RTC

HOW SAFE ARE YOUR SENSOR-BASED SYSTEMS?

CONQUERING UNDIAGNOSED CYBERSECURITY VULNERABILITIES WITH INNOVATIVE TESTING

COUNTER CYBERATTACKS ON IOT SYSTEMS

Real World Connected Systems Magazine. *Produced by Intelligent Systems Source*

SECURITY ISSUE APRIL 2017

Cybersecurity Requires a 360 Degree View

An RTC-Media Publication

Transform your business with the Internet of Things.

Start with powerful solutions from Dell

Designing Internet of Things (IoT) solutions can unlock innovation, increase efficiencies and create new competitive advantages. But in an emerging marketplace of mostly unknown and untested solutions, where should you start?

Start with a proven leader in technology solutions: Dell. Leveraging over 32 years of IT expertise and 16 years of partnering directly with operational technology leaders, we've recently expanded our IoT portfolio to include Dell Edge Gateways and Dell Embedded Box PCs.

Coupled with Dell data center, cloud, security, analytics and services capabilities, these powerful solutions can help you connect what matters and accelerate your IoT return on investment.

Dell Edge Gateway 5000

Dell Embedded Box PC 5000

Dell Embedded Box PC 3000

Learn More at **Dell.com/IoT** Today

CONTENTS

DEPARTMENTS

1.0: INTERVIEWS

2.0: CAN CYBERATTACKS BE STOPPED

CAN I TRUST YOU?

Can I Really Trust Your Embedded Devices

RTC MAGAZINE

PUBLISHER

President
John Reardon, johnr@rtcgroup.com

Vice President
Aaron Foellmi, aaronf@rtcgroup.com

EDITORIAL

Editor-In-Chief
John Koon, johnk@rtcgroup.com

ART/PRODUCTION

Art Director
Jim Bell, jimb@rtcgroup.com

Graphic Designer
Hugo Ricardo, hugor@rtcgroup.com

ADVERTISING/WEB ADVERTISING

Western Regional Sales Manager
John Reardon, johnr@rtcgroup.com
(949) 226-2000

Eastern U.S. and EMEA Sales Manager
Ruby Brower, rubyb@rtcgroup.com
(949) 226-2004

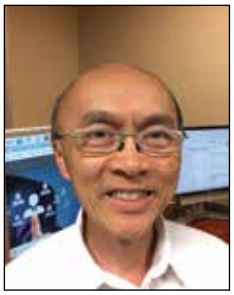
BILLING

Controller
Cindy Muir, cindym@rtc-media.com
(949) 226-2021

TO CONTACT RTC MAGAZINE:

Home Office
RTC-Media, 940 Calle Negocio, Suite 230,
San Clemente, CA 92673
Phone: (949) 226-2000
Fax: (949) 226-2050
Web: www.rtc-media.com

Published by RTC-Media Group



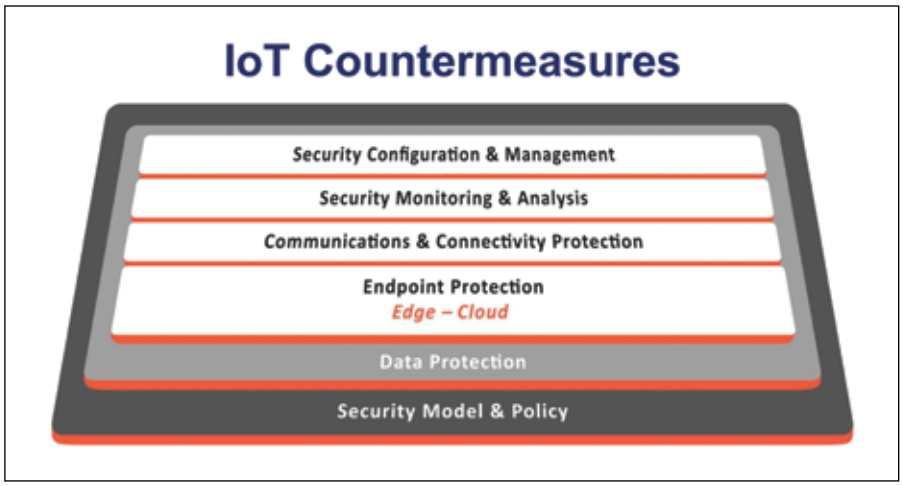
Is Cybersecurity an Illusion?

by John Koon, Editor-In-Chief

When I started writing about the topic on cybersecurity, I was eager to talk to the industry's experts in hope to find a fool-proof solution to protect us from any attacks. But to my dismay, I learned that there is no such thing as 100% security. The moment you think you have a secured system, some hackers will find a way to break it. The only time we have true security is when hackers stop hacking or we all disconnect from internet or the cloud. Otherwise cybersecurity will remain an illusion. A keynote speaker from ViaSat once made the observation that their daily cyber-attack attempts reach a billion times. This is mind boggling. Image the consequence of one successful attack. We live in a world of cyber threats and it will get worse!

So, what do you do?

I like to compare cybersecurity to the martial arts of self-defense. You want to be the best and win 100% of the time. One day, you ask the master, "What skill sets do I need to learn to win 100% of the time?" The master answers, "You will not win 100% of the time. There is always someone better than you." Then you go to search for the best teacher only to find that there are many teachers who claim they can help you to be the best. There are teachers from Kung Fu, Judo, Karate, Tae Kwon Do, Jiu Jitsu, Boxing and Kick Boxing. "Which one should I learn from?" you ask. Good question. I found that in search of cybersecurity, there are just as many answers but no one can guarantee me 100% security. One thing we know is you need to consider all aspects of security. In other words, you need to consider the 360-degree view of security to protect yourself. There are a few consortia who have devoted themselves to helping the industry to be better prepared and build security from the ground up. Among them is the Industrial Internet Consortium (IIC). Their founding members include heavyweights like Intel, IBM, SAP, Bosch, EMC, GE, Huawei, and Schneider. Their Industrial Internet of Things Volume G4: Security Framework is a comprehensive documentation covering the framework (configuration, monitoring, analysts and communication), functional view (actuation, sense operation and applications) and the system view (edge, cloud and connectivity). Additionally, it also spells out in details on how to do the end-point protection.



In this special issue of 360-degree Security: From Sensor to System, we explore security from top to bottom and have invited many security experts from the best to the biggest to share their expertise.

Caption: Industrial Internet of Things Volume G4: Security Framework is a comprehensive documentation that addresses the multilayers of security. It is critical to look at the comprehensive view of security to be effective. Image courtesy of ICC.

Future Vision on How to Fight Cyber Attacks

In this section, RTC Magazine has interviewed two special guests. Guy AlLee is the product manager, Internet-of-Things Security from Intel. Intel is the largest silicon manufacturer in the world and it has started an initiative to help the industry to be better equipped to fight cyber attacks. John Harbaugh is COO of root9B. root9B is a security company ranked number 1 on the Cybersecurity 500 list, higher than Cisco and IBM. (cybersecurity500.com)

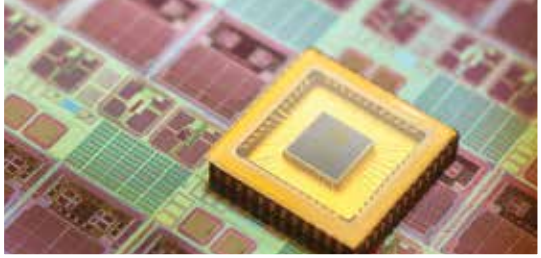
Guy AlLee, Product Manager, Internet of Things Security, Intel

Guy AlLee is a computer industry veteran with over 30 years of experience in computer architecture, hardware, software, energy, security and the Internet of Things (IoT), and has an MS in Computer Engineering. He is Product Manager for Internet of Things Security in Intel's Software & Services Group, Platform Security Division. He is currently working to enable the IoT on Enhanced Privacy ID (EPID). Before Intel, Guy was an embedded, real-time software engineer for process control, manufacturing and industrial companies Rosemont, Intercim (now DELMIA), MCT and Honeywell. He has three patents pending around anonymous attestation and IoT Security.

1. **We are seeing more and more cyber attacks in the news. It is worse if you add the unreported incidents. As we move forward with more IoT connections and more schemes such as ransomware, do you anticipate that cyber threats will stay at the same level, get worse or get much worse?**

   Let's face it: cyber-security is a continually escalating arms race. And from here, it's only going to get worse. Just in the last two or three years, the threats are becoming manifestly real to everyone. As early as 2010, Stuxnet showed us that you can weaponize cyber-attacks against things. Last fall's Mirai attack, using webcams to attack the public internet, showed that cyber-attacks using "things" is well within the capability of criminal organizations. And two summers ago, all it took was two researchers to drive a Jeep into the ditch over the wirelessly connected internet.

   Security on the Internet has historically been abou data security. But with the Internet of Things, you have expanded beyond the data domain to the control domain. IoT Security, especially in Critical Infrastructure is not just identity theft. It's the loss of reliability, safety and security that impact society and even people's health and lives. We're where we are today because there is a land-rush to get products to market in the "Wild West" of IoT. Revenue is dependent on having the features, but cyber-security is only an afterthought. Unfortunately, that is creating opportunity for cyber-criminals, as well. It's only going to get worse until we do three things. Here's your new Mantra: 1) harden the devices, 2) secure the


BE INSPIRED AT DAC

Where the Minds of Electronic Design Converge

Austin, Texas • June 18 - 22, 2017 • DAC.com

Registration Open Now

Advance Registration Discounts Until May 15, 2017

IoT – Embedded Systems & Software – Security – Electronic Design – Automotive Software – IP

DAC DELIVERS

- High Level Research & Special Sessions
- Four Engaging Keynotes
- SKYTalks (short keynotes)
- 10 Technical Panels
- Designer Track Presentations

Sponsored by: Mentor Graphics

- IP Track Presentations
- 10 Tutorials
- 3 Workshops
- Thursday is Training Day

Sponsored by: ChipEstimate.com

Sponsored by: DOULOS

WORLD OF IoT PAVILION

- Maker's Market
- IP Pavilion

DAC PAVILION

- Daily SKYTalks
- Daily CEO Fireside Chats
- Daily Analyst Market Review
- Three Tear-downs
- Industry Panels

KEYNOTE HIGHLIGHTS

Monday, June 19

IoT: Tales from the Front Line

Joe Costello

Chairman & CEO, Enlighted, Inc.

Tuesday, June 20

The Rise of the Digital Twin

Chuck Grindstaff

Executive Chairman, Siemens PLM Software

Wednesday, June 21

Accelerating the IoT

Tyson Tuttle

Chief Executive Officer, Silicon Laboratories, Inc.

Thursday, June 22

Emotion Technology, Wearables and Surprises

Rosalind Picard

Professor, MIT Media Laboratory

Sponsors as of January 11, 2017

Platinum Exhibitor: cadence®

Gold Exhibitors: Mentor Graphics, SYNOPSYS®

Sponsored by: SIGDA, Electronic System Design Alliance, CEDA (IEEE Council on Electronic Design Automation)

In technical cooperation with: SSCS

#54DAC

comms, and 3) monitor and manage security everywhere, end-to-end, edge-to-cloud.

2. How to achieve 100% cyber security?

As mentioned before, it's an arms race, so you are never going to achieve 100%. But that's not an excuse to do nothing, either. Just because you can still break into a bank, banks aren't abandoning time-lock steel vaults and just putting everything out there for anyone to take.

Today, there are five things, minimum, you have to do to harden an IoT Device:

1) Secure Boot – UEFI standard to know that you boot with the SW you expect to.

2) Secure Storage – a secure memory to keep secrets (crypto keys) that aren't vulnerable to OS or Application SW bugs and zero-day exploits.

3) HW and SW Identities – immutable identity based on a HW Root of Trust, like Intel® Enhanced Privacy ID (EPID) to know absolutely what device your application is dealing with while preserving privacy.

4) Trusted Execution Environment (TEE) – a secure protected compute environment not subject to user code privilege escalation attacks – things like the Intel® Dynamic Application Loader (DAL) or the open source software, Open-TEE. This can also provide your secure storage.

5) Application Whitelisting – An IoT device doesn't have to run any SW that shows up on it. In fact, it is likely only going to run one or two apps. So instead of anti-virus blacklisting – well after the fact when you finally know a piece of code is bad – whitelist only the one or two apps that are the only ones allowed to run on an IoT device.

3. What solutions or services does your company provide in fighting cyber threats?

Intel provides a wide range of IoT Security solutions. For the five things to harden a device, Secure Boot has been standardized. Most Intel processors ship with an embedded security engine that provides the TEE for execution and storage with the Dynamic Application Loader (DAL). Embedded in most Intel processors is an EPID private key that provides immutable identity, as well as, privacy preserving capabilities beyond PKI. Intel has standardized it with ISO/IEC 20008 and provided an open source SDK. We are licensing EPID to IoT processor manufacturers RAND-Z, and will be documenting how to use it with TPM 2.0. EPID can also be used to speed secure onboarding of IoT devices into services, and a market solution is being developed now. Finally, McAfee (an Intel company) provides application whitelisting.

For critical infrastructure, we offer Enhance Infrastructure Protection (EIP), which protects an application running at the edge, ties in comms security and uses security analytics to provide for the 3 steps in your new Mantra, above. It can even protect legacy apps on an Intel-based Gateway for brownfield deployments. A few years ago we were installing the first version in a Smart-Grid testbed, and alarms were going off everywhere. We thought we had a bug, but it turned out to be day one of Heartbleed. We identified it immediately and within an hour had a mitigation deployed to all our nodes. The rest of the world waited 14 days for a patch (that some folks still haven't deployed today).

> "Let's face it: cyber-security is a continually escalating arms race."

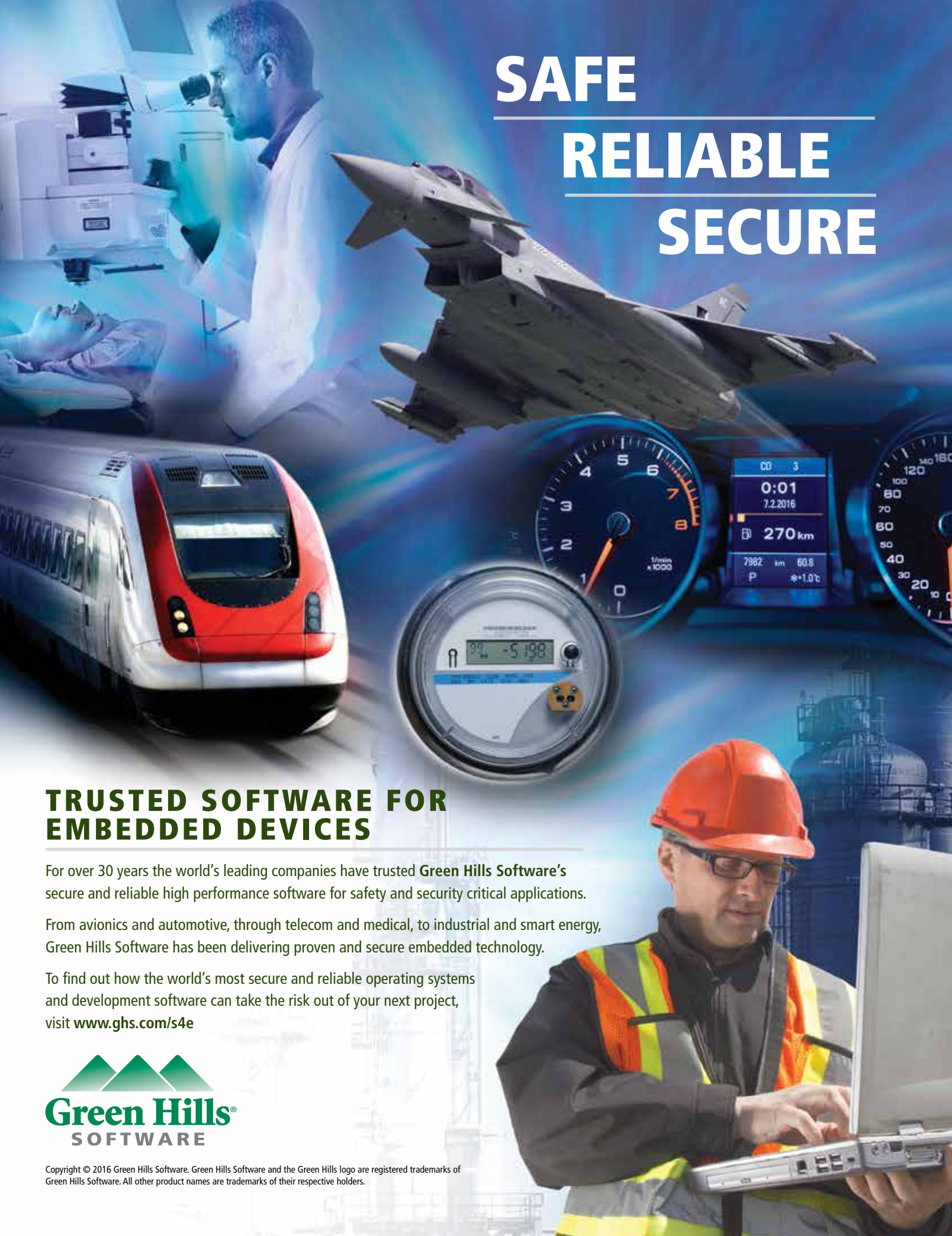
4. In view of the cyber insecurity situation we face today, what advices would you give to our readers (developers, system integrators and project managers) to be more equipped?

First, demand security of all your suppliers.

