Contents

[CHAPTER 1: PROPOSAL 2](#_Toc62441377)

[1.1. Introduction 2](#_Toc62441378)

[1.2. Background 2](#_Toc62441379)

[1.3. Problem Statement 3](#_Toc62441380)

[1.4. Aims 4](#_Toc62441381)

[1.5. Objectives 4](#_Toc62441382)

[1.6. Signiﬁcance of the Project 4](#_Toc62441383)

[1.7. Methodology 4](#_Toc62441384)

[1.8. Scope of the Project 6](#_Toc62441385)

[1.9. Deﬁnition of Key Variables 6](#_Toc62441386)

[1.10. Conclusion 7](#_Toc62441387)

[CHAPTER 2: LITERATURE REVIEW 8](#_Toc62441388)

[2.1. Introduction 8](#_Toc62441389)

[2.2. Synthesis of literature 8](#_Toc62441390)

[2.3. Conclusion 8](#_Toc62441391)

# CHAPTER 1: PROPOSAL

## Introduction

File integrity monitoring is simply the technique to keep the watch of the data, who is going to access the data, who has made changes in the data, and how the data has been changed. Integrity is one of the pillars of the CIA triad of information security, hence the scope of this project is cyber security. To understand the process easily let us first break down the term. Here file means data. Keeping the integrity of the data means protecting the data from any foreign changes. And Monitoring means watching the data closely to know whether the integrity of data is maintained or not [1].

## Background

In the Information Technology and Cyber Security industries, the most important asset to any firm are the files of its clients. With hackers and attackers developing new ways of illegally gaining access to organisation’s files, software developers are continuously coming up with counter-measures to detect unauthorised users having access to files as well as contain or deter the impact of system intrusions. File integrity monitoring (FIM) is an internal control or process that performs the act of validating the integrity of operating system and application software files using a verification method between the current file state and a known, good baseline. This comparison method often involves calculating a known cryptographic checksum of the file's original baseline and comparing with the calculated checksum of the current state of the file. Other file attributes can also be used to monitor integrity. Generally, the act of performing file integrity monitoring is automated using internal controls such as an application or process. Such monitoring can be performed randomly, at a defined polling interval, or in real-time [2].

The file integrity monitoring solutions also involve examining of the files and displays information about when it was changed, how it was changed, who changed it, and what can be done to restore those files if modifications are unauthorized.

There are three primary components to FIM:

* A Database: This database stores information on the original state of your files and configurations as cryptographic hashes.
* Agents: These technical components measure your hardware and applications and send data back to your database for comparison.
* User Interface: This is the interface for administrative users, which serves as the centralized portal for reporting, evaluation, change monitoring, and change control [3].

File integrity monitoring technology is considered as a major part of cyber security processes and technology, owing to its capabilities to scan, analyse, and report unexpected changes to important files in an IT environment such as operating system (OS), database, and application software files. Several benefits offered by file integrity monitoring include protected IT infrastructure, robust real-time change detection engine, unified security posture, and many others, which are shifting the attention of several end-user industries and driving the global file integrity monitoring market growth.

File integrity monitoring (FIM) is one of the security components that can be implemented in host environment. As a part of host based intrusion detection (HIDS) components, FIM should play a big role in detecting any malicious modification either from authorized or unauthorized users on their contents, access control, privileges, group and other properties. [5]

Favourable government regulatory compliance objectives, such as Payment Card Industry Data Security Standard (PCI-DSS), and Federal Information Security Management Act (FISMA) have led to the file integrity monitoring as one of the major requirements and acts as a major factor driving the global file integrity monitoring market growth. In addition, rising sophistication of cyber-attacks and increasing threats to IT infrastructure are also among some of the major factors driving the global file integrity monitoring market. However, high cost associated with file integrity monitoring advance solutions and financial constraints among SMEs are some of the major challenging factors that can hamper the market growth.

The global file integrity monitoring market is segmented based on component, installation mode, deployment, enterprise size, industry vertical, and region. Based on component, it is bifurcated into software and services. Based on installation mode, it is classified into agentless and agent-based. Based on deployment, the market is categorized into cloud and on-premises. Based on enterprise size, it is divided into large enterprises and small & medium enterprises (SMEs). Based on industry vertical, it is segregated into BFSI, manufacturing, government & defence, media & entertainment, retail & e-commerce, healthcare & life sciences, IT & telecommunications, and others. Based on region, the market is analysed across North America, Europe, Asia-Pacific, and LAMEA [4].

## Problem Statement

Health institutions and regulatory organisations such as TIMB store critical information about their clients on servers mounted in their data centres. Each server in turn serves a factor of virtualized environments continually spawning incomprehensible numbers of workloads that complete a function then die. A single compromised environment could lead to critical data loss and financial losses. The need for scalable security monitoring in such dynamic environments has become a critical need.

## Aims

This project aims to provide system and device administrators with a centralised tool where they can be able to monitor the files kept on their computers. Each files’ existence or lifetime on the computer can be monitored for any alterations or illegal access. Once a file is flagged to have been altered, the tool will send an alert via the appropriate channel to the responsible administrative agent to make changes or take precautionary action against the attacker.

