CSEC 793 Capstone in Computing Security
Project Report

A HYPOTHESISED SECURITY MODEL FOR AVAILABILITY

May 2, 2023

Pranav Sarma V.
Department of Computing Security
College of Computing and Information Sciences
Rochester Institute of Technology
ps4554@rit.edu

1 Abstract

The Cybersecurity triad - CIA, which stands for Confidentiality, Integrity, and Availability - has been the main basis to construct security frameworks, which are used to protect any given person/organization against security breaches from a malicious 3rd party. Though there are established models for Confidentiality (Bel-La Pedula) and Integrity (Biba), which are accepted by mostly everyone, that is unfortunately not the case for Availability. Hence in this paper, I will be focusing on the previously suggested security model designs for Availability, their strengths and failures, and my version on what would a good model for Availability would be like.

2 Introduction

The CIA triad, as it is known, is used to refer to the basic components of information security controls in computer systems. These three essential concepts have influenced not just our theoretical knowledge of information security, but also the methods used to build and implement security in businesses. Though it has been almost 50 years since it was first proposed, the CIA triad concept is still central to many security governance standards and codes of conduct that have been adopted by governmental, non-governmental, and private organizations.

A security model gives designers a mechanism to translate abstract claims into security policies that specify the algorithms and data structures required to create hardware and software. These models may be abstract or intuitive/mathematical, but they are all meant to offer a clear set of guidelines that a computer may adhere to. There are about 13 models discussed throughout history, with the most famous ones being the Bel-La Pedula and Biba models. The Bell-LaPadula model focuses on data confidentiality while ignoring the elements of integrity and availability, while the Biba model prioritizes data integrity. Though there have been a lot of other security aspects like IDS/IPS, Trust, and Identity Management - which either prioritizes 2 of the CIA triads or sometimes all 3 of them - there has never been one properly researched for Availability, which is what is being attempted in this paper.

Hence in this paper, an attempt to create a security model is done by - (1) providing proper background about the CIA triad and how it shapes the security mindset of an individual; (2) what a security model is, why is it needed, and 3 famous security models for reference; (3) Explain in depth about Availability, overview of the important factors which affect Availability, and the type of method used to create a model; (4) Explain in detail about the factors which affect Availability, the semantics and syntax of the model; and finally (5) Present my model via formal language theory.

3 Literature Review

- *Samonas, S., and Coss, D. (2014). The CIA strikes back: Redefining confidentiality, integrity, and availability in security. Journal of Information System Security, 10(3).*[1]

  This paper, in brief, discusses the history of CIA from both a practitioner's views and an academician's views - why and how those 3 tenets were created at first, the reports and analysis by security researchers which further stressed its importance in protecting information, the important types of information security models which defined the tenets, how the tenets evolved with time and led to a difference in opinion based on how it was dealt from the military sectors and the commercial sectors, the socio-technical aspects of security and the 3 tenets with respect to relevant papers, and finally how the tenets could be redefined in detail, based on the author's views.

  The history of CIA right from its inception to the current times and how it evolved over the years to still stay as a relevant basement-style knowledge to-go-to for a security researcher/analyst, in protecting information and accordingly creating security models - gave a very good idea of exactly what was needed during each and every time period for the security personnel in the respective time periods with respect to protecting information, and how those tenets were viewed by them and used to create security models to achieve their goals. In short, the paper gave a journal-like peek into how CIA tenets were given birth to and nurtured over the years along with its major feats and accomplishments, to achieve whatever was desired from it. This was really important to understand as an author who is attempting to create a security model for one of the 3 tenets - as to what aspects are we really dealing with, what/how it was done before, and what needed to be understood for attempting to bring up a hypothesis like what this paper suggests.

  The paper also explained how in this recent century, the 3 tenets have some concepts where they overlap one another, and sometimes even all 3 too, like - Trust (Confidentiality and Integrity), Correctness in Specification (Integrity and Availability), and Identity Management (all 3). Though there are papers and models like Denning intrusion-detection model which explains the IDS system and addresses all the 3 tenets in one way or another, there have never been any model or an attempt to understand/define more about Availability only, which is one big reason to choose this topic.

- *Goguen, J. A., and Meseguer, J. (1982, April). Security policies and security models. In 1982 IEEE Symposium on Security and Privacy (pp. 11-11). IEEE.*

It introduces a general concept of security policy and its specifications, mainly explaining - modeling secure systems based on its dynamic security-related aspects/factors, using a general automaton theoretic approach.

The study of abstract machines and automata, as well as the computational problems that can be solved with them, is known as automata theory. It considers the relationships between programs and their requirements to be linguistic relationships. Questions concerning programs and their requirements can be converted to questions about automata by translating them into automata. Automata hold significance in computer theory, compiler construction, artificial intelligence, parsing, and formal verification. It has a close relationship with formal language theory. In this context, automata are utilized as finite representations of potentially infinite formal languages.

- *Stewart, James Michael, et al. CISSP: Certified Information Systems Security Professional Study Guide : Certified Information Systems Security Professional Study Guide, John Wiley and Sons, Incorporated, 2012.*

  The book has been another source of information on what was needed to be known about the important aspects of this paper, like - Availability, CIA triad, and principles of security models and its designs and capabilities. This provides an understanding of the concepts required to know via an academician's lens, compared to a researcher's lens in the firstly mentioned paper.

