BEHAVIORAL MODEL FOR LIVE DETECTION OF APPS BASED ATTACK

[1] YALLAMELLI SIVA KUMAR

[2] Mr. Naga Srinivasa Rao

[1]PG STUDENT ,DEPT OF MCA

[2] Asst. Prof, Dept of MCA

SREE KONASEEMA BHANOJI RAMARS P.G. COLLEGE (AMALAPURAM)

Abstract Smart phones with the platforms of applications are gaining extensive attention and popularity. The enormous use of different applications has paved the way to numerous security threats. The threats are in the form of attacks such as permission control attacks, phishing attacks, spyware attacks, botnets, malware attacks, privacy leakage attacks. Moreover, other vulnerabilities include invalid authorization of apps, compromise on the confidentiality of data, invalid access control. In this paper, an application-based attack modeling and attack detection is proposed. Due to A novel attack vulnerability is identified based on the app execution on the smartphone. The attack modeling involves an end-user vulnerable application to initiate an attack. The vulnerable application is installed at the background end on the smartphone with hidden visibility from the end-user. Thereby, accessing the confidential information. The detection model involves the proposed technique of an Application-based Behavioral Model Analysis (ABMA) scheme to address the attack model. The model incorporates application-based comparative parameter analysis to perform the process of intrusion detection. The ABMA is estimated by using the parameters of power, battery level, and the data usage. Based on the source internet accessibility, the analysis is performed using three different configurations as, Wi-Fi, mobile data, and the combination of the two. The simulation results verify and demonstrates the effectiveness of the proposed model.

IndexTerms **– smartphone ABMA, power, battery level**

I.INTRODUCTION

In recent years smart phone application models have explosively increased from personnel to professional applications including education, online shopping, net banking, and healthcare. The platform of these applications has massively increased the threat of attacks by compromising trustworthiness and security capabilities [1]-[3]. Third party application marketing is one of the major threat, wherein interested application can be installed by the end-user. However, the applications from these platforms can prove menaces with the advent of vulnerable breaches. Various attacks were identified that can prove detrimental and have adverse effects on the overall security of the information concerned to the smart phone. The jamming attack is one of the prime issues against time-critical applications. The attack exposes the in transit confidential information to the intruders [4]. Inaudible voice attack manipulates voice controllable device with unnoticeable characteristics while operating modulation technique using ultrasonic carriers [5]. The camera based attack proves a serious security threat to the multimedia applications of smart phones [6]. The side-channel attack exploits the leakage data to limit the data confidentiality on smart phones [7],[8]. Pin inference attack is identified as the privacy threat for the devices controlled by smart phones [9]. Indirect eavesdropping attack is another possible menace that makes use of acoustic sensing to execute the attack on the smart phone [10].

Permission control is one of the primary countermeasures against the possible security risks in smart phones. The permission control enhances the security by incorporating conditional restrictions on the particular executions performed by the applications. Various permission control methodologies were formulated including context sensitive permission control [11], user driven access control [12], permission control using crowd sourcing [13], and Sig PID (Significant Permission Identification) [14]. However, the major limitation associated with the permission control technique is that the targeted functionality of the application is restricted such that the desirable and undesirable private data transmission is not well differentiated.

LITERATURE SURVEY

Conducting a literature survey is crucial for understanding the current state-of-the-art, identifying gaps in existing research, and informing the development of a behavioral model for live detection of app-based attacks. Here's an overview of potential areas to explore in your literature review:

1.Behavioral Analysis Techniques

- Review studies that explore various behavioral analysis techniques for detecting malicious activities within applications. This may include approaches based on machine learning, anomaly detection, heuristics, and rule-based systems.

2. Real-Time Detection Methods

- Investigate research papers that focus on real-time detection methods specifically tailored for identifying app-based attacks as they occur. Examine the efficacy of different approaches in providing timely alerts and responses to security threats.

3. App-Based Attack Scenarios

- Explore literature discussing common app-based attack scenarios, such as malware injection, data exfiltration, privilege escalation, and code injection. Understand the techniques and tactics employed by attackers to compromise applications and exploit vulnerabilities.

4. Datasets and Case Studies

- Look for datasets and case studies used in previous research to train and evaluate behavioral models for app-based attack detection. Assess the diversity and realism of these datasets, as well as the methodologies used for performance evaluation.

5. Machine Learning for Security

- Examine studies that apply machine learning techniques to various security domains, including intrusion detection, malware analysis, and anomaly detection. Identify relevant algorithms, feature selection methods, and evaluation metrics used in these studies.

6. Anomaly Detection in App Behaviors

- Review research on anomaly detection algorithms tailored for analyzing app behaviors and identifying deviations from normal usage patterns. Evaluate the effectiveness of these algorithms in distinguishing between benign and malicious activities.

7. Intrusion Detection Systems (IDS)

- Investigate literature related to intrusion detection systems designed specifically for monitoring and protecting applications against cyber threats. Analyze the architecture, components, and detection capabilities of existing IDS solutions.

