Building a Trusted Computing Foundation

Introduction:

In the early 1980's, the U.S. Government (largely the Department of Defense and the Intelligence Community) recognized that the only way to stop determined/sophisticated (e.g., Nation-State) hacking into computer networks, systems and applications was to establish requirements for raising the level of security and, as important, trust in our computing systems. Considerable time and energy was devoted to establishing principles, rules and, eventually, technical and programmatic requirements for building security and trust into contemporary computing systems. The result of this research and academic deliberations was the publication of the Department of Defense Trusted System Computer Evaluation Criteria (The Orange Book) in 1983. The Orange Book was followed up by an additional series of guides known as the Rainbow Series of trusted computer evaluation criteria for topics including networks, supply chain and application security.

While the Rainbow series of trusted computer evaluation guides were written (largely) by inside the beltway people, for use by inside the beltway organizations, many real systems were actually built by industry and at the higher levels of trust (i.e., B3-A1) were indeed "unhackable." While these systems mostly used proprietary firmware and operating systems (thus, why they failed in the commercial market), they succeeded in demonstrating that computing platforms can be built with high levels of security and trust and can *dramatically* raise the bar for would-be hackers, including sophisticated hackers. Their legacy lives on today in the implementation of many contemporary high-security products like SE/Linux, Trusted Solaris, the Blackberry phone and even the Apple IOS security architecture.

Why the Commission on Enhancing National Cybersecurity Needs to Know This Legacy:

By now, the commission, I hope, has learned that the key to enduring cyber security is to ensure that the computers are secure! No amount of security

software, firewalls, HTTP security headers, threat intelligence, policies, et cetera can substitute for building systems up-front with a security model and "trustable" firmware and software. This is the heart of the often-repeated statement (that most people saying it do not even understand): "Security must be baked in, not bolted on." *However, every (indeed every) technical approach to the cyber security crisis, to date, has come up short due to the failure to understand this concept.*

An example is the Advanced Persistent Threat (APT) malware risk. This risk is based on the fact that contemporary operating system kernel code does not operate in a secure/trusted/consistent manner and is vulnerable to memory manipulation, like Return Oriented Programming (ROP) attacks. This fundamental principle was recognized as far back as 1966 with the introduction of the Multics operating system. Multics used a multiple security ring architecture that ensured that only trusted processes executed in the security kernel portions of memory. This feature was last implemented in IBM's OS/2 operating system (1987) but has been largely forgotten since then. Multics was not vulnerable to the common buffer overflow and more sophisticated ROP attacks. Buffer overflows, despite band-aid measures like Microsoft's Enhanced Mitigation Experience Toolkit, is still the major avenue for more sophisticated attacks like ROP. Note the recent attack on the Democratic National Committee network or, more worrisome, attacks against Internet-Of-Things (IOT) operating systems.

<u>Trusted Computing Foundation – A Promising Approach to the Cybersecurity Challenge</u>:

The same organization hosting this commission recently published NIST Special Publication 800-160: Systems Security Engineering:*Consideration for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems.* This document represents a watershed opportunity for the government (and industry) to establish principles for a trusted computing foundation. These principles should be used to articulate detailed system/software design criteria (listed later in this memo). The criteria can eventually become the basis for federal standards for acquisition and operation of networks, systems and applications used by the national critical infrastructure, especially IOT devices. Similar to how the government used their regulatory function to build generations of increasingly safer automobiles, a true partnership with industry and academia can begin with NIST 800-160 and result in an implementation plan to integrate

trusted firmware and software into existing and planned IT products and services.

This concept is actually much more achievable than many may consider at first-blush. Unlike the authors of the Orange Book, we now have decades of literature and experiences that clearly demonstrate how computer systems are penetrated and, accordingly, where and how to build trusted firmware and software. Furthermore, we actually have elements (incomplete though they are) of trusted code security modules and libraries available for use in building secure networks, systems and applications.

An example is Intel Corporation's Trusted Execution Technology (Intel TXT). Intel TXT is specific hardware/firmware designed to “harden” platforms from the emerging threats of hypervisor attacks, BIOS, or other firmware attacks, malicious root kit installations (e.g., APTs) and other software based attacks. It is used, for example, by a company named Bromium to provide full user session isolation on a Windows desktop. There is only one known successful attack against Bromium's implementation of Intel TXT and that was an early (Beta) version. Other companies including AMD and even Microsoft are now introducing capabilities in their products (e.g., Microsoft's Windows 10 Credential Guard) that exploit the capabilities of the Intel TXT capability. Of course, these are all individual efforts that are not founded in any commonly accepted (e.g., National) set of principles or design specification languages and are truly only pieces of a larger puzzle. However, this is considerably more than what the authors of the Orange Book had in 1983 and now if combined with NIST 800-160 and decades of empirical experiences, this challenge becomes entirely “solvable.”

<u>A Trusted Computing Foundation Roadmap for Formal Research and Development</u>:

It is recommended that this commission recommend a formal research and development program that brings together government, academia and industry professionals under the sponsorship of NIST to establish design and evaluation principles/criteria for building future generations of increasingly more secure and trusted critical infrastructure computer networks, systems and applications. This effort should also consider participation from
The Common Criteria for Information Technology Security Evaluation, an international body that establishes computer product security evaluation criteria. As noted, the principles and criteria can become a foundation for U.S. Government acquisition standards (beyond the critical infrastructure) and, eventually, be

adopted by private industry (similar to the NIST Cyber Security Framework). Beginning with NIST 800-160, a national trusted computing framework research and development effort includes establishing:

1. A common cyber security risk/threat model
2. Principles/criteria for designing/developing trusted computer firmware and software
3. A trusted firmware, operating system and application functional model (e.g., ring architecture)
4. Principles/criteria for trusted operating system functions and features
5. Principles/criteria for attestation of firmware and operating system trust
6. Principles/criteria for low-level and high-level (application) trusted programming languages
7. Principles/criteria for trusted user/device/process identification, access authorization, and event logging
8. Principles/criteria for data protection (e.g., encryption)
9. Principles/criteria for trusted supply chain firmware and software support
10. Principles/criteria for enabling trusted application programming interfaces

Next Steps:

Establishing the aforementioned principles and criteria requires a true partnership between the government, academia and, of course, industry. Given the threat to our critical infrastructure (not to mention our economic base), the government needs to move quickly and become the champion for this effort. The government should use the auspices of NIST to bring together all interested parties (similar to this commission) to begin the dialogue on building a trusted computing foundation. The result of this dialogue should be a federally funded research and development effort that challenges academia and industry to propose solutions to the ten principles/criteria listed above. The most promising solution sets should be funded with a focus on either modifying existing open-source operating systems or even developing new operating systems. The government may even want to consider a contest, similar to the successful efforts to propose/produce a next-generation cryptographic algorithm and key management scheme.

Robert Bigman
2BSecure
Rybbigs@Gmail.com
301-922-3884