PRINTSHOP: ASSESSING OS AND CAPABILITIES OF SERIAL PRINT DEVICES

1st Micah Flack
The Beacom College of Computer and Cyber Sciences
Dakota State University
Idaho Falls, USA
micah.flack@trojans.dsu.edu

***Abstract*—(OUTLINE FILLER WORDS) in aliquam sem fringilla ut morbi tincidunt augue interdum velit euismod in pellentesque massa placerat duis ultricies lacus sed turpis tincidunt id aliquet risus feugiat in ante metus dictum at tempor commodo ullamcorper a lacus vestibulum sed arcu non odio euismod lacinia at quis risus sed vulputate odio ut enim blandit volutpat maecenas volutpat blandit aliquam etiam erat velit scelerisque in dictum non consectetur a erat nam at lectus urna duis convallis convallis tellus id interdum velit laoreet id donec ultrices tincidunt arcu non sodales neque sodales ut etiam sit amet nisl purus in mollis nunc sed id semper risus in hendrerit gravida rutrum quisque non tellus orci ac auctor augue mauris augue neque gravida in fermentum et sollicitudin ac orci phasellus egestas tellus rutrum tellus pellentesque eu tincidunt tortor aliquam nulla facilisi cras fermentum odio eu feugiat pretium nibh ipsum consequat nisl vel pretium lectus quam id leo in vitae turpis massa sed elementum tempus egestas sed sed risus pretium quam vulputate dignissim suspendisse in est ante in nibh mauris cursus mattis molestie a iaculis at erat pellentesque adipiscing commodo elit at imperdiet dui accumsan sit amet nulla facilisi morbi tempus iaculis urna id volutpat lacus laoreet non curabitur (OUTLINE FILLER WORDS).**

***Index Terms*—serial devices, thermal printer, PoS, ICS, badusb, hardware hacking, embedded devices.**

I. Introduction

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 [1], [2]. 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.

This research primarily focuses on physical point-of-sale (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 [3]. 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.

Serial devices pose a significant threat to the PoS security landscape, as well as, industrial control systems due to the limited security features of the devices. Because these devices implement no host monitoring, minimal hardware protections, dubious component suppliers, and unchecked communication with their host. Furthermore, the origin of the manufacturer, supplied components, firmware, and supporting libraries is a separate area of research; which, could provide further scrutiny into where these devices are deployed and in what environments.

A. Research Objectives

Some filler words.

II. Related Works

A. RTOS: Software and Security

[3] 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. [4] 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.

B. Embedded Firmware Patching

Typically, updating the firmware for a device or even delivering patches requires a complete shutdown and hardware debug access (if supported). In some cases, the reflashing is unsupported through the operating system or bootloader and the flash memory needs to be reprogrammed. [5] describes a method for hotpatching downstream RTOS devices without needing to shutdown or reboot. Any changes made are permanent and as effective as traditional delivery methods. RapidPatch was capable of patching over 90% of vulnerabilities for the affected device, only needing at least 64KB or more memory and 64 MHz MCU clock. This appears to be an effective method for attackers to sideload client or server implants without risking detection.

C. BadUSB-like Devices

BadUSB is a well-known and documented attack vector. One of the most popular hacker tools is built-on the concept [6]. 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 [7].

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 [8]. Although the experiment environment will not use such software, it is an important distinction to make.

In [9], 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 [10]) to cause physical damage or perform other side-channel attacks [11]. 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.

