JOURNAL OF
DoD RESEARCH & ENGINEERING

Volume 2 | Issue 2 | Special Edition August 2019

(U) Toward High-Assurance Interface Protocols For Department Of Defense Applications (Op-Ed)

Dr. Evan Austin[1], 1LT Ryan Gagnon[2], Dr. Adam Shull[2], Dr. Robert Templeman[2]

[1] Naval Information Warfare Center Atlantic
P.O. Box 190022
North Charleston, SC 29419-9022

[2] Naval Surface Warfare Center, Crane Division
300 Hwy 361
Crane, IN 47522

evan.austin@navy.mil, ryan.f.gagnon.mil@mail.mil, adam.m.shull@navy.mil, robert.templeman@navy.mil

Biography

(U) Dr. Evan Austin received his BS, MS, and PhD in computer science from the University of Kansas. Since 2015, Dr. Austin has worked as a scientist at the Naval Information Warfare Center Atlantic (NIWC-LANT). He is a Principal Investigator and the acting Cybersecurity Area Lead in NIWC-LANT's Research and Applied Science Competency. In addition to Dr. Austin's research activities, he also currently serves as the Chair of NIWC-LANT's Science and Technology Advisory Council.

(U) 1LT Ryan Gagnon is a computer engineer at the Naval Surface Warfare Center, Crane Division (NSWC Crane). He is a graduate of Rensselaer Polytechnic Institute, Troy, NY with a B.S. in Computer Engineering and a commission as an Army Officer through the ROTC program. He has co-authored and published research papers in technical and policy-level domains through NSWC Crane and through the Army Cyber Institute, West Point, NY. 1LT Gagnon is currently serving as an Army Signal Officer and Knowledge Manager deployed to the Middle East in support of Operation Spartan Shield and Operation Inherent Resolve.

(U) Dr. Adam Shull is a computer scientist at NSWC Crane. He graduated from Indiana University with a Ph.D. in Informatics and an M.A. in Mathematics, and Valparaiso University with a B.S. in Mathematics and a B.A. in Computer Science. He has co-authored research papers published at the IEEE Symposium on Security and Privacy and the RSA Conference Cryptographers' Track. He currently serves as the principal investigator for the High-Assurance Interface Protocols (HAIP) project at NSWC Crane.

(U) Dr. Robert Templeman is a Senior Scientific Technical Manager at NSWC Crane. Rob serves as a Naval Sea Systems Command (NAVSEA) Distinguished Engineer for Cybersecurity. He is a graduate of Indiana University (Ph.D. in Computer Science), the Rose-Hulman Institute of Technology (M.S. in Engineering Management), and Purdue University (B.S. in Electrical Engineering). In 2016, Dr. Templeman was appointed to the Indiana Executive Council on Cybersecurity to represent Defense issues to the State of Indiana. He is a member of the Defense Acquisition Corps and served as a non-commissioned officer in the U.S. Marine Corps with one deployment to Iraq in 2003.

(U) Toward High-Assurance Interface Protocols For Department Of Defense Applications

1. (U) Introduction

(U) Whether in office environments with commodity laptops or in austere environments with operational technology that underpins warfighting capabilities, implicit in everyday interactions with computer systems are numerous standards and protocols that allow interconnected pieces of hardware and software to communicate and function as one. An increased emphasis on cybersecurity has drawn some of these protocols to light; for example, most computer literate individuals recognize the existence and purpose of the Hypertext Transfer Protocol (HTTP) and its secure extension. However, it is easy to overlook other equally ubiquitous protocols of critical importance to the security of our systems (e.g. those of the various Universal Serial Bus (USB), Ethernet, and High-Definition Multimedia Interface (HDMI) standards). With the

possible exception of the USB family, whose inherent insecurity has made the news a number of times in recent years, each of these protocols typically is associated with the mundane attachment of computer peripherals. However, interacting with these peripherals requires crossing an interface boundary, potentially subjecting the host system to security threats.

(U) The primary threats associated with interface protocols are two-fold. First, given that a number of these protocols operate at least partially in kernel space, adversaries can exploit errors in their implementation for privilege escalation and unfettered lateral movement throughout the rest of the system. Second, these protocols are commonly designed to accommodate a large and diverse set of interactions with legacy, current, and future systems alike. While well intentioned, the extensible nature of such specifications has the potential to facilitate malicious effects, even if a corresponding implementation is correct.