4 Background

4.1 CIA Triad

The CIA triad, as it is known, is used to refer to the basic components of information security controls in computer systems. These three essential concepts have influenced not just our theoretical knowledge of information security, but also the methods used to build and implement security in businesses.

Though it has been almost 50 years since it was first proposed, along with the introduction of various other Information Security sub tenets since then (Identification, Authentication, Authorization, Auditing, Accountability, Non-repudiation), the CIA triad concept is still central to many security governance standards and codes of conduct that have been adopted by governmental, non-governmental, and private organizations.

Examples: The classification and qualitative assessment of information security risks (see NIST SP 800-30 Rev.1), as well as the implementation of pertinent security measures (see NIST SP 800-53 Rev.3) - are all based on the loss of confidentiality, integrity, and availability of information or information systems. The CIA triad also serves as the foundation for privacy laws and the security of electronically protected health information (see HIPAA).

1. **Confidentiality:** The quality of something which makes sure that private information is not disclosed to any 3rd parties, other than the concerned parties.
2. **Integrity:** The quality of something which makes sure that information is not unnecessarily modified by 3rd parties, other than the concerned parties.
3. **Availability:** The quality of something which makes sure that information is readily available to the concerned parties.

Confidentiality safeguards information from misuse and illegal access. Most information systems contain information that is sensitive to some extent. It could be confidential business knowledge that rivals could exploit, or it could be private information about the staff, clients, or consumers of a company. Companies are frequently attacked as a result of the value that their confidential information usually holds. Threat vectors include layered tactics like social engineering and phishing, as well as direct attacks like stealing credentials and intercepting network traffic. However, not all breaches are deliberate. Common inadvertent breaches include - sending private data through email to the incorrect person, putting private information on open web servers, and leaving private data visible on an unattended computer monitor.

Integrity checks give assurance that the data is accurate and comprehensive. Information must be protected both - when it is stored on computers, and when it is transferred between systems. In order to guarantee integrity, it is vital to not only restrict access at the system level but also to make sure that users of the system can only make changes to the data that they have been given permission to make. Compared to confidentiality protection, data integrity protection goes beyond deliberate violations. Effective integrity countermeasures must also guard against unintended alterations, such as mistakes made by users, or data loss brought on by a malfunctioning device.

Many businesses place high importance on a website's **availability** and responsiveness. Even a little interruption in website accessibility can result in lost sales, disgruntled customers, and reputational harm. Hardware malfunctions, unplanned software outages, and network bandwidth challenges are some of the most fundamental non-malicious threats to availability. Malicious attacks can take many different forms of sabotage, which aim to hurt a company by preventing people from accessing its information system. Another attack that hackers regularly employ is the Denial of Service (DoS) attack, which interferes with

web services. Hence, availability measures ensure prompt and uninterrupted access to the system.
High-availability systems should have a lot of hardware redundancy, with backup servers and data storage always available. It is typical for large enterprise systems to have redundant systems in several physical locations. Network traffic and system performance should be tracked using software tools. Firewalls and routers are two types of defenses against DoS attacks.

4.2 Information Security Models

According to NIST SP 800-95, Guide to Secure Web Services, policies are "statements, rules, or assertions that specify the correct or expected behavior of an entity." An organization's management, protection, and distribution of information is governed by an overall set of directives, regulations, rules, and practices known as an Information Security policy.

Security models offer a mechanism to define security policies in information security. Another definition would be that - A security model gives designers a mechanism to translate abstract claims into security policies that specify the algorithms and data structures required to create hardware and software. These models may be abstract or intuitive/mathematical, but they are all meant to offer a clear set of guidelines that a computer may adhere to, in order to carry out the core security concepts, practices, and procedures that constitute a security policy. Thus these provide a means to better comprehend how an operating system for a computer should be created, in order to support a particular security policy.

According to the reference (Stewart, James Michael, et al. CISSP Study Guide 2012), there are about 13 models discussed, from which 4 of them are the following:

4.2.1 Trusted Computing Base (TCB):

In the old US DoD standard "Orange Book" (DoD Standard 5200.28), hardware, software, and controls come together to form a trusted foundation known as a Trusted Computing Base (TCB) that is used to enact security regulations.

According to the Orange book: The only component of that system that can be relied upon to uphold and implement the security policies is the TCB, which is also in charge of regulating access to the system. Additionally, it is the duty of ***TCB components*** to guarantee that a system operates as intended and that it complies with the security policy at all times. The fictitious line dividing the TCB from the rest of the system is the **security perimeter**. The TCB must establish secure channels also known as **trusted paths**, in order to communicate with the rest of the system. Additionally, the trusted paths guard against compromises that occur as a result of TCB interchanges, for the system users.

The **reference monitor** is the portion of the TCB that verifies access to each resource before granting access requests, and the **security kernel** is the group of components that cooperate to implement reference monitor functions.

The Orange Book had a lot of problems. It was created with the idea of procurement in mind: the default settings of the computers didn't matter as much as how securely the DOD could configure them. Additionally, networking and security issues were not taken into account by the security levels. Now - The Common Criteria - an international framework, has taken the place of the Orange Book.

4.2.2 Bel-La Pedula Model:

In the 1970s, the US Department of Defense (DoD) established the Bell-LaPadula model in response to concerns about the security of sensitive information. The Bell-LaPadula concept is designed to prevent classified information from leaking or being transferred to less secure clearance levels. Lower-classified subjects are prevented from accessing higher-classified things. With these constraints, the Bell-LaPadula model focuses on data confidentiality while ignoring the elements of integrity and availability. In addition, BellLaPadula is the first mathematical model of a multilevel security policy.

