The Impact of Current Software Vulnerabilities in Large-Scale Networks

Sean Carroll

Villanova University

"This paper or presentation is my own work. Any assistance I received in its preparation is acknowledged within the paper or presentation, in accordance with academic practice. If I used data, ideas, words, diagrams, pictures, or other information from any source, I have cited the sources fully and completely in footnotes and bibliography entries. This includes sources that I have quoted or paraphrased. Furthermore, I certify that this paper or presentation was prepared by me specifically for this class and has not been submitted, in whole or in part, to any other class in this University or elsewhere, or used for any purpose other than satisfying the requirements of this class, except that I am allowed to submit the paper or presentation to a professional publication, peer reviewed journal, or professional conference. In adding my name following the word 'Signature', I intend that this certification will have the same authority and authenticity as a document executed with my hand-written signature.
Signature ___Sean Carroll_______"

Abstract

This paper explores current software vulnerabilities in our national infrastructure. The articles primarily focus on software vulnerabilities in cellular telecommunication systems and information systems involving the internet. The evolution of 4G-LTE and mobile device usage are of great concern in user privacy and mass manipulation of infrastructure in user equipment. The vulnerabilities in software defined networking (SDN) and the internet of things (IoT) are also discussed in the interest of protecting against future malicious software attacks, such as recent WannaCry ransomware and the commonly used distributed denial of service (DDOS) attacks that plague the internet. Based on these reports, this paper highlights potential implications of such security flaws on their specific environments and potential mitigation techniques to prevent future exploitation. In conclusion, master-controlled frameworks need to be open-sourced and updated in order to protect against mass attacks on user data.

The Impact of Current Software Vulnerabilities in Large-Scale Networks

Large-scale networks, such as cellular communication systems and its internet integration, are progressively growing and expanding access to information in the world around us. As it becomes more and more beneficial to use such systems in our daily lives, the increasing reliance on such growing infrastructure also creates and environment filled with growing risk of manipulation and exploitation. The purpose of this paper is to identify current software security vulnerabilities in infrastructure and to discuss the risk they pose to their affected environments upon exploitation. This paper will conclude by suggesting preventable measures that can be taken to mitigate risk caused by these software vulnerabilities.

Literature Review

As telecommunication companies adopt long-term evolution in its fourth generation of cellular network standards, current research suggests that serious security vulnerabilities exist in such specifications. This is contrary to the general belief that LTE specification will allow for increased privacy and coverage to mobile platforms. Asokan, Borgaonkar, Niemi, Shaik, and Seifertv (2017) have recently demonstrated attacks on commercial 4G-LTE technologies that allowed the them to monitor the location of devices more accurately than originally thought. The attack conducted in the publication manipulated a very similar authentication vulnerability in the third-generation standard. It was originally thought to have been prevented using the new standard. Asokan et al.'s (2017) paper also identifies unique denial of service (DoS) attacks that use low-cost hardware and open-source software to deny services to user equipment across an area of normal basestation coverage. It is important to realize, given these findings, that second generation and third generation technologies are available for use if fourth generation technology is not accessible. However, the ability to manipulate flaws in fourth generation technologies eliminates the task downgrading devices in exploitation.

It is not only the transmission of data over cellular networks that is of concern, but the applications on the host devices that are transmitting this data. Watanabe’s (2017) paper discusses third-party library use on mobile devices. It accounts these libraries for seventy percent of vulnerabilities in paid mobile application, with Ironsource, Conduit App, Apache Cordova, and Paypal mentioned as having vulnerabilities. The use of third-party application in development should be avoided if possible, however, with large-scale applications it is often unrealistic. Watanabe’s (2017) paper suggested that mobile applications can take advantage of others using inter-component communication (ICC) and Android’s WebView class. A common attack used by hackers on many different mobile development platforms, JavaScript injections through web classes enable unauthorized access to information within the mobile applications. This type of ICC exploitation exposes the hardware and software implementations that are frequently connected through cellular communications network using 3G and 4G standards.

