MINIMUM VIABLE CYBERSECURITY FRAMEWORK FOR PROTECTING CYBER ATTACKS FROM EXTERNAL THREAT VECTORS

by

Chidhanandham Arunachalam

DISSERTATION

Presented to the Swiss School of Business and Management Geneva

In Partial Fulfilment

Of the Requirements

For the Degree

DOCTOR OF BUSINESS ADMINISTRATION

SWISS SCHOOL OF BUSINESS AND MANAGEMENT GENEVA

March 2023

MINIMUM VIABLE CYBERSECURITY FRAMEWORK FOR PROTECTING CYBER ATTACKS FROM EXTERNAL THREAT VECTORS

by

Chidhanandham Arunachalam

APPROVED BY

____

<Chair's Name, Degree>, Chair

____

<Member's Name, Degree>, Committee Member

____

<Member's Name, Degree>, Committee Member

RECEIVED/APPROVED BY:

____

<Associate Dean's Name, Degree>, Associate Dean

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

DEDICATION

I extend my heartfelt gratitude and dedicate this research paper to the dedicated individuals who work tirelessly to safeguard our digital world from malicious hackers.

Their unwavering commitment to staying ahead of cybersecurity threats and protecting our sensitive data, assets, and infrastructure is critical in ensuring the security and privacy of our interconnected world.

As the unsung heroes of our digital age, they serve as the guardians of stability and trust in our online communities. Their diligent efforts do not go unnoticed and are greatly appreciated. May this research paper serve as a small token of appreciation for all that they do.

ACKNOWLEDGEMENT

Firstly, I would like to thank my research supervisor Dr. Minja Bolesnikov, for his guidance and support throughout the project. His expertise and feedbacks were invaluable in shaping this research. I would also like to thank the participants who generously gave their time and filled out the survey form. And a special thanks to CISO friends for taking out time for multiple rounds of interviews. Without their contribution, this research would not have been possible.

I would like to express my sincere gratitude to my guru, Sri Sri RaviShankar, for being a source of encouragement. I am grateful to the Chairman and Board Director of Sumeru, Dr. AL Rao, and Rajesh K. for providing the necessary resources and support to carry out this research. I am also indebted to my company Sumeru Software Solutions for sponsoring the research and supporting my growth.

A special thanks to Siva, the Principal consultant of Sumeru, to help me with the technical part and to provide support in designing the framework of this research. Furthermore, I thank Aswin & Reddy Prasad for helping and guiding me about python coding and data analysis. I appreciate the feedback and suggestions provided by my entire team of cybersecurity, who were always willing to provide valuable insights. A thank you note to Neha, content writer of Sumeru, for giving her creative inputs and to help to adhere to the thesis template and guidelines. And thank you to the entire team of Sumeru, for supporting and bearing my unavailability at work while completing this research paper.

Lastly, I acknowledge the blessings of my parents & in-laws and support of my wife and kids, , whose encouragement and patience was crucial in enabling me to devote the necessary time and effort to this research.

Thank you to all those who contributed to this project. I hope that these research findings will be useful and informative for future studies in this field.

ABSTRACT

In cybersecurity, an attack surface refers to the potential vulnerabilities and entry points that an attacker could use to compromise a system, network, or application. Thus, understanding and managing the attack surface is a critical component of effective cybersecurity, as it helps to reduce the risk of successful attacks and protect sensitive data and systems from unauthorized access or damage. Through this research, my main objective was to create a minimum viable cybersecurity framework for protecting cyber-attacks from external threat vector that helps in preventing and remediating the most common cyberattack threat vectors across industries, platforms, and threat landscapes with minimal effort. I used Alexa’s Top 1000 websites and 200 random websites as a source input and performed passive scans on those websites using the Threat Meter tool (An External Attack Surface Monitoring Tool built by Sumeru Software Solutions). From the scans, I obtained raw data containing classes such as Industry, Attack Vectors, Threat Vectors, Threat score, Total no of Threats, and Fail Ratio. To achieve the main objective, I first performed an initial data analysis on the raw data obtained from the scans and arrived at inferences based on the initial analysis. I then used the inferences to answer some questions which helped me to build the framework. Wherever initial analysis inference was inadequate, I performed data sampling over the raw data to arrive at new inferences. My goal was to build a security framework that would help in preventing and remediating the most common cyberattack threat vectors across industries, platforms, and threat landscapes with minimal effort.

Keywords: Alexa, Security, Attacks Vectors, Threats, Remediating, Proactive, Threat Meter, External Attack Surface Monitoring

CHAPTER I

INTRODUCTION

1.1 Introduction

The attack surface of a company can be defined as the sum of attack vectors that a cyber-criminal can use as an entry to gain access to private information. When organizations don't proactively protect their fort from threat vectors, they become the easy target for cyber-attackers.

Here are the challenges organizations face when they don't pay enough attention to their external attack surface –

- Organizations' security perspectives are mostly inside out and not outside in, resulting in zero visibility into shadow IT assets and external threats
- Working with third parties, vendors & partners without assessing their security posture will become a serious threat to the organization
- Organizations have no clue about rogue mobile apps and fake sites that cause brand impersonations. If organizations succeed in detecting threats like brand impersonation/phishing, taking it down is a gigantic task
- Managing the external attack surfaces and tracking new digital assets like cloud servers, containers, domains, and subdomains (since they're going public frequently with DevOps' speed and scale) is a challenging task
- Organizations have mechanisms to detect and prevent phishing threats only during the delivery of phishing emails and not in the early stage

VentureBeat (2022) states that 70% of companies had to go through a compromise due to unmanaged or poorly managed internet-facing assets. Since the average company takes around 80+ hours to manage and update the inventory of their external attack surface, it becomes hard to repeat the process frequently.

And that's why, according to Randori (2022), 75% of organizations depend on the spreadsheet to manage their external attack surface and less than 1 in every 3 organizations could find a potent solution to handle the complexities and chaos of their external attack surface. And investment in external attack surface has become the number one priority for large businesses in 2022 and around 67% of the nations around the world perceive that the external attack surface has been getting bigger and bigger.

While perusing through the research papers, I saw a clear gap in reviewing the holistic perspective of the external attack surface. My research area is throwing light on protecting the external attack surface of the organizations from a 360-degree point of view covering major threat landscapes and offering a minimum viable cybersecurity framework for protecting cyber-attacks from external threat vectors that can be immediately used by any organization. Any company that is willing to embark upon a cybersecurity initiative would be able to use this security framework immediately to effectively reduce its cyber risks significantly with less effort, minimal cost, and greater value.

1.2 Problem Statement

For any company today, there is no handy, holistic guide that can help to identify potential vulnerabilities and defend organizations from horrid cyber-criminals exploiting external attack surfaces.

Reviewing multiple vectors (which are presented in the hundreds of research papers I reviewed) could help address the attack perspective of the external attack surface individually, but the application perspective remained unaddressed since they didn't review the external attack surface from a holistic standpoint, which made the approach significantly less effective.

While analyzing the gap, I reached the same prognosis – there is a depth of records available for individual vectors, but not in conglomeration. In this regard, I tried to gather data from established companies and tried to put up a security framework that will help companies protect their fort from external attack vectors.

While gathering primary data (along with the secondary data that I have was cumulating from the research papers from cybersecurity thought leaders), I took the help of a tool Threat Meter that I co-created (Sumeru Threat Meter, n.d.). It ran100+ test cases on major threat vectors to detect and monitor the external attack surfaces of organizations so that I can ensure the accuracy of my analysis and offer a solid minimum viable cybersecurity framework (MVCSF) aligned with industry standards.

MVCSF refers to the tool, the processes (as per industry standards), the guidelines to read and prioritize, and the steps for remediation.

Every company needs something actionable to get started in their cybersecurity initiatives for their most ignored, external attack surfaces. Giving a solid framework will help them address the challenge with minimal effort and significant risk reduction.

1.3 Objectives

The long-term goal of the research is to arrive at Minimum Viable Cybersecurity Framework (MVCSF) that could be used by organizations (both, from start-ups and established companies) for protecting cyber-attacks from external threat vectors.

While developing an external attack surface monitoring framework for companies to use, here are the primary objectives of the research -

- To do a comprehensive review of the literature that is available and comprehend industry practices that are followed today
- To arrive at a list of major potential vectors from different threat landscapes which acts as the entry points for the hackers to get inside the organization
- To analyze thousands of data for the vectors that are identified above and infer the results to create inputs for MVCSF
- To outline a conceptual framework that can be used as a handy guide.

The research will be valuable to the entire start-up ecosystem, established companies, and anyone that are using digital assets. It will give them a clear direction and road map to act step by step and protect their external attack surfaces from cyber-attacks.

1.4 Hypothesis

Creating the MVCSF helps organizations protect their fort from external cyber-attacks. This is the chief aim of the research. Hence, my hypothesis of research will be as follows:

'Organizations that will use the MVCSF will get an effective and solid roadmap to prevent and predict the external attack surfaces with the least effort & significant risk reduction.'

'For organizations that will not take advantage of MVCSF, protection, and prevention of the external attack surfaces would be quite a complex activity.'

CHAPTER II

REVIEW OF LITERATURE

2.1 Introduction

The attack surface of a company can be defined as the sum of attack vectors that a cyber-criminal can use as an entry to gain access to private information. Attack surfaces can be categorized into external and internal attack surfaces.

VentureBeat (2022) states that 70% of companies had to go through a compromise due to unmanaged or poorly managed internet-facing assets. Since the average company takes around 80+ hours to manage and update the inventory of their external attack surface, it becomes hard to repeat the process frequently.

And that's why, according to Randori (2022), 75% of organizations depend on the spreadsheet to manage their external attack surface and less than 1 in every 3 organizations could find a potent solution to handle the complexities and chaos of their external attack surface.

Randori (2022) has also mentioned that investment in external attack surfaces has become the number one priority for large businesses in 2022 and around 67% of organizations around the world perceive that the external attack surface has been getting bigger and bigger.

Its increasing reach and the risks associated with the use of open source codes, complex digital supply chains, cloud applications, digital assets, and social media have turned out to be the top external threats for the horrid cyber-criminals.

Adding to this ever-changing dynamism of the ever-increasing external attack surface, the following elements enhance the risks of the organization's data being exposed as per:

- **Migration & adoption of cloud** – Assets that are exposed & vulnerable and the containers that store the datasets
- **The team that runs tests and works on development** – Emergence of modern assets & testing
- **Networks** – New Netblocks and Advertisements
- **Marketing** – New subdomains for landing pages hosted via external design companies
- **Sales** – Campaigns and e-Commerce
- **Operations of IT** – Modern Assets & Services, Patching, Changes in Configuration
- **Security Fixtures** – Modern assets, fixtures, deployments of agents
- **Mergers and Acquisitions** – Risks associated with newly acquired assets
- **Subsidiaries** – Complexities of assets not controlled
- **Supply Chain Risk** – Hosting providers, Third parties

In this review, I would discuss the elements of the external attack surface, i.e., the threat vectors and how the researchers had peeked into various domains to help curb cyber-threats.

2.2 Objective of the Research & Emergence of a New Cybersecurity Framework

While perusing through hundreds of research papers for my research subject, I got many research papers that are closely relevant to my topic, but unfortunately, I couldn't pinpoint a single paper that addresses an organization's attack surfaces from all angles.

And for writing this research paper, I will take a 360-degree perspective and understand each element of how the cyber-security thought leaders are solving the challenges that are all-pervasive.

Since there's no exact framework that's available (which can be applied immediately by the organizations), through my research work, I intend to offer a framework that covers all the aspects of the attack surface.

So, I picked up a bunch of research papers that are closely relevant and address individual areas e.g., network monitoring, security monitoring, attack surface management vulnerability management, emerging threat, cyber threat intelligence in risk management, etc.

I tried to pick only those papers that talked about the actual method of taking care of the threat vectors using manual and automated tools and also the ones that talked about attack surface management, manipulating the adversaries' point of view, and addressing risk management while using cyber threat intelligence.

By diving deep into the research papers to analyze the gap, I discovered that the cybersecurity thought leaders went to lengths to discover the jewels of solving each challenge backed up by both primary and secondary data.

Accumulating all these jewels from the cybersecurity thought leaders and after putting my research & interview with the cybersecurity thought leaders collectively I would like to offer a minimum viable cybersecurity framework (MVCSF) with the help of the tool (Threat Meter) that I co-created with my team.

And this framework would act as a guidebook to the organizations so that they can hold their fort against cyber-attacks.

I'm grateful to all of these thought leaders who have put so much into their respective papers. I will dive deep and look at each element and provide a brief overview of how these elements impact the external attack surface of an organization.

2.3 Definitions of Attack Surfaces

The Attack surface is used as a metaphor for the assessment of risk during the development of the software and also during maintenance. And since the attack surface is used for various purposes in cybersecurity, this study will show the light.

Christopher Theisen, et al. (2018) categorized a total of 644 papers related to the topic of the attack surface and determined the frequency with which the definitions of attack surface used in these papers are based on a citation, also determined the most frequently cited definitions for the phrase attack surface.

Based on their criteria, they recommend that researchers and practitioners choose an attack surface definition from one of the six identified themes with context-specific clues –

Methods: This theme consists of the implementation methods, channels of data, and the data inherent within the system. No particular attack features are mentioned.

Adversaries: Under this theme, the attack surface definition contains all the types of attacks an attacker could pursue to affect a system.

Flows: In this theme, the attack surface definition is depicted through the flow of control and data. No methods or possible types of attacks are not considered.

Features: Under this theme, the definition of the attack surface is the characteristics of the kinds of attacks on a target system.

Barriers: This attack surface definition focuses on the prevention of attacks by malicious parties.

Reachable Vulnerabilities: This attack surface is defined as the series of vulnerabilities exposed via flows or paths to the end users.

2.4 Reducing Application Attack Surface by OWASP Compliance

A system's attack surface is how much of its application area is exposed to adversaries.

Comparing similar applications or comparing applications with similar functionality but varying security risks can be achieved using the attack surface metric.

The attack surface metric can choose the right one by looking at the two applications that have similar functionalities. And then to estimate the security of the system, one needs to calculate the attack surface of the application.

When the attack surface of the web application is reduced, the vulnerability of the entire system gets reduced as well. The reduced attack surface then is used by programmers to improve their code, by testers to estimate the amount of testing needed, and by users to compare applications.

Sumit Goswami, et al. (2012) explained that to determine and compare the security of two versions of an in-house developed Project Management Web Application before and after OWASP compliance, various parameters of its attack surface were calculated based on a security audit.

2.5 Manipulating the Attacker's View of a System's Attack Surface

The reconnaissance phase is the stage where the cyber attackers seek to collect critical information about their target system, e.g., unpatched vulnerabilities, service dependencies, network topology, etc. The challenge is when the configurations are static, the cyber attackers would always be able to collect exact information about their target system and plan for desired exploits.

Massimiliano Albanese, et al. (2014) conducted a thorough analysis and figured that the problem could be solved from the perspective of control and proposed a graph-based approach to exploit and infiltrate the attacker's fundamental approach toward the system's attack surface.

Massimiliano Albanese, et al. (2014) discussed the system's attack surface such as open ports, operating system, web-pages content, etc., and changing the system's configuration dynamically to manipulate the system's attack surface to introduce uncertainty for the attackers. This would deceive the attackers and steer them away from critical resources and forces them to use a random strategy.

2.6 Eliminating the Hypervisor Attack Surface for a More Secure Cloud

Cloud computing has been the go-to platform for the majority of organizations. And virtualization enables cloud providers to host services for a large number of customers. But when I talk about virtualization software, its attack surface is way too complex and large.

As a result, it is prone to bugs and vulnerabilities that can be exploited by malicious virtual machines (VMs) to attack or obstruct other VMs - a major concern for organizations moving “to the cloud.”

Instead of hardening or minimizing the virtualization software, I eliminate the hypervisor attack surface by running guest virtual machines natively on the underlying hardware while maintaining the ability to run multiple VMs concurrently.

The NoHype system incorporates four key concepts:

- Pre-allocating processor cores and memory resources,
- Virtualizing I/O devices,
- Modifying the guest OS to perform all system discovery during bootup, and
- Avoid indirection by bringing the virtual machine closer to the hardware.

As per Jakub Szefer, et al. (2011), a hypervisor is therefore not required to assign resources dynamically, emulate I/O devices, support system discovery after bootup, or map interrupts and other identifiers.

With NoHype, customers specify resource requirements ahead of time and providers offer a variety of guest OS kernels.

2.7 State of the Art in Network Security Monitoring (NSM)

When it comes down to network security, it focuses fully on preventing cyber-attacks. And there are four steps through which a network security monitoring (NSM) system works –

- Monitoring
- Detection
- Diagnosis
- Response/Course-correction

The objective of the network is to monitor the condition of a network to identify any abnormal events and manage timely. It's one of the most challenging tasks since the network is all pervasive and produces a gigantic data set at a superfast pace.

Marta Fuentes-Garcia, et al. (2021) reviews the state-of-the-art in Network Security Monitoring (NSM) and derives a new taxonomy of the functionalities and modules in an NSM system.

This taxonomy is useful to assess current NSM deployments and tools for both researchers and practitioners. This taxonomy classifies such components as sensors, parsers, integrators, detectors, inspectors, and actuators. These modules can be combined in different ways, yielding a powerful and scalable architecture for incident detection. This work highlights the strengths and weaknesses of the identified modules as below –

- The NSM philosophy and how the modular schemes of classification for detection and response structures work as per the philosophy

- The classification of trade solutions as per the scheme
- The identification and examination of Network Security Monitoring for modern network
- Trending and upcoming challenges in network security as per the new paradigm of communication

2.8 Emerging Threats in Cybersecurity

Julian Jang-Jaccard, et al. (2014) focuses on two aspects of information systems: understanding vulnerabilities in existing technologies and emerging threats in upcoming advancements in telecommunication and information technologies.

Growing threats have been found in emerging technologies, such as social media, cloud computing, smartphone technology, and critical infrastructure, often taking advantage of their unique characteristics. They described the characteristics of each of the emerging technologies and various ways malware is being spread in these new technologies.

The developments of next-generation secure Internet and trustworthy systems have been suggested as important areas of research to look into in the future.

The development of global-scale identity management and traceback techniques to enable tracking down adversaries has also gained attention as an important issue to address in the future.

Singh (2012) discussed how the internet has grown and has become a very important component of life and explains why internet monitoring is important. This paper presents a bird-view of various cyber-criminal methods, countermeasures, and challenges posed by cyber security.

2.9 Security Monitoring of the Cyber Space

Fachkha (2016) discussed the rise of information sharing and increased internet usage but the users and the fact that computer attack tools and techniques are becoming more intelligently designed and hackers are capable of launching worldwide impacting attacks for various reasons such as large-scale denial-of-service, cyber-terrorism, information theft, hate crimes, defamation, bullying, identity theft, and fraud.

And proposes Trap-based Cyber Security Monitoring Systems to collect insights on the attack traces and activities such as probing/scanning for vulnerable services, worm propagation, malware downloads, and other command-and-control activities such as executing DDoS cyber-attacks using Botnet for further investigation.

Fachkha (2016) also mentioned that the idea behind these trap-based monitoring systems is to detect major cyber threats that exist on the Internet now. Here are some of the three most critical methods using which one can conduct the synthesis and analysis of these sensor-based monitoring systems –

- **Darknet Deployment:** Another name for the darknet is network telescope. Darknet a the series of routable IP addresses that are typically unused. And if any traffic that is destined for them seems suspicious, immediate action is taken through darknet deployment. Through darknet deployment, a sensor monitoring system is installed to understand the architecture of the darknet.

- **IP Gray Space Deployment:** IP Gray Space is almost identical to the darknet. The only difference between IP Gray Space and the darknet is for the former the IP address is unused for a limited time, i.e. one day or an hour, and for the darknet, it's unused permanently. IP Gray Space deployment is done when the IP addresses are in passive or inactive mode.
- **Honeypot Deployment:** It's a system that's connected to the internet to trap cyber attackers. The nature of honeypots is similar to the darknet, but honeypots have a specific goal to achieve i.e. to interact with the cyber attackers, as a result, honeypot deployment requires more resources than darknet deployment. Typically, there are three categories of honeypots, e.g., low interactive honeypot, medium interactive honeypot, and high interactive honeypot.

2.10 Cyber Threat Prediction with Machine Learning

Arvind Kok, et al. (2020) addresses the approaches, techniques, and results of applying machine learning techniques for cyber threat prediction.

Timely discovery of advanced persistent threats is of utmost importance for the protection of NATO's and its allies' networks. The experiments executed and described in this paper address data preparation and machine learning for technique and tactic prediction; potentially preparing for APT discovery.

Experiments for both known and unknown techniques are explored. At the time of conducting the Coalition Warrior Interoperability Exercise (CWIX), Red-Blue Team Simulation captured the event data. After that, the data set went through various Machine Learning techniques – clustering with outliers, auto encoding, deep learning, etc.

This work did not explore the possibilities of applying prediction techniques in operational systems or linking results to operational challenges.

2.11 Attack Surface Management of Top Global Enterprises

Palo Alto (2021) showcases the lessons in Attack Surface Management from Leading Global Enterprises. The research team studied the public-facing internet attack surface of some of the world's largest businesses. They monitored scans of 50 million IP addresses associated with 50 global enterprises to understand how quickly adversaries can identify vulnerable systems for fast exploitation and published their key findings.

PaloAlto (2021) portrays the following key elements –

- **Cyber-criminals are active all the time:** The attackers are always active, always meaning 24*7. The attackers conduct one scan every hour since the remote working scenarios have drastically increased to locate any vulnerability; whereas the global enterprise conducts a scan once a week.

- **Attackers act immediately:** Whenever there's any vulnerability is announced, attackers are super-fast to act on it. As a result, it becomes harder for a global enterprise to prevent the attack.

- **One-third of all security challenges happen due to Remote Desktop Protocol (RDP):** To be more accurate, RDP causes around 32% of security issues. Other than RDP, exposure to zero-day vulnerability, virtual network computing (VNC), misconfigured database servers, etc. are the reasons for critical security challenges.

- **Cloud footprint is the chief reason for the most critical security concerns:** In around 79% of the cases, cloud footprint remained responsible for critical security challenges in global organizations. It could be due to the drastic increase in remote work during and post-COVID-19.

2.12 How Attack Surface Management (ASM) Complements Vulnerability Management

When a company fails to identify and monitor its Internet attack surface (no attack surface management), the company exposes itself to the risk of a probable breach even if it’s utilizing the power of vulnerability management scanners. And that’s where lies the importance of an attack surface management (ASM) tool.

ASM helps an organization in identifying and monitoring its ever-expanding cloud-centric businesses and assists in increasing the visibility of assets to complement using a vulnerability scanner.

In this whitepaper, the focus is Attack Surface Management (ASM) and how ASM differs from and complements vulnerability management (VM).

Censys (2020) elaborated on how ASM tools discover new, previously unknown assets, which they then feed to vulnerability management tools. As a result, the combination of ASM and VM performs in-depth, detailed assessments of specific vulnerabilities present on hosts. A partnership that shortens the time between asset deployment and discovery and remediation of any vulnerabilities now exposed improves the overall security posture of the modern, online business.

2.13 Threat from the Dark – Research through Threat Intelligence

If you want to take proactive measures against cyber-attacks, it's wiser to conduct a thorough analysis of the contents of the Dark Web to understand the nitty-gritty of the criminal minds.

If you want to curb the cyber-crimes, the essential step should be either to take a peek into the Dark Web or to take an integrated approach of looking into both the Surface Web and the Dark Web.

Randa Basheer, et al. (2021) go into detail about the rapid increase in quantity and complexity of cyber threats emerging from different parts of the Internet and proposes Cyber Threat Intelligence (CTI) as a solution to tackle the challenge.

CTI leverages multiple information sources and produces valuable insights, analytics, and knowledge for decision-makers to take proper actions against cyber threats.

One of the most crucial sources is the Dark Web, which is growingly earning great interest from researchers due to its richness of information related to cyber threats presented by cyber criminals on different sorts of platforms such as forums (discussions, tutorials, and assets) and marketplaces (offered products and services).

2.14 Role of Cybersecurity in M&A

As per Deloitte (2021), 62 percent of participants in a recent survey by Forescout agree that acquiring new companies poses significant cybersecurity risks, and cyber risk is their biggest concern after acquiring them.

As per the estimation, by this year, i.e., 2022, 60 percent of the companies would consider cybersecurity posture as a critical factor of their due diligence process during any merger & acquisition.

Deloitte (2021) has highlighted four types of risks during any merger and acquisition –

- Technology disruption
- Dormant threats
- Information Technology (IT) resiliency risk
- Data security

Here are the three particular steps Deloitte (2021) recommends for reducing the cyber risk during M&A –

- **Cybersecurity Protection (CSP):** This step is recommended at the pre-stage of M&A. It will help you defend against emerging threats.
- **Cyber Vigilance & Operations (CVO):** This step you should take during M&A. This ensures having the threat intelligence and situational awareness to detect any harmful vulnerability.

- **Cyber Resilient:** It's done post-M&A. This step will ensure that you can recover from any mishap and minimize the impact.

2.15 Evaluating and Mitigating Software Supply Chain Security Risks

Managing and mitigating supply chain risk is critical when the focus is on manufacturing. The goal is typically to minimize production disruptions and to prevent low-quality or counterfeit products from being incorporated into systems, with a focus on manufacturing.

In software supply chain risk management, some of these aspects are present (e.g., a system may depend on the timely delivery of a subcontractor's products), but as software can be modified easily, the supply chain's focus shifts to -

- Minimizing the potential for unauthorized changes, and
- Having adequate methods for obtaining confidence that such opportunities have been minimized, particularly among lower-level participants.

Furthermore, software systems are more likely to be modified unauthorizedly than hardware systems because they can be configured and used in ways that increase security risks.

To manage supply chain security risks, it is necessary to consider how security risks could be introduced during the deployment, configuration, and operation of the software system, as well as during its design and development.

Robert J. Ellison, et al. (2010) focus on the following aspects of software supply chain security risk –

- Identification of software supply chain security risks throughout the acquisition life cycle
- Specifying the evidence that is gathered to understand if the risks are properly mitigated

Throughout the acquisition life cycle, Robert J. Ellison, et al. (2010) demonstrate how the gathered evidence is incorporated into an argument to demonstrate that supply chain security risks have been adequately addressed.

Identifying and monitoring an attack surface and developing and maintaining a threat model are two of the key strategies for reducing security risk outlined in the reference model.

2.16 Cyber Threat Intelligence in Risk Management

Amira M. Aljuhami, et al. (2021) perused 65 research papers and pursued a comprehensive examination of cyber threat intelligence (CTI) and risk management practices.

This study aims to review the impact of cyber threat intelligence on risk management in Saudi universities in mitigating cyber risks.

This study talks about the need to improve the defenses using cyber threat information(CTI), as CTI represents information about the nature of threats and a deep understanding of the attacker's objectives and thus the ability to respond to threats and take appropriate defensive measures.

The nature of cyber threats has been changing drastically. To deal with this huge flow of information and the ever-changing nature of cyber-threats, one needs advanced and deep information about the actual nature of these cyber-threats and also measures to deal with them on time.

The utilization of information on cyber threats in the management of risks expands the capacity of mitigators to mitigate the threats timely.

2.17 Cyber Threat Intelligence Framework for Improved Internet Facilitated Organized Crime Threat Management

The crime threats that are internet facilitated target the citizens and the companies as a whole. They are propagated typically via worms and botnets.

Even if various models are emerged to assess the intensity and impact of these threats with the sole intention of combatting them, they're not as technologically efficient as they need to be.

Oriola (2018) reviews the state-of-the-art in Cyber-Threat Intelligence with a focus on Threat Management.

The paper identifies the strengths and limitations of the works and proposes a Cyber-Threat Intelligence framework that maintains the strengths in the existing models and addresses the limitations for better Internet-facilitated Organized Crime Threat Management.

2.18 Summary of the Literature Review

The idea behind perusing through so many research papers, journals, and white papers was to find a multi-dimensional approach to look at and assess the external attack surface of an organization and how horrid cyber-criminals could be prevented from infiltrating.

From looking at the attack surface definitions to understanding emerging threats in cyberspace to network security monitoring to use machine learning in threat detection to understanding the critical role of cyber threat intelligence (CTI) in risk management – all theoretical frameworks advance us toward a few key factors.

Reviewing multiple vectors (which are presented in the hundreds of research papers I reviewed) helped in addressing the attack perspective of the external attack surface individually, but the application perspective remained unaddressed since they didn't review the external attack surface from a holistic standpoint, which made the approach significantly less effective.

While analyzing the gap, I reached the same prognosis – there is a depth of records available for individual vectors, but not in conglomeration. In this regard, I'm trying to gather data from established companies and trying to put up a framework that will help companies protect their fort from external attack vectors.

Since there's no exact framework that's available (which can be applied immediately by the organizations), through my research work, I intend to offer MVCSF that covers the significant area of the external attack surface.

Since I needed to look at the solution from all angles, I took inspiration from these research studies, white papers, and detailed reports to come up with a comprehensive approach.

CHAPTER III

METHODOLOGY

3.1 Overview of the Research Problem

The emergence of connected technologies has brought forth the modern threat points e.g. VPNs, marketing campaign managers that use external infrastructures, IaaS & SaaS providers, third-party vendors, challenges of shadow IT & BYOD, etc. As a result, the scope, size, and reach of the modern attack surface has been increasing every minute.

Proactive external attack surface management has become increasingly important than ever before as organizations face an expanding threat landscape and unprecedented level of attacks. The road toward the least resistance is the most loved path for the cyber-criminals since they hope to take advantage of any blind spots the organizations have missed out.

According to IBM (2020), all it takes is one exploitable weak point for an attacker to get inside any business and steal customer data; on average, it takes 280 days to detect and contain a data breach, and remediation can cost upwards of $8 million in the United States.

In my research, I will go in-depth on how organizations can be aware of their external attack surfaces, take proactive actions, and will also suggest industry recommendations through which they would be able to hold the fort proactively.

3.2 Operationalization of Theoretical Constructs

3.21 Data collection

Data is collected from the Threat Meter tool. Based on the scan results I have obtained four different datasets.

- **First Dataset** has Industry, Attack vectors, Total risk count, Threat score and Fail ratio of Alexa's Top 1000 websites.
- **Second Dataset** has Industry, Attack vectors, Total risk count, Threat score, and Fail ratio of 200 random websites.
- **Third Dataset** has Industry, Threat vectors, and Total risk count of 200 random websites. Deeper scans to identify threat vectors are performed only for the 200 companies and not for Alexa's Top 1000, as it involves significant effort and time to identify the emerging cyber threats such as phishing domains, data leaks on the internet and dark web, brand impersonation, data breaches, rogue apps.
- **Fourth Dataset** has the average cost of each attack vector and threat vector obtained from the IBM data breach report 2022.

Research Methodology:

The primary research method of this research was to conduct a comprehensive review of the available literature and the industry practices that are followed.

The research followed the following data collection methodology –

Phase 1: Scanning of 1000 top Alexa websites based on the following eight vectors (using Threat Meter tool, co-created by me with my team) –

- SSL Health

- IP Reputation
- DNS Health
- Public Data Leaks
- Site Reputation
- Service Misconfigurations
- Unnecessary Open Ports
- Outdated Component

Phase 2: Scanning of 200 companies in depth with the tool, Threat Meter, which is based on the five parameters –

- Phishing Threats
- Data Leaks
- Brand and Reputation Threats
- Data Breaches
- Rogue Mobile Apps

Phase 3: After data collection, I employed the following appropriate method/s as per the 'Selection of Appropriate Statistical Methods for Data Analysis', authored by Prabhaker Mishra, et al. (2019). And later through these method/s, a thorough analysis was done to derive the answers of the research questions and validate the hypothesis to determine the MVCSF.

- Descriptive Statistics (Mean & Median)
- Inferential Statistics (Parametric for normal distribution & Non-parametric for continuous data with non-normal distribution)

Phase 4: As a result of this detailed analysis, and inference of data, a questionnaire with 21 questions on Attack Surface Management on Typeform was created.

- Reached out to 30 CISOs to fill out the survey form.
- Inferred survey questions by qualitative data analysis.

Phase 5: Using the above data, I created MVCSF to help companies get started with cybersecurity initiatives for protecting their external attack surfaces.

Phase 6: I then validated the framework with the top 5 CISOs by taking their interview to ensure easy adoption and significant risk reduction.

3.22 Data Observations

In data observations, I have described each column/field/variable in the raw dataset which helped me to perform better analysis and ease my decision-making process.

Attack Vectors of 1000 data points

The different columns/fields/variables in the dataset have high cardinality, high correlation, uniform distribution, and constant value.

Table 3

Data Statistics Variables for 1000 Data Points and their Different Types

Dataset statistics	Values
1000 attack vectors dataset statistics	
Number of variables	14
Number of observations	1000
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Variable types	
Numeric	12

Categorical	2

Table 3.1

High Cardinality, High Correlation, and Uniformity between Variables for 1000 Data Points

Variables Names	Description
The domain name has a high cardinality: 1000 distinct values	High cardinality
Threat Score is highly correlated with the Fail Ratio and Service Misconfiguration, Outdated Version, and Unnecessary Open Ports	High correlation
The Fail Ratio is highly correlated with the Threat Score and SSL Health, Service Misconfiguration, Outdated Version, Unnecessary Open Ports, Total Risks Count	High correlation
SSL Health is highly correlated with the Fail Ratio and Total Risks Count	High correlation
Service Misconfiguration is highly correlated with Threat Score, Fail Ratio, and Total Risks Count	High correlation
The outdated Version is highly correlated with the Threat Score Fail Ratio and Total Risks Count	High correlation
Unnecessary Open Ports are highly correlated with Threat Score Fail Ratio and Total Risks Count	High correlation
Total Risks Count is highly correlated with Threat Score, Fail Ratio, SSL Health, Service Misconfiguration, and Outdated Version	High correlation

Data Breaches are highly skewed ($\gamma 1 = 22.32704629$)	Skewed
A domain name is uniformly distributed	Uniform
The domain name has unique values	Unique

Attack Vectors of 200 data points

The different columns/fields/variables in dataset have high cardinality and high correlation.