(U) The predominant approach to addressing vulnerabilities like these is to assume a reactive posture with a focus on expediting the cybersecurity OODA loop—vulnerabilities must be identified, assessed, patched, and a fix deployed as rapidly as possible. The dangers of this patch and pray attitude were highlighted on a national level when a group of hackers calling themselves Lopht Heavy Industries testified in front of the U.S. Senate about the insecurity of the protocols that formed the backbone of the Internet. [1] In the twenty years subsequent to this event, expeditious attempts to use software updates to address more systemic issues are still the root cause of everything from loss of availability to loss of life. [2, 3]

(U) For these and other reasons, there is a growing consensus that the DoD instead should be taking a proactive approach to cybersecurity, with an emphasis on delivering systems that are provably free of certain classes of vulnerabilities from the start. An emerging preventative and proactive approach is the idea of language-theoretic security. [4] Originally intended to address vulnerabilities due to shotgun parsers, a software anti-pattern where input validation and processing are inter-mixed, the central drive of LangSec is a reduction of untrusted inputs to formal representations that can be precisely expressed and interpreted in a verifiable way. Consequently, one can derive static guarantees of security as opposed to testing for it through ad-hoc means.

(U) A short-coming of LangSec and comparable approaches is that they can be difficult to apply to systems designed in ignorance of the underlying ideology of correctness through construction. This article expounds upon this disconnect, focusing on how current DoD policy and practice is amplifying it by promoting a stagnant culture of cybersecurity via compliance. Also presented is early research the authors are conducting with the specific intent of eliminating large classes of vulnerabilities associated with interface protocols. The motivation for sharing this information is two-fold:

- (U) To highlight a serious threat to DoD systems that the authors do not feel is being adequately addressed
- (U) To evangelize for principled design and development techniques as a mechanism for assuring the safety and security of critical systems

2. (U) Challenges in Interface Security

(U) Critical DoD systems depend on the aforementioned protocols that transverse physical interfaces, but there are scores of other logical interfaces between the software modules within these systems. Years ago, the defense community would design, document, implement, and test all of these boundary points in a rigorous fashion, resulting in well understood protocols that offered just enough functionality for their intended applications. This methodology was in line with the cybersecurity principle of least privilege – every element of a computing environment must have access to only the information and resources required to fulfill its role. Enforcement of least privilege reduces a system's attack surface by eliminating classes of vulnerabilities attributable to unintended interactions and inputs.

(U) Today, systems widely use standardized interfaces and protocols that egregiously violate this tenet. Open-source literature contains an almost un-enumerable, and ever lengthening, list of demonstrated protocol compromises that anecdotally support this point. These protocols remain vulnerable despite the use of cybersecurity methodologies like the Risk Management Framework (RMF). [5] Consider the control set applied by the RMF for critical DoD systems [6]; the following controls are included as part of the recommended baseline and are applicable to interface security:

- (U) CM-7 (least functionality): The organization configures the information system to provide only essential

capabilities and prohibits or restricts the use of selected functions, ports, protocols, and/or services.

- (U) SC-7 (boundary protection): Monitors and controls communications at the external boundary of the system and at key internal boundaries within the system
- (U) SI-10 (information input validation): Checking the valid syntax and semantics of information system inputs (e.g., character set, length, numerical range, and acceptable values) verifies that inputs match specified definitions for format and content.

(U) While the above controls can be implemented in a way that adheres to the principle of least privilege and protects a system from malicious input, in practice this is almost never the case. CM-7 often is applied at too coarse a granularity to be impactful, SC-7 is hand-waved away with appeals to unvalidated claims of employed commercial security products, requiring only that "the organization defines information inputs requiring validity checks…and (that) the information system checks the validity of organization-defined information inputs." [6]

(U) Two flawed assumptions underpin the assurances provided by the RMF. The first is the belief that security can be realized almost entirely through corrective actions (e.g., the real-time configuration management and monitoring prescribed by the above controls). The second is the belief that security can be asserted through subjective, human assessment of test and evaluation artifacts. LangSec and other comparably principled engineering practices obviates the need for such assumptions by constructing systems in a way that provides a machine-checkable trace of desired security properties from design to implementation. Continuous monitoring of the inputs and outputs of a protocol stack or assessing ancillary evaluation artifacts is not necessary if its functional behavior is both verifiably correct and self-documenting.

