Hakin9

PRACTICAL PROTECTION

IT SECURITY MAGAZINE

VOL.13, NO. 10

INTRODUCTION TO ETHICAL HACKING: BECOME THE OWNER OF NETWORK

Cyber Threat Intelligence Model

PinkKite analysis

Brief Walkthrough of Concurrency Model in Go

AND MORE...

Hakin9

TEAM

Editor-in-Chief
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

Editors:

Marta Sienicka
sienicka.marta@hakin9.com

Dominika Zdrodowska
dominika.zdrodowska@eforensicsmag.com

Marta Strzelec
marta.strzelec@eforensicsmag.com

Bartek Adach
bartek.adach@pentestmag.com

Proofreader:
Lee McKenzie

Senior Consultant/Publisher:
Paweł Marciniak

CEO:
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

Marketing Director:
Joanna Kretowicz
joanna.kretowicz@eforensicsmag.com

DTP
Marta Sienicka
sienicka.marta@hakin9.com

Cover Design
Hiep Nguyen Duc

Publisher
Hakin9 Media Sp. z o.o.
02-676 Warszawa
ul. Postępu 17D
Phone: 1 917 338 3631

www.hakin9.org

All trademarks, trade names, or logos mentioned or used are the property of their respective owners.

The techniques described in our articles may only be used in private, local networks. The editors hold no responsibility for misuse of the presented techniques or consequent data loss.

Betatesters & Proofreaders

Lee McKenzie

Hammad Arshed

Martin Renaud

Olivier Caleff

Ali Abdollahi

Robert Fling

Paul Mellen

Bernhard Waldecker

Avi Benchimol

Tim Singletary

Richard Takács

Sunny Wear

Ivan Gutierrez Agramont

Jeff Smith

John Webb

David Molik

Tom Updegrove

Da Co

Dear Readers!

Welcome to the new issue of Hakin9. This month we would like to talk about Threat Intelligence and identifying potential cybersecurity attacks. The primary purpose of threat intelligence is to help organizations (or companies) understand the risks of the most common and severe external threats. The articles we chose for this publication will give you in-depth knowledge about this topic.

Our magazine wouldn't be the same without a few articles dedicated to hacking or breaking through security systems! That's why we have articles such as Introduction to ethical hacking where you will learn how to take control over a network or PinkKite analysis, focused on malware.

All that and much more awaits you in our newest edition. We would also want to thank all authors, reviewers and proofreaders for participating in this project.

Have a nice read, folks!

Regards,

Hakin9 Magazine Editorial Team

Table of Contents

Introduction to ethical hacking:

Become the owner of network

Adrian Rodriguez Garcia

ABOUT THE AUTHOR

Adrian Rodriguez Garcia

Adrian is a graduate in telecommunication engineering in the specialty of telematics and a graduate of the Master in security of the information and communications in the University of Seville.

I'm a fan of cybersecurity, especially those thematic directed to the fight against malware, reason by which I design all kind of solutions to prevent and mitigate any incident that can be produced in network systems. In addition, I'm a curious person who likes to study and test new technologies to the extreme to take full advantage of its features or to know the limitations and improve them.

In short, I enjoy the world of cybersecurity and new technologies where I feel happy and wanting to learn something new every day.

Contact: www.linkedin.com/in/adrian-rodriguez-garcia-64257698.

Email: adrrodgar@gmail.com

Ethical hacking is one of the most demanded branches of security today. For this reason, this article is going to show a case study where any person without knowledge can start in this field of computer security.

Introduction

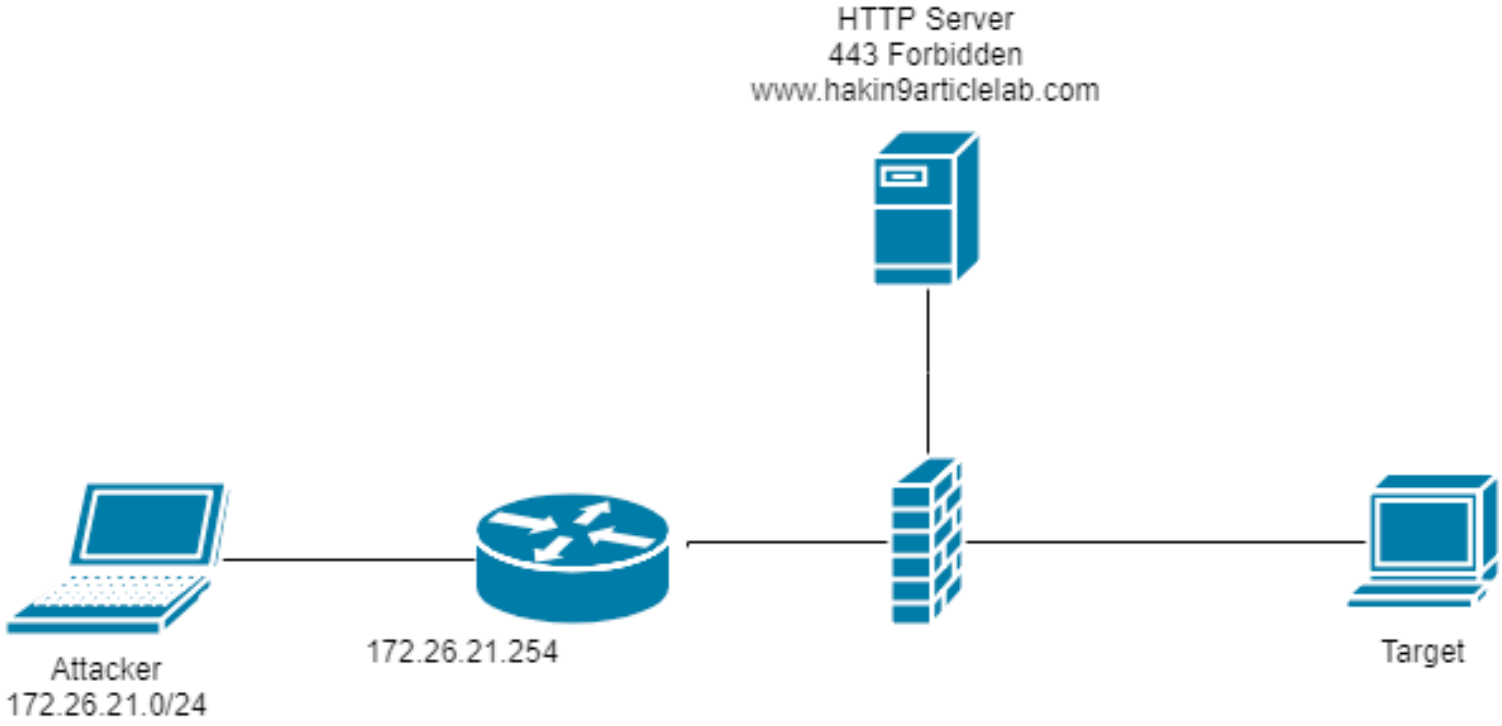
First we're going to talk about the lab that has been designed for this article and know how to act on it.

