PRINTSHOP: SERIAL PRINTER ENVIRONMENTS AND SECURITY

Research Proposal

Doctor of Philosophy

in

Cyber Operations

January 23, 2024

By

Micah Flack

Dissertation Chair:

Dr. Vaidyan Varghese

Dissertation Committee:

Dr. Yong Wang

Dr. Michael Ham

Beacom College of Computer and Cyber Sciences

TABLE OF CONTENTS

Introduction

1.1 Background

Serial printers are devices commonly used for instant reporting of system data for industrial control systems (ICS) and receipts for point-of-sale (POS) systems. These devices are connected to their host using Wi-Fi, bluetooth, ethernet, or USB; in some cases, serial RS232 is an option as well. The goal of this research is to assess what software and hardware protections are enabled, as well as, how configurable the serial printers are for further exploit research.

(a) Square POS (b) SurePOS

Figure 1.1: Comparison of common POS systems

Figure 1.1 shows us two similar looking point-of-sale systems, albeit one is much older looking. However, the operating system and required hardware is very different. Typically, unless you have the Square provided terminal, their software/client is installed onto an Android or iOS device and connected to a Square compatible card reader [1]. Whereas, the SurePoS, NCR, or other common EFTPoS system will run a proprietary OS based on Windows or Linux [2]. Furthermore, these PoS tend to require some form of printing receipts as record keeping for the business owner and customer. And these devices also vary in terms of processing capabilities and operating system.

For instance, a common thermal printer seen with PoS systems, integrated with fuel pumps, or other industrial control equipment, is the SNBC BTP-S80 thermal printer [3], [4]. There are multiple versions of the device with support for Bluetooth, USB only, or combination of USB/Serial/Ethernet. The bluetooth hardware is provided over an accessory 25-pin serial connection, with more I/O as a serial connection via RS232C connector and USB Type-B. It has driver support for various platforms: Android, iOS, Windows, Linux, and MacOS. The most interesting aspects are the processor, an Arm Cortex M4 clocked at 3.54MHz, and the operating system, a proprietary version of FreeRTOS. The system architecture is Armv7E-M with JTAG/SWD hardware debugging support [5], [6].

By default, the printer has enough headroom to process ESC/POS commands for printing paper and a webserver for debugging or general diagnostics. In theory, the uncompromised device could be flashed with modified firmware to act as a decoy and human-input-device (HID) against the host PoS. The viability of any vulnerabilities would likely be dependent upon supply chain attacks or physical bait-and-switch tactics [7].

1.2 Significance

According to the Federal Trade Commission (FTC), there were 37,932 reports of credit card fraud in 2012 and 87,451 reports in 2022. This marks an increase of credit card payment fraud by an estimated, 30.5%. By comparison, since 2020, there has been a 14.6% increase in credit card related fraud. Which does not include the millions of other fraud reports the FTC receives every year. In 2022 alone, there were around 5.1 million fraud, identity theft, and miscellaneous reports in total [8], [9]. The statistics for these reports stresses how crucial the security of payment systems are, both physical and online. And, the need to secure them grows every year.

Spyduino is [10] a working example of a programmable BadUSB device using an Arduino to mimic a Human Interface Device (HID). Arduinos are typically more accessible and easily developed compared to an embedded device whose design is more single purpose. Especially if the goal is to not modify hardware or require hands-on access for exploitation. However, the research shows us that is it possible create HID clones from scratch if the hardware is compatible.

The Arduino used in their research is powered by an ATmega328P microcontroller with 32KB flash memory, 2KB SRAM, and 1KB EEPROM. Compared to the most likely target device of our proposed research, the SNBC BTP-S80, it features an ARM Cortex M4 microcontroller with 512KB flash memory, 96KB SRAM, 4KB of EEPROM. BadUSBs are a known and tested area of research. The novelty of this proposal comes from the assessment of the printer devices and showing whether one could be used maliciously within their environments (e.g., PoS systems, or ICS).

1.3 Research Goals and Objectives

This research primarily focuses on physical POS systems or terminals and their hardware (serial accessories), rather than online solutions. For instance, not mobile payment apps like Venmo, CashApp, Zelle, or Paypal [11]. There are many reasons, but the types of systems being targeted varies greatly in terms of the hardware and software supported, as well as, how the transactions are handled with the payment processor. Presumably, the host-to-guest communication will not differ greatly between other environments (e.g., ICS). If the printers have demonstrable weaknesses with an Ubuntu host, that will fulfill the testing requirements.