(U) That said, DoD systems depend on standardized protocols because of the great benefits that they offer. They promote interoperability with, and accessibility to, commercial off the shelf solutions (COTS), significantly simplifying the acquisition life cycle. Architecting systems entirely with bespoke interfaces between elements in the name of cybersecurity is untenable. Rather, it is necessary to adopt rigorous approaches to integrating the use of interfaces and protocols in a demonstrably secure manner. What follows is a brief introduction to the initial steps toward providing one such approach.

3. (U) High-Assurance Interface Protocols

(U) Acknowledging the need for improved assurance for critical defense systems, researchers from Naval Information Warfare Center Atlantic and Naval Surface Warfare Center, Crane Division, have been collaboratively investigating solutions to fill this gap using Naval Innovative Science and Engineering (10 U.S. Code § 2363) funds since 2017. Central to this research is a methodology for securing interface boundaries predicated on applying LangSec tools and techniques. To briefly expand upon the earlier introduction, LangSec entails the following:

- (U) Trustworthy software must be able to change untrusted input into a formal language.
- (U) This formal language must be parsed by a verifiable language recognizer.
- (U) Critically, the language and its parsed outputs must not be more complex or expressive than the specific application requires.

(U) Implicit in the above definition is an enforcement of the principle of least privilege, as it applies to transmitting and receiving structured data. Unlike many of the interface protocols in use today, LangSec-inspired protocols have fully, formally specified communication models that are carefully constrained to permit a finite number of capabilities. The absence of both unintentional ambiguities and intentionally under-specified, future-proof components necessarily implies that, when properly implemented, these protocols permit only the transmission of packets that are both well-formed and expected. This eliminates large classes of vulnerabilities dependent on exploiting systems through unanticipated, malicious inputs.

(U) While LangSec's heavy draw on more theoretical facets of computer science can make it seem esoteric, one should not dismiss it as an academic exercise. This discipline, and many others like it, is widely employed in industry to solve thorny cybersecurity issues in provably secure ways. At the root of this increasing adoption of formal methods is a commoditization of its abstruse, core concepts in the form of higher-level programming languages and improved tooling.

The authors are actively investigating the application of these new languages and tools to rapidly field improved solutions for crossing system boundaries under a research program that they have dubbed High-Assurance Interface Protocols (HAIP).

3.1 (U) Toward HAIP for DoD Applications

(U) The primary focus of the authors' research so far has been on cyber-hardening the interface protocols used by legacy systems. These systems present a unique challenge, in that making an invasive change to their design may be impossible without threatening interoperability or inviting undue re-accreditation burden. Thus, the current path is one of producing a bump-in-the-wire device that protects an interface external to its host system.

(U) For example, consider a critical legacy system that by most measures is secure on its own but relies on inputs from COTS peripherals of unknown provenance; as is a common occurrence given the current acquisition strategy. If these peripherals communicate via one of the many standardized protocols demonstrated in open-source literature to be a vehicle for compromise (e.g., USB), the security of the host system would be at risk. Rather than begin a large-scale forensic effort to ascertain the level of trust one can place in these suspect devices, the authors instead deploy a physical piece of hardware, primarily consisting of a small, single-board computer and requisite interface components, that mediates all traffic between the target host system and its untrusted input devices. While the devices themselves may present a risk, their protocol traffic, when properly filtered, does not.

(U) At a high level, the protections afforded by this HAIP device are realized using a simple, three-stage pipeline:

1. (U) Raw data is decoded via methods automatically generated from a high-level specification of the protocol. Each packet maps directly to an isomorphic, intermediate representation in Haskell, our implementation language of choice. [7]

2. (U) The resultant stream of record values is inspected and filtered in a context-sensitive manner based on a provided set of application-specific rules.

3. (U) After the decoded data has passed through the filter, an action determination will occur. This could include logging filtered packets, forwarding benign packets, or capturing additional information to inform future filtration decisions.

