IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.67

Volume 5, Issue 1, February 2025

Juice Jacking Defender

Dr. Vaibhav Gandhi[1], Sasyareeth Malyala[2], Aditi Panchal[3], Obili Yashwanth Reddy[4], Rajesh Panigrahi[5]

Director and Faculty of Computer Science Engineering[1]
Students, Department of Computer Science Engineering[2,3,4,5]
Parul University, Vadodara, Gujarat, India

Abstract*:* *In an increasingly digital world, the proliferation of mobile devices has led to a growing dependence on public charging stations. However, these ubiquitous amenities also present a hidden danger: the threat of juice jacking. Juice jacking occurs when unsuspecting users connect their devices to compromised charging ports, inadvertently exposing their sensitive data to malicious actors. The "Juice Jacking Defender" project emerges as a proactive response to this pressing security concern. Through meticulous research, design, and development, this initiative seeks to create a robust defense mechanism against juice jacking attacks. By leveraging innovative technology, including portable devices or software solutions, the project aims to empower users with the means to detect and prevent potential security breaches while charging their devices in public spaces. Key components of the project include comprehensive analysis of existing juice jacking vulnerabilities and attack vectors, the creation of user-friendly interfaces for seamless interaction, and the implementation of sophisticated security measures to safeguard user data and device integrity. Rigorous testing ensures the effectiveness of the solution across diverse charging environments and device configurations. Moreover, the "Juice Jacking Defender" project extends beyond mere technical implementation. It places a strong emphasis on user education and awareness, striving to inform individuals about the risks associated with public charging stations and promote best practices for secure charging habits. Additionally, the project fosters collaboration with industry stakeholders, device manufacturers, and cybersecurity experts to advocate for improved security standards and regulations governing public charging infrastructure. Through continuous evaluation, feedback integration, and future enhancements, the "Juice Jacking Defender" project aims to evolve alongside emerging threats and technological advancements. By championing innovation, education, and collaboration, it seeks to establish a safer charging environment, thereby ensuring user confidence and privacy in an increasingly interconnected world.*

Keywords: Juice Jacking, USB Security Data Isolation, Public Charging Risks, Data Theft Prevention, USB Data Blocker ,Mobile Device Protection, Public Charging Stations ,Malware Defense, USB Data Isolation , IoT Security, Unauthorized Data Access , Cyber Attack Mitigation, Security Patch Analysis

I. INTRODUCTION

"Juice jacking is a type of cyberattack where malicious software is installed on a device through a compromised USB charging port or cable. This occurs when charging phones or other devices at public stations such as airports, malls, railway stations or cafes. As the world is moving rapidly towards digital transformation likely growing rapidly to many opportunities in every field i.e. in every aspect, the internet also has many advantages and disadvantages. We all know about the advantages but the major and important disadvantage that everyone should be aware of is an increase in cyber-attacks. As per the study, many government websites and systems were hacked in the past few years and it has caused a huge loss to nations like India, the USA, and China [1]. Cyber security is defined as technologies and processes constructed to protect computers, computer hardware, software, networks, and data from unauthorized access, and vulnerabilities supplied through the Internet by cybercriminals, terrorist groups, and hackers. Cybersecurity is now considered an important part of individuals and families as well as organizations, governments, educational institutes, and businesses. It is very essential nowadays to secure every individual from children to old people from online fraud. There is a vital need for internet users to understand how to protect themselves from identity theft. Appropriate learning

ISSN
2581-9429
IJARSCT

DOI: 10.48175/IJARSCT-23219

about online behavior and system protection results reduction in vulnerabilities and a safer online environment. [1] This paper discovers a new threat vector against charging phones in Public Places.

Scope of the project:

The scope of the ”Juice Jacking Defender” project aims to safeguard users against the security risks posed by public charging stations, specifically the threat of juice jacking, where charging points can compromise device data. The project involves thorough research into existing attacks and user behaviors, followed by the design and development of a user-friendly solution capable of detecting and preventing juice jacking attacks. Testing ensures its effectiveness, while deployment involves distribution, marketing, and user support. Maintenance includes regular updates and user education to keep the solution current and raise awareness about the risks. The project emphasizes ongoing evaluation and adaptation to address evolving threats and user needs in the mobile device security landscape.

1.1 Juice Jacking and the Need for Defense Mechanisms:

In an increasingly mobile world, public charging stations have become a common convenience for smartphone and device users. However, this convenience comes with significant cybersecurity risks, most notably a threat known as "juice jacking." Juice jacking refers to the malicious exploitation of USB charging ports, where attackers use the data connection within the USB to install malware or steal sensitive information from connected devices. With the growing reliance on mobile devices for both personal and professional purposes, the potential for data theft through public charging stations represents a critical vulnerability in the cybersecurity landscape.

