Hierarchical Expert System for Security Evaluation on Windows Mobile Devices and Laptops

Akshat Mehta
Department of Computer Science
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY, 14586
Email: am1717@rit.edu

Nishith Hingoo
Department of Computer Science
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY, 14586
Email: nh8145@rit.edu

Siddhartha Jejurkar
Department of Computer Science
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY, 14586
Email: sj6670@rit.edu

***Abstract*—This paper presents a hierarchical expert system for evaluating the security of Windows mobile devices and laptops. The system utilizes multiple metrics from device integrity, network security, user authentication, and data protection. We integrate machine learning techniques and rule-based expert systems to provide comprehensive security assessments. This paper compares different security evaluation methodologies, analyzing their strengths and applicability to Windows platforms.**

***Index Terms*—Security Evaluation, Hierarchical Expert System, Windows Devices, Machine Learning, Rule-Based Systems**

I. Introduction

Security evaluation of mobile devices and laptops is crucial in the digital age. As these devices become more integrated into daily business and personal activities, their security risks increase, making comprehensive evaluation methods essential. The increasing sophistication of cyber threats necessitates a robust framework to assess and mitigate potential vulnerabilities effectively.

The research focuses on developing a hierarchical expert system for evaluating security on portable windows devices. The system aims to automate data collection, assess security, and provide actionable recommendations. By leveraging multiple metrics and machine learning techniques, the system provides a comprehensive security evaluation.

The main focus of this research is on expert system development for security evaluation on windows laptops. Since windows mobile devices were discontinued in 2017, only a brief overview of existing security evaluation techniques for mobile OS platform is presented.

II. Computer Security Evaluation of the Specified Implementation Device

A. Reported Approaches

- **Anti-malware Solutions**: Tools like Windows Defender and third-party antivirus software are crucial for protecting against malware. These solutions regularly update their virus definitions to recognize and remove new threats effectively [1].
- **System Integrity**: Regular integrity checks of the operating system and critical system files help in the early detection of unauthorized changes. These checks can identify altered system files, unauthorized patches, and unexpected changes to configuration settings [4].
- **Network Security**: Firewalls and intrusion detection systems (IDS) are employed to monitor and protect network traffic [2]. Firewalls block unauthorized access while IDS detect suspicious activities that may indicate an ongoing attack.

B. Methods Employed

- **Vulnerability Scanning**: Tools such as Nessus and OpenVAS scan for known vulnerabilities in the system. These tools check for unpatched software, misconfigurations, and other common vulnerabilities that attackers exploit [6].
- **Behavioral Analysis**: Monitoring system behavior to detect anomalies indicative of security breaches. This involves tracking unusual activities such as unexpected spikes in network traffic, unauthorized access attempts, and irregular system performance metrics [5].
- **Penetration Testing**: Simulated attacks to test the effectiveness of security defenses [3]. Penetration testers attempt to exploit vulnerabilities in a controlled manner, providing insights into potential security weaknesses and the effectiveness of current defenses.

III. Design of the Hierarchical Metrics System

Security assessment and risk evaluation of a device requires the collection and processing of various metrics native to the target platform. For windows based laptops, we propose

a hierarchical rule based expert system with fuzzy logic to perform security and risk evaluation locally.

Use of an expert system allows flexibility, such that new rules and domains can be integrated into the model with ease, while the Hierarchical structure promotes modularity through which changes to any existing module do not affect other modules and the whole system does not require updating. Use of Fuzzy logic improves the system's ability to handle uncertain data allowing the system to provide linguistic feedback about a security parameter.