Then, with a series of tools and frameworks that will be seen throughout the article, we will begin the first stage of attack, which will consist of compromising an internal network element from an external network.

The next challenge will be to compromise an internal network using the pivoting method to finally make the attack persistent.

Lab introduction

The lab that has been prepared for this article is shown in the following image where we can see the network that is going to be used.



1 Lab

The idea is very clear. The web service of network is only accessible through some ports by an external computer due to firewall action that's between router and server . The target is to get the computer in the internal subnet by pivoting on the web server.

To achieve this goal, we will divide the attack in two parts. The first step will be to check the options to attack the web server and attack it later. The second step will be to check the internal network, check the computers and their technologies and attack to target.

Note that in this article, technologies (mostly software) have been used that allow the success of our goal. The objective of this article is to teach the necessary methodologies and technologies to someone who does not know anything and they can start in ethical hacking world.

The technologies that will be used are the following:

- **Nmap**: Network equipment analysis tool that obtains information about open ports, as well as the OS fingerprinting(is the process of gathering information that allows identifying the operating system on target computer). The objective is to explore the range of available networks and ports, as well as information related to systems.

- **Curl**: This tool allows us to send web requests to servers, receiving the corresponding answers. This software will help us to exploit shellshock vulnerabilities in this lab. Highlight that shellshock vulnerability is also known as Bash bug (Bash Remote Code Execution) which could allow an attacker to gain control over a targeted computer if exploited successfully.

- **Metasploit framework**: Framework that allows the analysis and operation of computers. The tools of Metasploit that can be most useful to us are:

    - **Meterpreter**: Terminal that allows us to have control over the attacked computer, obtaining information of it and allowing the injection of useful scripts or commands.

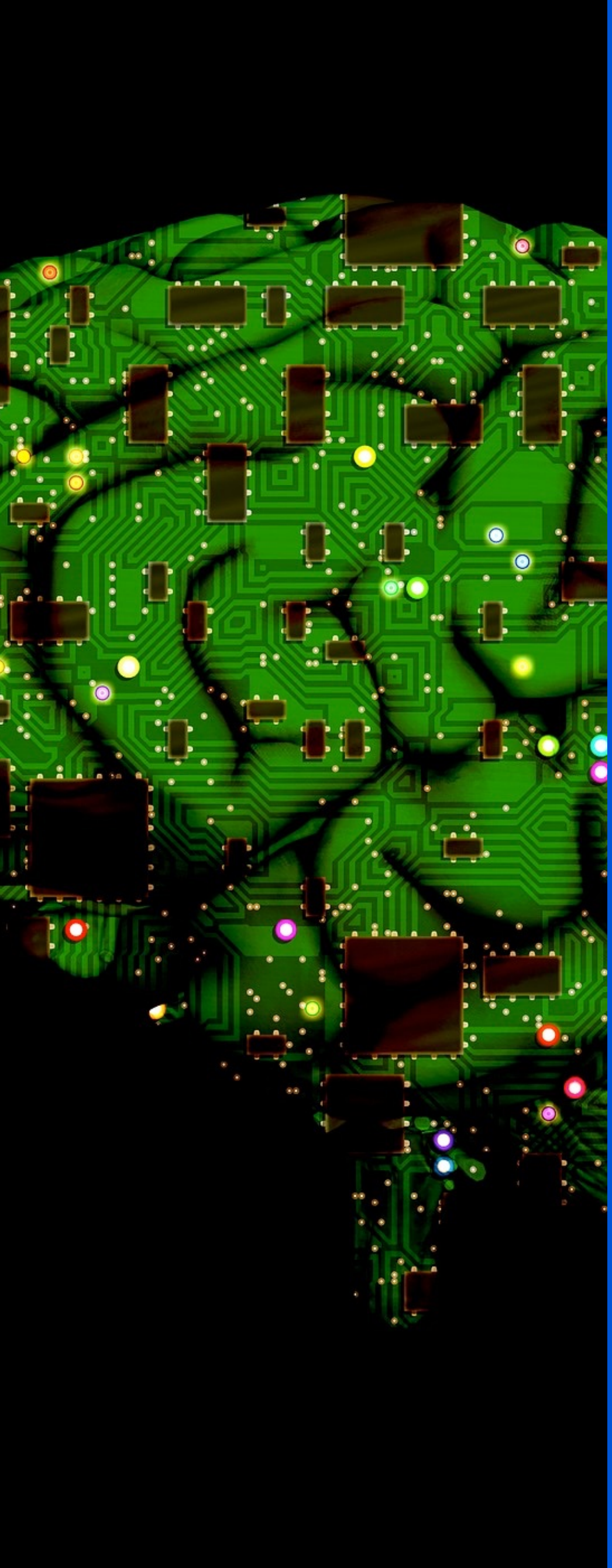
    - **Auxiliary**: Set of modules that allow the verification of vulnerabilities and external scanning of network computers.