Methodology

in aliquam sem fringilla ut morbi tincidunt augue interdum velit euismod in pellentesque massa placerat duis ultricies lacus sed turpis tincidunt id aliquet risus feugiat in ante metus dictum at tempor commodo ullamcorper a lacus vestibulum sed arcu non odio euismod lacinia at quis risus sed vulputate odio ut enim blandit volutpat maecenas volutpat blandit aliquam etiam erat velit scelerisque in dictum non consectetur a erat nam at lectus urna duis convallis convallis tellus id interdum velit laoreet id donec ultrices tincidunt arcu non sodales neque sodales ut etiam sit amet nisl purus in mollis nunc sed id semper risus in hendrerit gravida rutrum quisque non tellus orci ac auctor augue mauris augue neque gravida in fermentum et sollicitudin ac orci phasellus egestas tellus rutrum tellus pellentesque eu tincidunt tortor aliquam nulla facilisi cras fermentum odio eu feugiat pretium nibh ipsum consequat nisl vel pretium lectus quam id leo in vitae turpis massa sed elementum tempus egestas sed sed risus pretium quam vulputate dignissim suspendisse in est ante in nibh mauris cursus mattis molestie a iaculis at erat pellentesque adipiscing commodo elit at imperdiet dui accumsan sit amet nulla facilisi morbi tempus iaculis urna id volutpat lacus laoreet non curabitur

Results

in aliquam sem fringilla ut morbi tincidunt augue interdum velit euismod in pellentesque massa placerat duis ultricies lacus sed turpis tincidunt id aliquet risus feugiat in ante metus dictum at tempor commodo ullamcorper a lacus vestibulum sed arcu non odio euismod lacinia at quis risus sed vulputate odio ut enim blandit volutpat maecenas volutpat blandit aliquam etiam erat velit scelerisque in dictum non consectetur a erat nam at lectus urna duis convallis convallis tellus id interdum velit laoreet id donec ultrices tincidunt arcu non sodales neque sodales ut etiam sit amet nisl purus in mollis nunc sed id semper risus in hendrerit gravida rutrum quisque non tellus orci ac auctor augue mauris augue neque gravida in fermentum et sollicitudin ac orci phasellus egestas tellus rutrum tellus pellentesque eu tincidunt tortor aliquam nulla facilisi cras fermentum odio eu feugiat pretium nibh ipsum consequat nisl vel pretium lectus quam id leo in vitae turpis massa sed elementum tempus egestas sed sed risus pretium quam vulputate dignissim suspendisse in est ante in nibh mauris cursus mattis molestie a iaculis at erat pellentesque adipiscing commodo elit at imperdiet dui accumsan sit amet nulla facilisi morbi tempus iaculis urna id volutpat lacus laoreet non curabitur

Conclusion

in aliquam sem fringilla ut morbi tincidunt augue interdum velit euismod in pellentesque massa placerat duis ultricies lacus sed turpis tincidunt id aliquet risus feugiat in ante metus dictum at tempor commodo ullamcorper a lacus vestibulum sed arcu non odio euismod lacinia at quis risus sed vulputate odio ut enim blandit volutpat maecenas volutpat blandit aliquam etiam erat velit scelerisque in dictum non consectetur a erat nam at lectus urna duis convallis convallis tellus id interdum velit laoreet id donec ultrices tincidunt arcu non sodales neque sodales ut etiam sit amet nisl purus in mollis nunc sed id semper risus in hendrerit gravida rutrum quisque non tellus orci ac auctor augue mauris augue neque gravida in fermentum et sollicitudin ac orci phasellus egestas tellus rutrum tellus pellentesque eu tincidunt tortor aliquam nulla facilisi cras fermentum odio eu feugiat pretium nibh ipsum consequat nisl vel pretium lectus quam id leo in vitae turpis massa sed elementum tempus egestas sed sed risus pretium quam vulputate dignissim suspendisse in est ante in nibh mauris cursus mattis molestie a iaculis at erat pellentesque adipiscing commodo elit at imperdiet dui accumsan sit amet nulla facilisi morbi tempus iaculis urna id volutpat lacus laoreet non curabitur

References

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

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

[3] 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.

[4] W. D. Yu, D. Baheti, and J. Wai, "Real-Time Operating System Security,"

[5] Y. He, Z. Zou, K. Sun, *et al.*, "{RapidPatch}: Firmware Hotpatching for {Real-Time} Embedded Devices," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 2225–2242, ISBN: 978-1-939133-31-1. (visited on 10/24/2023).

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

[7] 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).

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

[9] 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).

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

[11] 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).