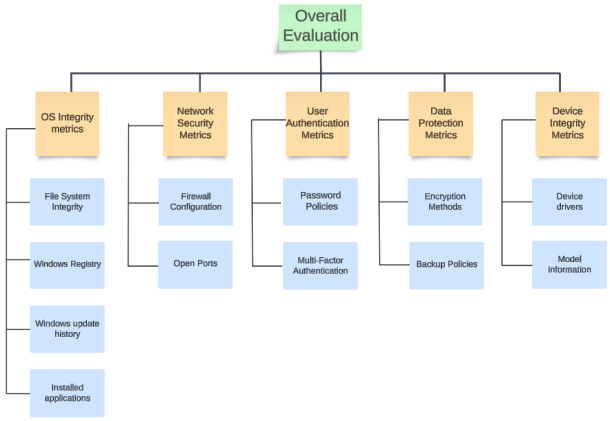

Fig. 1. Hierarchical Metrics System for Security Evaluation

A. Metrics selection

The overall expert system is designed using a hierarchical metrics structure consisting of 3 levels. The topmost level provides the overall security assessment and score. The middle layer provides domain specific metrics derived by integrating related metrics from the lowest level . For the lowest level, metrics are calculated using the device parameters listed below:

- **Windows Registry**
- **Windows update History**
- **File System Integrity**
- **Installed applications**
- **Firewall configuration**
- **Encryption Methods**
- **Backup Policies**
- **Password policies**
- **Multi Factor Authentication**
- **Device drivers**
- **Model information**

B. Metrics Composition and Content

- **OS Integrity metrics**:
  - **File System Integrity**: Monitors critical system files for unauthorized changes [4]. This includes check sums and cryptographic hashes to ensure files have not been tampered with. This metric also relies on analyzing file permissions for critical system directories such as System32 and Program Files(x86).
  - **Windows Registry**: The windows registry is an essential component of windows operating systems that acts as a central database storing critical information about the kernel and other low level services. Evaluation of this metric involves analyzing event logs, timestamps and key hashes. [8]
  - **Windows update history**: Microsoft rolls out OS updates and security patches for all windows devices on a regular basis. This metric yields a higher score if the device has the latest updates, security patches and kernel versions installed.
  - **Installed applications**: This metric checks all installed applications on the device and analyzes various parameters such as permissions, memory consumption patterns, source verification and if the application was flagged for suspicious activity, to evaluate the final security score.
- **Network Security Metrics**:
  - **Firewall Configuration**: Evaluates the configuration and rules of the firewall. Ensures that only necessary ports are open and that traffic rules are correctly implemented [1].
  - **Open Ports**: Identifies and evaluates open ports for potential vulnerabilities [5]. Open ports can be entry points for attackers, so it's essential to monitor and manage them.
- **User Authentication Metrics**:
  - **Password Policies**: Strength and enforcement of password policies. This includes minimum password length, complexity requirements, and regular password changes.
  - **Multi-Factor Authentication**: Implementation and usage of MFA for enhanced security [6]. Adds an additional layer of security through multiple verification instances.
- **Data Protection Metrics**:
  - **Encryption Methods**: Reviews the encryption techniques for data at rest and in transit. Ensures that sensitive data is encrypted using robust algorithms to protect against interception and unauthorized access. For this metric, Bitlocker encryption on Windows laptops is utilized for evaluating a security score.
  - **Backup Policies**: Assesses the frequency and security of data backups [7]. Security evaluation for this metric relies on event logs obtained from One drive, which is a native cloud storage service for offsite backups on windows devices.
- **Device Integrity Metrics**:
  - **Device drivers**: Reviews the encryption techniques for data at rest and in transit. Ensures that sensitive data is encrypted using robust algorithms to protect against interception and unauthorized access.

  - **Model information**: Apart from on device assessments, several reported vulnerabilities are also be used for security evaluation. A popular example is the bypassing of windows hello authentication on several laptops from giant manufacturers like Dell, Lenovo and Microsoft. [9]

C. System Organization

- **Metric Collection**: Data collection through system APIs and user inputs. This involves gathering data from various system components and user interactions to ensure comprehensive coverage.
- **Data Aggregation**: Combining metrics into a hierarchical structure. Aggregating data helps in understanding the overall security posture by summarizing individual metrics.
- **Scoring System**: Developing a scoring system for each metric and an overall security score [2]. The scoring system helps in quantifying security risks and prioritizing mitigation efforts.