Table 3.2

200 Attack Vectors Data Points

Dataset statistics	Values
Number of variables	12
Number of observations	200
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	6
Duplicate rows (%)	3.0%
Variable types	
Categorical	5
Numeric	7

Table 3.3

High Cardinality, High Correlation, and Uniformity between Variables for 200 Data Points

Variable Names	Description
Dataset has 6 (3.0%) duplicate rows	Duplicates
Threat score is highly correlated with Fail Ratio, SSL Health, IP Reputation, Service Misconfiguration, Outdated Version, Data Leaks, and Total Risks Count.	High correlation
Fail Ratio is highly correlated with Threat Score, SSL Health, IP Reputation, Service Misconfiguration, Outdated Version, and Data Leaks.	High correlation
SSL Health is highly correlated with Threat Score, Fail Ratio, Service Misconfiguration, and Total Risk Count	High correlation
IP Reputation is highly correlated with Threat Score, Fail Ratio, and Total Risk Count	High correlation
Service Misconfiguration is highly correlated with Threat Score, Fail Ratio, SSL Health, Outdated Version, and Total Risk Count	High correlation
Outdated Version is highly correlated with Threat Score, Fail Ratio, Service Misconfiguration and Total Risk Count	High correlation
Data Leaks is highly correlated with Threat Score	High correlation
Total Risk Count is highly correlated with Threat Score, SSL Health, IP Reputation, Service Misconfiguration, Outdated Version, and Fail Ratio	High correlation

- **Variable and Their Details**
    - Variable from each column or field in the data set are used in statistics (Quantile statistics, Descriptive statistics).
- **Threat Score Details for 1000 data points:**
    - Threat score **has more real numbers** and is **highly correlated** with Fail Ratio, SSL Health, IP Reputation, Service Misconfiguration, Outdated version, Data leaks, Total Risk count.
    - I found Threat Score has **mild negative skewness of -0.02014465438, kurtosis value of 2.166207698** and is **non-monotonic.**
    - When I removed zeros from the data, **normal distribution** was observed.

Table 3.4

Threat Score Data, Quantile, and Statistical for 1000 Data Points

Dataset Statistics	Values
Basic 1000 Data Statistical	
Distinct	118
Distinct (%)	11.8%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%

Mean	75.927
Minimum	0
Maximum	252
Negative	0
Negative (%)	0.0%

Quantile statistics

Minimum	0
5-th percentile	32
Q1	59
Median	77
Q3	92
95-th percentile	118
Maximum	252
Range	252
Interquartile range (IQR)	33

Descriptive statistics

Standard deviation	27.55102048
Coefficient of variation (CV)	0.3628619659
Kurtosis	2.166207698
Mean	75.927
Median Absolute Deviation (MAD)	16
Skewness	-.02014465438
Sum	75927
Variance	759.0587297

Monotonicity	Not monotonic

- Threat Score histogram with fixed size bins (bins=50) frequency data is shown below:

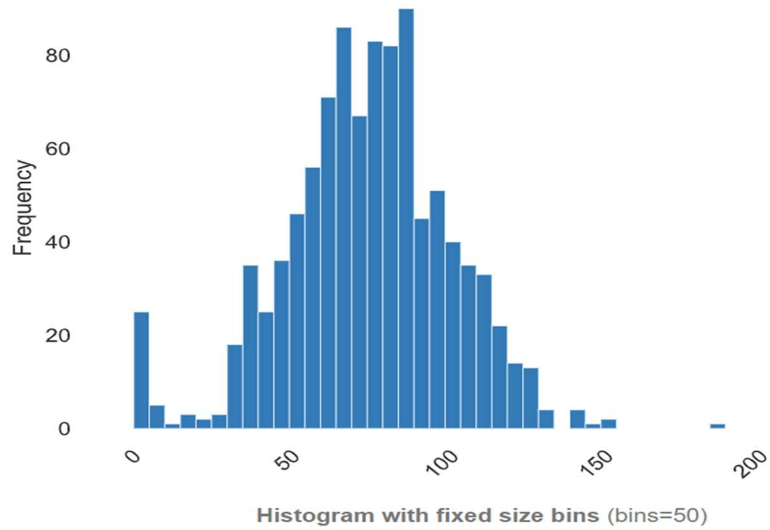

Figure 3

Threat Score Histogram Data

- **Threat score for 200 data points:**
    - Threat Score is highly correlated with 7 fields in the data set as Fail Ratio, SSL Health, IP Reputation, Service Misconfiguration, Outdate Version, Data Leaks, Total Risk Count.
    - We found the Threat Score has **negative skewness of -0.5669081818, has kurtosis value of 0.3455649374** and is **non-monotonic.**
    - Threat Score is normally distributed with less negative skewness in data.

Table 3.5

Threat Score data, Quantile and Statistical for 200 Points

Dataset Statistics	Values

Basic 200 Data Statistical	
Distinct	95
Distinct (%)	47.5%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	94.945
Minimum	-1
Maximum	234
Negative	8
Negative (%)	4.0%
Quantile statistics	
Minimum	-1
5-th percentile	0
Q1	75.25
Median	105
Q3	124
95-th percentile	158.1
Maximum	234
Range	235
Interquartile range (IQR)	48.75

Descriptive statistics	
Standard deviation	45.66473822
Coefficient of variation (CV)	0.4809599054
Kurtosis	0.3455649374
Mean	94.945
Median Absolute Deviation (MAD)	25
Skewness	-0.5669081818
Sum	18989
Variance	2085.268317
Monotonicity	Not monotonic

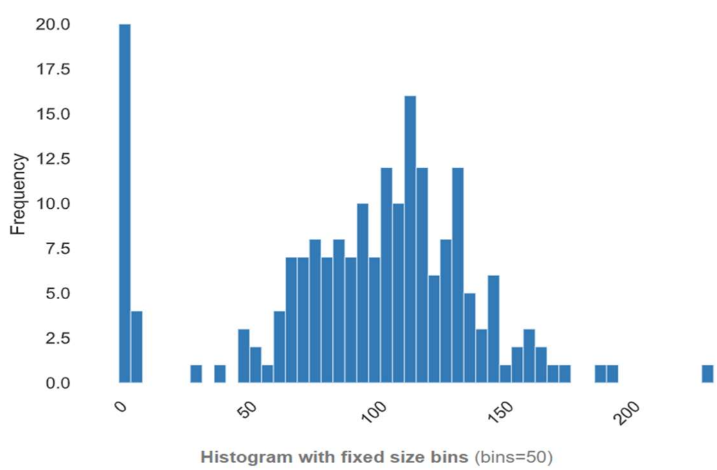

Figure 3.1

Threat Score Histogram for 200 Data Points

- **Fail Ratio Details for 1000 Data Points**:

- The Fail Ratio is highly correlated with the Threat Score and SSL Health, Service Misconfiguration, Outdated Version, Unnecessary Open Ports, Total Risks Count.
- We found the Fail Ratio has **negative skewness of -0.6136089028, has kurtosis** value of **1.730948557** and is **non-monotonic.**
- Fail ratio has **normal data distribution**.

Table 3.6

Fail Ratio for 1000 Data, Quantile, and Statistical Details

Dataset Statistics	Values
Basic 1000 Data Statistics	
Distinct	30
Distinct (%)	3.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	17.014
Minimum	0
Maximum	45
Negative	0
Negative (%)	0.0%
Quantile statistics	

Minimum	0
5-th percentile	8
Q1	15
Median	18
Q3	20
95-th percentile	25
Maximum	45
Range	45
Interquartile range (IQR)	5
Descriptive statistics	
Standard deviation	5.599567765
Coefficient of variation (CV)	0.329115303
Kurtosis	1.730948557
Mean	17.014
Median Absolute Deviation (MAD)	3
Skewness	-0.6136089028
Sum	17014
Variance	31.35515916
Monotonicity	Not monotonic

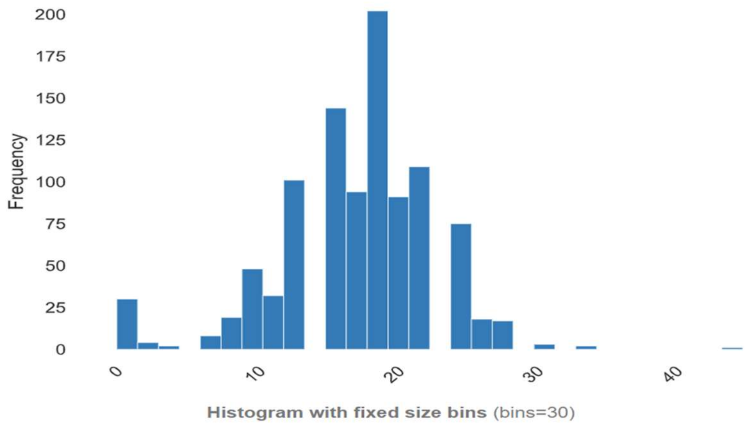

Figure 3.2

Fail Ratio Histogram

Fail Ratio Details for 200 data points:

- Fail Ratio has high correlation with Threat score, SSL Health, IP Reputation, Service misconfiguration, outdated version, Total Risk count.
- I found that Fail Ratio has **negative skewness of -1.218323509, has kurtosis** value of **0.67292356922** and is **non-monotonic.**
- Fail ratio has **normal data distribution.**

Table 3.7

Fail ratio for 200 attack vectors

Dataset Statistics	Values
Basic 200 Data Statistics	
Distinct	23
Distinct (%)	11.5%
Missing	0

Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	19.885
Minimum	0
Maximum	34
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	17
Median	22
Q3	26
95-th percentile	30
Maximum	34
Range	34
Interquartile range (IQR)	9
Descriptive statistics	
Standard deviation	8.598338919
Coefficient of variation (CV)	0.4324032647
Kurtosis	0.6729235692

Mean	19.885
Median Absolute Deviation (MAD)	4
Skewness	-1.218323509
Sum	3977
Variance	73.93143216
Monotonicity	Not monotonic

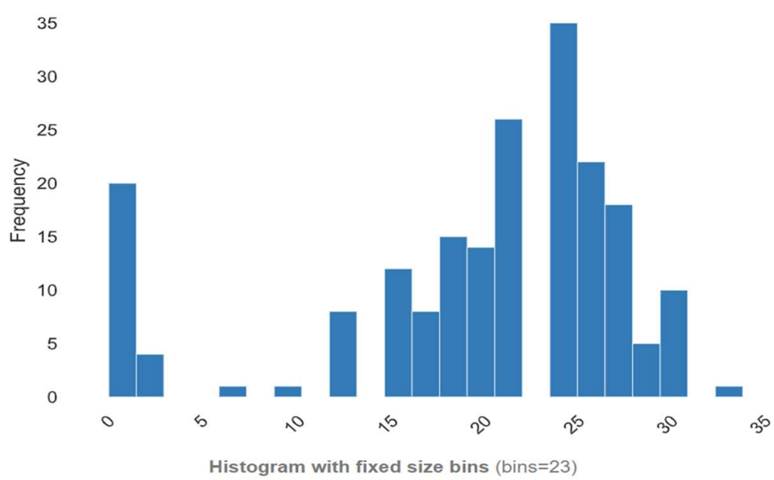

Figure 3.3

Fail Ratio for 200 Points

- **SSL Health Details for 1000 data points**:
  - SSL Health is influencing other variables such as Threat Score, Fail Ratio, and Total Risk Count in different industries and domains.
  - We observed that **122 websites were not affected by any SSL Health issue**, 107 websites were affected by 1 issue, 304 websites were affected by 2 issues, 350 websites were

affected by 3 issues, 88 websites were affected by 4 issues and 29 websites were by 5 issues.

- We found the SSL Health has **negative skewness of -0.2394763755, negative kurtosis value of -0.2890173848** and is **non-monotonic.**
- SSL Health has **negative data distribution.**

Table 3.8

SSL Health Data, Quantile and Statistical Details

Dataset Statistics	Values
Basic Data Statistics	
Distinct	6
Distinct (%)	0.6%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	2.262
Minimum	0
Maximum	5
Negative	0
Negative (%)	0.0%
Quantile statistics	

Minimum	0
5-th percentile	0
Q1	2
median	2
Q3	3
95-th percentile	4
Maximum	5
Range	5
Interquartile range (IQR)	1
Descriptive statistics	
Standard deviation	1.221002394
Coefficient of variation (CV)	0.5397888569
Kurtosis	-0.2890173848
Mean	2.262
Median Absolute Deviation (MAD)	1
Skewness	-0.2394763755
Sum	2262
Variance	1.490846847
Monotonicity	Not monotonic

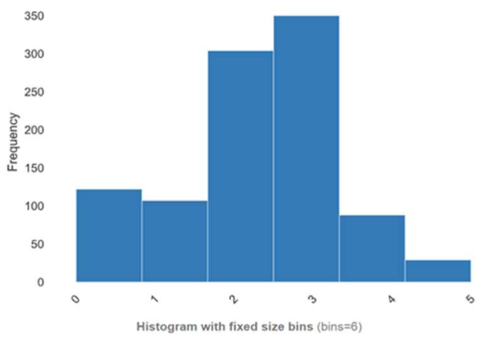

Figure 3.4

SSL Health Histogram Frequency

SSL Health Details for 200 data points:

- SSL Health is highly correlated with Threat Score, Fail Ratio, Service Misconfiguration, Total risk count.
- We observed that **39 websites were not affected by SSL Health issues**, 11 websites were affected by 1 issue, 37websites were affected by 2 issues, 65 websites were affected by 3 issues, 34 websites were affected by 4 issues, 12 websites were affected by 5 issues and 2 websites were affected by 6 issues. **Out of 200 websites, 161 have at least 1 SSL Health**.
- We found the SSL Health has **negative skewness of -0.238964295, kurtosis value of -0.7404127344** and is **non-monotonic.**

Table 3.9

SSL Health Data Statistics for 200 Data Points

Dataset Statistics	Values
Distinct	7

Distinct (%)	3.5%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	2.44
Minimum	0
Maximum	6
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	1.75
median	3
Q3	3
95-th percentile	5
Maximum	6
Range	6
Interquartile range (IQR)	1.25
Descriptive statistics	

Standard deviation	1.535640243
Coefficient of variation (CV)	0.6293607552
Kurtosis	-0.7404127344
Mean	2.44
Median Absolute Deviation (MAD)	1
Skewness	-0.238964295
Sum	488
Variance	2.358190955
Monotonicity	Not monotonic

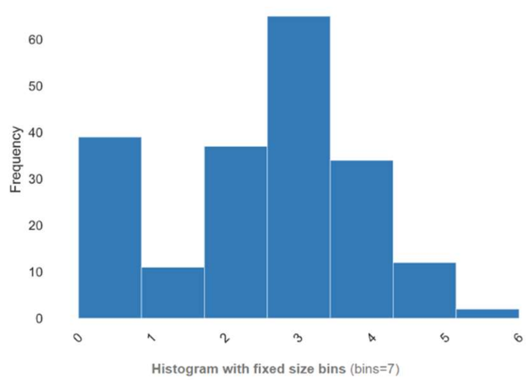

Figure 3.5

SSL Health Data Distribution for 200 Data Points

- **IP Reputation details for 1000 data points**:
    - IP Reputation influences other variables such as Threat Score, Fail Ratio, and Total Risk Count in different industries and different domains as well.
    - The Data distribution of IP Reputation is left skewness.

- We observed that **25 websites were not affected by IP Reputation issues**, 869 websites were affected by 1 issue, 19 websites were affected by 2 issues, 52 websites were affected by 3 issues, 23 websites were affected by 4 issues, 3 websites were affected by 5 issues, 7 websites were affected by 6 issues, 1 website is affected by 7 issues and 1 website is affected by 21 issues. **Out of 1000 websites, 975 have at least 1 IP Reputation**.
- Inference for IP Reputation kurtosis is 139.68686422.
- Inference for IP Reputation skewness is 8.746719491 with positive skewness and the data is normally distributed.
- IP Reputation has **positive skewness of 8.746719491, positive kurtosis of 139.68686422** and is **non-monotonic.**

Table 3.10

IP Reputation Data, Quantile, and Statistical Details

Basic Data Statistics	
Distinct	9
Distinct (%)	0.9%
Missing	0
Missing (%)	0.0%
Infinit	0
Infinite (%)	0.0%
Mean	1.24
Minimum	0
Maximum	21

Negative	0
Negative (%)	0.0%

Quantile statistics

Minimum	0
5-th percentile	1
Q1	1
Median	1
Q3	1
95-th percentile	3
Maximum	21
Range	21
Interquartile range (IQR)	0

Descriptive statistics

Standard deviation	1.030272518
Coefficient of variation (CV)	0.8308649339
Kurtosis	139.6868642
Mean	1.24
Median Absolute Deviation (MAD)	0
Skewness	8.746719491
Sum	1240
Variance	1.061461461

Monotonicity	Not monotonic

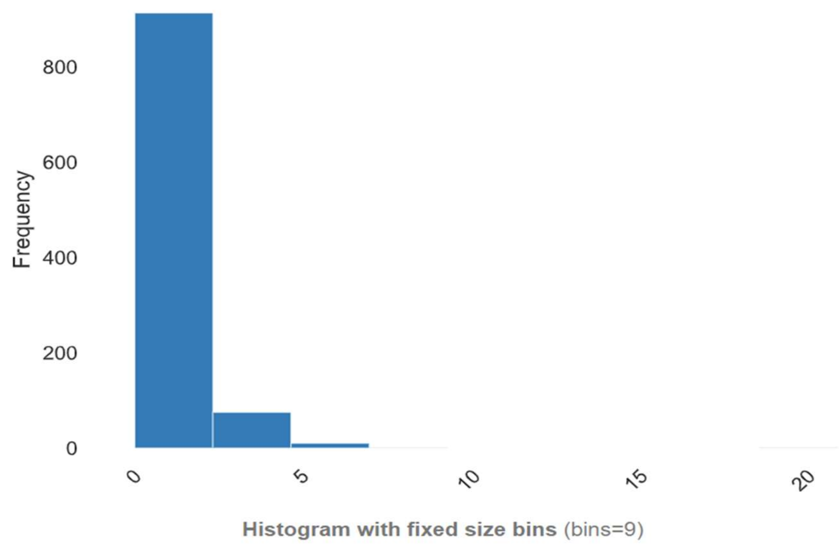

Figure 3.6

IP Reputation Histogram

- **IP Reputation details for 200 data points**:
  - IP Reputation is highly correlated with Threat score, Fail Ratio, Total Risk Count.
  - **IP Reputation has more positive Skewness 4.3738724,** as well as **Kurtosis 31.46794967** with 7 distinct values out of 200 observations.
  - We observed that **24 websites were not affected by IP Reputation issues**, 160 websites were affected by 1 issue, 7 websites were affected by 2 issues, 6 websites were affected by 3 issues, 1 website is affected by 4 issues, 1 website is affected by 5 issues, and 1 website is affected by 8 issues. **Out of 200 websites, 176 have at least 1 IP Reputation.**

Table 3.11

IP Reputation Data Details

IP Reputation Statistics	
Distinct	7
Distinct (%)	3.5%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	1.045
Minimum	0
Maximum	8
Negative	0
Negative (%)	0.0%
Quantile Statistics	
Minimum	0
5-th percentile	0
Q1	1
Median	1
Q3	1
95-th percentile	2
Maximum	8

Range	8
Interquartile range (IQR)	0
Descriptive Statistics	
Standard deviation	0.8038694111
Coefficient of variation (CV)	0.769253025
Kurtosis	31.46794967
Mean	1.045
Median Absolute Deviation (MAD)	0
Skewness	4.3738724
Sum	209
Variance	0.6462060302
Monotonicity	Not monotonic

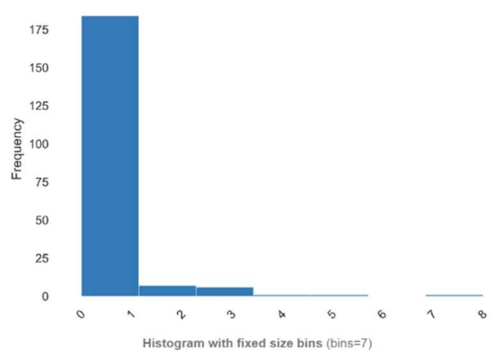

Figure 3.7

Reputation Frequency Distribution

- **Service Misconfiguration Details for 1000 data points**:

- Service misconfiguration is highly correlated with Threat Score, Fail Ratio, and Total Risk Counts.
- The data distribution of Service misconfiguration is **beta distribution**.
- We observed that **37 websites were not affected by Service Misconfiguration issues**, 1 website was affected by 2 issue, 4 websites were affected by 3 issues, 12 websites were affected by 4 issues, 19 websites were affected by 5 issues, 32 websites were affected by 6 issues, 35 websites were affected by 7 issues, 68 websites were affected by 8 issues, 101 websites were affected by 9 issues, 133 websites were affected by 10 issues, 182 websites were affected by 11 issues, 215 websites were affected by 12 issues, 108 websites were affected by 13 issues, 24 websites were affected by 14 issues, 24 websites were affected by 15 issues and 5 websites were affected by 16 issues. **Out of 1000 websites, 963 have at least 1 Service Misconfiguration**.
- We found the Service Misconfiguration has **negative Skewness of -1.461937818, positive Kurtosis value of 2.717981406** and is **non-monotonic.**

Table 3.12

Service Misconfiguration Details

Basic Data Statistics	Values
Distinct	16
Distinct (%)	1.6%
Missing	0
Missing (%)	0.0%
Infinite	0

Infinite (%)	0.0%
Mean	10.139
Minimum	0
Maximum	16
Negative	0
Negative (%)	0.0%

Quantile statistics

Minimum	0
5-th percentile	4
Q1	9
median	11
Q3	12
95-th percentile	14
Maximum	16
Range	16
Interquartile range (IQR)	3

Descriptive statistics

Standard deviation	3.038900116
Coefficient of variation (CV)	0.2997238501
Kurtosis	2.717981406
Mean	10.139

Median Absolute Deviation (MAD)	1
Skewness	-1.461937818
Sum	10139
Variance	9.234913914
Monotonicity	Not monotonic

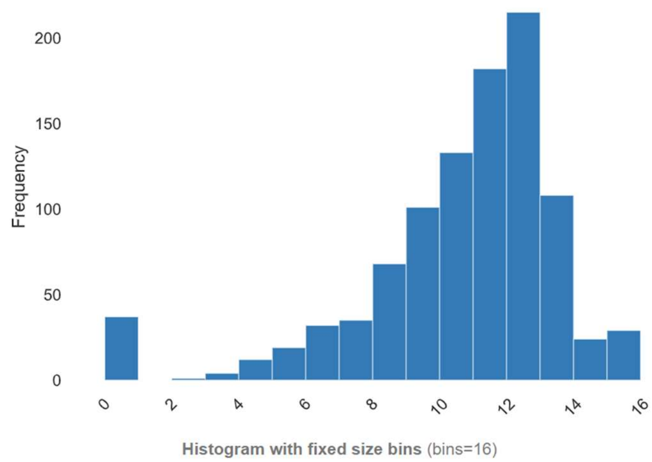

Figure 3.8

Service Misconfiguration and its Frequency Data Distribution Histogram

- **Service Misconfiguration Details for 200 data points**:
  - Service misconfiguration has high correlation with Fail ratio, SSL health, Outdated version, Total risk count.
  - We observed that **26 websites were not affected by any Service Misconfiguration issues**, 1 website is affected by 5 issues, 1 website is affected by 6 issues, 2 websites were affected by 7 issues, 10 websites were affected by 8 issues, 15 websites were affected by 9 issues, 15 websites were affected by 10 issues, 8 websites were affected by 11 issues, 26 websites were affected by 12 issues, 31 websites were affected by 13 issues, 31

websites were affected by 14 issues, 25 websites were affected by 15 issues, 2 websites were affected by 16 issues and 7 websites were affected by 17 issues. Out of 200 websites, **174 have at least 1 Service Misconfiguration issue.**

- We found the Service Misconfiguration has **negative Skewness of -1.302546528, Kurtosis value of 0.7248730251,** and is **non-monotonic.**

Table 3.13

Service Misconfiguration for 200 Data Points

Data Statistics	Values
Distinct	14
Distinct (%)	7.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	10.765
Minimum	0
Maximum	17
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0

5-th percentile	0
Q1	9
median	12
Q3	14
95-th percentile	15
Maximum	17
Range	17
Interquartile range (IQR)	5
Descriptive statistics	
Standard deviation	4.767959963
Coefficient of variation (CV)	0.442913141
Kurtosis	0.7248730251
Mean	10.765
Median Absolute Deviation (MAD)	2
Skewness	-1.302546528
Sum	2153
Variance	22.73344221
Monotonicity	Not monotonic

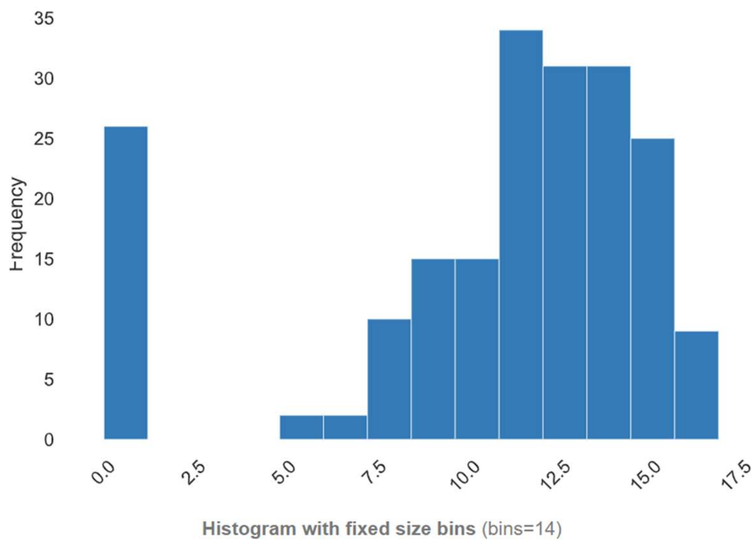

Figure 3.9

Service Misconfiguration Frequency Distribution

- **Outdated Version Details for 1000 Data points**:
  - The outdated version is highly correlated with threat score, fail ratio, and total risk counts.
  - The Outdated Version is **categorical data**.
  - We observed that **662 websites were not affected by any Outdated version issues** and 338 websites were affected by 1 issue. **Out of 1000 websites, 338 have at least 1 Outdated version.**
  - We found the outdated version has **positive Skewness of 0.6859774953, negative Kurtosis value of -1.532503894,** and is **non-monotonic.**

Table 3.14

Outdated Version

Basic Data Statistics	Values
Distinct	2
Distinct (%)	0.2%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.338
Minimum	0
Maximum	1
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	1
95-th percentile	1

Maximum	1
Range	1
Interquartile range (IQR)	1
Descriptive statistics	
Standard deviation	0.4732652322
Coefficient of variation (CV)	1.400192995
Kurtosis	-1.532503894
Mean	0.338
Median Absolute Deviation (MAD)	0
Skewness	0.6859774953
Sum	338
Variance	0.22397998
Monotonicity	Not monotonic

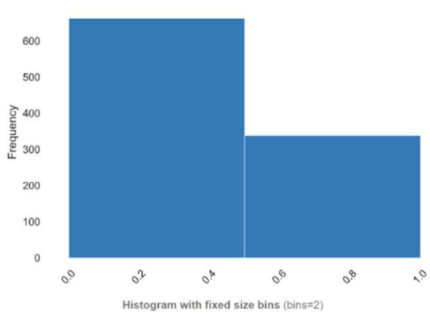

Figure 3.10

Outdated Version Histogram with Frequency of 2

Outdated Version Details for 200 Data points:

- Outdated Version is highly correlated with Threat score, Fail Ratio, Service Misconfiguration, Total Risk Count.
- Outdated Version is **categorical data**.
- We observed **104 websites were not affected by any Outdated Version issues**, 64 websites were affected by 1 issue and 32 websites were affected by 2 issues. **Out of 200 websites, 96 have at least 1 Outdated version.**

Table 3.15

Outdated Version Data Details

Data Statistics	Values
Distinct	3
Distinct (%)	1.5%
Missing	0
Missing (%)	0.0%
Memory size	1.7 KiB

Table 3.16

Common Values in the Outdated Version

Value	**Count**	**Frequency (%)**
0	104	52.0%
1	64	32.0%

2	32	16.0%

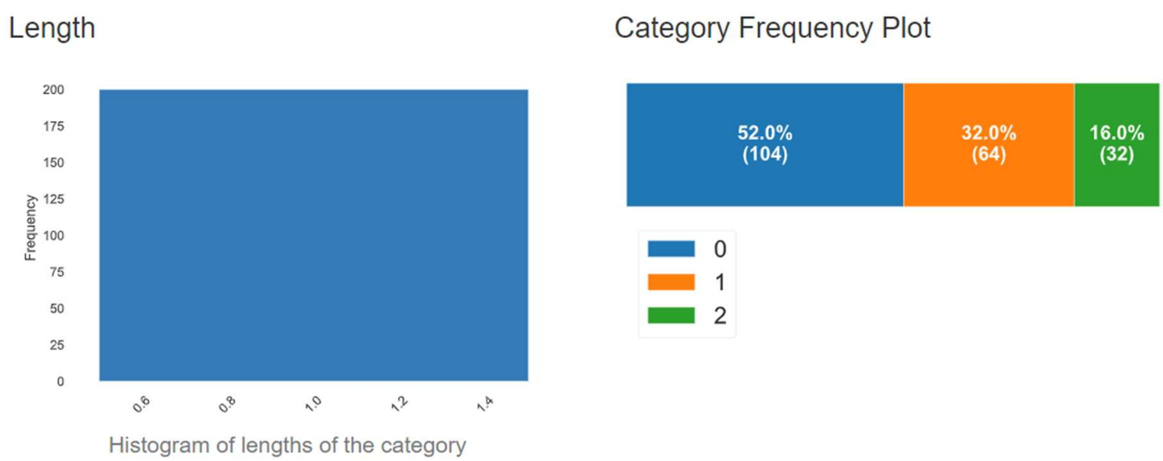

Figure 3.11

Category Frequency Plot

- **Data Leaks Details for 1000 Data points**:
    - Data leaks are highly correlated with Threat Score, Fail Ratio, and Total Risk Counts.
    - The Data leak is **categorical data**.
    - We observed **664 websites were not affected by Data Leaks**, 204 websites were affected by 1 leak and 132 websites were affected by 2.
    - We found the Data Leaks has **positive Skewness of 1.191968344, negative Kurtosis value of -0.04788201033** and is **non-monotonic.**

Table 3.17

Data Leaks Details

Data Leaks Statistics	Values
Distinct	3
Distinct (%)	0.3%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.468
Minimum	0
Maximum	2
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	1
95-th percentile	2
Maximum	2

Range	2
Interquartile range (IQR)	1
Descriptive statistics	
Standard deviation	0.7165818093
Coefficient of variation (CV)	1.531157712
Kurtosis	-0.04788201033
Mean	0.468
Median Absolute Deviation (MAD)	0
Skewness	1.191968344
Sum	468
Variance	0.5134894895
Monotonicity	Not monotonic

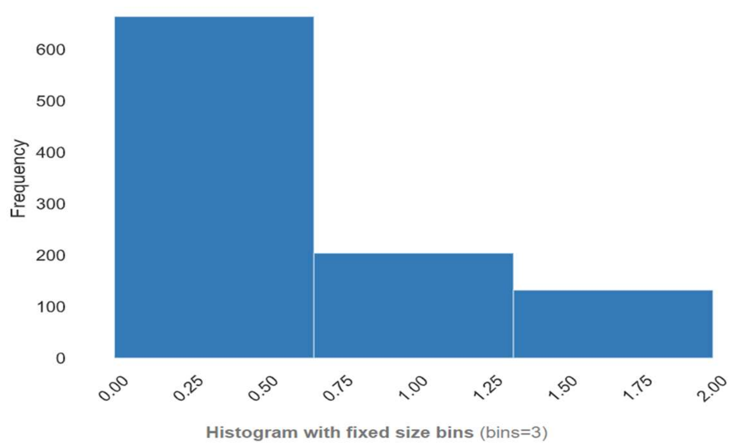

Figure 3.12

Data Leaks Frequency Data Distribution

- **Data Leaks Details for 200 Data points**:
  - Data Leaks are highly correlated with Threat Score.
  - We observed **87 websites were not affected by Data Leaks**, 83 websites were affected by 1 leak and 30 websites were affected by 2 leaks. **Out of 200 websites, 113 have at least 1 Data Leaks.**
  - Data Leaks is **categorical data**.

Table 3.18

Data Leaks

Data Statistics	Values
Distinct	3
Distinct (%)	1.5%
Missing	0
Missing (%)	0.0%
Memory size	1.7 KiB

Table 3.19

Common Values

Value	Count	Frequency (%)
0	87	43.5%
1	83	41.5%
2	30	15.0%

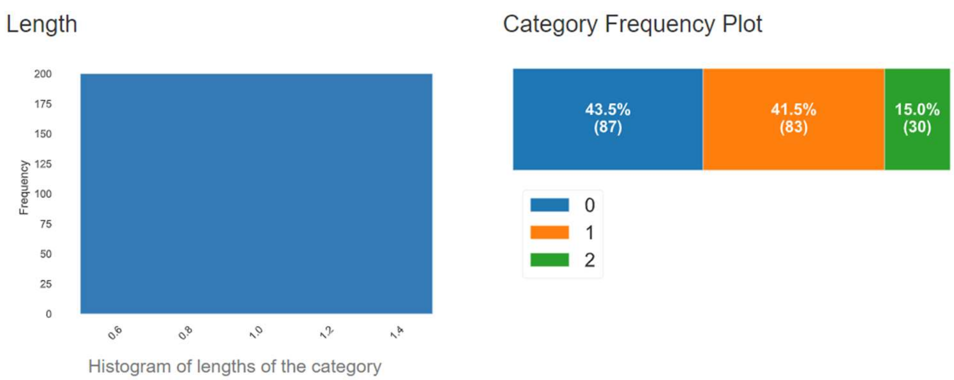

Figure 3.13

Data Leaks Category Frequency Plot

- **DNS Misconfiguration Details for 1000 data points**:
  - DNS Misconfiguration has zero attacks, so it has less impact on the other columns such as Threat Score, Fail Ratio, and Total Risk Count in the raw data set.
  - DNS Misconfiguration has zero values in the data set and **no Skewness and Kurtosis.**
- **DNS Misconfiguration Details for 200 data points**:
  - DNS Misconfiguration is highly correlated with data breaches.
  - We observed that **83 websites were not affected by DNS Misconfiguration**, 1 website is affected by 1 issue, 103 websites were affected by 2 issues, 12 websites were affected by 3 issues and 1 website is affected by 4 issues. **Out of 200 websites, 117 have at least 1 DNS Misconfiguration.**

Table 3.20

DNS Misconfiguration Statistics Details

Data Statistics	Values
Distinct	5
Distinct (%)	2.5%
Missing	0
Missing (%)	0.0%
Memory size	1.7 KiB

- DNS Misconfiguration has common values which is a categorical data.