## Objectives

1. To generate and store a baseline for all the files to be monitored
2. To verify the baseline of a registered host by scanning the monitored files and comparing the hash and HMAC against the baseline stored in the database
3. To display the baseline status of a registered host
4. To generate an alert which could then trigger some preventative action if a compromised file is detected

## Signiﬁcance of the Project

With the rapid increase in workflow in the workspace, there is a huge demand for automation in order to cope with and ease the pressure on responsible personnel. This solution does that by automating the whole process of monitoring a host machine’s files and triggering the necessary action whether a file has been discovered as compromised or not. This means IT and Cyber Security personnel need only to perform analytical tasks and not carry out the whole process of keeping an eye on each individual file.

## Methodology

This section describes and clarifies the entire FIM process from the initial setup stages to the point where a file is flagged as compromised and recommended action taken to restore the previous state of the file. As mentioned before that FIM is a process, it is broken down into small sequential stages as follows:

1. *Authentication of Users*

As part of every basic secure system, we would need to know which user performed what action on what file at what time. The first step is to have privileged users register or sign up onto the system. Having users sign up onto the system allows for analytics at a later stage in the project or simply enables for more efficient activity logging.

1. *Identification of Files to be monitored*

Once a user has been authenticated to access the system, the next step is for them to identify the files they would like the system to monitor. As different departments have different personnel handling different workloads and files, this step would need to be divided into departmental level. Responsible personnel locate the directory that they would like for the tool to monitor. Having each department monitor their own files allows for more collaboration between members. However, the overall FIM process would need to be overseen by the IT engineer responsible for managing the whole process.

1. *Baseline Files to be monitored*

Before they can actively monitor files for changes, responsible personnel need a reference point with which they can compare modifications. The system will, therefore, document a baseline, or a known good state for files that will be monitored. This is achieved by calculating a hash value for each of the files being monitored. This standard will take into account the version, creation date, modification date, and other data that can help to provide assurance that the file is legitimate.

1. *Monitoring changes*

With a detailed baseline, we can proceed to monitor all designated files for changes. A regular interval after which a system scan will be carried out will be set.

1. *Sending an alert*

If the FIM system detects an unauthorized change, an alert will be sent out to the relevant personnel who can fix the issue.

1. *Reporting results*

For analytics purposes, the system will log all activities and allow for a report to be generated.

## Scope of the Project

Within the vast domain of cyber security and the CIA triad that deals with Confidentiality, Integrity and Availability this project mainly focuses on and covers the concept of Integrity and involves a bit of Authentication in order to authorise the appropriate parties to perform the integrity monitoring process. It should be taken into consideration that the project is built after the following factors:

1. *Host Machine Environments*

This project is limited to monitoring files based on the host machine on which the tool will be installed. It does not focus on cloud-based files or server files that are hosted elsewhere. Administration and monitoring is limited only to the host machine or environment in which the software tool is running.

1. *Files and file types*

Since, when a file is changed or updated, its hash value changes, the project focuses on files that are considered as archive files, whose information need not be changed. If archived file’s hash values are changed, then it means the files have been tampered with thus meaning the files have been compromised. For dynamic files, considerations would need to be made as to whether the change was initiated from within the network or from outside, and whether the user that performed the change has the appropriate permissions and privileges to access the files or even change their contents.

## Deﬁnition of Key Variables

1. *Files*

A ‘file’ in this project refers to any document or database record contained and stored in the host machine. These files are organisational documents that are of utmost importance to them and can be of, but not limited to, the following file extensions: ***.pdf, .docx, .xlsx, .sqlite3, .py****.*

1. *Environment*

A web-based platform on which the system will run. We will create a virtual environment using Python in which Flask, a Python Web Framework, and other dependencies and libraries, will be installed.

1. *FIM*

FIM is an abbreviation that stands for File Integrity Monitoring. This refers to the whole process of tracking the integrity of files contained in a specified directory in the host machine by constantly comparing the current hash value against a baseline value already saved in the database.

1. *Baseline Value*

The baseline value is the unique hash value that is assigned to each file within the monitored directory. This value is of great importance because it is the one which will be compared against the current or dynamic hash value of the file. If the file’s contents are changed, the hash value changes and will therefore be different from the saved baseline value meaning the file will have been compromised.

1. *Compromised File*

A file will be described as compromised when it would have been illegally accessed and had its contents altered. For the purposes of this project and in order to prevent false positives, we will be focusing on archived records whose baseline value is set to be constant because there would be no need to alter their contents.

## Conclusion

This chapter sought to summarise the research problem and describe the proposed solution, its functionality and how it is intended to solve the existing problem. Having analysed the current technologies in the FIM industry and the gap that has been identified, it can be concluded that a more viable, robust and convenient solution is needed, hence this FIM tool.

**REFERENCES**

1. <https://dzone.com/articles/what-is-file-integrity-monitoring-and-how-it-works>
2. <https://en.wikipedia.org/wiki/File_integrity_monitoring#:~:text=File%20integrity%20monitoring%20(FIM)%20is,and%20a%20known%2C%20good%20baseline.&text=Other%20file%20attributes%20can%20also%20be%20used%20to%20monitor%20integrity>
3. <https://www.cimcor.com/blog/how-does-file-integrity-monitoring-work>
4. <https://www.alliedmarketresearch.com/file-integrity-monitoring-market>