The 3 states/properties in which the model works are:
1. The **Simple Security Property** states that a subject may not read the information at a higher sensitivity level (no read up).
2. The * **(star) Security Property** states that a subject may not write information to an object at a lower sensitivity level (no write down). This is also known as the Confinement Property.
3. The **Discretionary Security Property** states that the system uses an access matrix to enforce discretionary access control (No read down and write up)

Because it was created in the 1970s, it does not handle many modern functions, such as file sharing and networking. It also assumes security transitions between secure layers and does not handle covert channels. Because it handles confidentiality well, it is frequently used in conjunction with other models that handle integrity and availability.

4.2.3 Biba Model:

The Bell-La Padula model inspired the Biba model. In reality, Biba appears to be an inverted version of the Bell-La Padula model. The main distinction is their major focus: Biba prioritizes data integrity. Most companies are more concerned with data integrity than confidentiality. Because it focuses on data integrity, the Biba model is a more popular choice for commercial security models than the Bell-La Padula model.

Here are the basic properties of the Biba model state machine:
1. The **Simple Integrity Property** states that a subject cannot read an object at a lower integrity level (no read down).
2. The * **(star) Integrity Property** states that a subject cannot modify an object at a higher integrity level (no write up).

The Biba model has a few flaws, according to critics:
1. It only addresses integrity; it does not address confidentiality or availability.
2. It concentrates on defending objects against external threats while assuming that internal threats are dealt with programmatically.
3. It does not address access control management, and it does not allow you to assign or change the classification level of an object or subject.
4. It has no effect on covert channels.

5 Project Idea:

5.1 Availability as one of the Security Tenets:

A "service" is of no practical utility if no one can utilize it. Availability is a property that allows legitimate principals to access a service in a timely manner whenever they need to. Availability can be expressed numerically as a percentage of the total time period that a service is available.

Reduced availability can occur accidentally (due to hardware, software, or infrastructure failure), or intentionally (due to assaults on the service or infrastructure). The first can be reduced through redundancy, where the likelihood of all backups failing at the same time is ideally very low. The second cause of loss of availability is more concerning in terms of security. A Denial of Service attack occurs when an attacker is able to reduce availability. Malicious availability attacks can target either the service itself (e.g., exploiting a common software defect to cause all backups to fail at the same time) or the infrastructure that supports the service. (e.g., flooding network links between the service and the principal).

Samonas, S., and Coss, D (2014. The CIA strikes back: Redefining confidentiality, integrity, and availability in security. Journal of Information System Security, 10(3).) describes that the relationship between usability and security can be described as strenuous - that the security researchers has discussed the conflict between security and usability with regard to different aspects of authentication, such as password mechanisms (Weir, 2009), and single-factor and two-factor authentication solutions (Gunson et al., 2011). They also mentioned that - users based on empirical research typically chose usability and convenience, instead of security (Weir, 2009; Gunson et al., 2011), and that there are very few

balanced approaches to the development of security and usability (Dhillon et al., 2012), that - Ultimately, usability is linked to productivity through security. A heavy investment in information security can result in lower usability, and therefore a loss in productivity, which can, in turn, have an adverse effect on the business (Cowan, 2012). They cited Cowan (2012) saying - usability is a battle between security and productivity, as security measures can neither be so restrictive that they affect business processes and the flow of information, nor too relaxed, thereby causing harm.

Here in this paper, though usability is shown to be an important part of determining system behavior, it does not occur as a separate factor by itself, but as part of the factors namely - Security, Compatibility, and Data Quality (explained in the next subsection) - where the usability factor in each of them is so important to the system's behavior. In the Security factor, there are numerous software and hardware tools available today that are designed for proper security management against cyberattacks from both inside and outside the firm, and are managed by either personnel of the same company or contractors resourced for this purpose from another company. Hence here, the usability of the tools plays an important role in their performance and overall system behavior. According to the Compatibility factor, security integration and communication between tools play a major role in deciding system behavior, and proper usability helps them to overcome their respective barriers. Even in the Data Quality factor, proper data handling corresponds to proper usability of the tools which helps it perform the necessary tasks to uphold the factor.

5.2 Challenges while ensuring Availability:

Storage failure, Server failure, Network failure, Security failure, Poor data quality, Compatibility failure

5.2.1 Storage failure:

Right now, there is a greater need than ever for large data storage. At an unprecedented rate, businesses and organizations are looking for better ways to store and manage their data. The advent of the internet provided businesses with near-insurmountable amounts of data, most of it valuable. With this massive amount of data, businesses and organizations must realize that their big data capabilities, particularly storage, must be expanded to match.

Examples of storage failure (where the fault remains exclusive to its working) include: Electronic failure, Overheating, mechanical failure - of components and sub-components.

5.2.2 Network failure:

Unshakable Internet connectivity is a non-negotiable business imperative. Users expect it and companies rely on it for transactions and revenue. But the companies do experience network outages. When that happens, depending on the situation, it take days, weeks, or months to isolate network performance problems and fix them. Network failures can be caused by a range of issues, from human error to DDoS attacks, and can cause companies to lose revenue, fast.