Second, given it's an arms race, you need to start with today's best practices – the five things above to harden a device – those are today's minimum requirement. We're in this for the long term, so don't think this is one and done. You also have to design the device to be able to be secure from the first minute out of the box and then secure updates to the device in the field and over its entire lifecycle. And finally, you have to continually monitor the cyber security threatscape. Today's solutions will be obsolete in tomorrow's environment. So, if you think about it, cyber-security is just one more aspect of risk analysis and mitigation that you have to fold into your design resilience, design reviews and business processes from now on.

Finally, we don't have to take twenty or thirty years like we did with the Internet. The time for implementing cyber-security in IoT Devices is now! If people can't trust their things, they aren't going to invest in the Internet of Things. I'm confident that we have the tools to secure the Internet of Things and look forward to all your innovations and due diligence in creating the promise that the Internet of Things holds.

http://www.intel.com/content/www/us/en/internet-of-things/iot-security.html



TRUSTED SOFTWARE FOR EMBEDDED DEVICES

For over 30 years the world's leading companies have trusted **Green Hills Software's** secure and reliable high performance software for safety and security critical applications.

From avionics and automotive, through telecom and medical, to industrial and smart energy, Green Hills Software has been delivering proven and secure embedded technology.

To find out how the world's most secure and reliable operating systems and development software can take the risk out of your next project, visit **www.ghs.com/s4e**

Green Hills® SOFTWARE

John Harbaugh, COO, root9B

Mr. Harbaugh directs root9B operations, capability development, and training services, with over 25 years of US military and Federal Government Senior Executive experience. As a DoD certified Master-level technician, with skills in multi-discipline cyber defense and operations, John led several "first-ever" cyber events, to include: creation of the Air Force's cyber ops capability; is recognized as the Air Force Association's Cyber Ops Jimmy Doolittle Fellow; authored advanced cyber ops and security training programs; and holds a patent for developing innovative critical national security solutions. John also provides expert advice to senior government leaders related to cybersecurity and emerging technologies.

1. We are seeing more and more cyber attacks in the news. It is worse if you add the unreported incidents. As we move forward with more IoT connections and more schemes such as ransomware, do you anticipate that cyber threats will stay at the same level, get worse or get much worse?

The current approach of cybersecurity is not working which has been made abundantly clear by the multitude of reported, and most certainly, additional reported incidents. The damage caused by these events has affected every business sector: energy, retail, manufacturing, finance, medical, insurance, etc. private and public.

In most cases victims, adhere to regulatory requirements, implement industry accepted standard practices and were compliant. These measures, while necessary, have little to no impact on the adversary's ability to successfully breach a network. Current cybersecurity best practices guide how cybersecurity professionals should harden their infrastructure and what their passive technologies should monitor. Unfortunately, these same best practices, regulation, and compliance requirements provide a playbook for the adversary.

"cyber insecurity situation as a technology issue to be simply solved with better coding, automation, or compliance."

With the flood of new IoT platforms and technologies entering the space the problems we are facing in cyberspace today will only get worse. These new technologies are being rapidly developed and brought to a market hungry for automation and convenience with little to no look at security. For IoT this also translates into a direct impact on the communities' ability to understand the full impact from a best- practice, regulation, and compliance perspective. This provides yet another great attack vector for adversaries to be successful.

2. How to achieve 100% cyber security?

Given the over reliance on automation, machine learning, and passive technologies to solve the challenges of facing a human adversary, there is no point where "100% cybersecurity" is achieved. History has proven motivated and determined humans will take advantage of vulnerabilities, weaknesses, and features in machines and systems to achieve their goals and objectives.

Bottom-line, the current defense in depth approach continues to rely on automated solutions with hopes of outthinking a human adversary; this is a failed strategy if the expectation is "100% cybersecurity". The only effective counter to the human adversary is a human defender trained and equipped to serve at the core of a new cyber defense strategy which actively leverages these advanced automated technologies to pursue, counter, and defeat the adversary residing undetected inside an enterprise.

3. What solutions or services does your company provide in fighting cyber threats?

We find CISO's are frustrated with the current landscape in cybersecurity and the significant reliance on automation and passive technologies that are not solving their problem.

root9B's answer is reflected across our service and product offerings. We understand what our clients are facing day-to-day and tailor a unique defensive strategy focused on getting them great return on investment from their currently deployed technology. We then drive our HUNT operations platform (ORION), managed security services, and credential risk mitigation capability (ORKOS), while integrating our advanced threat intelligence services, to provide a solution tailored to the client's specific business context.

4. **In view of the cyber insecurity situation we face today, what advices would you give to our readers (developers, system integrators and project managers) to be more equipped?**

I would not only think of the cyber insecurity situation as a technology issue to be simply solved with better coding, automation, or compliance. The core of the issue is there is a human adversary on one side of the problem and no real counter presence of a human defender on the other side. So, when thinking about developing new products, systems, or technology platforms it is important to understand a bit about the adversary and what motivates them.

The Adversary lives with technology, analyzes their victim (business, market, intellectual property, etc.), technology (security products, boundary devices, systems and services, etc.), and people (leadership, locations, social engineering weak points, etc.). In the end the adversary will understand the victim's enterprise better than the victim's IT professionals with a cybersecurity role.

The adversary will range from nation state, organized crime, hacktivist, vandal, etc. Understanding the adversary's motive is critical for protecting those things that are most important to the business, customer, or mission. Awareness of why an adversary will target a business or technology drives a more effective defensive strategy, which properly leverages Threat Knowledge, automated security technologies to include human defenders hunting/maneuvering/actively securing a network.

https://www.root9b.com/

Tackling the 360-degree Security Challenges

You cannot stop cyberattacks! You cannot make your systems bullet-proof! You can only hope that your systems would be able to survive and withstand minimum damage after the attacks. What are you going to do about it?

by John W. Koon, Editor-in-Chief

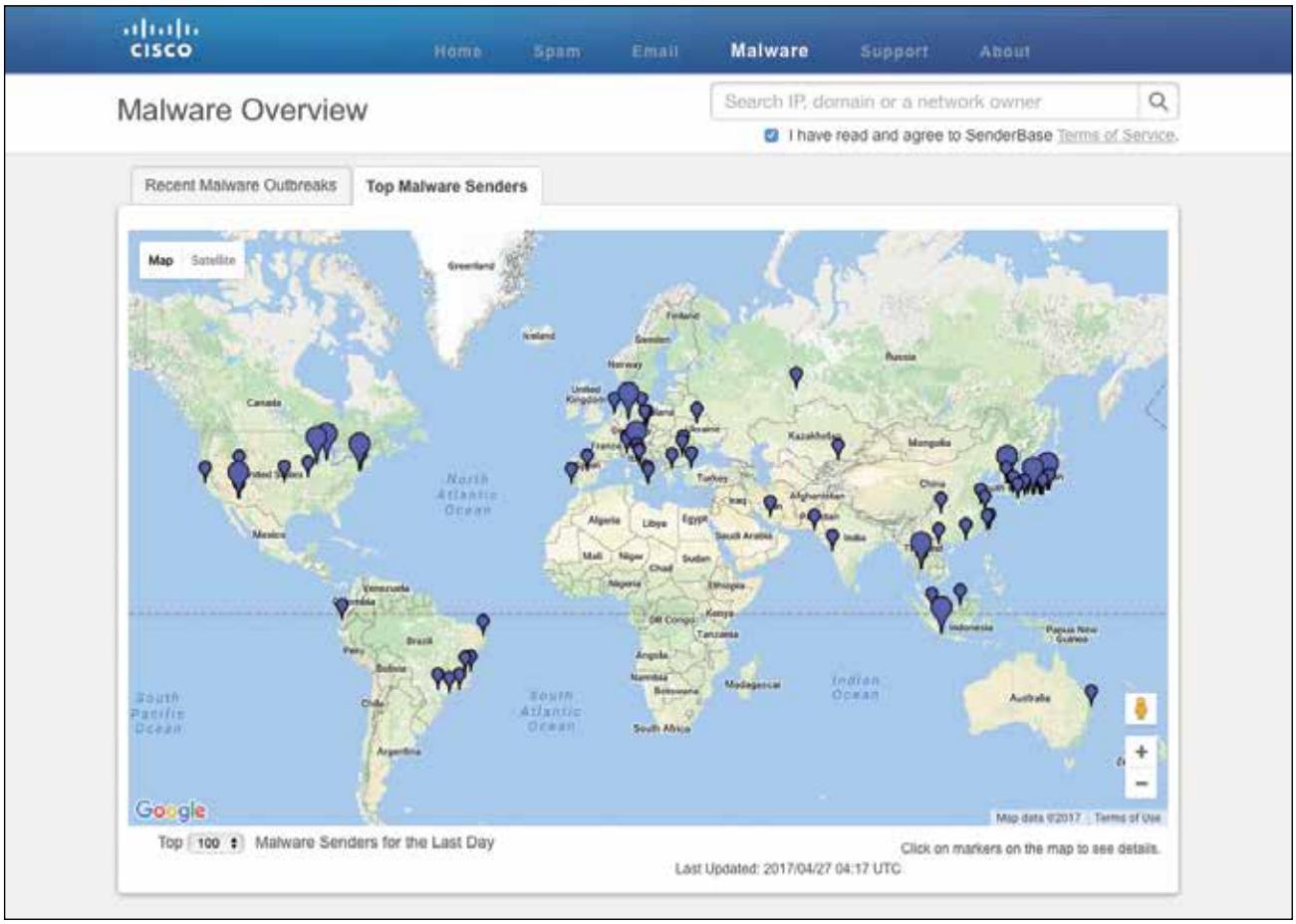


Figure 1
The Cisco Annual Cybersecurity Report, published by Cisco Talos, Cisco's threat intelligence organization, reveals threats come from many different countries and they are getting more intense. Image courtesy of Cisco Talos.

In recently years, Yahoo (500 million names hacked), Sony (unreleased movies stolen), Ukraine (power grids hacked twice with 230,000 users without power), and at least 14 hospitals (attacked by ransomware in 2016 including California-based Marin Medical Practices resulting in 2000 patients affected) were all victims of cybercrimes. Are these organizations uninformed or hackers too powerful? It is like the flu season we face each year, though we take flu shots we still get sick because the flu shot samples are only an educated guess. Big data, big storage centers and billions of IoT connections will only create more opportunities for hackers.

Cisco Talos, Cisco's threat intelligence organization keeps track of various cyber threats worldwide and publishes the Cisco Annual Cybersecurity Report, now in its tenth years, reveals how serious the problem is. Figure 1 shows attacks such as malware come from everywhere and it is getting more intense. Cloud networks can be complicated and they are as strong as the weakest links. Figure 2. With billions of connections, the clouds are totally vulnerable and open to attack anytime. That is why you need a 360-degree view of security to protect your systems.

Attempting the impossible?

It is important is to understand the nature of the attacks. They can be classified into three different categories.

- Attacks on Data in Motion (or in Transit)
- Attacks on Data at Rest
- Attacks on Hardware and systems including sensors, gateways and cloud servers

Attacks on Data in Motion

What is Data in Motion? When data files or packets are sent from one device to another whether it is an end-point such as a sensor/gateway or a server, hackers will find a way to attack, steal data, modify the data or inject malicious codes to cause damages. Attacks come in different forms. The most common example is phishing in which a fake browser pretending to be from, say, a bank you do business with when in fact it is a fake. Here is another example. It was demonstrated many times how data transfer wirelessly can be stolen. When wireless USB was first introduced, it stressed the importance of security when two devices are paired for the first time. In this case, an unauthorized third party could be waiting to be inserted in the middle (referred to as the man in the middle) and pretend to be the intended device to be connected. This is why warnings from Intel, Trusted Computing Group and many silicon companies on making sure the data to be received are indeed coming from a "real and trusted" source. Various methods such as authentication, verifying signatures and the secured use of public and private key are implemented. While it is important to ensure good data are coming from a trusted source, equally important is to prevent data ending up in the wrong hands; encrypting data using HTTPS, SSL, TLS and FTPS etc. before sending is always a good idea.

Attacks on Data at Rest

There are two types of attacks on data at rest. The first one is attacks on data collected, in a storage device, from another source to be analyzed or used. The second one is the data stored in the BIO or flash for booting purpose. It seems data in motion are in a vulnerable position because they are moving but in reliability data at rest can be even more vulnerable because the attacks can undetected for a long time. It was reported that a nuclear plant after being attacked had acted normally for a long time. Until one day, the control software started to behave abnormally. At first the readings of the control system exceed the temperature profile slightly and then back to normal. By repeating this pattern a few times, the operator was led to believe that the software was moody. One day, the system was out of control and caught everybody off guard.

Micron, one of the largest flash memory manufacturers in the world, pointed out that attacks against data at rest such as codes resides in a flash can be difficult to detect. Malicious codes can be dormant undetected for a long time as in the case of bootable media

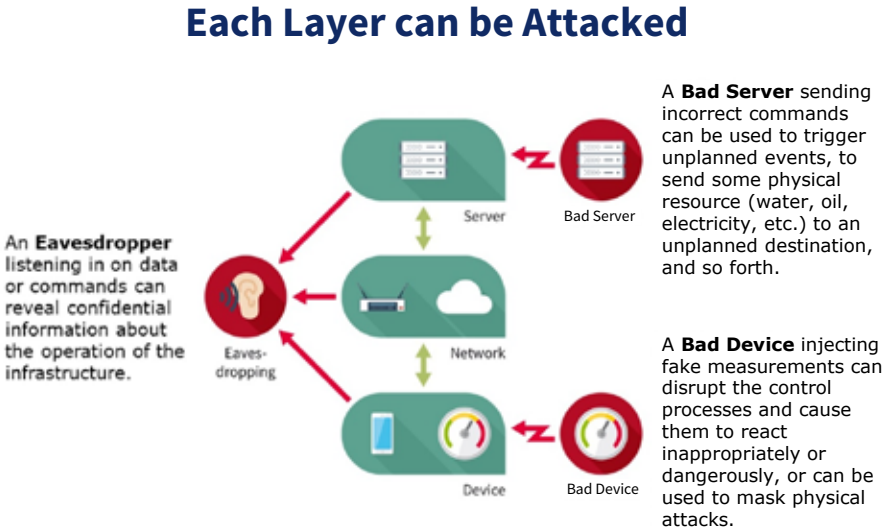


Figure 2
A network system is as safe as its weakest link. The cloud is made up of multiple connected network devices, gateways and links; hackers will find the weakest link to attack. Image courtesy of Infineon.

usually untouched by the operating system. There are, however, ways to protect such attack. One of them is replay-protected monotonic counter (RPMC) feature which provides a cryptographic primitive to select serial NOR devices, like Intel's Serial Flash Hardening Product External Architecture specification for use in the Intel Ultrabook series. With RPMC, flash memory can provide system-level anti-rollback capabilities for virtual time stamping and software version control. Attacker will be prevented from replacing system software with older versions that may contain vulnerabilities.

Some of the suggestions on protecting data in motion and at rest include implementing protective security policy by identifying at-risk areas, applying network security control such as firewall and establish policies for users to block, prompt and automatically encrypt sensitive data. Additionally, categorize and classify data for better protection as well as white list common applications. When unrecognized applications or behaviors occur, the system will automatically be warned of possible intruders.

Attacks on Hardware and systems including sensors, gateways and cloud servers

The cloud networks are built on hardware whether they are servers, gateway or sensors. These hardware and components are made all over the world and it is important to secure these hardware throughout the supply chain. For example, if the network chip does not have security measure, hackers can easily attack. That is why companies like Infineon and STmicro have security such as automatic authentication and signature recognition built-in at the silicon level. Additionally, it is important to control the device firmware while in production. Otherwise wrong or insecure codes may be installed. Segger's solution is to remotely control the test process to ensure the units in production are monitored and secure. Otherwise, the IT systems building on compromised network hardware will be open to hackers' attack.

Artificial Intelligence (AI) comes to the rescue

Humans are smart but machines are fast. Combining the two will potentially create a powerful weapon to fight cybercrime. At the recent cybersecurity conference, the RSA 2017, IBM Security, a division of IBM, announced a new weapon - Watson for Cyber Security – first augmented intelligence technology to support the cognitive security operations centers (SOCs). Sophisticated hackers attack IT systems from multiple fronts leading cybers experts through a mind game and waste many hours, in some cases up to 20,000 hours, chasing false alarms. Watson, the artificial machine, has been trained on the language of cybersecurity and process over a million security documents. As part of the IBM's QRadar security intelligence platform, the Watson-powered IBM QRadar Advisor can respond across endpoint, network, users and cloud aiding security analysts with speed and inform analysts of potential threats. Additionally, it also acts as a digital assistant allowing analysts to use voice input to assign tasks. Today, these new solutions are used by more than 40 customers including by Avnet and University of New Brunswick. This is a glimpse of what is to come. Expect more attacks as well as new weapons like the Watson to fight back.

Additionally, various consortia provide practical suggestions on how to build security from the ground up with security platform and policies. (See articles in this issue). Additionally, Intel, the largest silicon manufacturer in the world, has offered a security initiative and best practices to help the industry to be better prepared. These steps include protected boot, protected storage, creation of hardware/software ID, trusted security environments and implementation of white listing.

Conclusions:

Are we doing enough to protect ourselves? Not if hackers are becoming more innovative, resourceful and aggressive. They will prey on the uninformed or unprepared. Vulnerability will always exist. Here is a case in point. Cisco is the leading networking company providing end-to-end security solutions to Fortune 500 companies around the world. They also own Jasper a leading cloud company providing cloud connection to many companies. In March 2017, Cisco disclosed a critical software vulnerability known as the Cisco IOS and IOS XE Software Cluster Management Protocol Remote Code Execution Vulnerability. This flaw would "allow an unauthenticated, remote attacker to cause a reload of an affected device or remotely execute code with elevated privileges."