Table 3.21

Common Values Frequency

Value	Count	Frequency (%)
2	103	51.5%
0	83	41.5%
3	12	6.0%
1	1	0.5%
4	1	0.5%

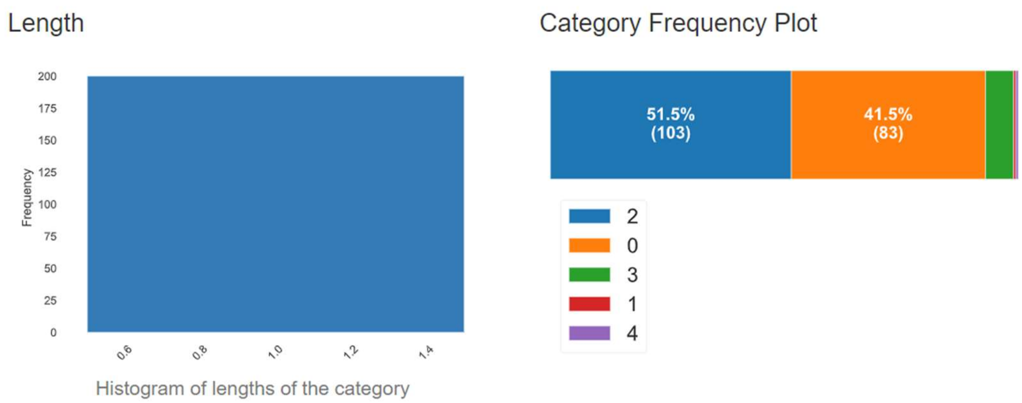

Figure 3.14

Category Frequency Plot

- **Data Breaches Details for 1000 data points**:
  - Data breaches are highly correlated with Threat Score, Fail Ratio, and Total Risk Counts.
  - Data breaches are **Categorical data**.
  - We observed that **998 websites were not affected by Data Breaches** and 2 websites were affected by 1 breach.
  - We found data breaches have **positive Skewness of 22.32704629, Kurtosis value of 497.4919779** and is **non-monotonic.**

Table 3.22

Data Breaching Data Statistics for 1000 Data Points

Data Breaches Statistics	Values
Distinct	2
Distinct (%)	0.2%

Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.002
Minimum	0
Maximum	1
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	0
95-th percentile	0
Maximum	1
Range	1
Interquartile range (IQR)	0
Descriptive statistics	
Standard deviation	0.04469897088
Coefficient of variation (CV)	22.34948544

Kurtosis	497.4919779
Mean	0.002
Median Absolute Deviation (MAD)	0
Skewness	22.32704629
Sum	2
Variance	0.001997997998
Monotonicity	Not monotonic

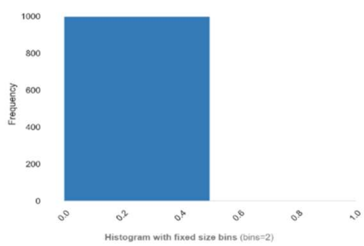

Figure 3.15

Data Breaches frequency

- **Data Breaches Details for 200 Data points**:
    - Data breaches have zero attacks, so it has less impact on the other columns such as Threat Score, Fail Ratio, and Total Risk Count in the raw data set.
    - Data breaches have zero values in the data set and **no Skewness and Kurtosis.**
- **Unnecessary Open Ports Details for 1000 data points**:
    - Unnecessary Open Ports are highly correlated with Threat Score, Fail Ratio, and Total Risk Counts.

- We observed that **714 websites were not affected by Unnecessary Open Ports**, 120 websites were affected by 1 issue, 42 websites were affected by 2 issues, 34 websites were affected by 3 issues, 60 websites were affected by 4 issues, 1 website is affected by 5 issues, and 29 websites were affected by 6 issues. **Out of 1000 websites, 286 have at least 1 Unnecessary Open Ports.**
- We found Unnecessary Open Ports have **Positive Skewness of 2.195941974, positive kurtosis value of 4.052796718** and is **non-monotonic.**

Table 3.23

Unnecessary Open Ports Details

Data Statistics	Values
Distinct	7
Distinct (%)	0.7%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.725
Minimum	0
Maximum	6
Negative	0
Negative (%)	0.0%
Quantile statistics	

Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	1
95-th percentile	4
Maximum	6
Range	6
Interquartile range (IQR)	1
Descriptive statistics	
Standard deviation	1.44895634
Coefficient of variation (CV)	1.998560469
Kurtosis	4.052796718
Mean	0.725
Median Absolute Deviation (MAD)	0
Skewness	2.195941974
Sum	725
Variance	2.099474474
Monotonicity	Not monotonic

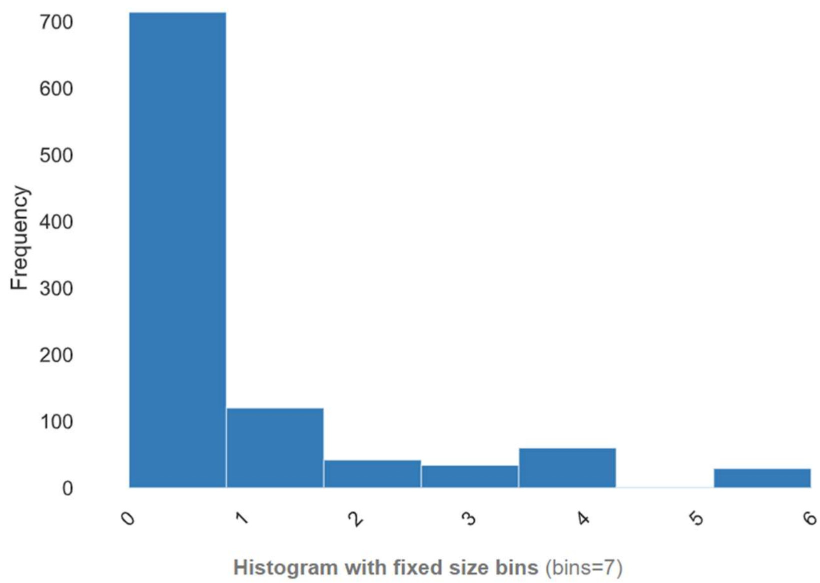

Figure 3.16

Unnecessary Open Ports Histogram

- **Unnecessary Open Ports Details for 200 data points**
  - Unnecessary Open Ports are highly correlated with Threat Score, Fail Ratio, and Total Risk Count.
  - We observed that **103 websites were not affected by Unnecessary Open Ports**, 45 websites were affected by 1 issue, 34 websites were affected by 2 issues, 14 websites were affected by 3 issues, 3 websites were affected by 5 issues and 1 website is affected by 6 issues, out of 2.
  - We found the Unnecessary Open Ports have **positive skewness of 1.519279276, kurtosis value of 2.729855323** and is **non-monotonic.**

Table 3.24

Unnecessary Open Port

Data Statistics	Values
Distinct	6
Distinct (%)	3.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.88
Minimum	0
Maximum	6
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	2
95-th percentile	3
Maximum	6

Range	6
Interquartile range (IQR)	2
Descriptive statistics	
Standard deviation	1.149874365
Coefficient of variation (CV)	1.306675415
Kurtosis	2.729855323
Mean	0.88
Median Absolute Deviation (MAD)	0
Skewness	1.519279276
Sum	176
Variance	1.322211055
Monotonicity	Not monotonic

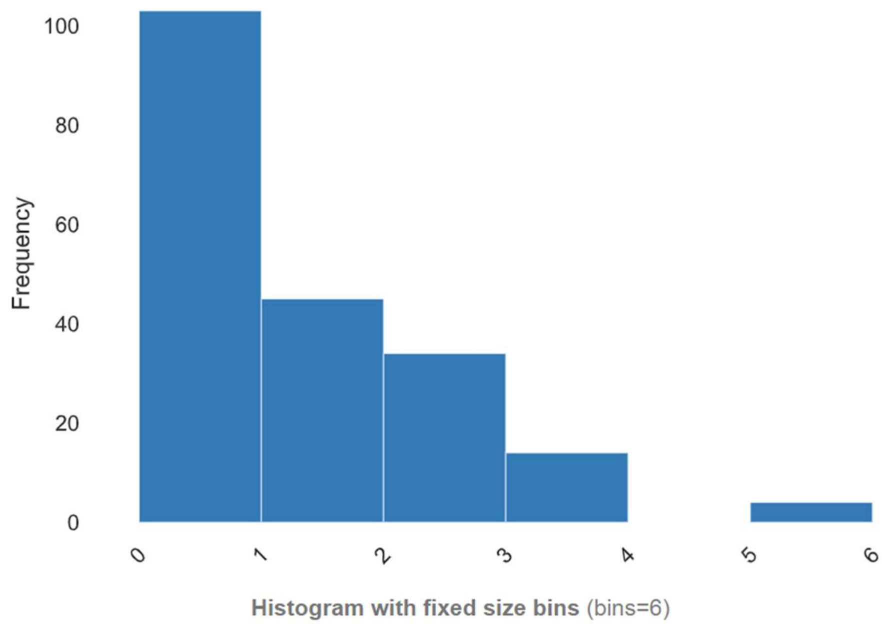

Figure 3.17

Unnecessary Open Port Frequency Data Distribution

- **Total Risk Count Details for 1000 data points**:
  - Total Risk count is highly correlated with Threat Score, Fail Ratio, SSL Health, Service Misconfiguration, and Outdated Version.
  - Data distribution of the Total Risk Count is the **normal distribution and Poisson distribution** as well.
  - We found Total Risk Count has **negative skewness of -0.6238191925, Kurtosis value of 1.719934859** and is **non-monotonic.**

Table 3.25

Total Risks Count Details

Data Statistics	Values
Distinct	30
Distinct (%)	3.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	15.174
Minimum	0
Maximum	40
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	7
Q1	13
Median	16
Q3	18
95-th percentile	22
Maximum	40

Range	40
Interquartile range (IQR)	5
Descriptive statistics	
Standard deviation	4.974582442
Coefficient of variation (CV)	0.3278359326
Kurtosis	1.719934859
Mean	15.174
Median Absolute Deviation (MAD)	3
Skewness	-.6238191925
Sum	15174
Variance	24.74647047
Monotonicity	Not monotonic

- **Total Risks Count Details for 200 data points**:
  - Total Risk Count is highly correlated with Threat Score, Fail Ratio, SSL Health, IP Reputation, Service Misconfiguration, Outdated version.
  - We found the Total Risk Count has **negative skewness of -1.217297117, Kurtosis value of 0.6955076327** and is **non-monotonic**

Table 3.26

Total Risk Count for 200 Data Statistics

Data Statistics	Values
Distinct	23
Distinct (%)	11.5%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	17.72
Minimum	0
Maximum	30
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	15
Median	20
Q3	23
95-th percentile	27
Maximum	30

Range	30
Interquartile range (IQR)	8
Descriptive statistics	
Standard deviation	7.644855728
Coefficient of variation (CV)	0.4314252668
Kurtosis	0.6955076327
Mean	17.72
Median Absolute Deviation (MAD)	3
Skewness	-1.217297117
Sum	3544
Variance	58.4438191
Monotonicity	Not monotonic

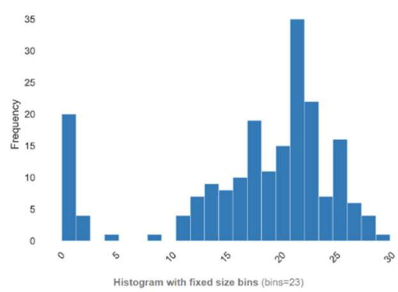

Figure 3.18

Total Risk Count for 200 Frequency Distribution

- **Domain name details**:
    - Domain names are categorical data.

Table 3.27

Domain Name Data Details

Data Statistics	Values
Distinct	1000
Distinct (%)	100.0%
Missing	0
Missing (%)	0.0%
Max length	28
Median length	24
Mean length	11.592
Min length	4
Unique	1000
Unique (%)	100.0%

- **Industry for 1000 data points**:
    - Industry is a categorical data and we found 11 different industries in dataset.

Table 3.28

Industry Names Data Details

Data Statistics	Values
Distinct	11
Distinct (%)	1.1%

Missing	0
Missing (%)	0.0%
Max length	22
Median length	20
Mean length	18.138
Min length	6
Unique	0
Unique (%)	0.0%

- **Industry for 200 data points:**
  - Industry is a categorical data and we found 13 different industries in dataset.

Table 3.29

Industry Data Details

Industry Data details	Values
Distinct	13
Distinct (%)	6.5%
Missing	0
Missing (%)	0.0%
Memory size	1.7 KiB
Max length	22
Median length	16

Mean length	9.1
Min length	2
Unique	1
Unique (%)	0.5%

Threats for 200 data points

Table 3.30

Data Statistics Variables for 200 Data Points and their Different Types

Threats Dataset statistics	Values
Number of variables	8
Number of observations	200
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	12.6 KiB
Average record size in memory	64.6 B
Variable Type	
Numeric	6
Categorical	2

Table 3.31

High Correlation between Threats for 200 Data Points

Data Statistics	Correlation
Phishing Threats are highly correlated with Data Leaks, Brand & Reputation Threats, Rogue Mobile Apps, and Total Threats	High correlation
Data Leaks are highly correlated with Phishing Threats, Brand & Reputation Threats, Rogue Mobile Apps, and Total Threats	High correlation
Brand & Reputation Threats are highly correlated with Phishing Threats, Data Leaks, Rogue Mobile Apps, and Total Threats	High correlation
Rogue Mobile Apps are highly correlated with Phishing Threats and Phishing Threats, Data Leaks, and Total Threats	High correlation
Total Threats are highly correlated with Phishing Threats, Data Leaks, Brand & Reputation Threats, and Rogue Mobile Apps	High correlation

- **Industry Details:**
  - Industry is a categorical data and has 13 distinct values.

Table 3.32

Industry Statistics Data Details

Industry Data Details	Values
Distinct	13
Distinct (%)	6.5%
Missing	0

Missing (%)	0.0%
Length	
Max length	22
Median length	16
Mean length	9.1
Min length	2
Characters and Unicode	
Total characters	1820
Distinct characters	38
Distinct categories	5
Distinct scripts	2
Distinct blocks	2
Unique	
Unique	1
Unique (%)	0.5%

- **Phishing Threats Details:**
  - Phishing threats statistical analysis is highly correlated with Data Leaks, Brand & Reputation Threats, Rogue Mobile Apps, and Total Threats.
  - We observed **153 websites were not affected by Phishing Threats**, 13 websites were affected by 1 threat, 12 websites were affected by 2 threats, 9 websites were affected by 3 threats, 4 websites were affected by 4 threats, 4 websites were affected by 5 threats, 1

website is affected by 6 threats, 1 website is affected by 7 threats, 2 websites were affected by 8 threats and 1 website is affected by 16 threats. **Out of 200 websites, 47 have at least 1 phishing threats.**

- We found Phishing Threats have **positive skewness of 4.317921488, Kurtosis value of 26.42416441** and is **non-monotonic.**

Table 3.33

Phishing Threats Details

Data Statistics	Values
Distinct	10
Distinct (%)	5.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.725
Minimum	0
Maximum	16
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0

Q1	0
Median	0
Q3	0
95-th percentile	4
Maximum	16
Range	16
Interquartile range (IQR)	0
Descriptive statistics	
Standard deviation	1.834722331
Coefficient of variation (CV)	2.53065149
Kurtosis	26.42416441
Mean	0.725
Median Absolute Deviation (MAD)	0
Skewness	4.317921488
Sum	145
Variance	3.36620603
Monotonicity	Not monotonic

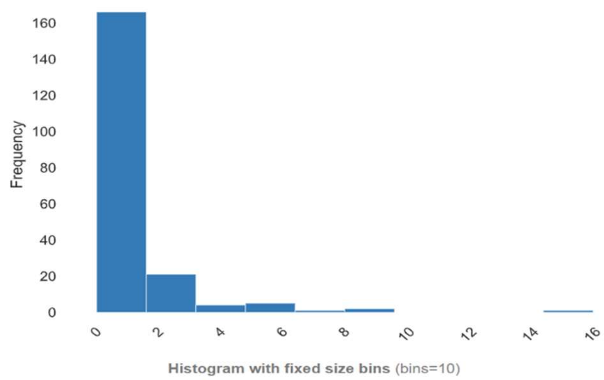

Figure 3.19

Phishing Threats Frequency Data Distribution

- **Brand & Reputation Threats:**
  - Brand & Reputation Threats are highly correlated with Phishing threats, Data leaks, Rogue Mobile Apps and Total Threats.
  - We observed that **157 websites were not affected by Brand and Reputation Threats**, 11 websites were affected by 1 threat, 14 websites were affected by 2 threats, 6 websites were affected by 3 threats, 3 websites were affected by 4 threats, 2 websites were affected by 5 threats, 3 websites were affected by 7 threats, 1 website is affected by 10 threats, 1 website is affected by 11 threats, 1 website is affected.
  - We found Brand & Reputation Threats have **positive skewness of 3.950850237, Kurtosis value of 17.54180886** and is **non-monotonic.**

Table 3.34

Brand & Reputation Threats

Data Statistics	Values
Distinct	11
Distinct (%)	5.5%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.73
Minimum	0
Maximum	13
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	0
95-th percentile	4
Maximum	13

Range	13
Interquartile range (IQR)	0
Descriptive statistics	
Standard deviation	1.996756163
Coefficient of variation (CV)	2.735282416
Kurtosis	17.54180886
Mean	0.73
Median Absolute Deviation (MAD)	0
Skewness	3.950850237
Sum	146
Variance	3.987035176
Monotonicity	Not monotonic

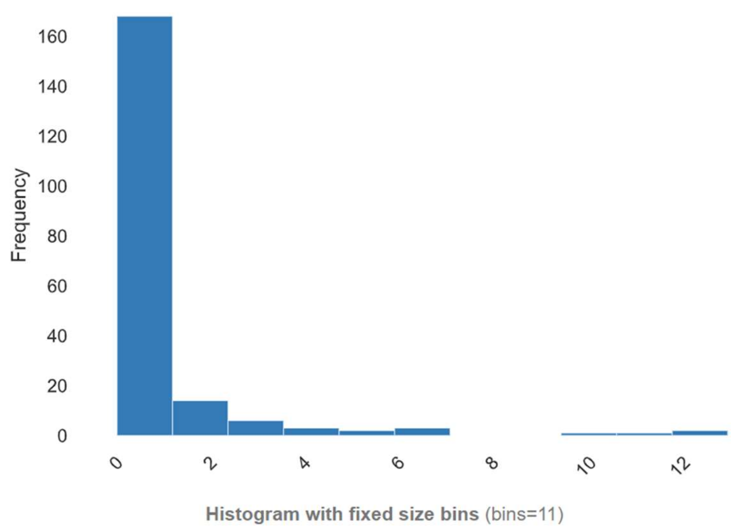

Figure 3.20

Brand & Reputation Threats Frequency Data Distribution

- **Rogue Mobile Apps Details:**
    - Rogue Mobile Apps details are highly correlated with Phishing threats, Data leaks, Brand & Reputation Threats, and Total Threats.
    - We observed that **149 apps were not affected by Rogue Mobile Apps,** 4 apps were affected by 1 threat, 8 apps were affected by 2 threats, 15 apps were affected by 3 threats, 12 apps were affected by 4 threats, 3 apps were affected by 5 threats, 2 apps were affected by 6 threats, 2 apps were affected by 7 threats, 1 app is affected by 10 threats, 1 app is affected by 12 threats, 1 app is affected by 25 threats, 1 app is affected by 26 threats and 1 app is affected by 36 threats. **Out of 200 apps, 51 apps have at least 1 Rogue Mobile threats**.
    - We found Rogue Mobile Apps have **positive skewness of 5.986265084, Kurtosis value of 42.89111044** and is **non-monotonic**

Table 3.35

Rogue Mobile Apps Details

Data Statistics	Values
Distinct	13
Distinct (%)	6.5%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	1.315

Minimum	0
Maximum	36
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	1
95-th percentile	5
Maximum	36
Range	36
Interquartile range (IQR)	1
Descriptive statistics	
Standard deviation	3.948891454
Coefficient of variation (CV)	3.002959281
Kurtosis	42.89111044
Mean	1.315
Median Absolute Deviation (MAD)	0
Skewness	5.986265084
Sum	263

Variance	15.59374372
Monotonicity	Not monotonic

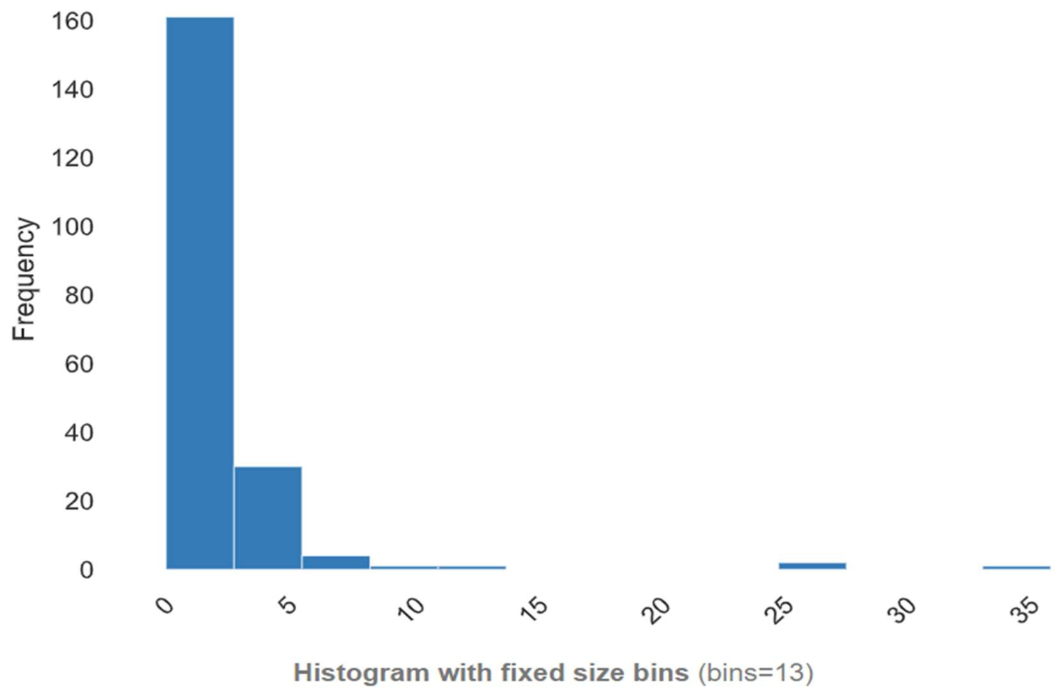

Figure 3.21

Frequency Data Distribution

- **Data Leaks**
    - Data Leaks are highly correlated with Threat Score, Fail Ratio, and Total Risk Count.
    - We observed **144 websites were not affected by Data Leaks**, 16 websites were affected by 1 leak, 12 websites were affected by 2 leaks, 6 websites were affected by 3 leaks, 11 websites were affected by 4 leaks, 3 websites were affected by 5 leaks, 3 websites were affected by 6 leaks, 2 websites were affected by 7 leaks, 2 websites were affected by 8 leaks and 1 website is affected by 11 leaks. **Out of 200 websites, 56 have at least 1 Data Leaks**.

- Data leaks have **kurtosis value of 7.32008017 and positive skewness of 2.584391458.**

Table 3.36

Data Leaks Statistics for 200 Data Points

Data Leak Details	Values
Distinct	10
Distinct (%)	5.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.88
Minimum	0
Maximum	11
Negative	0
Negative (%)	0.0%
Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
Median	0
Q3	1
95-th percentile	5

Maximum	11
Range	11
Interquartile range (IQR)	1
Descriptive statistics	
Standard deviation	1.833688103
Coefficient of variation (CV)	2.083736481
Kurtosis	7.32008017
Mean	0.88
Median Absolute Deviation (MAD)	0
Skewness	2.584391458
Sum	176
Variance	3.36241206
Monotonicity	Not monotonic

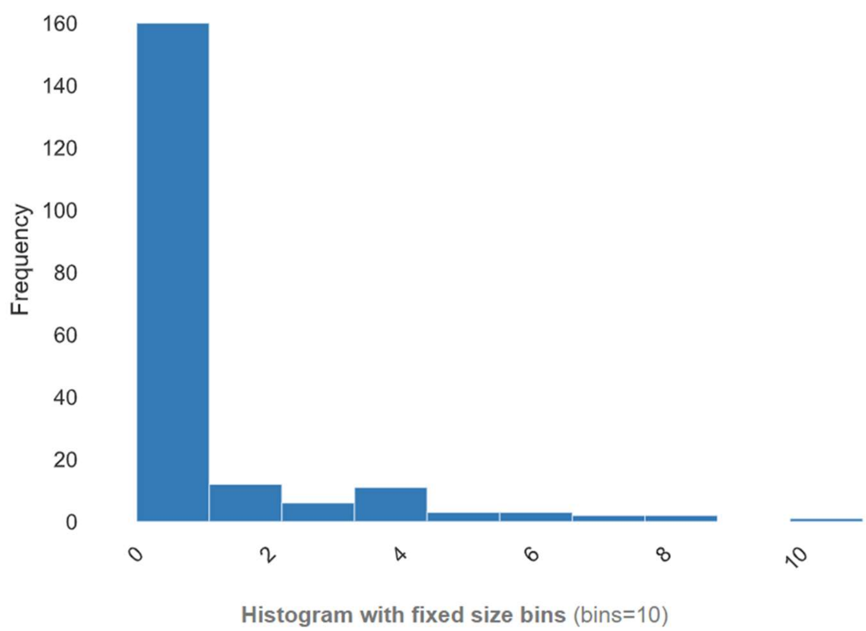

Figure 3.22

Data Leaks Frequency Data Distribution

- **Data Breaches:**
  - Data Breaches are **Categorical data**.
  - We observed **196 websites were not affected by Data Breaches** and **out of 200 websites, 4 have at least 1 Data Breaches.**

Table 3.37

Data Breaches Details

Data Details	Values
Distinct	2
Distinct (%)	1.0%
Missing	0

Missing (%)	0.0%

- **Total Threats:**
  - Total Threats is a real number which has a high correlation with Phishing Threats, Data leaks, Brand & Reputation Threats.
  - We found Total Threats have **positive skewness of 4.185440516, Kurtosis value of 25.46701965** and is **non-monotonic.**

Table 3.38

Total Threats Data Details

Data details	Values
Distinct	25
Distinct (%)	12.5%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	3.67
Minimum	0
Maximum	68
Negative	0
Negative (%)	0.0%

Quantile statistics	
Minimum	0
5-th percentile	0
Q1	0
median	0
Q3	5
95-th percentile	17.05
Maximum	68
Range	68
Interquartile range (IQR)	5
Descriptive statistics	
Standard deviation	7.889379669
Coefficient of variation (CV)	2.149694733
Kurtosis	25.46701965
Mean	3.67
Median Absolute Deviation (MAD)	0
Skewness	4.185440516
Sum	734
Variance	62.24231156
Monotonicity	Not monotonic

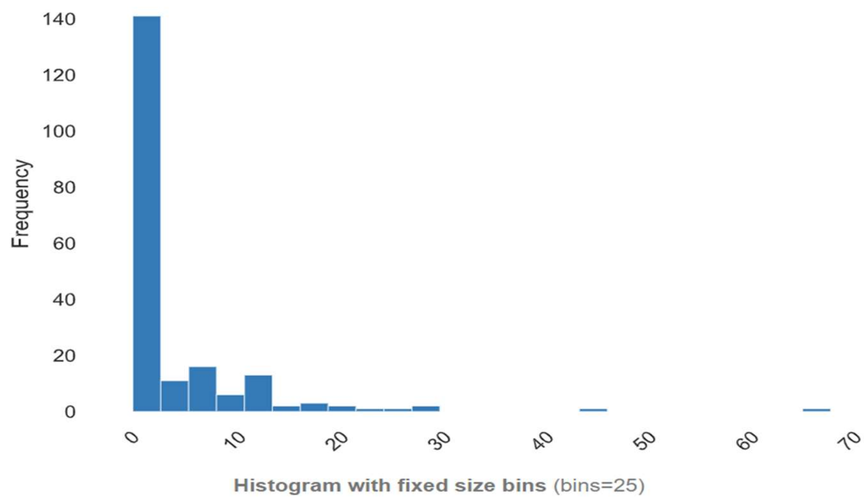

Figure 3.23

Total Threats Frequency Data Distribution

3.23 Initial Data Analysis

Overview of the data from Alexa's website.

Quality of data:

- We were able to do a better analysis because before starting the initial analysis we checked if the quality of the data met our business problem or not.
- Several types of quality data were found in 1000 data points:
  - Frequency counts
    - Range Index: 1000 entries, 0 to 999
    - Data columns (total 15 columns), 12 columns are integer data types, and 2 columns are object data types.
    - Data types: int64(13), object(2)

Table 3.39

Frequency Count of the Type and Non-Null

S.No	Column	Non-Null Count	Data Type
1	Threat Score	1000 non-nulls	int64
2	Fail Ratio	1000 non-nulls	int64
3	SSL Health	1000 non-nulls	int64
4	IP Reputation	1000 non-nulls	int64
5	Service Misconfiguration	1000 non-null	int64
6	Outdated Version	1000 non-null	int64
7	Data Leaks	1000 non-null	int64
8	DNS Misconfiguration	1000 non-null	int64
9	Data Breaches	1000 non-null	int64
10	Unnecessary Open Ports	1000 non-null	int64
11	Total Risks Count	1000 non-null	int64
12	Domain name	1000 non-null	object
13	Industry code	1000 non-null	int64
14	Industry name	1000 non-null	object

Table 3.40

Descriptive Statistics for 1000 Data Points

S. No	Feature	Skewness	Kurtosis	Mean	Min	Max	Mode	Count	Median
1	Threat Score	-0.02987	2.144286	75.82583	0	252	70	1000	77.0

2	Fail Ratio	-0.62188	1.717691	16.99299	0	45	19	1000	18.0
3	SSL Health	-0.2381	-0.29859	2.259259	0	5	3	1000	2.0
4	IP Reputation	8.734249	139.3651	1.238238	0	21	1	1000	1.0
5	Service Misconfiguration	-1.46658	2.702567	10.12813	0	16	12	1000	11.0
6	Outdated Version	0.684378	-1.5347	0.338338	0	1	0	1000	0.0
7	Data Leaks	1.197923	-0.03697	0.466466	0	2	0	1000	0.0
8	DNS Misconfiguration	0	-3	0	0	0	0	1000	0.0
9	Data Breaches	22.31585	496.992	0.002002	0	1	0	1000	0.0
10	Unnecessary Open Ports	2.203838	4.088505	0.722723	0	6	0	1000	0.0
11	Total Risks Count	-0.63216	1.707586	15.15516	0	40	17	1000	16.0

Inference:

- Based on above table 1.39, our data is normally distributed. 68.2% of the data lies within one standard deviation of the mean, and 95% lies within two standard deviations.
- **Outdated Version (0.684378), DNS Misconfiguration (0), Threat Score (-0.02987), Fail Ratio (-0.62188), SSL Health (-0.2381)**, **Total Risks Count (-0.63216)** are **normally distributed**.
- **Positive Skewness:**
    - **Data Leaks (1.197923) have moderate right skewness.**
    - **Data Breaches (22.31585), Unnecessary Open Ports (2.203838), IP Reputation (8.734249) have severe right Skewness.**
- **Negative Skewness:**
    - **Service Misconfiguration (-1.46658) has moderate left skewness.**

Kurtosis:

- Threat score, fail Ratio, IP Reputation, Service Misconfiguration, Data Breaches, Unnecessary Open Ports, and Total Risks Count have **positive Kurtosis.**
- SSL Health, Outdated Version, Data Leaks, Data Breaches, and DNS Misconfiguration have **negative Kurtosis.**

Table 3.41

Descriptive Statistics for 200 Data Points

Data Statistics	Count	Skewness	Kurtosis	Mean	Std	Min	Max
Threat score	200.0	-0.029874	2.144286	94.94	45.6	-1.0	234.0
Fail ratio	200.0	-0.621879	1.717691	19.88	8.59	0.0	34.0
SSL Health	200.0	-0.238099	-0.298593	2.44	1.53	0.0	6.0
IP Reputation	200.0	8.734249	139.365105	1.04	0.80	0.0	8.0
Service Misconfiguration	200.0	-1.466581	2.702567	10.76	4.76	0.0	17.0
Outdated Version	200.0	0.684378	-1.534703	0.64	0.74	0.0	2.0
Data Leaks	200.0	1.197923	-0.036966	0.71	0.71	0.0	2.0
DNS Misconfiguration	200.0	0.000000	0.000000	1.23	1.07	0.0	4.0
Data Breaches	200.0	22.315846	496.991970	0.00	0.00	0.0	0.0
Unnecessary Open Ports	200.0	2.203838	4.088505	0.88	1.14	0.0	6.0
Total Risk Count	200.0	-0.632161	1.707586	17.72	7.64	0.0	30.0

Inference:

- **Outdated Version (0.684378), DNS Misconfiguration (0), Threat score (-0.0298), SSL Health (-0.23480), Fail Ratio (-0.62187) and Total Risk Count (-0.63216)** are **normally distributed**.
- **Positive Skewness:**
    - **Data Leaks (1.197923) have moderate right skewness.**
    - **Data Breaches (22.31585), Unnecessary Open Ports (2.203838) and IP Reputation (8.734249) have severe right skewness.**
- **Negative Skewness:**
    - **Service Misconfiguration (-1.46658) has moderate left skewness.**

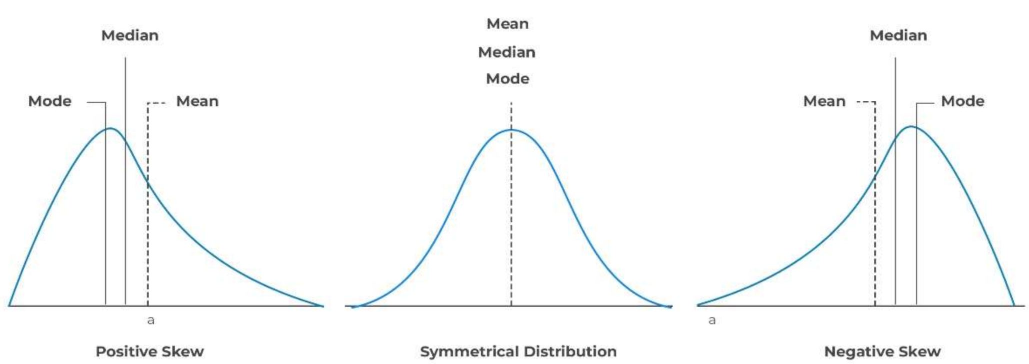

Figure 3.24

Kurtosis

Kurtosis:

- Threat score, fail Ratio, IP Reputation, Service Misconfiguration, Data Breaches, Unnecessary Open Ports and Total Risks Count have **positive Kurtosis.**

- SSL Health, Outdated Version, Data Leaks, Data Breaches, and DNS Misconfiguration have **negative Kurtosis.**

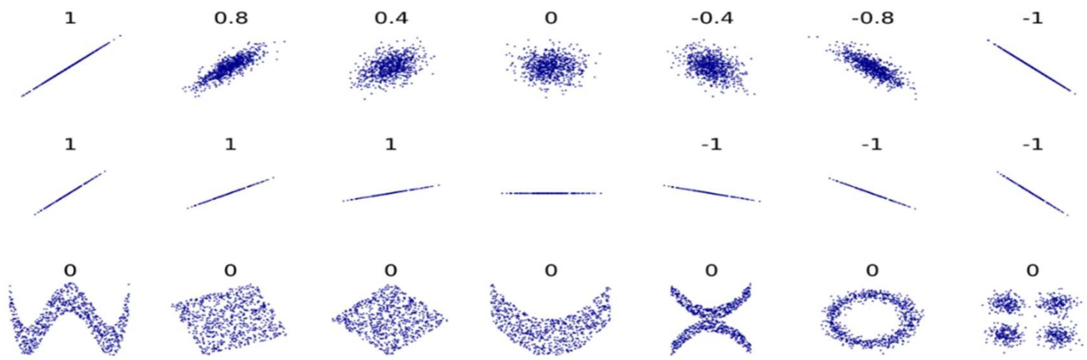

Figure 3.25

Correlation Analysis

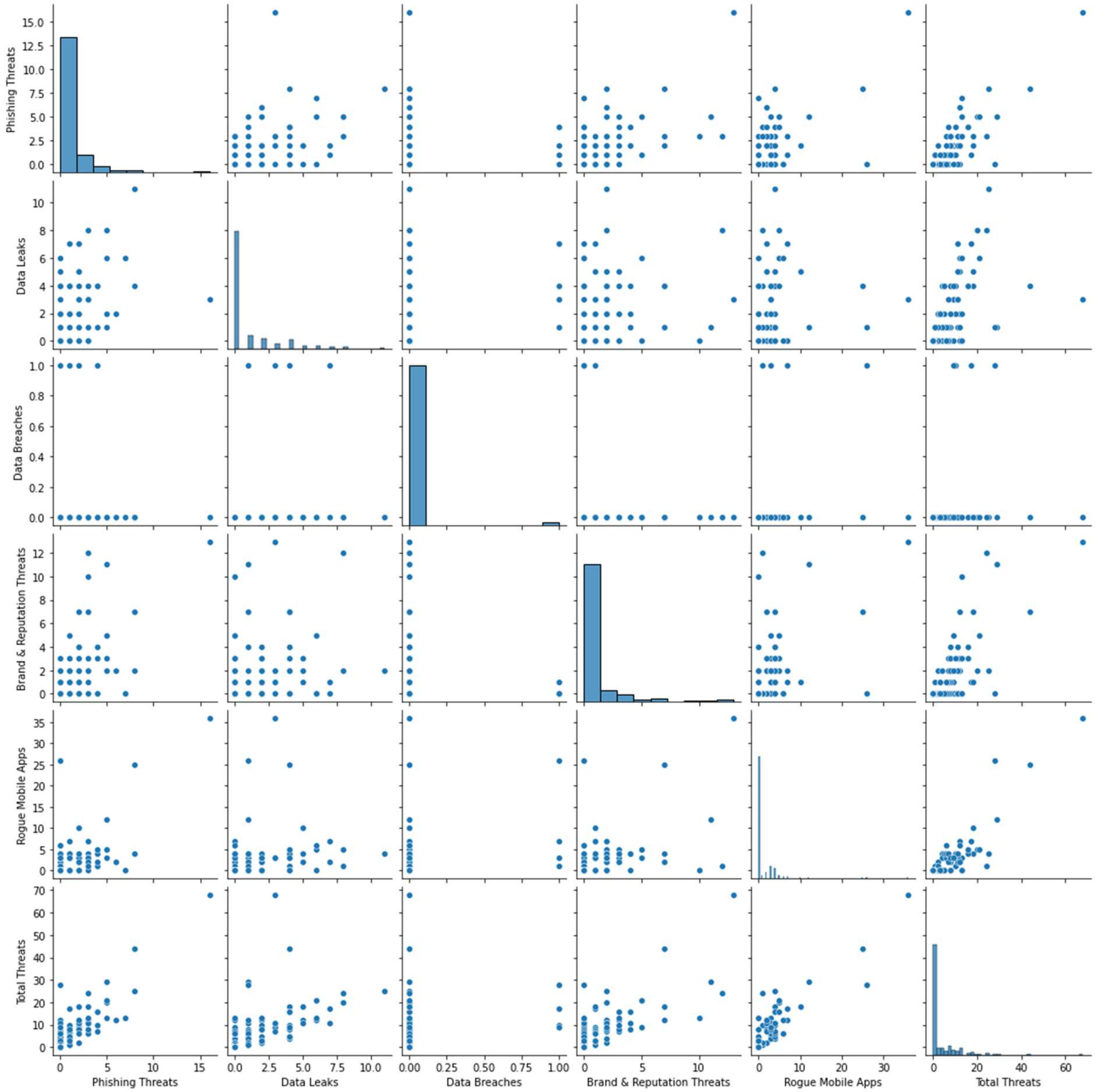

Figure 3.26

200 Data Points Correlation Relationship

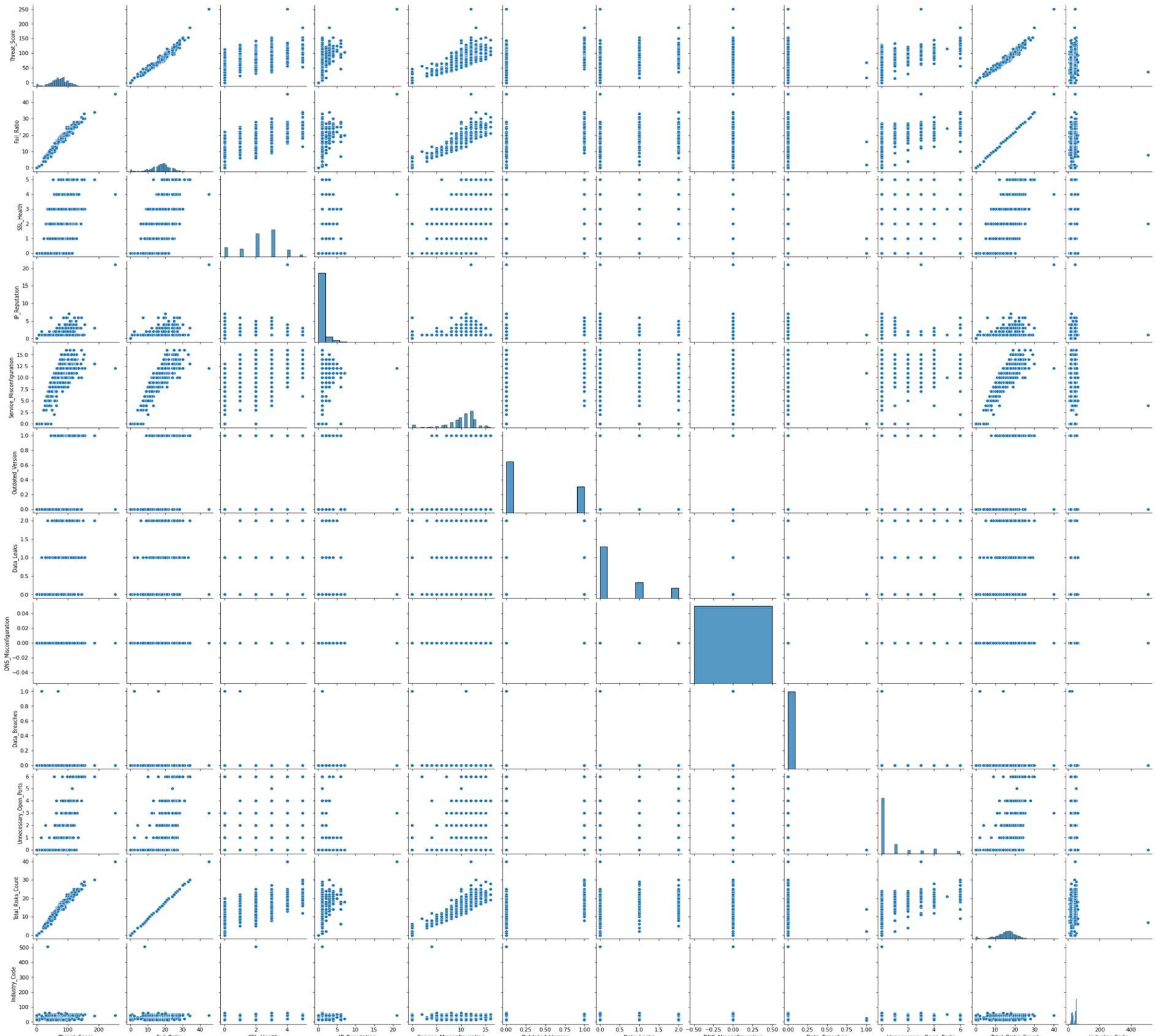

Figure 3.27

Alexa 1000 Data Point Correlation Relationship

Linear Correlation

Table 3.42

Correlation Coefficient between Attack Vectors, Threat Score, Fail Ratio and Total Risks Count

Feature Name	Threat Score	Fail Ratio	SSL Health	IP Reputation	Service Misconfi-guration	Outdated Version	Data Leaks	Data Breaches	Unnecessary Open Ports	Total Risks Count
Threat Score	1(Strong +ve	0.99(Strong +ve	0.59(Partially Strong	0.41(Partially Strong	0.88(Moderate Strong	0.50(Partially Strong	0.44(+ve Week Positive	-0.05(-ve Negatively	0.55(Partially Strong	0.98(Strong +ve

	Correlation)	Correlation)	+ve Correlation)	+ve Correlation)	+ve Correlation)	+ve Correlation)	Correlation)	Correlation)	+ve Correlation)	Correlation)
Fail Ratio	0.98(Strong +ve Correlation)	1(Strong +ve Correlation)	0.60(Partially Strong +ve Correlation)	0.33(Partially Strong +ve Correlation)	0.88(Strong +ve Correlation)	0.48(Partially Strong +ve Correlation)	0.37(+ve Week Positive Correlation)	-0.06(Negatively Correlation)	0.49(Partially Strong +ve Correlation)	0.99(Strong +ve Correlation)
SSL Health	0.52(Strong +ve Correlation)	0.60(Partially Strong +ve Correlation)	1(Strong +ve Correlation)	0.02(Zero Correlation)	0.43(Partially Strong +ve Correlation)	0.2 (+ve Week Positive Correlation)	0.15(+ve Week Positive Correlation)	-0.06(Negatively Correlation)	0.16(+ve Week Positive Correlation)	0.60(Partially Strong +ve Correlation)
IP Reputation	0.40(+ve Positive Correlation)	0.33(+ve Week Positive Correlation)	0.02(Zero Correlation)	1(Strong +ve Correlation)	0.16(+ve Week Positive Correlation)	0.062(Zero Correlation)	-0.035(-ve Negatively Correlation)	-0.01(-ve Negatively Correlation)	0.08(Zero Correlation)	0.34(+ve Week Positive Correlation)
Service Misconfiguration	0.81(Moderate Positive Correlation)	0.88(Moderate Strong +ve Correlation)	0.43(+ve Week Positive Correlation)	0.16(+ve Week Positive Correlation)	1(Strong +ve Correlation)	0.41(+ve Week Positive Correlation)	0.26(+ve Week Positive Correlation)	-0.06(-ve Negatively Correlation)	0.18(+ve Week Positive Correlation)	0.88(Moderate Strong +ve Correlation)
Outdated Version	0.50(+ve Week Positive Correlation)	0.48(+ve Week Positive Correlation)	0.20(+ve Week Positive Correlation)	0.06(Zero Correlation)	0.41(+ve Week Positive Correlation)	1(Strong +ve Correlation)	0.16(+ve Week Positive Correlation)	-0.03(-ve Negatively Correlation)	0.17(+ve Week Positive Correlation)	0.48(+ve Week Positive Correlation)
Data Leaks	0.44(+ve Week Positive Correlation)	0.37(+ve Week Positive Correlation)	0.15(+ve Week Positive Correlation)	-0.03(-ve Negatively Correlation)	0.26(+ve Week Positive Correlation)	0.16(+ve Week Positive Correlation)	1(Strong +ve Correlation)	-0.02(-ve Negatively Correlation)	0.09(Zero Correlation)	0.37(+ve Week Positive Correlation)
Data Breaches	-0.054(Zero Correlation)	-0.06(-ve Negatively Correlation)	-0.064(-ve Negatively Correlation)	-0.01(-ve Negatively Correlation)	-0.067(-ve Negatively Correlation)	-0.032(-ve Negatively Correlation)	-0.02(-ve Negatively Correlation)	1(Strong +ve Correlation)	-0.02(-ve Negatively Correlation)	-0.06(-ve Negatively Correlation)
Unnecessary Open Ports	0.55(Partially Strong +ve Correlation)	0.49(Partially Strong +ve Correlation)	0.16(+ve Week Positive Correlation)	0.083(Zero Correlation)	0.18(-ve Negatively Correlation)	0.17(+ve Week Positive Correlation)	0.091(Zero Correlation)	-0.022(Zero Correlation)	1(Strong +ve Correlation)	0.49(+ve Week Positive Correlation)
Total Risks Count	0.98(Strong +ve Correlation)	0.99(Strong +ve Correlation)	0.60(+ve Positive Correlation)	0.34(+ve Week Positive Correlation)	0.88(Moderate Strong +ve Correlation)	0.48(+ve Week Positive Correlation)	0.37 (+ve Week Positive Correlation)	-0.06(Zero Correlation)	0.49 (+ve Week Positive Correlation)	1(Strong +ve Correlation)

Table 3.43

Relationship Between Attack Vectors, Threat score, Fail Ratio and Total Risks Count

Feature Name	Relationship Summary
Threat Score	Strong correlation with Fail Ratio, Service Misconfiguration, Total Risk Counts.
Fail Ratio	Strong correlation with Threat score, Service Misconfiguration, Total Risk Counts.
SSL Health	Mild correlation with Fail Ratio, Total Risks Count, Threat score.
IP Reputation	No strong or mild Correlation
Service Misconfiguration	Strong correlation with Fail Ratio, Total Risks Count, Threat score.
Outdated Version	Mild correlation with Fail Ratio, Total Risks Count, Threat score.
Data Leaks	No strong or mild Correlation
Data Breaches	No correlation
Unnecessary Open Ports	Mild correlation with Fail Ratio, Total Risks Count, Threat score.
Total Risks Count	Strong correlation with Fail Ratio, Service Misconfiguration, Threat score.

Spearman's ‘ρ’

- Spearman correlation was used in this analysis to evaluate relationships involving ordinal variables and to identify if two variables relate in a monotonic function.

- The cluster numbers (0,1,2,3…...) are assigned in the data sampling approach based on spearman's rank correlation.
- Using spearman correlation coefficient, we got the linear relationship between each column as shown in the table 1.42
- We observed outliers in **Threat Score** from 0 to 10 and >140 to 250 data points.
- We observed outliers in **Fail ratio** from 0 to 8 and 28 to 50 data points.
- We observed outliers in **SSL Health** from 0 and 5 data points.
- We observed outliers in **IP Reputation** from 4 and 20 data points.
- We observed outliers in **Service misconfiguration** from 0 to 4 data points.
- We observed outliers in **Data Breaches** at 1 data point.
- We observed outliers in **Unnecessary Open Ports** from 3 to 6.
- We observed outliers in **Total Risk Count** 0 to 5 and 25 to 40.
- We found **DNS Misconfiguration has an invalid coefficient** (zero correlation).
- We arrived at inference by calculating 'ρ' for two variables 'x' and 'y'. One divides the **covariance** of the rank variables of 'x' and 'y' by the product of their standard deviations.
- Threat score, Total Risk Counts and Fail Ratio are the highest-ranked variables with a correlation coefficient of nearly 100%.
- Threat score, Total Risk Counts, Fail Ratio, and Service Misconfiguration are strongly correlated with each other.

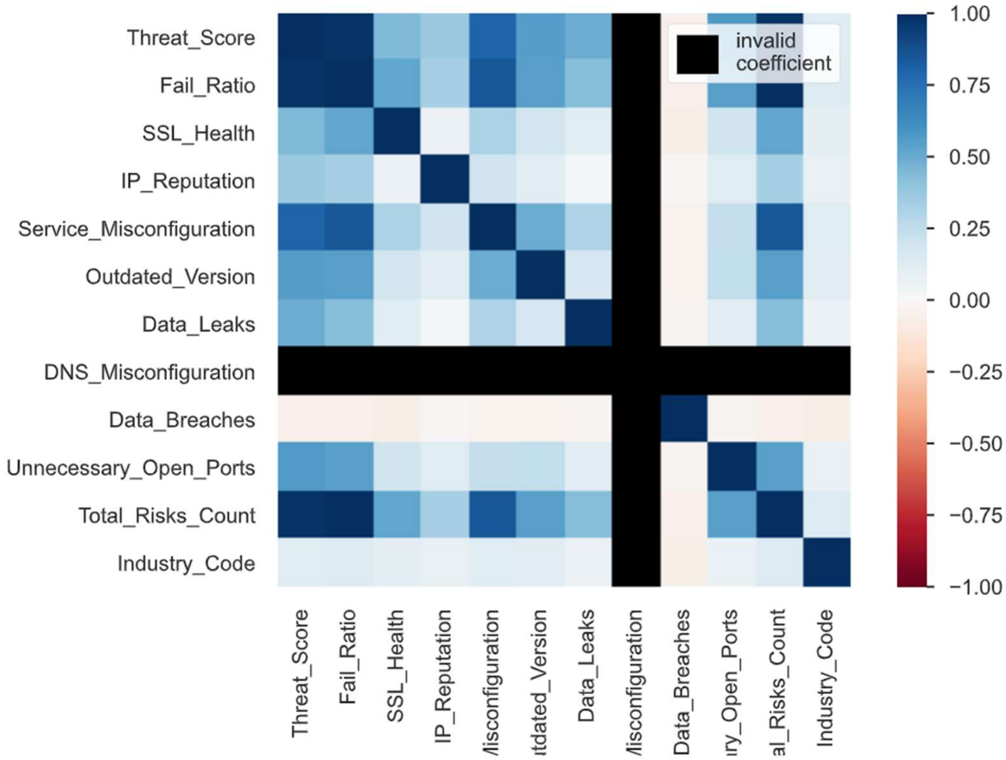

Figure 3.28

Spearman's Correlation

Initial transformations

- Log-transformation: It is used when when the distribution differs from normal.

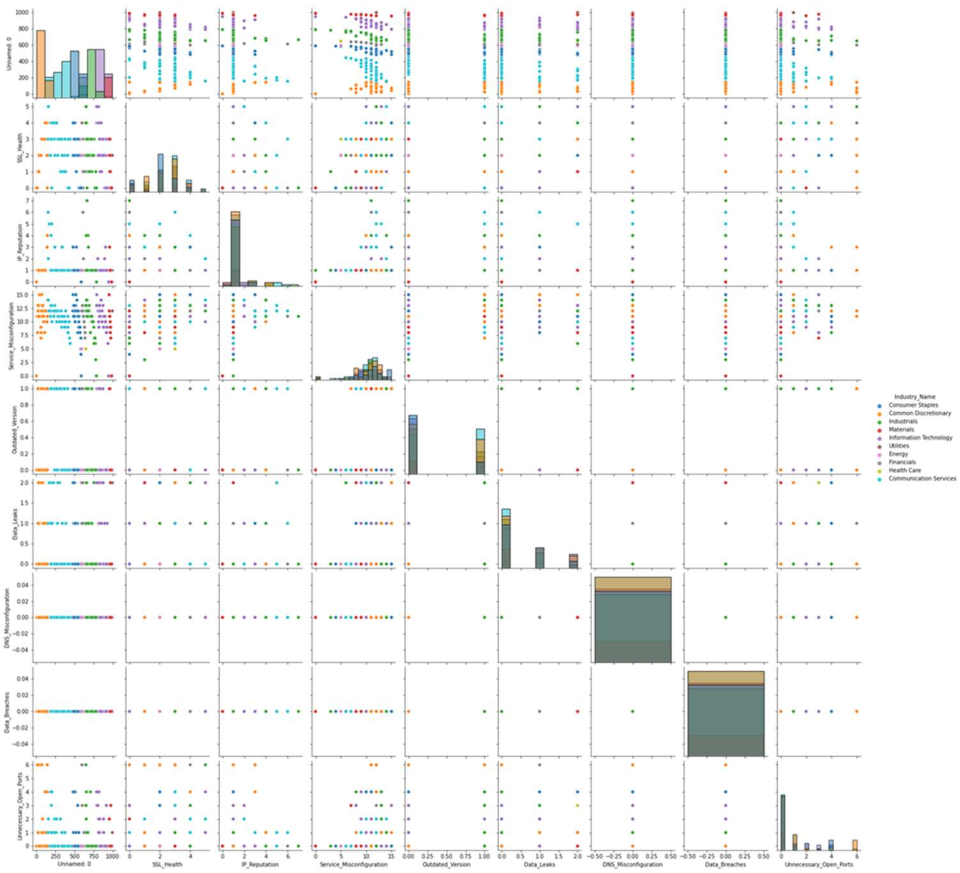

Figure 3.29

Data Transformation with Distribution

- The log transformation is the most popular among the diverse types of transformations becaused it is used to transform skewed data to conform to normality.
- If the original data follows a log-normal distribution or so, then the log-transformed data follows a normal or near-normal distribution.
- Example: A model is non-linear, but it can be transformed to a linear model such as logY=β0+β1t. We can take logarithms of ‘y’ to meet the specified model form.

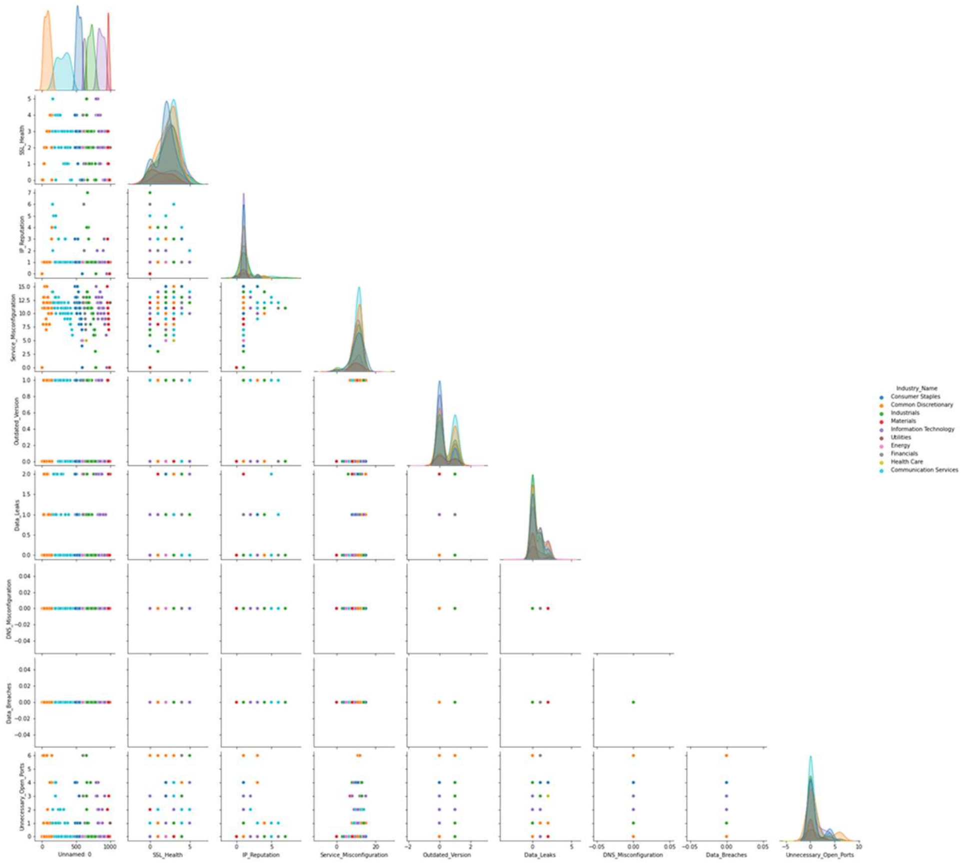

Figure 3.30

1000 Data Points KDE

3.3 Research Purpose and Questions

Below are the research questions that will help me derive answers by conducting the analysis of thousands of data points across different threat vectors and industries in line with the guidelines for the prevention and remediation of attacks –

1. What are the common attack vectors of external cyber-attacks?
2. How are the threat vectors distributed across different industry domains?
3. What's the topmost industry that got affected by the external attack vectors?
4. What's the most afflictive external attack vector?
5. What's the least afflictive external attack vector?
6. What's the monetary impact of these attacks on different vectors?
7. What's the priority matrix for implementing proactive controls for external attack vectors (with less effort for maximum risk reduction)?
8. What's the priority matrix for implementing remediation (with less effort for maximum risk reduction)?
9. What are the easy-to-implement guidelines for preventing attacks from external threat vectors?
10. What are the easy-to-implement guidelines for remediating attacks from external threat vectors?
11. What is the frequency of monitoring required for each external attack vector?
12. What are the different patterns that can be identified from the analysis? (by platform, by industry, by external attack vectors, by threats, by threat landscape)

3.4 Data Analysis

3.4.1 Data Sampling Approaches

- The data of Alexa's 1000 websites are exceedingly small and it has a different industry that acts as an imbalance class, so we need to choose the appropriate data sampling technique to solve imbalanced class data.

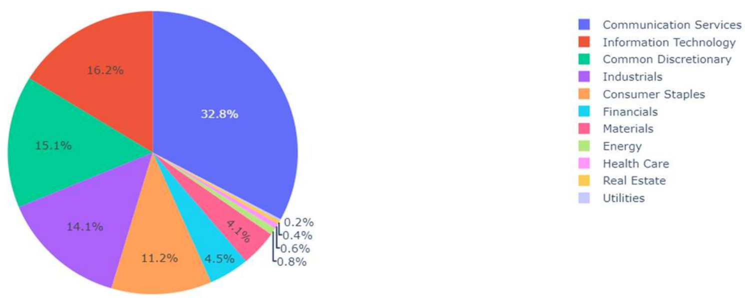

Figure 3.31

Distribution of Data based on Industries.

- Majority of data samples are from the Communication Services industry which are composed of **328 websites**.
- Minority of data samples are from the Real Estate and Utility industry which are composed of **4 and 2 websites** respectively.
- To balance the imbalanced dataset, we used permutation and combination without repetition of data sampling technique **(PCWORODS)**.

3.4.2 Data Assumption.

Table 3.44

Degree of Imbalance in Raw Data based on Industries

Feature Name	Proportion of Features
Communication Services	32.8 % of the Data set is occupied by communication services(majority)
Information Technology	16.2 % of the Data set is occupied by information technology(majority)
Real Estate	0.4 % of the Data set is occupied by real estate(minority)
Utilities	0.2% of the data set is occupied by utilities (minority)

- Data of Alexa's 1000 websites is small, and we have a different industry that creates an imbalance in the dataset, thus, we have chosen PCWORODS technique to solve the imbalance.
- We made two assumptions based on the transformed dataset:
    - Our **first assumption** is, 0.2 % and 0.4% of the data present in the transformed data are from the Utility and Real Estate industry. And they are in minority in the industry class.

$$\sum_i^n \text{minor}\big(len\ (n)\big) \qquad \text{--- equ-1}$$

- Our **second assumption** is, the combination of data(d) is created for each attack vector (av) based on the industry(in).

  $\sum_{in}^{av}(dav^1 + dav^2 + \ldots\ldots + dav^n)$ + in --- equ -2

- Our approach towards above assumptions
  - We used permutation that didn't appear in the previous iterations and made sure to consider each value from different attack vectors and industries.
  - At last, we selected data that was not repeated in the previous iteration.

3.4.3 Advantages

- Data sampling helped us to calculate the different combinations of data for every iteration.
- Data sampling helped us to achieve balanced data by industry, attack vector, and domain names.
- There was no repetition of data because data samples were selected based on different permutations.
- We didn't miss out on any different combinations of data.

3.4.4 PCWORODS approach for Alexa's 1000 websites.

- Since Alexa's 1000 websites have imbalanced data, by using this approach we can balance both majority and minority classes.
- Thus, to make a balanced dataset we used **Permutation and combination without repetition of the data sampling approach.**

3.4.5 Algorithm Steps

- For 1000 data records, we have taken up sampling classes as sampling methods to handle the data.
    - Input is 1000 data records, and the target class is Industry name, Domain, and Attack Vectors.
    - Output will be a balanced dataset based on our target.

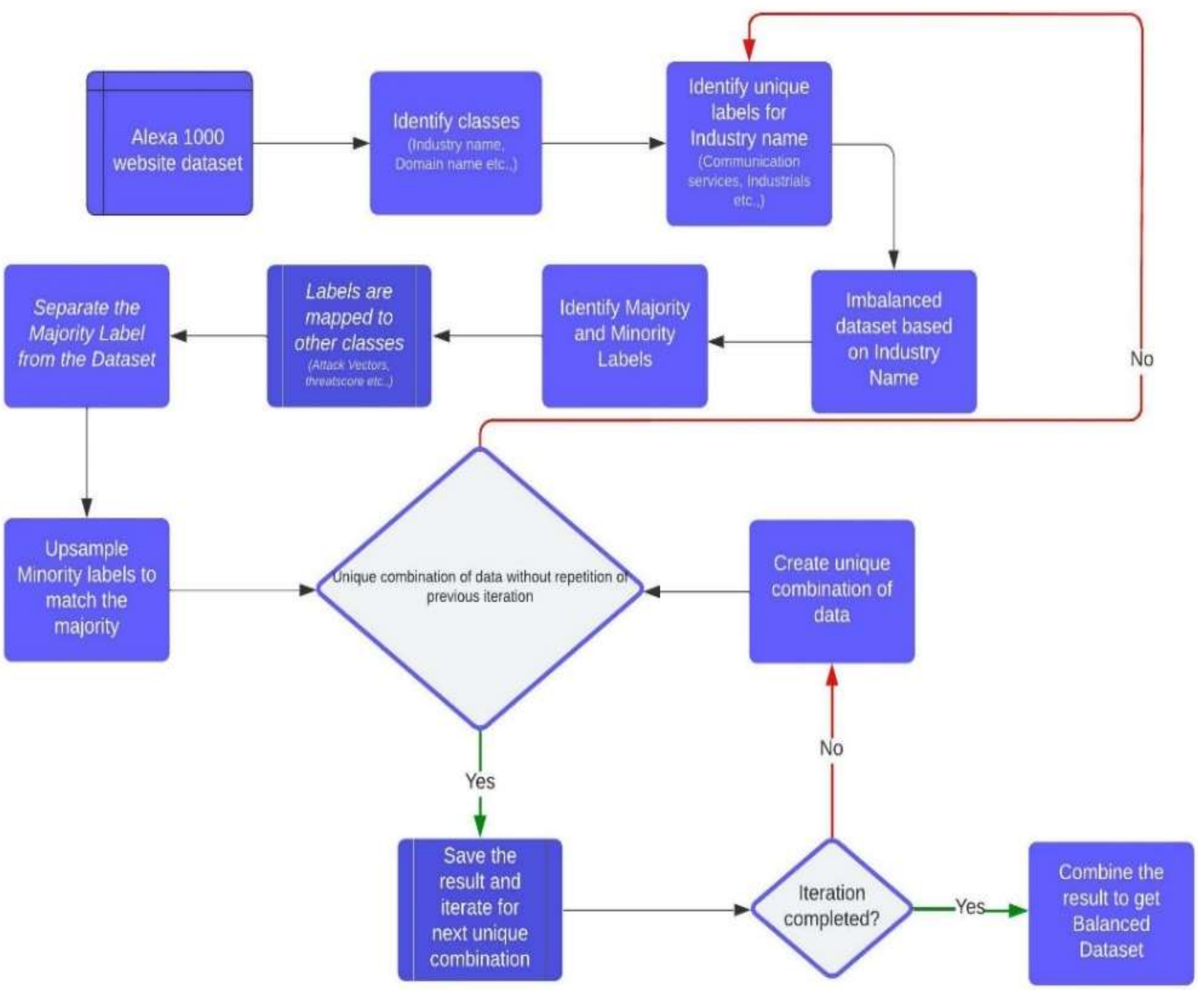

Figure 3.32

Flow Chart of Data Sampling Approach

- After iteration we sampled the class (Industry name) as a balanced dataset.
    - A combination of data is created in each iteration (based on the attack vector). In each iteration, only a unique combination of data is taken and any repeated combination from the previous iteration is ignored. Once all iteration is completed the result is obtained by combining data obtained from all iteration.

Steps to form combination of data.

- In each case, check the possibility and combination of every attack vector and take data samples. **Example:** We considered SSL Health 1 time attack, similarly we considered all other attack vectors as well ('SSL Health', 'IP Reputation’, ‘Service Misconfiguration', ‘Outdated Version', 'Data Leaks’, ‘DNS Misconfiguration', ‘Data Breaches', 'Unnecessary Open Ports')
- We checked the majority of data is from which domain and made the balance in each domain and data sample.
- At last, we combined the majority class with the other classeses. This is how we obtained an equal length of data for each class or balanced set. (Industry, top-level domain, in fact in each attack vector too)

3.4.6 Inference Result

- In the data sample process, we brought the data into normality with the help of different combinations and permutations of data samples as shown below.