The goal of this research is to further establish academic works in regards to embedded printer devices testing and security. This area is loosely documented within academia and only mentioned vaguely in relation to statistical reports or applied research using entirely different environments. For instance, most researchers limit their analysis of the environment to smartphones and the corresponding payment app, or detection systems for card skimmers [7]. Through this research we hope to apply gainful conclusions towards the development of an embedded environment for vulnerability assessment, penetration testing, and hardware-to-software interoperability against device hosts. Some examples of how the research could be applied in the future vary: BadUSB/BashBunny [12], JuiceShop [13], DVWA [14], or Webgoat [15]; no such work exists for embedded systems within the point-of-sale or serial printer context.

1.4 Research Questions

The research questions that this proposal seeks to answer are as follows:

- Q1: Can the hardware be reflashed with a modified firmware image (e.g., FreeRTOS, ReconOS, VxWorks)? Testing a version of the original firmware with additional libraries, or an alternative OS, allows us to see if supply chain attacks are a concern. Either by the manufacturer, supplier, or other party.

- Q2: Does the base OS have bandwidth to support HID functionality? In other words, can we maintain operation of standard printer command interpretation and side-channel input attacks without causing crashes or delays? The viability of the attack depends on it going unnoticed by operators or technicians.

- Q3: Besides HID cloning, what other threat areas are exposed (e.g., network stack, web management portal, memory protections)? Are there any identifiable or known exploits when accessing the configuration panel (e.g., HTTP/2)? These provide a non-invasive method for bootstrapping the device.

Each of these goals will be approached individually as prescribed by the methodology.

Related Works

2.1 RTOS: Software and Security

[16] introduces several embedded kernels and discusses their differences in regard to developing a secure mass storage device. For this research, we are primarily interested in RTOS-like kernels because of existing support for a sample device like the SNBC BTP-S80 printer. However, the paper criticizes such operating systems because their "real-time driven design is barely compatible with the overhead produced by security mechanisms." For many applications, there is a trade off with RTOS where performance is the main criteria and security is not a priority. [17] introduces several common RTOS and discusses their security issues. Notably, most RTOS are susceptible to code injection, cryptography inefficiency, unprotected shared memory, priority inversion, denial of service attacks, privilege escalation, and inter-process communication vulnerabilities. Depending on the MPU (microprocessor unit), the vendor has hardware protections like Intel SGX or Arm Trust Zone. These are all areas that can be used for pivoting onto the device, especially shared memory and privilege escalation. If the target device firmware is outdated (or, even libraries used by the firmware) and there are known CVEs that can be repeatedly exploited, persistence mechanisms are not a requirement to gain routine access.

2.2 PoS Attack Patterns

Typically, when discussing attack patterns for PoS systems they are limited to card skimming, fake payment processor requests, or EMV cloning. In rarer cases, they might deliver malware to perform memory scraping within the PoS system or attempt swapping hardware while employees are distracted. None of these attacks include thermal printers at any point during their attack chain or delivery.

Easily the most common and well known type of attack is card skimming. Attackers will place these devices directly on top of the existing equipment to skim, or gather, credit card information at the time of purchases. They can be incredibly difficult to identify because of the sleek and stealthy designs that fraudsters use. But there is plenty of research being presented on how to quickly detect these devices [7], [18].

Without going into too much technical detail, card skimming attacks are accomplished by reading the signals emitted when swiping a magnetic card or by using an NFC reader in proximity to the payment terminal. When the customer goes to pay and uses their card, the nearby skimmer will record the transaction data being transferred. NFC skimmers, however, are not limited to being used near the terminals. Skimmer capabilities vary, and in some cases they have cameras as well or keypads for capturing PIN and zip code data.

In response to the susceptibility of magnetic cards, EMV cards were created. They are able to avoid the issues that magnetic cards and NFC share by using a chip to securely exchange transaction data with the payment terminal using secret authentication codes. The idea is that these codes cannot be tampered with or easily cloned. Despite these security advancements, EMV cards are susceptible to pre-play attacks targeting the "unpredictable number" (UN) algorithm used by ATMs [19].

[11]

Social engineers use payment processor mobile applications to directly target their victims instead of using elaborate and technical attacks against servers or user equipment [20]. The attackers simply send payment requests disguised as payments using their preferred platform. Unwittingly, the victim will accept the request thinking they were receiving money instead.

In some cases, the fraudster sends the victim money but requests a refund shortly after. As a result, the victim is either charged fees for processing the transactions or they have already spent the refunded money. These attacks are much simpler in-terms of delivery compared to the others and the intended outcomes are different. There are instances where the user device is compromised by malware specifically for exfiltrating banking data or similar PCI, but further discourse is outside the scope of the proposed research [21].

Researchers at Stony Brook University [22], demonstrated a successful introspection-based memory scraping attack against nine commercial PoS applications. Within their environment, it is assumed that the given VM (i.e., Dom0) within the shared virtualization platform (i.e., Xen) is compromised and it has escaped the guest environment. Because the privileges associated with the first VM, it has read access to the others and can perform out-of-VM memory scraping. This exact attack is likely limited to the platform used for the experiment, Xen Hypervisor; attempting something similarly against VMWare, Virtualbox, or QEMU would require further experimentation due to architectural differences. Also, PCI-DSS and PA-DSS requirements were not an obstacle for this attack since the data is not stored to disk and it is read from memory instead.