IV. Novel Approaches to Security Evaluation

In our quest to develop a hierarchical expert system tailored for security evaluation on Windows laptops, we have listed several reported approaches to security evaluation and also presented several metrics used in the ES design. Novel metrics such as Device driver analysis and latop model information have been integrated into the ES design. Below we present some additional novel metrics to security evaluation which could be integrated in future iterations of the expert system:

- **Read / Write permissions**: Analysis of an application's read and write permissions to evaluate if it could possibly damage critical system infrastructure and services
- **Mic, camera and screen capture permissions**: Applications accessing the mic, camera or having screen capture permissions without any correlation to their specified tasks pose a huge security risk.
- **Start on boot**: Applications which are not critical system programs or low level kernel services that have permission to startup on boot can be flagged for suspicious activity.
- **BIOS and firmware ROMs**: Unauthorized rootkits and BIOS updates pose an extremely high risk to the firmware on the laptop's hardware.
- **Bootloader**: Unauthorized bootloaders and having multiple operating systems on the same device is also a potential security risk
- **Kernel integrity**: System kernel that have not been updated in a while and do not have the latest security patches installed are at a high risk of falling victim to external attacks.

While the proposed ES covers a large set of metrics, it is not flexible as the pre-defined rules are static and they do not learn from different device variations. below we propose a few variations of the exisitng ES design to improve on its existing limitations:

1) Hybrid Security Evaluation Engine: **Integration of Rule-Based Expert Systems with Machine Learning:** Fusing the deterministic prowess of rule-based expert systems with the foresight of machine learning models enables predefined security checks while facilitating adaptive anomaly detection.

2) Dynamic Rule Adjustment: **Automatic Refinement of Security Rules:** Allowing for automatic refinement of security rules by incorporating dynamic rule adjustment capabilities allows the system to improve evaluation results through updating rules based on ongoing assessments and new data, enhancing accuracy and minimizing false positives. Experta already facilitates effortless updates and modifications to the rule set based on fresh insights and data, bolstering the system's adaptability and precision.

3) Context-Aware Security Evaluation: **Incorporation of Operational Environment Data:** Assessing the operational environment of the device, such as network conditions and user behavior patterns, during security evaluations can help the system deliver a nuanced and precise security assessment.

4) Adaptive Learning Mechanism: **Continuous Improvement through Learning:** An adaptive learning mechanism can enable the system to continuously learn from new security incidents and updates. Employing PyTorch to develop deep learning models capable of discerning complex patterns and emergent threats can help achieve this task.

5) User Behavior Analysis: **Monitoring and Analyzing User Activities:** A module for user behavior analysis, scrutinizing and analyzing user activities to identify unusual or potentially malicious behavior will provide a better security assessment. Machine learning models trained on behavioral data can be used to detect deviations from normal user patterns.

6) Real-Time Threat Detection and Response: **Immediate Alerts and Automated Mitigation Actions:** Designing real-time threat detection and response, the system can deliver immediate alerts and execute automated mitigation actions upon identifying security incidents, to ensure swift and effective threat management.

V. Analysis of available technologies

A. Available Methods and Technologies

- **Expert Systems**:
  - **CLIPS/Py-CLIPS**: Rule-based systems for security evaluation. CLIPS (C Language Integrated Production System) is a tool for building expert systems, providing a way to represent knowledge in the form of rules.
  - **Experta**: Python library for building rule-based systems. Experta facilitates the development of expert systems using Python, making it easier to integrate with other tools and technologies.
  - **Prolog**: Prolog is a declarative programming language used for developing Expert systems. It's popularity is due to its easy to learn syntax and shallow learning curve.

