National College of Ireland

Integration of Elastic Search and Kibana SIEM for Malware Detection.

MSc Research Project
MSc in Cyber Security

Bhanu Prakash Rayabandi
Student ID: 21189731

School of Computing
National College of Ireland

Supervisor: Michael Pantridge

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	…Bhanu Prakash Rayabandi….
Student ID:	2118971
Programme:	……Msc in Cybersecurity…… **Year:** 2024
Module:	Research Project
Supervisor:	Michael Pantridge
Submission Due Date:	14/12/2023
Project Title:	Integration of Elastic Search and Kibana SIEM for Malware Detection.
Word Count:	5603 **Page Count**……21……

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: R.Bhanu……

Date: 12/12/2023 ……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Integration of Elastic Search and Kibana SIEM for Malware Detection

Bhanu Prakash Rayabandi
21189731

Abstract

Today, with computers being a big part of our lives, attackers create new approaches and tools specifically aimed at these systems. Lots of papers have been written about different security methods to identify these troublemakers in company computer setups. But, now a days, significant number of these security methods are outdated, and not as effective due to these main reasons: they mainly emphasis on either network or endpoint security, lacking a structured approach; moreover, they were too simple and vulnerable. To address these shortcomings, an integrated approach that employs a combination of proactive methods and predictive threat mechanisms would result in fast detection and an immediate response using custom rules. This novel method is suitable for small and medium-sized businesses, incorporating the integration of Elastic Search and Kibana with prebuilt detection rules from Elastic. Additionally, I integrated endpoint security also created sigma rules for the purpose of detecting malware. Thus, we put together an exhaustive system for malware detection and analysis both at the network and endpoint level. In this paper, we successfully analysed a series of malware attacks using techniques from the MITRE ATT&CK matrix and was able to create custom sigma rules and alerts using the Elastic search and Kibana. I have integrated windows elastic agent collect metrics and logs from your windows machine. Then visualize that data in Kibana, create custom sigma rules by querying the logs files coming from elastic agent to create alerts for the malwares.

Keywords— Elastic Search, Kibana, Sigma Rules. Malwares, Custom Query, Cyber Kill Chain, MITRE ATT&CK.

1 Introduction

Our society increasingly depends on technology, a trend showing no signs of slowing down. Enterprises are consistently storing larger volumes of data on the Internet, encompassing platforms such as social media and cloud storage. This becomes an attractive target for attackers seeking to acquire this information through various means.

Take, for example, the scenario of a malware attack. The attacker typically engages in activities like stealing sensitive data, exploiting system vulnerabilities, establishing remote control over the compromised system, encrypting files for ransom, causing system disruptions, and ensuring persistent access for future unauthorized actions. Transferring all

collected information over the network at once would be risky since the traffic volume will vastly increase. Gorment et al., 2023
The beaconing behavior they adopt in this situation is hard to detect, but this is where the custom sigma rules come into play. As many security experts say, prevention is ideal, but detection is a must. Therefore, we propose a combined solution that aims to use a mix of proactive techniques and threat hunting to bring fast detection. The system is designed for small and mid-size enterprises and uses Elastic search and Kibana integrated with endpoint security and custom crafted sigma rules which are further integrated into elastic for detection and creating an alert. This paper focuses on the development of an exhaustive system for malware detection using Elastic Search and Kibana. We achieve a series of malware attacks where each stage Fig.1 (Entry, Traffic Distribution, Exploit, Infection, Execution)

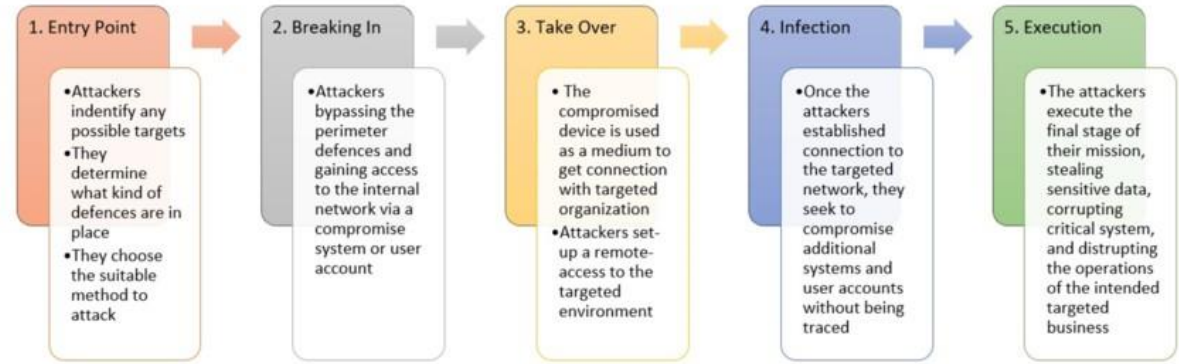


Fig.1: Stages of Malware Attack

is performed through techniques from the MITRE ATT&CK matrix. **Table 1**. (Gorment et al., 2023Al-Shaer, Spring and Christou, 2020)

Stage	**MITRE ATT&CK Techniques**
Entry (Initial Access)	Spear-phishing emails (T1193) Drive-by Compromise (T1189) Exploit Public-Facing Application (T1192)
Traffic Distribution	Command and Control (C2) Communication (T1043) Data Obfuscation (T1001) Domain Generation Algorithms (DGA) (T1568)
Exploit	Exploitation for Client Execution (T1203) Exploitation of Remote Services (T1210) Exploitation of Software Vulnerability (T1209)
Infection (Delivery, Dropper)	Malicious File (T1204) Scripting (T1064) Malicious Link (T1192) PowerShell (T1086)
Execution	Command and Scripting Interpreter (T1059) PowerShell (T1086) Scripting (T1064) Scheduled Task (T1053)

To analyse the PowerShell command logs, Sysmon logs , events logs and network logs coming from windows elastic agent contain malware , initially create virtual setup using three virtual machines, one, is dedicated from elastic search and Kibana, in which elastic search is the database Kibana is going to be the web interface to the Database, two, dedicated for integration of elastic i.e; Fleet Management this is going to be able to manage all the agents the agents we install on our windows vm and that configures things like having the windows vm send logs to the elastic database and the Fleet Management just automates that process so if you make a change it automatically pushes the configuration to all your agents so you don't have to keep pushing updates one at a time.

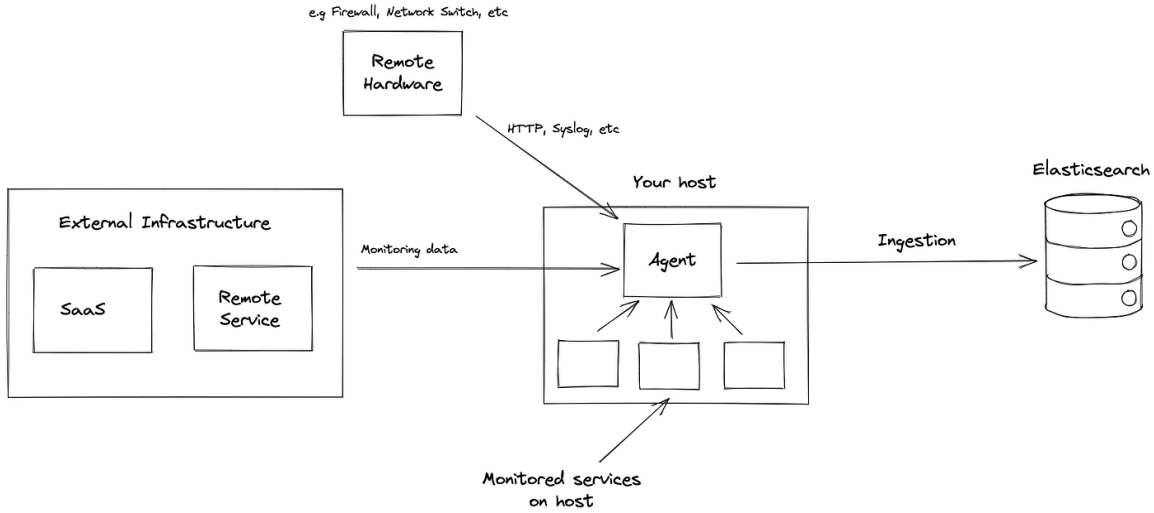


Figure 2: Elastic Agent Architecture.