2.3 BadUSB-like Devices

BadUSB is a well-known and documented attack vector. One of the most popular hacker tools is built-on the concept [12]. However, there are some limitations:

- Precision of attacks is limited since scripts or effects are typically deployed blind. There is no knowledge of the user environment nor ability to interact with functional user interface mechanisms (e.g., a mouse clicking a button).
- Limited to the USB 2.0 standard. Meaning, no support for video adapters like HDMI, DisplayPort, or PowerDelivery like with USB 3.0.
- There are existing methods for limiting USB access from the host, such as GoodUSB [23].

GoodUSB supports the Linux USB stack, so another solution would be required for

Windows systems or RTOS. This all depends on the environment of the connected host, the PoS system. It is entirely possible that the PoS could have software like Crowdstrike Falcon deployed, which would monitor system behavior and mass storage device access [24]. Although the experiment environment will not use such software, it is an important distinction to make.

In [25], they describe several attacks at each of the applicable layers to USB attacks: the human, application, transport, and physical layers. These attacks would typically require some human element for deployment, but that is not the focus of the research (e.g., social engineering versus hardware hacking). Whereas the physical layer could allow signal eavesdropping or injection. This could enable a modified printer to overvolt the host (USBKiller [26]) to cause physical damage or perform other side-channel attacks [27]. Either of those methods would require investigating the device hardware to determine what level of control the bootloader or operating system has over power delivery.

2.4 Summary

As demonstrated by the previous works, vulnerability assessment of an embedded device is a well documented process. However, the extent that a serial thermal printer (e.g., Figure 1.1) can be maliciously expanded through a modified FreeRTOS image, while supporting original functionality, has not. And, given success in the assessment, it could suggest room for continual and improved research.

References

[1] J. Ondrus and K. Lyytinen, "Mobile Payments Market: Towards Another Clash of the Titans?" In *2011 10th International Conference on Mobile Business*, Jun. 2011, pp. 166–172. DOI: `10.1109/ICMB.2011.41`. (visited on 10/23/2023).

[2] S. T. Ebimobowei, Z. Enebraye Peter, and Y. Pual, "THE ROLE OF SOFTWARE IN A CASHLESS ECONOMY (CASE STUDY NIGERIA)," *International Journal of Research -GRANTHAALAYAH*, vol. 6, no. 1, pp. 177–186, Jan. 2018, ISSN: 2350-0530, 2394-3629. DOI: `10.29121/granthaalayah.v6.i1.2018.1607`. (visited on 10/23/2023).

[3] *SNBC BTP-S80 Thermal Printer - Black Cabinet (USB/Serial/Ethernet)*, https://www.crs-usa.com/products/snbc-btp-s80-thermal-printer-black-(usb-serial-ethernet). (visited on 10/23/2023).

[4] *SNBC New Beiyang-Intelligent Micro-Super, Smart Express Cabinet, Barcode Label Printer, Ticket Printer_Electronics_Receipt/Log Printer, Barcode/Label Printer, Special Scanning Products, Mixed Print Scanning Products, Smart Express Cabinet, Smart Micro-Super_New Beiyang specializes in the development, production, sales and service of intelligent print identification and system integration products. It provides leading products and complete, one-stop application solutions for various industries around the world. It is the only core design in the industry in the country through independent innovation. Manufacturing technology and forming a large-scale production enterprise.* https://www.snbc.com.cn/. (visited on 10/23/2023).

[5] *Cortex-M4*, https://developer.arm.com/Processors/Cortex-M4. (visited on 10/23/2023).

[6] *FreeRTOS - Market leading RTOS (Real Time Operating System) for embedded systems with Internet of Things extensions*, https://www.freertos.org/index.html. (visited on 10/23/2023).

[7] N. Scaife, C. Peeters, and P. Traynor, "Fear the Reaper: Characterization and Fast Detection of Card Skimmers," in *27th USENIX Security Symposium (USENIX Security 18)*, 2018, pp. 1–14, ISBN: 978-1-939133-04-5. (visited on 10/23/2023).

[8] *Consumer Sentinel Network Data Book for January - December 2011*, https://www.ftc.gov/reports sentinel-network-data-book-january-december-2011, Oct. 2023. (visited on 10/23/2023).

[9] C. FortheSentinel, "Consumer Sentinel Network Data Book 2022," 2022.

[10] E. Karystinos, A. Andreatos, and C. Douligeris, “Spyduino: Arduino as a HID Exploiting the BadUSB Vulnerability,” in *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, May 2019, pp. 279–283. DOI: 10.1109/DCOSS.2019.00066.