Types of problems which cause network failure include- External (Powergrid blackout, Accidents, Natural disasters, human errors), Hardware (server hardware failures, equipment failures, failed firmware patches, equipment overheating), and Software (application/tool failure, generally leading to DDos).

5.2.3 Security failures

A data breach occurs when confidential information is intentionally or unintentionally exposed to unauthorized parties. A cyberattack is an attempt by one party to obtain unauthorized access to another party's computer systems. Businesses are subject to cyberattacks perpetrated by criminal organizations, governments, and private individuals. As days go by and technology gets sophisticated, the number of cybercrime events keep increasing, as scammers target increasingly weak places in corporate security. 53 percent of cyber attacks resulted in losses of 500,000 USD or more, indicating that ransom is a typical motivation. Another purpose to start a cyberattack is to cause harm - some attackers' purpose is to completely destroy systems and data as a form of "hacktivism."

There are various sorts of cyberattacks. Ransomwares are among the most damaging - encrypting data saved on the system until the target business pays a ransom. Other frequent sorts are DDoS, Phishing, Social Engineering, and so on.

There are numerous software and hardware tools available today that are designed for proper security management against cyberattacks from both inside and outside the firm, and are managed by either personnel of the same company or contractors resourced for this purpose from another company. Most of the companies existing today understand the importance of having security tools and measures to protect their data and follow them too, but the cyberattacks keep happening. These happen either due to failure of the secuirty tools in detecting the threat, or due to human errors.

5.2.4 Compatibility failure:

Many security tools on the market today use proprietary interfaces and data exchange languages, but not all are built to the same standards, and there is no common language for

data sharing. Another issue with security integration originates from companies deploying too many security products and services, and the sheer volume of diverse security technologies, along with a lack of native compatibility between them, is one of the most significant difficulties confronting cybersecurity operations today. This is due to the fact that in a company, each new security tool must be integrated with dozens of others, resulting in an increasing number of custom integrations that must be managed between each, growing at a scale that has become unfeasible and overwhelming for cybersecurity engineers and managers to handle security operations efficiently, because if the new tools can't communicate with other platforms or security tools, it makes it more difficult to get a useful view of the true threat.

Multiple security communities are working to address the issue of integration, with a focus on building more common data models, open standards, and open-source tooling that may be utilized across vendors and toolsets. Security teams and security tools will be able to swap out one tool for another more readily if they rely on similar types of data and data models, making it easier to integrate new tools and decreasing vendor lock in.

5.2.5 Poor Data Quality

The causes of poor data quality may appear to be minor issues, but they can quickly become amplified as repeat errors and other types of errors proliferate and compound. It can cause tremendous harm to a company, resulting in poor customer interactions, erroneous data analytics, and poor judgments - all of which impact company performance. The motivations for gathering and retaining low-quality data are straightforward. In general, the issues stem from converting data from one format to another, although there are other causes as well. Poor data quality can result in erroneous data analysis, reduced efficiency of company processes, missed opportunities, resource waste, and, most importantly - reputational damage to the company.

Sources of poor data quality include - Data integration issues, Data-capturing inconsistencies, Poor data migration, Data decay and Data duplication. These could be avoided by having proper data handling, timely and precise data auditing, among others.

5.3 Theory of my model:

I have planned to describe my model and its functions and sub-functions using formal language theory.

In contrast to natural languages such as English, Russian, and others - Formal languages are those that are planned and designed. They are designed to serve a certain function and, as a result, have extremely rigid restrictions from the start. They frequently employ symbols,

numbers, and letters that are not found in natural languages; and they're employed in subjects like algebra, logic, and computer programming.

Alphabets are used in formal languages. Formal languages, rather than "words," use their alphabets to construct strings. It has syntax, (the order and placement of strings are important) and semantics (the strings each have a distinct meaning).

Here, in my model, the syntax used is binary operations using its logical operators - AND and OR operators. The semantics will be explained in detail and defined in the next section "Implementation steps", and all those semantics individually will be of either 2 states - 1 or 0. The relationship between each individual semantics will also be explained both theoretically and formally in the next section "Implementation steps", and all of them will be shown how it affects the overall "Availability" tenet.

One may argue that overall system behavior depends on various factors and sub-factors, other than the ones mentioned in the upcoming section "Implementation steps", and that system behavior cannot be defined with a blanket statement that - they are binary in nature. The reason why I chose to explain and create my model using these factors only (Storage, Network, Security, Compatibility, and Data Quality) was because these were one of the most integral aspects which help in determining the system's state of behavior, and more will be mentioned about it correspondingly in the next section "Implementation steps". For the assumption of the binary state of the chosen semantics, it was done for the sake of simplicity, and if I had to properly explain I had needed more research in order to provide evidence as to how the system behavior would really be w.r.t. various scenarios.

6 Implementation steps

6.1 Storage

A storage device can store information, analyze information, or both. The following fundamental properties of a storage device include: (1) Accessibility describes how data on a device is arranged and how it can be accessed: serial or ad hoc (2) Capacity, which specifies how much storage space a device has (in bytes). (3) Lifespan, which specifies how long data may be retained under specific conditions (in years). (4) Mutability, which describes a device's functions: write, read, or both; and (5) Typology, which defines the types of storage devices: optical, magnetic, semiconductor or electronic, molecular, and so on.