This research paper explores the threat of juice jacking and examines various defense mechanisms designed to mitigate this risk. As the line between power charging and data transfer blurs, it becomes essential to develop and adopt secure technologies that protect users from unauthorized access. Through an analysis of current security measures, such as USB data blockers and software-based defenses, this paper aims to highlight the importance of enhancing public awareness and implementing robust solutions to safeguard against juice jacking attacks. By focusing on both technical and user-centric approaches to security, the study will provide a comprehensive overview of the potential solutions to this emerging threat. As public charging stations continue to proliferate in airports, cafes, and other public spaces, understanding and addressing the risks of juice jacking is vital for ensuring the privacy and security of users' digital lives.

1.2 Overview of Juice Jacking Defenders:

Juice Jacking Defenders are solutions designed to prevent such attacks by blocking data transfer while allowing safe charging. These defenses can be categorized into:

Physical Devices (USB Data Blockers): These devices act as an intermediary between the charging station and the user’s device. They block data pins in the USB connector, allowing only power to pass through. By eliminating the data connection, they prevent unauthorized data transfer or malware injection.

Software-based Defenses: Modern operating systems include features that prompt users when connecting to unfamiliar USB ports, asking whether they trust the source for data transfer or just power. Additionally, mobile devices can be configured to "charge-only" mode, which disables data transfer automatically when connected to public chargers.

1.3 Types of Juice Jacking Defenders

- USB Data Blockers: These are small adapters that prevent the USB port from transmitting data. They are inexpensive and widely available. However, their security depends entirely on the quality of the hardware, as poorly made blockers might not offer full protection
- Charge-only Cables: Similar to data blockers, these cables have been designed without data transfer capabilities. They are a more permanent solution compared to data blockers since the cable itself is designed solely for power transmission.
- Mobile Device Security Settings: Android and iOS devices offer security settings that limit data transfer when connected to public USB ports. For example, the latest iOS versions require explicit user approval for any data transfer initiated when connecting to a new device or port.

1.4 Significance :

Research on juice jacking defense is increasingly essential due to the widespread use of public USB charging stations, which can potentially expose users to data theft or malware installation. Juice jacking refers to a cyber threat where attackers manipulate public USB ports or charging cables to access private data or inject malicious code onto devices connected to these compromised charging points. With the growing reliance on mobile devices and the increasing availability of public charging stations, users often unknowingly expose themselves to these risks. Developing and researching effective juice jacking defenses, like isolators, data blockers, or alert systems, is crucial in mitigating these risks, especially as attackers use more sophisticated techniques to access sensitive information.

This research is necessary to increase awareness of this lesser-known but highly impactful threat, develop robust technical defenses, and establish best practices for safer mobile device charging. Juice jacking countermeasures can improve user trust in public charging infrastructure and enhance overall cybersecurity for individuals and enterprises alike. In high-security environments like government, corporate, or medical facilities, defending against juice jacking can protect valuable data and reduce the likelihood of security breaches. Therefore, this research not only contributes to personal security but also reinforces broader data protection standards in our increasingly mobile and interconnected society.

II. METHODS AND MATERIALS

2.1 Hardware-Based Solutions:

USB Data Blockers (USB Condoms)

A basic USB data blocker has no active electronic components but instead modifies the circuit so only power is transmitted. Advanced versions might include additional circuitry to monitor voltage or signal anomalies, indicating tampering.

Simple blockers can prevent data transfer but don’t offer dynamic defenses, such as recognizing specific threats or changes in the environment. They offer no protection if the charging port itself is compromised and attempts voltage manipulation. Integrating passive and active filtering systems to detect potentially malicious data attempts or voltage fluctuations. Combining USB data blockers with software-based notifications on the device side to alert users if there are any unexpected behaviors.

Dedicated Charge-Only Cables

Charge-only cables remove or disconnect the data wires in the cable itself, ensuring that only power flows through the connection. These are particularly useful in situations where the user doesn't have access to a dedicated USB data blocker but needs to use public charging stations regularly. The cable’s construction physically prevents data transmission by disconnecting or eliminating the data lines. Certain charge-only cables could be designed with locking mechanisms that prevent unauthorized devices from connecting, adding a layer of physical security

Security-Enhanced USB Ports on Devices

This ensures that even if the user connects to a malicious charging port, the device itself has internal mechanisms to block data theft or malware. Devices could come with dual-use USB ports that allow power-only connections, physically preventing any data transmission when the device is in charging mode. Manufacturers could embed security chips in devices that monitor USB activity and block any data-related actions unless explicitly authorized by the user.