    - **Msfvenom**: Tool used to create known exploits.

- **Netcat**: Tool used to establish connections with client - server structure. This software allows us to control a machine remotely from a netcat server.

- **Burp Suite**: Framework that allows us to perform an audit on an IP or domain. It is a tool with many possibilities to audits and very complete, so it's a difficult tool to explain. For this article, it's enough to know that Burp Suite performs vulnerability scanning, scanning of accessible files and directories and vulnerability exploitation tests. This is enough because Burp Suite has a professional version, a business version and a community version with an infinity of possibilities and options.

- **Proxychains**: Tools that allow us to obtain information about open ports, as well as fingerprint of operating system by pivoting technique.



Cyber Threat Intelligence Model:

An Evaluation of Taxonomies, Sharing Standards, and Ontologies within Cyber Threat Intelligence

Vasileios Mavroeidis

Siri Bromander

ABOUT THE AUTHOR

Vasileios Mavroeidis

PhD Research Fellow in Cyber Security - UiO/SecurityLab

ABOUT THE AUTHOR

Siri Bromander

Siri Bromander works as part of the Threat Intelligence and Incident Response group at mnemonic, while pursuing her PhD at the University of Oslo with her PhD project "Threat Ontologies for CyberSecurity Analytics (TOCSA)" and contributing to the research project "Semi-Automated Cyber Threat Intelligence (ACT). She has more than 10 years of work experience in IT security and information security research roles, including serving as Security Manager at mnemonic for five years.

Threat intelligence is the provision of evidence-based knowledge about existing or potential threats. Benefits of threat intelligence include improved efficiency and effectiveness in security operations in terms of detective and preventive capabilities. Successful threat intelligence within the cyber domain demands a knowledge base of threat information and an expressive way to represent this knowledge. This purpose is served by the use of taxonomies, sharing standards, and ontologies. This paper introduces the Cyber Threat Intelligence (CTI) model, which enables cyber defenders to explore their threat intelligence capabilities and understand their position against the ever-changing cyber threat landscape. In addition, we use our model to analyze and evaluate several existing taxonomies, sharing standards, and ontologies relevant to cyber threat intelligence. Our results show that the cyber security community lacks an ontology covering the complete spectrum of threat intelligence. To conclude, we argue the importance of developing a multi-layered cyber threat intelligence ontology based on the CTI model and the steps that should be taken under consideration, which are the foundation of our future work.

Index Terms

cyber threat intelligence, threat information sharing, cyber security, threat intelligence ontologies, cyber attack attribution, cyber threat detection, cyber threat prevention, knowledge representation

I. INTRODUCTION

The capabilities, persistence, and complexity of adversarial attacks in the present threat landscape result in a speed race between security analysts, incident responders, and threat actors. Coordinated cybercrime is at its peak. PwC's global economic crime survey of 2016 [1] reports that there are organizations that suffered cybercrime losses over $5 million, and of these nearly a third reported losses in excess of $100 million. In addition, Juniper Research [2] reports that cybercrime will increase the cost of data breaches to $2.1 trillion globally by 2019; four times the estimated cost of breaches in 2015.

Security analysts and incident responders need the right skills to recognize attacks before performing defense efforts. The development of adequate controls requires a thorough threat analysis, but most of the time, small and medium sized businesses have inadequate capabilities due to lack of skilled personnel and budget constraints. Threat intelligence is referred to as the task of gathering evidence-based knowledge, including context, mechanisms, indicators, implications and actionable advice, about an existing or emerging menace or hazard to assets that can be used to inform decisions regarding the subject's response to that menace or hazard[1]. Threat information reported and shared between security teams is overwhelming, making difficult its absorption and correlation to existing stored knowledge; as a result, threat intelligence vendors are increasingly shifting to ways of automating this process making threat analysis an available task. Analyzing and sharing threat data and threat information in an effective way requires common representation, standard formats and protocols for sharing, and a common understanding of the relevant concepts and terminology.