8. Security Challenges in Mobile and Desktop Environments

- Explore studies that discuss the unique security challenges posed by mobile and desktop environments, including issues related to app sandboxing, permissions, app store vetting processes, and third-party libraries.

9. Evaluation Metrics and Benchmarks

- Identify commonly used evaluation metrics and benchmarks for assessing the performance of behavioral models in detecting app-based attacks. Consider metrics such as detection accuracy, false positive rate, detection latency, and scalability.

10. Emerging Trends and Future Directions

Look for recent publications and academic conferences focusing on emerging trends and future directions in the field of app security and behavioral analysis. Identify areas for potential research and innovation, such as adversarial machine learning, explainable AI, and decentralized detection architectures.

III MODULES

Creating a behavioral model for live detection of apps-based attacks involves a comprehensive understanding of both the system specifications and the nature of potential attacks. Here's a breakdown of the subsystem specifications you might consider:

1. Data Collection Subsystem:

- Data Sources: Identify sources of data, such as system logs, network traffic, application logs, and user activity logs.

- Data Collection Mechanisms: Implement methods for collecting data in real-time or near real-time from various sources.

- Data Preprocessing: Cleanse, normalize, and preprocess the collected data to make it suitable for analysis.

2. Feature Extraction Subsystem:

-Behavioral Features: Determine relevant behavioral features that can indicate potentially malicious activities, such as abnormal process spawning, unusual network traffic, or unauthorized access attempts.

-Feature Extraction Techniques: Utilize techniques like statistical analysis, machine learning, or rule-based methods to extract meaningful features from the collected data.

3. Modeling Subsystem:

- Machine Learning Models: Choose appropriate machine learning algorithms such as anomaly detection algorithms (e.g., Isolation Forest, One-Class SVM) or supervised classifiers (e.g., Random Forest, Gradient Boosting) for modeling.

- Rule-based Systems: Incorporate rule-based systems to define explicit rules for detecting specific types of attacks based on known patterns or signatures.

- Ensemble Methods: Consider ensemble methods to combine the strengths of multiple models for improved detection accuracy.

4. Detection Subsystem:

- Real-time Analysis: Implement algorithms capable of analyzing data streams in real-time to promptly detect suspicious activities.

- Scalability: Ensure that the detection subsystem can scale efficiently to handle a large volume of data in high-traffic environments.

- Alerting Mechanisms: Develop mechanisms for generating alerts or notifications when potential attacks are detected, with varying levels of severity based on the perceived threat.

5. Feedback Loop Subsystem:

-Feedback Mechanisms: Establish mechanisms for incorporating feedback from detected incidents to continually improve the detection capabilities of the system.

- Model Adaptation: Implement techniques for dynamically adjusting detection thresholds or updating models based on the evolving threat landscape.

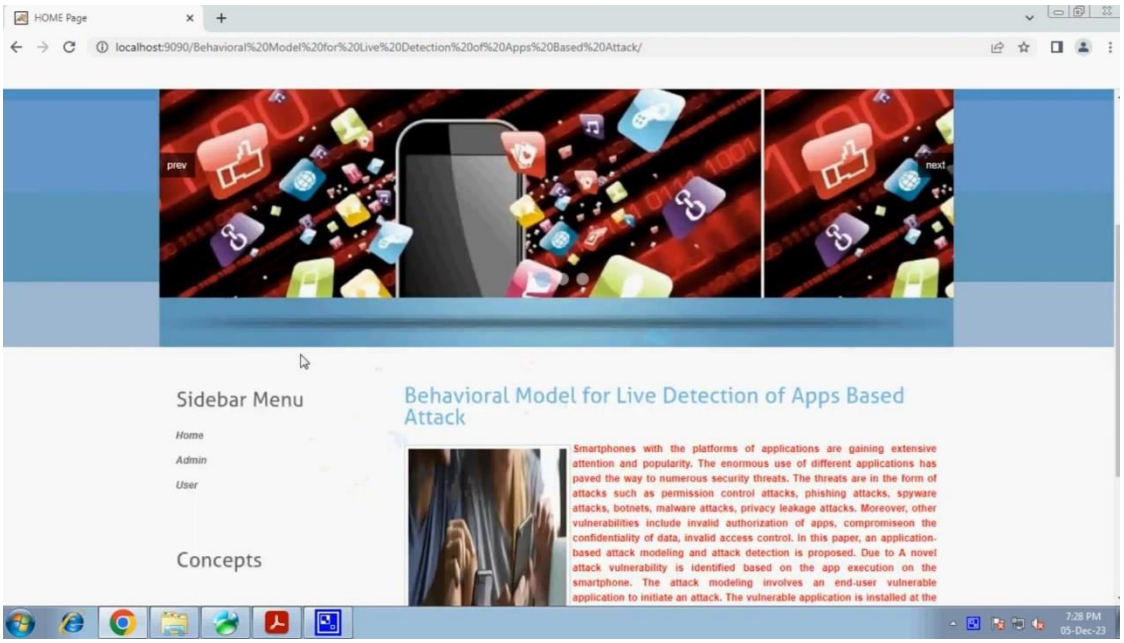
- Human Intervention: Enable human analysts to review detected incidents, provide feedback, and refine detection rules or models as necessary.

