.

**NOVANECTAR SERVICES PVT. LTD**

**Integrity Checker Project Report**

**Project Title:** Integrity Checker

**Name: Alen Maria Nirmal P R**

**ID - NN/22/2408**

**Introduction**

The Integrity Checker is a cybersecurity tool designed to ensure the security and integrity of operating system files. In today's digital landscape, where cyber threats are constantly evolving, it is essential to maintain the authenticity of system files. Malicious actors often target system files to implant malware, execute unauthorized commands, or steal sensitive information. This project aims to provide an automated mechanism to detect unauthorized modifications, ensuring the trustworthiness of system files and reducing security vulnerabilities.

**Objective**

* The primary goal of the Integrity Checker is to regularly monitor and verify system files against a known database of hashes or checksums. Any discrepancies or unauthorized modifications will be flagged immediately, alerting administrators or users to take necessary action. This real-time verification helps maintain a secure and reliable operating system. Additionally, the tool ensures compliance with security policies by preventing unauthorized tampering with critical files.
* **Baseline Hash Generation:** The tool will first generate a baseline by computing and storing cryptographic hashes (SHA-512) of system files. This baseline acts as a reference for future integrity checks, ensuring that all monitored files are in their original state.
* **Regular Scanning:** The system will periodically scan the monitored files and compare their current hashes against the stored baseline. Any mismatches indicate potential tampering or corruption of files. This helps detect unauthorized changes and maintain system integrity.
* **Unauthorized Modifications Detection:** If a file's hash does not match the baseline, it will be flagged as modified. Additionally, content differences for text files will be identified using a unified diff comparison, allowing users to analyze changes at a granular level.
* **New and Deleted File Detection:** The tool will also identify any newly added or deleted files within the monitored directory. This is crucial in detecting unauthorized file insertions (potential malware) or deletions (possible security breaches).
* **Real-Time Alerts and Notifications:** Upon detecting any unauthorized changes, users will receive real-time alerts, enabling them to respond swiftly. The notifications may be displayed on the console or logged for further analysis.
* **User Interaction and Control:** Users can start monitoring, generate baselines, or stop monitoring through an interactive console. This provides flexibility and allows administrators to configure settings as needed.
* **Automated Task Scheduling:** The tool will operate on a scheduled basis without requiring manual intervention, ensuring continuous security monitoring.

**Implementation Details**

The Integrity Checker is implemented in Python, utilizing key libraries such as:

* hashlib for cryptographic hashing (SHA-512) to ensure robust integrity verification.
* os for interacting with the file system to traverse directories and read files.
* json for storing and retrieving file hash data, ensuring efficient management of integrity records.
* difflib for performing text-based comparisons, enabling users to track specific changes in modified files.
* sched and threading for scheduling periodic scans without user intervention.

The tool operates efficiently in the background, consuming minimal system resources while providing continuous integrity monitoring

**Security Considerations and Enhancements**

To ensure the effectiveness of the Integrity Checker, additional security measures can be implemented:

1. **Encryption of Hash Database:** The stored hash values can be encrypted to prevent attackers from manipulating them to bypass detection.
2. **Access Control Mechanisms:** Restricting access to the integrity checker’s configuration files ensures that only authorized users can modify its settings.
3. **Logging and Audit Trail:** Maintaining detailed logs of all detected changes, including timestamps and file details, helps track security incidents and facilitates forensic analysis.
4. **Integration with Intrusion Detection Systems (IDS):** The tool can be enhanced to work alongside existing security frameworks to strengthen overall system protection.

**Project Execution Plan**

* **Phase 1 - Research & Planning:** Understanding security risks, defining tool requirements, and selecting the best cryptographic methods.
* **Phase 2 - Development:** Writing Python scripts for hash generation, scanning, real-time monitoring, and user interaction.
* **Phase 3 - Testing & Optimization:** Ensuring the tool functions accurately and efficiently under various scenarios, including controlled file modifications and external threats.
* **Phase 4 - Deployment & Documentation:** Preparing user guides, ensuring accessibility via a Google Drive link, and implementing final security measures.

**Potential Future Improvements**

* **GUI-Based Interface:** Enhancing the tool with a graphical user interface for easier usability and monitoring.
* **Cloud Storage Monitoring:** Extending the tool to verify the integrity of files stored on cloud platforms like Google Drive or OneDrive.
* **Multi-Platform Support:** Expanding compatibility to Linux and macOS systems for broader applicability.
* **Machine Learning Integration:** Implementing anomaly detection techniques to predict potential threats based on historical data.

**Conclusion**

By implementing this Integrity Checker, we aim to provide an additional layer of security for operating systems. This proactive approach helps prevent cyber threats, ensuring system files remain untampered and reliable. With features like real-time monitoring, detailed logging, and automated detection, this tool significantly enhances system security. The project will be completed within the given deadline, and all necessary documentation and reports will be shared through a Google Drive link for accessibility by professionals evaluating the project.