[1] *https://www.gartner.com/doc/2487216/definition-threat-intelligence*

A solution approach to this need is the use of artificial intelligence (AI), particularly the use of ontologies. An ontology is a form of knowledge representation that can integrate information coming from different sources. Working towards an ontology for cyber threat intelligence is not an easy task. Our research reports the following as the largest difficulties:

- Vaguely defined terminology leads to confusion among experts and additional work to extend or unify ontologies.
- Lack of formal standardized representation of relevant information results in strings of English prose, with no standard pattern. Standardizing well-defined taxonomies can eliminate this barrier.
- Lack of coherent relationships between the different layers of abstraction in ontologies. Modular ontologies containing several sub-ontologies need sound relationships between the different data points to leverage the power of semantics and reasoning. For example, to understand the behavior and the capabilities of a threat actor the connections and relationships between pieces of information must be sound.

This article evaluates taxonomies, sharing standards, and ontologies relevant to the task of creating an ontology for use within cyber threat intelligence. Some of the ontologies potentially can aid threat intelligence but initially have been introduced to address a specific domain within cyber security. Additionally, we pinpoint the relationship between our own Cyber Threat Intelligence model (CTI), the taxonomies, the sharing standards, and the ontologies discussed, aiming to classify them in terms of expressivity. Finally, we critically discuss the shortcomings of the present cyber threat intelligence ontology approaches and we address the directions that should be followed for their advancement.

II. METHODOLOGY

This section introduces two models related to threat detection maturity and cyber threat intelligence, respectively. The two models overlap and both can meet different needs that are explained in the next two subsequent subsections. The CyberThreat Intelligence model is the basis of the evaluation process conducted in this paper.

A. The Detection Maturity Level Model - DML

Ryan Stillions proposed the DML model in several blog postings in 2014 [3]. The model was originally used to describe the maturity of an organization in terms of their ability to consume and act upon given threat information. Threat information can include indicators of compromise, tactics techniques and procedures of an actor (TTPs), threat intelligence reports and many more. In 2016, we extended this model by adding an additional level (9) ”Identity” and presented it for use in semantic representation of cyber threats [4].

Preventing Poisoning Attacks on AI based Threat Intelligence Systems

Nitika Khurana

Sudip Mittal

Anupam Joshi

ABOUT THE AUTHOR

Nitika Khurana

ABOUT THE AUTHOR

Sudip Mittal

Sudip Mittal is a Ph.D. candidate at the Accelerating Cognitive Cyber Security Research Lab and Ebiquity Research Lab, in University of Maryland Baltimore County. He holds a M.Tech and a B.Tech degree in computer science from IIIT Delhi. His current research is focussed on Artificial Intelligence for Cybersecurity. He has also previously worked with Center for Hybrid Multicore Productivity Research (CHMPR) and Cybersecurity Education and Research Centre (CERC@IIITD).

ABOUT THE AUTHOR

Anupam Joshi

Dr. Joshi got involved in research activities, which lead him to Purdue University, from where he obtained his Ph.D degree in Computer Science. After a stint at Purdue as a research faculty, he headed off to the University of Missouri, and spent two years there as Assistant Professor of Computer Engineering and Computer Science. Currently, he is found roaming the campus of the University of Maryland, Baltimore County. After assisting and then associating with Professors, he was given the right to Profess on his own in 2005. He has since been made the Oros Family Professor, Director of the Cybersecurity Center, and Chair of the Department. He was elected a Fellow of the IEEE. He continues to ask many questions, though he concentrates on answering only some of them.

As AI systems become more ubiquitous, securing them becomes an emerging challenge. Over the years, with the surge in online social media use and the data available for analysis, AI systems have been built to extract, represent and use this information. The credibility of this information extracted from open sources, however, can often be questionable. Malicious or incorrect information can cause a loss of money, reputation, and resources; and in certain situations, pose a threat to human life. In this paper, we use an ensembled semi-supervised approach to determine the credibility of Reddit posts by estimating their reputation score to ensure the validity of information ingested by AI systems. We demonstrate our approach in the cybersecurity domain, where security analysts utilize these systems to determine possible threats by analyzing the data scattered on social media websites, forums, blogs, etc.

Index Terms — Cybersecurity, Artificial Intelligence, Threat Intelligence, Poisoning Attacks, Credibility

I. INTRODUCTION

Artificial Intelligence (AI) is widely utilized in diverse domains of industries, like finance, cars, cybersecurity, education, etc. AI systems are 'trained' to learn complex problems and automate them for a larger scale. These systems need training data, which is generally extracted and represented in a form that best suits the problem. One such source of data is overt or, in a traditional cybersecurity sense, a part of the 'Open-source Intelligence' (OSINT) [25]. OSINT includes data from sources such as newspapers, blogs, discussion groups, radio, social media websites, press conferences, journals, technical reports, etc. Online Social Media (OSM) is an OSINT source providing data that is ingested by AI tools working in various fields, like finance [15] and cybersecurity [21]. Some of the most commonly used OSM are Twitter, Reddit[1], etc.

In cybersecurity, threat intelligence can be mined using traditional sources like NIST's National Vulnerability Database (NVD)[2], United States Computer Emergency Readiness Team (US-CERT)[3], etc. Other sources that are more non-traditional are Twitter, Reddit, blogs, and news. Non-traditional sources are faster than the traditional ones. There is a significant gap between initial vulnerability announcement and NVD release [24]. Vulnerability threat intelligence appears first on non-traditional sources [23]. Mining non-traditional sources is becoming really important. In our previous work, we have developed CyberTwitter [21] and Cyber-All-Intel [22] systems that mine threat intelligence from various OSINT sources. The systems then represent cybersecurity intelligence in knowledge graphs and vector spaces so it can be used by artificial intelligence based cyber-defense systems.