In Security section of Detection Rules (SIEM), loaded elastic prebuilt detection rules which run in the background and create alerts when their conditions are met. By default, all prebuilt rules except the Endpoint Security rule are disabled. You can select additional rules you want to enable. I integrated Elasticsearch endpoint security, this is just going to have a bunch of rules in elastic for detections like malware or any ransomware. The additional rules are the rules which I have created initially as a generic rules which are independent of SIEM which are **sigma rules** for malware detection by analysis windows logs which are further filtered into Sysmon logs, events logs, PowerShell command logs and network logs by considering the IOC's like file path, network destination IP, registry path keyword, and process name by executing the different types of emotet, trickbot and IcedId malware. Lastly, integrating those rules into elastic search by writing the rules in KQL Query language to detect the malware and give alert. Figure.3.

Research Question?

"What is the impact of crafting and integrating custom Sigma rules in Elasticsearch for malware detection, specifically examining the effectiveness, efficiency, and adaptability of these rules in identifying and responding to malware variants such as Emotet, Trickbot, and IcedID?"

Understanding the malware in order to create rules at every step to stop it:

Trick Bot is a versatile banking Trojan discovered in 2016, known for stealing banking info, credentials, and more. (Malwarebytes, n.d.)It spreads through malicious spam campaigns, using URLs or infected attachments. Once on a system, it exploits SMB vulnerabilities to move laterally and gain access to networks. TrickBot can also be dropped by other malware like Emotet. Initially targeting a broad audience, it has become more specific, often masquerading as tax-themed spam. Notably, it has been found collecting email and messenger credentials from millions of users, affecting platforms like Gmail, Hotmail, Yahoo, AOL, and MSN.

Emotet is a Trojan spread through spam emails, using malicious attachments or links. It evolves, employing JavaScript and later macro-enabled documents to deliver its payload from command-and-control servers. (MalwareBytes, 2022), It detects virtual machines, laying dormant to evade analysis. Emotet updates via C&C servers, enabling the installation of new versions or additional malware. It primarily spreads through malspam, sending itself to contacts, making it appear less like spam. It also uses brute-force attacks on connected networks, exploiting weak passwords. It targets individuals, companies, and government entities globally. It steals banking logins, financial data, and Bitcoin wallets. Noteworthy attacks include the City of Allentown, costing over $1M to fix. It has hit the U.S., Europe, Canada, and the U.K., evolving to download and deliver other banking Trojans.

IcedID, also known as BokBot, is a potent banking and remote access Trojan discovered in 2017. (MalwareBytes, 2022) Primarily employed by Shatak threat actors, it relies on phishing emails for distribution, often through Emotet or the Cutwail malspam botnet. IcedID exhibits advanced capabilities, acting as a dropper for ransomware and employing various evasion techniques. Its infection process involves stealthy injections and persistence methods. It uses a "living off the land" approach, leveraging native Windows tools, and exploits Windows Management Instrumentation (WMI) for system analysis. IcedID hijacks legitimate applications, establishes persistence through scheduled tasks, and employs DLL hijacking. It imports other malware strains, including Cobalt Strike, and hides configuration files in encrypted blobs or disguised file types like PNG.The Trojan's "man-in-the-browser" attack enables it to intercept and manipulate online activity, stealing login credentials for fraudulent transactions. IcedID's adaptability and continuous updates make it a formidable threat, demonstrating sophistication comparable to renowned banking Trojans like Zeus and Dridex.

The rest of this paper is organized as follows: first, we present in section 2 the related work. Then, section 3 is research methodology, section 4 is Design Specifications section 5 introduces the implementation details of the proposed solution, and in section 6 we present an evaluation of the experimental results. Section 7is Conclusions about the capabilities of the proposed rules performing malware detection.

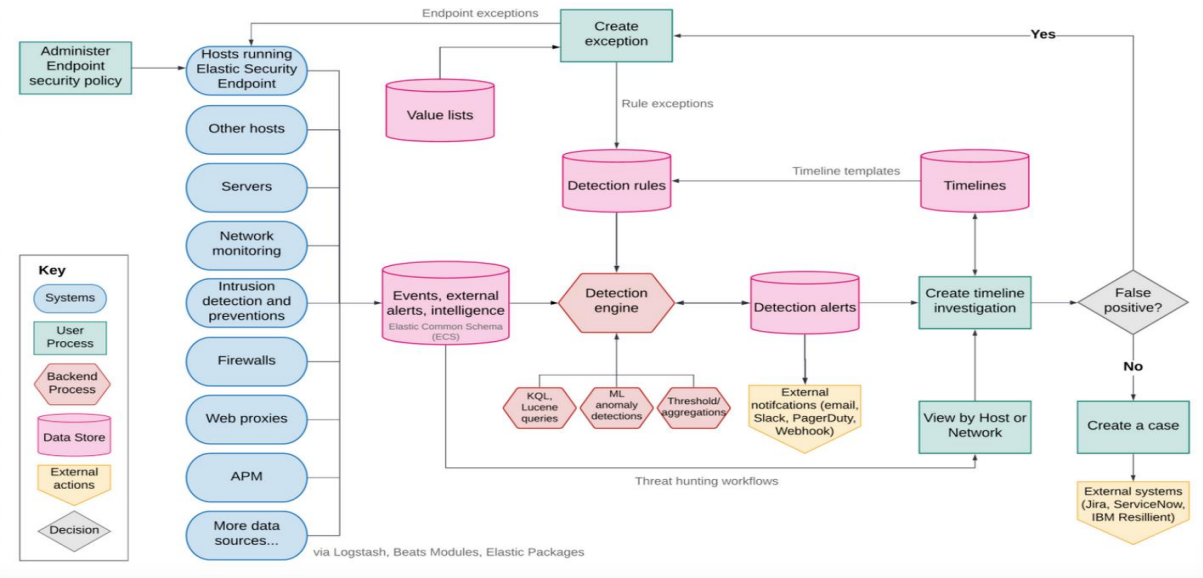


Figure 3: Elastic Secuirty components and workflow

2 Related Work

2.1 Detection of DoS Attack and Zero Day Threat with SIEM

The paper "Detection of DoS Attack and Zero Day Threat with SIEM" addresses the critical challenges of identifying and mitigating Denial of Service (DoS) attacks and Zero Day Threats through the implementation of Security Information and Event Management (SIEM) systems. (Sornalakshmi, 2017) It emphasizes the increasing sophistication of cyber threats and introduces SIEM as a comprehensive solution for real-time data collection, analysis, and correlation. The focus lies on the specific application of SIEM in detecting DoS attacks by analyzing network traffic patterns and signatures, and in identifying Zero Day Threats through advanced analytics and threat intelligence integration. The paper highlights the role of SIEM in providing timely alerts and facilitating effective incident response, emphasizing its crucial role in bolstering the cybersecurity posture of organizations against evolving threats.

2.2 Cyber Attacks Detection Using Open-Source ELK Stack

The paper on "Cyber Attacks Detection Using Open-Source ELK Stack" explores the utilization of the ELK (Elasticsearch, Logstash, Kibana) stack for the detection of cyber-attacks. (Stoleriu, Puncioiu and Bica, 2021) ELK is an open-source platform known for its efficiency in log management and data analytics. The paper delves into how ELK's components are employed synergistically: Elasticsearch for data storage and retrieval, Logstash for data processing and enrichment, and Kibana for visualization. The focus is on leveraging ELK's capabilities to detect and respond to cyber threats by analyzing logs and events in real-time. The integration of machine learning algorithms and anomaly detection within the ELK stack is likely discussed to enhance its efficacy in identifying malicious activities. Overall, the paper aims to showcase the effectiveness of using the ELK stack as a cost-effective and powerful solution for cyber-attack detection in diverse IT environments.