There are approximately fifty times as many stars in the observable universe as there are data. Every day, 2.5 quintillion bytes of human and machine-generated data are created,

and the rate is only increasing. The quantity of data created in 2020 is expected to surpass 35 zettabytes (ZB) (35 trillion GB) but back in 2018 itself, 33 ZB were already attained. Data creation and consumption patterns indicate that the devices and storage media we use will require greater physical space. Even DNA, being a novel data storage medium, is being evaluated as a possible replacement.

A good data storage solution should also be able to provide the required input/output operations per second (IOPS), for data delivery to analytical tools. It should be able to cope with various data models, support both unstructured and structured data, and only operate with encrypted data to aid with privacy protection. Data storage solutions can be classified into 2 types: Warehouse storage and Cloud storage.

6.1.1. Warehouse storage: Its major job is to store and process massive amounts of data. Data warehouses are designed with data retention and processing in mind. It improves accessibility and analysis while also being size-adaptable. Typically, data warehouses rely on large storage capacity that are durable, inexpensive, and efficient.

6.1.2. Cloud storage: Because of its ubiquity, this is a far more recognizable sort of storage. An authorized user can recover data stored in cloud storage from any location with an internet connection. They can get what they need, when they need it, no matter where they are or what device they are using. It eliminates the need for complicated hard drives and the utilization of computers while simultaneously providing significant flexibility, reliability, and security.

For simplicity sake, let us assume the semantics of - warehouse storage factor as "W-ST", cloud storage factor as "C-ST", the backup storage as B-ST, overall storage factor as "ST", and availability of the system as "A".

- In this situation, assuming that company's availability ONLY depends on its storage: **ST = A**
- In companies where only warehouse storage usage is practised: **W-ST = ST = A**
- In companies where only cloud storage usage is practiced: **C-ST = ST = A**
- In companies where both types of storage usage are practiced, and each have their own copies of the same data without a backup: **W-ST + C-ST = ST = A**
  Here, the logical OR operand (+) is used since the company's availability would not get disturbed due to the fact that one of the storage types being a backup for the other
- In companies where both types of storage usage are practiced, and each have their own unique data without a backup: **W-ST * C-ST = ST = A**

the logical AND (*) operand is used because the company is dependent on both storage types to run, and loss of either one of them causes a disruptance in availability.

- In companies where both types of storage usage are practiced, and each have their own unique data along with a backup: **(W-ST * C-ST) + B-ST = ST = A**
  Here, we can see that - due to the presence of a backup, even if either one of the main storage types fail, the availability doesn't get disturbed.

6.1.1 Network

ARPANET (Advanced Research Projects Agency Network) established the first connected computer network in 1969, marking the beginning of contemporary computer networking technology. It was the first to use the TCP/IP protocol suite, which evolved into the Internet. Datapoint Corporation created ARCNET, a communications protocol for local area networks (LANs), in 1986. While ARCNET was popular during its time, it was less reliable and flexible than other systems, particularly Ethernet. Token ring systems gained popularity in the 1980s. This protocol prevents information packet collisions on a network by assuring that only a host with a token can send data and that tokens are released only when data receipt is confirmed. It was introduced in October 1985 and operated at a speed of 4 Mbit/s. A 16 Mbit/s Token Ring was eventually standardized, and its speed was extended to 100 Mbit/s near the conclusion of its life. In a LAN, the fiber distributed data interface (FDDI) employs optical fiber to transmit data. It was capable of rates of up to 100 Mbit/s. It, too, is a ring-based token network, but it employs a protocol developed from IEEE 802.4. Bob Metcalfe at Xerox PARC invented Ethernet in 1973, although it wasn't patented until 1975. It took another five years for the open Ethernet protocol to be standardized as IEEE 802.3 in 1983. The earliest Ethernet system, which used coaxial cable as a shared medium, had speeds of 2.94 Mbit/s. Ethernet now has progressed over time to twisted pair or fiber optic cables, as well as switches, allowing it to rise in speed, which is now a blistering 40 Gb/s.

As mentioned in the previous section, network failure can be classified by 3 ways:

1. Hardware:
These type of network failures happen due to problems arising from hardware components, thus affecting availability of data in the network - both internal and external ones. Hardware failures occur when components of your IT infrastructure cease to function. Hardware failure can occur for a variety of causes, including power grid voltage spikes, water damage, or electrical component failure due to age or a lack of maintenance. Servers and equipment such as power supply, motherboards, and hard drives are examples of components that can fail. To deal with this, one must have an abundance of spare hardware components and a reliable service provider. The hardware should also be properly kept to ensure that there is no wear and tear. Overheating can cause significant harm to the server's architecture, as well as odd failures. If these systems do not function properly, the hardware in that room

will be damaged by heat, reducing the hardware's lifespan. As a result, a well-functioning cooling system is essential. (Researchers discovered that running servers in a helium gas environment can minimize the resistance of the small fans inside the server to run without air resistance, hence decreasing wear, and can also manage the heat that emenates and damages the servers.) Maintenance of fans, ducts, and filters should be performed on a regular basis on all of the servers and rack-mounted equipment. It is also critical to replace equipment when it nears the end of its useful life, and to make adequate replacements before running into any more serious problems later on. Manufacturers will issue firmware upgrades and patches on a regular basis to keep devices working smoothly. Failed firmware patching can occur when a device loses power in the middle of what is being done or when a communications cable is mistakenly disconnected in the middle of the process. Sometimes a faulty firmware file can be flashed to a device, or an incompatible device accepts a flash, rendering it useless.

