Synopsis

For

"TrustCheck Reputation Assurance Platform"

Prepared by

Specialization	**SAP ID**	**Name**
Cyber Security and Forensics	500084998	Vivek Singh Chandel
Cyber Security and Forensics	500086232	Arpita Kumari
Cyber Security and Forensics	500086019	Arihant Vardhan
Cyber Security and Forensics	500087856	Vikash Kumar

Department of Systemics
School Of Computer Science
UNIVERSITY OF PETROLEUM & ENERGY
STUDIES, DEHRADUN- 248007. Uttarakhand

Akashdeep Bhardwaj

Dr. Akashdeep Bhardwaj	Dr. Neelu J. Ahuja
Project Guide	**Cluster Head**

Abstract

As cyber-attacks and crimes have grown exponentially in recent years, Threat Intelligence has gained momentum as a response. By collecting intelligence on how attackers operate, Threat Intelligence increases defenders' understanding of the threat landscape. In a nutshell, the purpose of TI is to help defenders identify their adversaries and understand their methods of attack and techniques of attack. In order to be a step ahead of attackers, defenders need to have a detailed understanding of their moves and reinforce their infrastructure in order to anticipate their moves.

There has been substantial research on Threat Intelligence (TI) and its benefits, but there is still a lack of literature on how to establish a Threat Intelligence Program (TIP). Consequently, organizations wishing to develop TIPs are on their own in this complex process.

Keywords: Threat Intelligence, Threat Intelligence Program, Information Security.

INTRODUCTION

1.1. Threat Intelligence

The traditional approach of Information Security (IS) against cyber-attacks has been generally reactive: attackers strike first, and defenders react. Defenders have the drawback of being constantly responding to the attackers' actions [1]. A common example is the URL filtering done by proxies. A proxy stops users from accessing sites only when these are known to be harmful. If a malicious actor uses a site that is unknown to the proxy, then the site will not be banned. Thus, attackers have the first-mover advantage.

As defenders increased their efforts to catch up with the attackers, so did the attackers. The attackers relentlessly keep developing their capabilities to be one step ahead. It soon become apparent that a new approach was needed to address this cat and mouse situation. To overcome this, Threat Intelligence (TI) emerged. Threat Intelligence is not a novel concept as it has been used in military strategy for a long time.

The objective of Threat Intelligence is to allow organisations to increase their visibility into the ever-changing threat landscape with the aim of detecting and preventing threats before they hit them. To achieve this, Threat Intelligence gathers vital intelligence about the adversaries, their methods and techniques. The information that Threat Intelligence provides can aid reinforcing the defences in the IT infrastructure as well as to provide essential intelligence to the decision-making process Threat Intelligence. This proactiveness is one of the main reasons that explain the increasing reliance of organisations on Threat Intelligence.

Fig[1.0] depicts this evolution as an iceberg metaphor. In the early stages of Threat Intelligence, analysts had access to internal logs and events; thus they could see only the tip of the iceberg, which resulted in a limited view of the threat landscape. With the introduction of new and external sources like the dark web, forums and feeds, analyst could see the underneath, which increased their overall visibility:

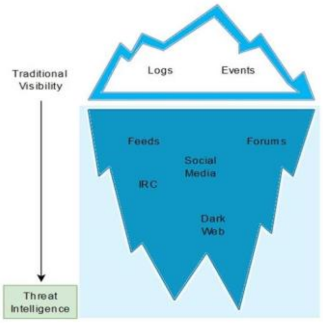


Fig [1.0] Evolution of Threat Intelligence

Approach:

- A comprehensive Python-based Threat Intelligence Platform designed to scrutinize URLs, IP addresses, and uploaded files for malicious attributes.
- Leveraging the VirusTotal API and a MongoDB database populated with open-source malicious IP and file hashes.
- **Integration of Various APIs:**
  - URL Scanning: Utilizing the VirusTotal API and Kaspersky API to assess the reputation of entered URLs.
  - IP Address Analysis: Querying multiple databases to examine IP addresses for malicious behavior.
- **File Analysis:**
  - Uploaded Files Inspection: Utilizing the VirusTotal API and Kaspersky API to identify any malicious content.
  - Hash Matching: Comparing file hashes against a MongoDB database containing known malicious file hashes.
- **MongoDB Database:**
  - Malicious IP Repository: Storing a curated list of malicious IP addresses, continually updated.
  - Malicious File Hash Repository: Maintaining a comprehensive collection of malicious file hashes.
- **User-Friendly Interface:**
  - Dashboard: Providing users with ease of searching and file uploads, and a holistic view of threat analysis results.