It is a positive step that such post is made public so users are made aware and can be prepared. Let us think about this for a moment. If a leading security company struggles with vulnerability, imagine how many unprotected network devices in the world waiting to be attacked. This cyber world is full of danger and challenges and it is not for the faint hearted. Are you prepared?

360 Degree Paranoia Is Only Good Sense

Connected devices are attacked whenever they are connected and powered on. These attacks come from all directions. This is especially true for IoT devices that are deployed outside organizational perimeter defenses. The device has to defend itself in all directions from threats that may come at any time, in any combination. This requires 360-degree security.

by Richard Fetik, Data Confidential

From the perspective of designing a reliable system that can protect its operation and data, there are several challenges. We have roughly the same problems as 1997, 20 years ago, but with more stuff to attack (more features), creating a larger attack surface. System and device design considerations should include security requirements, which will vary as a result of differences in the attack surface due to device mission features, and communication / connectivity model. And remember, a security system is only as secure as its weakest component.

The philosophy of the best practices approach is to provide for total system security, not elements of the system as isolated components. The first step is to provide hardening against attacks from any direction. The following step is to consider end-to-end security: sensor to local application, then to cloud storage and datacenter application, which makes heavy use of key management and exchange technologies. This article focuses on the device security part of the larger challenge.

Reasonable security objectives should include (a) that the device reliably provide the intended services/

features ("behaves as expected"), (b) that there be no undetected attacks, with few false positives, (c) that the design prevents successful penetration and compromise of the device, and (d) that the design provides graceful degradation of performance while under attack.

I am not aware of a sufficiently secure out-of-the-box environment and platform you can leverage for your device development; you will have to design security into your device from the hardware up. Both the ARM and Intel platforms have their own answers and security architectures, but using either you will need components from multiple vendors.

The secure design approach provided here is built upon principles and mechanisms such as "separation" (virtualization and firewalls), authentication, verification, authorization, access control, layers of encryption, and key management.

And even with all of these components, there may still be penetrations leading to unauthorized configuration and software changes. To detect anomalies and then to remediate discovered problems there needs to be a monitoring and management system. The most useful approach is to capture the correct configuration of this and every other managed device in a database, then to securely monitor and scan each device to find differences in the actual device configuration from the stored configuration. If a difference is discovered, then the stored configuration should be reset into the device.

There are also scoping and budget issues to be considered: there are three overlapping design paths provided in this article, corresponding to whether there is a high price point, mid-price, or low-price for the device under design, where these constrain the BOM (Bill Of Materials) budget. In general, stronger security can be achieved at higher BOM expense, but other "budget" aspects include weight, physical dimensions, energy/power, whether battery-powered, performance and latency requirements, etc. These three design paths are interleaved through the topics covered below; of necessity, to save space, many details are glossed over. The effect, though, is that there will be devices that are known in advance to be less secure; this has to be considered in the context of the larger IoT system design.

"I am not aware of a sufficiently secure out-of-the-box environment and platform you can leverage for your device development; you will have to design security into your device from the hardware up."

My advice is to build layers of security, into both the endpoint devices and the network equipment resources. The recommended approach to ameliorate and remediate the vulnerabilities in each layer of this "security stack" using the strengths of the other layers.

Minimally, the security stack layers for a secure device should include:

- virtualization and MMU support in the processor
- secure virtualization-enabled kernel or operating system
- cryptochip tied into a PKI (Public Key Infrastructure) CA (Certificate Authority)
- network firewall
- communications encryption

Some IoT device designs may be seen as secure enough for their anticipated use without all of these components; design choices will be influenced by risk management calculations and assumptions, as well as by cost/revenue projections. But keep in mind that IoT devices may be deployed in ways other than you anticipate, so document the expected security of the design, to obtain feedback from your customer. There are strong security components not on this list, including secure boot and storage firewall – these will be covered in a later article.

Separation and Virtualization Support In The Processor

The processor must support separation; there is no other way to adequately provide a secure foundation for security technologies that will layer on top. This generally is an MMU (Memory Management Unit).

Hardware virtualization support, such as Intel's VT-x, is also very valuable, some say essential. Virtualization support can be found on processors such as the ARM A53 or A73, some of the Intel Atom processors, some of the AMD processors, etc.

But the lowest end IoT devices will employ smaller, simpler, cheaper processors, less RAM, etc. These are more difficult to use as part of a secure design, but secure software coding and careful selection of other

device components can reduce the magnitude of the vulnerabilities.

Secure Virtualization-Enabled Kernel Or Operating System (O/S)

This is a complicated choice, which will drive other choices. At this time, there seem to be two "best" choices for O/S, application support. These are LynxSecure Separation Kernel Hypervisor and Green Hills Integrity.

LynxSecure Separation Kernel Hypervisor is a virtualization platform that can host "bare metal" coded application (device mission) software, and/or any combination of secure and nonsecure O/S's. Lynx requires and integrates with your processor's hardware virtualization features.

Using LynxSecure with a "bare metal" application may enable a less expensive BOM since there may be reduced RAM and storage requirements, and this combination should also permit a reduced attack surface.

Green Hills Integrity is a secure O/S. The processor must have at least an MMU (Memory Management Unit).

I recommend you look into the Yocto project (yoctoproject.org); this provides tools and templates so you can construct your own hopefully-secure embedded Linux bespoke to meet the needs of your device design.

Or you can use one of the many attempting-to-be-secure embedded O/S runner-ups. There are too many good ones to list; to evaluate one for your needs you should use good "penetration test" tools and a rigorous testing process.

Or, if you are limited to a less expensive processor, you can roll-your-own bare metal mission application. This should be statically-linked binary software, linked with any libraries needed, then stripped of anything not needed by your application. Lookup the strip and ELFKicker sstrip commands.

Device Identity Tied Into PKI, Secure Key Management

Attacks against the device you are designing may come from insider agents and peer devices, which means devices can not blindly trust each other. But IoT devices must also interoperate with data aggregation and device management systems, which implies trust relationships; this is the sort of dilemma resolved by proper use of keys, Certificates, and a Certificate Authority.

The strongest device identity and PKI (Public Key Infrastructure) solution seems to be that based on the device itself providing or generating a unique private/public key pair. It is a weaker solution if the device identity is published to the device because of the threat of a "man in the middle" attack.

A cryptochip is a cryptographic processor (cryptoprocessor). This is generally a chip or a portion of another larger processor. It provides device identity and private key generation, plus functionality for the secure generation of communication and storage cryptographic keys.

The most common cryptochip is the Trusted Platform Module (TPM), created by a standards organization the Trusted Computing group (TCG), and manufactured by several vendors. Another popular cryptochip design is from Atmel / Microchip.

There is a vulnerability if cryptochip generated keys are shared with software for data encryption / decryption, therefore sent to the mission processor over the bus, or stored even temporarily in RAM. It is a truism that anything stored can, in theory, be stolen. Cryptochips can work well in your design, if all device encryption/decryption is done within the cryptochip.

The Intrinsic ID SRAM PUF (Physically Unclonable Function) technology can provide the foundation of device identity. PUFs use the electrical characteristics of the processor to generate a private key, which can be employed to generate a public key and other unique device identity information. In theory, the private key need not be stored, as it can be re-measured and re-generated as needed. The public key can be shared with a Certificate Authority (CA), which will return a Certificate. If you use PUFs, I recommend the GlobalSign CA, which has a tested solution for working with the Intrinsic ID PUF.

It is a significant difference from the PUF approach that TPMs have their secret RSA key manufactured in, stored within, not generated as needed. The TPM security model is designed to work in conjunction with a device BIOS, bootloader, and operating system, but can also be employed by other device architectures.

A less expensive "roll-your-own" approach to key management is to use the PUF key pair to encrypt/decrypt stored encryption keys. This substitutes for the more expensive cryptochip BOM cost, physical board space, power, cooling, EMI, etc., but is certainly less secure, not least because the encryption / decryption operations on the stored encryption keys are done by the mission processor, which, assuming it is also a lower cost part, does not have virtualization separation.

One way to consider the selection of the cryptochip or similar mechanisms is through the lens of cost. TPMs cost more than the Atmel cryptochip, which cost more than the use of PUFs. A TPM is worth the expense if the budget(s) can afford it; otherwise try to afford another variety of cryptochip. But if all else fails, you can build and/or integrate most of the cryptochip functionality in software.

GAÏA CONVERTER

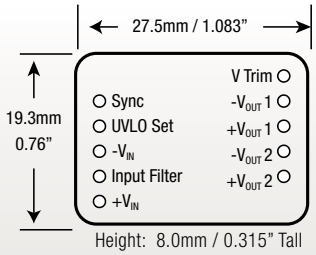
Next-Gen Platform of DC-DC Converters

FOR MILITARY & HIGH RELIABILITY APPLICATIONS

- -40 to +105°C Operation (optional -55°C)
- Compliant with Military Transient Standards
- Integrated Soft Start and LC Filter
- Synchronization Circuitry
- High Power Density / Compact Size
- No optocouplers for high reliability
- MIL-STD-461 Compliant with Filter
- Encapsulated with Metallic Enclosure

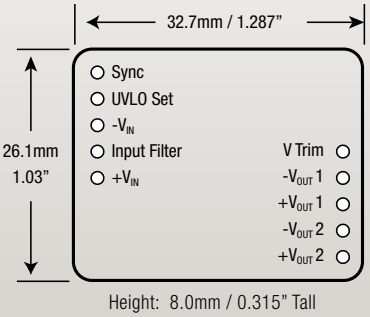
8 Watts: MGDD-08 Series

- Ultra Wide input ranges
  - 4.5-33V_{IN} Range (45V ≤ 100ms transient)
  - 9-60V_{IN} Range (80V ≤ 1sec transient)
- Dual isolated / unbalanced outputs for 3.3 ~ 50V_{OUT}
- DO-160 & MIL-STD-704 compliant
- MTBF >1.2M Hrs @ 40°C per MIL-HDBK-217F



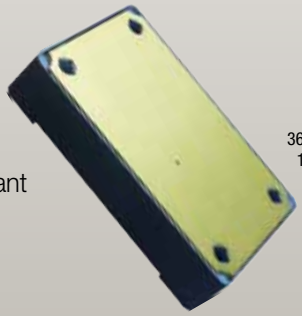
20 Watts: MGDD-21 Series

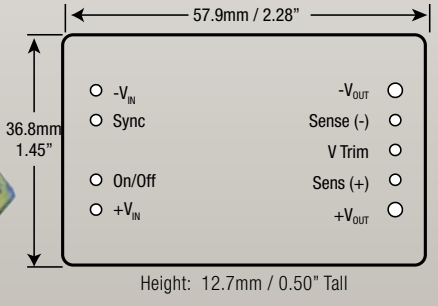
- Ultra Wide input ranges
  - 4.5-33VIN Range (45V ≤ 100ms transient)
  - 9-60VIN Range (80V ≤ 1sec transient)
- Dual isolated & unbalanced outputs for 3.3 ~ 50VOUT
- DO-160 & MIL-STD-704 compliant
- MTBF >1,060kHrs @ 40°C per MIL-HDBK-217F



150 Watts: MGDS-155 Series

- Ultra Wide input ranges
  - 9-45VIN Range (50V ≤ 100ms transient)
  - 16-80VIN Range (100V ≤ 100ms transient)
  - 150-480VIN Range
- MIL-STD-1275, MIL-STD-704 & DO-160 Compliant
- Single outputs from 3.3 ~ 28VOUT
- MTBF >490kHrs @ 40°C per MIL-HDBK-217F





Visit our website for detailed product specifications & application notes

REDEFINING THE SOURCE OF POWER

www.gaia-converter.com

Network Firewall

A network firewall sits at the network interface and prevents unauthorized communication (both in and out) across the network boundary of the device. Your device must be designed to withstand network-borne attacks from any direction, even from a trusted direction.

The network firewall for your design has to block unauthorized access, provide access control that leverages the device identity and PKI, plus it has to enable (not block) application controlled end-to-end communication encryption, and it has to maintain one or more secure TLS tunnels. Oh, and it has to prevent the communication channel from being overloaded by DOS (Denial of Service) attacks. And it has to do these with low latency.

The network firewall protects access to the device through an authorization and access control scheme, managed through a combination of the PKI solution and the management and monitoring solution (or some other credential management system).

Attempts to electronically attack this device will generally be defeated by the network firewall access control, and secure communications enabled by device-generated (unspoofable) identities and end-to-end encryption, all of which are supported by the underlying foundational technologies such as virtualization and the storage firewall. But don't forget the management and monitoring solution.

The problem you will run into is that none of the network firewalls, or any of the other network security technologies currently available, such as Irdeto, are a good match for any but the largest, most expensive, IoT devices.

Communication Encryption

Some of the IoT data aggregation, management, and monitoring solutions claim that the use of SSL / TLS (Secure Socket Layer / Transport Layer Security) as part of the network stack provides a sufficient encrypted data path. TLS is essential but not sufficient.

TLS is an "edge to intermediate node encrypted connection", and there is a significant vulnerability when the data is no longer encrypted upon reaching the intermediate node. The kludge of re-encrypting it at that point is not acceptable.

The application or application layer should take responsibility for ensuring there is an additional layer of communication encryption between the application on the IoT device and the application in the cloud; this can be referred to as an "end-to-end encrypted datapath".

Both the TLS connection and the end-to-end encrypted datapath leverage the key generation, key management, network firewall, and PKI choices described above.

Next Steps

This was a brief overview of the minimal set of security components, choices, for your device design. It omits a great deal of important information, but should provide a starting point for your planning. Focus on the value of what is being protected to determine what the security level should be, balanced by cost considerations. And don't forget to budget for the operational device management system.

About the author

Richard, CISSP, a recognized security expert, is CEO/CTO of Data Confidential, a consulting, services, and technology licensing firm for the embedded, IoT, and cloud spaces. Richard is the inventor of Data Confidential' s innovative technologies such as the Storage Firewall, secure container objects for cloud computing and secure IoT-to-cloud interactions, and a customizable storage controller architecture, built on a security framework, that accelerates application performance 100x to 1000x.
www.data-confidential.com

Ultra-High Bandwidth Recording Solutions

Get There in Record Time!

Turnkey Recording Solutions
Record at up to 10 GB/s
Up to 96 TB SSD Storage

StoreRack

Low Cost Turnkey
Prototype Platform

StorePak & StoreEngine

VPX Blades for Customized Recording Platforms

StoreBox

Compact,
Rugged,
Deployable

CRITICAL IO

www.criticalio.com

Assuring End-to-End IP Protection in Manufacturing

Cloud networks depend on secure hardware. Yet, hardware systems and components are made all over the world. How to be sure that the units in products are secure and that the IP is not copied or manipulated?

by Dirk Akemann, SEGGER

The ongoing progression of a globalized society has meant that many OEMs rely heavily on off-shore contract manufacturing partners to take care of their production in order to stay competitive. This not only allows them to keep the costs involved as low as possible, it also gives them opportunity to supplement their own limited resources. One of the major risks associated with this approach, however, is that as well as the units they have been assigned to produce, the contract manufacturer might also make counterfeit goods - thereby exploiting the IP they have been given access to.

Even with the most conservative of estimates, the electronics industry currently loses many billions of dollars' worth of revenue each year to this issue. Increasing use of contract manufacturers is only going exacerbate things further. Action clearly needs to be taken to prevent OEM production chains from being compromised. As well as to combat the increasing prevalence of overproduction practices. If OEMs are going to ensure that their business is not impacted upon, it is vital that they can keep their IP safe. Furthermore, they need to prevent the contract manufacturer from carrying out additional production runs that haven't been sanctioned. To achieve this goal, it is required to retain full control and transparency of the production process. Figure 1.

By implementing what is in principle a smart con-

cept and turning it into a highly effective solution, SEGGER has made this happen. The company has just announced the launch of the Flasher Secure offering. This highly advanced mass production in-circuit programming system can protect vendor IP regardless of the location or nature of the production site. It provides full control and detailed visibility over the entire programming procedure, through use of sophisticated authentication algorithms. This means that only authorized boot loaders and firmware can be utilized within the programming process. If any element of the system is found not to be genuine, then it simply ceases operation.

To prevent counterfeiting of electronics hardware, the Flasher Secure system reads out the MCU´s unique identifier (UID) number which have been embedded by silicon vendors into each MCU device by default that it is going to program. The UIDs are then sent to a server across a secure SSL/TLS connection. This server is under direct physical control of the OEM (or a trusted third party). Upon receiving the UID data, the server subsequently validates it - determining whether the proposed programming should be supported or not. If it is confirmed that this is acceptable, then the respective signature is generated for the MCU and sent back to the programmer so that the programming process can begin. The time taken for all this to be accomplished is so short that it has no significant effect on the programming run period - production throughput will continue as normal. The signature, which is based on a complex asymmetric algorithm where the private key is not accessible by anyone other than the OEM. Thereby the signature protects against exposure of valuable IP to potential counterfeiters - it is basically a closed system that cannot be breached. Each signature is related to the UID and is therefore completely unique to its allocated device. There is no scope for a counterfeiting operation to add extra units to any programming run (as there will be no signature for these MCUs' UIDs) - so the pre-set volume restrictions still apply. The contract manufacture can't do any unauthorized programming runs in tandem with the authorized ones, or extend runs on for longer than was planned. Neither can they create a non-approved system via copying the firmware. It precludes the copying of either the firmware or the bootloader from one device to another. Also, as every action is logged and accessible through an easy to interpret user interface, the OEM is instantly alerted if something is occurring that hasn't been previously agreed.