[11] Y. Wang, C. Hahn, and K. Sutrave, “Mobile payment security, threats, and challenges,” in *2016 Second International Conference on Mobile and Secure Services (MobiSecServ)*, Feb. 2016, pp. 1–5. DOI: 10.1109/MOBISECSERV.2016.7440226.

[12] Hak5, *Bash Bunny*, https://shop.hak5.org/products/bash-bunny. (visited on 10/23/2023).

[13] *OWASP Juice Shop — OWASP Foundation*, https://owasp.org/www-project-juice-shop/. (visited on 10/23/2023).

[14] R. Wood, *DAMN VULNERABLE WEB APPLICATION*, Oct. 2023. (visited on 10/23/2023).

[15] *OWASP WebGoat — OWASP Foundation*, https://owasp.org/www-project-webgoat/. (visited on 10/23/2023).

[16] R. Benadjila, M. Renard, P. Trebuchet, P. Thierry, and A. Michelizza, “Wookey : Usb devices strike back,” 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:199552896.

[17] W. D. Yu, D. Baheti, and J. Wai, “Real-Time Operating System Security,”

[18] N. Scaife, J. Bowers, C. Peeters, *et al.*, “Kiss from a Rogue: Evaluating Detectability of Pay-at-the-Pump Card Skimmers,” in *2019 IEEE Symposium on Security and Privacy (SP)*, May 2019, pp. 1000–1014. DOI: 10.1109/SP.2019.00077. (visited on 02/13/2024).

[19] M. Bond, O. Choudary, S. J. Murdoch, S. Skorobogatov, and R. Anderson, “Chip and Skim: Cloning EMV Cards with the Pre-play Attack,” in *2014 IEEE Symposium on Security and Privacy*, San Jose, CA: IEEE, May 2014, pp. 49–64, ISBN: 978-1-4799-4686-0. DOI: 10.1109/SP.2014.11. (visited on 02/13/2024).

[20] D.-G. Beju and C.-M. Făt, “Frauds in Banking System: Frauds with Cards and Their Associated Services,” in *Economic and Financial Crime, Sustainability and Good Governance*, ser. Contributions to Finance and Accounting, M. V. Achim, Ed., Cham: Springer International Publishing, 2023, pp. 31–52, ISBN: 978-3-031-34082-6. DOI: 10.1007/978-3-031-34082-6_2. (visited on 02/13/2024).

[21] H. Darvish and M. Husain, “Security Analysis of Mobile Money Applications on Android,” in *2018 IEEE International Conference on Big Data (Big Data)*, Seattle,

WA, USA: IEEE, Dec. 2018, pp. 3072–3078, ISBN: 978-1-5386-5035-6. DOI: 10.1109/BigData.2018.8622115. (visited on 02/13/2024).

[22] J. Hizver and T.-c. Chiueh, "An Introspection-Based Memory Scraper Attack against Virtualized Point of Sale Systems," in *Financial Cryptography and Data Security*, D. Hutchison, T. Kanade, J. Kittler, *et al.*, Eds., vol. 7126, Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 55–69, ISBN: 978-3-642-29888-2 978-3-642-29889-9. DOI: 10.1007/978-3-642-29889-9_6. (visited on 02/13/2024).

[23] D. J. Tian, A. Bates, and K. Butler, "Defending Against Malicious USB Firmware with GoodUSB," in *Proceedings of the 31st Annual Computer Security Applications Conference*, ser. ACSAC '15, New York, NY, USA: Association for Computing Machinery, Dec. 2015, pp. 261–270, ISBN: 978-1-4503-3682-6. DOI: 10.1145/2818000.2818040. (visited on 10/24/2023).

[24] J. Backer, "Sdn-controlled isolation orchestration to support end-user autonomy," Ph.D. dissertation, WORCESTER POLYTECHNIC INSTITUTE, 2021.

[25] J. Tian, N. Scaife, D. Kumar, M. Bailey, A. Bates, and K. Butler, "SoK: "Plug & Pray" Today – Understanding USB Insecurity in Versions 1 Through C," in *2018 IEEE Symposium on Security and Privacy (SP)*, May 2018, pp. 1032–1047. DOI: 10.1109/SP.2018.00037. (visited on 11/05/2023).

[26] *USB Kill devices for pentesting & law-enforcement*, https://usbkill.com/. (visited on 11/05/2023).

[27] K. Sridhar, S. Prasad, L. Punitha, and S. Karunakaran, "EMI issues of universal serial bus and solutions," in *8th International Conference on Electromagnetic Interference and Compatibility*, Dec. 2003, pp. 97–100. DOI: 10.1109/ICEMIC.2003.237887. (visited on 11/05/2023).