Smart USB Hubs and Cables

Smart USB hubs and cables can provide an added layer of protection by being equipped with microcontrollers that can detect and block unwanted data connections. A microcontroller inside the hub or cable monitors the connection and ensures that only the necessary data transfer protocols are engaged. These smart hubs can also communicate with the connected device’s operating system, providing notifications if unusual activity is detected.

DOI: 10.48175/IJARSCT-23219

ISSN
2581-9429
IJARSCT

2.2 Software-Based Solutions:

Operating System-Level Protections

Some operating systems now automatically switch devices to "charge-only" mode when connected to a public or unknown USB port. The device will only transfer power and restrict data transfer unless explicitly authorized by the user.

Real-Time Malware Detection and Prevention

Software that actively monitors for malware, particularly focused on threats associated with USB connections, can offer another layer of protection. This kind of real-time detection is especially important as malware could be transferred to a device without the user's awareness. Many modern antivirus and anti-malware programs offer USB scanning features, detecting potential threats when a device is connected to a USB port. These tools could be enhanced to specifically address juice jacking scenarios by focusing on unusual data transfer patterns that could indicate an ongoing attack. Advanced software can deploy Intrusion Detection Systems that monitor all input/output operations at the USB port. If any unauthorized or suspicious activity is detected, the IDS can terminate the connection or raise an alert.

2.3 Tools

- ESP32
- ADuM4160 (data isolator)
- INA219 (data logger)
- SSD1306 (display)
- Buzzer
- Power supply

2.4 Working of the Tools :

ESP32: The ESP32 microcontroller, developed by Espressif Systems, is renowned for its versatility, combining low power consumption with robust wireless connectivity and high processing capabilities. Its key features include a dual-core Tensilica Xtensa LX6 processor, which operates at speeds up to 240 MHz, making it ideal for both real-time tasks and complex computations. The ESP32 supports both 2.4 GHz Wi-Fi (802.11b/g/n) and Bluetooth, enabling it to connect to a wide array of wireless devices and networks. This makes it perfect for Internet of Things (IoT) applications, where seamless communication is crucial. The ESP32 has a rich set of peripherals, including GPIOs (General-Purpose Input/Output), ADC(Analog-to-Digital Converters),DAC(Digital-to-Analog Converters), PWM (Pulse Width Modulation), and integrated hardware for SPI, I2C, UART, and CAN communication protocols, enabling connectivity with various sensors and actuators. Additionally, its low-power modes, such as deep sleep, make it suitable for energy-sensitive applications like battery-powered devices. The microcontroller also supports advanced security features like secure boot, flash encryption, and TLS/SSL for secure data communication, making it reliable in applications that demand high levels of data security. Its support for development environments like Arduino IDE and ESP-IDF it simplifies programming.

ADuM4160 (data isolator): The ADuM4160 works by isolating the USB data lines (D+ and D–) while allowing bidirectional communication between the host and peripheral devices. It supports low-speed (1.5 Mbps) and full-speed (12 Mbps) USB communication and can be used in both host and peripheral modes, thanks to its flexible pin configuration for direction control. The isolator itself requires no external drivers, making it simple to integrate into USB designs. Additionally, the ADuM4160 eliminates the need for optocouplers or transformers, reducing design complexity and power consumption. A critical feature is its ability to isolate both power and data. The ADuM4160 allows power isolation on the VBUS line, ensuring that the downstream USB device receives isolated power, which is especially important in applications where electrical safety is critical, such as in medical-grade USB interfaces.

INA219 (data logger): The INA219 is a high-side shunt and power monitor with an I2C interface, commonly used for measuring voltage, current, and power in low-power systems. Using it for a juice jacking defender research paper can involve tracking the power draw of a device during charging to detect malicious behavior, such as unauthorized data

ISSN 2581-9429 IJARSCT

transfers or unusual power consumption patterns. The INA219 would be connected between the power source (USB port) and the target device to log electrical parameters. An Arduino, ESP32, or Raspberry Pi could be used to interface with the INA219 via I2C and log the data. Logged data would be stored locally on an SD card or sent wirelessly for analysis. Abnormal patterns could trigger alerts.

SSD1306 (display): The display could alert users if any data transmission attempt is detected while charging. It can notify users if the connection is purely power or if there is a potential data breach attempt. It can show the current status of the connection, such as "Safe Charging Mode" (power only) or "Potential Data Transfer Detected." This visual confirmation can provide peace of mind or act as a warning. Although the SSD1306 is a small screen, it could be used in conjunction with buttons to allow users to configure the defender device or check logs. The SSD1306 can display power draw information, helping users verify if the charging device is safe. A warning message if the defender detects unauthorized data transmission attempts. The design highlights the importance of user awareness in security devices. It allows the paper to emphasize the role of real-time, easy-to-understand feedback in preventing juice jacking.