About the author

Based in Hilden, Germany, Dirk Akemann is Partnership Marketing Manager at embedded software specialist SEGGER. He has been with the company for the last 8 years. Prior to this he spent several years working as a Field Application Engineer for Unitronic. Dirk was formerly a Research Fellow at the Control Engineering & Mechatronics Institute of Paderborn University.
www.segger.com

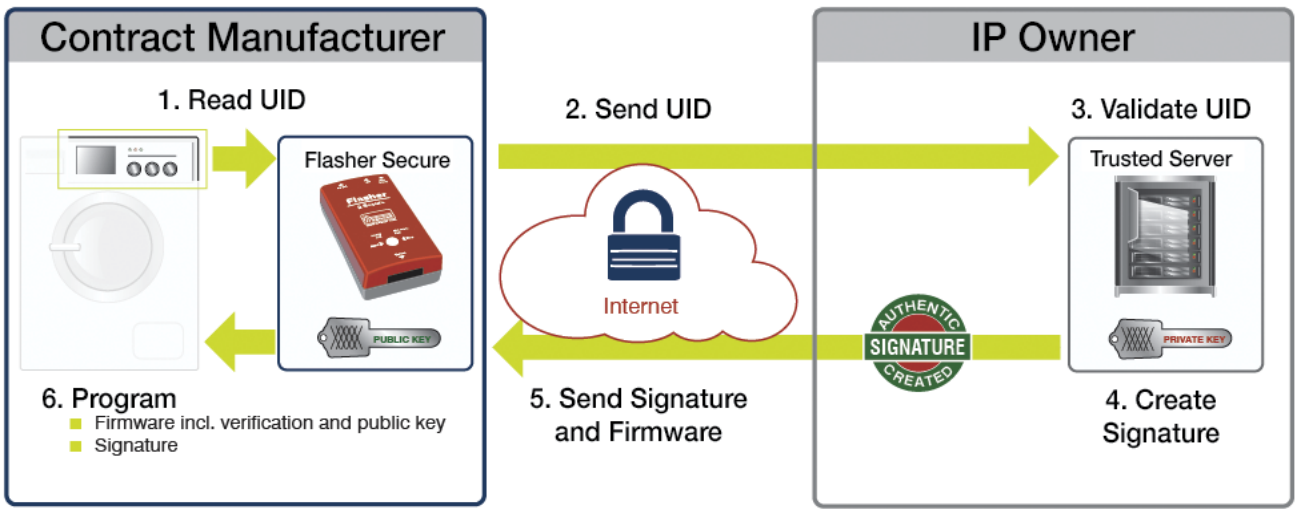
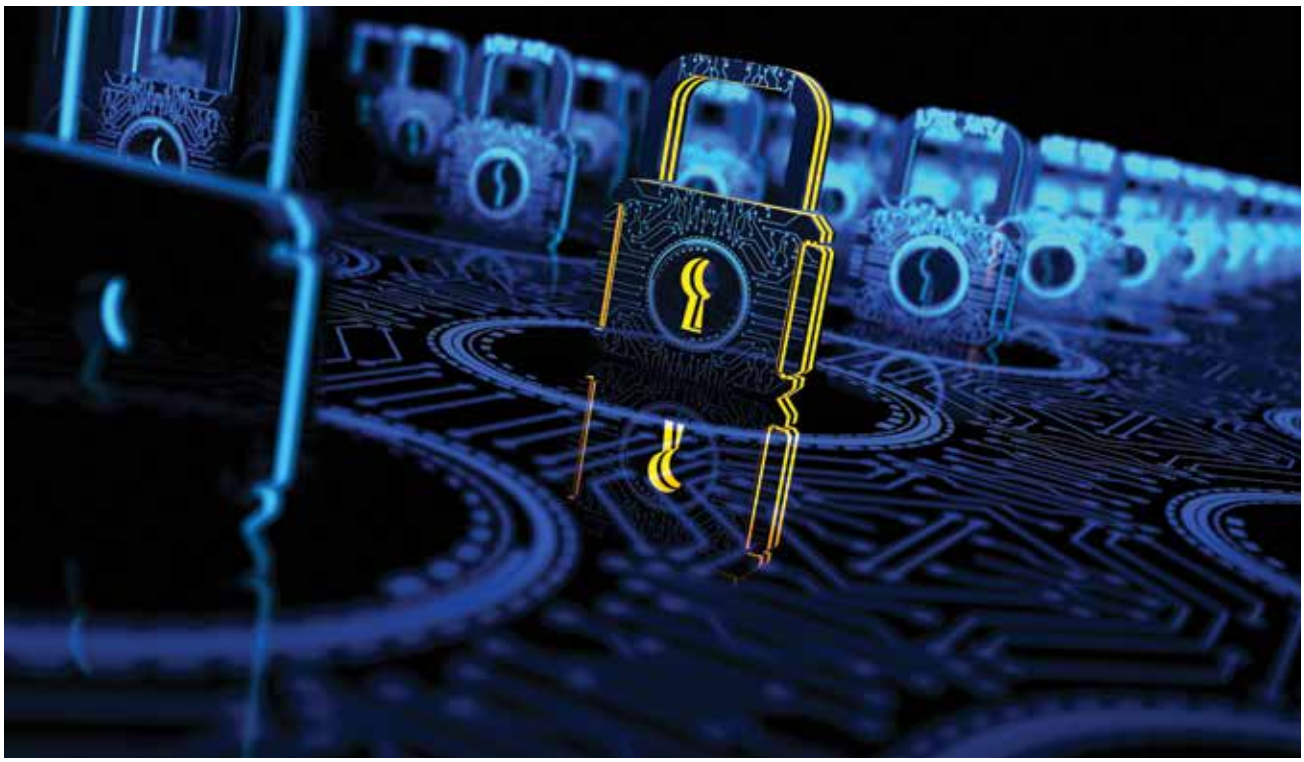


Figure 1
Schematic Showing the Mechanics of the Flasher Secure System

Safeguarding Sensor-Based Systems from Security Breaches

Hacking can cost companies more than $15 million per attack. Designing in security early via techniques like secure authentication can protect valuable systems and data.

by Hal Kurkowski, Managing Director, and Scott Jones, Executive Director, Embedded Security, Maxim Integrated



As an experiment to examine exactly how vulnerable smart, connected gadgets can be to security breaches, a reporter at The Atlantic created a virtual internet-connected toaster. After Andrew McGill placed the virtual toaster online, he expected that it would take perhaps a week for someone to perform the hack. It took only 41 minutes.

Unfortunately, hacking has grown so sophisticated that cybercriminals are writing scripts and bots that randomly—and quickly—scan ports for areas to attack. Not terribly long ago, sensor-related products simply consisted of the sensor, an analog front-end for signal conditioning, and an analog-to-digital converter (ADC). There was also software that would manage these functions, ensuring that each block could work together, and perhaps some basic data analysis routines and even a wired/wireless link. But today, we've got the internet, the cloud, and cybercriminals to consider.

Sensor-related systems that we rely on include everything from wearable health monitors to cars to industrial control systems. Breaches of such systems could lead to costly, wide-ranging damage. That's why today, it's more critical than ever for security to be built into a design from the ground up, spanning each sensor node to the web server. Yet, many companies are reluctant to seriously consider security because of concerns about associated costs, time, or effort. According to its 2016 cybercrime report, Cybersecurity Ventures predicts that such crimes will cost the world more than $6 trillion annually by 2021. Clearly, as new innovations emerge, the threats are expecting to continue to grow.

Design Security in Early to Prevent Costly Breaches

The good news is, implementing security into a design early doesn't have to be costly or time-consuming. From reference designs to secure authenticators and secure microcontrollers, there are resources and techniques available to help design engineers efficiently build in robust protection. Good reference designs, for example, now go beyond providing only the basics. Many now have drivers, test data, Spice and other models, Gerber files, detailed bill of materials (BOM), code, and evaluation and development tools to accelerate the design cycle. For example, Maxim's MAXREFDES155# and MAXREFDES143# are internet of things (IoT) embedded security reference designs that protect endpoints and sensing nodes via authenticated command and measurement between a web server and the IoT reference device. To offer flexibility, the MAXREFDES155# (shown in Figure 1) is implemented using public key-based Elliptic Curve Digital Signature Algorithm (ECDSA) cryptography and the MAXREFDES143# is based on secret-key SHA-256. Both reference designs include sensing nodes that measure environmental conditions such as temperature and light intensity as well as the secure web server implementation for system operation. To simplify end equipment development, the references are based on Arduino® form factor ARM mbed shield that represents a controller node responsible for monitoring one or more sensor nodes. Featuring a standard shield connector, engineers can use either reference design for immediate testing using an mbed board, such as the MAX32600MBED#.

For secure authentication in IoT, engineers can look to ECDSA-based ICs given the inherent benefit of simplified key distribution for the network-connected environment. Consider the example of the sensor card, which controls the field devices, I/O modules, and smart sensors that monitor and control a distributed control system (DCS) or a supervisory control and data acquisition (SCADA) system. Adding a secure authentication IC to a sensor card can help ensure that the card doesn't get spoofed, cloned, or counterfeited. Such an IC can authenticate sensor cards to the highest controlling level of the DCS/SCADA network. ECDSA cryptography, moreover, provides the security of challenge-and-response authentication. Maxim's DS28C36 DeepCover authenticator is an example of a secure IC that implements cryptography including both FIPS 186-based ECDSA and FIPS 180-based SHA-256. The product supports bi-directional authentication, secure download/boot, secure GPIO, session-key establishment, and more. The DS28C36 can be used for applications including medical and industrial sensors, peripheral authentication, and authentication of consumables.

Figure 1
The MAXREFDES155# reference design provides crypto-strong authentication without the need for secure-key storage memory on the processor.

Summary

By tapping into feature-rich reference designs and secure ICs, integrating security into a design early on doesn't have to be an onerous task. Rather, including security can—and should—be an essential consideration for any smart, connected design.

About the authors

Hal Kurkowski is Managing Director of Embedded Security at Maxim Integrated, where he has been involved with security-related products for more than 30 years, including work at Dallas Semiconductor prior to its acquisition by Maxim in 2001. He is a graduate of the University of Illinois at Urbana-Champaign with a master's degree in electrical and electronics engineering.

Scott Jones is Executive Director of Embedded Security at Maxim Integrated, where he leads a team responsible for secure authentication products. With more than 15 years at Maxim Integrated, Scott is responsible for product line management and end-customer business development. Prior to joining Maxim, he spent 15 years in applications engineering and embedded HW/SW design roles at Dallas Semiconductor and other technology companies.
www.maximintegrated.com

Conquering Undiagnosed Cybersecurity Vulnerabilities with Innovative Testing

Embedded software is at the center of the rapidly growingly Internet of Things technology evolution because it serves as the critical technology foundation. However, the requirements for securing IoT devices is complex, as these devices do not use the traditional web stack where security mitigations are commonly focused.

by Jeff Fortin, Vector Software

Gartner stated that the number of connected things in use will reach 25 billion by 2020. Technologies including the Internet of Things (IoT) continue to have a profound effect on the software industry because they enable the interconnection of the physical and virtual world based on interoperable communication technologies. Ultimately, this will result in a very large portion of electronic devices having network connectivity, and every manufacturer of those devices will necessarily enter the software business.

This also means that IoT has redefined the need for security by expanding the scope of responsibility into an entirely new category of platforms and services. Cybersecurity vulnerabilities are problematic in any situation, but with IoT applications, safety can actually become an issue when security is compromised because these applications often power safety-critical machines such as automobiles, healthcare devices, manufacturing equipment and much more.

A New Era of Security Risks

In November of 2016, The Associated Press profiled a report issued by the Department of Homeland Security (DHS) that detailed security problems with a wide

variety of IoT devices, and noted they posed "substantial safety and economic risks." Former Homeland Security Secretary Jeh Johnson was quoted as saying, "Securing the Internet of Things has become a matter of homeland security." The DHS report recommended immediate action by software developers and other stakeholders in the development and commercialization of IoT devices.

Embedded software is at the center of the rapidly growingly IoT technology evolution because it serves as the critical technology foundation. However, the requirements for securing IoT devices complex, as these devices do not use the traditional web stack where security mitigations are commonly focused. Instead, they use a combination of internet protocols as well as embedded protocols, so it is hard to apply existing penetration tools (such as those targeting HTTP interfaces or SQL injection attacks) to such devices, given their development is typically done in C or C++. Embedded protocols are nearly immune to these because they don't understand the protocol. As a result, serious issues may only be exploitable when run on the physical device.

Just like quality, security is a process that is best implemented at inception. Security vulnerabilities can enter a product as soon as the first few lines of code are written, and the real danger is if they are not detected until much later. Developing secure applications requires constant vigilance in all stages of development. Challenges need to be addressed during development because it will be too costly and complex to redesign these advanced systems after they have already been shipped. This means using tools that are capable of detecting possible vulnerabilities when writing code, integrating modules and testing compiled binaries on target hardware.

Creating the Next Generation of Secure Devices

Gartner predicts that 25% of identified security attacks in the enterprise will involve IoT devices and by 2018, over 50% of IoT device manufacturers will remain unable to address threats emanating from weak security practices.

One of the most commonly used tools by security testers is static application security testing (SAST). This type of testing is designed to analyze application source code, byte code and binaries for common vulnerabilities, including coding and design conditions that might lead to potential security vulnerabilities.

Adopting SAST is good in theory, as it is common for developers to want to know the following: a) are there any issues with the software; b) how many; and c) what and where are they? Assessing the code with a static analyzer will provide some direction, but is not a catch-all solution, especially when security is at stake. This is because SAST tools do not actually execute the code, but instead try to understand what the code is doing "behind the scenes" to identify where errors are. They analyze elements such as syntax, semantics, variable estimation, as well control and data flow to identify issues in the code.

Usually rule-based and run late in the development cycle, the results from SAST when used alone can create potential false positives (when the tool reports a possible vulnerability that is not an actual vulnerability). That leaves security engineers looking for a 'needle in the haystack' when identifying the genuine vulnerabilities. Furthermore, many SAST tools only help zero in on at-risk portions of the code to help developers find flaws more efficiently, rather than finding the actual security issues automatically. This can lead to time-consuming processes as well as incomplete analyses, both of which can be detrimental in the software development world. (See Figure 1)

In the example above, taken from the open-source web-server that powers part of Wikipedia, there is a code sample that contains a potentially exploitable NULL pointer vulnerability. While one of the inputs is checked if it is null, the other is used without such

```
1  int buffer_copy_string_buffer(buffer *b, const buffer *src) {
2    if (!src) return -1;
3
4    if (src->used == 0) {
5      b->used = 0;
6      return 0;
7    }
8    return buffer_copy_string_len(b, src->ptr, src->used - 1);
9  }
```

Figure 1
When static analysis may overlook an issue.

assurances. In tests with some common open-source static analysis tools, this issue was overlooked, while the commercial analyzer, Lint, reported it as only a possible error.

To address this problem, new dynamic unit testing methods are emerging that actually expose defects in software by generating a test case and confirming exploitability. Utilizing MITRE's classification of a common weakness enumeration (CWE), the approach uses automated software testing methods to interrogate an application's software code and identify possible weaknesses. The community-developed formal CWE list serves as common language for describing software security weaknesses in architecture and code, and is a standard, common lexicon for tools detecting such potential weaknesses.

In the CWE taxonomy, there are numerous weaknesses where the use of dynamic testing can highlight vulnerabilities -- in particular anything with hard errors such as the use of null pointers or dividing by zero.

In the dynamic testing approach, once a potential CWE is found, a test exploiting the identified issue is generated and executed. After execution, test tools can analyze the execution trace and decide if the potential CWE is a genuine threat. That issue can then be classified as a common vulnerability and exposure (CVE). (See Figure 2)

The approach is based on the "synthesis" of executions leading to specific software issues (e.g., the automatic construction of a dynamic test exploiting a given vulnerability), allowing for the identification and automatic testing of undiagnosed cybersecurity vulnerabilities. The construction of this exploit is then paired with its dynamic execution to determine if the vulnerability is genuinely exploitable. This type of dynamic testing method performs an upfront analysis of the code to detect potential issues (much like a static analyzer), which could actually contain false positives. However, once a potential issue has been identified, it also attempts to perform "automatic exploit construction."

Unlike static analysis-based approaches, this type of software security testing will only flag an issue if it is genuinely exploitable, mitigating the issues of false-positives. The generation of test artifacts allows for their future re-execution to demonstrate the mitigation of a potential issue after software redesign.

Conclusion

As new technologies continue to evolve that change the threat landscape, security is more important than ever. Every development team needs a unique process to achieve its application security goals, and there is no single tool that fits every circumstance. Static analysis security testing has its own set of benefits, but dynamic testing can further expose defects in software by generating a test case and confirming exploitability to find vulnerabilities more definitely – and ultimately creating a more secure product.

About the author

Jeffrey Fortin is head of product management at Vector Software. In this role, Mr. Fortin leads product management, driving business for all VectorCAST product lines into legacy markets as well as emerging market segments. Previously, Mr. Fortin served more than 16 years at Wind River. As director of product management, he oversaw product planning and strategy for Wind River's Intelligent Device Platform (IDP), an IoT gateway software product. Also during his time there, he served as director of field engineering where he led field teams focused on Industrial, Medical, IoT, and Aerospace and Defense (A&D) applications.
www.vectorcast.com

Figure 2
Dynamic unit testing methods can expose software defects by generating a test case and confirming exploitability. Once a potential CWE is found, a test exploiting the identified issue is generated and executed. After execution, test tools analyze the execution trace and decide if the potential CWE is a genuine threat, which is then be classified as a CVE.