IV PROPOSED METHODOLOGY

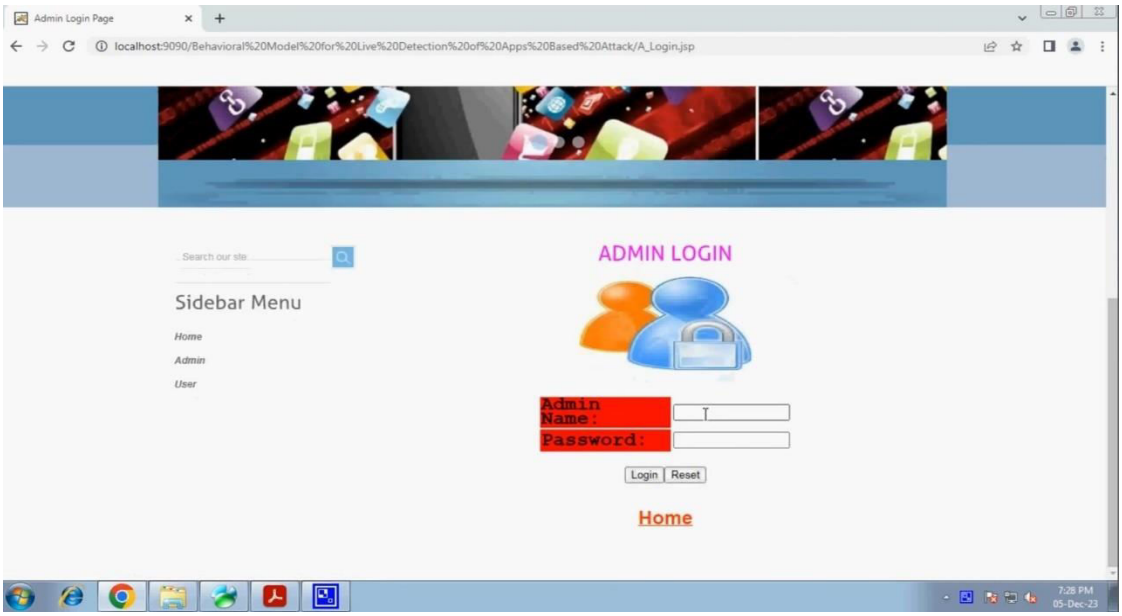
Application-based Behavioral Model Analysis (ABMA) is a novel methodology defined for security improvement of smartphone platforms. Conventional schemes of security enhancement mechanisms in smartphones are attack-specific or application-specific. A generalized security scheme independent of versions and type of application is yet to be addressed. Also, the reliability with upgradation and optimized statistical parameters require immense attention. The behavioral model for smartphone based applications is an innovative initiative to address these challenges with an effectual performance. The model is independent of the technology and the updates of the applications of the smartphone. It provides the live detection app based attack with adaptive capability. The detection model ensures the apps basedattack detection on authorization, confidentiality, and integrity with efficient and less complex methodology.