Buzzer: In the Juice Jacking Defender, a buzzer plays a crucial role as an audible alert system, providing real-time feedback to the user when suspicious or unauthorized activities are detected. The buzzer is integrated into the system to emit a sound in scenarios where the USB charging port exhibits unusual behaviour, such as unauthorized data transfer attempts during device charging, which is a hallmark of juice jacking attacks. The buzzer is triggered by the detection mechanism, which continuously monitors the communication lines (D+ and D–) for any data transfer initiation while charging.When the system identifies potential juice jacking activities, the buzzer instantly alerts the user through a loud, distinctive sound, allowing them to disconnect the device before any data is compromised. This audio feedback ensures that users are immediately informed, even if they are not actively monitoring the device. The buzzer is typically powered by the system's microcontroller (such as ESP32 or Arduino) and operates at low voltage, making it an energy-efficient yet effective alert component.

III. LITERATURE REVIEW

Research on defending against juice jacking, the cyber-attack method exploiting public USB charging ports for unauthorized data access or malware installation, has emphasized multiple technical and preventive approaches to enhance user security. Yuvraj Kumar's article, "Juice Jacking - The USB Charger Scam," provides an overview of juice jacking, highlighting how attackers target public charging points to gain unauthorized access to connected devices, exploiting both personal and commercial data vulnerabilities through modified USB ports (Kumar, 2020). Another study,{ "Analisis Patch Keamanan Android Terhadap Serangan Juice Jacking," by Fakhriansyah and Setiawan}, focuses on how Android security patches address juice jacking risks. The study stresses the significance of secure USB connections, as well as the need for smartphone OS improvements to counteract juice jacking by blocking unauthorized data access during charging {(Fakhriansyah & Setiawan, 2024)}. This literature illustrates how juice jacking defenses have evolved from awareness-focused strategies to more robust technical defenses in mobile operating systems, enabling secure, data-isolated charging in public settings. Researchers advocate for improved hardware and software solutions, such as USB data blockers or isolators, which provide critical safeguards for consumers while using public infrastructure.

IV. METHODOLOGY

Malware attacks in today's world are on the rise and are causing damage to organizations' reputations so immediate action must be taken to detect the malware. In terms of detecting malware, there is a lack of focus on scalability and interpretability of ensemble methodology and selection of necessary features, as well as on adequately adapting ML models to developing malware techniques. In comparison to the previous study mentioned above, this research focuses on enhancing the malware detection system which is done by achieving high accuracy as shown in the evaluation section of the report. This results in increasing the effectiveness and robustness in classifying or handling diverse malware instances. Systems can be made more effective and efficient by selecting features that are important for malware identification in cybersecurity. This increases model accuracy, reduces computing burden, and makes the system more interpretable. Cyber threat identification and mitigation are both facilitated by this optimization. The research involves a comprehensive analysis of the suggested hybrid ensemble model's performance for malware

ISSN
2581-9429
IJARSCT

detection in PE files using tree-based algorithms, and the stacking classifier, within the hybrid ensemble framework. It involves training and evaluating each classifier on the dataset. Understanding of the model's robustness and effectiveness in managing varied malware instances are provided by the evaluation metrics, which help classify PE files as safe or not safe.

