Developing a Secure COTS-Based Trusted Computing System

Trusted Computing: The COTS Perspective Series

Read About

Introduction

When examining security and Trusted Computing, system-level protection is not simply the sum of its parts. While the individual modules, operating system, and boot software all are important, system security is also not an additive process; it can't simply be bolted-on to make the system secure.

A systems designer must look at all system elements individually and understand how they work together. If a single discrete element of the system is insecure, it is not possible for the designer to claim confidently that the entire system is secure. You need to know how each system element integrates with the rest, what interfaces are available to those elements, and how each element communicates with the other parts of the system.

The systems designer must make all efforts possible to eliminate any risk of inadvertently providing an insecure port of entry into the system that makes it vulnerable to malicious attack.

Architecture Implementation

Frequently, designers can meet system-level security requirements in several different ways. For example, to resist a particular malicious attack targeting one weak hardware component, the designer could use either a more resistant component or implement a distributed system approach, presenting the attacker with the far more difficult task of targeting multiple components at the same time.

The systems designer must understand how he or she can implement different system-level architectures to address the same security requirements. Designers also must understand the tradeoffs of each architectural option. Different program managers will want to make different choices based on their unique priorities and key performance parameters.

Designers can achieve some security protections either at the system or the module level. While that decision may already be made, sometimes cost, schedule, or the limitations of a particular component may drive the designer's decision as to the level they will implement a protection on.

Interfaces and Data Communications

Another important topic for system-level security is how to set up interfaces, both within and between systems. The designer must consider the best way to design communication pathways to ensure that the system handles data in transit appropriately.

The designer must make decisions about intra-system communications between the individual modules and how to protect that data. Where interfaces send data is also important, because intra-system communications often have very different security considerations from inter-system communication.

What's more, the ways in which the system uses those interfaces, whether during operation or in maintenance activities, may raise additional considerations. The designer must perform an appropriate level of validation or authentication of data prior to its acceptance for processing at each different level of integration. That means the designer must make choices about what sort of access control or authentication capability is necessary at the I/O boundary.

COTS Hardware Vendors

Security requirements flow down either from the customer or the program office to the systems designer to provide specifics for anti-tamper or cybersecurity protections. The designer must decompose those requirements to define what each one means for the system, as well as for each of the components within it.

Sometimes designers can meet requirements for anti-tamper and cybersecurity with one solution. In other cases the designer must resolve the competing requirements of different domains. Security requirements next flow down to the commercial off-the-shelf (COTS) hardware vendors to implement any necessary mitigating technologies.

Security challenges often stem from using modules from different suppliers. Frequently, the system integrator must deal with multiple COTS vendors, but not every COTS vendor can support the same levels of Trusted Computing capabilities, so the designer also must weigh supply chain management issues and their COTS vendor's ability to meet the system's needs.

Performance and Testing

The designer must also address and understand the performance aspects of security implementations, because secure networking, encryption, and authentication can affect nominal throughput for booting, processing data, networking data, and system latency.

The impact of Trusted Computing on performance will vary from system to system, and may require tradeoffs between security and performance. Size, weight, and power (SWaP) limitations can also play a part in weighing tradeoffs. For new, cutting-edge system designs, the tradeoff might be between implementing security (and/or other overhead) and meeting the mission requirements.

Testing for system-level security brings its own set of challenges. For some security implementations, the designer might need to classify pieces of a system and test them in a classified laboratory. Afterward, when security is enabled, the system can be tested in an unclassified environment where the sensitive component can be integrated with additional elements of the system. The challenge lies in the integration of classified and unclassified components in a way that's cost-effective and ensures the ability to perform debugging in each set of environments.

System Security in the Field

Maintaining and upgrading fielded systems is another area with potential implications for security. The designer has to make decisions about whether to allow software updates in the field and, if so, how to validate the software.

If one system module fails in the field, there must be a process to authenticate its replacement properly. When the system comes into a maintenance depot for repair, it's important to understand what parts of the system its maintenance ports can access.