(U) The authors elected to use Haskell, one of the higher-level languages alluded to in the previous section, as it is considered best in class for tackling challenges rooted in programming language theory. Its unique combination of referential transparency (i.e., purity), first-class data types, and a highly-expressive, strong, static type system, all modeled in the functional paradigm, greatly simplifies both the implementation and verification of our protocol firewall. Furthermore, the language's ecosystem provides straightforward transliterations to both low-level software and hardware description languages, allowing for rapid prototype in a general-purpose environment before ultimately transitioning to an embedded system or field-programmable gate array.

(U) As discussed above, research developments thus far have centered on securing interactions with USB human input devices. However, the purpose of the development of the HAIP methodology and implementation strategy was easy generalization and porting to other protocols of interest. Provided that the target protocol operates using structured data and the desired subset of possible interactions is finitely enumerable, HAIP-style protections could be deployed. This precludes applications like general network security, where the complexity of data payloads challenges the precise specification the authors require. But one could envision other applications targeting serial- or Ethernet-based communications with more limited application layers, such as those used by industrial control systems or internet of things devices.

4. (U) Conclusion

(U) Interfaces and protocols will remain a profitable vector for adversaries to exploit for maneuver into and through systems if the DoD continues to use them naïvely as is the current practice. Well-founded concerns about supply-chain

risk highlight the numerous insertion points for interface-based attacks and open-source demonstrations of their potential impacts reinforce the severity of this threat. Yet current efforts to overhaul federal cybersecurity policy are falling short of adequately addressing and mitigating this very large and very real problem.

(U) Ongoing policy reforms are predilected toward two main thrusts: overhauling the assessment and authorization process and standardizing around a more homogenous information technology infrastructure. [8,9] In either case, the ultimate goal appears to be expediting what the authors feel are tragically flawed approaches to cybersecurity. DoD systems operate in highly contested environments where a combined reliance on human-centric processes and wholly-reactive responses is destined to fail.

(U) The authors encourage policymakers and system engineers alike to instead embrace more holistic and intrinsic approaches to cybersecurity. LangSec, HAIP, and other correct-by-construction methodologies have demonstrated the viability of securing otherwise vulnerable systems through a combination of principled design and formal verification. Their opinion is that assuming a more proactive and objective posture, like the one enforced by these methodologies, is the best way to end the cybersecurity game of cat and mouse in which the DoD finds itself.

(U) References

1. "Hackers Testifying at the United States Senate, May 19, 1998 (Lopht Heavy Industries)". Accessed 18 June 2019. https://www.youtube.com/watch?v=VVJldn_MmMY.
2. "Unbootable state for AMD devices in Windows 10 Version 1709." Accessed 18 June 2019. https://support.microsoft.com/en-us/help/4073290/unbootable-state-for-amd-devices-in-windows-10-version-1709.
3. Travis, Gregory. *How the Boeing 737 Max Disaster Looks to a Software Developer*. IEEE Spectrum. Accessed 18 June 2019. https://spectrum.ieee.org/aerospace/aviation/how-the-boeing-737-max-disaster-looks-to-a-software-developer.
4. "LANGSEC: Language-Theoretic Security." Accessed 18 June 2019. http://langsec.org.
5. U.S. Department of Defense. "Risk Management Framework (RMF) for DoD Information Technology (IT)." *DoD Instruction (DoDI) 8510.01, with changes 1-2*, Washington, D.C., 28 July 2017.
6. National Institute of Standards and Technology. "Assessing Security and Privacy Controls in Federal Information." *NIST Special Publication 800-53A*, revision 4, Gaithersburg, MD, 18 December 2014.
7. "Haskell Language." Accessed 18 June 2019. https://www.haskell.org.
8. Tracy, Richard and Gianna Price. "The Risk Management Framework is Dead. Long Live the RMF." Accessed 18 June 2019. https://www.nextgov.com/ideas/2019/06/risk-management-framework-dead-long-live-rmf/157717.
9. "Navy Aims for "Compile to Combat in 24 Hours." *CHIPS: The Department of the Navy's Information Technology Magazine*. July-September 2018. Accessed 18 June 2019. https://www.doncio.navy.mil/chips/ArticleDetails.aspx?ID=10501.