V RESULTS

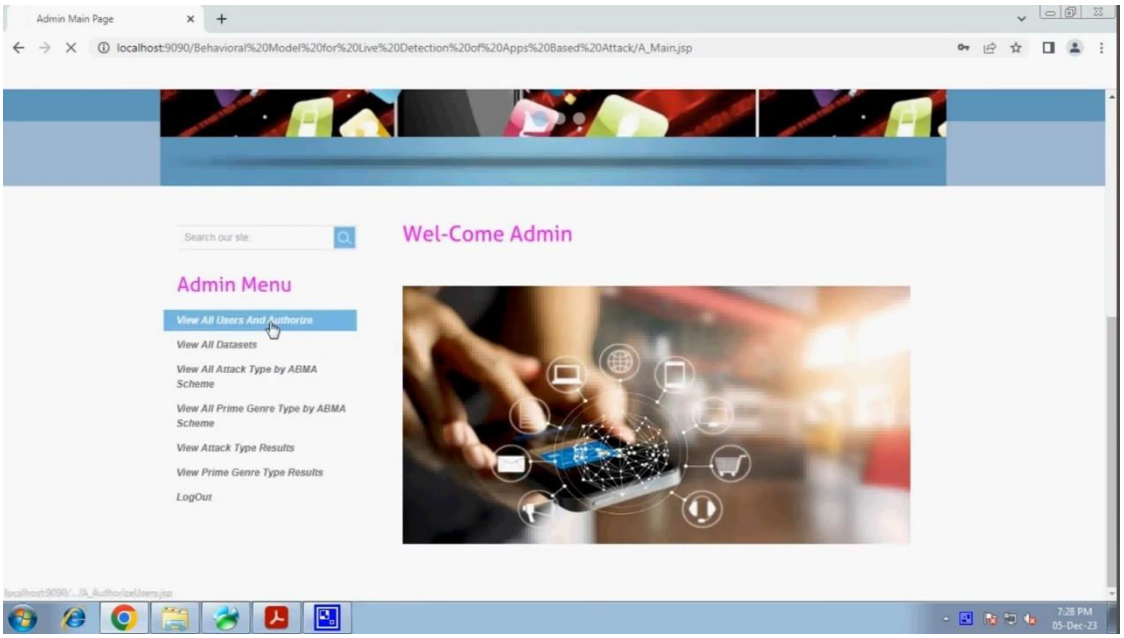
Home Page



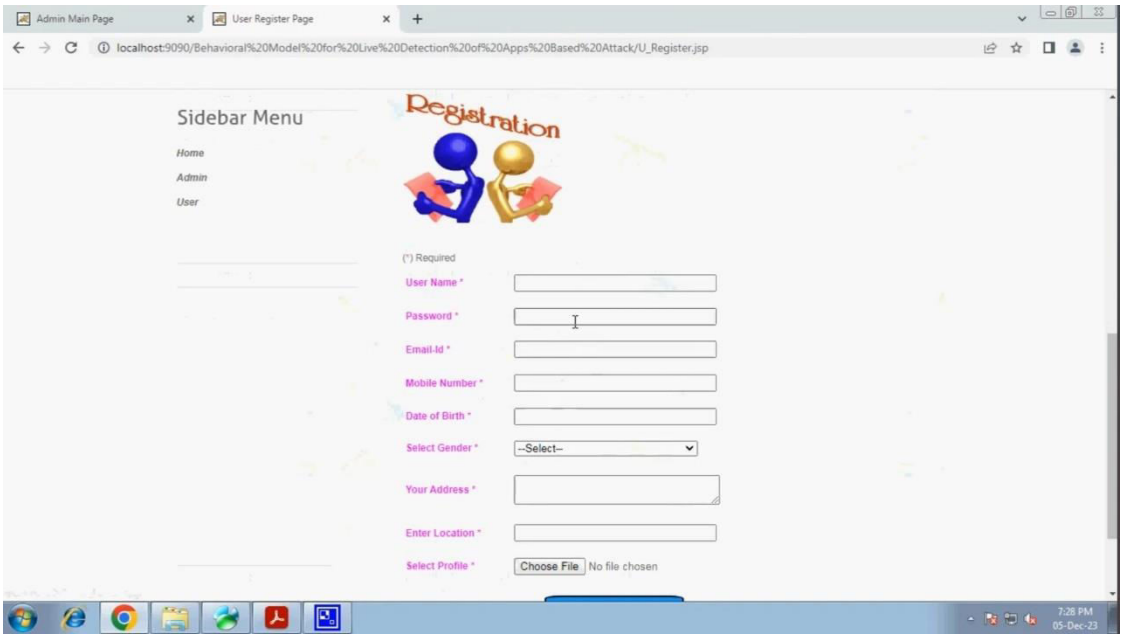
Admin Login Page



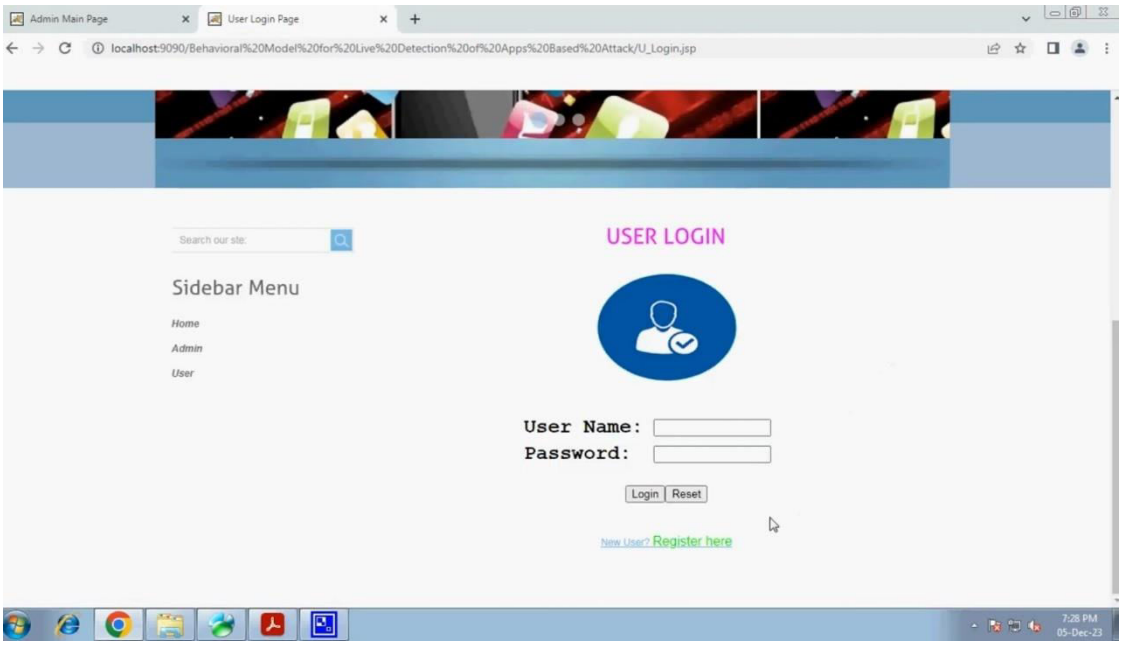
Welcome Admin Page



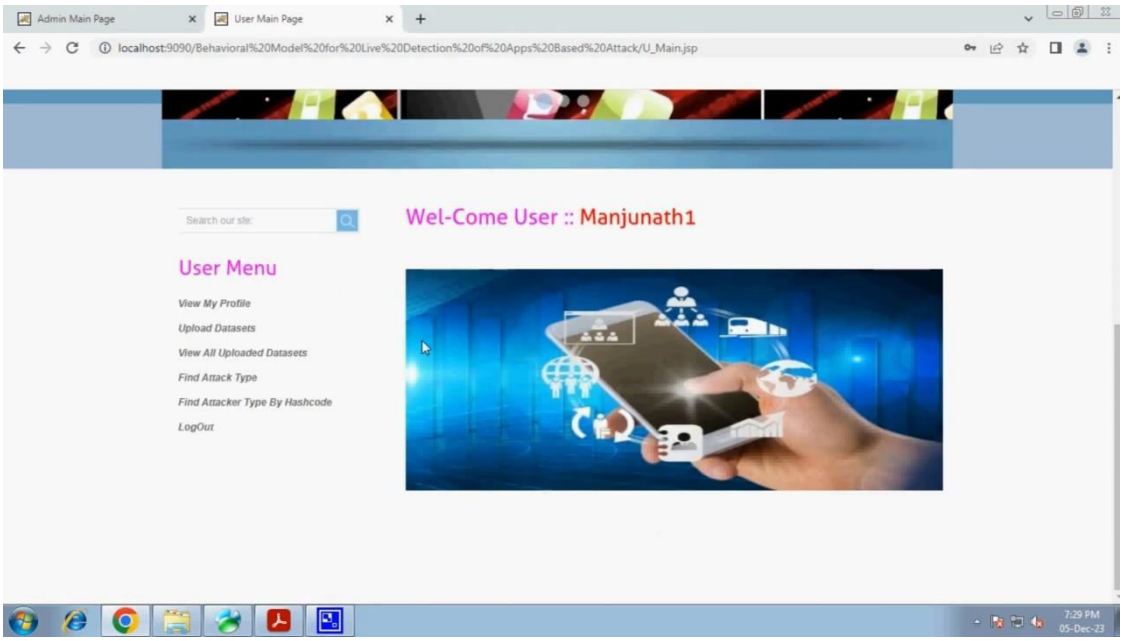
User Registration Page



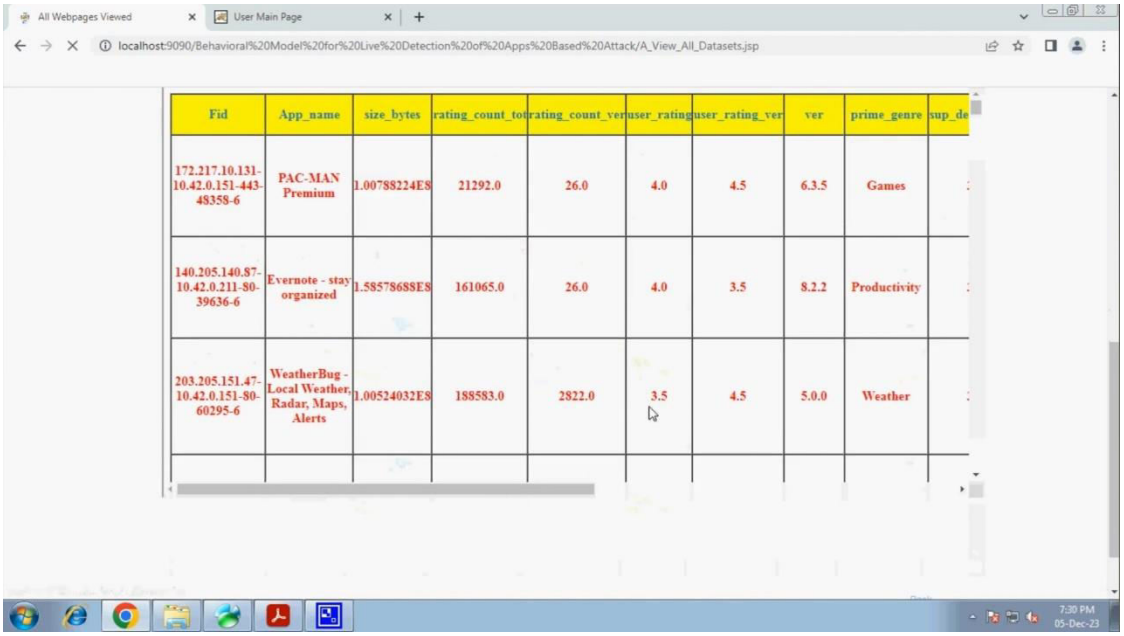
User Login Page



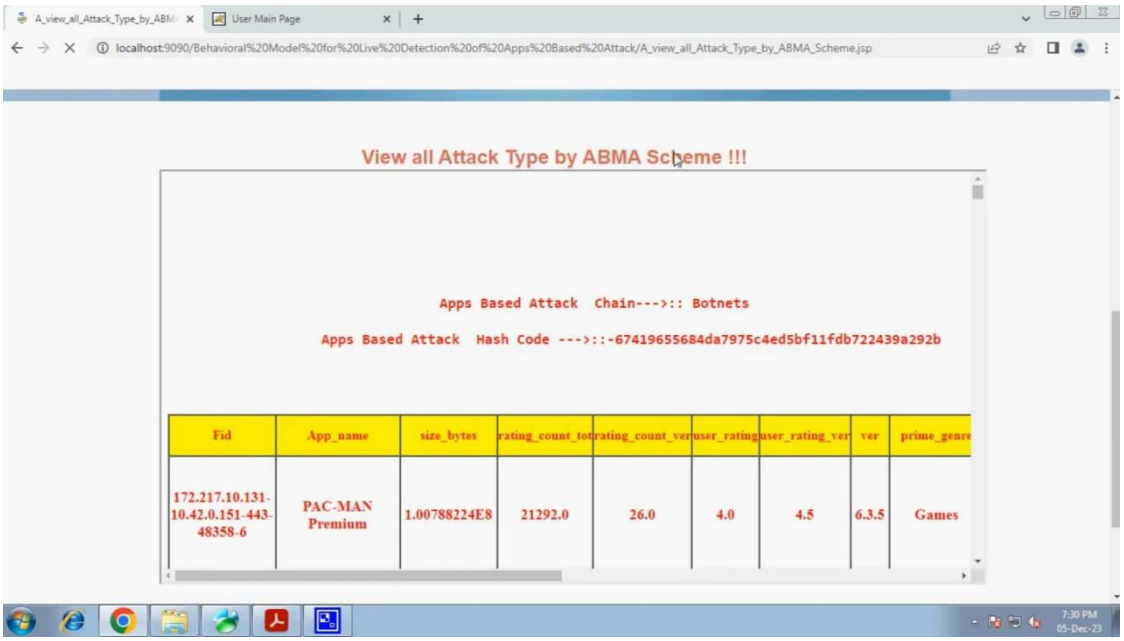
Welcome User Page



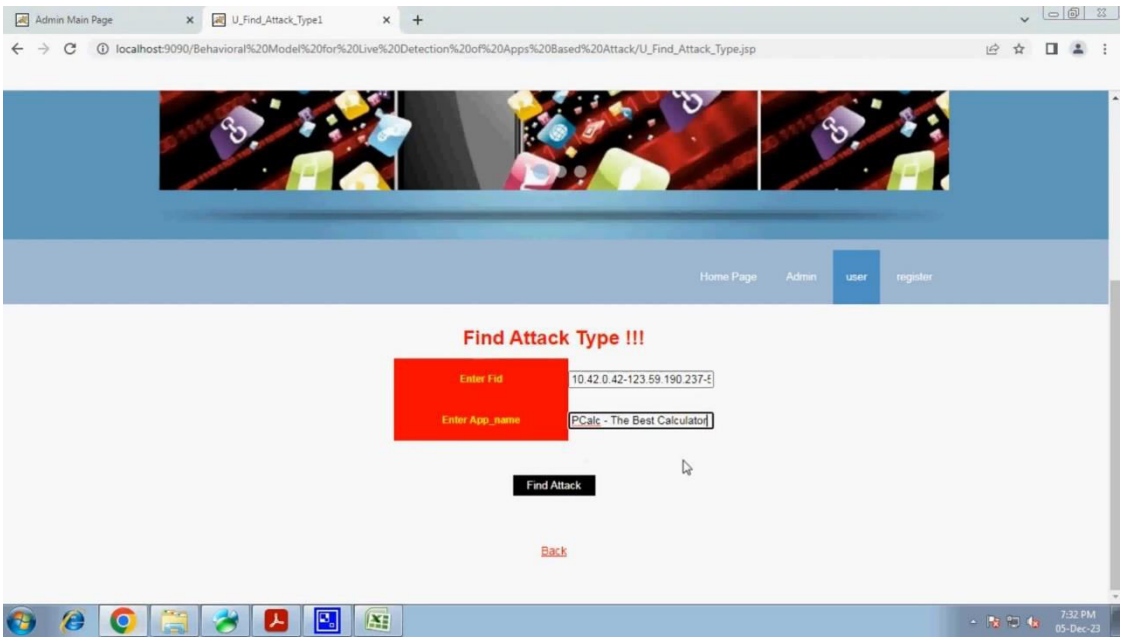
DATASET DETIALS



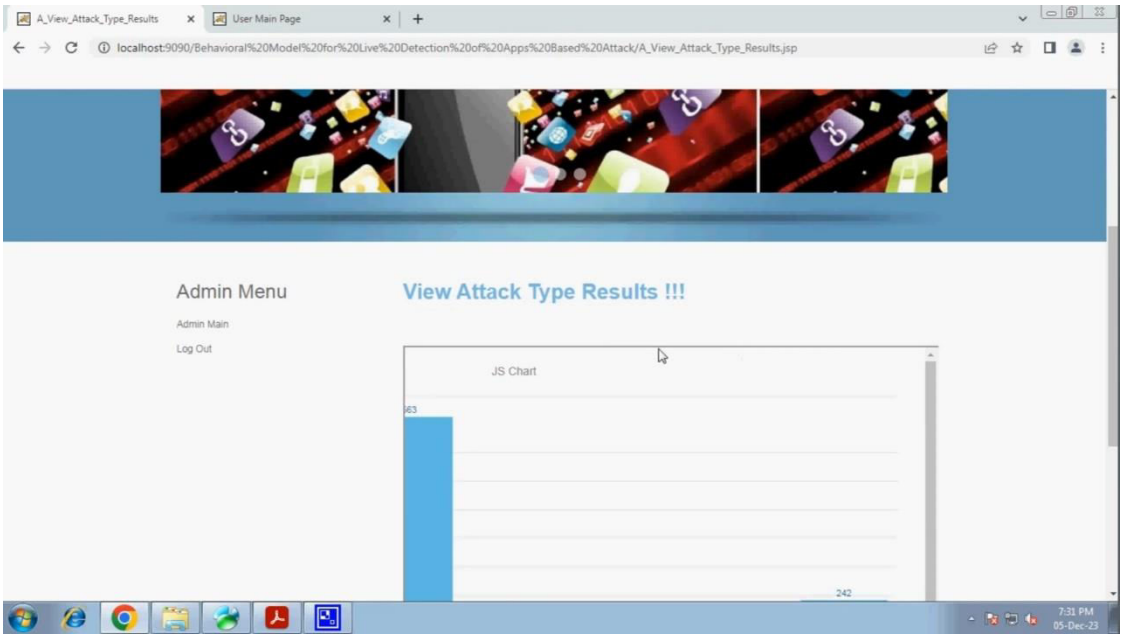
ATTACKS DETECTION USING ABMA ALOGRITHAM

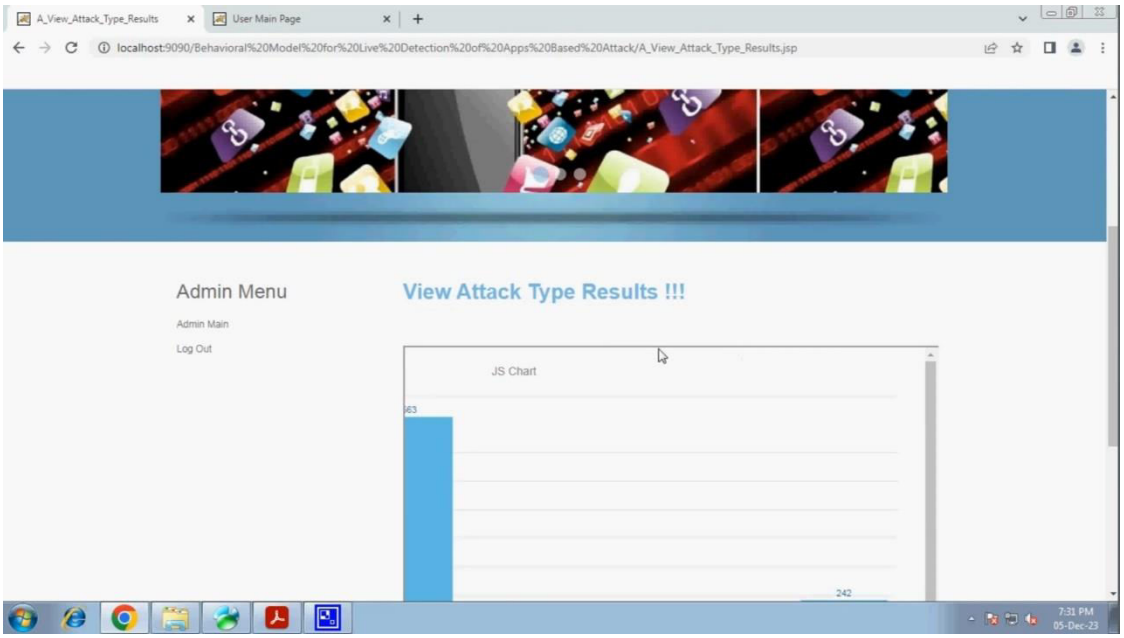


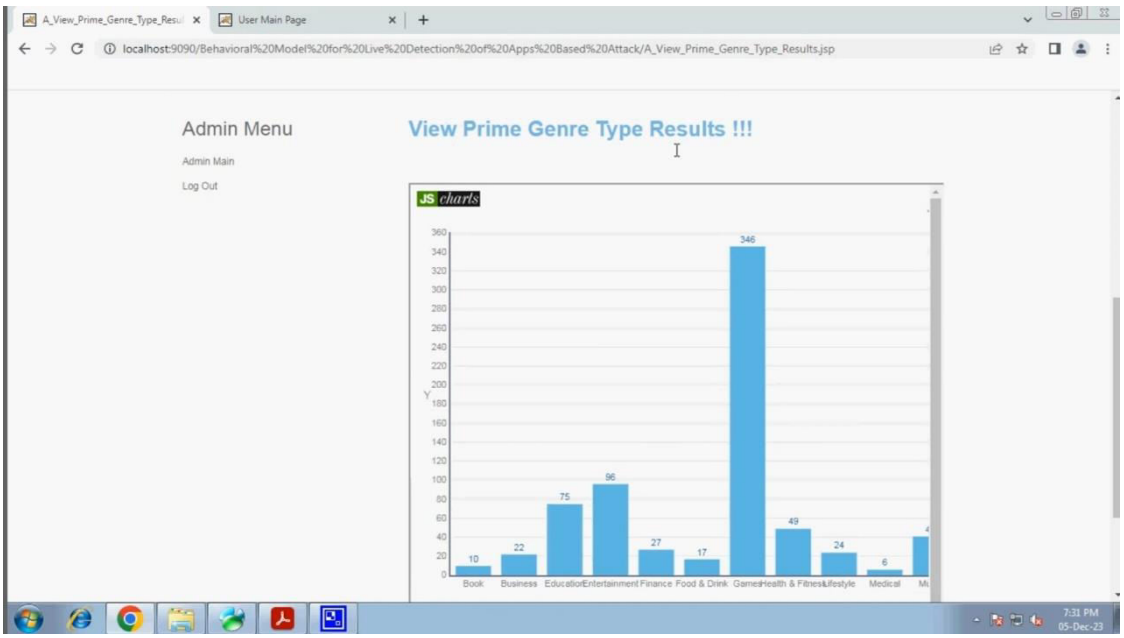
FINDING ATTACKS TYPE



FINDING ATTACK TYPE RESULTS IN GRAPH







II. CONCLUSION

The increase in the use of applications on the smart phone has enhanced numerous vulnerabilities and threats in the form of loss in