The administration of mass control in corporate environments, as in the introduced software defined networking model, are not excluded from these vulnerabilities. Mostovich’s (2017) paper discusses the high-level vulnerabilities with this model in the context of telecommunication network evolution. The SDN paradigm has been introduced as a secure architecture to control telecommunication networks. However, Mostovich identifies and confirms threats within the control and data layers of the SDN concept. These high-level vulnerabilities do not include the specifications of the subsystem architecture. Software-defined security (SDsec) subsystems are a needed form of encapsulation within SDN, according to Mostovich, with special attention to software-defined internet of things (SDIoT). His paper highlights threats within software development which appear as a recurrent problem. Unauthorized access, package transmission leaks, unreliability of keys within programmed data planes, man-in-the-middle (MITM) attacks, denial of service, and misconfiguration of security policies are identified within his paper, however, they are widespread issues that are not unique to the SDN paradigm. The SDN paradigm enables controllers to have complete access to the forwarding plane through the OpenFlow base protocol. Version 1.3.0 of the OpenFlow standard enables the optional absence of TLS protocol by configuration. A TLS absence within the control channel between controller and switch exposes a threat where SwitchFlow tables cannot verify the expected rule configuration (Mostovich, 2017). The packet loss concept, although optimizing network congestion through its best-effort delivery service, exposes numerous threats in the data layer between communication processes that intelligent communications do not expose. Mostovich's TLS absence threat identified, which succumbs to threat by misconfiguration and lack of required verification, is accompanied by numerous other threats under similar miscommunication issues in the packet loss concept. Mostovich outlines a concept in which hackers with access to a physical or virtual network can exploit interrupted or misconfigured channels—that are not immediately detected by the controller—leading not only to data loss, but a possibility for attackers to block a channel or node and cause entire network instability. This paper further realizes entire network instability when multiple destructive commands are sent to OpenFlow compatible devices.

In the consideration of secure development lifecycles, switching to a framework of SDN everywhere raises numerous concerns for subsystem activity and design. Mostovich explains in the fifth section of his paper that "Lack of third-party applications and OS control, no control for insecure APIs and no prevention from misconfigurations are directly inherited from earlier concept model of telecom networks.” Regardless of the attribution to these threats, a solution to SDsec with particular consideration the progression of SDIoT requires research beyond Mostovich’s publication alone. The most recent publication in this area of concern is Simpson’s article “Securing Vulnerable Home IoT Devices with an In-Hub Security Manager.” The expanding ecosystem of consumer devices connected to the internet poses significant vulnerabilities in the subsystem and the system entire. The system vulnerabilities cannot be ignored, however, securing subsystems from the bottom-up, especially in the case of connected IoT devices, does provide mitigation of attack to these systems. In this regard, Simpson proposes a central security manager that is built above gateway routers or device hubs to patch or mitigate vulnerabilities. The security manager proposed is positioned to intercept and monitor device traffic. By way of encapsulation, the security manager would offer software updates, detection of traffic from exploited devoices, and strengthen authentication. Through monitoring and filtering device traffic in such overarching fashion, Simpsons estimates that fifty-percent of common vulnerabilities and exposures (CVEs) could be mitigated. Within the scope of threat model, DDoS attacks and malware attacks are of serious concern. Berger (2017) details the ransomware attack on various organizations, in which a weakness in Microsoft’s server message block was manipulated in order to acquire slave devices. The extensibility of such ransomware is of great concern in a IoT environment. A complex ransomware attack has the potential to enslave an entire IoT device network, which necessitates the monitoring of network ports and the traffic of services using them. Although Berger’s article demonstrated a different medium in which the WannaCry ransomware chose to attack—computers themselves as opposed to other common consumer devices—the sever message block protocol is a file-sharing protocol that assumes connectivity with common IoT devices, including printers and bluetooth-connected devices nonetheless. Simpson's proposed central security manager concept proposes a method to rapidly patch and deter threats to alleviate potential security threats. By referencing identified CVEs from a trusted third party, Simpsons' proposed manager could impose a rate-limiting effect on certain devices given their activity and credentials, thereby addressing mitigation and prevention of realized ransomware, DDoS, and cross-site request forgery (XSRF) attacks.

Discussion

Simpson's encapsulation concept provides a possible means of increasing security at various points of the secure software, hardware, and vulnerability lifecycle. By creating a system of monitoring traffic, there creates a solution to backwards compatibility of legacy software and devices. Legacy hardware and software can be streamlined in a linear fashion—as opposed to uprooting legacy systems for the sake of security—in order to update current systems for the evolution of technologies.

