JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Holy-Eye security scanner

Atharva Kothawade[1] **Sahil Sakpal**[2] **Pritam Jain**[3] **Dr. Nilakshi Jain**[4] **Deepali Shende**[5]
Engineering Student[1] Engineering Student[2] Engineering Student[3] Head Of Department[4] Co-Guide[5]
Department of Cyber Security[1,2,3,4,5]
Shah & Anchor Kutchhi Eng. College, Mumbai, India[1,2,3,4,5]

Abstract: The rapid rise in cybercrime has posed significant challenges for organizations worldwide, particularly with data integrity attacks becoming increasingly common. Cybercriminals' growing sophistication in using malware and other techniques makes it difficult to maintain secure systems, despite high integrity standards. Studies have revealed that social engineering and espionage techniques account for 60-80% of cyber-attacks, emphasizing the need to raise user awareness about potential threats. To address this, we developed Holy-Eye, a comprehensive security scanner that enables users to scan their systems, identify vulnerabilities, malware infections, and other security threats, empowering organizations to enhance their protection. By utilizing Holy-Eye, users gain a deeper understanding of their system security and can take appropriate action, reducing the risk and impact of cyber-attacks. In conclusion, Holy-Eye is an essential tool for organizations aiming to uphold system integrity and security amid the escalating cyber threat landscape.

Index Terms **- Cyber Security, Network Security, Scanner, Security Scanner, Nmap, Servers, Server Monitoring.**

I. INTRODUCTION

About

The increasing prevalence of data integrity attacks has created significant challenges for organizations globally. Many organizations lack proper monitoring of their systems and data files, leaving vulnerabilities that cybercriminals exploit to tamper with sensitive data, compromising its confidentiality. Human errors can also lead to file tampering, highlighting the need for robust security systems. To address these issues, we developed Holy-Eye, a web-based network scanner that identifies vulnerabilities, open ports, and services running on systems, enabling users to take appropriate action to enhance security. This report provides an overview of Holy-Eye's features, benefits, and effectiveness in improving an organization's security posture, including implementation guidelines and best practices.

Motivation behind the project

The motivation behind this project study is to develop Holy-Eye, a user-friendly security scanner that can help organizations and individuals protect their systems against cyber-attacks. Holy-Eye provides a comprehensive scan of the system, identifies potential vulnerabilities, and provides recommendations for improving security. By contributing to the development of more effective cybersecurity tools and strategies, the project aims to address the growing threat of cyber-attacks and create a safer digital environment for all users.

II. METHODOLOGY

Block Diagram

Fig. Block Diagram

In this diagram, the LAN Scanner is responsible for scanning the network and detecting the status of each device. The Network Monitoring Agent is responsible for continuously monitoring the devices and reporting any changes in their status. The Alerting System is responsible for receiving alerts from the Network Monitoring Agent and notifying the appropriate parties.

Entity Relationship Diagram

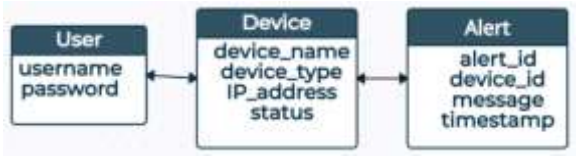

Fig. ER Diagram

In this ER diagram, we have three entities: User, Device, and Alert. The User entity stores information about the user, such as their username and password. The Device entity stores information about each device on the LAN, including the device name, type, IP address, and status. The Alert entity stores information about each alert that is triggered by the tool, including an alert ID, the device ID that triggered the alert, a message describing the alert, and a timestamp indicating when the alert was triggered.

Data Flow Diagram

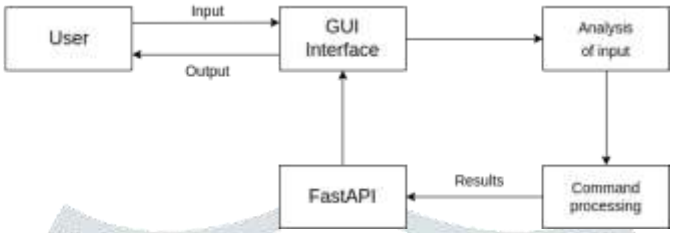

Fig.: Data Flow Diagram

Activity Diagram

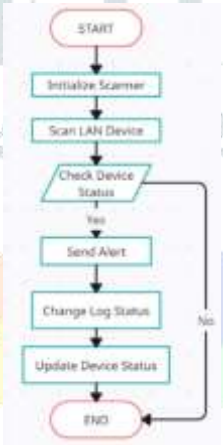

Fig.: Activity Diagram

This diagram shows the main steps of your tool's functionality. The tool starts by initializing the scanner, then scans the LAN devices and checks their status. If a status change is detected, the tool sends an alert, logs the status change, and updates the device status. Otherwise, it does nothing and the activity ends.