Iteration-1 Data Transformation

Table 3.45

Data Transformation by using Data Sampling Approach

Threat_Score	Fail_Ratio	SSL_Health	IP_Reputation	Service Misconfiguration	Outdated Version	Data_Leaks	DNS_Misconfiguration	Data_Breaches	Unnecessary_Open_Ports	Total_Risks_Count	Domain_name	Industry_Code	Industry_Name	Cluster
69	17	3	1	10	1	0	0	0	0	15	xiumi.us	45	Information Technology	2
103	24	3	1	13	1	0	0	0	3	21	nvidia.com	45	Information Technology	2
0	0	0	0	0	0	0	0	0	0	0	argos.co.uk	15	Materials	2
106	22	3	1	11	0	1	0	0	4	20	mercari.com	30	Consumer Staples	2
82	19	1	1	14	0	1	0	0	0	17	youdao.com	45	Information Technology	2

Iteration-2 Data Sample

Table 3.46

Data Transformation by using Data Sampling Approach

ThreatScore	Fail_Ratio	SSL_Health	IP_Reputation	Service_Misconfiguration	Outdated_Version	Data_Leaks	DNS_Misconfiguration	Data_Breaches	Unnecessary_Open_Ports	Total_Risks_Count	Domain_name	Industry_Code	Industry_Name	Cluster
85	18	0	1	12	1	0	0	0	2	16	zippyshare.com	15	Materials	0
56	13	0	1	11	0	0	0	0	0	12	doorblog.jp	50	Communication Services	0
84	18	1	1	12	1	1	0	0	0	16	abc.net.au	50	Communication Services	0
64	16	4	1	9	0	0	0	0	0	14	livejournal.com	50	Communication Services	0
72	16	2	1	10	0	1	0	0	0	14	savefrom.net	45	Information Technology	0

Iteration-3 Data Sample

Table 3.47

Data Transformation by using Data Sampling Approach

Threat_Score	Fail_Ratio	SSL_Health	IP_Reputation	Service_Misconfiguration	Outdated_Version	Data_Leaks	DNS_Misconfiguration	Data_Breaches	Unnecessary_Open_Ports	Total_Risks_Count	Domain_name	Industry_Code	Industry_Name	Cluster
63	16	3	1	9	0	0	0	0	1	14	office365.com	25	Common Discretionary	0
97	20	1	3	12	1	0	0	0	1	18	torob.com	20	Industrials	0
45	11	3	1	6	0	0	0	0	0	10	unblockit.blue	45	Information Technology	0
86	19	2	1	12	1	1	0	0	0	17	kayak.com	20	Industrials	0
77	18	2	1	12	1	0	0	0	0	16	blackboard.com	45	Information Technology	0

- By iterating 3 times we got different cluster numbers which were used to identify and validate the result obtained from different permutations and combinations of the dataset.
- We arrived at inference by **eliminating outliers from the dataset**, and thus, achieved data **normality** or a **balanced dataset.**

Using a consistent data sampling method across multiple industries and datasets can greatly facilitate the development of machine learning models capable of forecasting future trends and attack vectors. Thus, I have used a sampling approach only for accurately understanding the scope of prediction and creating related machine-learning models. It's important to note that this research only focuses on creating the framework, rather than developing specific machine-learning models for predicting trends. With this in mind, I am ending this chapter and leaving the implementation of these models to future research.

3.5 Tools used

Sumeru's Threat Meter

Holding the fort from the external attack surface from all threat attack vectors is a huge task. Without having a tool that can assess in all directions, it's impossible to arrive at a framework;

hence the tool I have been a co-creator of, along with my team is leveraged here for further research to arrive at a framework. Sumeru's Threat Meter helps in the following ways –

External Attack Surface Monitoring:

Sumeru's Threat Meter monitors organizations' external attack surface and offers them critical assets and risk coverage.

Assets Discovery Coverage –

- Domains (TLDs)
- Subdomains
- IPs, Cloud Servers
- Open Ports
- Publicly exposed employees' email addresses
- Mobile Apps
- Social Media Profiles, etc.

Risk Coverage –

It runs 100+ test cases under the following categories to uncover risks:

- SSL Misconfigurations
- IP Reputation
- DNS Misconfiguration

- Public Credential Leaks
- Website Reputation
- Service Misconfiguration
- Unnecessary Open Ports
- Outdated version

Brand and Reputation Monitoring:

Threat Meter protects the brand and helps the organization from the fallout of reputation damage.

Threat Coverage –

- Unofficial Social Media Profiles
- Impersonating Domains
- Impersonating Mobile Apps
- VIP Profile Monitoring

Data Leak Monitoring in Dark & Deep Web:

Threat Meter gives all the visibility that the organization needs to detect sensitive data exposed over the darknet by employees, contractors, or third parties in 100+ dark web & internet sources.

Threat Coverage –

- Source Code

- Employee Emails & Credentials
- API/DB Credentials
- Sensitive Corporate Data & Files
- Intellectual Properties
- Data Breaches
- Customer Data, etc.

Phishing Threat Detection:

Most phishing solutions in the market work at the perimeter level and will only help organizations detect any incoming phishing emails. They do not protect the end customer from phishing attacks targeted towards them in the name of the organization.

Threat Coverage –

Threat Meter solves this challenge by acting at the initial phase of the phishing attack by detecting the following threats and protecting the end customer:

- Possible Typo squatting Domains
- Registered Typo squatting Domains
- Phishing Pages & Domains
- Phishing Email Servers

Rogue Mobile Apps Detection:

Threat Meter helps to discover fraudulent mobile apps that are leveraging customer brands to infect end users or steal credentials.

Threat Coverage –

- Unofficial & Untrusted Apps
- Rogue Apps
- Repackaged Apps

3.6 Conclusion

Our statistical analysis, including Quantile and Descriptive statistics, has enabled us to gain a deeper understanding of the data's Skewness and Kurtosis. We have also employed data balance techniques to ensure that the data is normalized across different industry types and attack vectors, allowing us to answer our research questions with greater accuracy and confidence.
Our analysis has involved extracting meaningful information from raw data for both 200 and 1000 data points and presenting these insights in a visualization report. This report provides decision-makers with a clear overview of industry-wise trends and common attack vectors, enabling them to make informed decisions about website security measures. Through the use of data balance techniques, we have ensured that our analysis is unbiased and accurate, reflecting the true nature of the data. Overall, our analysis highlights the importance of leveraging statistical techniques and data balance techniques to extract valuable insights from large datasets and make well-informed decisions based on those insights.

CHAPTER IV

RESULTS

4.1 Final Data Analysis

Most and least common attack vectors and threats of external cyber-attacks:

For 1000 Data points (Attack Vectors):

Table 4

Min, Max, and Average of Unique Occurrences of each Attack Vector and Average of Total Findings for each Attack Vector.

S. No	Attack Vector	Minimum of Unique occurrences	Maximum of Unique occurrences	Average of Unique occurrences	Average of Findings
1	Service Misconfiguration	0.0	16.0	8.43	10.13
2	SSL Health	0.0	5.0	2.50	2.26
3	IP Reputation	0.0	21.0	5.44	1.24
4	Unnecessary Open Ports	0.0	6.0	3.00	0.72
5	Data Leaks	0.0	2.0	1.00	0.46
6	Outdated Version	0.0	1.0	0.50	0.33
7	Data Breaches	0.0	1.0	0.50	0.002

8	DNS Misconfiguration	0.0	0.0	0.00	0.00

The total number of findings: 15174

Table 4.1

Total No of Findings for each Attack Vector

S. No	Attack Vector	Total No of Findings	Percent	Unique Occurrences of Attack Vectors
1	Service Misconfiguration	10139	66.82%	963
2	SSL Health	2262	14.91%	878
3	IP Reputation	1240	8.17%	975
4	Unnecessary Open Ports	725	4.78%	286
5	Data Leaks	468	3.08%	336
6	Outdated Version	338	2.23%	338
7	Data Breaches	2	0.01%	2
8	DNS Misconfiguration	0	0.0%	0

The most common attack vectors found in Alexa's top 1000 websites are:

- **66.82%** were Service Misconfiguration.

- **14.91%** were SSL Health.
- **8.17%** were IP Reputation.

The **least common Attack Vectors** found are **DNS Misconfiguration and Data Breaches.**

For 200 Data Points (Attack Vectors):

Table 4.2

Min, Max, and Average of Unique Occurrences of each Attack Vector and Average of Total Findings for each Attack Vector.

S. No	Attack Vector	Minimum of Unique occurrences	Maximum of Unique occurrences	Average of Unique occurrences	Average of Findings
1	Service Misconfiguration	0	17	10.21	10.76
2	SSL Health	0	6	3.00	2.44
3	DNS Misconfiguration	0	4	2.00	1.23
4	IP Reputation	0	8	3.28	1.04
5	Unnecessary Open Ports	0	6	2.83	0.88
6	Data Leaks	0	2	1.00	0.71
7	Outdated Version	0	2	1.00	0.64
8	Data Breaches	0	0	0.00	0.00

The total number of findings: 3544

Table 4.3

Total No of Findings for each Attack Vector

S. No	Attack Vector	Total No of Findings	Percentage	Unique Occurrences of Attack Vectors
1	Service Misconfiguration	2153	60.75%	174
2	SSL Health	488	13.77%	161
3	DNS Misconfiguration	247	6.97%	117
4	IP Reputation	209	5.9%	176
5	Unnecessary Open Ports	176	4.97%	97
6	Data Leaks	143	4.03%	113
7	Outdated Version	128	3.61%	96
8	Data Breaches	0	0.0%	0

Most common attack vectors found in 200 websites are:

- **60.75%** were Service Misconfiguration
- **13.77%** were SSL Health
- **6.97%** were DNS Misconfiguration

The least common attack vector found in 200 websites is **Data Breaches.**

Common inference for both 1000 and 200 Datapoints:

- **Service Misconfiguration** and **SSL Health** are the **most common attack vector**s found in both datasets.
- **Data Breaches** are the **least common attack vector** found in both datasets.

For 200 Data Points (Threats Vectors):

Table 4.4

Min, Max, and Average of Unique Occurrences of each Attack Vector and Average of Total Findings for each Attack Vector

S. No	Threats	Minimum of Unique Occurrences	Maximum of Unique Occurrences	Average of Unique Occurrences	Average of Findings
1	Rogue Mobile Apps	0.0	36.0	10.538462	1.315
2	Data Leaks	0.0	11.0	4.700000	0.880
3	Brand & Reputation Threats	0.0	13.0	6.181818	0.730
4	Phishing Threats	0.0	16.0	5.200000	0.725
5	Data Breaches	0.0	1.0	0.500000	0.020

The total number of threats found: 734

Table 4.5

Total No of Threats found for each Threat Vector

S. No	Threats	Total No of Threats	Percentage	Unique Occurrences of Atack Vectors
1	Rogue Mobile Apps	263	35.83%	51
2	Data Leaks	176	23.98%	56
3	Brand & Reputation Threats	146	19.89%	43
4	Phishing Threats	145	19.75%	47
5	Data Breaches	4	0.54%	4

The most common threat vectors found in 200 websites are:

- **35.83%** were Rogue Mobile Apps
- **23.98%** were Data Leaks
- **19.89%** were Brand & Reputation Threats

The least common Threat vector found in 200 websites is **Data Breaches.**

Most afflictive and least afflictive external attack vector: For 1000 Data Points (Attack Vectors)

No. of unique attack vectors: 56

Table 4.6

Top 10 Attack Vectors Based on Severity (Weight)

S. No	Attack Vector	Attack Landscape	No of Occurrences	Weight	Total Weight
1	Content-Security-Policy	Service Misconfiguration	15572	5.8	90317.6
2	X-XSS-Protection	Service Misconfiguration	14174	6.1	86461.4
3	Cookie Attribute - HTTP Only	Service Misconfiguration	12766	6.1	77872.6
4	Cross-Origin Resource Sharing	Service Misconfiguration	16377	4.6	75334.2
5	Strict-Transport-Security	Service Misconfiguration	12556	5.1	64035.6
6	X-Frame-Options	Service Misconfiguration	13361	4.5	60124.5
7	Employee Credentials Available in Breached Site Data	Data Leaks	5749	10.0	57490.0
8	Referrer-Policy	Service Misconfiguration	15948	3.4	54223.2

9	Missing Pragma	Service Misconfiguration	14944	3.6	53798.4
10	X-Content-Type-Options	Service Misconfiguration	13561	3.8	51531.8

No of Occurrences – Denotes count of each attack vector across 1000 websites.

Weight – Denotes severity of attack vector.

Total Weight – Denotes the sum of severity of each attack vector across 1000 websites (**No of Occurrences x Weight**)

The most afflictive external attack vectors are:

- Content-Security-Policy **of Service Misconfiguration**
- X-XSS-Protection **of Service Misconfiguration**
- Cookie Attribute – HTTP Only **of Service Misconfiguration**

Table 4.7

Least 10 Attack Vectors Based on Severity (Weight)

S. No	Attack Vector	Attack Landscape	No of Occurrences	Weight	Total Weight
1	CONNECT HTTP Method Enabled	Service Misconfiguration	1	4.3	4.3
2	Spameatingmonkey - Spam Emails	IP Reputation	1	8.6	8.6

3	Spameatingmonkey - Policy Blocklist	IP Reputation	1	8.6	8.6
4	Sorbs DB - Web	IP Reputation	1	8.6	8.6
5	Sorbs DB - Socks Proxy	IP Reputation	1	8.6	8.6
6	Sorbs DB - SMTP	IP Reputation	1	8.6	8.6
7	Sorbs DB – No server	IP Reputation	1	8.6	8.6
8	Sorbs DB - HTTP Proxy	IP Reputation	1	8.6	8.6
9	Sorbs DB - Escalations	IP Reputation	1	8.6	8.6
10	Sorbs DB - Misc Proxy	IP Reputation	1	8.6	8.6

No of Occurrences – Denotes count of each attack vector across 1000 websites

Weight – Denotes severity of attack vector.

Total Weight – Denotes the sum of the severity of each attack vector across 1000 websites (**No of Occurrences x Weight**)

The least afflictive external attack vectors are:

- CONNECT HTTP Method Enabled of **Service Misconfiguration**

For 200 Data Points (Attack Vectors):

No of unique attack vector: 48

Table 4.8

Top 10 Attack Vectors Based on Severity (Weight)

S. No	Attack Vector	Attack Landscape	No of Occurrences	Weight	Total Weight
1	Employee Credentials Available in Breached Site	Data Leaks	3723	10.0	37230.0
2	Content-Security-Policy	Service Misconfiguration	3040	5.8	17632.0
3	X-XSS-Protection	Service Misconfiguration	2716	6.1	16567.6
4	Cross Origin Resource Sharing	Service Misconfiguration	3139	4.6	14439.4
5	Strict-Transport-Security	Service Misconfiguration	2389	5.1	12183.9
6	Cookie Attribute - HttpOnly	Service Misconfiguration	1897	6.1	11571.7
7	Referrer-Policy	Service Misconfiguration	3086	3.4	10492.4
8	X-Frame-Options	Service Misconfiguration	2224	4.5	10008.0
9	X-Content-Type-Options	Service Misconfiguration	2472	3.8	9393.6

10	Missing Pragma	Service Misconfiguration	2574	3.6	9266.4

No of Occurrences – Denotes count of each attack vector across 200 websites

Weight – Denotes severity of attack vector.

Total Weight – Denotes the sum of the severity of each attack vector across 200 websites (**No of Occurrences x Weight**)

Most afflictive external attack vectors are:

- Employee Credentials Available in Breached Site **of Data Leaks**
- Content-Security-Policy **of Service Misconfiguration**
- X-XSS-Protection **of Service Misconfiguration**

Table 4.9

Least 10 Attack Vectors Based on Severity (Weight)

S. No	Attack Vector	Attack Landscape	No of Occurrences	Weight	Total Weight
1	Missing Cache-Control	Service Misconfiguration	2	3.6	7.2
2	Spamhaus - xbl	IP Reputation	1	8.6	8.6
3	Sorbs DB - Dynamic	IP Reputation	1	8.6	8.6
4	Sorbs DB - Web	IP Reputation	1	8.6	8.6
5	Sorbs DB - Noserver	IP Reputation	2	8.6	17.2

6	Zone Transfer	DNS Misconfiguration	3	6.8	20.4
7	Self-Signed Certificate	SSL Health	5	4.1	20.5
8	Sorbs DB - Spam	IP Reputation	3	8.6	25.8
9	Sorbs DB - Root	IP Reputation	4	8.6	34.4
10	Spamhaus - pbl	IP Reputation	6	8.6	51.6

No of Occurrences – Denotes count of each attack vector across 200 websites

Weight – Denotes severity of attack vector.

Total Weight – Denotes sum of severity of each attack vector across 200 websites (**No of Occurrences x Weight**)

Least afflictive external attack vectors are:

- Missing Cache-Control of **Service Misconfiguration**

Common inference for both 1000 and 200 datapoints:

- **Content-Security-Policy of Service Misconfiguration and X-XSS-Protection of Service Misconfiguration** are the **most afflictive external attack vector** found in both datasets.

Monetary impact of attack vectors and threats: Assumptions:

- Data needs to be transformed based on the assumption that the monetary impact of multiple occurrences of an attack vector will be the same as a single occurrence of the same attack vector.

- Considering the assumption, we will transform the data based on the below steps:
    - If no of occurrences per attack vector and website > 1, set 1
    - If no of occurrences per attack vector and website is 0, set 0
- For example: Exploitation of 2 different misconfigurations in a domain led to compromise of the same server and impact as well.

For 1000 data points:

- Based on the above method and manipulation of data, we arrive at the **table 3.10**

The total number of threats found: 3778

Table 4.10

Total No. of Findings by each Attack Vector

S. No	Attack Vector	Total No of Findings	Percent
1	IP Reputation	975	25.81%
2	Service Misconfiguration	963	25.49%
3	SSL Health	878	23.24%
4	Outdated Version	338	8.95%
5	Data Leaks	336	8.89%
6	Unnecessary Open Ports	286	7.57%
7	Data Breaches	2	0.05%
8	DNS Misconfiguration	0	0.0%

Table 4.11

Table Cost of each Attack Vector and Threat

S. No	Attack Vector	Cost in USD (Millions)
1	Data Leaks	3.94
2	Data Breaches	4.35
3	DNS Misconfiguration	4.14
4	Service Misconfiguration	4.14
5	Outdated Version	4.14
6	SSL Health	4.35
7	IP Reputation	4.35
8	Unnecessary Open Ports	4.35

Note:

- **We have arrived at the cost of each attack vector by using the data from the IBM Data Breach report 2022. Cost** in the **table 3.11** is the average cost of each attack vector on successful exploitation.
- IBM used activity-based costing, which identifies activities and assigns a cost according to actual use.
- The activity-based costing was based on four activities:
    - Detection and escalation
    - Notification
    - Post-breach response

- Lost business

Total Cost in million USD: 16023.33 or 16 billion USD

Table 4.12

Sum of Cost of each Attack Vector across 1000 websites

S. No	Attack Vector	Sum of Cost in million USD
1	IP Reputation	4241.25
2	Service Misconfiguration	3986.82
3	SSL Health	3819.30
4	Outdated Version	1399.32
5	Data Leaks	1323.84
6	Unnecessary Open Ports	1244.10
7	Data Breaches	8.70
8	DNS Misconfiguration	0.00

Top 3 attack vectors based on the cost:

- IP Reputation
- Service Misconfiguration
- SSL Health

Top 3 attack vectors contribute **74.54%** of the total no of threats and would have costed **12.04 billion USD** in case of successful exploitation.

For 200 data points (Attack vectors)

- Based on the method mentioned in assumption and manipulation of data we arrive at the t**able 3.13:**

The total number of findings: 934

Table 4.13

Total No of Findings by each Attack Vector

S. No	Attack Vector	Total No of Findings	Percentage
1	IP Reputation	176	18.84%
2	Service Misconfiguration	174	18.63%
3	SSL Health	161	17.24%
4	DNS Misconfiguration	117	12.53%
5	Data Leaks	113	12.1%
6	Unnecessary Open Ports	97	10.39%
7	Outdated Version	96	10.28%
8	Data Breaches	0	0.0%

Total Cost in million USD: 3935.30 or 3.9 billion USD

Table 4.14

Sum of Cost of each Attack Vector Across 1000 Websites

S. No	Attack Vector	Sum of Cost in million USD
1	IP Reputation	765.60

2	Service Misconfiguration	720.36
3	SSL Health	700.35
4	DNS Misconfiguration	484.38
5	Data Leaks	445.22
6	Unnecessary Open Ports	421.95
7	Outdated Version	397.44
8	Data Breaches	0.00

Top 3 attack vector based on cost:

- IP Reputation
- Service Misconfiguration
- SSL Health

Top 3 attack vectors contribute **54.71%** of total no of threats and would have costed **2.17 billion USD** in case of successful exploitation.

Common inference for both 1000 and 200 Datapoints:

Top 3 attack vectors based on cost for both datasets are:

- IP Reputation
- Service Misconfiguration
- SSL Health

Combining both datasets, the total Cost: 19.90 billion USD

Top 3 attack vectors contribute **71.40%** of total no of threats and would have costed **14.21 billion USD** in case of successful exploitation.

For Threats:

The total number of threats found: 201

Table 4.15

Total No of Threats by each Threat Vector

S. No	Threats	Total No of Threats	Percentage
1	Data Leaks	56	27.86%
2	Rogue Mobile Apps	51	25.37%
3	Phishing Threats	47	23.38%
4	Brand & Reputation Threats	43	21.39%
5	Data Breaches	4	1.99%

Table 4.16

Table of Cost of each Attack Vector and Threat

S. No	Threats	Cost in USD (Millions)
1	Phishing Threats	4.91
2	Rogue Mobile Apps	4.10
3	Data Leaks	3.94
4	Data Breaches	4.35
5	Brand & Reputation Threats	4.35

Note:

- **The cost of each threat** is obtained by using data from the IBM Data Breach report 2022. The cost in **table 3.16** is the average cost of each threat vector on successful exploitation.
- IBM used activity-based costing, which identifies activities and assigns a cost according to actual use.
- The activity-based costing was based on four activities:
    - Detection and escalation
    - Notification
    - Post-breach response
    - Lost business.

Total Cost in million USD: 864.96

Table 4.17

Total cost of each Threat Vector

S. No	Threats	Total Cost in million USD
1	Phishing Threats	230.77
2	Data Leaks	220.64
3	Rogue Mobile Apps	209.10
4	Brand & Reputation Threats	187.05
5	Data Breaches	17.40

Top 3 threat vectors based on cost:

- Phishing Threats
- Data Leaks
- Rogue Mobile Apps

Top 3 threats contribute **76.61%** of total no of threats and would have costed **660.51 million USD** in case of successful exploitation.

Priority matrix for implementing proactive controls for external attack vectors (with less effort for maximum risk reduction):

For 1000 data points:

Table 4.18

Reduction of Threat Score in % when Attack Vector is Removed from Threat Score Calculation and Complexity to Fix each Attack Vector

S. No	Attack Vector	Total Weight	Average Weight	Threat Score Reduced in % (AVG)	Complexity for Fixing each Attack Vector
1	Service Misconfiguration	88.1	4.89	58.57%	2
2	SSL Health	23.1	4.62	13.02%	3
3	IP Reputation	189.2	8.60	12.60%	5

4	Unnecessary Open Ports	46.9	7.82	6.46%	4
5	Data Leaks	20.0	10.00	5.54%	8
6	Outdated Version	19.0	9.50	3.79%	6
7	Data Breaches	10.0	10.00	0.02%	7
8	DNS Misconfiguration	0.0	0.00	0.00%	1

Weight – Denotes severity of attack vector.

Total Weight – Addition of weight(severity) of each attack vector present in the attack vector landscape.

Average Weight – Denotes average severity of each attack vector of each attack landscape

For 200 data points:

Table 4.19

Reduction of Threat Score in % when Attack Vector is Removed from Threat Score Calculation and Complexity to Fix each Attack Vector

S. No	Attack Vector	Total Weight	Average Weight	Threat score Reduced in % (AVG)	Complexity for Fixing each Attack Vector
1	Service Misconfiguration	91.5	5.08	53.78%	2
2	SSL Health	27.2	4.53	11.57%	3

3	IP Reputation	86.0	8.60	8.77%	5
4	Data Leaks	20.0	10.00	8.08%	8
5	Unnecessary Open Ports	46.9	7.82	6.83%	4
6	Outdated Version	19.0	9.50	5.59%	6
7	DNS Misconfiguration	21.2	5.30	5.40%	1
8	Data Breaches	0.0	0.00	0.00%	7

Table 4.20

Priority Matrix for Implementing Proactive Controls for External Attack Vectors (with less effort for maximum risk reduction)

S. No	Attack Vector	Priority
1	Service Misconfiguration	1
2	SSL Health	2
3	IP Reputation	3
4	Unnecessary Open Ports	4
5	Outdated Version	5
6	Data Leaks	6
7	Data Breaches	7

8	DNS Misconfiguration	8

For 1000 data points:

Table 4.21

Reduction of Threat score in % when Attack Vector is Removed from Threat Score Calculation and Complexity to Fix each Attack Vector

S. No	Attack Vector	Total Weight	Average Weight	Threat Score Reduced in % (AVG)	Complexity for Fixing each Attack Vector
1	Service Misconfiguration	88.1	4.89	58.57%	2
2	SSL HEALTH	23.1	4.62	13.02%	3
3	IP Reputation	189.2	8.60	12.60%	6
4	Unnecessary Open Ports	46.9	7.82	6.46%	4
5	Data Leaks	20.0	10.00	5.54%	7
6	Outdated Version	19.0	9.50	3.79%	5
7	Data Breaches	10.0	10.00	0.02%	8
8	DNS Misconfiguration	0.0	0.00	0.00%	1

Weight – Denotes severity of attack vector.

Total Weight – Addition of weight(severity) of each attack vector present in the attack vector landscape.

Average Weight – Denotes the average severity of each attack vector of each attack landscape.

For 200 data points:

Table 4.22

Reduction of Threat Score in % when Attack Vector is Removed from Threat Score Calculation and Complexity to Fix each Attack Vector

S. No	Attack Vector	Total of Individual Weight	Average of Individual Weight	Threat Score Reduced in % (AVG)	Complexity for Fixing each Attack Vector
1	Service Misconfiguration	91.5	5.08	53.78%	2
2	SSL Health	27.2	4.53	11.57%	3
3	IP Reputation	86.0	8.60	8.77%	6
4	Data Leaks	20.0	10.00	8.08%	7
5	Unnecessary Open Ports	46.9	7.82	6.83%	4
6	Outdated Version	19.0	9.50	5.59%	5
7	DNS Misconfiguration	21.2	5.30	5.40%	1
8	Data Breaches	0.0	0.00	0.00%	8

Table 4.23

Priority Matrix for Implementing Remediation (with less effort for maximum risk reduction)

S. No	Attack Vector	Priority
1	Service Misconfiguration	1
2	SSL Health	2
3	IP Reputation	3
4	Unnecessary Open Ports	4
5	Outdated Version	5
6	Data Leaks	6
7	Data Breaches	7
8	DNS Misconfiguration	8

The different patterns identified during the analysis-

For 200 data points:

Data Leaks by Platforms (Public Websites)

Table 4.24 shows the summary of data leaks across different public platforms available on the internet like search engines, code sharing/file sharing platforms, etc.

Table 4.24

No of Occurrences of Data Leak on Different Platforms

S. No	Platforms	No of Occurrences
1	Credential Leaks (Others)	65
2	Code Leaks	22
3	Google Index Leaks	20
4	Pastebin Leak	15
5	DarkWeb Forums	8
6	StackOverFlow Leak	7
7	Github Leak	6
8	Bitbucket Leak	5
9	Code to Scrap Site Data	4
10	Trello	1
11	Shodan.io	1
12	Telegram Leak	1

Top threats found are:

- **Google and Pastebin platforms have a higher number of data leaks present.**

By Threat Landscape:

Below is the summary of external threats found in the different threat landscapes.

Table 4.25

No of Occurrences by each Threat Landscape

S. No	Threat Landscape	No of Occurrences
1	Rogue Mobile Apps	258
2	Data Leaks	155
3	Brand Reputation	143
4	Phishing	134
5	Data Breaches	4

Top Threat Landscape is **Rogue Mobile Apps.**

Least Threat Landscape is **Data Breaches**.

Phishing Type:

Below is the summary of different types of phishing threats found.

Table 4.26

No of Occurrences by each Threat Landscape

S. No	Phishing Type	No of Occurrences
1	Typosquatting domain names	126
2	Phishing pages	7
3	Domain Threat	1

Top threats in phishing is **Typo squatting domain names.**

Brand Impersonation by type:

The table 4.27 shows the summary of different types of brand impersonation threats found.

Table 4.27

No of occurrences by each threat landscape

S. No	Brand Impersonation	No of Occurrences
1	Unofficial Social Media Profile	94
2	Brand Impersonation (Websites)	38
3	Brand Damage	6
4	Unclaimed Social Media Profile	2
5	Search Engine Indexed Unwanted Info	2
6	Public Vulnerability Disclosure	1

Top threats in brand reputation is '**Unofficial Social Media Profile'.**

By threats: Table 3.28 shows the summary of all the external cyber threats identified across the 200 websites.

Table 4.28

No of Occurrences by each Threat Type

S. No	Threats	No of Occurrences	Threat Landscape
1	Unofficial App Store	250	Rogue Mobile Apps

2	Typosquatting domain names	126	Phishing
3	Unofficial Social Media Profile	94	Brand Reputation
4	Credential Leak	65	Data Leaks
5	Brand Impersonation	38	Brand Reputation
6	Code Leak	22	Data Leaks
7	Google Index Leaks	20	Data Leaks
8	Pastebin.com Leak	15	Data Leaks
9	Impersonated App	8	Rogue Mobile Apps
10	DarkWeb Forums	8	Data Leaks
11	Phishing Pages	7	Phishing
12	Stack overflow Leak	7	Data Leaks
13	Github Leak	6	Data Leaks
14	Brand Damage	6	Brand Reputation
15	Bitbucket Leak	5	Data Leaks
16	Code to Scrap Site Data	4	Data Leaks
17	Possible Past Data Breach	4	Data Breaches
18	Unclaimed Social Media Profile	2	Brand Reputation
19	Search Engine Indexed Unwanted Info	2	Brand Reputation
20	Trello	1	Data Leaks
21	Shodan.io	1	Data Leaks
22	Telegram Leak	1	Data Leaks

23	Domain Threat	1	Phishing
24	Public Vulnerability Disclosure	1	Brand Reputation

Top 3 threats found are:

- **Unofficial App store** of threat landscape **Rouge Mobile Apps.**
- **Typosquatting domain names** of threat landscape **Phishing.**
- **Unofficial Social Media Profile** of threat landscape **Brand Reputation.**

Least threats found are:

- **Trello, Shodan.io and Telegram Leak of threat** landscape **Data Leaks**.
- **Domain Threat** of threat landscape **Phishing.**
- **Public Vulnerability Disclosure** of threat landscape **Brand Reputation.**

Table 4.29

Analysis Summary

Inferences	Attack Vectors for 1000 data points		Attack Vectors for 200 data points		Threats for 200 data points	
	Attack vector	**Percent**	**Attack Vector**	**Percent**	**Threats**	**Percent**
Most Common Attack Vectors or Threats	Service Misconfiguration	66.82%	Service Misconfiguration	60.75%	Rogue Mobile Apps	35.83%
	SSL Health	14.91%	SSL Health	13.77%	Data Leaks	23.98%
	IP Reputation	8.17%	DNS Misconfiguration	6.97%	Brand & Reputation Threats	19.89%

	Attack Vector	**Percent**	**Attack Vector**	**Percent**	**Threats**	**Percent**
Least Common Attack Vectors or Threats	DNS Misconfiguration	0%	Data Breaches	0%	Data Breaches	0.54%
	Data Breaches	0.2%				
	Attack Vector	**Attack Landscape**	**Attack Vector**	**Attack Landscape**		
Most Affiliative Attack Vector	Content-Security-Policy	Service Misconfigure	Employee Credentials Available	Data Leaks		
	X-XSS-Protection	Service Misconfigure	Content-Security-Policy	Service Misconfigure		
	Cookie Attribute – HTTP Only	Service Misconfigure	X-XSS-Protection	Service Misconfigure		
	Attack Vector	**Attack Landscape**	**Attack Vector**	**Attack Landscape**		
Least Affiliative Attack Vector	CONNECT HTTP Method Enabled	Service Misconfigure	Missing Cache-Control	Service Misconfigure		
	Attack Vector	**Total Cost in Million USD**	**Attack Vector**	**Total Cost in Million USD**	**Threats**	**Total Cost in Million USD**
Monetary impact of attack vectors and threats						

IP Reputation	4241.25	IP Reputation	765.60	Phishing Threats	230.77
Service Misconfiguration	3986.82	Service Misconfiguration	720.36	Data Leaks	220.64
SSL Health	3819.30	SSL Health	700.35	Rogue Mobile Apps	209.10

4.2 Conclusion

After analyzing 1200 websites, it was found that Service Misconfiguration was the most prevalent vulnerability, affecting 66.82% of the sites. Content Security Policy and

X-XSS-Protection were identified as the most frequent associated attacks. According to IBM report 2020, if Service Misconfiguration is successfully exploited, the potential financial impact alone could be as high as 3.99 billion USD. SSL Health and IP Reputation vulnerabilities were also identified as significant contributors to website vulnerabilities, with potential financial losses of 3.8 billion USD and 4.24 billion USD, respectively, if successfully exploited across the 1200 websites.

Among the 200 websites analysed, Rogue Mobile Apps posed the most significant threat, accounting for 35.83% of the total, followed by Data Leaks with 23.98%, and Brand & Reputation Threats with 19.89%. The successful exploitation of Rogue Mobile Apps, Data Leaks, and Phishing threats could lead to potential financial losses of 209.10 million USD, 220.64 million USD, and 230.77 million USD, respectively.

These findings underscore the importance of identifying and mitigating common vulnerabilities and threats to prevent potential financial losses and safeguard the security and reputation of websites. It is recommended that website owners and administrators take proactive measures to regularly assess and address vulnerabilities in their systems and applications.

CHAPTER V

FRAMEWORK

The objective of the study on attack surface monitoring and data analysis was to develop a guideline for organizations to safeguard themselves against external cyber-attacks. Through the research, a holistic comprehension of the prevalent and uncommon attack vectors and threats was attained, along with a monetary impact analysis and priority matrices to assist in implementing proactive controls and remediation measures.

The study was executed by utilizing two datasets, one comprising 1000 data points and the other with 200 data points. The research findings were consistent in both datasets, with minor variations in the identification of specific attack vectors and threats.

Based on the data analysis of 1000 data points, the most common attack vectors and threats of external cyber-attacks were found to be service misconfiguration, SSL health, and IP reputation. This data aligns with second research conducted with 200 data points, which also found service misconfiguration, SSL health, and DNS misconfiguration to be the most common attack vectors. Service Misconfiguration is the most common attack vector as it refers to a vulnerability that arises when a service or application is not configured correctly. SSL Health refers to vulnerabilities that arise when a website does not have a valid SSL certificate or when the SSL certificate is not configured correctly. IP Reputation refers to vulnerabilities that arise when an IP address is blacklisted or has a poor reputation. The least common attack vectors of external cyber-attacks are Data Breaches. Data Breaches refer to vulnerabilities that arise when sensitive data is stolen or compromised.