V. SYSTEM DESIGN

Data preparation is a critical step in this research. This step includes several critical actions that must be completed to ensure the quality of data and suitability for effective model training. A key component is data preparation, which comprises translating categorical variables into numerical form, which is required by machine learning algorithms. In this project, class imbalance concerns are addressed by balancing the dataset using the SMOTE function which stands for Synthetic Minority Over-sampling Technique. By prioritizing essential features and excluding irrelevant ones, the model's training process is more focused, leading to a reduction in overfitting and enhanced generalization of malware samples. This was done with the help of using the ”Pefile library. In certain situations, some features may not provide valuable information so it makes sense to remove or combine them with other similar features to reduce data dimensionality. This focused feature selection has not only helped to optimize the efficiency and accuracy of the model but has also made it more interpretable. By following the above-mentioned process, the research gains clearer insights into the critical factors influencing malware classification, facilitating informed decision-making. Moreover, the dataset has been optimized ensuring a more scalable and practical solution for malware detection. The next step was to arrange these features in a feature dataset. One of the most important steps in ML is separating the feature dataset into training and testing sets for malware identification. Importantly, the training set exposes the machine learning models to a wide variety of PE files, which helps them learn the data. The research follows an organized procedure with training numerous ensembles of learning algorithms (including Adaboost, Random Forests, Gradient Boosting, and Stacking) on a training dataset, which was accomplished using Google Colab. The models can then learn to distinguish between malicious and safe files based on these patterns. However, the training model's capacity to categorize data is evaluated on the testing set. This guarantees that the models improve their dependability and performance in real-world situations by generalizing well beyond the training data. Individual models are trained and combined with the stacking classifier for the final prediction. The next step involves integrating the machine learning models into a web application and also moves on to the building of a PE file classification system utilizing Flask. In this research, Flask (Braganca and Kho, 2023) is preferred for web applications due to its lightweight design, simplicity, and ease of integration compared to other frameworks. Its minimalistic structure allows for quick deployment and efficient handling of requests. This simple application allows users to submit executable files for risk evaluation. When a file is submitted, the pre-trained Stacking Classifier examines its properties to estimate its risk level—high or low risk based on malware or benign classification, respectively. The Flask-based application incorporates the pre-trained Stacking Classifier. The interface provides a user-friendly application for individuals to communicate with the malware detection system, simplifying the process of uploading the exe files and understanding the associated risk levels. When the model classifies a file as malicious, the system generates high-risk alerts, signaling a potential security threat that requires immediate attention and action. Conversely, if the file is identified as benign, the application issues low-risk alerts, providing users with confidence in the safety of the file. This approach allows users to proactively take security steps based on the severity of the risk identified by the model, contributing to a more responsive and preemptive cybersecurity strategy

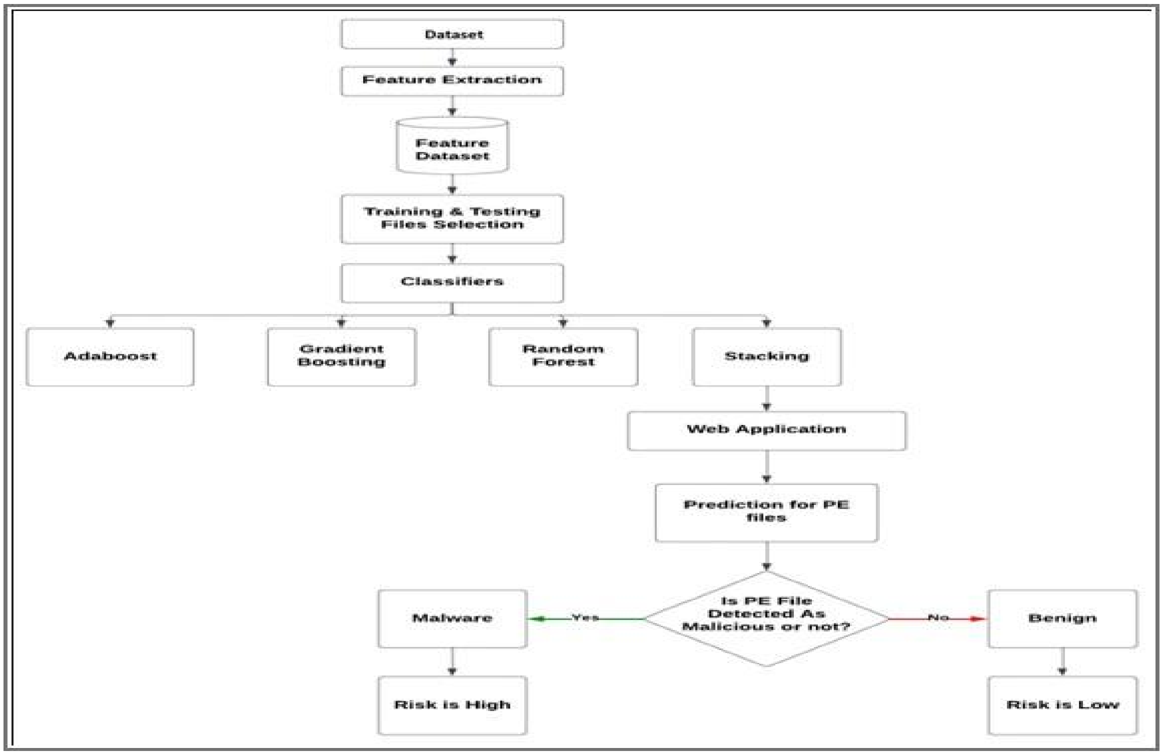
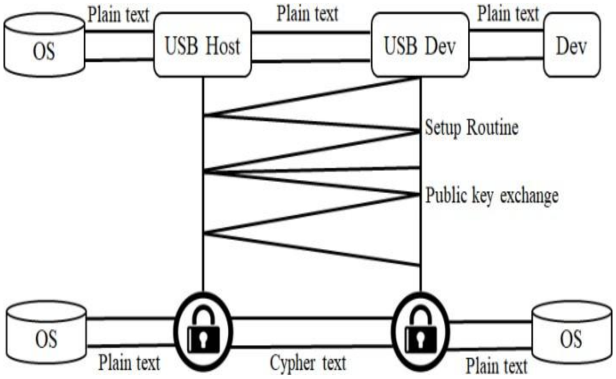