Embedded/IoT Solutions

Connecting the Intelligent World from Devices to the Cloud

Long Life Cycle · High-Efficiency · Compact Form Factor · High Performance · Global Services · IoT

IoT Gateway Solutions

E100-8Q

Compact Embedded Server Appliance

SYS-5028A-TN4

Network, Security Appliances

SYS-5018A-FTN4 (Front I/O)

High Performance / IPC Solution

SYS-6018R-TD (Rear I/O)

Cold Storage

SYS-5018A-AR12L

4U Top-Loading 60-Bay Server and 90-Bay Dual Expander JBODs

Front and Rear Views

SC946ED (shown)
SC846S

- Low Power Intel® Quark™, Intel® Core™ processor family, and High Performance Intel® Xeon® processors
- Standard Form Factor and High Performance Motherboards
- Optimized Short-Depth Industrial Rackmount Platforms
- Energy Efficient Titanium - Gold Level Power Supplies
- Fully Optimized SuperServers Ready to Deploy Solutions
- Remote Management by IPMI or Intel® AMT
- Worldwide Service with Extended Product Life Cycle Support
- Optimized for Embedded Applications

TITANIUM

We Power the Cloud

80 PLUS TITANIUM

Learn more at www.supermicro.com/embedded

Can I Really Trust Your Embedded Devices?

Techniques based on open standards have been in place for several years to protect personal computers, servers, networks and even smartphones but they were difficult to implement and too expensive for embedded applications. With a focus on embedded, network-connected products, a new specification from the Trusted Computing Group extends the security and trust to several new markets.

by Steve Hanna, Infineon and Stacy Cannady, Cisco Systems

Determined people with bad intentions may stop at nothing to achieve their goals. For many years, companies and individuals have known that computers and smartphones were a means for hackers to access private information, steal money and, in general, wreak havoc - if adequate prevention was not implemented to restrict access. Today, with the unlimited connectivity of the Internet of Things (IoT), any embedded device connected to the internet becomes a target if it is the weak link to gain access to computers, networks, automobiles, facilities, and more.

To provide the protection and security to prevent unauthorized access, connected IoT devices must be trusted and must trust the things being connected to them.

The Concept of Trust

Establishing a basis for trust is essential for trusting people and connected things. For devices, a root of trust (RoT) provides a foundation for encryption, au-

thentication and more. The RoT is a minimized, strongly protected security function used for highly security-sensitive activities, including:

- Generating random numbers
- Storing and using long-term keys
- Verifying system integrity

The user benefits from the RoT include reduce risks by avoiding the compromise of long-term keys and any undetected system compromise.

Embedding Trust

After addressing the trust issue and the associated security benefits for PCs, servers and networking gear for more than a decade, the Trusted Computing Group (TCG) recently extended its efforts into embedded applications.

TCG's TPM 2.0 uses a "library" approach to allow users to choose applicable aspects of TPM functionality for different implementation levels and levels of security. TPM 2.0 includes new features and functions such as algorithm agility - the ability to implement new cryptographic algorithms as needed - and additional capabilities to enhance the security of platform services.

Figure 1 shows the TPM attached to platform. Endorsement keys (EK) verify that this is a legitimate TPM. The TPM contains a key hierarchy and can load and use a virtually unlimited number of Attestation Identity Keys (AIKs), signature and encryption keys created by the TPM owner/user. In addition, the TPM contains 24 Platform Configuration Registers (PCRs). The PCRs are populated by the platform's components starting with the root of trust for measurement (RTM) – an integrity measurement in the TPM.

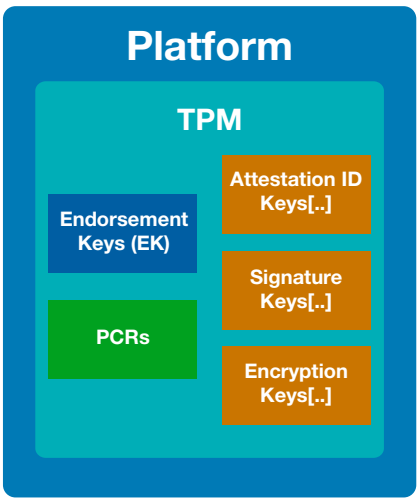


Figure 1
The TPM's general architecture includes Platform Configuration Registers (PCRs), endorsement and Attestation Identity Keys (AIKs).

As shown in the Figure 2, for improving security and establishing trust, the basic TPM has shielded locations and protected capabilities. The TPM's Shielded Locations contain data that is shielded/isolated from access by any entity other than the TPM and which may be operated on only by a Protected Capability. TPM Protected Capabilities include operations performed by the TPM on data in a Shielded Location in response to a command sent to the TPM. As part of its instantiating environment, the TPM can provide integrity reporting of software and cryptographic key creation, storage, management and use.

Today, four types of TPM are available for embedded applications, offering different tradeoffs between cost, features, and security: (1) the discrete TPM, (2) integrated TPM, (3) firmware TPM and (4) software TPM. Figure 3 shows different implementations.

The discrete TPM provides the highest level of security to ensure that it's protection does not get hacked even using sophisticated methods. Designed, built and evaluated for the highest level of security, the discrete TPM can resist tampering, including probing it and freezing it with all sorts of sophisticated attacks.

An integrated TPM has a hardware TPM integrated into a chip that provides functions other than security. While the hardware implementation makes it resistant to software bugs, the integrated TPM is not designed to be tamper-resistant.

The firmware TPM is implemented in protected software. Since, the code runs on the main CPU, a separate chip is not required. The code operates in a trusted execution environment (TEE) that is separated and protected from the rest of the programs running on the CPU. This approach creates a more difficult path for hackers to access secrets such as private keys that might be needed by the TPM but should not be accessed by others. However, it lacks tamper resistance, and the TPM is dependent on many additional aspects to keep it secure, including the TEE operating system.

Finally, the software TPM can be implemented as a software emulator of the TPM. While, a software TPM has more vulnerabilities, not only tampering but also the operating system bugs, it does have key applications. For example, a software TPM can be used to build or test a system prototype with a TPM in it. Also, the software TPM can be adapted for other form factors such as a software TPM ported to run in a TEE that creates a firmware TPM.

Use Case Examples

With the capability to provide different levels of embedded trust, several markets can now operate

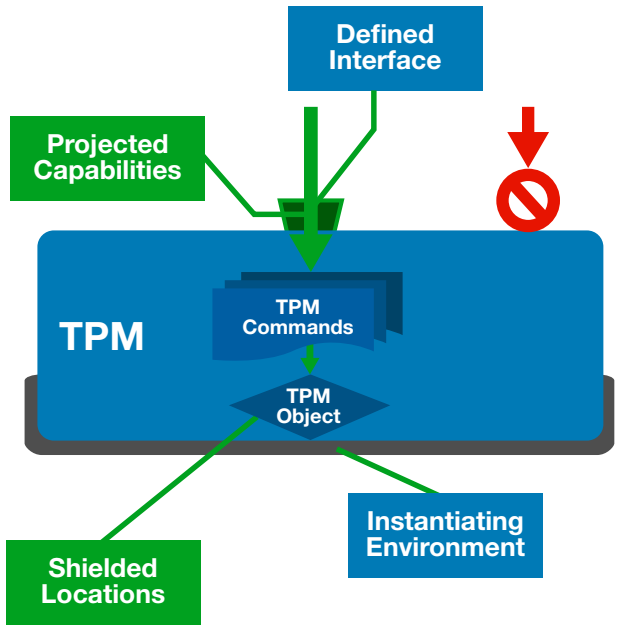


Figure 2
Basic security aspects of the TPM.

more safely. Automotive, industrial and smart buildings provide examples of the problems can be prevented.

Automotive: Today's automobiles have several interconnected networks including cellular internet access. As a mass-produced product, a vulnerability found in one vehicle through physical access can be used to exploit the same vulnerability in other vehicles - remotely. In 2015, security researchers explained how they took control of vehicle's control system (steering wheel, engine, transmission and braking systems) through a vulnerability in the infotainment system and other less than secure portions of a vehicle.

To address the unique automotive environment, TCG developed a TPM 2.0 profile called the "Automotive Thin" profile, intended to meet the requirements of the electronic control units (ECUs) that perform specific vehicle functions and only require a subset of TPM 2.0 capabilities. Figure 4 shows an example of how ECUs and data exchanged between networks can be protected by TPM 2.0.

Combined with TCG's Trusted Network Communications (TNC) capability of verifying and transferring trusted data, the TPM 2.0 Automotive Thin Profile provides higher security to carmakers and their electronic system suppliers as well as their customers. With the new approach, unauthorized access to critical control systems is much more difficult.

Manufacturing: A typical factory operational technology (OT) network includes many devices that have little or no ability to defend themselves from outside attacks since the networks were initially established for a closed factory environment. Now that they connect to external suppliers, customers and other enterprise locations through the internet, they are highly vulnerable to attacks. These attacks can put large expensive processes and people's lives at risk.

A recent steel mill incident provides a real-world example. By gaining access to the computerized systems that help control the mill's blast furnace through a spear phishing campaign, the attackers made it impossible to shut off the furnace in a controlled manner. While no one died and no major injury occurred this time, the attack resulted in significant physical damage.

In a factory environment, combining TPM hardware with public key infrastructure (PKI) and enrollment techniques, enables robust identity assumptions. The result is high confidence in the device's identity with an identity credential issued from a known and trusted root (source) and assurance that the hardware to protect keys is genuine as well as assurance that the keys associated with the identity credential are protected with hardware.

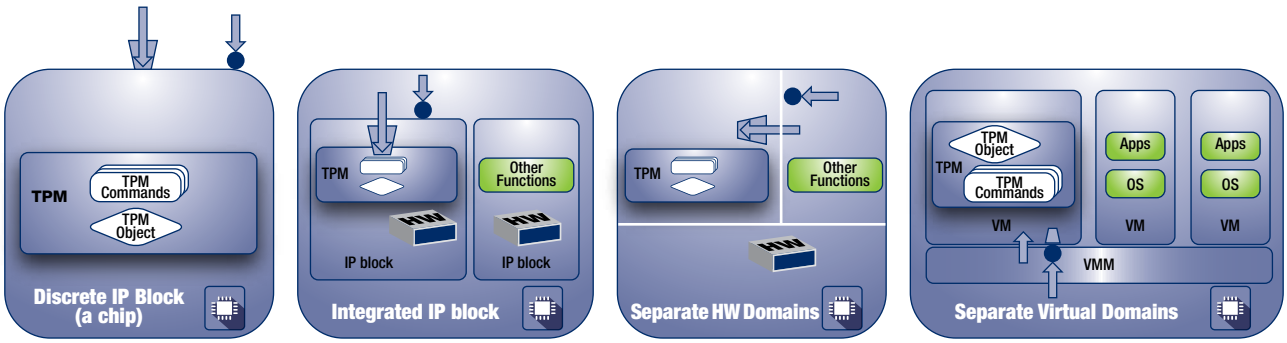


Figure 3
TPM implementation can take many forms.

IoT Know-How Starts Now

Build development skills for the Internet of Things.

Start your 4-week online class "A Developer's Guide to the IoT" to earn your certification from COURSERA® and gain a free trial of the Watson IoT Platform, hands-on access to sophisticated analytics, industry-leading security technologies, and multi-device connectivity.

Start Learning Now

ibm.com/iot/coursera

Watson IoT™

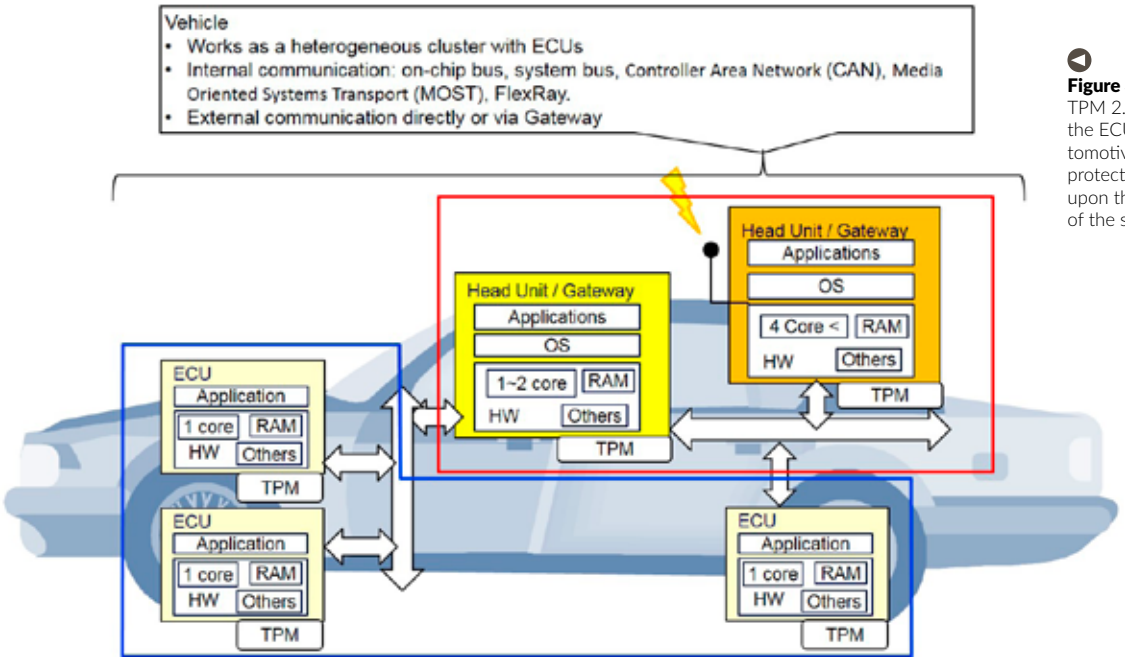


Figure 3
TPM 2.0 can protect the ECUs in several automotive networks. The protection level depends upon the critical nature of the system.

Smart Buildings: The largest smart buildings may have hundreds and even thousands of networked devices including cameras, RFID tracking, key cards, finger print and face recognition, motion and/or other sensors, as well as heating, ventilating and air conditioning (HVAC) sensors and actuators and more. Complicating the security process, the building's central management could be in a remote datacenter. Cyber security is necessary to protect the networks as well as the physical security aspects of these buildings.

A smart building with the TPM 2.0 and TCG's Trusted Network Communications (TNC) network security architecture, protects data across the network and in the cloud and credential authentication ensures authorized physical access to the smart building.

Trust Before Connecting

Essentially any embedded connected product can benefit from one of the new levels of embedded trust to reduce its potential of being the weak link in the network that allows hackers access. To take the first step towards improved embedded product security, designers need to review the readily available open specifications for the Trusted Computing Group, including TPM 2.0 and Trusted Network Communications. TCG's "Guidance for Securing IoT Using TCG Technology" is a next step and includes several references for further information. TPM suppliers can also provide insight on the choice of security level as well as cost implications. Since security and trust continue to be IoT concerns, embedded security should be a product differentiator providing increased value to users.

About the author

Stacy Cannady, CISSP, is technical marketing of Trustworthy Computing
TRIAD (Threat Response, Intelligence, and Development) for Cisco Systems. He is a representative to the Trusted Computing Group's Embedded Systems Work Group. Cannady has worked in trusted computing for Cisco and was responsible for computing at DMI, IBM and Lenovo. At IBM, he played a principal role in making the TPM standard equipment in ThinkPad and ThinkCenter PCs.
Steve Hanna is a senior principal at Infineon Technologies. He co-chairs the Trusted Computing Group embedded systems work group. He participates in the Industrial Internet Consortium and other industry groups. He is the author of several IETF and TCG standards and published papers and an inventor/co-inventor on 41 issued U.S. patents, and a regular speaker at industry events. He holds a bachelor's degree in computer science from Harvard University.
https://trustedcomputinggroup.org

MISSION CONTROL

Rugged, reliable and resilient embedded computing solutions

Whatever the operational environment—aerial, space, ground or submersible—WinSystems has you covered with a full line of embedded computers, I/O cards, cables and accessories. Our rugged, reliable and resilient single board computers are capable of processing a vast array of data for controlling unmanned systems, machine intelligence, mission management, navigation and path planning,

From standard components to full custom solutions, WinSystems delivers world-class engineering, quality and unrivaled technical support. Our full line of embedded computers, I/O cards, and accessories help you design smarter projects offering faster time to market, improved reliability, durability and longer product life cycles.

Embed success in every application with ***The Embedded Systems Authority!***

EBC-C413
EBX-compatible SBC with Latest Generation Intel® Atom™ E3800 Series Processor

EPX-C414
Quad-Core Freescale i.MX 6Q Cortex A9 Industrial ARM® SBC

PX1-C415
PC/104 Form Factor SBC with PCIe/104™ OneBank™ expansion and latest generation Intel® Atom™ E3900 Series processor

SCADA | ENERGY | IOT | AUTOMATION | TRANSPORTATION

Single Board Computers | COM Express Solutions | Power Supplies | I/O Modules | Panel PCs

WinSystems®
The Embedded Systems Authority

817-274-7553 | www.winsystems.com
ASK ABOUT OUR PRODUCT EVALUATION!
715 Stadium Drive, Arlington, Texas 76011

intel IoT Solutions Alliance

QUALITY ISO 9001:2008 CERTIFIED

IoT Device Security Starts at the Chip Level

The connectivity that makes the IoT useful also exposes any connected device to risk of cyberattack. This makes design for security a priority in any IoT project to prevent attacks that might compromise individual devices in ways that expose data or allow attackers to gain access to other assets. Security is particularly challenging for edge devices that are expected to operate in remote and relatively unsupervised scenarios. An approach that combines software and dedicated hardware secure elements – which have been validated in networked systems and payments applications – is considered optimal for cost-effective, lifetime protection of IoT devices.

by Steve Hanna, Infineon Technologies

As the IoT continues to expand, more and more of the smart devices that companies and people rely on throughout every moment of every day are connected to the Internet. With an IP address, they can be discovered by almost any machine or individual with a desire to do so - whether benign or malicious. Thus, it is imperative to block attackers from gaining unauthorised access with the intent to copy or interfere with data or code in the device. Highly publicized attacks already have made the world aware of the absolute necessity to protect connected devices in the IoT against attacks launched using techniques that can range from subtle to brutal, to achieve goals ranging from the obvious to the arcane.