  - **JESS**: JESS is a rule based engine used to developed expert systems in java
  - **Blaze Advisor**: Blaze advisor is a large and extremely powerful rule engine developed for large scale systems.
- **Machine Learning Techniques**:
  - **Supervised Learning**: Models trained on labeled data to predict security risks. Supervised learning algorithms use historical data to learn patterns associated with security incidents, allowing them to predict future risks.
  - **Unsupervised Learning**: Identifying patterns and anomalies in system behavior [3]. Unsupervised learning algorithms can detect unusual activities that do not match known patterns, indicating potential security breaches.
- **TensorFlow/PyTorch**: For building and training ML models. These frameworks provide powerful tools for developing and deploying machine learning models [7].
- **Weka/Matlab**: For data analysis and algorithm development [4]. Weka and Matlab offer extensive libraries for data processing, analysis, and machine learning.

B. Available languages

- **Python**: For scripting, data analysis, and ML model development. Python is widely used in data science and machine learning for its simplicity and extensive library support [1].
- **Java**: For developing robust, cross-platform applications [5]. Java's portability and performance make it suitable for building enterprise-grade applications.
- **C**: For developing expert systems C is often useful in conjunction with CLIPS, but the language itself is not portable and hence requires recompilation of source code when the system is used on a different device architecture.

VI. Evaluation System Design and Implementation on a Specified Device

Based on the finalized system design and requirements, the following technologies are chosen for the system development:

Python: Embraced as the primary programming language for its versatility, ease of use, and extensive library support. Python's robust ecosystem facilitates the seamless integration of the various components of the security evaluation system, making it an ideal choice for developing sophisticated applications.

Experta: Utilized for its dynamic rule management capabilities, Experta enables the definition and management of security rules and policies within the system. Its flexibility and seamless integration with Python allow for dynamic adjustments and modifications, enhancing the system's adaptability and accuracy.

PyTorch: Employed for developing and training machine learning models, PyTorch supports anomaly detection, predictive analytics, and user behavior analysis. Its capabilities in deep learning and real-time data processing are vital for addressing intricate security patterns and evolving threats, ensuring the system's efficacy in detecting and responding to new security challenges.

A. Task specification

1) **Data coolection:** Harvest data from various system components and user interactions using system APIs and python scripts.
2) **Aggregate Data for Analysis:** Integrate data into a hierarchical structure for thorough analysis, incorporating user inputs to provide additional context.
3) **Define Security Rules:** Using Experta, we delineate initial security rules and policies covering various domains such as access control, data protection, and network security.
4) **Apply Rule-Based Evaluation:** Utilize Experta to implement predefined security rules, evaluating collected data to identify potential security issues.
5) **Machine Learning Analysis:** Employ PyTorch models to analyze data and approximate the ES input/output characteristics.
6) **Train Machine Learning Models:** Develop and train machine learning models with PyTorch, leveraging historical security data generated using the ES to approximate the ES input/output surfaces.

B. Project Specification and implementation details

To simplify architecture and development, the entire system is broken into 3 successive modules:

- **Data acquisition for metrics** Data acquisition for the Expert System input requires using a combination of conventional and programmatic approaches. To simplify architecture, we propose building a single python library for data acquisition. This library will consist of several python scripts each authored for a specific metric of the expert system:
  - **OS Integrity**: Within the python script we use the winreg for exposing the windows registry API to python, os module to get file permissions, windows-tools package to retrieve windows update logs and subprocess module for getting list of all installed applications on the windows device
  - **Network Security**: The windows-tools package and os modules allow for retrieving the native firewall rules and open ports configuration.
  - **User Authentication**: Password policies are fetched using a combination registry logs and batch/shell scripts
  - **Data Protection**: Bitlocker data is fetched using the windows-tools package in python. Onedrive logs are usually stored as binary files in formats like odl and hence fetching the backup logs requires using a custom odl parser in python.
  - **Device Integrity**: Pywin32 module provides the API to access device driver and system data.

