# Introduction

## Background

Throughout the last ten years, mass cyberattacks against major organizations have amplified. Security breaches are the most prominent cause for attacks being allowed to happen. Although different types of organizations become victim of cyberattacks, an analysis of data breaches experienced in multiple organizations established that medical organizations and BSOs are the least prepared against attacks . Furthermore, the latest reports confirm of almost half of businesses have been confronted with cyberattacks since companies are settling into the new normal. Cyberattacks have grown in frequency and severity since the pandemic. Lallie, Harjinder Singh et al observed that there appears to be a loose correlation between the announcement and a corresponding cyber-attack campaign that utilizes the event as a hook to increase the likelihood of success . Thus, to minimize or eliminate the damage caused by cyberattacks, software security has to be addressed during software development.

Security breaches are caused by security vulnerabilities in source code introduced by software developers when creating software, and therefore developers are often blamed for vulnerabilities . However, application security is primarily performed by security experts causing a separation between security and development. As a result, the probability of insecure software is increased .

Writing secure code is critical with the prevalence of security vulnerabilities. To achieve this, developers need to be aware of the potential vulnerabilities they might introduce when developing software features and understand how to mitigate them. Still, the knowledge and skills to produce secure software are lacking and often are not taught in computing curricula despite the existence of secure coding guidelines .

Such secure coding guidelines are provided by the Open Web Application Security Project (OWASP). For instance, the OWASP Top Ten brings awareness about the most common security vulnerabilities found in software. Furthermore, security standards such as the OWASP Application Security Verification Standard (ASVS) and the Mobile Application Security Verification Standard (MASVS) enable organizations and developers to produce and maintain secure software.

To address security during software development, this study will implement a secure agile Software Development Life Cycle (SDLC) with the OWASP Security Knowledge Framework (SKF) to develop CheFeed, a mobile application developed for this thesis. SKF is an expert system application that uses secure coding guidelines such as the OWASP ASVS and OWASP MASVS to assist developers in pre-development and post-development phases to create secure-by-design software.

## Aims and Benefits

### Aim

Security vulnerabilities found in applications introduced by software developers are the cause of many security breaches and data theft. To reduce the introduction of security vulnerabilities security must be addressed during software development. This study aims to implement a secure agile SDLC with testable security controls configured through SKF.

### Benefits

This study will provide insights into the process of developing software in an agile SDLC where security is addressed from the beginning. It provides a clear presentation of the benefits of using SKF to configure feature sprints where security should be in place. Students and software developers may benefit from understanding how secure code guidelines can be used. Furthermore, the implementation of SKF for a secure SDLC could serve as a tool for future studies to build upon the process presented in this study.

## Scope

The scope of this study is limited to configuring MASVS Level 1 (MASVS-L1) security controls with SKF for the development of CheFeed. MASVS-L1 ensures that mobile applications cohere to the best security practices concerning code quality, handling of sensitive data, and interaction with the mobile environment.

This project is a group effort where two fellow students will contribute to the development of CheFeed. Briefly their study contributes to the following:

* The development of the API and database design by Stephanus Jovan Novarian in his thesis .
* The development of a sentiment analysis model which uses recipe reviews as its input by Ikshan Maulana in his thesis .

# Theoretical Foundation

## Information Security

This section defines information security, specifically on how it is different from cybersecurity. Moreover, the section will discuss how the evolution of computers and the World Wide Web has introduced challenges to information security. Lastly this section defines what secure by design means.

### Defining Information Security

It is critical to understand what precisely is implied when discussing security. Therefore, we have to properly define it. For instance, the terms cybersecurity and information security are commonly used indistinguishably. The Cambridge Dictionary defines information security as , and defines cybersecurity as . The definition for cybersecurity describes a much larger scope for security.

Solms et al supports this argument and reasons that information security is solely about securing the information, generally referred to as the asset, from potential threats posed by inherent vulnerabilities. Furthermore, they outlined that cybersecurity goes beyond protecting assets. Cybersecurity includes the insurance of those that operate in cyberspace in addition to any of their assets that can be achieved through cyberspace. Although the definition of information security and cybersecurity overlap each other, the latter is much more extensive in its definition. Overall, security is about securing assets against the most probable types of attacks, to the best ability .

Information security can be defined as the protection of information from potential abuse subsequent from various threats and vulnerabilities . In a general sense, security means protecting our data and systems assets from whoever who intends to misuse it. It includes several aspects of business, involving financial controls, human resources and protection of the physical environment, as well as health and safety measures . Security strives to secure ourselves against the most likely forms of attack, to the best ability.

## Fundamentals of Information Security

Security is built on top of well established principles like the Confidentiality, Integrity and Availability (CIA) triad and the gold standard. This section discusses those principles as well as the role of cryptography. In addition, the software development life cycle is discussed and the technologies and languages.

### The Confidentiality, Integrity, and Availability Triad

Security can be complicated as discussed in Section [[sec:challenges-in-is]](#sec:challenges-in-is). Nonetheless, as described in Table [[fig:cia]](#fig:cia), the CIA triad, provides a model to think about and discuss security concepts. It is commonly discussed in the information security literature . The CIA triad is commonly referred to as tenets of information security. Information assets that are tied to an application can be associated to a specified CIA requirement represented as a number or values suchlike, high, medium, or low, which can be determined through risk analysis .

#### Confidentiality

Confidentiality describes the capability to secure assets from parties that do not have the authorization to view.

#### Integrity

The second concept describes the ability to prevent data from being changed in an unauthorized or undesirable manner. Integrity preserves the consistency of information both internally and externally.

#### Availability

The final concept describes the ability to access data when required.

### The Gold Standard

#### Authentication

Identification is the claim of identity by a person, process, or other entity without implying the authenticity of the claim or privileges that could be affiliated with the identity. Many methods exists to claim our identity such as name abbreviations, fingerprints, portraits, and many more.