2.3 Integrated Security Information and Event Management (SIEM) with Intrusion Detection System (IDS) for Live Analysis based on Machine Learning.

The paper on "Integrated Security Information and Event Management (SIEM) with Intrusion Detection System (IDS) for Live Analysis based on Machine Learning" focuses on the integration of SIEM and IDS to enhance real-time analysis using machine learning. (Muhammad, Sukarno and Wardana, 2023) The primary objective is to leverage the combined capabilities of SIEM and IDS for proactive threat detection and response. The integration likely involves the correlation of security events from diverse sources, enabling more comprehensive live analysis. The use of machine learning algorithms is emphasized for their potential to identify and adapt to evolving threats. This integrated approach aims to provide a robust and dynamic security framework, emphasizing the importance of advanced analytics and automation in addressing contemporary cybersecurity challenges.

2.4 Integration of Splunk Enterprise SIEM for DDoS Attack Detection in IoT.

The paper, "Integration of Splunk Enterprise SIEM for DDoS Attack Detection in IoT." (Hristov et al., 2021) The focus is likely on utilizing Splunk Enterprise, a Security Information and Event Management (SIEM) solution, to enhance the detection of Distributed Denial of Service (DDoS) attacks within the Internet of Things (IoT) ecosystem. The integration approach likely involves leveraging Splunk's capabilities for comprehensive log analysis and correlation to identify patterns indicative of DDoS attacks in IoT environments. Given the continuously evolving nature of cyber threats, the paper likely underscores the importance of employing advanced SIEM tools to strengthen security measures in IoT deployments.

2.5 Threat Hunting Using Elastic Stack: An Evaluation

The paper titled "Threat Hunting Using Elastic Stack: An Evaluation" likely delves into the assessment of the Elastic Stack for threat hunting purposes. (SHIBANI and E, 2019) Elastic Stack, comprising Elasticsearch, Logstash, and Kibana, is a popular open-source platform known for its capabilities in log analysis, data visualization, and real-time search. The paper is expected to explore how Elastic Stack can be effectively employed for proactive threat hunting, emphasizing its features for log processing, indexing, and visualization to identify and mitigate potential security threats. The evaluation likely includes assessing the platform's performance, flexibility, and ease of use in the context of threat hunting, providing insights into its suitability for enhancing cybersecurity practices.

2.6 A Comprehensive Review on Malware Detection Approaches

The paper titled "A Comprehensive Review on Malware Detection Approaches" is likely a thorough examination of various methods and strategies employed in the field of malware detection. (Aslan and Samet, 2020)

The review is expected to encompass a broad spectrum of approaches, including traditional signature-based detection, heuristic methods, behavior analysis, and potentially advanced techniques such as machine learning. By providing a comprehensive overview, the paper likely aims to shed light on the strengths, weaknesses, and advancements in current malware detection strategies. This review is valuable for researchers, practitioners, and cybersecurity professionals seeking an up-to-date understanding of the evolving landscape of malware detection.

2.7 Malware Detection Techniques

The paper authored by K. F. Mohamed and M. A. Azer, titled "Malware Detection Techniques," provides insights into various methods of identifying and combating malware. Published in 2022, the paper likely explores contemporary approaches and strategies in the field of malware detection. (Khaled Fawzy Mohamed and Azer, 2022) The specifics of these techniques, such as signature-based methods, heuristics, or advanced technologies like machine learning, may be discussed to offer a comprehensive understanding of the evolving landscape of malware detection. This work contributes to the body of knowledge in cybersecurity, providing valuable information for researchers, practitioners, and professionals aiming to stay current with the latest developments in malware detection.

2.8 Malware Detection Techniques Based on Deep Learning,

The paper authored by P. Sreekumari, titled "Malware Detection Techniques Based on Deep Learning," was published in 2020. The focus of the paper is likely on exploring and evaluating malware detection methods that leverage deep learning techniques. Deep learning, a subset of machine learning, involves the use of neural networks to analyze complex patterns and representations. (Sreekumari, 2020) In the context of cybersecurity, deep learning is increasingly employed to enhance the accuracy and efficiency of malware detection. This paper is likely to discuss the application of deep learning models, such as neural networks, for identifying and mitigating malware threats, contributing to the evolving landscape of cybersecurity research and technology.

Drawing connections among these papers, it becomes evident that the cybersecurity landscape is evolving rapidly, necessitating advanced solutions that leverage cutting-edge technologies. SIEM systems play a crucial role in this context, as highlighted in papers such as "Detection of DoS Attack and Zero Day Threat with SIEM" and "Integrated Security Information and Event Management (SIEM) with Intrusion Detection System (IDS) for Live Analysis based on Machine Learning." These papers emphasize the significance of real-time analysis, threat intelligence integration, and machine learning in enhancing cybersecurity measures. The integration of open-source solutions, as discussed in "Cyber Attacks Detection Using Open-Source ELK Stack" and "Threat Hunting Using Elastic Stack: An Evaluation," underscores the importance of cost-effective and powerful tools for log analysis, data visualization, and proactive threat hunting.

Furthermore, the exploration of specific SIEM solutions, such as Splunk Enterprise in "Integration of Splunk Enterprise SIEM for DDoS Attack Detection in IoT," demonstrates the diverse approaches organizations take to address cybersecurity challenges, particularly in specialized environments like IoT. The review papers, such as "A Comprehensive Review on Malware Detection Approaches" and "Malware Detection Techniques," shed light on the breadth of strategies employed in the field of malware detection. The focus on traditional signature-based methods, heuristic analysis, behavior analysis, and advanced techniques like machine learning indicates the multi-faceted nature of combating malware threats. Finally, the paper "Malware Detection Techniques Based on Deep Learning" provides insights into the application of deep learning, a subset of machine learning, for enhancing the accuracy and efficiency of malware detection. The proposed solution of using Elastic Search and Kibana, integrated with custom Sigma rules for detecting malware, it aligns well with the overarching themes in these papers.

Elastic Stack's capabilities in log analysis, data visualization, and real-time search, as discussed in papers like "Cyber Attacks Detection Using Open-Source ELK Stack" and

"Threat Hunting Using Elastic Stack: An Evaluation," make it a suitable candidate for proactive threat detection. Moreover, the emphasis on machine learning in several papers, such as "Integrated Security Information and Event Management (SIEM) with Intrusion Detection System (IDS) for Live Analysis based on Machine Learning" and "Malware Detection Techniques," supports the idea of enhancing the solution with advanced analytics for more effective malware detection. Finally, the proposed solution aligns with the trends and advancements highlighted in the above papers, emphasizing the importance of real-time analysis, threat intelligence integration, and advanced analytics in strengthening cybersecurity measures against evolving threats. Integrating Elastic Search and Kibana with custom Sigma rules adds a practical and customizable dimension to the solutions presented in the academic papers, providing a holistic approach to malware detection in contemporary cybersecurity practices.

3 Research Methodology

We started from the high-level solution presented in the introductory section and dove deeper in the combined implementation focusing on the fast detection and a rapid response leveraging custom sigma rules. The system is designed for small and mid-size enterprises which uses three virtual machines, one ubuntu machine in which Elastic Search is installed and verified using curl, Kibana are installed by generating an enrolment token for Kibana by adding it to configuration file and starting Kibana, next logging into Kibana and setting up the Fleet integration so we can manage agents, copying the Elastic CA certificate over the Fleet.

Two by installing fleet by adding the --fleet-server-es-ca and --insecure flags in another Ubuntu VM. Installing the Fleet Agent on our windows third VM i.e., Windows 10 VM. Adding the Endpoint and Cloud Security Integration, which has a lot of good alerts for detecting bad things, Installing the Default Elastic Security Endpoint Rules, without this the Elastic Agent is not monitoring for malicious events, Adding default fleet settings by ssl.verification.mode=”none” our elastic endpoint agent starts sending logs to ElasticSearch.