Figure: Design of software

5.1 USB Cable Encryption

There are many types of USB attacks, but some commonly used attacks are programmable microcontrollers, a bad USB, and a rubber ducky. The programmable microcontrollers form a well-known USB attack. Human interface devices (HID) such as keyboards and mice are used by USB microprocessors to execute keystroke commands on a target gadget. A bad USB can inexpensively cause a worrying attack. This attack exploits a flaw in the USB firmware. It reprograms the USB device so that it functions as a human interface. When a victim's computer is connected to a USB device, it can be used to execute commands or launch a malicious program. Rubber Ducky injects keystrokes at a rapid speed, violating the inherent trust computers have in humans by posing as a keyboard. It also saves a lot of effort by targeting susceptible systems or programming processes



5.2 Security challenges in USB Technology:

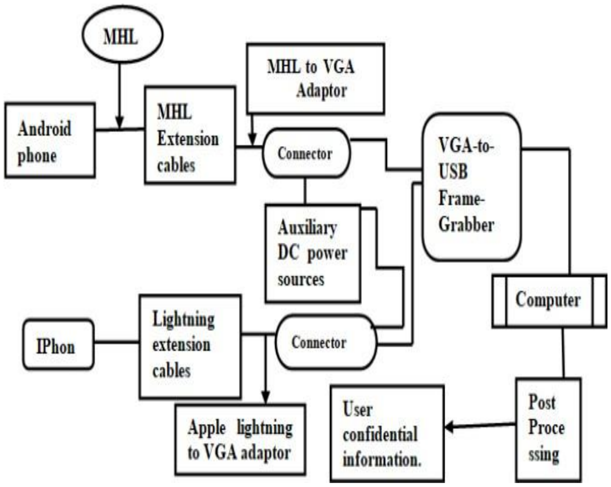
Retaining the confidentiality, sensitivity, and isolation of a bus line, how do these USB devices maintain their integrity? How does a USB line or traffic send data to a particular malicious node or an organization? To overcome the above

DOI: 10.48175/IJARSCT-23219

ISSN 2581-9429 IJARSCT

problems, a software overlay can be used for encryption on used devices. For secure communication, these three techniques mainly focus on the validity of the information, data confidentiality, and preventing other devices and different attacks As the data and information transmitted through the plain text and encryption techniques are applied to secure the bus line , both the USB device and the host controller built a secure channel to improve the USB line using the concept of encryption. A secure socket layer (SSL) can be used for secure communication through the Internet. Through an encrypted channel on a bus line, both the host and the device can initiate a key exchange. As long as both devices use the same cipher, they can exchange data. Private as well as public keys are chosen to decrypt the data, encryption setup, and key exchange concepts, To ensure that data integrity is maintained, the USB needs to be improved. For secure communication, some type of verification and authentication is required to prevent data from being exploited or tampered with in any way [38]. Digital signatures and public key cryptography can be used to validate and optimize firmware queries.

Real setup of Juice Jacking filming charging attack Using VGA2USB:



The architecture of the juice filming charging (JFC) attack is based on the consideration that no acceptance would be asked for when we plug android or iPhones into a projector; then, the projector can automatically display the phone screen and does not show any bulletin for a plug-in device. VGA/USB allows the attacker to capture user inputs such as password, PIN code, or email address. Displays can be exposed via a regular microUSB connector that supports the mobile high-definition link (MHL) standard and, for iPhones, a lighting connector is used instead of the micro-USB connector. Figure 4: When a user connects their phone to a JFC charger, the phone’s screen can be used to collect the video into several video files at the back-end while this procedure extracts private information from the user. A red, green, blue (RGB) VGA frame-grabber is responsible for the conversion of a video signal from VGA to USB.