2. Software:
Failures in software can happen for a variety of reasons. A license can expire, a configuration file can become corrupted or go missing, a faulty software update can cause problems, a software defect can cause problems, and so on. Sometimes a software update is put into the environment but is not vetted, resulting in defects and mistakes. To avoid such failures, it is necessary to follow a proper testing and validation process for all software changes, including monitoring and documentation, pre and post updates.

3. External:
These kind of errors are totally aside from blaming the fault on the hardware and software equipments of the company, and more on - human errors while handling those equipment, and totally from outside the company's environment. A faulty power substation or transformer can produce sudden power grid blackouts, knocking out electricity to the entire area. These are side effects that can have an impact on data availability. Repairing it might sometimes take a long time, depending on the situation that produced it in the first place. Alternative energy sources are one solution to this type of challenge. Few company members might still fall prey to Social Engineering techniques and hence compromise their account credentials to malicious people, hence compromising the data they have access to and possibly, the company's network system. These can be solved by training the employees with proper security etiquettes - like having a big length password, not reusing passwords, making sure passwords are updated regularly, etc. IT Security Professionals might misconfigure security settings while handling security tools designed to protect the company from outside parties, or even might have missed something critical while performing security procedures - thus jeopardizing the company to data availability failure. Notably, these last 2 instances also overlaps with the security factors, which comes up next.

In this case, let us assume the semantics of - the hardware factor as H-N, software factor as

S-N, the external factor as E-N, overall network availability factor as N, and availability as A.

- Just in this case, let us assume that the availability of the company resides ONLY on the Network factor: **N = A**
- As discussed in the above theory portions: **H-N * S-N * E-N = N = A**
  Here, the logical AND (*) operator is used - indicating that all 3 network factors are important to be addressed in terms of upholding a company's data availability, and the fall of even 1 of them cannot sustain availability.

6.1.2 Security

In the 1970s, researcher Bob Thomas developed Creeper, a computer software that could roam throughout ARPANET's network, leaving a breadcrumb trail wherever it went. Ray Tomlinson, the creator of email, created Reaper, a program that chased and removed Creeper. Reaper was the first computer worm, as well as the first example of antivirus software and the first self-replicating program. By the mid-1990s, network security threats had grown tremendously, necessitating the mass production of firewalls and antivirus software to protect the public. As the internet evolved, so did information security, but regrettably, so did viruses.

Keeping the IT device network and infrastructure secure also involves protecting the company from potential downtimes caused by cyberattacks, failing network devices, and data loss, etc. Spending on IT security software, antivirus, antimalware, and other security solutions will rise in tandem with the evolution of IT and communication technologies, and the adoption of Internet-ready devices has become a standard across many industries. Zion Market Research predicts massive growth in the cyber security market in 2019. The cybersecurity market, which was valued at 105.45 billion USD in 2015, is predicted to grow to 181.77 billion USD by the end of 2021. IDS/IPS, antivirus software, firewalls, VPNs, and others are namely a crucial applications. It is also vital to have a sufficient number of cybersecurity engineers in the company/resourced from a different company who can operate the security tools with the most efficiency. It is also to be noted that AI can jump into the bandwagon sooner or later by making the engineer unnecessary and taking its own decisions from the data it collects, favourable to the company.

Few company members might still fall prey to Social Engineering techniques and hence compromise their account credentials to malicious people, hence compromising the data they have access to and possibly, the company's network system. These can be solved by training the employees with proper security etiquettes - like having a big length password, not reusing passwords, making sure passwords are updated regularly, etc. IT Security Professionals might misconfigure security settings while handling security tools designed to

protect the company from outside parties, or even might have missed something critical while performing security procedures - thus jeopardizing the company to data availability failure. It is also highly recommended to follow security standards based on the company's kind of industry, and its geographical region and laws. Famous security standards include NIST, GLBA, HIPAA, etc. Notably, these same instances also overlap with the external type network factors, which came earlier this section.

In this case, let us assume the semantics of - the security tools as T-SEC, engineer's work as E-SEC, overall security factor as SEC, and availability as A.

- Just in this case, let us assume that the availability of the company resides ONLY on the Security factor: **SEC = A**
- Considering the security tool definitely needs an engineer to function it and make decisions out of it: **T-SEC * E-SEC = SEC = A**
  Here, the logical AND (*) operator is used - indicating that the security tools are dependent only on how well the engineer uses it to uphold data availability
- Considering the security tool is run fully by AI: **T-SEC = SEC = A**
  Here it indicates that the availability of the company totally depends only on the AI-run security tool

6.1.3 Compatibility

Multiple security communities are working to address the issue of compatibility, with a focus on building more common data models, open standards, and open-source tooling that may be utilized across vendors and toolsets. Security teams and security tools will be able to swap out one tool for another more readily if they rely on similar types of data and data models, making it easier to integrate new tools and decreasing vendor lock in.

Data from your cybersecurity tools can flow into other apps and systems via integrations, enabling integrated security workflows. The integration capabilities of free programs are often limited, although some may supply the source code so that users can construct their own integrations. Before selecting a solution, its recommended to consider a comapny's integration needs, as well as their in-house development capabilities. Examples of commonly used open source security tools include - Wireshark (Packet analysis), NMap (Network discovery), Snort (IDS), pfsense (firewall) and OpenVAS (vulnerability scanning).