A new class of 'Analyst Augmentation Systems' are being developed. More security analysts use these Artificial Intelligence based organizational cyber-defense systems to listen for threat intelligence mined from traditional and non-traditional sources, identify new vulnerabilities, analyze network and endpoint activity, find evidence of preplanned attacks and hints of data breaches.

[1]*https://www.twitter.com, https://www.reddit.com*

[2]*https://nvd.nist.gov/*

[3]https://www.us-cert.gov/

Should Chess Players Learn Computer Security?

Gildas Avoine,

Cedric´ Lauradoux,

Rolando Trujillo-Rasua

ABOUT THE AUTHOR

Gildas Avoine

Gildas Avoine is a professor of information security and cryptography at INSA Rennes (France). He is the co-leader of the research group on Embedded Security and Cryptography (EMSEC) at IRISA Rennes. Previously, he was PhD student at EPFL (Switzerland), a postdoctoral researcher at the MIT (USA), and a professor at UCL (Belgium).

ABOUT THE AUTHOR

Cedric´ Lauradoux

Cedric´ Lauradoux is a junior researcher in team Privatics at INRIA Grenoble - Rhône-Alpes (France). He is working on privacy and data protection: pseudonymization, re-identification and the practice of the GDPR are his main interests. He is the author of a MOOC on privacy which was followed by more than 20000 students.

ABOUT THE AUTHOR

Rolando Trujillo-Rasua

Rolando is a lecturer in Cyber Security at Deakin University (Australia). He obtained a Master's and PhD degree in Computer Engineering from Rovira i Virgili University (Spain), and shortly after joined the University of Luxembourg as a Postdoctoral researcher. His research interests span the areas of formal methods, computer security and privacy protection.

The main concern of chess referees is to prevent players from biasing the outcome of the game by either colluding or receiving external advices. Preventing third parties from interfering in a game is challenging given that communication technologies are steadily improved and miniaturized. Chess actually faces similar threats to those already encountered in computer security. We describe chess frauds and link them to their analogues in the digital world. Based on these transpositions, we advocate for a set of countermeasures to enforce fairness in chess.

Index Terms — Security, Chess, Fraud.

1. INTRODUCTION

Chess still fascinates generations of computer scientists. Many great researchers, such as John von Neumann, Claude Shannon, and Alan Turing, to name a few, have spent time studying chess programming. John Conway was also attracted by the game of chess and, in general, by combinatorial game theory. He introduced a popular chess fraud known as the *chess grandmaster problem*, where a little girl, Anne-Louise, who has never heard of chess, wants to face two grandmasters, Bobby Fischer and Boris Spassky, in correspondence chess. Has she lost her mind? Not really, she has a clever strategy which consists in playing Black against Fischer and White against Spassky. Once Fischer sends his first move, Anne-Louise copies this move and sends it to Spassky. Then, she waits Spassky's move and forwards it to Fischer. And so on, until she either wins one of the games or draws both of them. Anne-Louise only relays the moves between the two grandmasters. So the two grandmasters are indeed playing against each other, instead of playing against a little girl as one would expect.

Conway's work on the chess grandmaster problem was pursued and extended to authentication protocols by Desmedt, Goutier and Bengio[3] in order to break the Feige-Fiat-Shamir protocol[4]. The attack was called *mafia fraud*, as a reference to the famous Shamir's claim: "I can go to a mafia-owned store a million times and they will not be able to misrepresent themselves as me." Desmedt *et al.* proved that Shamir was wrong via a simple application of Conway's chess grandmaster problem to authentication protocols. Since then, this attack has been used in various domains such as contactless credit cards, electronic passports, vehicle keyless remote systems, and wireless sensor networks.

The origin of this article comes from recent news about cheating in top-level chess tournaments. A famous case is the Georgian chess champion and grandmaster Gaioz Nigalidze who was caught in April 2015. He used a smartphone with a headset in the restroom during the Dubai Open Chess Tournament. He was banned for three years and his grandmaster title was revoked.

In 2010, several French grandmasters, including the coach of the national team, cheated during the chess Olympiad at Khanty-Mansiysk. The grandmaster Sebastien´ Feller was helped by his fellows Arnaud Hauchard and Cyril Marzolo to win his games by using a combination of cell phones, computer, and body language to inform the grandmaster of the best moves to play.

In the same year, Veselin Topalov accused Vlad´ımir Kramnik´ of cheating during their match for the world chess title at Elista. Topalov's accusations against Kramnik´ were motivated by his opponent's suspicious behavior: he visited the

Formjacking:

A major threat for online shoppers

Sayani Banerjee

ABOUT THE AUTHOR

Sayani Banerjee

Sayani Banerjee is a Cyber Security Researcher and Blogger currently working for ISOAH Data Securities Pvt. Ltd. India. Her interests revolve around active research on penetration testing and new exploits & vulnerabilities across the cyber world with a penchant for penning down security articles. She can be reached on twitter: https://twitter.com/sayanisays and quora: https://www.quora.com/profile/Sayani-Banerjee-27

The festive frenzy is not over. And the New Year is here with hopes and promises along with smarter technical innovations, and more dangerous security threats.

Among cakes and ale, celebrations and happiness around – it's all about sharing the joy and spreading smiles. Buying something for ourselves or gifting to our near and dear ones is part of the festivities. But in this fast-paced world, where everything you desire is just a click away from coming to your doorstep, why not embrace it?