Authentication is the procedure used to validate whether the claim of identity is correct. A real world example of authentication would be the usage of a username and password combination inside an application. Depending on the security level required of an asset, more factors can be used for the authentication mechanism, also known as multifactor authentication.

#### Authorization

Besides claiming an identity and confirming the validity of that claim, we need to decide what the party is allowed to do and if access to specific resources are allowed or denied.

#### Principle of least privilege

An important authorization concept is the principle of least privilege. It mandates that only the bare minimum of access to a party should be allowed to function. As an example, a user account is only granted the access needed to perform their routine work. It is a very simple security measure that requires minimal effort, and it is highly effective.

##### Access control

At a high level, access control is about restricting access to a resource. Access control can be divided into two groups to either improve the design of physical security or cybersecurity. Generally, four basic actions can be performed:

* allowing
* access
* denying access
* limiting access
* revoking access

Most access control issues or situations can be described through these. Moreover, users that are not authorized should not be granted access. Therefore, it is best practice to disallow access by default.

There are two main methods that can be considered to implement access controls: access control lists (ACLs) and capabilities. ACLs, often referred to as "ackles", are a very common choice of access control implementation. Typically, ACLs are implemented in the file systems on which our operating systems run and to control the flow of traffic in the networks to which our systems are connected. A capability-based approach to security uses tokens that manages our access. A good analogy would be the usage of a personal badge that grants access to certain doors inside a building. Notably, the right to access a resource is based completely on possession of the token, not who possesses it.

#### Auditing

After going through the process of identification, authentication, and authorization, it is important to keep track of the activities that have occurred. Despite access being granted to the party, it is important that the party behaves according to the rules as it concerns to security, ethics, business conduct, and so on. With an abundance of digital assets it has become a vital task to ensure that rules set forth are abided by.

### Cryptography

Cryptography is the science of ensuring that assets are kept secure. The foremost security measure allowing cryptography is encryption, and often the terms are used interchangeably. Although in reality, encryption is a subset of cryptography. Encryption is the transformation of plaintext into ciphertext.

Cryptology is not a recent invention. At the very least cryptology can be traced back as far as 2500 years and was considered an obscure science. It was well established with both ancient Greeks and Romans who practiced different forms of cryptography. A classic example of ancient cryptography is the Ceasar cipher as seen in Figure [[fig:ceasar-cipher]](#fig:ceasar-cipher). After the fall of the Roman Empire, cryptology was flourishing in the Arabic world .

Without cryptography, much of the internet-based activities we benefit from today would be at great risk. In fact, cryptography is essential in computing, networking and the great set of transactions that take place over such devices in everyday life. Cryptography has permitted us to become a very network-centric society. Data can be protected at rest, in motion, and to a certain extent, in use, because of cryptography. Thus allowing us to securely communicate and perform transactions when sensitive data is involved.

The process of encrypting plaintext and decrypting ciphertext is described as a cryptographic algorithm. In order to either encrypt or decrypt a message, cryptographic algorithms commonly use a key, or multiple keys, with a range of possible values for the key referred to as the keyspace. The harder the keyspace, the harder it is to decrypt the message. We will take a brief look at some popular cryptographic algorithms.

#### Symmetric cryptography

Symmetric cryptography, also referred to as private key cryptography, utilizes a single key for both encryption of the plaintext and the decryption of the ciphertext as can be seen in Figure [[fig:sym-crypt]](#fig:sym-crypt). A symmetric cipher only works if both the sender and the receiver are in possession of the same key to unlock the cipher. Therefore, everyone who uses a symmetric cipher must have the same set of keys and must use them in the correct order .

#### Asymmetric cryptography

When a different key is used for encryption and decryption, we have an asymmetric system in place. Asymmetric cryptography can also be referred to as public key cryptography. Asymmetric cryptography relies on a public key to encrypt data from the sender, and a private key to decrypt data that arrives at the receiving end as seen in Figure [[fig:asym-crypt]](#fig:asym-crypt). Due to the mathematical complexity of the operations to create the private and public keys, no method exist at present to reverse the private key from the public key.

#### Hash functions

Unlike both symmetric and asymmetric cryptography, which relies on keys for encryption and decryption, there are algorithms that do not require keys, known as hash functions. Hash functions generate a generally unique and fixed-length hash value, referred to as a hash, based on the original message as seen in Figure [[fig:hash]](#fig:hash). Any form of change to the message will change the hash as well. Furthermore, hash functions do not allow for contents of the message to be read, though it can be utilized to determine the confidentiality of the message. Some hash algorithms include: Message-Digest 5 (MD5), MD2, MD4, SHA-2, and RACE.

##### Digital signatures

A good example of where hash functions are utilized are digital signatures. To detect any changes to the content of the message, digital signatures make it possible to sign a message to ensure the authenticity from the sending party. This is accomplished by generating a hash of the message, and then use the senders private key to encrypt the hash, thereby creating a digital signature. The receiving party can use the sender’s public key to decrypt the digital signature, thereby restoring the original hash of the message.

Digital signatures are now recognized as legally binding in many countries, allowing them to be used for certifying contracts or notarizing documents, for authentication of individuals or corporations, as well as components of more complex protocols. Broadly speaking, a digital signature is analogous to a handwritten signature, that provides much stronger security guarantees .

###### Certificates

Another form of cryptography for message signing, is the usage of digital certificates, commonly known as certificates. Certificates link together a public key and an individual, typically by taking the public key and something to identify the individual, suchlike a name and address, and having them signed by a certificate authority (CA). A CA is a trusted entity that is responsible for digital certificates. The advantage of using a certificate is that it provides verification that a public key actually is associated with a particular individual.