VI. CONCLUSION

The ”Juice Jacking Defender” project endeavors to address the growing concern of security vulnerabilities present in public charging stations. By leveraging research, design, and development efforts, the project aims to equip users with a reliable defense mechanism against juice jacking attacks. Through ongoing education, maintenance, and advocacy, the project seeks to foster a safer charging environment, ensuring that individuals can charge their devices with confidence while safeguarding their sensitive data and privacy. Ultimately, the project’s success lies in its ability to empower users and promote awareness, thereby enhancing overall security in the mobile device ecosystem. Juice jacking is a well-known cyber-attack used to attack USB-enabled devices. It generally utilizes the charging port of a given device, and whenever someone connects a given device to the system using this port, then hackers obtain personal information or may upload some malware onto the device. Therefore, it is necessary to detect and prevent these kinds of attacks. Therefore, in this paper, a juice jacking attack was analyzed using the maximum possible ways through which a system can be affected by USB. Ten different malware attacks were used for experimental purposes. The overall analyses were performed on 1250 participants. Initially, various stateless features were collected, and the target class was set as

ISSN 2581-9429 IJARSCT

malware type. Thereafter, various machine learning and deep learning models, such as J48, SVM, KNN, ANN, RF, ANFIS, and CNN, were utilized to build the juice jacking classification model. Categorical features were converted to numeric using one-hot encoded vectors. Z-score normalization was then used for normalizing the data. From comparative analyses, the deep learning model was shown to achieve an average improvement over the competitive models in terms of accuracy, Fscore, precision, recall, and hit rate, by 2.1789. Finally, various techniques that can either prevent or avoid juice jacking attacks were also discussed.

VII. FUTURE WORK

Future work for the ”Juice Jacking Defender” project could include

1. Enhanced Detection Capabilities.
2. Compatibility Expansion.
3. Integration with Mobile Platforms.
4. Cloud-Based Security Updates.
5. User Education and Awareness Campaigns.
6. Partnerships and Collaboration.
7. Integration with Public Infrastructure.
8. Behavioral Analysis and Anomaly Detection.
9. International Standards Advocacy.
10. Continuous Evaluation and Improvement.

REFERENCES

[1]. "Juice jacking", Wall of Sheep

[2]. Rowley, Robert, Juice jacking 101

[3]. Osborn, Kyle, P2P-ADB

[4]. BlackHat Briefings 2013 Mactans (PDF)

[5]. "BadUSB - On Accessories That Turn Evil", BlackHat Briefings USA 2014

[6]. Nohl, Karsten; Lell, Jakob, BadUSB Presentation at Blackhat USA 2014

[7]. "Turning USB peripherals into BadUSB", SRLabs.de

[8]. "Road Warriors: Beware of 'Video Jacking'", Krebs on Security

[9]. Iarchy, Roy, iOS Trustjacking

[10]. O.MG Cable

[11]. "Beware of Juice Jacking?", Krebs on Security

[12]. P2P-ADB on Hak5

[13]. "How American Spies Use iPhones and iPads", Fast Company

[14]. "Security Configuration Recommendations for Apple iOS 5 Devices. NSA Mitigations Group" (PDF)

[15]. Drake, Joshua; Lanier, Zach; Mulliner, Collin; Fora, Pau; Ridley, Stephen; Wicherski, Georg (March 2014). Android Hacker's Handbook. Wiley. p. 576. ISBN 978-1-118-60864-7.

[16]. "CSI:Cyber L0M1S", Vulture Screencap Recap

[17]. LADA Juice Jacking PSA

[18]. "Is Juice-Jacking via Public USB Ports a Real Security Threat?", Snopes

[19]. "New Android 4.2.2 Feature USB Debug Whitelist", Android Police

[20]. "Juice Jacking, Why You Should Be Concerned!". Custom Computers, Inc. 2019-11-22. Retrieved 2020-02-19.

[21]. Crossl, Robert. "Explained: juice jacking". Absolute Cental Technologies. Retrieved 2020-02-19.

[22]. "What is Data Theft?". www.computerhope.com. Retrieved 2020-02-19.

[23]. "What is Juice Jacking and how to prevent it & protect your smartphone". The Windows Club. 2017- 04-03. Retrieved 2020-02-19.