Considering the semantics of - overall compatibility factor as "C", the compatible security tools as "T-C", and availability as "A:"

- Just in this case, let us assume that the availability of the company resides ONLY on the compatibility factor: **C = A**

- As discussed in above theory portions: **T-C = C = A**
  Here there are not much important factors to consider, and to expand "T-C" with all the compatible security tools used in the company would be futile, and hence assumed in this paper.

6.1.4 Poor Data Quality:

Since the beginning of the twenty-first century, several significant technological changes have happened in the information technology business, including cloud computing, the Internet of Things, and social networking. Because of the advancement of these technologies, the volume of data continues to grow and amass at an unparalleled rate. All of the technologies mentioned above herald the arrival of big data.

Researchers and decision-makers have increasingly understood that rapidly obtaining and analyzing big data from diverse sources and for varied purposes provides benefits for understanding customer wants, enhancing service quality, and forecasting and mitigating hazards. However, in order to generate value from big data, the use and analysis of big data must be based on accurate and high-quality data.

Research on data quality started abroad in the 1990s, and many scholars proposed different definitions of data quality and division methods of quality dimensions. According to the U.S. National Institute of Statistical Sciences (NISS) (2001), the principles of data quality are: 1. data are a product, with customers, to whom they have both cost and value; 2. as a product, data have quality, resulting from the process by which data are generated; 3. data quality depends on multiple factors, including (at least) the purpose for which the data are used, the user, the time, etc.

Sources of poor data quality include - Data integration issues, Data-capturing inconsistencies, Poor data migration, Data decay and Data duplication. These could be avoided by having proper data handling, timely and precise data auditing, among others.

When data is acquired from many databases that do not integrate with the organization's database, **Data integration issues** might arise. Converting data from one format to another frequently results in errors, and conversion challenges can become even more complicated if data from an older legacy system is converted for storage in a NoSQL system. **Data capture** is the process of converting information stored on a document into data for computer storage. Because of the variances and inaccuracies that can emerge due to the dual-department scenario, a company with two or more departments that employ distinct formatting procedures should definitely be concerned with data quality. **Poor data migration** is prevalent when data is transferred from a legacy system to a new database or to the cloud. Moving data to a new system is fraught with danger. Some data values

may be missing or irregular. If the data is not of high quality from the start, new issues such as data corruption and missing data can occur. **Data decay** is the degradation of data quality. It is frequently a manifestation of old, obsolete information, and sometimes a database crash is also referred to as a type of data decay. **Data duplication** has the potential to bias business intelligence. When duplicated data is used for statistical purposes, difficulties might arise. Furthermore, if duplicated data is correct in one spot but incorrect in another, challenges could arise.

In most cases, poor-quality data is the result of a lack of set norms and procedures. Implementing **data-handling standards** aids in the production of high-quality data, and workplace culture must support intelligent data rules designed to standardize data forms and avoid bad data accumulation. Creating distinct field markers on data, as well as defining data ownership and data logs, are some examples of data rules in a company. Continuous, real-time **data maintenance** prevents the usage of stale and decayed data, identifies data issues, and guarantees that company employees and customers are working with valuable data. As a result, it is critical to regularly audit the quality of a company's data. Lastly, it is important to check if the data is - **accurate**, **complete** with required information, **consistent** everywhere, **unique**, and if there is **timeliness** in handling old, unnecessary data.

Considering the semantics of - overall data quality factor as "DQ", data handling standards factor as "DS-DQ", data maintenance factor as "DM-DQ", the other 5 miscellaneous factors as "M-DQ", and availability as "A":

- Just in this case, let us assume that the availability of the company resides ONLY on the data quality factor: **DQ = A**
- As discussed in above theory portions: **DS-DQ */+ DM-DQ */+ M-DQ = DQ = A**
  Here, I have used either of both logical AND (*) and logical OR (+) operators because the result is very much reliant on the situation - that the company might need the co-operation of all the factors, or just 2 of them, or just 1 of them - to sustain company availability. This would need more analysis and research separately as to what operand would need to be definitely used. Till then, there will be 4 different results (++, +*, *+, **).

7 Results:

Mathematical equations formed from each decided factor:

- Storage: ST = (W-ST * C-ST) + B-ST

- Network: N = H-N * S-N * E-N
- Security: SEC = (T-SEC * E-SEC)—(T-SEC)
- Compatibility: C = T-C
- Data quality: DQ = DS-DQ */+ DM-DQ */+ M-DQ

All these factors are needed to uphold a company's data availability.
Hence: ST * N * SEC * C * DQ = A

This can be expanded as: ((W-ST * C-ST) + B-ST) * (H-N * S-N * E-N) * (T-SEC * E-SEC)—(T-SEC) * (T-C) * (DS-DQ */+ DM-DQ */+ M-DQ) = A

Here, every single main factor is connected together with logical AND (*) operand because everything together is needed to uphold company availability and not just 1 of them.

8 Future Work:

In my paper I had classified the main factors to uphold data availability into 5 - Storage, Network, Security, Compatibility and Data quality. These factors were, in fact, existent problems which most commonly plagued a company's data availability. While implementing my mathematical equations to relate them with availability, I realised that there could also be many other ways of classifying the main factors, like - Software, Hardware and Humans, etc. Including that, many of my selected factors, more than the one I had mentioned - had overlapped one over the other. Hence there are N number of classifications which can be taken into consideration.

Another thing which I realised while researching about my selected factors, was that - there were so many sub-factors which can affect availability of a company. Though they may not be that big to prioritise if seen separately - added together - they form a big chunk which would need attention while truly discussing and deciding on a company's availability in a real time company environment.