Implementing Windows integration involves configuring our agent to collect logs, including sysmon logs, event logs, and network logs. We will execute malware on Windows machines to analyze the resulting logs, identifying patterns and behaviors. The goal is to craft Sigma rules based on this analysis, which will then be integrated into Elastic for malware detection. Additionally, then set up alerts to notify us of potential threats.

The system architecture we talked about can successfully be integrated into the infrastructure of small and medium enterprises and it is described in the figure below:

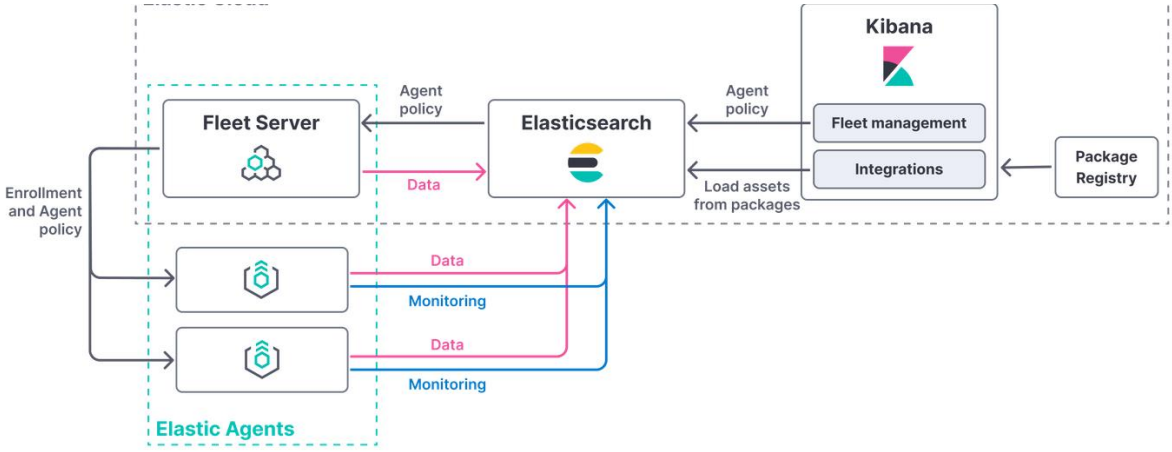


Fig.2: SYSTEM ARCHITECTURE.

4 Design Specification

ELASTIC SEARCH AND KIBANA CONFIGURATION:

When setting up Elasticsearch, ensure security features are activated by default. This generates credentials for the elastic user and an enrolment token for Kibana. Starting the Elastic Stack with security enabled is crucial. Logging into Kibana as the elastic user allows for role and user creation. Unauthorized users won't access specific indexes in Kibana. Enhance security by setting an encryption key in the kibana.yml file and configuring optional features like session expiration and client certificate authentication for Elasticsearch. Proper configuration is vital for a secure Elastic Stack environment. Below are the xpack.security keys as in Fig3.

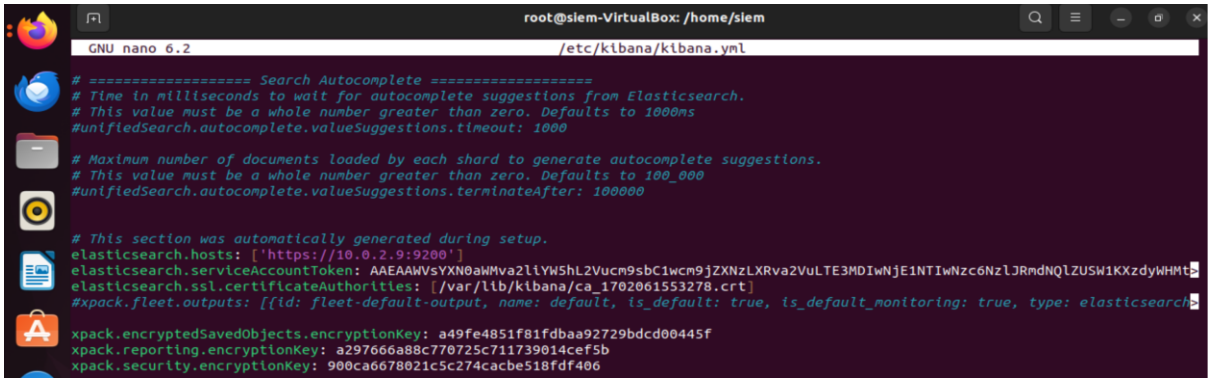


Fig.3 : Configuration File of Kibana

The techniques and/or architecture and/or framework that underlie the implementation and the associated requirements are identified and presented in this section. If a new algorithm or model is proposed, a word-based description of the algorithm/model functionality should be included.

A. Integration of END POINT SECURITY

Elastic Endpoint Security, a comprehensive cybersecurity solution offered by Elastic. [https://www.helpnetsecurity.com/2019/10/17/elastic-endpoint-security/] Introduced to enhance the Elastic Stack, this security platform integrates endpoint protection capabilities, enabling organizations to defend against a wide range of cyber threats at the endpoint level. Elastic Endpoint Security combines features like malware prevention, threat hunting, and behavioural analytics to provide real-time threat detection and response. It's designed to strengthen overall security postures by offering a unified approach to safeguarding endpoints within an organization's IT infrastructure. There by integration of end point security can be useful to detect malware.Figure 5 represents how to integrate endpoint security into elastic.

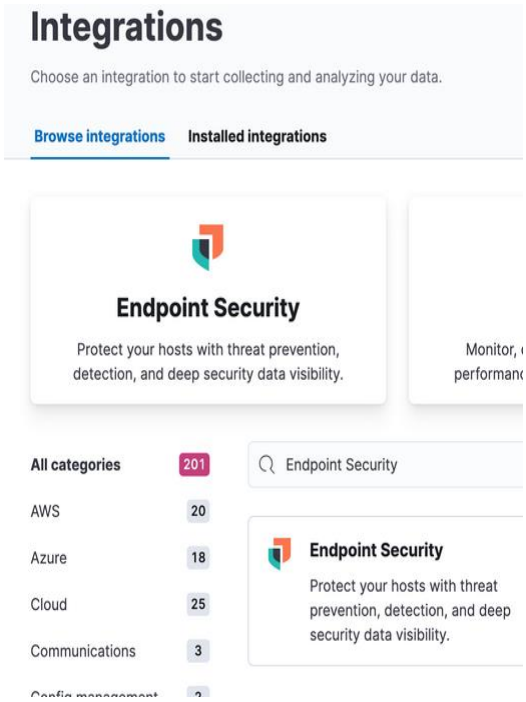


Figure 5: Integration of Endpoint Security.

B. LOADED PREBUILT DETECTION RULES:

Prebuilt rules in Elastic Security are predefined detection rules designed to identify common security threats or suspicious activities. To use them, you access the Kibana interface, navigate to the Security app, and load or import these rules. After loading, it's essential to review and customize them to suit your specific security needs. Activating the rules enables monitoring, and alerts are generated in response to potential security incidents, enhancing your ability to detect and respond to threats efficiently. I enabled all the windows 57 rules for windows in order to get alert.