- **Metrics Integration** For implementing the expert system, we have decided to use experta along with custom python scripts to evaluate security scores for each metric. Metrics integration will be performed using the Expert system rules and scoring system as described below:
  - **ES rules**: For the lowest level, depending on the metric being considered, specific rules are employed for evaluating the final security score for that metric. Rule definitions for the metrics can provide the number of potential vulnerabilities, degree to which certain security requirements are fulfilled and flagging blacklisted services and applications. The overall aggregation of these rule outputs will aid in evaluating the final security score for the metric.
  - **Scoring**: Depending on the metric being considered, the score assignment will be done on based on a 0 to 10 rating system based on a linear scale. For example, if a firewall has more number of open ports it has a high probability of falling victim to external attacks, thus a lower security score will be assigned to this metric.

  Lower order metrics will produce individual security scores which will be then integrated into the middle layer metrics using the weighted means technique. Here the weight for lower order metrics will be assigned during the rule definition stage itself.

C. Machine Learning Techniques

To implement an approximation of the ES system we propose the creation of a ML model using PyTorch along with MATLAB as the design package. For training the ML model the implemented ES system will be used to generate different datasets using different windows device platforms and incorporating small deltas and variations in the input parameters for low order metrics.

VII. Conclusion

This paper has presented a hierarchical expert system designed to evaluate the security of Windows laptops. By integrating a hierarchical rule-based expert systems our approach provides a robust, adaptive, and comprehensive security evaluation framework.

A. Content of the Security Evaluation System

The evaluation system is composed of 3 levels of hierarchical metrics. At the lowest level the following metrics have been chosen:

- **Windows Registry**
- **Windows update History**
- **File System Integrity**
- **Installed applications**
- **Firewall configuration**
- **Encryption Methods**
- **Backup Policies**
- **Password policies**
- **Multi Factor Authentication**
- **Device drivers**
- **Model information**

The 2nd order metrics in the middle layer are formed by integrating the lowest level metrics. In the middle layer the following metrics are evaluated:

- **OS integrity**
- **Network security**
- **User authentication**
- **Data protection**
- **Device Integrity**

At the end the final security evaluation score is generated by integrating the 2nd order metrics form the middle layer.

Data acquisition for metrics to feed as input to the expert system is done by constructing a library composed of several python and shell scripts. Following data acquisition a metrics integration module is executed that integrates the lower order metrics using expert system rules and a predefined scoring system to produce a final security assessment and score.

Since ES models are computationally expensive on portable platforms such as Windows laptops, the development of a machine learning model is also proposed. The ML model will be constructed using PyTorch and MATLAB and trained using datasets generated using the implemented ES system. The goal of the ML model is to significantly reduce the computational requirements of the ES model by creating modular functions that approximate the ES system's input/output surfaces.

B. Tools for Design and Implementation:

For the proposed expert system and ML model, the following technologies have been chosen:

- **Python**: Base language for authoring data acquisition and security evaluation scripts, chosen specifically for its cross platform and fast prototyping capabilities.
- **Experta**: Python library for building the rule based expert system
- **PyTorch**: A python based machine learning library to generate and train the ML model approximation of the expert system.
- **MATLAB**: Design package to be used in conjunction with PyTorch for training the ML model

The proposed hierarchical expert system offers a robust framework for evaluating the security of Windows laptops. By integrating various metrics and leveraging rule-based expert systems the system provides a comprehensive assessment and actionable recommendations. This approach not only identifies current vulnerabilities but also predicts potential risks, enabling proactive security measures. Training machine learning models on the datasets generated using the expert system allows a significant reduction in computational resources on the target platform.

VIII. Comparative Analysis

A. Methods and Results Comparison

- **"Security Evaluation of the Windows Mobile Operating System"**: Focuses on specific security challenges

of Windows mobile OS, providing a detailed analysis of vulnerabilities and mitigation strategies [5]. This study highlights the unique security issues faced by mobile operating systems and offers targeted solutions.