Connectivity: Essence and Attack Surface

Connected devices are naturally exposed to attacks launched from across a network. On the other hand, this connectivity is key to the tremendous advantages smart things can bring to the lives of all types of users – whether they are homeowners seeking peace of mind over the status of domestic appliances, manufacturing managers using Industry 4.0 innovations to streamline production and reduce maintenance overheads, utilities monitoring smart grids, fitness fanatics collecting and sharing their physical performance data, or many other consumer or professional users.

IP connections are the route by which many smart things deliver useful data, receive instructions about what to do next, and accept software fixes and updates to stay current. To prevent attacks that may either compromise these devices directly or result in unauthorised access to other assets, all connected devices must be able to challenge any device that attempts to connect, and deny access to those that cannot present the right credentials. They also need to authenticate themselves when connecting with other devices.

The goals of attacks on IoT devices are varied. They may include attempts to disable the connected device or force it to perform unauthorised tasks; theft of application code for reuse in another product; obtaining copies of information gathered by the device; creating a backdoor into the network a device operates on; or stealing passwords to gain access to other user accounts.

As the IoT grows, many of its assets will operate as remote, autonomous, and unsupervised nodes or endpoints. They are thus not only vulnerable to online attacks, but may also be attacked physically. These attack vectors may include local uploads of malicious code via cable, replacement of E/EPROM or memory card with bogus code and/or data, complete replacement with a counterfeit unit, and even unauthorised repair of mechanical components monitored by a device.

If just one device on a network is compromised, an attacker can be free to launch any number of software-based exploits or use the device as a conduit to reach other connected assets. Serious disruption, loss, damage or even personal injury may result.

Hence the need for smart things to be able to trust other connected devices, and to authenticate themselves online when taking part in any transaction, can be easily understood.

Keeping Smart Things Safe in the IoT

While software alone is not sufficient to protect connected devices, a well-designed IoT device will employ software signing and password encryption as part of its security architecture. Digital signing provides a means of authenticating software to verify its origin and prevent unauthorised alteration. In one common technique, the software publisher applies a hashing algorithm to the code. The result cannot be mathematically reversed, which makes it impossible for hackers to write bogus code that will produce an identical hash. The publisher then uses a private key to encrypt the hash, and distributes the code together with the encrypted hash and a signing certificate that contains the associated public key.

These three elements, together, enable a receiving device to verify the code's authenticity. The hash is decrypted using the public key. The code is then hashed with the same algorithm known to be used by the publisher, and the result compared with the decrypted hash. A match confirms the code has come from the stated publisher, and is unmodified. Updates or patches received via the network can be authenticated in the same way. This process enables secure booting, which ensures the device will only load and run properly authenticated application software. Figure 1 illustrates the logical sequence of secure boot.

Encryption is also used to protect sensitive transmitted data against possible interception or eavesdropping. A receiving device has a unique private key embedded during manufacture and a corresponding public key that cannot be reversed to reveal the private key. The public key can be distributed over an open network to any sending device, which uses it to encrypt the data. When the receiving device receives data encrypted using its own public key, that receiver alone can decrypt the data using its secure private key. An independent third-party organisation maintains the Public Key Infrastructure by controlling key generation, distribution, and the binding of public keys to known identities.

Strong Identity, Rooted in Hardware

The risk of relying solely on software is that keys and passwords may be stolen, or easily overwritten in ordinary unsecured memory. If this happens, authentication processes and system integrity can be broken. To address this threat, a chip-based Hardware Root of Trust (HRoT) may be employed as a way to assure the authenticity of IoT devices and their application software.

Security chips provide attack-resistant storage for keys and passwords. They also can continuously check component authenticity as well as data and system integrity to prevent manipulation. They can

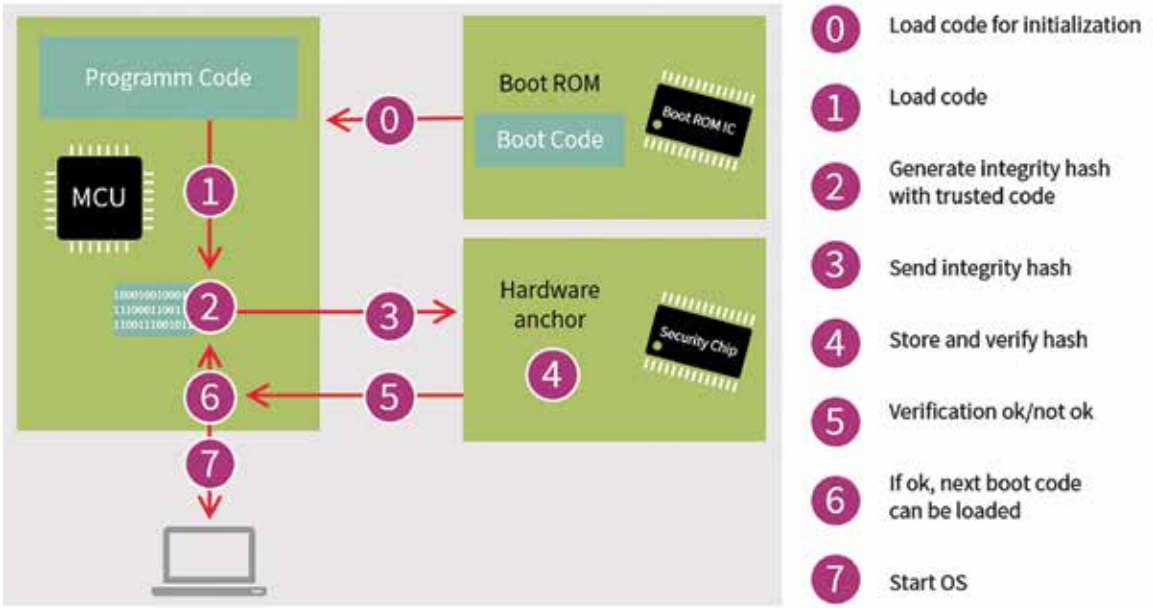


Figure 1
Secure boot authenticates program code before running.

verify the authenticity of software updates and enable protection of remote access activities, and can also provide robust protection against low-quality, counterfeit spare parts and repair tools.

In an industrial situation, manufacturers are often particularly concerned with protecting their products against counterfeiters. Infineon OPTIGA Trust security ICs provide a solution that comprises a chip and software focused on the need for authentication of electronic equipment and accessories. The chip is based on asymmetric cryptography, and is extremely compact measuring just 2mm x 3mm. Its small size and turnkey set-up allow easy integration with minimal impact on PCB size or application software. To check that a part is genuine, the host system sends a challenge (essentially a random number) to the chip in the accessory. The IC subsequently generates a response using the chip-individual key. If successfully authenticated by the host, the accessory or replacement part is accepted by the system and can be used without restrictions.

Following the same principle, a version of this IC was specifically developed for protection of high-value goods in industrial applications. It features an I2C interface as well as an extended temperature range (-40 to +85 degree C). This enables manufacturers of equipment deployed in harsh conditions, such as wind turbines, to guard against counterfeit replacement parts that can cause damage to the overall system. Both of these authentication ICs are delivered with code to simplify integration of the chip into spare parts.

Preventing counterfeiting through authentication is just the first step in the process of safeguarding the overall system. Further security functions are necessary to protect application-specific information such as customer data, intellectual property, or know-how concerning manufacturing processes or operating procedures. The OPTIGA Trust P security solution comes as a security controller with a Java Card operating system and can be flexibly programmed for a wide range of applications. This in turn allows the applications to be managed in the field, as OPTIGA Trust P supports a Global Platform specification.

Complete HRoT Solution in a Chip

A Trusted Platform Module (TPM) integrates a complete set of features that cover the broadest range of security requirements. TPMs are security controllers based on international standards developed by the Trusted Computing Group (TCG), an association of leading manufacturers from the IT industry.

A TPM uses a secure processor and operating system, and usually a hardware-based cryptographic accelerator, to execute security algorithms such as encryption, decryption, and hashing algorithms for verifying software signatures. It also provides secure storage for cryptographic keys and other data such as passwords. Additional security circuitry and secret techniques to prevent physical attack are also built-in. Unlike any other security controllers or firmware-based approaches, a TPM has full and sole

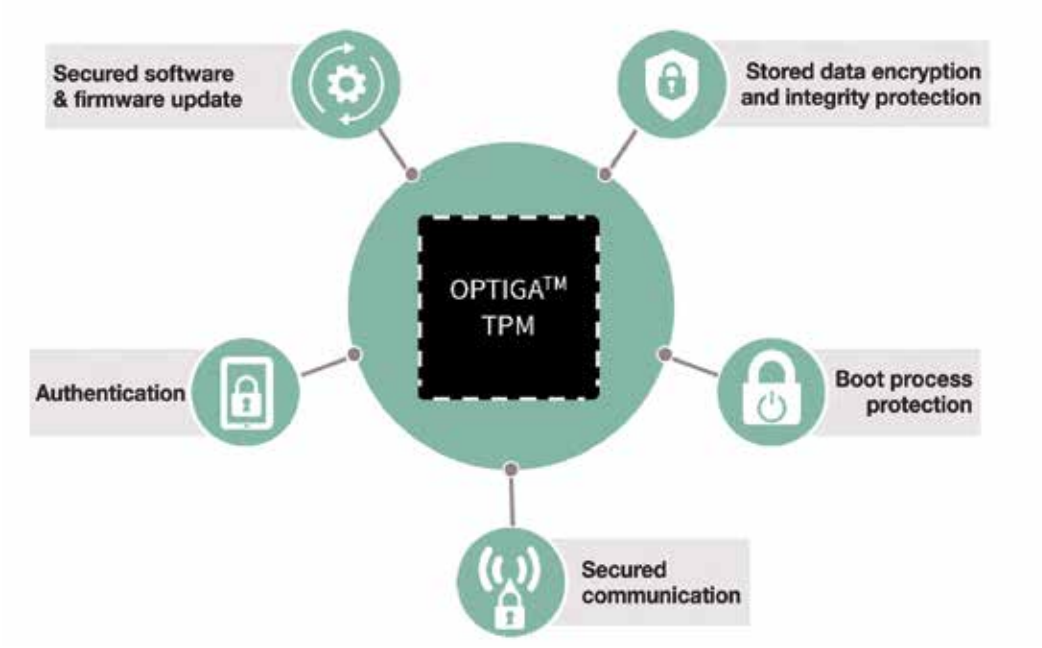


Figure 2
TPMs provide secure storage and processing for a full range of electronic security measures.

control of the dedicated internal security resources.

TPMs have already proven themselves in more than a decade of use in desktop-PC applications, and this technology is making its way into new networked systems and devices such as routers, industrial facilities and cars. Figure 2 illustrates typical TPM use cases.

TPMs are validated and certified according to the Common Criteria certification process. Manufacturers provide a broad range of security controllers for use in various processing architectures and deployments, such as x86 or non-x86 processors, and desktop or embedded systems. Depending on the application area, they are available for various temperature ranges and offer different interfaces such as SPI, I2C and LPC. Designers can choose from devices complying with either of the TCG standards in current use; namely, TPM 1.2 and TPM 2.0. Infineon was the first provider to offer products complying with the latest TPM 2.0 specification, which introduces extra features and flexibility, including an authorization hierarchy that defines four types of users, as well as additional algorithms such as ECC BN256 asymmetrical cryptography and SHA-2 256 bit hashing.

To permit easy integration into a system, the OPTIGA TPM family supports commercial and open source code for Windows and Linux derivatives and Infineon tools. Packages as small as 5mm x 5mm VQFN-32, and sleep current as low as 110µA, ensure easy integration with minimal effect on solution size or power.

One area where TPMs are commonly used in industrial applications is secured data transmission or storage. In such an application, the key factor is the combination of secured hardware and software-based security mechanisms.

Conclusion

The remote, autonomous and resource-constrained nature of IoT devices makes them particularly vulnerable to cyber-attacks. Software- or firmware-based security techniques such as signing and cryptography leveraging a PKI provide a starting point for device authentication, software protection and network access control to thwart hackers. For these to be fully effective, a secure environment for key storage and algorithm execution is needed. Secure ICs like dedicated authentication chips or comprehensively-featured TPMs enable equipment designers to embed such a robust and unalterable hardware anchor, allowing IoT devices to trust and be trusted.

About the author

Steve Hanna is a Senior Principal at Infineon Technologies. On a global basis, he is responsible for Trusted Computing strategy and for finding new markets and applications for hardware security. Mr. Hanna has a deep background in information security, especially in software and systems. He is an inventor or co-inventor on 43 issued patents, the author of innumerable standards and white papers, and a regular speaker at industry events such as the RSA Conference. He holds a Bachelor's degree in Computer Science from Harvard University.
www.infineon.com

Counter Cyberattack on IoT Systems

IoT devices and systems lack sufficient security. They are compromised by bugs—the doorways through which attackers gain access and make devices misbehave. Industry and government have created large databases of security-related software product vulnerabilities and weaknesses, and have developed specifications and tools for defining, detecting, and fixing these vulnerabilities and weaknesses. Using these resources can make IoT solutions more secure.

by Robert Hoffman, High Assurance Systems, Inc.

If you're an IoT developer, you likely worry about fielding hundreds, thousands, or even millions of devices and systems with security vulnerabilities, vulnerabilities that may cause damages for which your organization may be liable. Devices may have vulnerabilities because security is massively complicated, or your organization fears more security will be too expensive, or that usability will suffer. These are legitimate concerns.

Dr. Dan Geer is Chief Information Security Officer for In-Q-Tel, a not-for-profit investment firm supporting the missions of the U.S. Intelligence Community. In his keynotes speech at Black Hat USA 2014, Cybersecurity as Realpolitik, Dr. Geer made two points particularly relevant to IoT:

"I wish that I could tell you that it is still possible for one person to hold the big picture firmly in their mind's eye, to track everything important that is going on in our field [of cybersecurity], to make few if any sins of omission. It is not possible; that phase passed sometime in the last six years." Dr. Geer's second point, "Embedded systems either need a remote management interface, or they need to have a finite lifetime. They cannot be immortal and unfixable. Because to do so is to guarantee that if they live long enough, something bad will happen."

Dr. Geer's first point implies that good cybersecurity for embedded systems *is* complicated and difficult, and the ability to update those systems demanded by his second point brings additional security challenges.

But the biggest reason for security vulnerabilities with devices is plain old bugs—and that you can do something about. No large code base will be entirely bug-free—including third-party operating systems, middleware, libraries, or applications that a product uses (see Figure 1). But there are information resources and tools that can help detect potential problems as part of product development and operation.

National Vulnerabilities Database and SCAP

The National Vulnerabilities Database (NVD, nvd.nist.gov) contains records of vulnerabilities in software products. It is updated daily and on March 24, 2017, contained records for 84,164 vulnerabilities for 117,909 products! The NVD is part of the U.S. National Institute of Standards and Technology's (NIST) *security automation program*, which brings together open

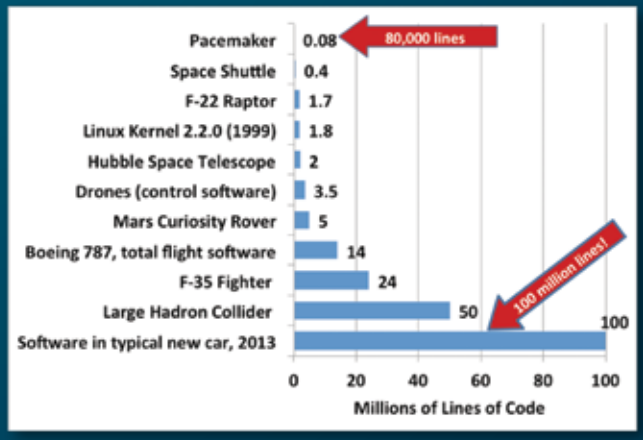


Figure 2
Code Bases. Even "small" embedded systems, such as a pacemaker, can have substantial lines of code, and software in large systems, such as a modern high-end car, has become massive (Information is Beautiful, www.informationisbeautiful.net/visualizations/million-lines-of-code).

and testing tools, and research—all to enable an organization to determine what security vulnerabilities exist in its software, measure effectiveness of its security controls, measure compliance to security policies, and prevent and detect cyberattacks.

The main content of the NVD is defined by the *Security Content Automation Protocol* (SCAP, pronounced "ess-cap"). SCAP is a suite of specifications that standardize representations of vulnerabilities, system configurations, checking scripts, metrics, and more. In addition to the data defined by the SCAP specifications, a second resource, the *Common Weakness Enumeration* (CWE) is used to help characterize each vulnerability. Figure 2 shows the key elements.

- *Common Platform Enumeration (CPE) name*. SCAP defines a name for each product/version in the NVD. For example, for XYZ Visualizer Enterprise Suite, versions 4.*.*, the well-formed CPE Name could be:

  wfn:[part="a", vendor="xyz_corp", product="visualizer ", version="4\.*", update="sp?", edition="NA", sw_edition="enterprise_suite", target_sw="os42_1979", target_hw="x86", language="eng", other="NA"]

In addition to NVD use, an organization can use these unique product names to maintain an accurate inventory of all software products on all of its devices, helping to manage upgrades, patches, product dependencies, and licenses; security policy configuration; performance measurement; and statistical analysis of vulnerabilities—all contributing to eliminating security vulnerabilities.