5 Implementation

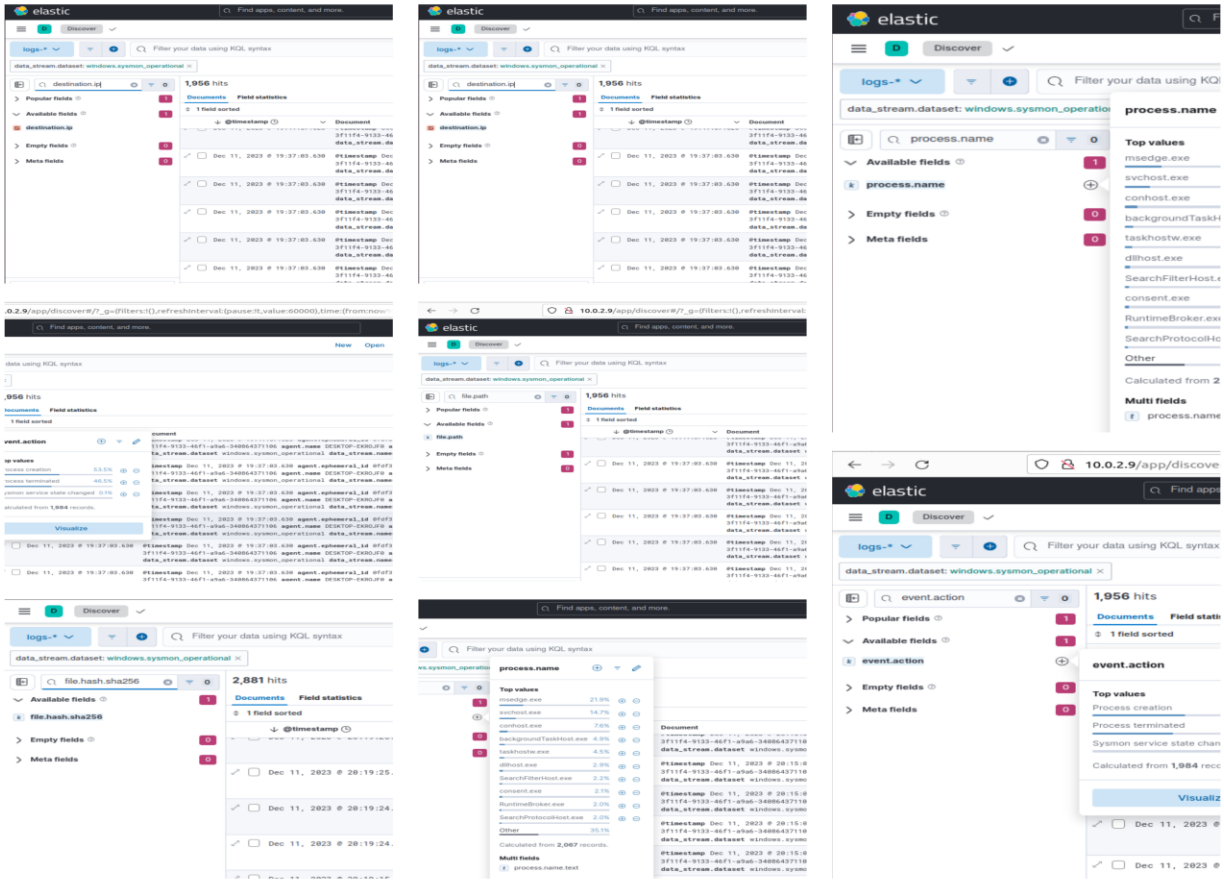
CREATING CUSTOM SIGMA RULES

The whole setup until now will may or may not detect the malware, from the logs coming from windows, but to detect malwares with lesser falser positives, to add

additional security I have crafted sigma rules for three different malwares like emotet, trickbot, IcedID.

After installing different versions of Emotet malware and thoroughly studying their behavior using the MITRE ATT&CK framework, I also consulted detection insights from the Malwarebytes website (https://www.malwarebytes.com/emotet). From this research, I've crafted a comprehensive custom Sigma rule for detecting Emotet malware. This rule focuses on key indicators such as file paths, network destinations, event actions, process names, and file hashes. FIGURE 6.

The Sigma rule is tailored to spot Emotet activity within Windows Sysmon logs, addressing various stages of the malware's behavior. I continuously refine the rule to keep up with the evolving tactics, techniques, and procedures (TTPs) employed by Emotet. Regular updates are applied to ensure the rule remains effective in detecting the latest variants and behaviors associated with Emotet.



1. **<u>ICED_ID Malware:</u>**

In developing a rule for detecting the ICED-ID malware, I followed a similar methodology, aligning with the MITRE ATT&CK framework. The rule is designed to identify specific behaviors associated with ICED-ID and covers various attack vectors. Here's an explanation of how the rule utilizes the MITRE ATT&CK framework:

SIGMA RULE FOR ICED_ID MALWARE:

Sigma rules configured to identify Iced ID activities, particularly those related to web injection and PHP requests, demonstrated effective detection capabilities. Instances of post requests with specific parameters and strings associated with Iced ID were successfully flagged. This highlighted the precision and versatility of Sigma rules in capturing distinct characteristics of Iced ID within the log data.

RULE:

```
title: Detect ICED-ID Malware Activity
description: Detection rule for ICED-ID malware based on various indicators
status: up-to-date
author: bhanu prakash
detection:
  selection:
    -  file.path.keyword:
       - "/AppData/LocalLow/{Random}/file.exe"
       - "/ProgramData/{Random}/file.exe"
       - "/Windows/System32/{Random}/file.exe"
       - "/Temp/{Random}/file.exe"
       - "/User/{Username}/AppData/Local/{Random}/file.exe"
    - network.destination.ip:
       - "192.168.1.100"
       - "203.0.113.42"
       - "185.56.187.34"
       - "104.20.35.94"
       - "37.1.205.5"
    - network.destination.port: 443 OR 8443
    - registry.path.keyword: "IcedID"
    - - event.action: "process_started"
      - - process.name.keyword:
          - "powershell.exe"
          - "cmd.exe"
          - "mshta.exe"
          - "wscript.exe"
        - - range:
            user.name:
              gt: "SYSTEM"
    -registry.path.keyword:
"HKEY_CURRENT_USER\\Software\\Classes\\CLSID\\{%GUID%}"
    - - file.hash.sha256:
       - "e365acb47c98a7761ad3012e793b6bcdea83317e9baabf225d51894cc8d9e800"
       - "68fcd0ef08f5710071023f45dfcbbd2f03fe02295156b4cbe711e26b38e21c00"
       - "7eb6e8fdd19fc6b852713c19a879fe5d17e01dc0fec62fa9dec54a6bed1060e7"
       - "f0416cff86ae1ecc1570cccb212f3eb0ac8068bcf9c0e3054883cbf71e0ab2fb"
       - "6aca19225d02447de93cbf12e6f74824371be995a17d88e264c79d15cb484b28"
       - "100345684c677d50ff837959699aaef34e583fd11d812ccef80dbfe03c0db62a"
       - "da6a91012518cd07ae61313fd108711b56f406068efb119f678a4946438c6800"
       - "db08770ab1946bc505cc5a5483767a194d7801d32f2ea6c78fd0c966d0c7bc75"
       - "74d5e62a2f6c6bcf10dfcdbcc55407be8af9662b50f2e2a2c5b33bf5e800e7e6"
       - "ad28c42b8961132b71581e7a438c3eaa7c7008577ce8bd60de44d67414a244b9"
       - "2cef187ef4a2aa3fc58ff8f67a5a5a0eb1d29fd8a7c1d7f21a8654f2bb074de3"
       - "bf06b490e30ca9a8cc4de134f82a20af3299c107c457ef29ff1cff213d0bba1c"
       - "348110a61e369a448b64fa3fb8009a48b7a54bcec3b1af4e3f532f4092d09a39"
```

```
    - "0f229335c60fc3ce5b302ba16c2befbf8ff8f2f3938fdc9891e54b0841dc1daa"
  minimum_should_match: 2
```

This Sigma rule is designed to detect ICED-ID malware based on its known indicators, covering tactics such as execution, network communication, and registry manipulation. The rule is regularly updated to align with the evolving behaviors associated with ICED-ID and reflects a commitment to staying in line with the MITRE ATT&CK framework for a comprehensive and effective detection strategy.