confidentiality, invalid access control permissions, and invalid authorizations, links to vulnerable sources. In this paper, an application-based attack modeling and attack detection is proposed to address such challenges. The attack modeling incorporates the end-user vulnerable application installation on the smart phone. The possible installation integrates hidden visibility activation mode to process the mechanism. The detection process evaluates ABMA scheme for the invalid application entry. The application-based analysis is estimated using power consumption, battery level, and data usage. The comparative analysis is observed for application intrusion detection. For the immediate countermeasure of the attack, an alarm is raised followed by the disconnection of cellular services and internet accessibility.

REFERENCES

[1] A. of Chief Police Officers, "Good practice guide for computer based electronic evidence," ACPO, Tech. Rep.

[2] K. Kent, S. Chevalier, T. Grance, and H. Dang, "Guide to integrating forensic techniques into incident response," National Institute of Standards and Technology, Tech. Rep.

[3] J. Dykstra and A. T. Sherman, "Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques," Digital Investigation, vol. 9, 2012, pp. S90– S98.

[4] "Sleuth Hadoop," http://www.sleuthkit.org/tsk hadoop/, retrieved April 2013.

[5] P. Mell and T. Grance, "The NIST definition of cloud computing," http: //csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf.

[6] J. Erickson, M. Rhodes, S. Spence, D. Banks, J. Rutherford, E. Simpson, G. Belrose, and R. Perry, "Content-centeredcollaboration spaces in the cloud," IEEE Internet Computing, vol. 13, September 2009, pp. 34–42.