- *Common Vulnerabilities and Exposures* (CVE, at cve.mitre.org): The CVE is dictionary of *vulnerabilities* (an occurrence of a flaw, fault, bug, or other error in a software's architecture, design, code, or implementation that can enable an attack on the software) and *exposures* (a system configuration issue or a mistake in software that allows invalid access to information or capabilities). Each CVE entry contains its CVE ID, a brief description, and references to help identify the vulnerability. Keeping it simple allows vulnerability databases (like the NVD) and tools to use it without conflict. Vulnerabilities are now added to the CVE by dozens of product vendors and others worldwide at a rate of a dozen or more per day. The CVE and NVD are synced: updates to the CVE are fed to the NVD immediately and then augmented by NVD analysts with additional vulnerability attributes within two business days.
- *Common Vulnerability Scoring System (CVSS*, maintained by the Forum of Incident Response and Security Teams, FIRST, www.first.org/cvss): The CVSS, as used by the NVD, assigns "base," "impact," and "exploitability" scores to a vulnerability. Each score ranges from 0.0 to 10, with a base score of 7.0 or more indicating "high severity."
- *Common Weakness Enumeration (CWE*, at cwe.mitre.org): The CWE is a list of common software security *weaknesses* (errors in software that can lead to a software vulnerability). There are currently 707 weaknesses in the list, of which 123 are applied by NVD analysts for each vulnerability added to the CVE/NVD. The CWE is released about every six months. See Figure 3 for an example.
- *National Vulnerability Database (NVD*, at nvd.nist.gov): The NVD is the culmination of all the above. NVD analysts take new CVE records, score them using the CVSS and CWE, and ensure the entries work in the NVD fine-grained Vulnerability Search Engine to enable search for a specific vulnerabil-

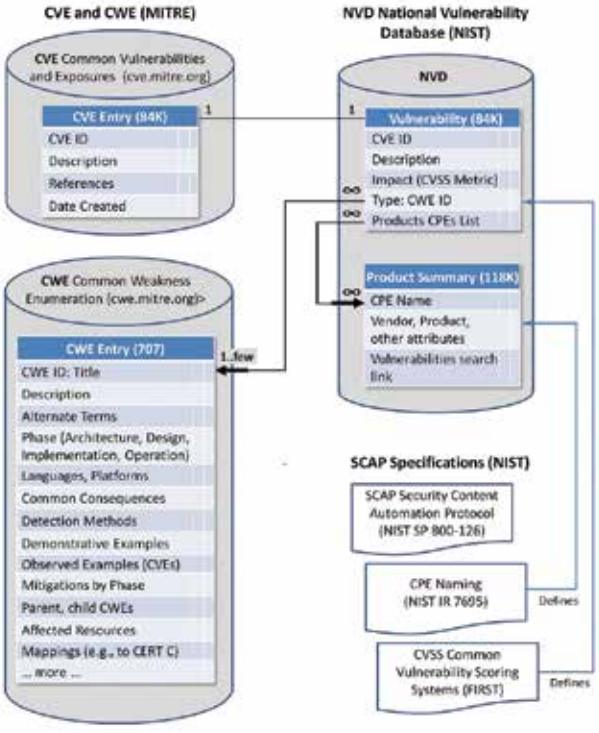


Figure 2
How the Pieces Fit: CVE, CWE, NVD, and SCAP. The NVD is the "content repository" for data defined by SCAP. SCAP is a rich set of specifications; three of the twelve for Version 1.2 are shown.

ity, for all vulnerabilities based on keywords (e.g., "buffer overflow"), for vulnerabilities related to a specific vendor, or a specific operating system, or for Open Vulnerability and Assessment Language scripts (OVAL is another SCAP specification) that can automate vulnerability checking on a system. Figure 3 shows an NVD entry for CVE-2016-5180, "heap-based buffer overflow …," for the weakness caused by CWE-787, and Figure 4 shows vulnerabilities by CWE type since 2001.

In addition to the above, SCAP defines over 400 security configuration checklists, asset identification and reporting formats, software identification tags that go beyond CPEs, and more.

SCAP and the NVD are managed by NIST; the CVE and CWE are sponsored by the Cybersecurity and Communications office in the U.S. Department of Homeland Security. However, all of these resources are the result of an international cybersecurity community effort and accessible to anyone worldwide. Organizations from around the world submit vulnerabilities to the CVE/NVD and are authorized to create CVE numbers, have made their products CVE-compatible, use CVE and CWE Identifiers in their security advisories and tools, and have adopted or promoted the use of CVE.

How to Use These Resources

There are several ways to use these resources to automate detection and elimination of security vulnerabilities in IoT products. The first is to use tools that have been validated under the CWE Compatibility and Effectiveness program and the NIST SCAP validation program. Over 90 tools and services from over 50 companies using the CWE are available for static analysis, vulnerability assessment, development lifecycle management with a security element, penetration testing, web application security assessment, and more. See cwe.mitre.org/compatible. NIST SCAP-validated products provide automated scanning against security-related configuration requirements. See scap.nist.gov/validation.

Beyond automating security management, these re-

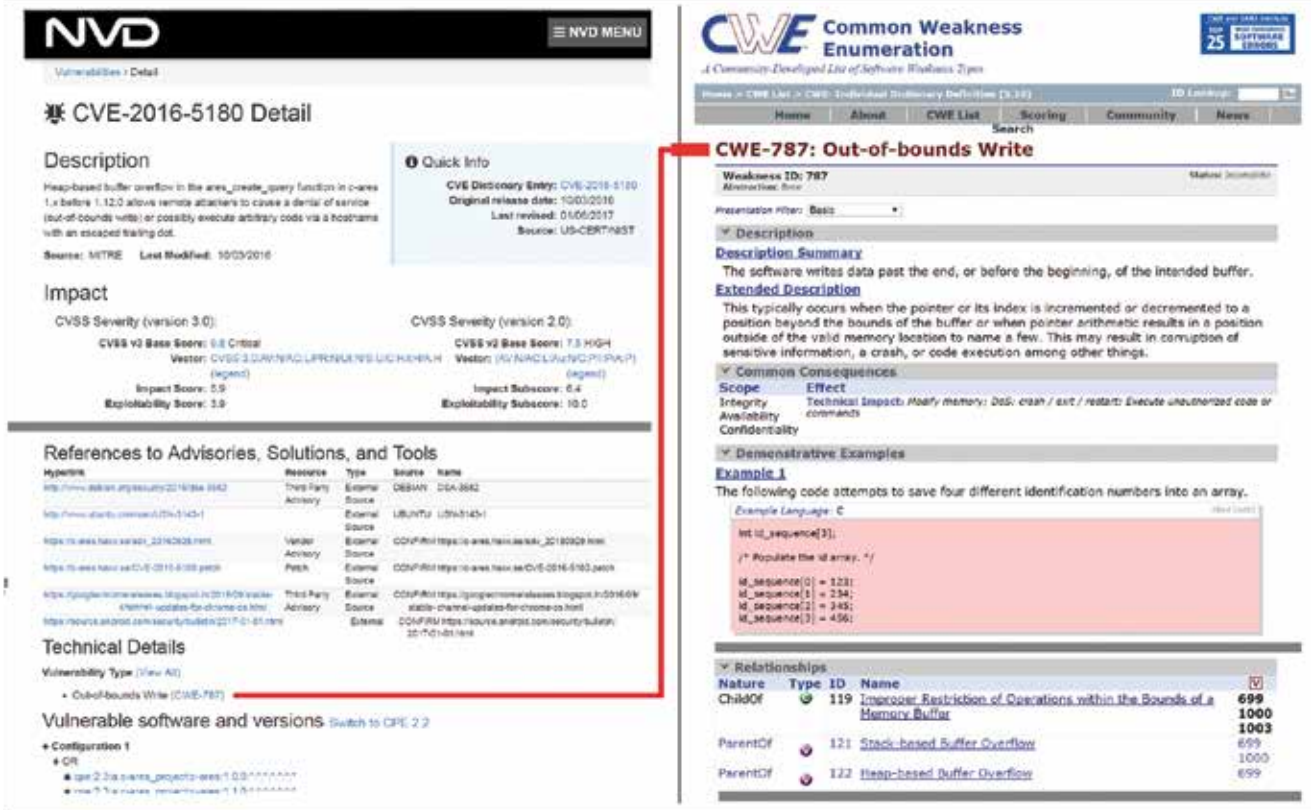


Figure 3
NVD CVE-2016-5180 and CWE-787. CVE-2016-5180 is a "heap-based buffer overflow …" (nvd.nist.gov/vuln/detail/CVE-2016-5180) caused by underlying weakness CWD-787, "Out of bounds Write" (cwe.mitre.org/data/definitions/787). It appears in Android, Debian, and Ubuntu. As Android reported, this is a "critical security vulnerability that could enable remote code execution on an affected device through multiple methods such as email, web browsing, and MMS when processing media files" (source.android.com/security/bulletin/2017-01-01.html).

sources, especially the NVD and CWE, can be used to help allocate and budget efforts to eliminate security vulnerabilities and exposures. The *Common Weakness Risk Analysis Framework* (CWRAF, cwe.mitre.org/cwraf), part of the CWE, enables an organization to rank weaknesses relative to a specific business or application and using that to focus mitigation efforts.

The Future

Estimates of the number of IoT devices range to 25 billion devices by 2020. Many of those devices will not be secure and, as a result, will cause damage to people and property. This article has been a fast tour of the NVD and related resources that can be used to reduce security vulnerabilities in IoT solutions. Other techniques are use of the NIST cybersecurity framework; guidelines and standards for defining threats, risks, policies, and procedures; participation in industry working groups and resources; security-focused design and coding standards (CERT C/C++, MISRA C/C++, NASA's Top Ten Rules, etc., and tools that detect violations of the rules for each); security-focused operating systems; secure protocols; secure device management; security and safety certification; detection and management of insider threat; and more.

The takeaway is that a worldwide public/private partnership of industry, research and educational institutions, and government parties is evolving a careful set of specifications, rules, and tools for identifying products, security vulnerabilities, and weaknesses to enable detection, remediation, and management of tens of thousands of security vulnerabilities and exposures in software products down to the version and patch level! You can reduce security vulnerabilities and encourage attackers to go elsewhere.

About the author

Robert Hoffman is President of High Assurance Systems, Inc. (HAS) specializes in developing secure and safe software and providing consulting services for developers of IoT products and products with security and safety requirements. The founders of HAS were previously responsible for a security operating system submitted to NSA/NIAP for high assurance evaluation and for an avionics safety operating system used by over 400 subsystems on over 80 commercial and military aircraft.
www.highassure.com

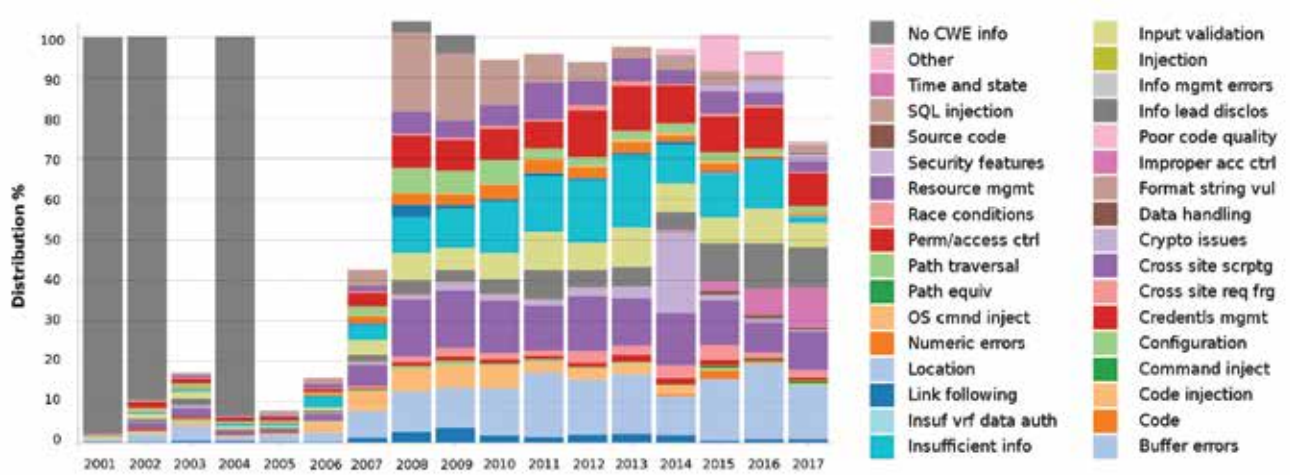


Figure 3
Vulnerabilities by Weakness by Year. Most of the 84,161 vulnerabilities (as of March 23, 2017) are assigned to a member of the "NVD CWE slice". Each bar shows the relative distribution of vulnerabilities for the top CWEs in that slice. Buffer errors (light blue near bottom) remain prominent. (nvd.nist.gov/vuln/visualizations/cwe-over-time)

Experience Real Design Freedom

Only TQ allows you to choose between ARM®, Intel®, NXP and TI

- Off-the-shelf modules from Intel, NXP and TI
- Custom designs and manufacturing
- Rigorous testing
- Built for rugged environments: -40°C... +85°C
- Long-term availability
- Smallest form factors in the industry
- All processor functions available

Embedded Modules

Powered by Convergence Promotions

For more information call 508 209 0294
www.embeddedmodules.net

What Each Developer Needs to Know to Survive

Developers must become the front line of security defense. They must internalize fundamental application security knowledge, awareness of the threat landscape, knowledge of

by Christopher Romeo, Security Journey

Imagine a product with a public IP address sitting out on the public Internet. How long does it take for that device to be noticed, much less attacked? Say an attacker scanning your factories IP space finds a device and begins to probe it. This product has software vulnerabilities, and the attacker will find them. Where did those vulnerabilities begin? Who is the person ultimately responsible for the injection of a vulnerability in software code? Is it the attacker? Or the person who wrote the original code?

The modern-day developer is a Jack or Jill of multiple trades. They must understand programming languages, test procedures, and operational constraints within a DevOps world. Developers tend to be overworked and always driven by impending deadlines. Add to all this the need to do security. Developers are fond of saying that you can have two out of three of these things: good, fast, or cheap, but not all three. "Secure" doesn't appear on that list. Developers do not inherently think security.

You could argue that software vulnerabilities first make their way into a product during the design and architecture phase, and that would be correct for some vulnerabilities. Design time vulnerabilities are usually big and ugly, such as a web service that forgot about authentication.

Software vulnerabilities are usually a result of a coding error or a bad decision. In the web world, the OWASP Top 10 is the definitive list of the top 10 most prevalent types of web based attacks. The top two items on the OWASP Top 10 are a result of developers not sanitizing input correctly. In the world of embedded, where many developers use C, issues exist in the C language itself. Unsafe functions are as old as the C language, and most developers do not understand even how to determine if a buffer overflow exists in their code.

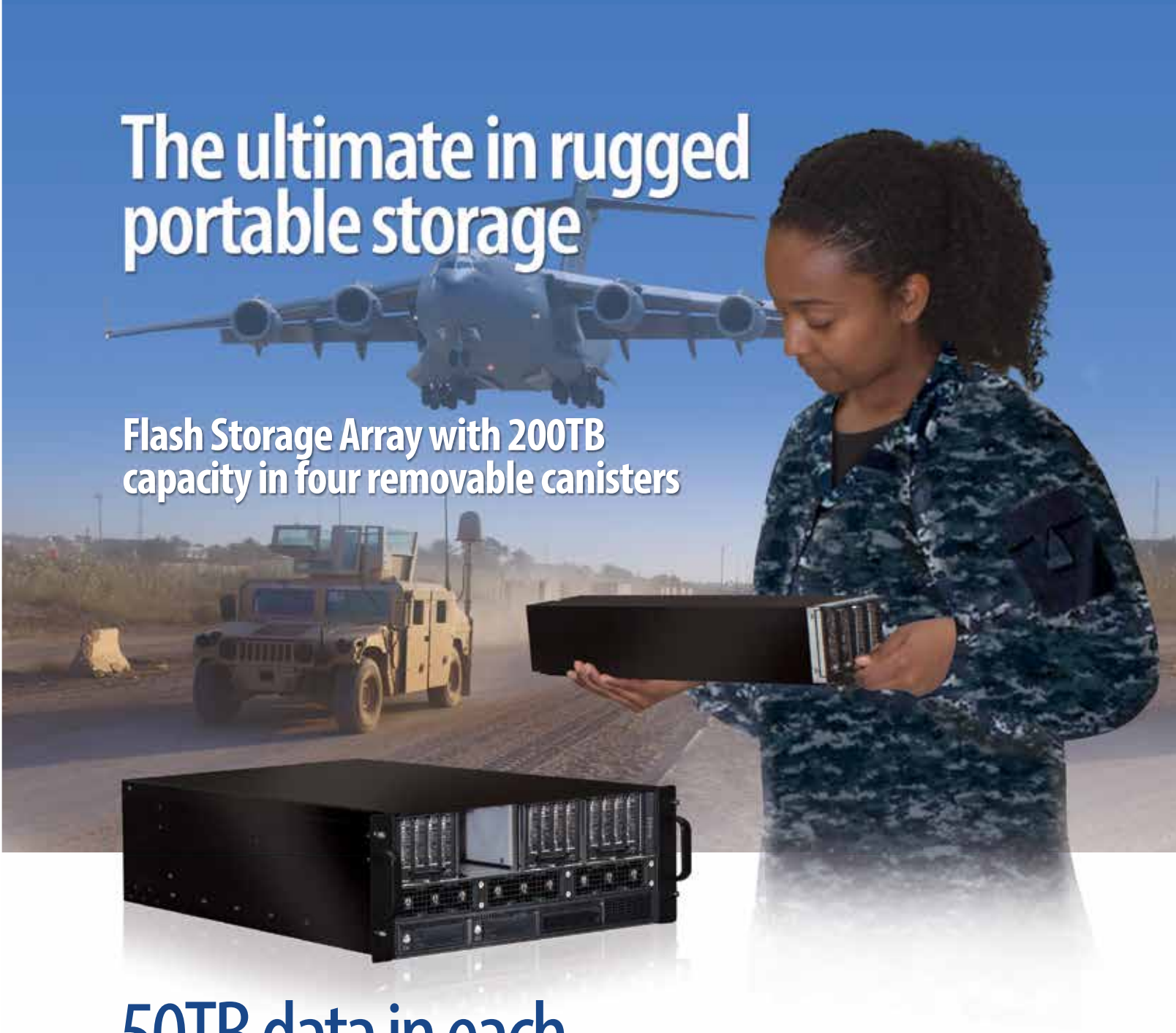
But luckily, you have a security department to do all your security so that developers do not have to worry about it. This is still a common misconception. The security department does have the knowledge and passion for security, but they are unable to scale to meet the demands of an entire development organization. The BSIMM 7 survey of software security programs found that for each 100 developers, the average company that cares highly about software security, has 1.61 security people. That is for organizations that are passionate about security. If your organization is currently lukewarm towards security, you will have much less of a presence.