Prior to a system's deployment, the designer must decide how to manage security certificates and keys. The key management plan must be fully vetted, and include the ability to revoke keys to reduce the program's long-term maintenance costs. These decisions should be made early in the program, since the choices will affect how they are designed into the system.

Defining Protection Requirements Early On

It's always easier to make decisions about Trusted Computing protections at the very beginning of system development. Security touches every element of the system; to add in needed "hooks" after a module or system is already designed often requires undoing a lot of previously undertaken and costly design work.

While requirements for security might not exist at the beginning of a program, the systems designer can often anticipate them for the future. It's important for the systems integrator and his or her suppliers to discuss long-term expectations for security early on in the design stage.

Implementing Trusted Computing protections later in the design cycle can be expensive and wasteful, since all other system development must wait for the security approach to be chosen, and all the related implications that those decisions have on the rest of the system are fully understood.

Not All Solutions Are Created Equal

When defense organizations, aerospace companies, and system integrators are evaluating embedded computing solutions, it is extremely important to understand exactly what vendors mean when they say their solutions provide Trusted Computing. It is even more important to understand the difference between solutions that provide Trusted Computing and those that offer a TrustedCOTS™ level of protection.

Curtiss-Wright TrustedCOTS solutions extend Trusted Computing best practices to every part of the development process, from design and testing to supply chain and manufacturing. The highest possible levels of protection are built into every aspect of solution development to increase the overall value that COTS solutions can provide in a secure system. This process includes careful analysis of the relationships, intersections, and dependencies among all of the various protection domains, and investigation into potential faults and failures at the lowest levels to identify the associated security vulnerabilities.

Curtiss-Wright takes a holistic view of Trusted Computing, going above and beyond the efforts of other vendors to apply the advanced protection capabilities needed to develop truly secure COTS solutions. It's one of the main reasons the company has been a trusted, proven leader in the global defense and aerospace industries for decades.

Author(s)

Steve Edwards
Director, Secure Embedded Solutions & Technical Fellow
Curtiss-Wright Defense Solutions

David Sheets
Security Architect
Curtiss-Wright Defense Solutions

Conclusion

When looking at security at the system level, it's critical for a systems designer to understand the Trusted Computing capabilities provided by each individual module, as well as how all modules interface with one another. A vulnerability in any component or communication link can compromise the entire system, and proactively identifying security requirements and implementing the appropriate protections early in the design stage can help prevent costly delays in bringing a secure solution to market.

Additionally, recognizing the trade-offs between security and performance, performing the proper testing and analysis, and working with an experienced partner that has a proven reputation deploying trusted solutions can greatly simplify the development process and help bring a secure, COTS-based Trusted Computing system to market faster.

Learn More

Curtiss-Wright Products

- Data at Rest Protection
- Security Enabled Curtiss-Wright SBC and DSP Products

Curtiss-Wright Case Study

- Building a Truly Trusted Computing Solution with COTS Hardware and Intel Security Capabilities

Curtiss-Wright White Papers

- The Many Faces of Trusted Computing: What You Need to Know to Protect Critical Platforms and Data
- Getting Secure, Intel-Based Solutions to Market Faster - Why the Hardware Vendor's Boot Security Implementation Is So Important
- Beyond Trusted Computing: Extending Protection Capabilities to Deliver TrustedCOTS Solutions
- Trusted Computing: The COTS Perspective Series
  1. Introduction to COTS-based Trusted Computing
  2. Trusted Boot
  3. Hardware Features for Maintaining Security During Operation
  4. Considering the Role of Hardware in Securing OS and Hypervisor Operation
  5. Application Development, Testing, and Analysis for Optimal Security
  6. Developing a Secure COTS-Based Trusted Computing System
  7. The Impact of Protecting I/O Interfaces on System Performance
  8. Decomposing System Security Requirements
  9. Establishing a Trusted Supply Chain
  10. Certification Authorities for Trusted Computing in Military and Avionics Products