- **"Feature Grouping-based Parallel Outlier Mining of Categorical Data Using Spark"**: Discusses advanced data mining techniques for detecting anomalies in high-dimensional data, which can be applied to security anomaly detection [2]. This method leverages the power of distributed computing to efficiently handle large datasets.
- **"A Hierarchical Security Event Correlation Model for Real-Time Threat Detection and Response"**: Proposes a model to reduce the number of IDS alerts by correlating events in real-time, enhancing the efficiency of threat detection [3]. This approach improves the accuracy of threat detection by combining multiple correlation techniques.
- **"Research on the Security Problem in Windows 7 Operating System"**: Analyzes security vulnerabilities in Windows 7 and proposes a graph model to describe and mitigate these vulnerabilities [4]. Although focused on an older OS version, the methodologies can be adapted for modern systems.
- **"A Security Evaluation Model Based on Fuzzy Hierarchy Analysis for Industrial Cyber-physical Control Systems"**: Provides a fuzzy hierarchy model for security evaluation, which can be adapted for evaluating Windows devices [6]. This model is effective in dealing with uncertainties and provides a comprehensive risk assessment.
- **"Security Issues and Challenges in Windows OS Level"**: Explores various security challenges in different versions of Windows OS and provides insights into the evolving nature of threats and the importance of continuous security evaluation [1].
- **"Survey of Intrusion Detection Systems: Techniques, Datasets, and Challenges"**: Reviews various intrusion detection systems, highlighting the techniques and challenges in detecting and mitigating security threats [7]. The survey provides a comprehensive overview of the current state of IDS technology and its applicability to Windows platforms.

B. Strengths and Weaknesses

- **Windows Mobile OS**: Strong focus on specific OS challenges but may lack general applicability to all Windows devices.
- **Feature Grouping**: Effective for anomaly detection but requires significant computational resources.
- **Hierarchical Event Correlation**: Reduces alert volume but relies on accurate event correlation for effectiveness.
- **Windows 7 Security**: Provides detailed vulnerability analysis but may be outdated for newer Windows versions.
- **Fuzzy Hierarchy Model**: Robust in handling uncertainties but may be complex to implement.
- **Security Challenges in Windows OS**: Comprehensive coverage of security issues but requires continuous updates to remain relevant.
- **Intrusion Detection Systems Survey**: Extensive review of IDS techniques but highlights the challenges in real-time threat detection.

REFERENCES

[1] "Security Issues and Challenges in Windows OS Level," *Journal of Information Systems I& Information Technology*, 2023.

[2] "Feature Grouping-based Parallel Outlier Mining of Categorical Data Using Spark," *Information Sciences*, 2019.

[3] H. Maosa, K. Ouazzane, M. C. Ghanem, "A Hierarchical Security Event Correlation Model for Real-Time Threat Detection and Response," *Network*, 2024.

[4] "Research on the Security Problem in Windows 7 Operating System," *IEEE Xplore*, 2019.

[5] "Security Evaluation of the Windows Mobile Operating System," *Chalmers University of Technology*, 2020.

[6] "A Security Evaluation Model Based on Fuzzy Hierarchy Analysis for Industrial Cyber-physical Control Systems," *IEEE Xplore*, 2021.

[7] "Survey of Intrusion Detection Systems: Techniques, Datasets, and Challenges," *Cybersecurity*, 2019.

[8] Medium.com. 2023. The Role of Windows Registry in Cybersecurity. https://medium.com/@sakthisrini23/the-role-of-windows-registry-in-cybersecurity-21d18eca848c . Accessed: May 30, 2024.

[9] Theverge.com. 2023. Microsoft's Windows Hello fingerprint authentication has been bypassed. https://www.theverge.com/2023/11/22/23972220/microsoft-windows-hello-fingerprint-authentication-bypass-security-vulnerability . Accessed: May 31, 2024.