1.6 Preliminary User's Manual

- **Dashboard Overview:** Upon visiting the web page, users will encounter a user-friendly dashboard with two primary options.
- **Checking Threats by Link or IP Address:** Users can enter a URL or IP address in the provided field and click "Submit" to cross-reference against a comprehensive database and VirusTotal.
- **File Analysis by File Upload:** Users can submit a file for analysis using the upload button, with supported file types including **'txt', 'pdf', 'png', 'jpg', 'jpeg', 'gif', 'zip', and 'exe'**.
- **Result Interpretation:** Analysis results will be displayed on the dashboard, indicating whether the link, IP address, or file is identified as malicious or not.

LITERATURE REVIEW

[1] Machine Learning has proven to be useful in detecting security threats, by analyzing security and log data to identify potential threats. Over the past decade ML techniques have been widely used to enable systematic learning and building of enterprise systems' normal profiles to detect anomalies and zero-day threats (Conti et al., 2018). ML includes a large variety of models in continuous evolution, presenting weak boundaries and cross relationships, and has already been successfully applied within various contexts in cybersecurity (Dua and Du, 2011; Ford and Siraj, 2014; Singh and Silakari, 2015; Buczak and Guven, 2016; Fraley and Cannady, 2017; Ghanem et al., 2017; Yadav et al., 2017; Apruzzese et al., 2018). The book by Dua and Du (2011) provides a comprehensive guide to how ML and data mining are incorporated in cybersecurity tools, and in particular, it provides examples of anomaly detection, misuse detection, profiling detection, etc. This study also provides a thorough analysis of where ML approaches can achieve maximum impact and a discussion of their limitations. The concluding chapters discuss emerging challenges and how ML and data mining can be used to effectively deal with them.

[2] Buczak and Guven (2016) survey ML and data mining approaches in intrusion detection whilst Fraley and Cannady (2017) discussed the future possibilities of incorporating ML into the cybersecurity landscape. In particular, problems such as malware detection, data breaches, profiling, etc. can significantly enhance the threats to organizations. Deep learning in cybersecurity has also been investigated (e.g., by Apruzzese et al., 2018). This study looks at whether current state of the art approaches in ML are effective for identifying malware, spam and intrusions and also allude to the current limitations of these approaches. Studies have also considered support vector machines for dealing with cybersecurity issues (Singh and Silakari, 2015; Ghanem et al., 2017; Yadav et al., 2017). Singh and Silakari (2015) explore support vector machines for cyber attack detection, and in similar fashion, Yadav et al. (2017) focus on the problem of classifying cyberattacks. The study by Ghanem et al. (2017) develop an intrusion detection system which is enhanced by support vector machines. Among other cybersecurity issues, the study by Ford and Siraj (2014) investigates ML approaches for detecting phishing, intrusions, spam detection, etc.

[3] As we have previously mentioned, a few studies have focused on supervised and unsupervised learning and in particular, methods such as support vector machines, artificial neural networks and deep learning for dealing with a range of cybersecurity related issues. Given the availability of large amounts of data, these approaches and their enhancements provide great potential for future work. Additionally, recent trends in ML have shown that reinforcement learning can be very effective (Sutton and Barto, 2018). In this direction, a recent study demonstrates the usefulness of a deep reinforcement learning approach for cybersecurity (Nguyen and Reddi, 2019). A number of problems such as the detection of intrusions, breaches, etc. can be effectively dealt with this approach given that they are constantly evolving. Another promising ML technique is modeling with Bayesian networks (BNs), which developed in the ML community since the late 1980s (Neapolitan, 2003; Korb et al., 2010). They are causal probabilistic models and there are several studies in a number of domains that demonstrate the applicability (Straub, 2005; Bonafede and Giudici, 2007; Fenton and Neil, 2012; Sýkora et al., 2018). The book by Fenton and Neil (2012) provides a comprehensive overview of how BNs can be applied to risk modeling in different domains, such as systems reliability, law, finance, etc. The recent study by Sýkora et al. (2018) shows how BNs can be used for risk assessment

in energy. Straub (2005) use BNs to study the risks associated with natural hazards and the study by Bonafede and Giudici (2007) investigates enterprise rick via BNs. As these studies show, they are particularly suited to modeling risk and can be very effective for the probable of modeling threats associated with data breaches.