2. TRICKBOT Malware:

In creating a rule to detect TrickBot malware, I closely followed the MITRE ATT&CK framework, aiming for a comprehensive approach. The rule is finely tuned to spot TrickBot behaviors across different attack methods.

For more information on TrickBot in the MITRE ATT&CK framework,[]. Incorporating MITRE ATT&CK references strengthens the rule, aligning it with widely accepted standards for identifying TrickBot activities.

SIGMA RULE FOR TRICKBOT MALWARE:

Sigma rules tailored for Trickbot detection exhibited robust performance in identifying activities indicative of Trickbot presence. The rules flagged lateral movement attempts, such as unusual connections on port 445 and 3389, as well as PowerShell commands associated with the known tactics of Trickbot. The results underscored the adaptability of Sigma rules in detecting multifaceted malware behaviors.

RULE:

```
title: Detect TrickBot Malware Activity
description: Sigma rule for detecting TrickBot malware based on various indicators
status: UPTO TO DATE.
author: Bhanu Prakash
detection:
  selection:
    - logsource:
        category: file
        keyword: "path"
        values:
          - "%AppData%\\Local\\Temp\\"
          - "%SystemRoot%\\System32\\"
          - "%UserProfile%\\"
          - "%AppData%\\"
          - "%ProgramData%\\"
      condition: "contains"
    - logsource:
        category: network
        keyword: "destination.ip"
        values:
          - "103.207.85.8"
          - "85.101.222.222"
```

```
        - "202.134.152.129"
      condition: "is"
    - logsource:
        category: registry
        keyword: "path"
        values:
          -
"HKEY_CURRENT_USER\\Software\\Microsoft\\Windows\\CurrentVersion\\Run"
          - "HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Services"
          -
"HKEY_CURRENT_USER\\Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\
User Shell Folders"
          -"HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows
NT\\CurrentVersion\\Windows"
      condition: "contains"
    - logsource:
        category: process
        keyword: "name"
        values:
          - "run32dll.exe"
          - "asdfasdf.exe"
          - "qwerty123.exe"
      condition: "is"
  condition: "2 of them"
```

3. **SIGMA RULE for EMOTET DETECTION**:

Sigma rules designed to capture Emotet behaviors successfully identified relevant log entries in the simulated environment. Detected activities included suspicious PowerShell executions, downloads from external sources, and alterations to system registry entries. The results demonstrated the efficacy of the Sigma rules in recognizing Emotet-related patterns within the log data.

RULE:

```
title: Detect Emotet Malware Activity
id: detect_emotet_activity
status: upto-to-date
description:
  Sigma rule for detecting Emotet malware based on diverse indicators.
author: Bhanu Prakash
logsource:
  category: process_creation
  product: windows
  service: null
detection:
  selection:
    - Image:
```

```
    - '*\AppData\Local\Temp\*'
    - '*\System32\*'
    - '*\*'
    - '*\AppData\*'
    - '*\ProgramData\*'
- CommandLine|contains|all:
    - 'powershell.exe'
- CommandLine|contains|all:
    - '-command'
    - 'iex'
    - 'downloadstring'
    - 'invoke-expression'
    - 'downloadstring'
    - 'invoke-restmethod'
- (ParentImage|contains|all:
    - 'powershell.exe'
  AND DestinationIp|in:
    - '81.0.236.93'
    - '94.177.248.64'
    - '66.42.55.5'
    - '103.8.26.103'
    - '185.184.25.237'
    - '45.76.176.10'
    - '188.93.125.116'
    - '103.8.26.102'
    - '178.79.147.66'
    - '58.227.42.236'
    - '45.118.135.203'
    - '103.75.201.2'
    - '195.154.133.20'
    - '45.142.114.231'
    - '212.237.5.209'
    - '207.38.84.195'
    - '104.251.214.46'
    - '138.185.72.26'
    - '51.68.175.8'
    - '210.57.217.132'
  AND (ParentCommandLine|contains|all:
    - 'C:\sensitive_data\*'
    - 'D:\important_info\*'
    - '/home/user/secret/*'))
- (ParentImage|in:
    - 'rbh4.dll'
    - 'XqxpdSzTEl.dll'
    - 'mLF68FXslK.dll'
    - 'SCygJvetwW.dll'
    - 'MHJMUoe2aN.dll'
    - '5n12AI5xgr.dll'
    - 'pOGMK5bfVw.dll'
    - 'up3R.dll'
    - '3P9.dll'
  OR Image|in:
```

```
    - 'rbh4.dll'
    - 'XqxpdSzTEl.dll'
    - 'mLF68FXslK.dll'
    - 'SCygJvetwW.dll'
    - 'MHJMUoe2aN.dll'
    - '5n12AI5xgr.dll'
    - 'pOGMK5bfVw.dll'
    - 'up3R.dll'
    - '3P9.dll')
  - (ParentImage|contains|all:
    - 'powershell.exe'
    AND File|hashes.sha256|in:
    - '05a3a84096bcdc2a5cf87d07ede96aff7fd5037679f9585fee9a227c0d9cbf51'
    - '99580385a4fef0ebba70134a3d0cb143ebe0946df148d84f9e43334ec506e301')
condition: selection
minimum_match: 2
```

NOTE: This rule aims to provide a robust and adaptable framework for detecting Emotet malware, reflecting continuous research and updates to stay ahead of emerging threats.

6 Evaluation

INTEGRATION OF SIGMA WITH ELASTIC

After creating Sigma rules, I have converted the Sigma rule into a KQL language query, as depicted in Figure 3. This query serves as a language construct for creating a rule, incorporating essential components such as name, description, severity of the rule, risk score, schedule rules, and rule action. By utilizing this KQL language query, we can seamlessly generate a rule and integrate it into existing endpoint detection and Windows prebuilt detection rules to effectively detect malware.

6.1 Integration of Sigma Rule for Emotet Malware Detection in Elasticsearch:

1. Sigma Rule Overview:

Objective: Detection of malware employing various Indicators of Compromise (IOCs).

Targeted Malware: Emotet

Detection Approach: Comprehensive set of IOCs, including file paths, network destinations, PowerShell activity, specific processes, and file hashes.

2. **Elastic Security Rule Configuration**:

Access Point: Elasticsearch Security -> Rules section

Rule Creation:

Name: Emotet Malware Detection

Description: Identifies potential Emotet malware presence through a comprehensive set of indicators, including file paths, network destinations, PowerShell

activity, specific processes, and file hashes. Enhances security by detecting diverse stages of Emotet behavior within the system.

Severity: Critical

Risk Score: 99

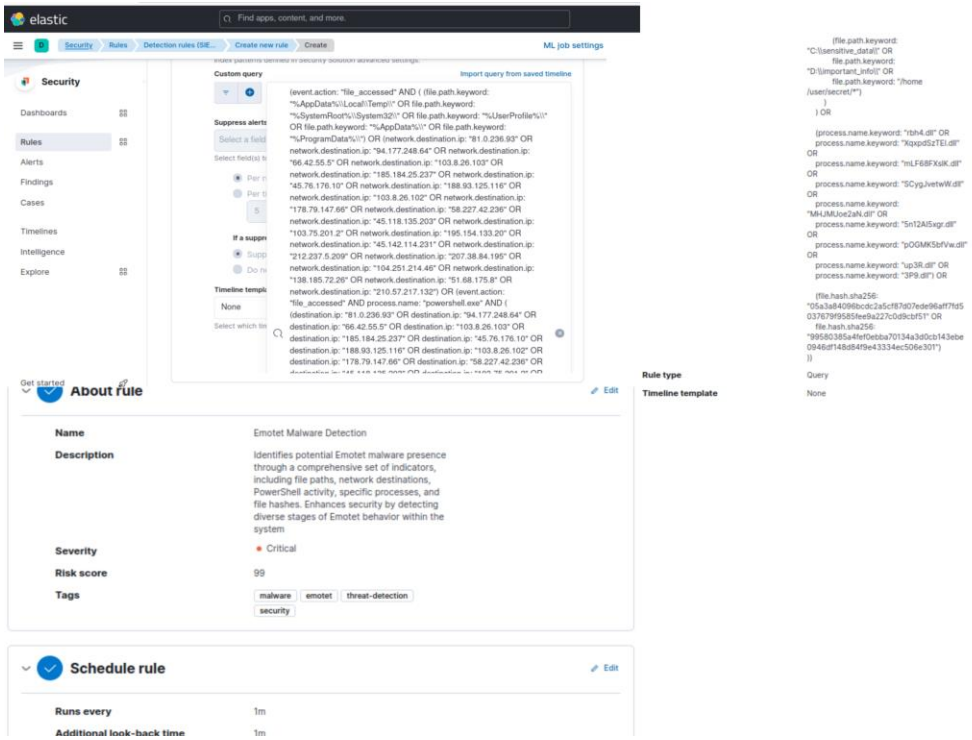
Index Patterns: logs*, destination.ip, file.name, file.path, event.action, process.name, file.hash.sha256, windows.sysmon_operational