As digitalization is taking over, online shopping sites are gaining popularity. In fact, online shopping has changed the way we used to shop. From electronic gadgets to fashion apparel, even groceries are available online and changing the shopping experience for the customers. The impressive returns make it a 'must have' platform for retailers and brands.

The festive season is synonymous with online shopping and cyber criminals are well aware of the fact. While shopping, we tend to believe that the site is secure and share a lot of personal and financial information without worrying. But before making any online transaction, be very cautious. Hackers can wash off your account within a blink of an eye! Such attacks are called 'Formjacking'.

Though it is not a new technique, recently the attacks have become more sophisticated and increased dramatically.

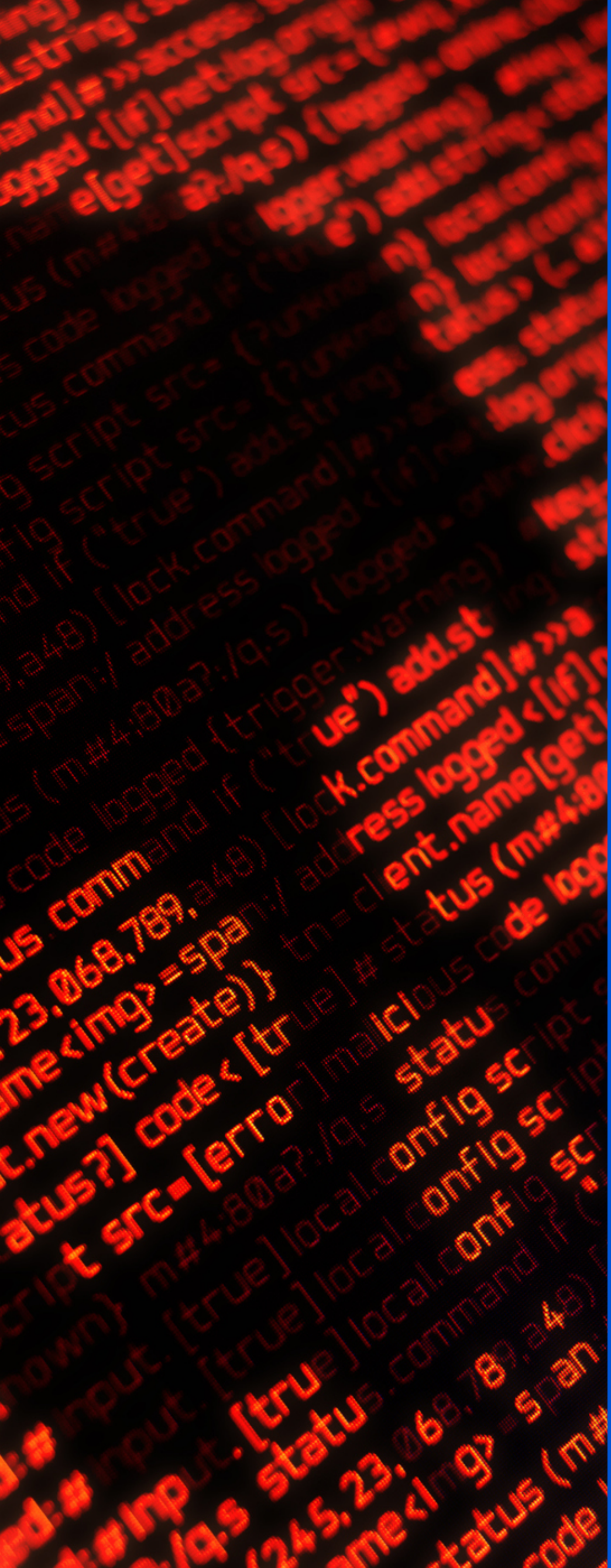
Previously, 'phishing' was the most popular way to steal sensitive financial information from consumers where malicious links play an important part. Also, there are plenty of examples where hackers cloned an ATM card and stole all the money from the account. But Formjacking is a completely different threat altogether.

How does this work?

There are two main reasons attackers are able to take advantage of this vulnerability. First, while developing the web apps, adequate attention is not given to the security of that site. And secondly, automated website vulnerability scanners are not being used. Companies often find it expensive or irrelevant to hire white hat hackers to safeguard their website.

In Formjacking, basically, a malicious JavaScript code is injected to steal financial information, like credit or debit card details, etc., from the payment gateway of e-commerce sites.

"Digital theft is likely to increase during Festive seasons," said Cyber Expert Sandeep Sengupta, Director - Indian school of Anti Hacking when asked about Formjacking. "A new form of attack called 'form jacking' has been predominantly used where, in an e-commerce site, when someone clicks 'submit' or its equivalent call-to-action button after entering their details into a website's payment form, malicious JavaScript code that has been injected there by the cyber criminals collects all the data that has been entered. The sensitive information includes the user's name, address, payment card details, etc. The information is then sent to the attacker's servers where they can exploit it in many ways including stealing money or selling the information to other criminals on the dark web", said Mr. Sengupta.



PinkKite analysis

Nikolaos Tsapakis

ABOUT THE AUTHOR

Nikolaos Tsapakis

Nikolaos Tsapakis works as a Security Researcher in Citrix. He had also been working as Threat Analysis Engineer in Symantec and Software Engineer in NCR and ATEBank.

***Note** : This article is by no means related to my current employer and only represents me and my opinions.*

1. Introduction

In this paper, I will analyze a POS malware named PinkKite. During analysis, you will learn the internals of magnetic stripe cards and basic functionality of malware targeting POS. The reader should already have a basic knowledge of assembly on a Windows OS environment and be familiar with tools used in reverse engineering.