[4] Preliminary studies have already demonstrated the usefulness of BNs in cybersecurity (Ramakrishnan, 2016; Wang et al., 2020). The study by Wang et al. (2020) shows that BNs can more accurately classify cybersecurity risk, especially compared to previously known Monte Carlo models. Furthermore, BNs also prove to be more flexible. Ramakrishnan (2016) shows how BNs can be used to model and visual the causal models underlying cyberrisks. Unlike other ML approaches, BNs are not back-boxes. Their main advantages are the ease of explaining their findings and the ability to perform a systematic sensitivity analysis. In the context of threats in cybersecurity, another key advantage of BNs, is that they can be used to build a causal model of the factors that contribute to threats. This can be achieved through expert elicitation (Kuusisto et al., 2015) (i.e., through knowledge derived from experienced professionals in the field) or built from data sources or a combination of both. In particular, BNs can be applied to problems in security to predict threats and potential data breaches and also to diagnose how these threats came about.

Problem Statement:

The increasing sophistication and frequency of cyber threats pose significant challenges to organizations worldwide. Despite advancements in cybersecurity technology, many organizations struggle to effectively detect and mitigate these threats in a timely manner. Existing threat intelligence platforms often lack scalability, robust security measures, and user-friendly interfaces, hindering organizations' ability to proactively defend against evolving cyber threats.

Objectives:

1. Enhance Threat Detection Capabilities: Develop a comprehensive threat intelligence platform capable of accurately identifying and analyzing various types of cyber threats, including malicious URLs, IP addresses, and files.

2. **Improve Scalability:** Design the platform to scale horizontally, allowing for seamless integration of new data sources and accommodating increasing volumes of threat data without compromising performance.

3. **Strengthen Security Measures:** Implement stringent security protocols, including input validation for search URLs and robust file upload validation, to mitigate the risk of potential cyberattacks such as buffer overflow and malicious uploads.

4. **Enhance User Experience:** Create a user-friendly interface with intuitive dashboard features that provide users with easy access to threat analysis tools, actionable insights, and real-time threat alerts.

Methodology:

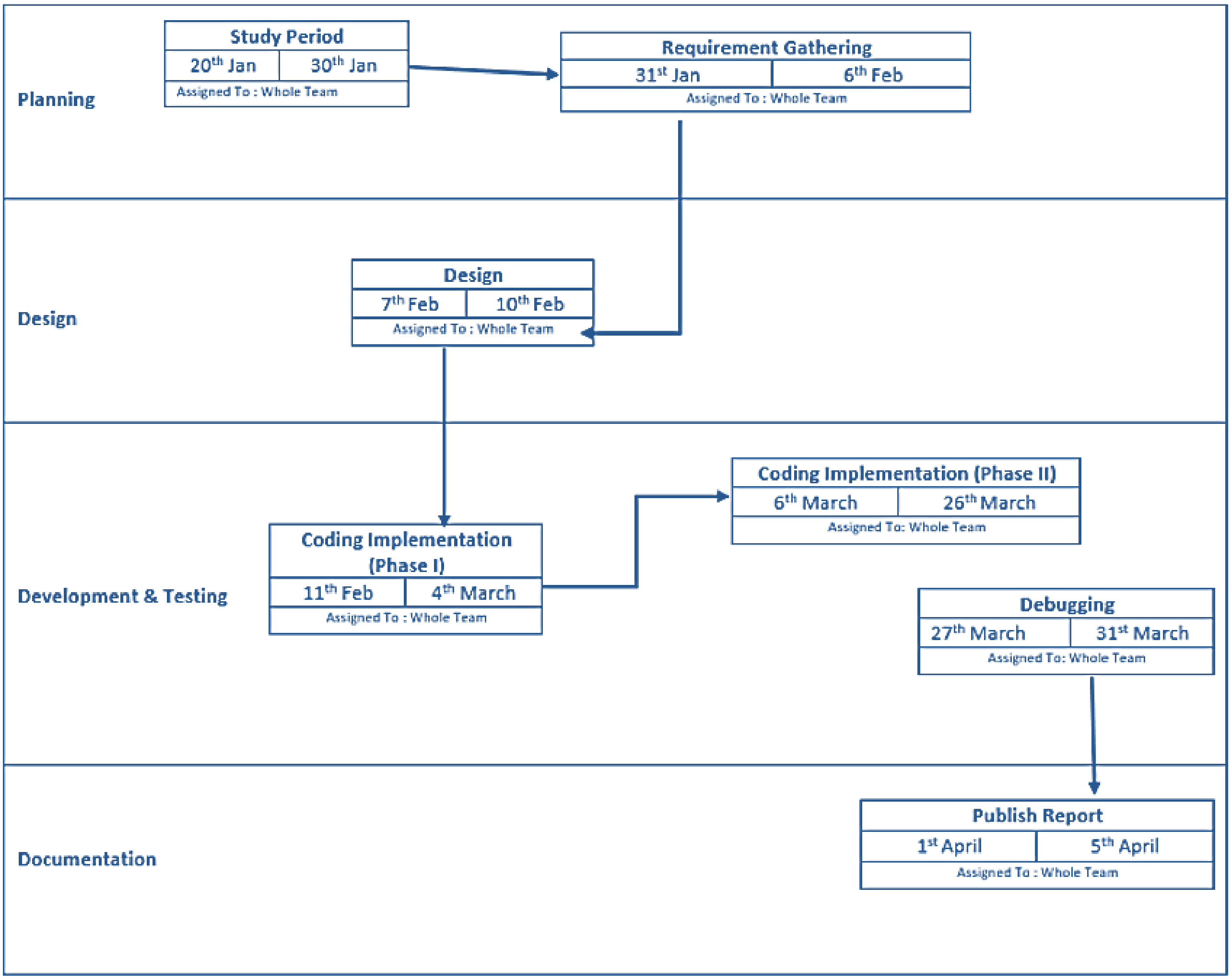
1. **Research and Analysis:** Conduct a comprehensive review of existing threat intelligence platforms, cybersecurity frameworks, and best practices to identify key requirements and challenges.

2. **Platform Design:** Collaborate with cybersecurity experts and software developers to design a scalable and secure threat intelligence platform architecture. Define the system's components, functionalities, and integration points.

3. **Development and Testing:** Utilize agile software development methodologies to iteratively build and test the platform's features and functionalities. Conduct rigorous testing to ensure the platform's reliability, scalability, and security.

4. **Implementation and Deployment:** Deploy the developed platform in a controlled environment, collaborating with pilot users to gather feedback and fine-tune the system. Implement necessary enhancements and optimizations based on user feedback

Pert Chart



References

[1] D. Wagner, K. Mahbub, E. Palomar, and A. E. Abdallah, "Cyber threat intelligence sharing: Survey and research directions," Computers & Security, vol. 87, p. 101589, 2019.

[2] W. Tounsi and H. Rais, "A survey on technical threat intelligence in the age of sophisticated cyber attacks," Computers & Security, vol. 72, pp. 212–233, 2018.

[3] J. Cha, S. K. Singh, Y. Pan, and J. H. Park, "Blockchainbased Cyber Threat Intelligence System Architecture for Sustainable Computing," Sustainability, vol. 12, no. 16, p. 6401, 2020.

[4] J. Zhao, Q. Yan, J. Li, M. Shao, Z. He, and B. Li, “TIMiner: Automatically extracting and analyzing categorized cyber threat intelligence from Social Data,” Computers & Security, vol. 95, p. 101867, 2020.

[5] Y. Gao, X. Li, H. Peng, B. Fang, and P. S. Yu, “Hincti: A cyber threat intelligence modeling and identification system based on Heterogeneous Information Network,” IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 2, pp. 708–722, 2022.