Methodology/Algorithm

1. Start the LAN scanner tool.
2. Set up the scanner with the necessary network parameters, such as the IP address range to scan, the scanning method, and the frequency of scans.
3. Begin scanning the LAN network for devices using the selected method.
4. For each device found, check its status and record it in the scanner database.
5. If the device status changes, send an alert to the user.
6. Continue scanning periodically and update the device status in the scanner database.
7. Provide the user with an option to view the status of all devices on the network.
8. End the LAN scanner tool.

Software/Hatrdware Requirements

a) Python
b) Bash
c) Html
d) CSS
e) FastAPI
f) Javascript

Use Case Diagram

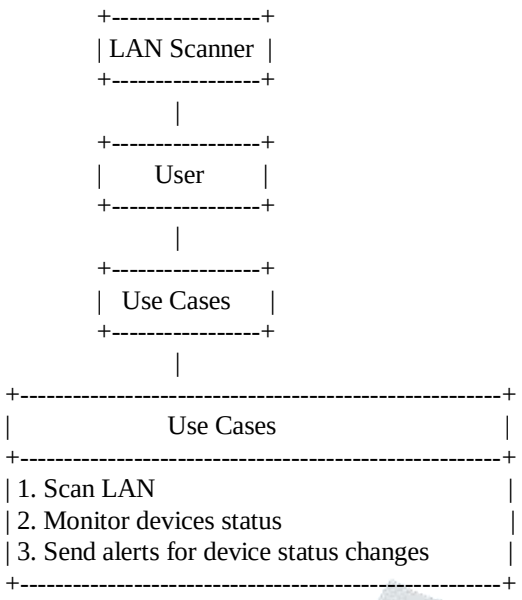

Fig. 4.2.1 Use Case Diagram

User Interface Design

a) Homepage

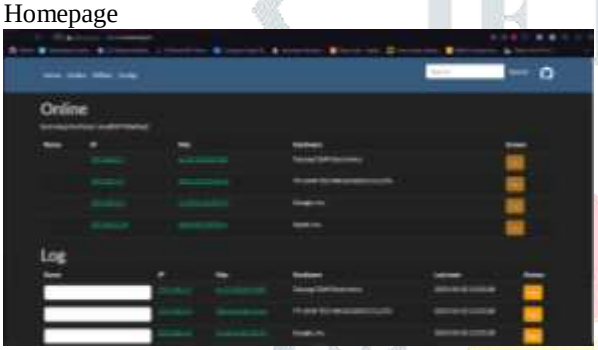

Fig: Homepage

b) Execution

Fig: Execution – Online IPs

Fig: Execution – Offline Ips

Fig: Configuration Tab

Result

In conclusion, the Holy Eye security scanner efficiently scans systems for vulnerabilities, providing detailed security information to address the increasing concern of cyber-attacks. Its user-friendly interface and cross-platform compatibility make it suitable for system administrators, network maintenance providers, and general users interested in computer security. The successful implementation of Holy Eye has proven its effectiveness in identifying vulnerabilities and enhancing system security. Future plans involve further development, including integration of artificial intelligence and machine learning algorithms, to improve accuracy and keep pace with evolving technology. Holy Eye aims to contribute to the advancement of data security and protection against cyber-attacks.

III. CONCLUSION

In conclusion, the Holy Eye security scanner is a powerful tool developed to address the rising threat of cyber-attacks. It efficiently scans systems, identifies vulnerabilities, and provides detailed security information. Holy Eye's user-friendly interface and cross-platform compatibility make it suitable for various users, from system administrators to general users concerned about computer security. Its successful implementation has proven its effectiveness in enhancing system security. Future plans involve further development and integration of artificial intelligence and machine learning algorithms to improve accuracy. Overall, Holy Eye contributes to advancing data security and protecting against cyber-attacks.

IV. ACKNOWLEDGEMENT

I take this opportunity to acknowledge everyone who have helped us in every stage of this project. Firstly, I am indebtedly grateful to our Guide and HoD of Cyber Security Department Dr. Nilakshi Jain and Co-Guide Ms. Deepali Shende for their support. Without their support this project would not have been completed.

Secondly, I would like to thank my group member Sahil Sakpal and Pritam Jain for their contribution to the project. Also, I would like to thank all the faculty members of our school/college for their kindness and support.

Lastly, I should really thank my friends and family who were always there to support me whenever needed.

REFERENCES

[1] F. Li, F. Xiong, C. Li, L. Yin, G. Shi and B. Tian, "SRAM: A State-Aware Risk Assessment Model for Intrusion Response," 2017 IEEE Second International Conference on Data Science in Cyberspace (DSC), 2017, pp. 232-237, doi: 10.1109/DSC.2017.9.

[2] R. R. Jueneman, "Integrity controls for military and commercial applications," [Proceedings 1988] Fourth Aerospace Computer Security Applications, 1988, pp. 298-322, doi: 10.1109/ACSAC.1988.113351.

