gets before the email is sent. 2013.

[144] Lance Whitney. Nsa planted surveillance software on hard drives, report says. `http://www.cnet.com/news/nsa-planted-surveillance-software-on-hard-drives-report/`. Accessed: 2015-04-15.

[145] Wikipedia. Five ws. `http://en.wikipedia.org/wiki/Five_Ws`. Accessed: 2014-10-31.

[146] RSA Advanced Threat Intelligence Team Will Gragido. Lions at the watering hole – the “voho” affair. `https://blogs.rsa.com/lions-at-the-watering-hole-the-voho-affair`, 2012. Online; accessed 15-October-2014.

[147] Tyler Wrightson. *Advanced Persistent Threat Hacking : The Art and Science of Hacking Any Organization*. McGraw-Hill Osborne Media, 2014.

[148] Candid Wueest. Targeted attacks against the energy sector. *Symantec Security Response, Mountain View, CA*, 2014.

[149] Christos Xenakis and Christoforos Ntantogian. An advanced persistent threat in 3g networks: Attacking the home network from roaming networks. *Computers & Security*, 40:84–94, 2014.

[150] Kim Zetter. Researchers connect flame to us-israel stuxnet attack. `http://www.wired.com/2012/06/flame-tied-to-stuxnet/`. Accessed: 2015-01-29.

[151] Kim Zetter. *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*. Crown, 2014.