2. Analysis tools

The tools used during the analysis are the following. The References section includes links for downloading the tools.

X64dbg	64 bit debugger used as dynamic analysis tool
IDA 7.0 Free	Static analysis tool
VirtualBox	VM for running sample

3. Sample

I used the following sample for analysis.

SHA2	9d28fa89f1a24228509fb0516ca2b01f026a4cb53d9b5b1a705ec07a1967a0c5
MD5	0ff7d6a0f54d18cadd18a62d8c94ea71
SHA1	e4fa892f2252e562001b28b1c6f407f80832db62
Size	5,632 bytes

The sample is available in repositories like VirusTotal. It is x64 architecture. Also, notice the size of the sample, which appears to be almost 6KB. That is a small size for such a “modern” malware.

4. Main functionality

By opening the sample in IDA notice that code is clear, without any obfuscation. Imports indicate that the sample is able to enumerate running processes, read data from each process and send data out through the network. From the imports, there is no indication that sample receives data back. Thus, one can assume that an attacker cannot control the sample through network.

Following is the list of the most important imports with comments on functionality:

Bad practices that will make your SIEM and SOC implementation fail

Luis Escobar

ABOUT THE AUTHOR

Luis Escobar

Luis Escobar has an extensive and broad career in cybersecurity and more specifically in Cyber-Defense.

Started on Security at the early age of 14 years old with his first home computer researching with war-dialers, later blue-boxes (captain crunch) and black-boxes.

Started on SOCs at Bruce Schneier Counterpane, pioneers in SOCs and SIEM development first SOC and SIEM in history. Worked also at ArcSight, European Parliament, Projects for Airbus, Oil and Gas leading companies worldwide, telecoms and many other industries.

Always involved in multicultural projects worldwide and promoting ethical business.

In all these years of experience in the SOC, SIEM and Incident Response arena, working on different SOCs and with different SIEMs, since the time of Bruce Schneier's Counterpane, later with ArcSight, the design, implementation and audit of so many others that I have already lost the count (I believe more than 30 SOCs?), I want to share today with you the bad practices and fake ideas that will make your SIEM and SOC implementation fail.

- Building the walls of shame: When people try to hide information from others, promoting mediocrity. Managers that permit this kind of attitude enforce mediocrity among their collaborators. Hopefully, you won't find any of these mediocre verging on bad managers.
- Avoid documentation: Not having documentation enforces mistakes and it is a strong reason of lack of maturity and a stopper for improvement. This point in somehow related to the former and vice-versa.
- Keep improvising: This is a must in some cultures that prevents structure in your security operations and the success of them.
- Choosing your tools just because they are fashionable: Your tools should serve your organization not your ego. I choose this flavor because it is expensive. I choose that other flavor because I saw it in a Garner's magic quadrant and if it is there it should be good... But, would it be good for you?
- Oppose improvement: Many times organizations, leaders and security analysts are afraid of change, so afraid that it prevents improvements. They prefer to continue committing the same mistakes day after day instead of investing some time thinking how to improve operations and avoid mistakes. The typical "we have always done it this way" when a weakness is found is just a barrier keeping you from improvement and progress.
- Provoke Monotony: This causes analyst to become burned-out. So you are losing your most valuable resources in your SOC. It is also related to the former point because when somebody finds the same problems every day because "we are already used to doing things in the wrong way in our organization", it enforces laziness and incompetence.

The former points provoke stagnation and, besides making your SIEM and SOC implementations fail, they will for sure make all your security operations fail also. That said, other reasons why your SIEM implementation fails are:

- You believe that implementing a SIEM is an easy, short task consisting of: install, next, next...
- You believe that having a SIEM is having a device that shows a green light on the corner. Have you ever thought of configuring your SIEM, maintaining it, etc.? So you think that SIEM is a magic intelligent box that will substitute all your security department and without any effort.
- You make a poor SIEM selection. Have you ever thought: how a SIEM could help you or what a SIEM is used for?

- You don't know what SIEM is and you are still stubborn enough to engage in a purchase just because you have heard that it is the magic autonomous tool that will work by itself.

Besides all that, you will need resources like time, people with real experience and expertise on SIEMs and SOCs, budget, a structured approach, and, last but not least, big doses of patience.

SOC Services: What if your service provider is selling you smoke and mirrors?

Luis Escobar

ABOUT THE AUTHOR

Luis Escobar

Luis Escobar has an extensive and broad career in cybersecurity and more specifically in Cyber-Defense.

Started on Security at the early age of 14 years old with his first home computer researching with war-dialers, later blue-boxes (captain crunch) and black-boxes.

Started on SOCs at Bruce Schneier Counterpane, pioneers in SOCs and SIEM development first SOC and SIEM in history. Worked also at ArcSight, European Parliament, Projects for Airbus, Oil and Gas leading companies worldwide, telecoms and many other industries.

Always involved in multicultural projects worldwide and promoting ethical business.

It might be a surprise for you but, unfortunately, if you are a customer of third-party SOC services, you are probably wasting your money and leaving your business exposed to external threats. One of the biggest concerns of companies that contract an external SOC service is that they don't actually have enough information about their risks and the best approach to mitigate them. The most important thing you should keep in mind related to SOCs: "If you don't have an Operational SOC, YOU DON'T HAVE A SOC."