[3] C. Zeidler and M. R. Asghar, "CloudEFS: Efficient and secure file system for cloud storage," 2016 14th Annual Conference on Privacy, Security and Trust (PST), 2016, pp. 239-246, doi: 10.1109/PST.2016.7906969.

[4] Nguyen Anh Quynh and Y. Takefuji, "A Real-time Integrity Monitor for Xen Virtual Machine," International conference on Networking and Services (ICNS'06), 2006, pp. 90-90, doi: 10.1109/ICNS.2006.13.

[5] I. F. A. Shaikhli, A. M. Zeki, R. H. Makarim and A. -S. K. Pathan, "Protection of Integrity and Ownership of PDF Documents Using Invisible Signature," 2012 UKSim 14th International Conference on Computer Modelling and Simulation, 2012, pp. 533-537, doi: 10.1109/UKSim.2012.81.

[6] S. Deepika and P. Pandiaraja, "Ensuring CIA triad for user data using collaborative filtering mechanism," 2013 International Conference on Information Communication and Embedded Systems (ICICES), 2013, pp. 925-928, doi: 10.1109/ICICES.2013.6508262.

[7] Y. Wang, B. Zhang, W. Lin and T. Zhang, "Smart grid information security - a research on standards," 2011 International Conference on Advanced Power System Automation and Protection, 2011, pp. 1188-1194, doi: 10.1109/APAP.2011.6180558.

[8] I. V. Mashkina, M. B. Guzairov, V. I. Vasilyev, L. R. Tuliganova and A. S. Konovalov, "Issues of information security control in virtualization segment of company information system," 2016 XIX IEEE International Conference on Soft Computing and Measurements (SCM), 2016, pp. 161-163, doi: 10.1109/SCM.2016.7519715.

[9] M. Tvrdikova, "Information system integrated security," 2008 7th Computer Information Systems and Industrial Management Applications, 2008, pp. 153-154, doi: 10.1109/CISIM.2008.41.

[10] J. Kim, I. Kim and Y. I. Eom, "NOPFIT: File System Integrity Tool for Virtual Machine Using Multi-byte NOP Injection," 2010 International Conference on Computational Science and Its Applications, 2010, pp. 335-338, doi: 10.1109/ICCSA.2010.79.

[11] Z. Wang, T. Huang and S. Wen, "A File Integrity Monitoring System Based on Virtual Machine," 2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control, 2012, pp. 653-655, doi: 10.1109/IMCCC.2012.396.

[12] J. Kaczmarek and M. Wrobel, "Modern approaches to file system integrity checking," 2008 1st International Conference on Information Technology, 2008, pp. 1-4, doi: 10.1109/INFTECH.2008.4621669.

[13] F. Tomonori and O. Masanori, "Protecting the integrity of an entire file system," First IEEE International Workshop on Information Assurance, 2003. IWIAS 2003. Proceedings., 2003, pp. 95-105, doi: 10.1109/IWIAS.2003.1192462.

[14] G. Daci and M. Shyle, "Improving data integrity and performance of cryptographic structured log file systems," 2011 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2011, pp. 1-5, doi: 10.15208/ati.2011.8.

[15] A. Pinheiro, E. D. Canedo, R. T. De Sousa and R. De Oliveira Albuquerque, "Monitoring File Integrity Using Blockchain and Smart Contracts," in IEEE Access, vol. 8, pp. 198548-198579, 2020, doi: 10.1109/ACCESS.2020.3035271.

[16] B. Wilbert and L. Chen, "Comparison of File Integrity Monitoring (FIM) techniques for small business networks," Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT), 2014, pp. 1-7, doi: 10.1109/ICCCNT.2014.6963090.

[17] B. Shi, B. Li, L. Cui and L. Ouyang, "Vanguard: A Cache-Level Sensitive File Integrity Monitoring System in Virtual Machine Environment," in IEEE Access, vol. 6, pp. 38567-38577, 2018, doi: 10.1109/ACCESS.2018.2851192.

[18] Udzir, Nur & Samsudin, Khairulmizam. (2011). "Towards a Dynamic File Integrity Monitor through a Security Classification". International Journal of New Computer Architectures and their Applications (IJNCAA). 3. 789-802.

[19] NARAYAN KULKARNI, Neha; KUMAR A. JAIN, Shital; Survey on Data Integrity, Recovery, and Proof of Retrievability Techniques in Cloud Storage. International Journal of Engineering & Technology, [S.l.], v. 7, n. 3.6, p. 55-58, july 2018. ISSN 2227-524X.

[20] Anusha Priya.G, Mrs. Esther Daniel: A Literature Survey on Integrity Verification Techniques, International Journal of Engineering Research & Technology (IJERT)ISSN: 2278-0181 Vol. 4 Issue 05, May-2015.

[21] A protocol Article " A 'nightmare scenario': Data-tampering attacks are hard to detect, with devastating consequences.