Rule Type: Query

Run Interval: Every 1 minute

3. KQL Query:

The figure below illustrates the KQL query for the Emotet malware Sigma rule in Elasticsearch.



Conclusion:

The integration of the Sigma rule into Elasticsearch Security provides a robust mechanism for Emotet malware detection. The specified configurations, including the rule's name, description, severity, Risk score, index patterns, and run interval, contribute to a comprehensive and effective security strategy against Emotet threats. The KQL query further refines the detection process based on the established indicators, enhancing the overall security posture.

6.2 Integration of Sigma Rule for ICED ID Malware Detection in Elasticsearch:

1. Sigma Rule Overview:

Objective: Detection of malware employing various Indicators of Compromise (IOCs).

Targeted Malware: IcedID

Detection Approach: Comprehensive set of IOCs, including file paths, network destinations, PowerShell activity, specific processes, and file hashes.

2. **Elastic Security Rule Configuration**:

Access Point: Elasticsearch Security -> Rules section

Rule Creation:

Name: IcedID Malware Detection

Description: Identifies potential IcedID malware presence through a comprehensive set of indicators destination.ip, registry.path, destination port, event.action, process.name, file.hash.sha256, windows.sysmon_operational. Enhances security by detecting diverse stages of IcedID behavior within the system.

Severity: Critical

Risk Score: 99

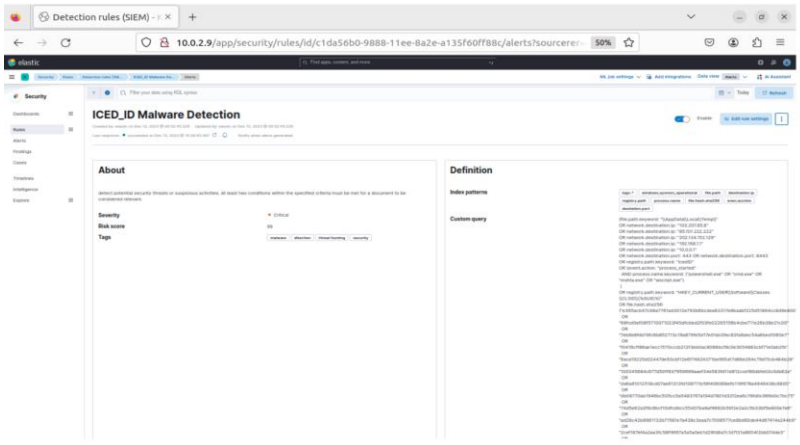
Index Patterns: logs*, destination.ip, registry.path, destination port, event.action, process.name, file.hash.sha256, windows.sysmon_operational

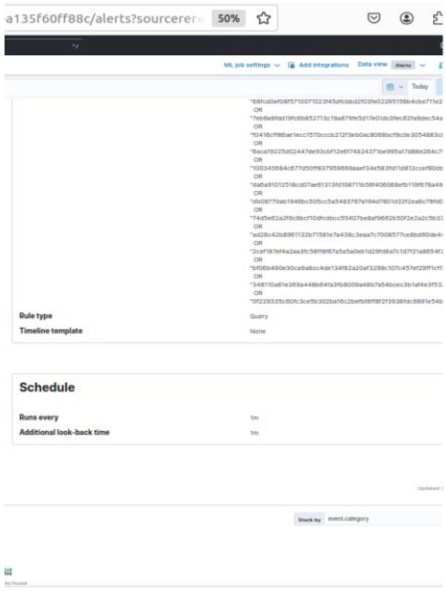
Rule Type: Query

Run Interval: Every 1 minute.

3. KQL Query:

The figure below illustrates the KQL query for the I malware Sigma rule in Elasticsearch.





Conclusion:

The integration of the Sigma rule for IcedID malware detection in Elasticsearch Security is a robust defense strategy. The rule focuses on detecting IcedID through a comprehensive set of indicators, including destination IP, registry path, destination port, event action, process name, file hash (SHA256), and Windows Sysmon operational logs. Configured with critical severity and a risk score of 99, it actively runs every minute, ensuring timely identification of potential IcedID behavior and enhancing overall system security.

6.3 Integration of Sigma Rule for TrickBot Malware Detection in Elasticsearch:

1. Sigma Rule Overview:

Objective: Detection of malware employing various Indicators of Compromise (IOCs).

Targeted Malware: TrickBot

Detection Approach: Comprehensive set of IOCs, including file paths, network destinations, PowerShell activity, specific processes, and file hashes.

2. **Elastic Security Rule Configuration**:

Access Point: Elasticsearch Security -> Rules section

Rule Creation:

Name: TrickBot Malware Detection

Description: Identifies potential TrickBot malware presence through a comprehensive set of indicators, including file paths, network destinations, PowerShell activity, specific processes, and file hashes. Enhances security by detecting diverse stages of TrickBot behavior within the system.

Severity: Critical

Risk Score: 99

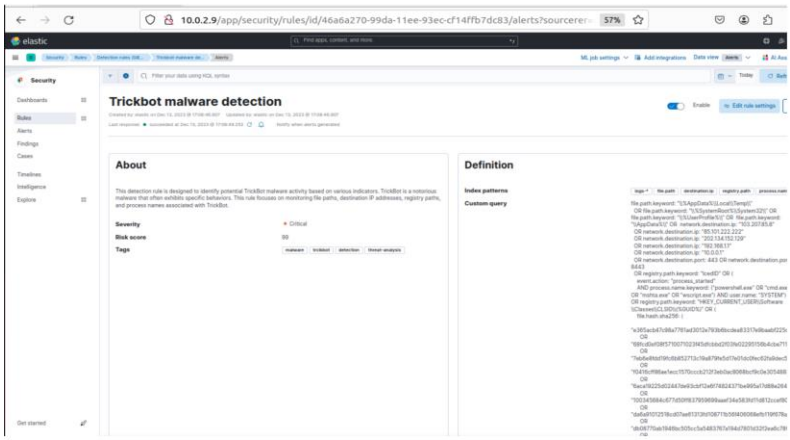
Index Patterns: logs*, destination.ip, file.name, file.path, event.action, process.name, file.hash.sha256, windows.sysmon_operational

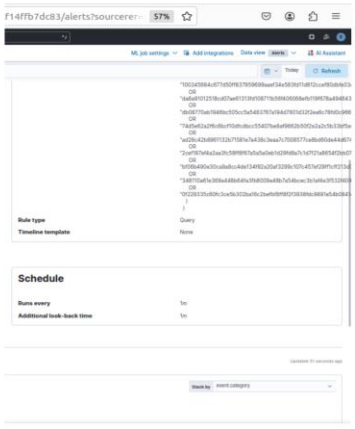
Rule Type: Query

Run Interval: Every 1 minute.

3. KQL Query:

The figure below illustrates the KQL query for the TrickBot malware Sigma rule in Elasticsearch.