Lastly - While coming up with the mathematical equations, I noticed that there were moments when sub factors react with each other, more than just an AND and OR operator. In the sense, the situation is very much a part of the requirement to create a mathematical equation by itself to truly map out a practically working equation. But also, this would lead to also including the sub-factors which were less important to be included in the main equation. Thus for the sake of simplicity in this paper and the complexity to involve all those mentioned factors into main consideration, I had to avoid mentioning them as part of the equation.

9 Conclusion:

Though there have been research papers and publishings about Confidentiality, Integrity, and aspects that overlap each other and sometimes all 3 too; there had not been one on Availability and with this paper, I was hoping to create a precursor reference for future researchers who are interested in getting into this unexplored topic of creating security models for Availability, and for Availability as a security aspect itself too. Here I have created a relationship between 5 major information security factors which affect Availability (Storage, Network, Security, Compatibility, and Data Quality) and Availability itself, using formal language theory. I have also provided the background information and history of each of the factors, along with a background on general topics which is integral to my main contribution like - Information Security Models and the CIA triad - for the reader to get an overall idea on how to approach this topic and idea.

10 Acknowledgment

I thank both my advisors - Prof Justin Pelletier and Prof Sumita Mishra - for their unwavering guidance and unconditional support in making me complete my capstone the way it is right now. I would also like to thank my parents, peers and well-wishers who have been with me on this journey and made it the success it is.

References

[1] S. Samonas and D. Coss, "The cia strikes back: Redefining confidentiality, integrity and availability in security.," *Journal of Information System Security*, vol. 10, no. 3, 2014.

[2] A. J. A. Wang, "Information security models and metrics," in *Proceedings of the 43rd Annual Southeast Regional Conference - Volume 2*, ACM-SE 43, (New York, NY, USA), p. 178â184, Association for Computing Machinery, 2005.

[3] D. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 222–232, 1987.

[4] W.-P. Lu and M. Sundareshan, "A model for multilevel security in computer networks," *IEEE Transactions on Software Engineering*, vol. 16, no. 6, pp. 647–659, 1990.

[5] C. Aaron Estes, *Biba Integrity Model*, pp. 81–81. Boston, MA: Springer US, 2011.

[6] F. Alt and E. von Zezschwitz, "Emerging trends in usable security and privacy," *i-com*, vol. 18, no. 3, pp. 189–195, 2019.

[7] Amarti, "How Organisations Store Their Data — Amarti," 3 2022.

[8] A. Anžel, D. Heider, and G. Hattab, "The visual story of data storage: From storage properties to user interfaces," *Computational and Structural Biotechnology Journal*, vol. 19, pp. 4904–4918, 2021.

[9] A. S. Bhadouria, "Study of: Impact of Malicious Attacks and Data Breach on the Growth and Performance of the Company and Few...," *ResearchGate*, 9 2022.

[10] L. Cai and Y. Zhu, "The challenges of data quality and data quality assessment in the big data era," *Data Science Journal*, May 2015.

[11] E. C. Cankaya, *Bell-LaPadula Confidentiality Model*, pp. 71–74. Boston, MA: Springer US, 2011.

[12] E. Cronin, *Availability*, pp. 68–69. Boston, MA: Springer US, 2011.

[13] S. De Capitani di Vimercati and P. Samarati, *Clark and Wilson Model*, pp. 208–209. Boston, MA: Springer US, 2011.

[14] D. International, "Causes of Network Failure: How Retail Businesses Can Eliminate Downtime," 8 2019.

[15] K. Fenrich, "Securing your control system: the" cia triad" is a widely used benchmark for evaluating information system security effectiveness," *Power Engineering*, vol. 112, no. 2, pp. 44–49, 2008.

[16] K. D. Foote, "The Impact of Poor Data Quality (and How to Fix It) - DATAVERSITY," 3 2023.

[17] J. Fruhlinger, "The cia triad: Definition, components and examples," *CSO Online*, 2020.

[18] R. Heymsfeld, "Confidentiality, Integrity and Availability 8211; The CIA Triad," *CertMike*, 8 2018.

[19] Meenakshi, "Data Failure for Storage devices and solutions to minimise data loss," 9 2022.

[20] J. G. J. Mesajuer, "Sri international menlo park ca 94025,"

[21] M. Nieles, K. Dempsey, V. Y. Pillitteri, *et al.*, "An introduction to information security," *NIST special publication*, vol. 800, no. 12, p. 101, 2017.

[22] J. M. Stewart, E. Tittel, and M. Chapple, *CISSP: Certified information systems security professional study guide*. John Wiley & Sons, 2011.

[23] C. Tozzi, "6 Threats to the High Availability of Your Data (and How to Solve Them)," *Precisely*, 11 2022.

[24] "Types of Security Models," *bartleby*, 12 2021.

[25] "Data Availability: Ensuring the Continued Functioning of Business Operations," 7 2022.

[26] "Beginnerâs guide: Open source intrusion detection tools," 5 2023.

[27] B. Violino, "7 top challenges of security tool integration," 2 2022.

[28] W. contributors, "Automata theory," *Wikipedia*, 4 2023.

[2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] [14] [15] [**?**] [16] [17] [18] [**?**] [19] [20] [21] [**?**] [**?**] [22] [23] [**?**] [24] [25] [26] [27] [28]