It is important to identify that in the majority of developed countries the long-term evolution has not begun, and we are in a state that is often termed "3.9G" or "Pre-4G." Due to the marketing of wireless telecommunication companies, such 4G-LTE can be misinterpreted as meeting the exact standards for 4G-LTE specifications. The current telecom infrastructure necessitates backwards compatibility for safety concerns, which is a known trade-off with regards to security practices. The amount of legacy devices and hardware present in large-scale systems appears to be preventing the implementation of such standards. The regulations on radio frequencies put in forth by the Federal Communication Commission (FCC), large amounts of physical infrastructure, concerns for backwards compatibility, interference issues between various protocols, amongst other concerns make evolution of such systems very difficult in practice. Previous paradigms used to define the operation of cellular networks cannot be changed and thus patching and building upon current infrastructure appears to be the only solution. The implementation of small cell devices in order to reach consumer demands in high-usage areas represents this struggle. With such implementation, as outlined in Asokan et al.'s (2017) paper, there exists clear security concerns. MITM and DoS attacks on such cellular networks, combined with mobile devices, SDN, and SDIoT create a culmination of vulnerabilities that can be linked together to cause signification harm or loss.

Wantabe's (2017) paper rooted the origin of mobile app vulnerabilities in the use of third-party libraries. Within the current mobile application development environments, the prevention of third-party library usage is simply infeasible. The third-party dependencies used in the vast majority of commercial mobile applications, whether open-source or private libraries that the creators pay for their use, are needed for core functionality alone. Wantabe's research notes Android's WebView class as a potential for vulnerability because of its JavaScript functions, but from a programmer's perspective, the library's use is inevitable to most core functionality. The use of JavaScript embedded within another language such as Objective-C, C, or Java does expose security risks upon execution due to potential for escalation of privilege. However, attributing a certain class does not identify the issue to be addressed. The WebView class has clear potential to enable XSRF and ICC, however, it is not the class itself that needs to be addressed. Prevention and mitigation of injectable code—and that code escalating privileges to conduct unauthorized communications—is the direct issue of concern. Implementation of secure software practices and educating programmers when they are implementing potentially vulnerable, albeit necessary, classes or third-party libraries would potentially address these vulnerabilities. Simply abstaining from using third-party libraries inherently inhibits growth in the mobile software development process.

Conclusion

The progression towards new and secure technologies in large-scale networks is often prevented by legacy infrastructure and the necessity of backward compatibility of such devices. Telecommunication networks and the ongoing progression of information systems on the internet fall victim to this large-scale paradigm. Tradeoffs between security and other requirements is a necessity to the progression of large-scale implementations. An ideal standardization has and may not be proposed by academia, for the point of optimal balance between requirements is in constant flux. With standardization in mind, the encapsulation concept that Simpson proposes in context to SDIoT systems could be extrapolated to large-scale implementations. Implementing an open-source central security manager to quickly update and monitor traffic going through such networks proposes a possible solution to this dilemma. This model—whereby rapid patching of known vulnerable devices and intervening upon various points of the vulnerability lifecycle is performed—provides a simplified and security-focused methodology to the progression of large-scale networks. The secure central management system shows a viable direction in small-scale analysis. It is the desire that this paper motivates ongoing research into the secure advancement of large-scale systems.

References

Asokan, N., Borgaonkar, R., Niemi, V., Shaik, A., & Seifert, J. (2017). Practical Attacks Against Privacy and Availability in 4G/LTE Mobile Communication Systems. CoRR, abs/1510.07563.

Berger, B. (2017). *WannaCry exposes defense supply chain vulnerabilities* National Defense Industrial Association.

Mostovich, D. (2017). High-level vulnerabilities of software-defined networking in the context of telecommunication network evolution. 184-186. doi:10.1109/EIConRus.2017.7910524

Simpson, A. K. (2017). Securing vulnerable home IoT devices with an in-hub security manager. 551-556. doi:10.1109/PERCOMW.2017.7917622

Watanabe, T. (2017). Understanding the origins of mobile app vulnerabilities: A large-scale measurement study of free and paid apps. 14-24. doi:10.1109/MSR.2017.23