Based on the research, the Service Misconfiguration category was found to be the most harmful external attack vector. This is because Service Misconfiguration can result in several vulnerabilities, including Content-Security-Policy, X-XSS-Protection, and Cookie Attribute - HttpOnly, which can lead to serious security breaches if not addressed. Conversely, the Content-Security-Policy and X-XSS-Protection were identified as the least harmful attack vectors, and can be remedied through appropriate configuration adjustments.

The monetary impact of these various attack vectors and threats differed based on the specific attack and the organization's level of preparedness. Nonetheless, the IBM Data Breach 2022 report showed that the cost of a data breach can be substantial, with an average cost of $4.35 million.

Furthermore, the study discovered that the financial impact of different attack vectors and threats was considerable. The top three attack vectors that had the highest cost implications were IP reputation, service misconfiguration, and SSL health. In the first dataset, these three attack vectors accounted for 74.54% of total threats and could have resulted in a loss of 12.04 billion USD if successfully exploited. In the second dataset, they represented 54.71% of total threats and could have resulted in a loss of 2.17 billion USD if successfully exploited. Additionally, the top three threat vectors with the highest cost implications were phishing threats, data leaks, and rogue mobile apps. In the event of successful exploitation, these three threats contributed to 76.61% of total threats and could have cost 660.51 million USD.

The research also identified a priority matrix for implementing proactive controls and remediation for external attack vectors. The matrix ranked service misconfiguration as the highest priority, followed by SSL health, IP reputation, unnecessary open ports, outdated version, data leaks, and data breaches. The research also identified easy-to-implement guidelines for preventing and remediating attacks from external threat vectors that were covered in the guideline below. Also, the research found that prioritizing efforts to address service misconfiguration and SSL health can provide the greatest risk reduction with the least effort.

The research also found that the frequency of monitoring required for each external attack vector will vary depending on the specific vector and the organization's level of risk. However, it was recommended that organizations regularly review and update their security configurations and monitor for suspicious activity to minimize the risk of a successful attack.

In terms of monitoring frequency, the research found that service misconfiguration, SSL health, and IP reputation should be monitored on a daily basis, while DNS misconfiguration, should be monitored on a weekly basis and data leaks, and data breaches should be monitored on a continuous basis.

The research also identified several patterns from the analysis, including that phishing domains and rogue apps were the most common across all platforms, and that the threat landscape was constantly evolving, with new threats emerging regularly. Overall, the research provided a comprehensive understanding of the external cyber-threat landscape and the most effective ways to protect against them. The guideline document created as a result of this research can be used by organizations to effectively protect against external cyber-attacks and minimize the impact of any breaches that may occur.

I wrote 21 survey questions on Attack Surface Management and reached out to CISOs to fill them. The questionnaire was made on the Typeform platform, then I emailed it to all the

CISOs of my reference and requested them to fill it out. I inferred survey questions by qualitative data analysis. And later, the top 5 CISOs sat for multiple rounds of one to one interviews. The interview of 30-40 minutes was taken on Teams application. Again the data was inferred by using qualitative data analysis.

The survey revealed insightful findings regarding the visibility of external assets and the use of automated tools for asset discovery. Interestingly, the majority of CISOs responded negatively to having complete visibility of external assets, with most selecting "No" or "Partially." However, an overwhelming 95% of CISOs agreed that using automated tools for discovering, maintaining, and updating assets is a necessity for every organization.

Regarding the discovery of unsanctioned shadow IT assets, CISOs preferred using Asset Discovery Tools, Attack Surface Monitoring Tools, and periodic asset reviews by the IT team. Additionally, to address the use of unsanctioned shadow IT assets, most CISOs recommended companies to understand the needs of their employees and adapt IT policies accordingly, educate employees about shadow IT and its risks, and identify the business requirements that Shadow IT meets and provide an approved alternative.

The biggest problem that the CISOs identified with shadow IT assets were unknown/undiscovered assets and the use of unsanctioned software. To mitigate these problems, most CISOs believed that implementing proactive controls for internet-facing assets is essential for safeguarding the attack surface. Additionally, the majority of CISOs agreed that organizations must have documented configuration baselines for domains, servers, cloud, DNS, social media accounts, and other external assets.

Regarding security risk assessments, the CISOs thought that it is necessary to consider the entire attack surface and do it continuously. The survey also revealed that most CISOs include third parties in their attack surface management, but only partially. Furthermore, vulnerability remediation was perceived as a lengthy process that often misses urgency by most CISOs.

Finally, the CISOs thought that bi-weekly or monthly frequency for reviewing scan results and prioritizing vulnerabilities found in external attack surfaces was appropriate. In conclusion, the survey highlighted the importance of having complete visibility of external assets, using automated tools for asset discovery, discovering unsanctioned shadow IT assets, and implementing proactive controls to safeguard the attack surface. The survey also highlighted the need to consider the entire attack surface and third-party interactions in security risk assessments and the importance of timely vulnerability remediation.

As part of the research, several Chief Information Security Officers (CISOs) were interviewed to gain insight into their experiences and perspectives on managing attack surfaces. The CISOs who were interviewed indicated that the most significant challenge posed by the expanding external attack surface is the sheer number of entry points available for hackers to gain access to the corporate network. In addition, the dynamic and unpredictable nature of the unknown attack surfaces also presents a significant obstacle.

The CISOs reached a consensus that security leaders lack complete visibility into external assets. The primary cause of this limited visibility is the large number of unknown assets and the lack of effective monitoring. They stressed the importance of aligning people, processes, and technology towards managing the external attack surface to gain the necessary visibility.

The CISOs also discussed challenges in managing the attack surface, including identifying and tracking shadow IT assets and educating employees to provide alternatives for shadow IT. They recommended implementing proactive controls for internet-facing assets to tackle today's threats. Additionally, continuous monitoring of the attack surface is necessary to identify the most afflictive attack vectors and maintain a good security posture for the organization. However, measuring the effectiveness of attack surface management efforts and communicating the state of the organization's attack surface to senior leadership and stakeholders remains a challenge.

Overall, the research and guideline document provides valuable insights and recommendations for organizations looking to improve their attack surface monitoring efforts. By prioritizing efforts to address service misconfiguration and SSL health, regularly reviewing and updating security configurations, and having a response plan in place, organizations can better protect themselves from external cyber threats.

The research sets a strong baseline for effectively managing attack surface. However, it is important to note that a full-fledged attack surface management program needs several customizations to be made in order to effectively address the unique needs of an organization. This includes customizing the guideline document to fit the specific infrastructure, applications, and threat landscape of an organization. Additionally, it is important for organizations to continuously monitor and update their guideline document to adapt to the ever-changing threat landscape. Therefore, it is crucial for organizations to continuously assess and adapt their attack surface management program to ensure they are effectively addressing their specific attack surface.

The methodology that we have created as part of this research is a six-phase process that aims to effectively manage external attack surface. The six phases of the methodology are: ‘Discover’, ‘Reduce’, ‘Protect’, ‘Assess’, ‘Prioritize’, and ‘Remediate’.

The first phase, **Discover**, aims to identify and understand the organization's external attack surface. This includes identifying all the assets, systems, and applications that are exposed to the internet. This phase is crucial for understanding the organization's attack surface and for identifying areas that need to be addressed.

The second phase, **Reduce,** aims to minimize the attack surface by removing unnecessary assets and applications, closing unnecessary ports, and removing any outdated versions. This phase is important for reducing the organization's attack surface and for making it less attractive to attackers.

The third phase, **Protect,** aims to implement security controls to protect the organization's assets, systems, and applications. This includes implementing security controls, firewalls, intrusion detection and prevention systems. This phase is crucial for preventing external cyber-attacks and for detecting any potential threats.

The fourth phase, **Assess,** aims to evaluate the effectiveness of the security controls that have been implemented. This includes identifying vulnerabilities and misconfigurations present in the identified assets. This phase is important for identifying any potential vulnerabilities and for identifying any new threats that have emerged.

The fifth phase, **Prioritize,** aims to prioritize vulnerabilities and threats based on the organization's risk appetite and the potential impact of successful exploitation. This phase is crucial for effectively allocating resources and for addressing the most critical vulnerabilities and threats first.

The sixth and final phase, **Remediate,** aims to address and fix any vulnerabilities and threats that have been identified. This includes applying patches and updates, implementing security controls, and developing incident response plans. This phase is critical for preventing external cyber-attacks and for ensuring the organization's attack surface is secure.

It's important to note that this methodology is a cyclic process, which means once the initial phases have been completed, organizations should continue to monitor and assess their attack surface to identify new vulnerabilities or misconfigurations and to ensure that the security controls are still effective. This process helps organizations to continuously improve their security posture.

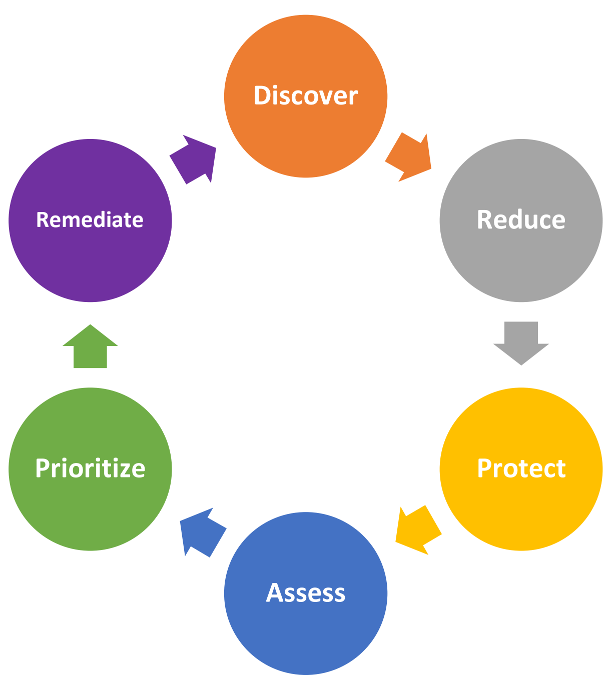

Figure 5

The Six Phases of Methodology

DISCOVER – Discover the attack surface

You can't protect what you don't know about, so the first step is to identify all systems, applications, and networks that make up your organization's attack surface. This may include hardware, software, mobile devices, cloud systems, and external partners.

Table 5.1

Guidelines for Discover Phase

1. Asset Discovery		
1.1 Domains and Subdomains		
Discovering Domains and Sub-domain helps to broader the attack surface, find hidden applications, and forgotten subdomains.		
1.1.1	TLD Listing	List all the TLDs used by the organization.
1.1.2	Domain Name System (DNS) Enumeration	Discover all the subdomains using both active and passive methods available. This should include subdomains that are inactive currently but used in the past by the organization. Passive Enumeration # • Certificate Transparency • Google Dorking • DNS Aggregators

- ASN Enumeration
- Subject Alternate Name (SAN)
- Rapid7 Forward DNS dataset

Active Enumeration #

- Brute Force Enumeration
- Zone Transfer
- DNS Records
- Content Security Policy (CSP) Header

1.2 IP Addresses

IP Address Listing	List all the IP Addresses used by the organization.
IP Address Enumeration	Discover all the IPs using both active and passive methods available. This should include both IPv4 as well as IPv6. Some of the IP Address Enumeration methods. • DNS Records • CT Logs • Censys • Shodan • Reverse IP Lookup

1.3 Cloud Assets

Cloud Assets Listing	List all the Cloud Assets used by the organization.
Cloud Assets Discovery	Discover all the cloud assets using both active and passive methods available. This should include cloud instances, storage, databases, load balancers, etc. Some of the discovery methods. • DNS Records • Reverse DNS Lookup • Cloud Storage Discovery Tools • SpiderFoot

Code-Repos

Code-Repos Listing	List all the Code Repositories used by the various dev teams within the organization.
Code-Repos and Secret Discovery	Discover all the code-repos that have the organization code by using various discovery methods. • Code-Repo Discovery Tools using Keywords • git-secret • grep.app

- searchcode

1.4 Social Media Pages

Official Social Media Pages Listing	List all the Official Social Media pages used by the organization.
Social Media Pages Discovery	Use tools and manual OSINT methods to discover social media pages used by the organization

1.5 Mobile Apps

Official Mobile Apps Listing	List all the Official Mobile Apps published by the organization in Google Play Store, IOS App Store, etc.
Official Mobile Apps Discovery	Use tools and manual OSINT methods to discover other Mobile Apps published by the organization

1.5.1 Vendors and Suppliers

Vendors and Suppliers Listing	List all the Vendors and Suppliers to whom organization data are shared.

1.5.2 SaaS Tools

SaaS Tools Listing	List all the SaaS Tools that are consumed by the organization.

1.5.3 A1 Fingerprinting

Fingerprinting the digital assets	Use tools and manual OSINT methods to fingerprint all the discovered assets. • BuiltWith • Nmap • p0f • httprecon
1.5.4 Asset Classification	
Classification	Classify the discovered assets using the following parameters • Asset exposure (Internal, External, Public, Private, etc) • Business criticality • Has valuable data? • Has sufficient Security Controls?

The discovery section plays a crucial role in understanding the complete attack surface of the organization. The research conducted showed that organizations should discover all the assets that are connected to the Internet including domains, cloud servers, SaaS apps, mobile apps, etc., as these assets pose a higher risk of being targeted by external attackers. In addition, it is important to identify all third-party software and services that the organization uses, as these can also pose a risk to the organization's attack surface. By identifying all assets and third-party services, organizations can better understand their attack surface and prioritize security measures accordingly. The consecutive scans done for the sample taken from the 200 companies showed that regular monitoring of the attack surface can help organizations to identify new assets that are added to the network and take proactive measures to secure them. This can be achieved through the baseline given above either manually or using an automated asset discovery or ASM tool. The use of Automated ASM tools like Threat Meter will help simplify the continuous discovery of complete attack surfaces including the above baseline.

REDUCE – Remove the unwanted assets to reduce the attack surface.

Attack surface reduction (ASR) is a security strategy that aims to minimize the opportunities for an attacker to exploit vulnerabilities in a system or network. This is accomplished by reducing the number of potential entry points for an attacker, as well as limiting the potential damage that can be caused if a successful attack is launched. By reducing the attack surface, organizations can better protect themselves against cyber threats and limit the potential damage from a successful attack. Additionally, as the number of connected devices and systems continues to grow, the attack surface of many organizations is becoming increasingly complex and difficult to secure, making attack surface reduction an essential component of any comprehensive security strategy.

Table 5.2

Guidelines for Reduce Phase

2. Asset Reduction
The simplest way to reduce your attack surface is to eliminate assets no longer relevant to your organization's operations.

2.1	Shadow IT Assets	Restrict access to identified unsanctioned applications and assets.
2.2	Unused Assets	• Remove unused and auto-created subdomains. • Clean up application codes that are out of date or no longer necessary. • Review the list of active email accounts and deactivate the unused ones. • Remove cloud instance services/ports that are no longer in use. • Clean up any other assets that are no longer necessary.

Reducing the attack surface is important because it minimizes the number of entry points for potential cyber attacks and reduces the risk of a successful attack. By reducing the attack surface, organizations can minimize the risk of exploitation. To reduce the attack surface, organizations can implement the above guidelines as a baseline such as regularly identifying and removing shadow IT assets and unused assets.

PROTECT – Implement proactive controls to prevent it.

Analysis of the data suggests patterns such as a higher frequency of attacks on specific platforms, a higher impact of certain external attack vectors, and variations in the threat landscape. It is important for organizations to regularly conduct research on their attack surface and implement proactive measures to reduce the risk of external cyber-attacks. As an outcome of the research, a priority matrix for implementing proactive controls has arrived. The priority matrix was based on the effort required to implement the controls and the maximum risk reduction achieved. The research found that service misconfiguration was the top priority for implementing proactive controls, followed by SSL health and IP reputation. The research also found that the priority required for each external attack vector will vary depending on the organization's level of risk.

Priority matrix for implementing proactive controls (with less effort for maximum risk reduction)

The priority matrix for implementing proactive controls was arrived at through extensive data analysis of 1000 and 200 data sets on external cyber-attacks and their associated attack vectors and threats. The research focused on identifying the most common and afflictive attack vectors, as well as the monetary impact of these different attack vectors and threats. Based on this information, the priority matrix for implementing proactive controls was determined, placing a higher priority on addressing service misconfiguration and SSL health. The priority matrix was designed to prioritize risk reduction with minimal effort.

Table 5.3

Reduction of Threat Score in % when Attack Vector is Removed from Threat Score Calculation and Complexity to Fix each Attack Vector

S. No	*Attack Vector*	*Total Weight*	*Average Weight*	*Threat Score Reduced after Remediation % (AVG)*	*Complexity for Fixing each Attack Vector*	*Priority*
1	Service Misconfiguration	91.5	5.08	53.78%	2	1
2	SSL Health	27.2	4.53	11.57%	3	2
3	IP Reputation	86.0	8.60	8.77%	5	3
4	Unnecessary Open Ports	46.9	7.82	6.83%	4	4
5	Outdated Version	19.0	9.50	5.59%	6	5
6	Data Leaks	20.0	10.00	8.08%	8	6
7	Data Breaches	0.0	0.00	0.00%	7	7
8	DNS Misconfiguration	21.2	5.30	5.40%	1	8

To protect the attack surface, organizations can implement the below guidelines as baseline security controls.

Table 5.4

Guidelines for Protect Phase

SSL Configurations

Discovering Domains and Sub-domain helps to broader the attack surface, find hidden applications, and forgotten subdomains.

Certificate Authority (CA)	Obtain Certificates from a reliable and trustworthy Certificate Authority (CA)
Private Keys	Use Strong Private Keys: At least a 2048-bit RSA key or 256-bit ECDSA key is recommended Protect Your Private Keys: • Generate your own private keys on a secure and trusted environment (preferably on the server where they will be deployed or a FIPS or Common Criteria compliant device). Never allow a CA (or anyone else) to generate private keys on your behalf. • Only give access to private keys as needed. Generate new keys and revoke all certificates for the old keys when employees with private-key access leave the company. • Renew certificates as often as practically possible (at least yearly would be good),

	preferably using a freshly-generated private key each time.
Hostname	Make sure all hostnames are covered as part of the certificate.
Certificate Chains	Install Complete Certificate Chains
SSL/TLS Protocols	Use Current SSL/TLS Protocols (TLS 1.2 or 1.3)
Cipher Suites	Use a Short List of Secure Cipher Suites
Forward Secrecy	Use Forward Secrecy: prefer ECDHE suites in order to enable forward secrecy with modern web browsers. To support a wider range of clients, use DHE suites as a fallback after ECDHE.

IP Reputation

The most common reason for elevated IP risk scores is due to previous abusive behavior from the IP address. This could include sending SPAM, compromised devices, or any form of suspicious behavior.

Email Bounce Rate	Keep the email bounce rate low
Spam Keywords	Avoid any spammy words or phrases that would trigger a red flag for an ISP or spam filter.
Dedicated IPs	• Use dedicated IPs over shared IPs

- Protect Your Dedicated IP: Be sure to put safety measures in place, so your IP address isn't compromised by a cybercriminal. Limit IP access to people you trust, and consider using two-factor authentication for logging in.

DNS Configurations

It is essential to check your Domain DNS Health every once in a while after editing your DNS parameters to ensure your changes are up to the mark and are following the standards.

SPF Record	Generate the SPF record using an online tool and include all the IPs that are going to send emails or follow the best practices/configuration instructions given by the Email Provider
DMARC Record	Follow the best practices/configuration instructions given by the Email Provider
DKIM Record	Follow the best practices/configuration instructions given by the Email Provider
Zone Transfer	Configure the DNS server to only accept zone transfers from trustworthy IP addresses
DNSSEC	Configure the DNS to comply with DNSSEC

Open Ports

It is essential that all open ports be identified and secured using proactive techniques.

Inactive Ports	Identify and close any port not actively needed
Access Control Lists	Restrict port access to specific source IP addresses (or ranges)
Least Privilege	Implement the principle of least privilege on all endpoints
Restrict Direct Access	Don’t allow anyone direct access to highly privileged accounts
Information Exposure	Reduce the exposed information on open ports such as server version, components used, etc.
Outdated Protocols	Do not use outdated protocols such as FTP (Port 20 and 21), Telnet (Port 23)
Firewalls	Install firewalls on every host and patch firewalls regularly
VPN for sensitive ports	Access sensitive ports using a secure virtual private network (VPN) such as SSH - 22, RDP – 3389, etc.
Secure Protocols	Use only secure protocols such as SSH, SFTP, TLS, etc.

Corporate Emails

2FA	Mandate Two-factor Authentication (2FA) for all corporate email users.

Password Policy	Set a strong password policy
Cybersecurity Awareness	Train the email users in Cybersecurity Awareness
Email Protection	Implement Email Security Solutions to prevent malware and phishing
Patching	
Inventory	Develop an up-to-date inventory of all your production and non-production systems
Patch Management Policy	Create a Patch Management Policy covering inventory, frequency of patching, etc.
Apply Patches Quickly	Ensure that any patches that are needed for your software/OS are applied in a timely manner.
Service Configurations	
Application Server - HTTPS	• Eliminate Mixed Content: JavaScript files, images, and CSS files should all be accessed with SSL/TLS. • Use Secure Cookies: Setting the *Secure* flag in cookies will enforce transmission over secure channels (e.g. HTTPS). You can also keep client-side JavaScript from accessing cookies via the *HttpOnly* flag,

	and restrict cross-site use of cookies with the *SameSite* flag. • Deploy HTTP Strict Transport Security (HSTS) • Deploy Content Security Policy
Application Server – Security Headers	Implement all the necessary security headers in the application server.
Information Exposure	Reduce the exposed information on running services such as server version, components used, etc.
Vendor Best Practices	Use best practices guides give by the vendor for secure configuration of the assets

The Protect section is an essential aspect of the Attack Surface Management guideline document as it outlines the proactive measures that can be taken to minimize the risk of external cyber-attacks. Based on the research data, it has been inferred that the top three attack vectors based on impact and ease to fix are Service Misconfiguration, SSL Health, and DNS Configuration. Hence, these three should be given the highest priority in implementing proactive controls which can give maximum risk reduction with minimal effort. In conclusion, the Protect section outlines the baseline steps that organizations must take to implement proactive controls and minimize the risk of external cyber-attacks.

ASSESS – Detect the vulnerabilities, misconfigurations and other risks in the attack surface

Assess the vulnerabilities and risks present in each element of your attack surface. This may involve conducting regular assessments and penetration testing, as well as analyzing data from security tools and incident reports.

Table 5.5

Guidelines for Assess Phase

SSL Configurations	
Discovering Domains and Sub-domain helps to broader the attack surface, find hidden applications, and forgotten subdomains.	
Certificate Expiry	Test for SSL certificate expiration for enumerated subdomains.
SSL/TLS vulnerabilities	Test for the most recent SSL/TLS vulnerabilities and weaknesses;
Private Keys	Test for RSA/ECDSA key length
Compliance Requirement	Test for compliance with applicable standards
HTTP Content	Test for insecure external content (HTTP)
Self Signed Certificate	Check for self-signed certificate
Weak Ciphers	Test for weak ciphers
Certificate Chains	Test for invalid certificate chains

IP Reputation

All internet activity is linked to an IP address or a set of IP addresses that work as a network. If a given network or IP address exhibits suspicious behavior, ISPs could label the entire network's IP reputation as poor.

Malware Monitoring	Scan all the servers for malware infections frequently
Blacklist Monitoring	Perform IP reputation checks using multiple online tools

DNS Configurations

It is essential to check your Domain DNS Health every once in a while after editing your DNS parameters to ensure your changes are up to the mark and are following the standards

SPF Record	Test for SPF record misconfigurations
DMARC Record	Test for DMARC record misconfigurations
DKIM Record	Test for DKIM record misconfiguration
Zone Transfer	Check for Zone Transfer Vulnerability
DNSSEC	Use an online tool to test whether a domain is compliant with DNSSEC or not
Recursive DNS Resolver Test	Detect if IP or domain is vulnerable to DNS amplification attacks.

Open Ports

Outdated Ports	Check for the following outdated ports • FTP (Port 20 and 21)

	• Telnet (Port 23)
Public Access	Check for public accessibility of database, SSH, etc.
Corporate Emails	
Third-Party Breaches	Check for corporate email leaks in third-party breaches using available tools • HIBP • Firefox Monitor • DeHashed • LeakCheck
Pastes	Check for corporate email leaks in pastes. • HIBP • Pastebin • Throwbin • Anonfile
Site Reputation	
Website Reputation	Check the reputation of the website using available tools • Google Safe Browsing • VirusTotal • URLScan

Patching		
	Outdated Components	Test for servers that are running outdated components
	Missing Patches	Test for servers that are having patches missing
Service Configurations		
	HSTS	Check whether the application is allowing only HTTPS connection
	CORS	Test all the servers for CORS misconfiguration
	HTTP Methods	Test for excessive HTTP methods such as HTTP TRACE.
	Security Headers	Test for Security Headers in the HTTP Response
	Cookie Attribute	Test for Secure and HTTPOnly cookie attributes
	Information Exposure	Test for server information exposures like version disclosure, stack disclosure, etc.
	Cache-Control	Check for sufficient Cache Control mechanisms
Active Scan		
	Deep Vulnerability Scan	Perform a thorough security scan for the critical assets to uncover all the vulnerabilities

The Assess section of the guideline document focuses on baseline guidelines for detecting potential risks and vulnerabilities within an organization's attack surface. The research conducted for the guideline document has provided insight into the different external attack vectors and threats and potential impacts that helped in arriving at this baseline. To assess the risks, the guideline document suggests covering all the aspects mentioned above either manually or using an automated ASM tool. The use of Automated ASM tools like Threat Meter and regular monitoring of various attack vectors of the organizational attack surface including the above baseline help in detecting potential risks in a timely manner. The research findings have emphasized the importance of regularly assessing the attack surface and updating security measures to mitigate potential threats.

PRIORITIZE - Risk-based prioritization of attack surface findings

Prioritizing vulnerabilities in an attack surface can help organizations focus their resources on the most critical issues and reduce the overall risk to their systems and data. This includes ranking the vulnerabilities based on their risk level and potential impact. This phase also includes identifying which vulnerabilities need to be addressed first and which can be addressed later. This allows organizations to effectively plan and allocate resources for vulnerability management.

Table 5.6

Guidelines for Prioritize Phase

Prioritizing Findings	
Accurate vulnerability prioritization helps you avoid unnecessary work on fixing security issues that do not matter and focus instead on risk items which are likely to have a bigger business impact.	
Unified View	Create a unified view of all the identified vulnerabilities
Use Risk Calculation Standards	Use risk calculation industry standard for prioritizing • Common Vulnerability Scoring System (CVSS) • OWASP Risk Rating Methodology • CISA- Stakeholder-Specific Vulnerability Categorization(SSVC)
Context-based Risk Calculation	Include the likelihood of the vulnerability, classification of the asset and exposure time of the vulnerability for arriving at the priority
Document and Track	Document and track reasons for risk exceptions and revisit and review periodically

The table of prioritizing guidelines in this document is based on research conducted on external attack vectors. One of the main objectives of the research is to provide guidelines for minimal efforts with maximum risk reduction. Prioritizing vulnerabilities in an attack surface is important in order to focus resources on the most critical issues and reduce overall risk. The section starts by highlighting the importance of ranking vulnerabilities based on their risk level and potential impact. A table is then provided that outlines the prioritizing guidelines, including creating a unified view of all identified risks, using risk calculation standards, context-based risk calculation, and documenting and tracking risk exceptions. This can be taken as the risk prioritization baseline for organizations to effectively plan and allocate resources for vulnerability management, ensuring maximum risk reduction with minimal effort.

REMEDIATE – Act on the attack surface findings before hacker.

Implement measures to mitigate identified vulnerabilities and reduce the risk of a successful cyber attack. This may include patching software, implementing security controls, and training employees on best practices. As an outcome of the research, a priority matrix for implementing remediation has arrived. The priority matrix was based on the effort required to implement the remediation and the maximum risk reduction achieved. The research found that service misconfiguration was the top priority for implementing remediation, followed by SSL health and DNS Misconfiguration. The research also found that the priority required for each external attack vector will vary depending on the organization's level of risk.

Priority matrix for implementing remediation (with less effort for maximum risk reduction)

The priority matrix for implementing remediation of identified attack surface risks was arrived at by considering two main parameters - the complexity of fixing the risk and the need for maximum risk reduction with less effort. We analyzed the data from two separate research studies, each with a sample size of 200 and 1000, to determine the most afflictive attack vectors and threats and their monetary impact. Based on this data, we created a priority matrix that ranked the attack vectors and threats in order of importance, with the least complex and costly risks being at the top of the list. This matrix will help organizations prioritize their remediation efforts and reduce the risk of external cyber-attacks effectively.

Table 5.7

Reduction of Threat Score in % when Attack Vector is Removed from Threat Score Calculation and Complexity to Fix each Attack Vector

S. No	*Attack Vector*	*Total of Individual Weight*	*Average of Individual Weight*	*Threat Score Reduced after Remediation % (AVG)*	*Complexity for Fixing each Attack Vector*	*Priority*
1	Service Misconfiguration	91.5	5.08	53.78%	2	1

2	SSL Health	27.2	4.53	11.57%	3	2
3	IP Reputation	86.0	8.60	8.77%	6	3
4	Unnecessary Open Ports	46.9	7.82	6.83%	4	4
5	Outdated Version	19.0	9.50	5.59%	5	5
6	Data Leaks	20.0	10.00	8.08%	7	6
7	Data Breaches	0.0	0.00	0.00%	8	7
8	DNS Misconfiguration	21.2	5.30	5.40%	1	8

Organizations can implement the below guidelines as the baseline for remediating the attack surface risks.

Table 5.8

Guidelines for Remediate Phase

SSL Configurations		
	Host Name Not Listed	Fix the Server Hostname in the host file to match with the hostname mentioned in the Certificate.
	Client-Initiated Secure Renegotiation Enabled	Disable SSL/TLS client-initiated renegotiation in the server SSL configuration.
	Invalid Certificate Chain	Download the intermediate CA certificates from the CA website and include them in the server SSL configuration.

	RSA Key Smaller Than 2048 Bits	Migrate to 2048-bit key length.
	Heartbleed Attack	Upgrade the OpenSSL version to latest stable version.
	Weak Cipher Suites Enabled	Use https://ssl-config.mozilla.org/ tool for configuring the Cipher Suites.
	Certificate Expired	Contact your Certificate Authority to renew the SSL certificate.
	CRIME	Disable compression and/or SPDY service.
	Insecure SSL/TLS Protocols	Use Current SSL/TLS Protocols (TLS 1.2 or 1.3)
IP Reputation		
	Delisting	Navigate to the blacklisted sites that have your IP address on them, and follow the steps given by them to delist the IP.
	Forensic Investigation	Check the listed servers for malware infections
DNS Configurations		
	SPF Record Misconfiguration	Generate the SPF record using an online tool and include all the IPs that are going to send emails or follow the best practices/configuration instructions given by the Email Provider

	DMARC Record Misconfiguration	Follow the best practices/configuration instructions given by the Email Provider
	DKIM Record Misconfiguration	Follow the best practices/configuration instructions given by the Email Provider
	Zone Transfer Vulnerability	Configure the DNS server to only accept zone transfers from trustworthy IP addresses
	DNSSEC Misconfiguration	Configure the DNS to comply with DNSSEC
Open Ports		
	Unnecessary Open Ports	Close the port that are not actively needed
	Publicly Accessible Services (Databases, SSH, etc)	Restrict port access to specific source IP addresses (or ranges)
	Information Exposure	Reduce the exposed information on open ports such as server version, components used, etc.
	Outdated Protocols	Do not use outdated protocols such as FTP (Port 20 and 21), Telnet (Port 23). Use only secure protocols such as SSH, SFTP, TLS, etc.
Corporate Emails		
	Data Leaked in Third Party Breaches	• Enforce a password change for the mail and other corporate accounts. • Run a training or awareness session for employees, and, tell them about your digital platforms usage policy.

		• Make employees aware of the dangers of using the corporate ID for outside registration, and, the consequences.
Patching		
	Outdated Component Usage	Upgrade the component to the latest stable version available with the vendor.
	Missing Patches	Follow the instruction given by the vendor for installing missing patches
Service Configurations		
	Cross Origin Resource Sharing Misconfiguration	• Never set Access-Control-Allow-Origin header as "*" • With Access-Control-Allow-Methods you should specify exactly what methods are valid for approved domains to use. Some may only need to view resources, while others need to read and update them, and so on. • Request credentials from requestors by setting up the header Access-Control-Allow-Credentials.
	Excessive HTTP Methods Enabled	Disable the excessive HTTP method enabled in the application server.

Information Exposure	Reduce the exposed information on running services such as server version, components used, etc.
Security Headers Missing	Enable all the required security headers in the application server response
Cookie Attributes Missing	Enable secure and httpOnly cookie attribute.
Cache Control Missing	Enable Cache-control headers in the application server response

The Remediation section of this guideline document highlights the steps that organizations need to take to address and mitigate the risks identified in the prioritization phase. The inference from the research is included in this section in the form of a table, outlining the baseline remediation steps for effective remediation risk identified in the SSL Health, IP Reputation, and other categories. The research findings have helped in developing these guidelines and ensuring that organizations have a structured approach to remediation, reducing their overall risk and improving their overall security posture.

In conclusion, this guideline document provides a baseline for protecting from the common attack vectors and threats of external attack surface. It highlights the priority matrix for implementing proactive controls and remediating attacks, and offers tips and tricks for attack surface asset discovery and reduction. However, it should be noted that these findings are based on limited research and organizations should customize these guidelines to best fit their specific attack surface.

Disclaimer: This guideline document is based on the results obtained from limited research and should be used as a baseline reference only. The information provided in this document is not intended to be a comprehensive or definitive guide to attack surface management. The findings and recommendations are subject to change and may vary based on the complexity of the organization's attack surface and risk tolerance. Each organization should tailor its attack surface management program to its specific needs and constraints by taking this as a baseline. This document does not guarantee the security of an organization's assets or systems, and it is the organization's responsibility to discover, reduce, protect, assess, prioritize, remediate, and continuously monitor its attack surface to ensure the protection of its assets.

CHAPTER VI

IMPLICATIONS AND RECOMMENDATIONS

6.1 Implications

The research on attack surface monitoring and data analysis aimed to create a guideline document for organizations to effectively protect against external cyber-attacks. The research answered several key questions to provide a comprehensive understanding of the most and least common attack vectors and threats, as well as the monetary impact and priority matrices for implementing proactive controls and remediation.