Selling smoke and mirrors

Some years ago many service providers began offering security services as Security Operation Centers following the movement started in the USA with the pioneers on SOCs and SIEMs. However, during the process, something was forgotten. Those providers only focused on their own profits, missing the core concept of the Security Operation Center.

The selection of your SOC Services

In the implementation of your SOC, you will need make a good choice of:

1. The tools you are going to use in your SOC
2. The professional "People" that will operate your SOC
3. Your SOC structure/organization
4. The services that your SOC will operate

In this briefing, we will see just some of the aspects when designing your SOC services. We will see the rest of the aspects of an operational SOC in other articles if I finally have time to do so and the readers demand it, of course, because at the end, the only and ultimate purpose of this small article/briefing is to bring a bit of light to the readers on this complex task that will last long months of effort and good focus.

Well, now is the time to take into account the type of SOC you are going to implement. In this sense, we can have several different options that will determine and define some aspects of your future SOC related to the phases of design, implementation, and exploitation. Some options you might be thinking of are:

1. Internal SOC
2. SOC that gives service to external customers (as an MSSP for example)
3. Global SOC 24x7

Brief Walkthrough of Concurrency Model in Go

Souvik Haldar

ABOUT THE AUTHOR

Souvik Haldar

Souvik is a skilled backend developer with Go and Python as primary weapon in his arsenal. He is also into the DevOps world mostly using Ansible. He can create and manipulate geo-spatial databases using postGIS over postgreSQL. When not coding, he is either on long bike rides or partying!

Linkedin - https://www.linkedin.com/in/souvikhaldar/

Introduction

Around the year 2005, CPU manufacturers reached the limit of increasing the speed of the processors physically, which may also be thought of as reaching the saturation point of *Moore's Law*. Hence they started adding more cores to the CPU viz. Dual Core processor, Quad Core processor, etc., for enhancing the performance.

Unfortunately, older languages are not very efficient at exploiting the benefits of the modern hardware comprising of multi core CPUs. Here, Go fills the need by having concurrency support built in, which can very efficiently make use of the multiple cores for better performance.

This article is a high level and brief walkthrough of the concurrency model of Go.

Concurrency in Go

Goroutines are light weight threads that are managed by the Go runtime.

`go f(x,y)` starts are new goroutine running `f(x,y)`.

Simple example :

```
package main

import (
     "fmt"
     "time"
)

func say(s string) {
     for i := 0; i < 5; i++ {
          time.Sleep(100 * time.Millisecond)
          fmt.Println(s)
     }
}

func main() {
     go say("world")
     say("hello")
}
```

Playground- https://play.golang.org/p/fosLl05L4f6

The sample code above will print "world" and "hello" five times after an interval of 100 milliseconds in a random

Cyber Threat Intelligence: weaponizing cyber defense

Debojyoti Chakraborty

ABOUT THE AUTHOR

Debojyoti Chakraborty

Debojyoti Chakraborty, a security researcher, his area of expertise in Threat intelligence, Incident Response, Vulnerability Assessment, Data Analytics. He has been working as a Threat Investigator for Paladion Network since they have introduced Artificial Threat Intelligence. Apart from that he is an active member of null open security committee, security blogger and author of thecybersploit.com. You can reach out to him at debojyoti.chakraborty19@gmail.com

The ability to dig in deep to understand what a person really cares about or who they are, without their knowledge, is a valuable skill set in the Cyber Intelligence world. The data gathered, such as credentials, personal information, credit card or banking details and access to personal devices (mobile, laptop, etc.), could be used to steal money, or destroy a reputation. It is also illegal. Similarly, for an organization, information gathered related to revenue and profit, company shareholders, brand value, equity, market share and customer details could be used to cause untold amount of damage. A physical battlefield and cyber warfare pose a similar threat to mankind. Cyber threats could be carried out by a threat actor, leveraging the targets' potential to exploit their vulnerabilities. History has shown that battles have always had strategies and tactics to leverage the enemy's infrastructure for attack and their destruction, quite similar to cyber-attacks. However, while a physical battle takes time, money and massive effort to change the battle strategy, cyber-attacks are very different. Small groups with little money, but intense skill, can take down larger opponents and quickly pivot to attack or defend as the cyber strategy changes. Every time an adversary comes up with different techniques, procedures, communication processes or knowledge while attacking a target or defending itself from an attack, the tide of the cyber battle can change dramatically in seconds. How can you defend against an enemy so agile and difficult to detect? The following article will shed some light on this and give you a few cyber tools to defend against cyber threat intelligence gathering.

What is cyber threat intelligence?

Sun Tzu once said, "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." Threat intelligence is all about knowing your security threats, identifying them, and making a proper preventive decision.

It is not something that comes inside an Excel sheet or anything you can get by following some template. It requires an organization to understand their capabilities and that of the threat actors and their adversaries. An organization is an easy target for a malicious actor if the organization does not understand its assets, personnel, infrastructure and the operational process. You must first seek to know yourself.

The core purpose of Threat Intelligence is to analyse and process data about identified threats. A specific intelligence type must follow the intelligence life cycle of planning and direction, collection, processing, analysis and production, dispersion and integration of the information.

For better understanding, let's take an example. We all might have heard about "GameOver Zeus", one of the most effective cyber-attacks ever that infected thousands of people world wide. The operation created zombies which stretched the network to the level where it reached nearly one million bots. Operators used those bots for more than two years to steal millions of dollars from banks all over the world using CryptoLocker ransomware.