[24]. https://www.bloomberg.com/news/articles/2018-10-09/new-evidence-of-hacked- supermicrohardware-found-in-u-s-telecom

[25]. https://www.forbes.com/sites/daveywinder/2019/06/20/confirmed-nasa-has-been-hacked/

[26]. https://www.israel21c.org/how-to-prevent-juice-jacking-and-other-cyber-hacks/

[27]. McIntire, . G., Martin, B. & Washington, L., 2020. Python Pandas Tutorial: A Complete Introduction for Beginners. [Online]

[28]. Available at: https://www.learndatasci.com/tutorials/python-pandas-tutorial-complete-introduction-for-beginners/

[29]. Anon., 2019. PE File. [Online] Available at:https://resources.infosecinstitute.com/2-malware-researchers-handbook- demystifying-pe-file/#gref

[30]. Anon., 2019. spyware. [Online]
Available at: https://searchsecurity.techtarget.com/definition/spyware

[31]. Anon., 2020. Adware. [Online]
Available at: https://www.malwarebytes.com/adware/

[32]. Anon., 2020. Lucy: A File Encryption Android Malware that for Ransomware Operations. [Online] Available at: https://go.newsfusion.com//security/item/1638195

[33]. Anon., 2020. Microsoft Warns of Malware Hidden in Pirated Film Files. [Online] Available at: https://go.newsfusion.com//security/item/1638307

[34]. Anon., 2020. ROOTKIT: WHAT IS A ROOTKIT?. [Online] Available at: https://www.veracode.com/security/rootkit

[35]. Anon., 2020. Schneier on Security. [Online]
Available at: https://go.newsfusion.com//security/item/1641641

[36]. Anon., 2020. Welcome to wxPython!. [Online] Available at: https://wxpython.org/

[37]. Anon., 2020. What is a Computer Virus and its Types. [Online]
Available at: https://antivirus.comodo.com/blog/computer-safety/what-is-virus-and-its-definition/

[38]. Anon., n.d. CYBER EDU. [Online]
Available at: https://www.forcepoint.com/cyber-edu/malware [Accessed 2020].

[39]. Anon., n.d. Malware. [Online]
Available at: https://us.norton.com/internetsecurity-malware-what-is-a-trojan.html

[40]. Anon., n.d. malware (malicious software). [Online]
Available at: https://searchsecurity.techtarget.com/definition/malware

[41]. Anon., n.d. NumPy. [Online] Available at: https://numpy.org/

[42]. Anon., n.d. NumPy Introduction. [Online]
Available at: https://www.w3schools.com/python/numpy_intro.asp

[43]. Anon., n.d. Pandas Basics. [Online]
Available at: https://www.learnpython.org/en/Pandas_Basics

[44]. Anon., n.d. scikit-learn. [Online]
Available at: https://scikit-learn.org/stable/

[45]. Anon., n.d. What is a computer worm, and how does it work?. [Online]
Available at: https://us.norton.com/internetsecurity-malware-what-is-a-computer-worm.html

[46]. C, E., 2018. Are Antivirus Programs Effective. [Online]
Available at: https://www.safetydetectives.com/blog/are-antivirus-programs-effective/ [Accessed 2020].

[47]. Collins, M., Schapire, R. E. & Singer, Y., 2002. Logistic Regression, AdaBoost and Bregman Distances. IEEE.

[48]. Desarda, A., 2019. Understanding AdaBoost.

[49]. Dutta, A., 2019. Stacking in Machine Learning. [Online]
Available at: https://www.geeksforgeeks.org/stacking-in-machine-learning/

[50]. Fruhlinger, J., 2018. Ransomware explained: How it works and how to remove it. [Online]
Available at: https://www.csoonline.com/article/3236183/what-is-ransomware-how-it-works-and-how-to-remove-it.html

[51]. Gandhi, R., 2018. Naive Bayes Classifier. [Online]

DOI: 10.48175/IJARSCT-23219

ISSN
2581-9429
IJARSCT

Available at: https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c

[52]. G., G., Stiborek, M. & Zunino, J., 2014. An empirical comparison of botnet detection methods.” computers & security. IEEE.

[53]. Gupta, A., 2019. ML | Extra Tree Classifier for Feature Selection. [Online]
Available at: https://www.geeksforgeeks.org/ml-extra-tree-classifier-for-feature-selection/ [Accessed 2020].

[54]. Huilgol, P., 2019. Accuracy vs. F1-Score. [Online]
Available at: https://medium.com/analyticsvidhccuracy-vs-f1score6258237beca2 Hunter, J., 2020. matplotlib. [Online] Available at: https://matplotlib.org/

[55]. Loi, H. & Olmsted, A., 2017. Low-cost Detection of Backdoor Malware. IEEE.

[56]. Morde, V., 2019. XGBoost Algorithm: Long May She Reign!. [Online]
Available at: https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm- long-she-may-rein-edd9f99be63d

[57]. Muhamad, I. M. & Rahardjo, B., 2019. Malware Detection Using Honeypot and Machine Learning.IEEE.

[58]. Narkhede, S., 2018. Understanding AUC - ROC Curve. [Online]
Available at: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

DOI: 10.48175/IJARSCT-23219

ISSN
2581-9429
IJARSCT