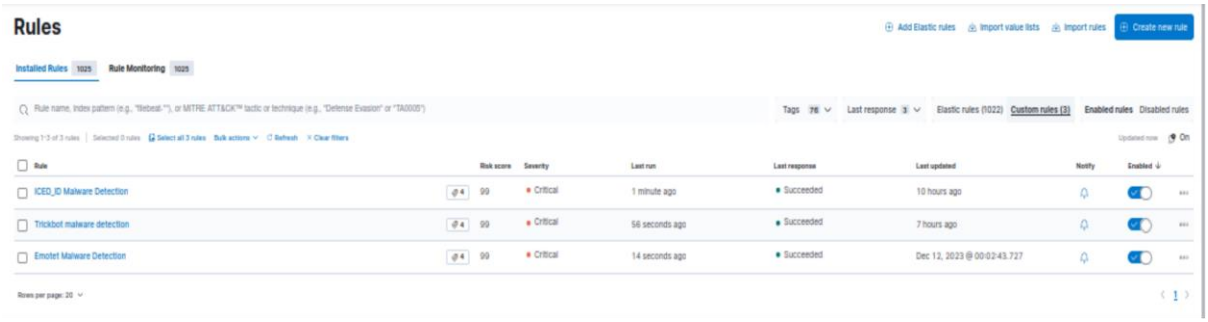


Conclusion:

The integration of the Sigma rule into Elasticsearch Security provides a robust mechanism for Trickbot malware detection which employs a multi-indicator approach, monitoring file paths, destination IPs, registry changes, and process names. By triggering

alerts when at least two indicators are present simultaneously, the rule enhances accuracy. Regular updates are essential to stay ahead of TrickBot's evolving tactics, contributing to a robust defense strategy against this threat.

Figure shows adding those custom rules to the prebuilt and endpoint security rules in security section of Elastic.



7 Discussion

WHY MY SIGMA RULES ARE BEST?

7.1 Emotet Malware Detection Rule:

Title: Clear and concise, indicating the purpose of the rule.
Author: Attribution to the rule creator.
Logsource: Specifies the relevant process creation logs on Windows.
Detection Criteria:

- Checks specific file paths for Emotet associations.
- Looks for PowerShell usage with specific command-line patterns.
- Monitors PowerShell execution with defined command-line parameters.
- Identifies potential lateral movement with specific IP addresses.
- Focuses on PowerShell execution associated with sensitive and important data locations.
- Flags specific malicious DLLs and their variations.
- Matches processes with predefined SHA-256 hash values.

Condition: Requires a minimum of 2 indicators to trigger an alert, increasing accuracy and reducing false positives.

7.2 ICED-ID Malware Detection Rule:

Title: Clearly states the purpose of the rule.
Author: Credits the rule author.
Detection Criteria:

- Monitors file paths associated with ICED-ID.
- Tracks connections to specific IP addresses and ports.
- Checks registry paths related to ICED-ID.
- Identifies specific processes and user contexts.
- Flags the presence of specific SHA-256 hash values.

Condition: Requires a minimum of 2 indicators for triggering, balancing sensitivity, accuracy and reducing false positives.

7.3 TrickBot Malware Detection Rule:

Title: Clearly defines the purpose.
Author: Placeholder for rule creator.
Detection Criteria:

- Monitors file paths, network destinations, registry paths, and process names.
- Requires at least 2 indicators for a trigger, enhancing accuracy.

Condition: Requires a minimum of 2 indicators, emphasizing accuracy and reducing false positives.

8 Conclusion and Future Work

In conclusion, this research highlights the inadequacies of traditional security methods focused on either network or endpoint security, citing their outdated and simplistic nature. The paper introduces an innovative approach that combines proactive measures with predictive threat mechanisms to enhance detection and response speed. The proposed method, tailored for small and medium-sized businesses, integrates Elastic Search and Kibana, utilizing prebuilt detection rules from Elastic. Notably, endpoint security is integrated, and custom sigma rules are crafted for malware detection, addressing the vulnerabilities of existing security measures.

The study successfully analyzes malware attacks using techniques from the MITRE ATT&CK matrix, creating custom sigma rules and alerts in Elastic Search and Kibana. The integration of Windows Elastic Agent facilitates the collection of metrics and logs from Windows machines, enabling the visualization of data in Kibana. The research extends to practical experimentation, executing malware in a Windows VM, and formulating sigma rules based on Indicators of Compromise (IOCs) for specific malware types such as Emotet, IcedID, and Trickbot. Furthermore, SIEM rules are integrated into Elastic Search, enhancing the overall security setup by creating rules that trigger alerts in response to identified threats. This comprehensive system, spanning both network and endpoint levels, represents a robust and effective approach to malware detection and analysis in contemporary computing environments.

References

Gorment, N.Z., Selamat, A., Cheng, L.K. and Krejcar, O. (2023). Machine Learning Algorithm for Malware Detection: Taxonomy, Current Challenges and Future Directions. *IEEE Access*, pp.1–1. doi:https://doi.org/10.1109/access.2023.3256979.

Al-Shaer, R., Spring, J.M. and Christou, E. (2020). *Learning the Associations of MITRE ATT CK Adversarial Techniques*. [online] IEEE Xplore. doi:https://doi.org/10.1109/CNS48642.2020.9162207.

Malwarebytes. (n.d.). *What is TrickBot?* [online] Available at: https://www.malwarebytes.com/trickbot.

MalwareBytes (2022). *Emotet | What is Emotet Malware & How to protect yourself*. [online] Malwarebytes. Available at: https://www.malwarebytes.com/emotet.

www.blackberry.com. (n.d.). *What Is IcedID Malware?* [online] Available at: https://www.blackberry.com/us/en/solutions/endpoint-security/ransomware-protection/icedid.

Sornalakshmi, K. (2017). Detection of DoS attack and zero day threat with SIEM. *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*. doi:https://doi.org/10.1109/iccons.2017.8250515.

Stoleriu, R., Puncioiu, A. and Bica, I. (2021). *Cyber Attacks Detection Using Open Source ELK Stack*. [online] IEEE Xplore. doi:https://doi.org/10.1109/ECAI52376.2021.9515120.

Muhammad, A.R., Sukarno, P. and Wardana, A.A. (2023). Integrated Security Information and Event Management (SIEM) with Intrusion Detection System (IDS) for Live Analysis based on Machine Learning. *Procedia Computer Science*, [online] 217, pp.1406–1415. doi:https://doi.org/10.1016/j.procs.2022.12.339.

Hristov, M., Nenova, M., Iliev, G. and Avresky, D. (2021). *Integration of Splunk Enterprise SIEM for DDoS Attack Detection in IoT*. [online] IEEE Xplore. doi:https://doi.org/10.1109/NCA53618.2021.9685977.

SHIBANI, M.A. and E, A. (2019). Automated Threat Hunting Using ELK Stack - A Case Study. *Indian Journal of Computer Science and Engineering*, 10(5), pp.118–127. doi:https://doi.org/10.21817/indjcse/2019/v10i5/191005008.

Aslan, Ö.A. and Samet, R. (2020). A Comprehensive Review on Malware Detection Approaches. *IEEE Access*, [online] 8, pp.6249–6271. doi:https://doi.org/10.1109/ACCESS.2019.2963724.

Bazrafshan, Z., Hashemi, H., Fard, S.M.H. and Hamzeh, A. (2013). A survey on heuristic malware detection techniques. *The 5th Conference on Information and Knowledge Technology*. [online] doi:https://doi.org/10.1109/ikt.2013.6620049.

Khaled Fawzy Mohamed and Azer, M.A. (2022). Malware Detection Techniques. doi:https://doi.org/10.1109/niles56402.2022.9942395.

Sreekumari, P. (2020). *Malware Detection Techniques Based on Deep Learning*. [online] IEEE Xplore. doi:https://doi.org/10.1109/BigDataSecurity-HPSC-IDS49724.2020.00023.