The research and guideline document offer valuable insights and recommendations for organizations seeking to enhance their attack surface monitoring efforts. By prioritizing the resolution of service misconfigurations and SSL health issues, regularly reviewing and updating security configurations, and establishing a response plan, organizations can fortify themselves against external cyber threats.

Although the research establishes a solid foundation for managing the attack surface, it's essential to recognize that a comprehensive attack surface management program requires customization to meet an organization's unique requirements. This involves tailoring the guideline document to the specific infrastructure, applications, and threat landscape of the organization. Furthermore, organizations must continuously monitor and update their guideline document to adapt to the evolving threat landscape.

As a result, organizations must regularly evaluate and adjust their attack surface management program to ensure they're adequately addressing their distinct attack surface.

In conclusion, this guideline document provides a baseline for protecting from the common attack vectors and threats of external attack surface. It highlights the priority matrix for implementing proactive controls and remediating attacks and offers tips and tricks for attack surface asset discovery and reduction. However, it should be noted that these findings are based on limited research and organizations should customize these guidelines to best fit their specific attack surface.

6.2 Recommendations for Future Research

The survey results have yielded multiple suggestions for future research in the field of attack surface management. One potential avenue for exploration is to target specific industries when collecting samples and designing industry-specific frameworks. This approach could aid in comprehending the distinct challenges faced by various sectors and developing tailored solutions to meet their particular needs.

Another area of inquiry is the exploration of diverse tool sets to enhance the effectiveness of attack surface management. The survey emphasized the requirement for automated tools for asset discovery; thus, future research can focus on identifying and testing different tools to assist in discovering and maintaining external assets.

Future studies can explore larger sample sizes to increase generalizability, to develop a more comprehensive understanding of the attack surface management landscape and identify trends relevant across different organizations.

Conducting a detailed investigation of individual threat vectors can provide a more effective understanding of the threats faced by organizations. This analysis could help identify the most significant threats and design targeted strategies to address them. Using a consistent data sampling method across multiple industries and datasets can greatly facilitate the development of machine learning models capable of forecasting future trends and attack vectors.

Lastly, future research could concentrate on remediating individual threat vectors, which may involve identifying the most effective remediation strategies for different types of threats and evaluating their impact on the organization's overall security posture. It may also involve identifying barriers to effective remediation and developing strategies to overcome them.

Taken together, these recommendations can improve the effectiveness of attack surface management strategies and enable the development of more robust frameworks for addressing the constantly evolving threat landscape.

6.3 Conclusion

In conclusion, this research provides a comprehensive understanding of the importance of attack surface monitoring and reduction in protecting organizations from cyber-attacks. The attack surface management plays a vital role in identifying potential vulnerabilities and reducing the risk of cyber-attacks. Through the research conducted on attack surface management, it is evident that service misconfigurations (66.82%), SSL health (14.34%), and IP reputation (7%) are the most common attack vectors. Moreover, phishing threats (19.75%), data leaks (23.98%), and rogue mobile apps (35.83%) are the top threat vectors.

Furthermore, the six-phase guideline document provides easy-to-implement guidelines for preventing and remediating attacks from external threat vectors, along with the frequency of monitoring required for each external attack vector. It is essential to note that the guidelines provided are a baseline and organizations need to customize them according to their attack surface.

Finally, organizations must recognize the significance of protecting their attack surface and implementing preventive and remedial measures to reduce the risk of cyber-attacks. By following the guideline document and implementing the proposed framework, organizations can improve their cybersecurity posture and safeguard their assets, customers, and reputation.

REFERENCES

(no date) *NumPy*. Available at: https://numpy.org/ (Accessed: December 23, 2022).

Admin (2021) *Sampling methods (techniques) - types of sampling methods and examples*, *BYJUS*. BYJU'S. Available at: https://byjus.com/maths/sampling-methods/. (Accessed: December 23, 2022).

Albanese, M., Battista E., Jajodia, S., Casola, V. (2014) 'Manipulating the Attacker's View of a System's Attack Surface', *Institute of Electrical and Electronics Engineers.* Available at: https://ieeexplore.ieee.org/document/6997517 (Accessed: 8 August 2022)

Aljuhami, M., A., Bamasoud, M., D. (2021) 'Cyber Threat Intelligence in Risk Management – A Survey of the Impact of Cyber Threat Intelligence on Saudi Higher Education Risk Management', *International Journal of Advanced Computer Science and Applications.* Available at: https://thesai.org/Downloads/Volume12No10/Paper_18-Cyber_Threat_Intelligence_in_Risk_Management.pdf (Accessed: 8 August 2022)

Alshehhi, S. (2020) 'Developing a Framework for Measuring Organizational Cyber Resilience Against External and Internal Cyber Threats', *Theses for Project Management – Faculty of Business & Law, BSpace.* Available at: https://bspace.buid.ac.ae/handle/1234/1777 (Accessed: 9 May 2022)

Basheer, R. S., & Alkhatib, B. (2021). Threats from the Dark: A Review over Dark Web Investigation Research for Cyber Threat Intelligence. *Journal of Computer Networks and Communications*, *2021*, 1–21. https://doi.org/10.1155/2021/1302999

Censys (2020) What You Don't Know Will Hurt You – How Attack Surface Management can supercharge your Vulnerability Management Program, Available at: https://f.hubspotusercontent00.net/hubfs/5851803/VM_ASM%20Whitepaper%20FINAL.pdf (Accessed: 8 August 2022)

Cost of a data breach 2022 (no date) *IBM*. Available at: https://www.ibm.com/reports/data-breach (Accessed: December 23, 2022).

Deloitte (2021) Role of Cybersecurity in M&A, Available at: https://www2.deloitte.com/content/dam/Deloitte/in/Documents/risk/in-ra-cybersecurity-for-mergers-and-acquisitions-noexp.pdf (Accessed: 10 August 2022)

Donner, H. (2019) New Black Hat USA Research: Your Private Information Is Already Available to Criminals; U.S. Elections, Critical Infrastructure Also at Risk, Available at: https://www.globenewswire.com/news-release/2019/07/01/1876681/0/en/New-Black-Hat-USA-Research-Your-Private-Information-Is-Already-Available-to-Criminals-U-S-Elections-Critical-Infrastructure-Also-at-Risk.html (Accessed: 9th May 2022)

Ellison, J., R., Goodenough, B., J., Weinstock, B., C., Woody, C. (2010) 'Evaluating and Mitigating Software Supply Chain Security Risks', *Software Engineering Institute.*

Available at: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=9337 (Accessed: 10 August 2022)

Fachkha, C. (2016) 'Security Monitoring of Cyber Space', *Cornell University.* Available at: https://arxiv.org/abs/1608.01468 (Accessed: 8 August 2022)

Fuentes-García, M., Camacho, J., & Maciá-Fernández, G. (2021). Present and Future of Network Security Monitoring. *IEEE Access*, *9*, 112744–112760. https://doi.org/10.1109/access.2021.3067106

Goswami, S., Krishnan, R., N., Verma, M., Swarnkar, S., Mahajan, P. 'Reducing Attack Surface of a Web Application by Open Web Application Security Project Compliance', *Department of Management Information System & Technologies (MIST), DRDO, New Delhi.* Available at: https://publications.drdo.gov.in/ojs/index.php/dsj/article/view/1291 (Accessed: 9 August 2022)

Home (no date) *Threat Meter*. Available at: https://sumeruthreatmeter.com/ (Accessed: December 23, 2022).

IBM (2020) Cost of a Data Breach Report 2020, Available at: https://www.ibm.com/security/digital-assets/cost-data-breach-report/1Cost%20of%20a%20Data%20Breach%20Report%202020.pdf (Accessed: 9th May 2022)

Jang-Jaccard, J., & Nepal, S. (2014). A survey of emerging threats in cybersecurity. *Journal of Computer and System Sciences*, *80*(5), 973–993. https://doi.org/10.1016/j.jcss.2014.02.005

Jelen, S. (2021) Attack Surface Monitoring: Definitions, Benefits, and Best Practices, Available at: https://securitytrails.com/blog/attack-surface-monitoring (Accessed: 9th May 2022)

King, A. (n.d.). *Mozilla SSL Configuration Generator*. https://ssl-config.mozilla.org/

Kok, A., Mestric I., I., Valiyev, G., Street, M. (2020) 'Cyber Threat Prediction with Machine Learning', *PfP Consortium of Defense Academies and Security Studies Institutes.* Available at: http://connections-qj.org/article/cyber-threat-prediction-machine-learning (Accessed: 8 August 2022)

Liebl, S., Lathrop, L., Raithel, U., Aßmuth, A., Ferguson, I., Sollner, M. (2021) 'Analyzing the attack surface and threats of industrial Internet of Things devices', *Technical University of Applied Sciences OTH Amberg-Weiden.* Available at: https://rke.abertay.ac.uk/ws/portalfiles/portal/35554088/Liebl_AnalyzingTheAttack_Published_2021.pdf (Accessed: 9 May 2022)

Microsoft (2022) Anatomy of an external attack surface: Five elements organizations should monitor, Available at: https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE4VjO9 (Accessed: 9th May 2022)

Mishra, P., Pandey, M., C., Singh, U., Keshri, A., Sabaretnam, M., 'Selection of Appropriate Statistical Methods for Data Analysis', *PubMed Central,* Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6639881/#:~:text=Two%20main%20statis tical%20methods%20are,such%20as%20student%27s%20t%2Dtest. (Accessed: 1 September 2022)

Oriola, O. (2018) 'A Cyber-Threat Intelligence Framework for Improved Internet Facilitated Organized Crime Threat Management', *International Journal of Computer Trends and Technology.* Available at: https://www.ijcttjournal.org/archives/ijctt-v60p101 (Accessed: 8 August 2022)

Otuoze, A. O., Mustafa, M. W., & Larik, R. M. (2018). Smart grids security challenges: Classification by sources of threats. *Journal of Electrical Systems and Information Technology*, *5*(3), 468–483. https://doi.org/10.1016/j.jesit.2018.01.001

PaloAlto (2021) 2021 Cortext Xpanse Attack Surface Threat Report, Available at: https://start.paloaltonetworks.com/asm-report (Accessed: 8 August 2022)

Pandas (no date) *pandas*. Available at: https://pandas.pydata.org/ (Accessed: December 23, 2022).

Randori (2022) *The State of Attack Surface Management 2022.* Available at: https://info.randori.com/hubfs/State%20of%20ASM%20Report.pdf (Accessed: 7 August 2022)

Rimol, M. (2022) Gartner Identifies Top Security and Risk Management Trends for 2022, Available at: https://www.gartner.com/en/newsroom/press-releases/2022-03-07-gartner-identifies-top-security-and-risk-management-trends-for-2022 (Accessed: 9th May 2022)

Singh, T. (2012) 'Emerging Challenges to Cyber Security – Internet Monitoring with Specific reference to National Security, *International Journal of Scientific Research.* Available at: https://www.worldwidejournals.com/international-journal-of-scientific-research-(IJSR)/article/emerging-challenges-to-cyber-securityandndash-internet-monitoring-with-specific-reference-to-national-security/NzY=/?is=1&b1=173&k=44 (Accessed: 8 August 2022)

Spearman's rank-order correlation (no date) *Spearman's Rank-Order Correlation - A guide to when to use it, what it does and what the assumptions are.* Available at: https://statistics.laerd.com/statistical-guides/spearmans-rank-order-correlation-statistical-guide.php (Accessed: December 23, 2022).

SSL Corp. (2021, February 25). *SSL/TLS Best Practices for 2021 - SSL.com*. SSL.com. https://www.ssl.com/guide/ssl-best-practices/

Stewart, M. (2020) *Guide to classification on Imbalanced Datasets*, *Medium*. Towards Data Science. Available at: https://towardsdatascience.com/guide-to-classification-on-imbalanced-datasets-d6653aa5fa23 (Accessed: December 23, 2022).

Subdomain Enumeration: The Ultimate Guide. (2022, July 3). 0xffsec Handbook. https://0xffsec.com/handbook/information-gathering/subdomain-enumeration/

Sumeru (2022) *Threat Meter*. Available at: https://sumeruthreatmeter.com/ (Accessed: 10 August 2022)

Szefer, J., Keller, E., Lee, R. B., & Rexford, J. (2011). Eliminating the hypervisor attack surface for a more secure cloud. *Computer and Communications Security*. https://doi.org/10.1145/2046707.2046754

Thiesen, C., Munaiah, N., Al-Zyoud, M., Carver, J. C., Meneely, A., Williams, L. (2018) 'Attack Surface Definition: A Systematic Literature Review', *North Carolina State University.* Available at: https://www.sciencedirect.com/science/article/abs/pii/S0950584918301514 (Accessed: 8 August 2022)

Understanding your organization's attack surface and why it poses a risk - Darktrace Blog. (n.d.). https://de.darktrace.com/blog/understanding-your-organizations-attack-surface-and-why-it-poses-a-risk

Unit 3 - Linear relationships (no date) *Mr. Scott's Math Class*. Available at: https://mrscottmathclass.weebly.com/unit-3---linear-relationships.html (Accessed: December 23, 2022).

VentureBeat (2022) *Trend Micro launches new attack surface management platform.* Available at: https://venturebeat.com/2022/04/25/trend-micro-launches-new-attack-surface-management-platform/ (Accessed: 7 August 2022)

Verizon (2021) 2021 Data Breach Investigation Report, Available at: https://www.verizon.com/business/resources/reports/dbir/ (Accessed: 9th May 2022)

Verizon (2019) 2019 Data Breach Investigation Report, Available at: https://www.verizon.com/business/resources/reports/dbir/2019/results-and-analysis/ (Accessed: 12th May 2022)

Visualization with python (no date) *Matplotlib*. Available at: https://matplotlib.org/ (Accessed: December 23, 2022).

What is exploratory data analysis? (no date) *IBM*. Available at: https://www.ibm.com/cloud/learn/exploratory-data-analysis (Accessed: December 23, 2022).

APPENDIX A

GLOSSARY

- **Attack Vectors**

  An attack vector is the actual act of exploiting the information security system's weaknesses.

- **Brand Impersonations**

  Threat Meter protects the brands and helps from the fallout of reputation damage by identifying brand impersonation threats like Unofficial Social Media Profiles, Impersonating Domains, Impersonating Mobile Apps, Cloned VIP Profiles, etc.

- **Combination**

  Combination is defined as grouping or selection of ‘r’ things that can be formed out of given total of ‘n’ objects or things. The number of arrangements is denoted by ‘nCr’ which is equal below equation:

  $n!/(r!(n-r)!\ )$ Combination Formula

- **Data Breaches**

  Breaches are publicly disclosed events of unauthorized access, often involving data loss or theft. These events are graded based on several factors, including the number of data records lost or exposed.

- **Data Leaks**

  Threat Meter gives all the visibility needed to detect sensitive data exposed over the darknet by employees, contractors, or third parties in 100+ dark web & internet sources.

It covers Source Code leaks, Employee Emails & Credentials, API/DB Credentials, Intellectual Properties, Customer Data, etc.

- **Descriptive Statistics**

  In Descriptive statistics, we get the inference of central tendency in the data set which measures Mean, mode, median, standard deviation, Skewness, and Kurtosis.

- **DNS Health**

  The DNS Health includes checking which DNS parameters that need attention and also those who follow DNS standards. Altogether it includes DNS health test, MX record test, Mail (MX), DMARC test, SMTP test for mail records, and SPF records test.

- **Fail Ratio**

  Fail Ratio range from 0% to 100% and indicate the percentage of failed test from the total number of tests performed.

- **IP Reputation**

  IP Reputation scans identify spam propagation events that are observed when devices on a company's network are sending unsolicited commercial or bulk emails. This type of activity can damage a company's reputation and cause legitimate company emails to be caught in spam filters.

- **Kurtosis**

  - Kurtosis is a measure of the "tailedness" of the probability distribution of a real-valued random variable.

- When the excess kurtosis is around 0, or the kurtosis equals is around 3, the tails' kurtosis level is like the normal distribution.
- A kurtosis ‘greater than three’ will indicate **positive kurtosis**. The value of kurtosis will range from ‘1 to infinity’. Further, a kurtosis ‘less than three’ will indicate a ‘**negative kurtosis**’. The range of values for a negative kurtosis is from ‘-2 to infinity’.
- Kurtosis describes a particular aspect of a probability distribution.

Type of Kurtosis	Kurtosis	Excess Kurtosis
Leptokurtic	>3	>0
Platykurtic	<3	<0
Mesokurtic	=3	=0

- **Linear Relationship**
  - A linear relationship or correlation is a statistical expression that occurs when two variables satisfy the mathematical formula y = mx + b.
  - Relationship between a scalar response and one or more explanatory variables.
  - Relationship where multiple correlated dependent variables are predicted, rather than a single scalar variable.
  - A linear relationship (or linear association) is a statistical term used to describe a straight-line relationship between two variables.

- Linear relationships can be expressed either in a graphical format where the variable and the constant are connected via a straight line or in a mathematical format where the independent variable is multiplied by the slope coefficient, and added by a constant, which determines the dependent variable.
  - 1 indicates a strong positive relationship.
  - -1 indicates a strong negative relationship.
  - Result of zero indicates no relationship at all.

- **Normal Distribution**
  - Normal distribution, also known as the Gaussian distribution, is a probability distribution that is symmetric about the mean, showing that data near the mean are more frequent in occurrence than data far from the mean.
  - The normal distribution is the proper term for a probability bell curve.
  - In a normal distribution the mean is zero and the standard deviation is 1. It has zero skew and a kurtosis of 3.
  - All normal distributions can be described by just two parameters: the mean and the standard deviation.

- **Outdated Components**

Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. This scan helps to track security holes created by server software that is no longer supported by its original developers or has become out-of-date (deprecated).

- **Permutation**

  The permutation is defined as an arrangement of 'r' things that can be done out of a given total of 'n' things. The number of arrangements is denoted by 'nPr' which is equal to as shown in the below equation:

  n!/(n-r)! Permutation Formula

- **Phishing Threats**

  This scan identifies the initial phase of the phishing attack by detecting the following threats and protecting the end customer: possible typosquatting domains, registered typosquatting domains, phishing pages & domains, phishing email servers, etc.

- **Public Data Leaks**

  Public Data Leaks scans across multiple data breaches and phishing password dumps to see if the employee email address has been compromised.

- **Rogue Apps**

  This scan discovers fraudulent mobile apps that are leveraging customer brands to infect end users or steal credentials.

- **Service Misconfigurations**

  Security misconfiguration can happen at any level of an application stack, including the network services, platform, web server, application server, database, frameworks, custom code, and pre-installed virtual machines, containers, or storage. Attackers will often attempt to exploit unpatched flaws or access default accounts, unused pages, unprotected files, and directories, etc., to gain unauthorized access or knowledge of the system.

- **Site Reputation**

  As more websites are created, organizations need finely tuned security to protect their users from malicious sites. This scan discovers unsafe sites which are legitimate websites but have been compromised.

- **Skewness**
  - Skewness is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean.
  - The skewness value can be positive, zero, negative, or undefined.
    - Negative skew commonly indicates that the tail is on the left side of the distribution, and positive skew indicates that the tail is on the right.
    - The variables which fall under skewness between -2 to -1 have **moderate left skewness.**
    - The variables which fall under Skewness between 1 to 2 have **moderate right skewness.**
    - The variables which have Skewness greater than or equal to 2 then they have **severe right skewness.**
    - The variables which fall under Skewness between -1 to 1 have a normal distribution.

- **Spearman's Correlation** 'ρ'

  The Spearman's rank correlation coefficient (ρ) is a measure of the monotonic correlation between two variables and is, therefore, better at detecting nonlinear monotonic correlations than Pearson's 'r'. Its value lies between '-1' and '+1'. -1 indicating total negative monotonic correlation, 0 indicating no monotonic correlation, and 1 indicating total positive monotonic correlation. To calculate ‘ρ’ for two variables ‘x’ and ‘y’, one divides the covariance of the rank variables of ‘x’ and ‘y’ by the product of their standard deviations.

- **SSL Health**

  SSL Health scans evaluates TLS/SSL certificates, which includes the strength of their cryptographic keys. Certificates are responsible for verifying the authenticity of company servers to their associates, clients, and guests, and serve as the basis for establishing cryptographic trust.

- **Threat Score**

  Threat score provides a means for monitoring the security hygiene of organizations and determining whether their security posture is improving or declining over time. The organizations with lower threat scores have a more robust security posture and have the lowest risk. (0-25 low-risk with good security, 26-75 a medium risk with medium security, and 75+ an elevated risk with bad security)

- **Threat Vector**

  A threat vector is something that can gain access to, harm, or eliminate an asset by exploiting a vulnerability.

- **Unnecessary Open Ports**

  Unnecessary open ports on a server are security vulnerabilities that can potentially allow a hacker to exploit services on your network. Open ports scan shows which port numbers and services are exposed to the internet. Certain ports must be open to support normal business functions; however, unnecessary open ports provide ways for attackers to access a company's network.

APPENDIX B

CODE BASE

(I)

- **Library Importing:**

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from sklearn.utils import resample
```

- **Data Loading:**

```
df = pd.read_excel('Data/threatmeter_1000_with_industries.xlsx')
df.head()
```

- **Industry wise Correlation plot and data distribution**

```
sns.pairplot(df,hue='Industry_Name',diag_kind="hist",corner=True)
```

- **Data Sampling Approaches:**

Up sample Data.

```
def factorial(n):
    if n==1:return 1
    else: return n * factorial(n-1)
def permutation_without_repetition(n,r):
    return (factorial(n)/(factorial(n-r)))
```

```
        def permutation_with_repetition(n,r):
            return n ** r
        def combinations_without_repetition(n,r):
            return (factorial(n)/(factorial(r)*(factorial(n-r))))
        def combinations_with_repetition(n,r):
            return ((factorial(n+r-1))/(factorial(r)*(factorial(n-1))))
from sklearn.utils import resample
from imblearn.over_sampling import SMOTE
import pandas as pd
import numpy as np


def upsample_classes(data, target):
    """
        Input is data and what the target feature from that data is
        Output is a balanced dataset
            - First, we make a list of unique labels in data
            - Next, we split up the rows of data by their labels into different sets of data
            - Next, we search for the majority class label
            - Next, we get the classes back together using pandas.concat (more on this function can
be found at the documentation) and separate off the majority class based on it's newly found
label
```

- Next, we remove the majority class and upsample the other classes to match the length of the majority class

- Finally, we combine the majority class with the other classes, which are now of equal length

```
    """

    lst = list(data[target].unique())

    classes = []
    for c in lst:
        classes.append(data[data[target]==c])
    length = 0
    class_lab = None
    for c in classes:
        if len(c)>length:
            length=len(c)
            class_lab = c
    class_lab = class_lab[target].unique()[0]

    regroup = pd.concat(classes)
    maj_class = regroup[regroup[target]==class_lab]
```

```
    lst.remove(class_lab)

    new_classes=[]
    for i in lst:
        new_classes.append(resample(data[data[target]==i],replace=True,
n_samples=len(maj_class)))

    minority_classes = pd.concat(new_classes)
    upsample = pd.concat([regroup[regroup[target]==class_lab],minority_classes])

    return upsample
```

Assign New Cluster ID for Each Iteration.

```
def get_clustered_Sample(df, n_per_cluster, num_select_clusters):
    N = len(df)
    K = int(N/n_per_cluster)
    data = None
    for k in range(K):
        sample_k = df.sample(n_per_cluster)
        sample_k["cluster"] = np.repeat(k,len(sample_k))
        df = df.drop(index = sample_k.index)
        data = pd.concat([data,sample_k],axis = 0)
```

```
        random_chosen_clusters = np.random.randint(0,K,size =
    num_select_clusters)
        samples = data[data.cluster.isin(random_chosen_clusters)]
        return(samples)

    sample = get_clustered_Sample(df = df, n_per_cluster = 100, num_select_clusters
    = 2)
    sample.head(2)

print("Before Data Sampling")
top_industry = df.copy()
print(top_industry)
sns.pairplot(top_industry,hue='Industry_Name',kind='hist')

t1 = upsample_classes(top_industry,'Industry_Name')
ti.head()
print("After Data Sampling")
sns.pairplot(top_industry_results,hue='Industry_Names',kind='hist')
```

Report Generation:

- **Type inference**: automatic detection of columns' data types (*Categorical*, *Numerical*, *Date*, etc.)

- **Warnings**: A summary of the problems/challenges in the data that you might need to work on (*missing data*, *inaccuracies*, *skewness*, etc.)
- **Univariate analysis**: including descriptive statistics (mean, median, mode, etc) and informative visualizations such as distribution histograms
- **Multivariate analysis**: including correlations, a detailed analysis of missing data, duplicate rows, and visual support for the variable's pairwise interaction
- **Type inference:** automatic detection of column's data types (Categorical, Numerical, Date, etc.)
- **Warnings:** A summary of the problems/challenges in the data that you might need to work on (missing data, inaccuracies, skewness, etc.)
- **Univariate analysis:** including descriptive statistics (mean, median, mode, etc) and informative visualizations such as distribution histograms
- **Multivariate analysis:** including correlations, a detailed analysis of missing data, duplicate rows, and visual support for the variable's pairwise interaction

Report = pd.ProfileReport("Threadmeter.html",df)

(II)

Libraries used:

import pandas as pd

import seaborn as sns

import numpy as np

```
import matplotlib.pyplot as plt

import plotly.express as px
```

Extracting data from excel:

Data from IBM breach cost report:

```
cost_data = pd.read_excel('avg_cost_threat.xlsx',sheet_name='Sheet1')
```

Attack Vectors for 1000 data points and manipulation:

```
threat_1000_data_with_industries =
pd.read_excel('threatmeter_1000_with_industries.xlsx',sheet_name='Cyber Threats')

threat_1000_data_with_industries

threat_1000_data_with_industries = threat_1000_data_with_industries.iloc[:,1:16]

for x in range(len(threat_1000_data_with_industries.iloc[:,1])):

sum = 0

for y in range(3,11):

sum = sum + threat_1000_data_with_industries.iloc[x,y]

threat_1000_data_with_industries.iloc[x,11] = sum

threat_1000_data_with_industries

data_sep_1_ind = threat_1000_data_with_industries.iloc[:,[0,1,2,11,14]]

data_sep_2_ind = threat_1000_data_with_industries.iloc[:,[0,3,4,5,6,7,8,9,10,14]]
```

manipulated_threat_data_1000_sep_2.reset_index(inplace=True)

manipulated_threat_data_1000_sep_2

Attack Vectors for 200 Data Points and manipulation:

```
threatmeter_200_data =
pd.read_excel('threatmeter_200_IAV_EAV_IND.xlsx',sheet_name='Analysis_200')

data_sep_1_200 = threatmeter_200_data.iloc[:,[2,12,13,14,15,16,17,18,19]]

data_sep_1_200["Total No of Threats"]=0

for x in range(len(data_sep_1_200.iloc[:,1])):

sum = 0

for y in range(1,9):

sum = sum + data_sep_1_200.iloc[x,y]

data_sep_1_200.iloc[x,9] = sum

# threat_1000_data_with_industries

data_sep_1_200.columns =['Industry','SSL Health','IP Reputation','Service
Misconfiguration','Outdated Version','Data Leaks','DNS Misconfiguration','Data
Breaches','Unnecessary Open Ports','Total No of Threats']

data_sep_1_200
```

Threats for 200 datapoints:

threat_data = pd.read_excel('threatmeter.xlsx',sheet_name='Cyber Threats')

copy_of_threat_data = threat_data

threat_data = threat_data.iloc[:,1:8]

manipulated_threat_data = threat_data.groupby('Industry').sum()

manipulated_threat_data.reset_index(inplace=True)

Unique Occurrences for attack vectors for 1000 data points:

unique_occurences_1000 = pd.DataFrame()

unique_occurences_1000.loc[0,"Attack Vector"] = 'SSL Health'

unique_occurences_1000.loc[0,"Minimum of Unique occurrences"] = data_sep_2['SSL Health'].unique().min()

unique_occurences_1000.loc[0,"Maximum of Unique occurrences"] = data_sep_2['SSL Health'].unique().max()

unique_occurences_1000.loc[0,"Average of Unique occurrences"] = data_sep_2['SSL Health'].unique().mean()

unique_occurences_1000.loc[0,"Average of Findings"] = data_sep_2['SSL Health'].mean()

unique_occurences_1000.loc[1,"Attack Vector"] = 'IP Reputation'

unique_occurences_1000.loc[1,"Minimum of Unique occurrences"] = data_sep_2['IP Reputation'].unique().min()

unique_occurences_1000.loc[1,"Maximum of Unique occurrences"] = data_sep_2['IP Reputation'].unique().max()

unique_occurences_1000.loc[1,"Average of Unique occurrences"] = data_sep_2['IP Reputation'].unique().mean()

unique_occurences_1000.loc[1,"Average of Findings"] = data_sep_2['IP Reputation'].mean()

unique_occurences_1000.loc[2,"Attack Vector"] = 'Service Misconfiguration'

unique_occurences_1000.loc[2,"Minimum of Unique occurrences"] = data_sep_2['Service Misconfiguration'].unique().min()

unique_occurences_1000.loc[2,"Maximum of Unique occurrences"] = data_sep_2['Service Misconfiguration'].unique().max()

unique_occurences_1000.loc[2,"Average of Unique occurrences"] = data_sep_2['Service Misconfiguration'].unique().mean()

unique_occurences_1000.loc[2,"Average of Findings"] = data_sep_2['Service Misconfiguration'].mean()

unique_occurences_1000.loc[3,"Attack Vector"] = 'Outdated Version'

unique_occurences_1000.loc[3,"Minimum of Unique occurrences"] = data_sep_2['Outdated Version'].unique().min()

unique_occurences_1000.loc[3,"Maximum of Unique occurrences"] = data_sep_2['Outdated Version'].unique().max()

unique_occurences_1000.loc[3,"Average of Unique occurrences"] = data_sep_2['Outdated Version'].unique().mean()

unique_occurences_1000.loc[3,"Average of Findings"] = data_sep_2['Outdated Version'].mean()

unique_occurences_1000.loc[4,"Attack Vector"] = 'Data Leaks'

unique_occurences_1000.loc[4,"Minimum of Unique occurrences"] = data_sep_2['Data Leaks'].unique().min()

unique_occurences_1000.loc[4,"Maximum of Unique occurrences"] = data_sep_2['Data Leaks'].unique().max()

unique_occurences_1000.loc[4,"Average of Unique occurrences"] = data_sep_2['Data Leaks'].unique().mean()

unique_occurences_1000.loc[4,"Average of Findings"] = data_sep_2['Data Leaks'].mean()

unique_occurences_1000.loc[5,"Attack Vector"] = 'DNS Misconfiguration'

unique_occurences_1000.loc[5,"Minimum of Unique occurrences"] = data_sep_2['DNS Misconfiguration'].unique().min()

unique_occurences_1000.loc[5,"Maximum of Unique occurrences"] = data_sep_2['DNS Misconfiguration'].unique().max()

unique_occurences_1000.loc[5,"Average of Unique occurrences"] = data_sep_2['DNS Misconfiguration'].unique().mean()

unique_occurences_1000.loc[5,"Average of Findings"] = data_sep_2['DNS Misconfiguration'].mean()

unique_occurences_1000.loc[6,"Attack Vector"] = 'Data Breaches'

unique_occurences_1000.loc[6,"Minimum of Unique occurrences"] = data_sep_2['Data Breaches'].unique().min()

unique_occurences_1000.loc[6,"Maximum of Unique occurrences"] = data_sep_2['Data Breaches'].unique().max()

unique_occurences_1000.loc[6,"Average of Unique occurrences"] = data_sep_2['Data Breaches'].unique().mean()

unique_occurences_1000.loc[6,"Average of Findings"] = data_sep_2['Data Breaches'].mean()

unique_occurences_1000.loc[7,"Attack Vector"] = 'Unnecessary Open Ports'

unique_occurences_1000.loc[7,"Minimum of Unique occurrences"] = data_sep_2['Unnecessary Open Ports'].unique().min()

unique_occurences_1000.loc[7,"Maximum of Unique occurrences"] = data_sep_2['Unnecessary Open Ports'].unique().max()

unique_occurences_1000.loc[7,"Average of Unique occurrences"] = data_sep_2['Unnecessary Open Ports'].unique().mean()

unique_occurences_1000.loc[7,"Average of Findings"] = data_sep_2['Unnecessary Open Ports'].mean()

unique_occurences_1000.sort_values(by="Average of Findings",ascending=False)

Unique Occurrences for attack vectors for 200 data points:

unique_occurences_200 = pd.DataFrame()

unique_occurences_200.loc[0,"Attack Vector"] = 'SSL Health'

unique_occurences_200.loc[0,"Minimum of Unique occurrences"] = data_sep_1_200['SSL Health'].unique().min()

unique_occurences_200.loc[0,"Maximum of Unique occurrences"] = data_sep_1_200['SSL Health'].unique().max()

unique_occurences_200.loc[0,"Average of Unique occurrences"] = data_sep_1_200['SSL Health'].unique().mean()

unique_occurences_200.loc[0,"Average of Findings"] = data_sep_1_200['SSL Health'].mean()

unique_occurences_200.loc[1,"Attack Vector"] = 'IP Reputation'

unique_occurences_200.loc[1,"Minimum of Unique occurrences"] = data_sep_1_200['IP Reputation'].unique().min()

unique_occurences_200.loc[1,"Maximum of Unique occurrences"] = data_sep_1_200['IP Reputation'].unique().max()

unique_occurences_200.loc[1,"Average of Unique occurrences"] = data_sep_1_200['IP Reputation'].unique().mean()

unique_occurences_200.loc[1,"Average of Findings"] = data_sep_1_200['IP Reputation'].mean()

unique_occurences_200.loc [2,"Attack Vector"] = 'Service Misconfiguration'

unique_occurences_200.loc [2,"Minimum of Unique occurrences"] = data_sep_1_200['Service Misconfiguration'].unique().min()

unique_occurences_200.loc [2,"Maximum of Unique occurrences"] = data_sep_1_200['Service Misconfiguration'].unique().max()

unique_occurences_200.loc [2,"Average of Unique occurrences"] = data_sep_1_200['Service Misconfiguration'].unique().mean()

unique_occurences_200.loc [2,"Average of Findings"] = data_sep_1_200['Service Misconfiguration'].mean()

unique_occurences_200.loc [3,"Attack Vector"] = 'Outdated Version'

unique_occurences_200.loc[3,"Minimum of Unique occurrences"] = data_sep_1_200['Outdated Version'].unique().min()

unique_occurences_200.loc[3,"Maximum of Unique occurrences"] = data_sep_1_200['Outdated Version'].unique().max()

unique_occurences_200.loc[3,"Average of Unique occurrences"] = data_sep_1_200['Outdated Version'].unique().mean()

unique_occurences_200.loc[3,"Average of Findings"] = data_sep_1_200['Outdated Version'].mean()

unique_occurences_200.loc[4,"Attack Vector"] = 'Data Leaks' unique_occurences_200.loc[4,"Minimum of Unique occurrences"] = data_sep_1_200['Data Leaks'].unique().min()

unique_occurences_200.loc[4,"Maximum of Unique occurrences"] = data_sep_1_200['Data Leaks'].unique().max()

unique_occurences_200.loc[4,"Average of Unique occurrences"] = data_sep_1_200['Data Leaks'].unique().mean()

unique_occurences_200.loc[4,"Average of Findings"] = data_sep_1_200['Data Leaks'].mean()

unique_occurences_200.loc[5,"Attack Vector"] = 'DNS Misconfiguration'

unique_occurences_200.loc[5,"Minimum of Unique occurrences"] = data_sep_1_200['DNS Misconfiguration'].unique().min()

unique_occurences_200.loc[5,"Maximum of Unique occurrences"] = data_sep_1_200['DNS Misconfiguration'].unique().max()

unique_occurences_200.loc[5,"Average of Unique occurrences"] = data_sep_1_200['DNS Misconfiguration'].unique().mean()

unique_occurences_200.loc[5,"Average of Findings"] = data_sep_1_200['DNS Misconfiguration'].mean()

unique_occurences_200.loc[6,"Attack Vector"] = 'Data Breaches'

unique_occurences_200.loc[6,"Minimum of Unique occurrences"] = data_sep_1_200['Data Breaches'].unique().min()

unique_occurences_200.loc[6,"Maximum of Unique occurrences"] = data_sep_1_200['Data Breaches'].unique().max()

unique_occurences_200.loc[6,"Average of Unique occurrences"] = data_sep_1_200['Data Breaches'].unique().mean()

unique_occurences_200.loc[6,"Average of Findings"] = data_sep_1_200['Data Breaches'].mean()

unique_occurences_200.loc[7,"Attack Vector"] = 'Unnecessary Open Ports'

unique_occurences_200.loc[7,"Minimum of Unique occurrences"] = data_sep_1_200['Unnecessary Open Ports'].unique().min()

unique_occurences_200.loc[7,"Maximum of Unique occurrences"] = data_sep_1_200['Unnecessary Open Ports'].unique().max()

unique_occurences_200.loc[7,"Average of Unique occurrences"] = data_sep_1_200['Unnecessary Open Ports'].unique().mean()

unique_occurences_200.loc[7,"Average of Findings"] = data_sep_1_200['Unnecessary Open Ports'].mean()

unique_occurences_200.sort_values(by="Average of Findings",ascending=False)

Unique Occurrences for Threats for 200 data points:

unique_occurences_200_tv = pd.DataFrame()

unique_occurences_200_tv.loc[0,"Attack Vector"] = 'Phishing Threats' unique_occurences_200_tv.loc[0,"Minimum of Unique occurrences"] = threat_data['Phishing Threats'].unique().min()

unique_occurences_200_tv.loc[0,"Maximum of Unique occurrences"] = threat_data['Phishing Threats'].unique().max()

unique_occurences_200_tv.loc[0,"Average of Unique occurrences"] = threat_data['Phishing Threats'].unique().mean()

unique_occurences_200_tv.loc[0,"Average of Findings"] = threat_data['Phishing Threats'].mean()

unique_occurences_200_tv.loc[1,"Attack Vector"] = 'Brand & Reputation Threats'

unique_occurences_200_tv.loc[1,"Minimum of Unique occurrences"] = threat_data['Brand & Reputation Threats'].unique().min()

unique_occurences_200_tv.loc[1,"Maximum of Unique occurrences"] = threat_data['Brand & Reputation Threats'].unique().max()

unique_occurences_200_tv.loc[1,"Average of Unique occurrences"] = threat_data['Brand & Reputation Threats'].unique().mean()

unique_occurences_200_tv.loc[1,"Average of Findings"] = threat_data['Brand & Reputation Threats'].mean()

unique_occurences_200_tv.loc[2,"Attack Vector"] = 'Rogue Mobile Apps'

unique_occurences_200_tv.loc[2,"Minimum of Unique occurrences"] = threat_data['Rogue Mobile Apps'].unique().min()

unique_occurences_200_tv.loc[2,"Maximum of Unique occurrences"] = threat_data['Rogue Mobile Apps'].unique().max()

unique_occurences_200_tv.loc[2,"Average of Unique occurrences"] = threat_data['Rogue Mobile Apps'].unique().mean()

unique_occurences_200_tv.loc[2,"Average of Findings"] = threat_data['Rogue Mobile Apps'].mean()

unique_occurences_200_tv.loc[3,"Attack Vector"] = 'Data Breaches'

unique_occurences_200_tv.loc[3,"Minimum of Unique occurrences"] = threat_data['Data Breaches'].unique().min()

unique_occurences_200_tv.loc[3,"Maximum of Unique occurrences"] = threat_data['Data Breaches'].unique().max()

unique_occurences_200_tv.loc[3,"Average of Unique occurrences"] = threat_data['Data Breaches'].unique().mean()

unique_occurences_200_tv.loc[3,"Average of Findings"] = threat_data['Data Breaches'].mean()

unique_occurences_200_tv.loc[4,"Attack Vector"] = 'Data Leaks'

unique_occurences_200_tv.loc[4,"Minimum of Unique occurrences"] = threat_data['Data Leaks'].unique().min()

unique_occurences_200_tv.loc[4,"Maximum of Unique occurrences"] = threat_data['Data Leaks'].unique().max()

unique_occurences_200_tv.loc[4,"Average of Unique occurrences"] = threat_data['Data Leaks'].unique().mean()

unique_occurences_200_tv.loc[4,"Average of Findings"] = threat_data['Data Leaks'].mean()

unique_occurences_200_tv.sort_values(by="Average of Findings",ascending=False)

Data transformation before calculation monetary impact for attack vectors for 1000 data points:

```
for x in range(3,11):
for y in range(len(threat_1000_data.iloc[:,x])):
if(threat_1000_data.iloc[y,x]>0):
threat_1000_data.iloc[y,x] = 1
threat_1000_data.head()
manipulated_threat_1000_data = threat_1000_data.groupby('TM ID').sum()
manipulated_threat_1000_data.reset_index(inplace=True)
manipulated_threat_1000_data.head(100)
data_sep_1 = manipulated_threat_1000_data.iloc[:,[0,1,2,11]]
data_sep_2 = manipulated_threat_1000_data.iloc[:,[0,3,4,5,6,7,8,9,10]]
for x in range(len(data_sep_2)):
avg_cost_usd = 0
```

```
avg_cost_inr = 0

for y in range(1,9):

avg_cost_usd = avg_cost_usd + data_sep_2.iloc[x,y] * filtered_cost[y-1]

avg_cost_inr = avg_cost_inr + data_sep_2.iloc[x,y] * filtered_cost_inr[y-data_sep_2.at[x,'AVG
COST IN USD'] = avg_cost_usd

 data_sep_2.at[x,'AVG COST IN INR'] = avg_cost_inr

data_sep_2
```

Data transformation before calculation monetary impact for attack vectors for 200 data points:

```
for x in range(1,9):

for y in range(len(data_sep_1_200.iloc[:,x])):

if(data_sep_1_200.iloc[y,x]>0):

data_sep_1_200.iloc[y,x] = 1

data_sep_1_200["Total No of Threats"]=0

for x in range(len(data_sep_1_200.iloc[:,1])):

sum = 0

for y in range(1,9):

sum = sum + data_sep_1_200.iloc[x,y]
```

```
data_sep_1_200.iloc[x,9] = sum
# threat_1000_data_with_industries
data_sep_1_200.columns =['Industry','SSL Health','IP Reputation','Service
Misconfiguration','Outdated Version','Data Leaks','DNS Misconfiguration','Data
Breaches','Unnecessary Open Ports','Total No of Threats']
data_sep_1_200
data_sep_1_200['Total COST IN USD']=0
for x in range(len(data_sep_1_200)):
avg_cost_usd = 0
for y in range(1,9):
avg_cost_usd = avg_cost_usd + data_sep_1_200.iloc[x,y] * filtered_cost[y-1]
data_sep_1_200.at[x,'Total COST IN USD'] = avg_cost_usd
data_sep_1_200
```

Data transformation before calculation monetary impact for threats for 200 data points:

```
for x in range(1,6):
 for y in range(len(threat_data.iloc[:,x])):
if(threat_data.iloc[y,x]>0):
threat_data.iloc[y,x] = 1
```

```
manipulated_threat_data = threat_data.groupby('Industry').sum()

manipulated_threat_data.reset_index(inplace=True)

for x in range(len(manipulated_threat_data.iloc[:,1])):

sum = 0

for y in range(1,6):

sum = sum + manipulated_threat_data.iloc[x,y]

manipulated_threat_data.iloc[x,6] = sum

cost_in_inr =[]

for cost in cost_data['AVG COST']:

cost_in_inr.append(round(c.convert(cost,'USD','INR')/10000000,2))

cost_in_inr = pd.DataFrame({'AVG COST in INR CRORES': cost_in_inr})

new_cost_data = pd.DataFrame([cost_data['AVG COST']/1000000,cost_in_inr['AVG COST in
INR CRORES']])

new_cost_data.columns= cost_data['Threat']

new_cost_data

x_axis_name =[x for x in manipulated_threat_data.columns[1:6]]

filtered_cost = []

for x in x_axis_name:
```

```
filtered_cost.append(new_cost_data.loc['AVG COST',x])
filtered_cost
filtered_cost_inr = []
for x in x_axis_name:
filtered_cost_inr.append(new_cost_data.loc['AVG COST in INR CRORES',x])
filtered_cost_inr
copy_of_manipulated_data = manipulated_threat_data
copy_of_manipulated_data["COST IN USD (millions)"] = ''
copy_of_manipulated_data["COST IN INR (CRORES)"] = ''
for x in range(len(copy_of_manipulated_data)):
avg_cost_usd = 0
avg_cost_inr = 0
for y in range(1,6):
 avg_cost_usd = avg_cost_usd + copy_of_manipulated_data.iloc[x,y] * filtered_cost[y-1]
 avg_cost_inr = avg_cost_inr + copy_of_manipulated_data.iloc[x,y] * filtered_cost_inr[y-1]
 copy_of_manipulated_data.at[x,"COST IN USD (millions)"] = avg_cost_usd
copy_of_manipulated_data.at[x,"COST IN INR (CRORES)"] = avg_cost_inr
copy_of_manipulated_data
```

Total no of findings and monetary impact for attack vectors for 1000 data points:

```
column_name = manipulated_threat_data_1000_sep_2.iloc[:,1:9].columns

sum_of_attack_vector = []

for x in column_name:

sum_of_attack_vector.append(manipulated_threat_data_1000_sep_2[x].sum())

total_threat = 0

for x in sum_of_attack_vector:

 total_threat = total_threat + x

#percent_tot_thr = sum_of_attack_vector*(100/total_threat)

percent=[]

for x in sum_of_attack_vector:

percent.append(str(round(x*(100/total_threat),2))+'%')

percent

by_att_vect = pd.DataFrame({'Attack Vector':column_name,'Total No of
Threats':sum_of_attack_vector,'Percent':percent})

print('The total number of threats found: '+str(total_threat))

by_att_vect.sort_values(by="Total No of Threats",ascending=False)

by_att_vect["Total Cost"] = 0
```

```
for x in range(len(by_att_vect)):

by_att_vect.iloc[x,3] = round(by_att_vect.iloc[x,1] * filtered_cost[x],2)

by_att_vect.iloc[:,[0,3]]
```

Total no of findings and monetary impact for attack vectors for 200 data points:

```
x_axis_name =[x for x in data_sep_1_200.columns[1:9]]

x_axis = np.arange(len(x_axis_name))

data =[]

cost_threat_200=[]

for x in range(1,len(x_axis)+1):

 data.append(data_sep_1_200.iloc[:,x].sum())

total_number_of_threats = np.array(data).sum()

print(total_number_of_threats)

for x in range(0,len(x_axis)):

cost_threat_200.append(round(data[x] * filtered_cost[x],2))

print(f'Out of {total_number_of_threats} Total Threats, {data[x]} is {x_axis_name[x]} and
occupies {round((data[x]*100)/total_number_of_threats,2)}% of Total no of Threats. The
{x_axis_name[x]} has produced loss of {round(data[x] * filtered_cost[x],2)} millions in USD or
{round(data[x] * filtered_cost_inr[x])} Crores in INR' )
```

```
temp_data = pd.DataFrame({'Attack Vector':x_axis_name,'count':data})

temp_data

fig = px.pie(temp_data, values='count', names='Attack Vector',title='Distribution of Attack
Vectors')

fig.show()

av_by_tot_200 = temp_data

percent_av_by_200 =[]

for x in list(temp_data.iloc[:,1]):

percent_av_by_200.append(str(round(((x*100)/temp_data.iloc[:,1].sum()),2))+'%')

av_by_tot_200 ["Percentage"] = percent_av_by_200

print("Total count: ",temp_data.iloc[:,1].sum())

av_by_tot_200.sort_values(by="count",ascending=False)

temp_data["Total Cost"]=0

for x in range(len(temp_data)):

 temp_data.at[x,'Total Cost']= temp_data.iloc[x,1] * filtered_cost[x]

av_by_tc_200=temp_data.iloc[:,[0,3]]

print("The Total cost: ",temp_data.iloc[:,3].sum()," million USD")

av_by_tc_200
```

Total no of findings and monetary impact for threats for 200 data points:

```
x_axis_name =[x for x in threat_data.columns[1:6]]

x_axis = np.arange(len(x_axis_name))

data =[]

cost_threat_200=[]

for x in range(1,len(x_axis)+1):

data.append(threat_data.iloc[:,x].sum())

percent_tv_200 = []

total_number_of_threats = np.array(data).sum()

print(total_number_of_threats)

for x in range(0,len(x_axis)):

cost_threat_200.append(round(data[x] * filtered_cost[x],2))

print(f'Out of {total_number_of_threats} Total Threats, {data[x]} is {x_axis_name[x]} and
occupies {round((data[x]*100)/total_number_of_threats,2)}% of Total no of Threats. The
{x_axis_name[x]} has produced loss of {round(data[x] * filtered_cost[x],2)} millions in USD or
{round(data[x] * filtered_cost_inr[x])} Crores in INR' )

for x in data:

percent_tv_200.append(str(round(x*(100/total_number_of_threats),2))+'%')
```

```
temp_data = pd.DataFrame({'Attack
Vector':x_axis_name,'count':data,'percentage':percent_tv_200})
temp_data.sort_values(by="count",ascending=False)
temp_data["Total Cost"]=0
for x in range(len(temp_data)):
temp_data.at[x,'Total Cost']= temp_data.iloc[x,1] * filtered_cost[x]
eav_by_tc_200=temp_data.iloc[:,[0,2]]
print("The Total Cost: ",eav_by_tc_200.iloc[:,1].sum()," million USD")
eav_by_tc_200
```

Priority matrix for Proactive and Remediation for attack vector for 1000 data points:

```
threatscore_1000_data=pd.merge(threatscore_1000,data_sep_1,on='TM ID').iloc[:,0:11]
for i in range(len(threatscore_1000_data)):
for j in range(1,9):
if(threatscore_1000_data.iloc[i,9]!=0):
threatscore_1000_data.iloc[i,j] = round(100-
(threatscore_1000_data.iloc[i,j]*100)/threatscore_1000_data.iloc[i,9],2)
threatscore_1000_data
threatscore_1000_min= threatscore_1000_data.min()
```

```
threatscore_1000_min = pd.DataFrame({'Name':threatscore_1000_min.keys(),"Minimum Threat score Reduction in %":threatscore_1000_min.values}).iloc[1:9,:]

threatscore_1000_min

threatscore_1000_max= threatscore_1000_data.max()

threatscore_1000_max = pd.DataFrame({'Name':threatscore_1000_max.keys(),'Maximum Threat score Reduction in %':threatscore_1000_max.values}).iloc[1:9,:]

threatscore_1000_max

result_ts

result_ts

priority_matrix_list =[]

attack_name=[]

for i in result_ts.keys():

if i != "Threat Score" and i != 'Latest Threat score':

attack_name.append(i)

priority_matrix_list.append(round(1-result_ts[i]/result_ts["Latest Threat score"],4))

priority_matrix = pd.DataFrame({'Attack_vector':attack_name,'Threatscore reduced / Latest Threatscore':priority_matrix_list,'Complexity proactive':[2,3,5,4,8,6,7,1],'complexity remediation':[2,3,6,4,7,5,8,1]})
```

```
priority_matrix_sum=priority_matrix.iloc[:,1].sum()

priority_matrix_sum_diff = 1 - priority_matrix_sum

for i in range(len(priority_matrix)):

 priority_matrix.iloc[i,1] = round(priority_matrix.iloc[i,1]+
(priority_matrix.iloc[i,1]/priority_matrix_sum* priority_matrix_sum_diff),4)

priority_matrix

for i in range(len(priority_matrix)):

priority_matrix.loc[i,'Proactive Priority'] = priority_matrix.iloc[i,1] * (9-priority_matrix.iloc[i,2])

priority_matrix.iloc[:,[0,1,2,4]].sort_values(by="Proactive Priority",ascending=False)


for i in range(len(priority_matrix)):

priority_matrix.loc[i,'Remediation Priority'] = priority_matrix.iloc[i,1] * (9-
priority_matrix.iloc[i,3])

priority_matrix.iloc[:,[0,1,3,5]].sort_values(by="Remediation Priority",ascending=False)

result_ts["Latest Threat score"]

name =[]

percent_chng=[]

or i in result_ts.keys():
```

```
 name.append(i)

 percent_chng.append(round(100-(result_ts[i] * 100)/result_ts["Latest Threat score"],2))

ts_percent_change =  pd.DataFrame({'Name':name,'Threatscore reduced in %
(AVG)':percent_chng}).iloc[[0,1,2,4,5,6,7,8],:]

ts_percent_sum=ts_percent_change.iloc[:,1].sum()

ts_percent_sum_diff = 100 - ts_percent_sum

for i in range(len(ts_percent_change)):

 ts_percent_change.iloc[i,1] = round(ts_percent_change.iloc[i,1]+
(ts_percent_change.iloc[i,1]/ts_percent_sum* ts_percent_sum_diff),2)

ts_percent_change

att_vec_by_tot_and_mean_indiv_weight =
pd.read_excel('Attack_Vector_By_totandmean_of_indiv_weight.xlsx')

att_vec_by_tot_and_mean_indiv_weight = att_vec_by_tot_and_mean_indiv_weight.iloc[:,1:4]

att_vec_by_tot_and_mean_indiv_weight.iloc[:,2]=
round(att_vec_by_tot_and_mean_indiv_weight.iloc[:,2],2)

att_vec_by_tot_and_mean_indiv_weight

threatscore_1000_resultant =  pd.merge(ts_percent_change,threatscore_1000_min,on="Name")

threatscore_1000_resultant =
pd.merge(threatscore_1000_resultant,threatscore_1000_max,on="Name")
```

```
threatscore_1000_resultant =
pd.merge(threatscore_1000_resultant,att_vec_by_tot_and_mean_indiv_weight,on="Name")
threatscore_1000_resultant.columns=['Attack Vector','Threatscore reduced in %
(AVG)',   'Minimum Threat score Reduction in %','Maximum Threat score Reduction in %','Total
Of Individual Weight','Average Of Individual Weight']
threatscore_1000_resultant
```

Priority matrix for Proactive and Remediation for attack vector for 200 data points:

```
temp_200_data = pd.read_excel('threatmeter.xlsx',sheet_name='Cyber Threats')
temp_list=list(temp_200_data.iloc[:,0])
temp_list.sort()
print(temp_list)
threatscore_200 = pd.read_excel('threatmeter_score_200.xlsx')
threatmeter_200_data.sort_values(by="Customer ID")
threatscore_200_previous = threatmeter_200_data.iloc[:,[1,10]]
threatscore_200_previous.columns=["TM ID", "Previous Threat score"]
threatscore_200 = pd.merge(threatscore_200,threatscore_200_previous,on="TM ID")
threatscore_200_filtered = threatscore_200.iloc[:,1:10]
threatscore_200_filtered
```

```
result_ts_200 = threatscore_200_filtered.sum().sort_values()

result_ts_200

threatscore_200_filtered.mean().sort_values()

for i in range(len(threatscore_200)):

for j in range(1,9):

 if(threatscore_200.iloc[i,9]!=0):

 threatscore_200.iloc[i,j] = round(100-(threatscore_200.iloc[i,j]*100)/threatscore_200.iloc[i,9],2)

threatscore_200

threatscore_200_min= threatscore_200.min()

threatscore_200_min = pd.DataFrame({'Attack Vector':threatscore_200_min.keys(),"Minimum
Threat score Reduction in %":threatscore_200_min.values}).iloc[1:9,:]

threatscore_200_min

threatscore_200_max= threatscore_200.max()

threatscore_200_max = pd.DataFrame({'Attack Vector':threatscore_200_max.keys(),'Maximum
Threat score Reduction in %':threatscore_200_max.values}).iloc[1:9,:]

threatscore_200_max

result_ts_200

priority_matrix_list_200 =[]
```

```
attack_name_200=[]

for i in result_ts_200.keys():

if i != "Threat Score" and i != 'Latest Threat score':

attack_name_200.append(i)

priority_matrix_list_200.append(round(1-result_ts_200[i]/result_ts_200["Latest Threat
score"],4))

priority_matrix_200 = pd.DataFrame({'Attack_vector':attack_name_200,'Threatscore reduced /
Latest Threatscore':priority_matrix_list_200,'Complexity proactive':[2,3,5,8,4,6,1,7],'complexity
remediation':[2,3,6,7,4,5,1,8]})

priority_matrix_sum_200=priority_matrix_200.iloc[:,1].sum()

priority_matrix_sum_diff_200 = 1 - priority_matrix_sum_200

for i in range(len(priority_matrix_200)):

priority_matrix_200.iloc[i,1] = round(priority_matrix_200.iloc[i,1]+
(priority_matrix_200.iloc[i,1]/priority_matrix_sum_200* priority_matrix_sum_diff_200),4)

priority_matrix_200

for i in range(len(priority_matrix_200)):

priority_matrix_200.loc[i,'Proactive Priority'] = priority_matrix_200.iloc[i,1] * (9-
priority_matrix_200.iloc[i,2])

priority_matrix_200.iloc[:,[0,1,2,4]].sort_values(by="Proactive Priority",ascending=False)
```

```
for i in range(len(priority_matrix_200)):
priority_matrix_200.loc[i,'Remediation Priority'] = priority_matrix_200.iloc[i,1] * (9-
priority_matrix_200.iloc[i,3])
priority_matrix_200.iloc[:,[0,1,3,5]].sort_values(by="Remediation Priority",ascending=False)
for i in range(len(priority_matrix_200)):
 priority_matrix_200.loc[i,'Proactive Priority'] = priority_matrix_200.iloc[i,1] * (9-
priority_matrix_200.iloc[i,2])
priority_matrix_200.iloc[:,[0,1,2,4]].sort_values(by="Proactive Priority",ascending=False)
result_ts_200["Latest Threat score"]
name =[]
percent_chng=[]
for i in result_ts_200.keys():
 name.append(i)
percent_chng.append(round(100-(result_ts_200[i] * 100)/result_ts_200["Latest Threat
score"],2))
ts_percent_change_200 =  pd.DataFrame({'Attack Vector':name,'Threatscore reduced in %
(AVG)':percent_chng}).iloc[[0,1,2,3,4,5,6,7],:]
# ts_percent_change.sort_values(by='Threat score reduced in %',ascending=False)
```

```
ts_percent_sum_200=ts_percent_change_200.iloc[:,1].sum()

ts_percent_sum_diff_200 = 100 - ts_percent_sum_200

for i in range(len(ts_percent_change_200)):

ts_percent_change_200.iloc[i,1] = round(ts_percent_change_200.iloc[i,1]+
(ts_percent_change_200.iloc[i,1]/ts_percent_sum_200* ts_percent_sum_diff_200),2)

ts_percent_change_200

att_vec_by_tot_and_mean_indiv_weight_200 =
pd.read_excel('Attack_Vector_By_totandmean_of_indiv_weight_200.xlsx')

att_vec_by_tot_and_mean_indiv_weight_200 =
att_vec_by_tot_and_mean_indiv_weight_200.iloc[:,1:4]

att_vec_by_tot_and_mean_indiv_weight_200.iloc[:,2]=
round(att_vec_by_tot_and_mean_indiv_weight_200.iloc[:,2],2)

att_vec_by_tot_and_mean_indiv_weight_200

threatscore_200_resultant =  pd.merge(ts_percent_change_200,threatscore_200_min,on="Attack
Vector")

threatscore_200_resultant =
pd.merge(threatscore_200_resultant,threatscore_200_max,on="Attack Vector")
```

```
threatscore_200_resultant =
pd.merge(threatscore_200_resultant,att_vec_by_tot_and_mean_indiv_weight_200,on="Attack Vector")

threatscore_200_resultant.columns=['Attack Vector','Threatscore reduced in %
(AVG)',  'Minimum Threat score Reduction in %','Maximum Threat score Reduction in %','Total
Of Individual Weight','Average Of Individual Weight']

threatscore_200_resultant
```

(II)

- **Library Importing:**

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import plotly.express as px

from sklearn.utils import resample
```

- **Data Loading:**

```
df = pd.read_excel('Data/threatmeter_1000_with_industries.xlsx')

df.head()
```

- **Industry wise Correlation plot and data distribution**

```
sns.pairplot(df,hue='Industry_Name',diag_kind="hist",corner=True)
```

- **Data Sampling Approaches:**

Up sample Data.

```
def factorial(n):
    if n==1:return 1
    else: return n * factorial(n-1)
def permutation_without_repetition(n,r):
    return (factorial(n)/(factorial(n-r)))
def permutation_with_repetition(n,r):
    return n ** r
def combinations_without_repetition(n,r):
    return (factorial(n)/(factorial(r)*(factorial(n-r))))
def combinations_with_repetition(n,r):
    return ((factorial(n+r-1))/(factorial(r)*(factorial(n-1))))
def upsample_classes(data, target):
lst = list(data[target].unique())
            classes = []
```

```
            for c in lst:
classes.append(data[data[target]==c])
            length = 0
            class_lab = None
                  for c in classes:
                        if len(c)>length:
                                    length=len(c)
                              class_lab = c
                  class_lab = class_lab[target].unique()[0]
            regroup = pd.concat(classes)
            maj_class = regroup[regroup[target]==class_lab]
lst.remove(class_lab)
            new_classes=[]
            for i in lst:
                  new_classes.append(resample(data[data[target]==i],replace=True,
n_samples=len(maj_class)))
  minority_classes = pd.concat(new_classes)
  upsample = pd.concat([regroup[regroup[target]==class_lab],minority_classes])
```

```
   return upsample
```

Assign New Cluster ID for Each Iteration.

```
def get_clustered_Sample(df, n_per_cluster, num_select_clusters):
N = len(df)
    K = int(N/n_per_cluster)
    data = None
    for k in range(K):
        sample_k = df.sample(n_per_cluster)
        sample_k["cluster"] = np.repeat(k,len(sample_k))
        df = df.drop(index = sample_k.index)
        data = pd.concat([data,sample_k],axis = 0)

    random_chosen_clusters = np.random.randint(0,K,size = num_select_clusters)
    samples = data[data.cluster.isin(random_chosen_clusters)]
    return(samples)
sample = get_clustered_Sample(df = df, n_per_cluster = 100, num_select_clusters = 2)
sample.head(2)
print("Before Data Sampling")
```

```
top_industry = df.copy()

print(top_industry)

sns.pairplot(top_industry,hue='Industry_Name',kind='hist')

t1 = upsample_classes(top_industry,'Industry_Name')

ti.head()

print(“After Data Sampling”)

sns.pairplot(top_industry_results,hue='Industry_Names',kind='hist')
```

Report Generation:

- **Type inference**: Automatic detection of columns’ data types (*Categorical*, *Numerical*, *Date*, etc.)
- **Warnings**: A summary of the problems/challenges in the data that one might need to work on (*missing data*, *inaccuracies*, *skewness*, etc.)
- **Univariate analysis**: Including descriptive statistics (mean, median, mode, etc) and informative visualizations such as distribution histograms
- **Multivariate analysis**: Including correlations, a detailed analysis of missing data, duplicate rows, and visual support for the variable’s pairwise interaction
- **Type inference:** Automatic detection of column’s data types (Categorical, Numerical, Date, etc.)

- **Warnings:** A summary of the problems/challenges in the data that one might need to work on (missing data, inaccuracies, skewness, etc.)
- **Univariate analysis:** Including descriptive statistics (mean, median, mode, etc) and informative visualizations such as distribution histograms
- **Multivariate analysis:** Including correlations, a detailed analysis of missing data, duplicate rows, and visual support for the variable’s pairwise interaction

Report = pd.ProfileReport(“Threadmeter.html”,df)

APPENDIX C

SURVEY QUESTIONS

1. Do you think CISOs/Security Heads have complete visibility of external assets?
    a. Yes
    b. Partially
    c. No
    d. Not Sure
2. Do you see the need to use automated tools to discover, maintain and update **assets** (High Lighted)?
    a. Yes, it's a must-have for every organization
    b. Only required for Enterprises
    c. Can be discovered and maintained manually
    d. Not needed
    e. Not sure
    f. Other (please specify)
3. How can be unsanctioned shadow IT assets discovered?
    a. By using Asset Discovery Tools
    b. By using Attack Surface Monitoring Tools
    c. Time-to-time review of assets by the IT team
    d. During incident response
    e. Not sure
    f. Other (please specify)

4. What should we do when employees use unsanctioned Shadow IT assets?
    a. Companies can understand the needs of their employees and adapt IT policies.
    b. Educate the employees about Shadow IT and its risks.
    c. Identify the business requirements that Shadow IT meets and provide an approved alternative.
    d. Not sure
    e. Other (please specify)
5. What are the biggest problems in shadow IT assets?
    a. Unknown/undiscovered assets
    b. Use of unsanctioned software
    c. Cloud instance deployed without approval
    d. Company code published on the developer's personal code repository
    e. Use of document/file-sharing platforms
    f. Use of personal storage devices
    g. Not sure
    h. Other
6. What are your views on implementing proactive controls for the internet-facing assets (external attack surface)?
    a. Important for safeguarding attack surface
    b. Not important
    c. Not sure
    d. Other (please specify)

7. What do you think is the biggest challenge of the growing external attack surface?
8. Do you think organizations must have documented configuration baselines for domains, servers, cloud, DNS, social media accounts, and other external assets?
    a. Yes, it's a must-have for every organization
    b. Yes, need it for compliance
    c. Optional – depends on the organization's needs
    d. Not needed
    e. Not sure
    f. Other (please specify)
9. In your experience, what are the available methods in the industry to detect vulnerabilities or anomalies in the attack surface? 9TH Question

Hint: Opensource Intelligence (OSINT) Process, Attack Surface Monitoring tools

10. Do you have any preferences in the attack surface management tools available in the industry?
    a. CloudSek X-Vigil
    b. Upguard
    c. Digital Shadows
    d. Izoologic
    e. Sumeru Threat Meter
    f. Not sure
    g. Other (please specify)

11. Do you think it is necessary to consider the entire attack surface in security risk assessments?
    a. Yes, required
        i. If yes, then (How frequently do you think it should be done?)
            a) Continuously
            b) Weekly
            c) Monthly
            d) Quarterly
            e) Half Yearly
            f) Annually
    b. Not required
    c. Not sure
    d. Other (please specify)
12. Do you think organizations include third parties they interact with in the attack surface management?
    a. Yes
    b. No
    c. Partially
    d. Not sure
13. Do you feel vulnerability remediation is a lengthy process and often misses urgency?
    a. Yes
    b. No

c. Not sure

14. What do you think is an appropriate frequency for reviewing scan results and prioritizing vulnerabilities found in external attack surface?

a. Daily

b. Weekly

c. Bi-Weekly

d. Monthly

e. Quarterly

f. Not sure

15. Do you think there is a good standard or framework in the industry for prioritizing external attack surface findings?

a. Yes

If yes, Please specify the standards or frameworks available for prioritizing the findings.

b. No

If No. Does the industry need a good standard or framework for prioritizing the findings?

c. Not sure

16. Do you think there is a mechanism readily available in the industry to calculate the value of the asset and its context?

a. Yes

If yes, Please specify the mechanism available in the industry to calculate the value of the asset and its context.

b. No

c. Not sure

17. How do CISOs/Security leaders manage remediation for vulnerabilities identified in the external attack surface?

18. Do you think companies are prepared to takedown emerging external cyber threats themselves or do they need third-party support?

    a. Prepared

    b. Take third-party support

    c. Prepared but take third-party support when required.

    d. Not prepared

    e. Not sure

    f. Other (please specify)

19. How should organizations handle sensitive data leaked on the dark web?

20. What are your current biggest challenges in remediation/patching?

    a. Legacy system/software

    b. Time/efforts required for remediation/patching

    c. Expertise required for remediation/patching

    d. Resistance from cross functional business teams

    e. Not sure

    f. Other – please specify

21. What should be the rubrics for measuring the improvements in the attack surface?

APPENDIX D

INTERVIEW QUESTIONS

1. What do you think is the biggest challenge of the growing external attack surface?
2. Do you think CISOs/Security Heads have complete visibility of external assets?
3. What are the biggest problems in shadow IT assets?
4. How do you ensure that new technologies and systems being introduced to the organization do not create additional risks to the attack surface?
5. Our recent research on the attack surface of Alexa's Top 1000 companies revealed that Service Misconfiguration (e.g Web Server Misconfiguration, Application Misconfiguration) is the most common risk organizations are facing. Do you feel similar experiences align with the research findings or differ?
6. The research also revealed that continuous monitoring of the attack surface will help identify the most afflictive attack vectors and it will help in maintaining a good security posture for the organization. Do you feel the same and what tools and processes do you use to continuously monitor and assess the attack surface of your organization?
7. Can you discuss any specific challenges you have faced in managing the attack surface and how you have addressed them?
8. The research also identified that 87% of organizations have at least one risk and implementing proactive controls will minimize the risk of external cyber-attacks. What are your views on implementing proactive controls for the internet-facing assets (external attack surface)?

9. Our research aimed at creating guidelines to prioritize vulnerability to provide the greatest risk reduction with the least effort. How do you prioritize remediation efforts related to the attack surface of your organization?
10. How do you measure the effectiveness of your attack surface management efforts and how do you communicate with senior leadership and stakeholders about the state of your organization's attack surface?