TECHNOLOGY WHITEPAPER

The Impact of Protecting I/O Interfaces on System Performance

Trusted Computing: The COTS Perspective Series

CURTISS-WRIGHT

DEFENSE SOLUTIONS

Read About

- Performance and security trade-offs
- Cost and time to market implications
- I/O security protocols

Introduction

A trusted computing system can ensure security at the potentially vulnerable entry points of system interfaces, yet this may compromise performance through design trade-offs that systems designers must recognize, understand, mitigate, and compensate for.

Systems development often involves several different engineering groups, and the team dealing with security isn't necessarily the one responsible for project performance. The group concerned with security will identify which parts of the system need protecting to meet the program's security plan. An entirely different team can dictate performance issues involving processor speed, compute power, memory, and I/O bandwidth – all of which affect system hardware. It's not unusual for these disparate teams to be out of sync with each other.

This affects trusted computing systems because decisions made about system security inevitably affect system performance.

In addition to performance implications, there are costs to consider. Designers may opt to build and test a system mockup to gauge the overhead that security will impose, but it's important to understand the true cost of implementing trusted computing features, including the cost of additional development time and potential delays to market.

The Impact of Security

Design teams must have conversations internally at the highest level to understand trade-offs that implementing security protections can create. Authentication and encryption, for example, can influence processor use and available data bandwidth, potentially forcing designers into augmenting the system's processing power to make up for security mechanisms' effects on overall performance. It also could force system designers to consider relaxing security requirements in order to maintain performance.

Authentication

Designers must consider how to define security at the I/O boundary at the board and subsystem level. At the subsystem level are interfaces like Ethernet, MIL-STD-1553, and others that communicate outside of the box. As these interfaces communicate and receive information from other equipment, designers need authentication to ensure that communications happen only among authorized entities, and that only trusted data flows in both directions.

When design teams discuss security and performance tradeoffs, they must make choices about how to implement authentication. Will it occur only once, at power-up, or every time a message is sent? What kind of authentication should be performed? Should some sort of key exchange be used to pass keys back and forth? Such decisions have associated overhead costs. Depending on the system architecture, those overhead costs may increase startup timeline, reduce overall system throughput, or introduce additional system latency.

Encryption

Authentication is just one issue involved with security and performance tradeoffs. Calculating the implications of complex processes like cryptography and acknowledgements as data passes through the system also can be difficult.

Encryption is a two-way street when it comes to processing costs. Encrypted data must be decrypted, adding additional processor overhead. Key exchanges can introduce overhead when creating ephemeral session keys. Throughput hiccups caused from key renegotiation can impose additional overhead.

Data and Interfaces

Another often-overlooked consideration is the availability or lack of existing industry standards that define how to ensure security over data interfaces. System integrators need to understand which interfaces their system will use, and determine if standard security protocols exists for them. Ethernet, for example, has standard security protocols like Internet Protocol Security (IPsec) and Transport Layer Security (TLS).

Designers might implement such standard protocols at different levels of the IP stack; their impact on system performance will differ depending on where in the IP stack the security standard is located. What's being authenticated, and what is being encrypted also influences system performance.

Using standard interface security protocols has clear benefits. For one, the designer can implement and interoperate them on hardware modules from several vendors. When no pre-packaged security protocol exists for a particular interface, the designer, the program, or the standards body must define a secure approach for using that interface.

Legacy Interfaces

Sometimes the system designer must use an older interface that wasn't designed with security in mind. Doing this brings related concerns, like deciding whether to layer additional security code on top of the interface, or designing a unique solution.

The target platform itself may drive many of these interface and performance issues. If an Ethernet network is available, the designer can use its built-in security. If a sensor must communicate over MIL-STD-1553, there won't be as rich an ecosystem to support security. MIL-STD-1553 has been around for many decades, and is common in military systems. Unfortunately, a lot of deployed equipment can't support modern trusted computing authentication techniques.

If a legacy sensor cannot perform authentication over MIL-STD-1553, the designer must decide how, or if, to implement authentication. He or she has to determine risks and vulnerabilities, like how critical it will be not to authenticate that link.

The designer should identify and understand not only all interfaces in a system, but also any associated potential security concerns. This includes the module interfaces that already are enabled, as well as those interfaces that possibly could be enabled. Common interfaces that can have serious security implications, so design teams should not overlook them during security reviews. Designers also should consider debugging interfaces and maintenance interfaces.

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.

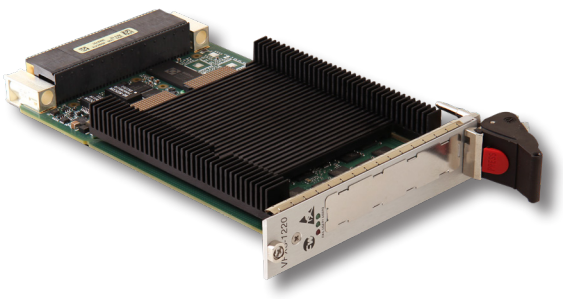

Figure 1: The VPX3-1220 is one of Curtiss-Wright's single board computers with a robust and proven security profile.

Author(s)

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

David Sheets
Security Architect
Curtiss-Wright Defense Solutions

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