[7] D. D. Roure, C. Goble, and R. Stevens, "The design and realisation of the myexperiment virtual research environment forsocial sharing of workflows," Future Generation Computer Systems, vol. 25, no. 5, 2009, pp. 561 – 567.

[8] I. Foster, "Globus online: Accelerating and democratizing science through cloud-based services," Internet Computing, IEEE,vol. 15, no. 3, May-June 2011, pp. 70 –73.

[9] S. Caton and O. Rana, "Towards autonomic management for cloud services based upon volunteered resources," Concurrencyand Computation: Practice and Experience, 2011.

[10] S. Distefano, V. D. Cunsolo, A. Puliafito, and M. Scarpa, "Cloud@home: A new enhanced computing paradigm," inHandbook of Cloud Computing, B. Furht and A. Escalante, Eds. Springer US, 2010, pp. 575–594.

[11] A. Chandra and J. Weissman, "Nebulas: using distributed voluntary resources to build clouds," in Proceedings of the 2009conference on Hot topics in cloud computing. USENIX Association, 2009.

[12] S. Xu and M. Yung, "Socialclouds: Concept, security architecture and some mechanisms," in Trusted Systems, ser. LectureNotes in Computer Science, L. Chen and M. Yung, Eds. Springer Berlin / Heidelberg, 2010, vol. 6163, pp. 104–128.

[13] "Amazon EC2," http://aws.amazon.com/ec2/, retrieved April 2013.

[14] Y. Song, H. Wang, Y. Li, B. Feng, and Y. Sun, "Multi-tiered on-demand resource scheduling for vm-based data center," in Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, ser. CCGRID "09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 148–155.

[15] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," Commun. ACM, vol. 51, Jan. 2008, pp. 107–113.