Developers are the front-line defense for any piece of software. Developers are the only job role that could scale to meet the security demands of every product. Developers must embrace security and become security knowledgeable people. This does not mean that every developer must become a security expert. Being knowledgeable and aware does not equal expertise. But they do equal ability to change.

Developers are not monsters. Most are not looking for ways to inject software vulnerabilities in the code they write. They take pride in their code. The developer must become the first line of defense for any modern-day product. The challenge is that they are not educated about security.

For a developer's software to survive on the modern day wild west of an Internet, they must internalize five key principles. The principles prepare them to understand enough about application and product security to make a difference. These principles can also uncover a passion for security that takes a developer further down the road into additional study.

The first principle is a foundational knowledge of application and product security. As a security person, I used to forget that when I am speaking to a developer, they do not have the same knowledge and vocabulary concerning security that I do. Foundational knowledge includes security vocabulary, including terms such as threat, vulnerability, exploit, and attack. I've seen



50TB data in each 7 Lb. removable canister

- 100Gb Infiniband or Ethernet connections
- MIL-STD 810 and 461 tested
- Two versions: airborne and ground
- 4U rackmount unit

ONE STOP SYSTEMS

(877) 438-2724

www.onestopsystems.com

organizations where developers have been empowered through just the understanding of basic security terms. This understanding acts as a gateway to deeper understanding of security principles, and unmasks the difficulty of security.

The second principle is awareness of the threat landscape. The threat landscape is all the potential attackers that are coming after your product, and all the possible attacks they could use. The threat landscape is constantly evolving, as new attacks are invented daily, and new vulnerabilities are discovered in existing products, vulnerable to both old and new products. Developers must have an appreciation for who is trying to attack them (cyber criminals, nation states, or hacktivists) and the details about those attacks (OWASP Top 10, buffer overflows, Denial or Service, etc.)

The third principle is experience with tools and processes to improve security. The underlying security process for any development organization is the Secure Development Lifecycle (SDL). An SDL is the security activities that must occur in a standard fashion for any development. The SDL includes security requirements, secure design analysis (threat modeling), secure coding, and security test.

The field of application security tools has exploded over the last few years. Static Application Security Testing (SAST) tools exist that can scan a developer's code at certain time periods, or via an Integrated Development Environment (IDE) plugin. With the plugin, notification of insecure code is detected just like a spell checker pointing out incorrect spellings. The developer gets the notification in real time of a potential problem. Dynamic Application Security Testing (DAST) is automated software that searches for problems such as the OWASP Top 10, by profiling an application and generating various test cases that expose software flaws. DAST also can exercise protocols to attempt to uncover flaws in their implementation against a standard.

The fourth principle is understanding why someone would want to attack you. Educators who try to change the organizational approach to security fail because most times they begin with the "what". They teach people what they need to do, including following the process (SDL) and running a set of tools. They follow this up with the "how". They explain how to execute those tools and how to interpret results. The challenge is that they have missed the concept of "why". Simon Sinek has a great book and concept that he calls "Start with Why". If we begin developer education by explaining why they should care about security, we adapt their thinking before we fill in the details of what we want them to do and how to do it. An example of this is educating developers about the ramifications of data breach. If we explain why they should care about a data breach (customer's private data is leaked on the Internet and causes personal and financial pain), then when we ask them to do security activities to prevent data breach, they have a better appreciation for why they are expending extra effort.

The fifth principle is to acknowledge that products are under attack. Some developers are still naïve to the fact that attackers will attack their code. They believe such fallacies as "my code is not important" or "nobody will ever send input to my code in any way other than how I intended". The solution here is for developers to admit they have a problem. Any software that is placed on the Internet will come under attack. Attackers are probing, searching for the weakest link in software. Developers must admit that their code will come under attack versus hiding their heads in the sand.

> "Software vulnerabilities are usually a result of a coding error or a bad decision."

In conclusion, modern day developers have much to do, and security has not historically fallen high enough on their priority list. To prepare developer's creations to survive the modern Internet, teach them the fundamental principles of application and product security, explain the threat landscape, describe the tools and processes, explain why they should care, and cause them to acknowledge that their creations are under attack.

About the author

Chris Romeo is CEO and co-founder of Security Journey. His passion is to bring security culture change to all organizations. Chris is first and foremost a security culture hacker, designing security training programs and building internal security community. He was the Chief Security Advocate at Cisco for five years, where he guided Security Advocates, empowering engineers to "build security in" to all products at Cisco. He led the creation of Cisco's internal, end-to-end security belt program launched in 2012. Chris has twenty years of experience in security, holding positions across the gamut, including application security, penetration testing, and incident response.
www.securityjourney.com

STMicroelectronics - a leader in IoT

Internet of Things

- Sensors & Actuators
- Microcontrollers & Memories
- Ultra-low power connectivity
- Analog & Mixed Signal components
- Smart energy management

Build your prototype. Make your product.

STM32 Open Development Environment

A fast and affordable way to develop innovative devices and applications with state-of-the-art ST components.

For more information, visit www.st.com

Don't Monitor Your Network Without FPGA Acceleration

Monitoring a government, enterprise or telco network for cybersecurity purposes requires sophisticated packet processing. Software alone is not well-suited or efficient for packet processing at line rate, but an FPGA is ready and willing.

by Ameet Dhillon, Accolade Technology

No matter how many whiz-bang, security algorithms computer scientists develop, ultimately the "truth" lies in the packets which are flowing across the compromised network. In other words, if you are not able to in real-time, faithfully capture, process and classify the raw packets of data within which the proverbial "needle in the haystack" lies, no amount of sophisticated software will help you find the intruder who seeks to destroy your organization. At low speeds, software can provide the performance needed to capture all packets without loss; but in modern 10, 40 and 100G environments, packet loss is almost inevitable if you rely only on software. But what is the best way to deal with this problem?

The first instinct for most application developers is to solve this problem with more software, particularly given the dramatic rise in CPU processing power over the years. A good example in the networking world is the recent advent of DPDK (Data Plane Development Kit). Put simply, DPDK is a software based approach that provides a set of libraries and drivers to boost packet processing performance. It is designed to improve the notoriously poor packet processing capabilities of Linux. To be sure DPDK has great value,

but this software only approach is simply not enough; there must be some hardware acceleration or offload thrown in to the mix, particularly at 10, 40 and 100G speeds. In fact, the optimal solution may be DPDK software combined with a NIC that offers hardware offload.

If software and a standard x86 CPU is not enough, then what are the remaining hardware options? There are essentially three viable options: 1) ASIC, 2) Network Processor (NPU) or 3) FPGA. An ASIC has many attractive qualities but lack of programmability, high cost at low volume and long development cycle make it very unattractive for packet processing and application offload functionality. Both FPGAs and NPU are programmable so they work well in an environment where requirements can change and there is some need for customization, as is the case with security and network monitoring. FPGAs however have some very distinct advantages over NPUs, chief among them that new silicon technologies (e.g. 25G SerDes, DDR4 DRAM support) are available on FPGAs first. This is mostly because the FPGA market is larger (FPGAs are used across industries as diverse as defense, broadcasting and medical) and therefore more stable and profitable. This leads to greater investment, innovation, and better pricing. On the technical side, FPGAs use less power and have more deterministic latency and performance which are distinct advantages for packet processing and application offload.

If we accept that assistance from an FPGA is required to meet performance, throughput and even customization requirements, how does one go about implementing a solution? Assuming you don't have an office full of Verilog (language used to program an FPGA) designers at your disposal, you will have to purchase a third-party solution, which comes in two different forms. The first is an adapter which plugs in to a PCIe slot in your security or monitoring appliance. The second is a fully integrated, appliance platform that has an FPGA on the motherboard. With either solution, there are a few critical points to keep in mind as you make a vendor selection. The vendor should have ample in-house, design expertise to provide support and customization services as required. Often a security or monitoring application can greatly benefit from FPGA offload of a repetitive task such as a hash calculation or flow lookup; but the application developer may not even know this is possible. Share details of your application with your vendor of choice and you may be pleasantly surprised with what they can do for you. Also, make sure your vendor provides a robust API that keeps modification to your application to a minimum. Most often the API is in the form of a shared library which you link with your application and call appropriate functions to interact with the FPGA. See Figure 1.

Figure 1 shows a representative example of an FPGA-based acceleration adapter. The components in green all reside inside the FPGA and perform various critical offload tasks prior to any network traffic reaching the security or monitoring application. Packets typically enter the network ports (1,10, 40

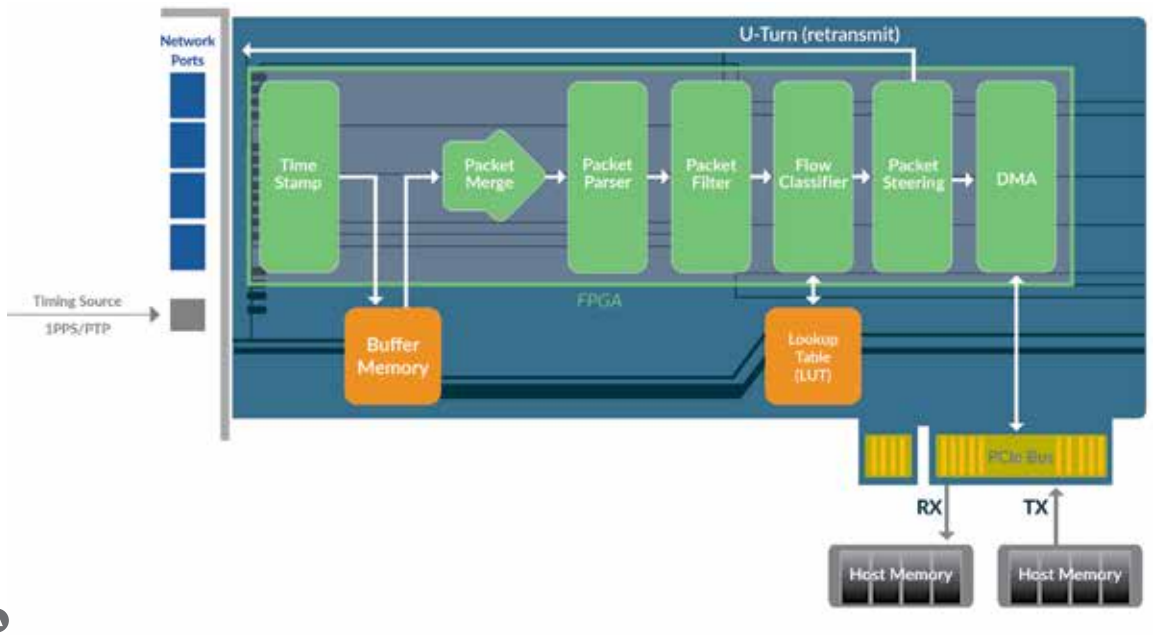


Figure 1
FPGA-based, PCIe Adapter Packet Processing and Application Offload Pipeline

or 100 GE), shown on the left, from a network TAP which passively replicates data that is flowing on the live network. Loss of data cannot be tolerated, even at 100GE speeds, so there is buffer memory to absorb temporary bursts. Nanosecond precision time-stamping is performed on each packet followed by a series of optional packet processing functions such as packet filtering, deduplication and flow classification. At the end of this pipeline, only the data or information that is explicitly requested by the host application is transferred via direct memory access (DMA) straight into host memory. Multi-core CPUs can take advantage of the adapter by explicitly requesting that certain packets be steered in to specific segments of host memory. The benefit of packet steering is that each thread in a multithreaded application (often utilizing multiple CPU cores) can process packets from its own segment of host memory. In this way a security or networking application can take advantage of parallel processing of data thus achieving higher levels of speed and efficiency.

Accolade Technology offers a comprehensive range of FPGA-based, PCIe adapters from 1 to 100G. In addition, for those software companies that simply don't want to deal with hardware at all, Accolade offers a fully integrated, 1U (half width) platform exclusively for OEM customers developing cyber security and monitoring appliances. Like COTS appliances, the ATLAS-1000 provides a multi-core Intel CPU, memory and storage along with an FPGA loaded with packet processing capabilities described earlier such as timestamping, deduplication, flow classification and multi-core DMA. In addition, the platform offers very robust timing interfaces including direct GPS decode on the motherboard. Figure 2 shows a comprehensive architectural layout of this hardened platform. Figure 2.

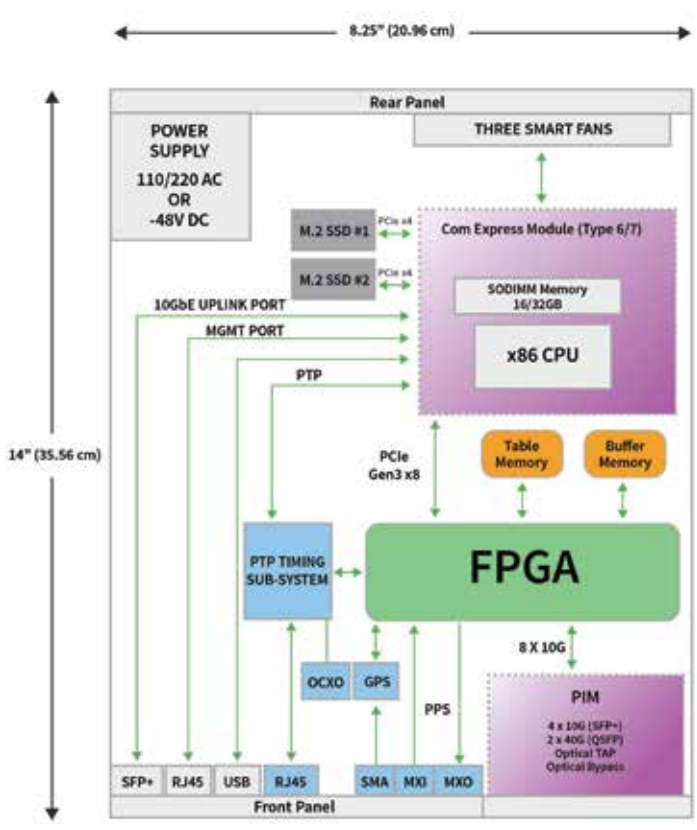


Figure 2
Accolade Technology ATLAS-1000 Architecture

About the author

Ameet Dhillon is the Director of Business Development at Accolade Technology and is based in Silicon Valley. He is responsible for evangelizing the company's FPGA-based, advanced packet capture adapters and acceleration platforms. He is an accomplished marketing and business development executive in the software and networking industries with over 25 years' experience.
www.accoladetechnology.com

Dive deep into the world of Military electronics and computing.

COTS Journal brings engineers and technical decision-makers the latest information on electronics and computers driving tomorrow's military technology. Get your FREE subscription now!

COTS JOURNAL

SUBSCRIBE @ www.cotsjournalonline.com/subscribe/

ADVERTISER INDEX

Company	Page	Website
Acrosser	20	www.acrosser.com
COTS Journal	53	www.cotsjournalonline.com
Critical IO	21	www.criticalio.com
DAC	7	www.dac.com
Dell	2	www.dell.com
Elma	4	www.elma.com
Gaia Converter	19	www.gaia-converter.com
Green Hills Software	9	www.ghs.com
High Assurance Systems	43	www.highassure.com
IBM	33	www.ibm.com
Micro Digital	39	www.smxrtos.com
NVIDIA	15	www.nvidia.com
One Stop Systems	47	www.onestopsystems.com
Pentek	56	www.pentek.com
Picmg	11	www.picmg.org
STMicroelectronics	49	www.st.com
Supermicro	29	www.supermicro.com
Teledyne Dalsa	55	www.teledynedalsa.com
TQ	45	www.embeddedmodules.net
WinSystems	35	www.winsystems.com

RTC (Issn#1092-1524) magazine is published monthly at 940 Calle Negocio, Ste. 230, San Clemente, CA 92673. Periodical postage paid at San Clemente and at additional mailing offices. POSTMASTER: Send address changes to RTC-Media, 940 Calle Negocio, Ste. 230, San Clemente, CA 92673.

Our newest innovation is cost.

The New Genie™ Nano. Better in every way that matters. Learn more about its **TurboDrive™** for GigE, Trigger-to-Image Reliability, its uncommon build quality... and its surprisingly low price.

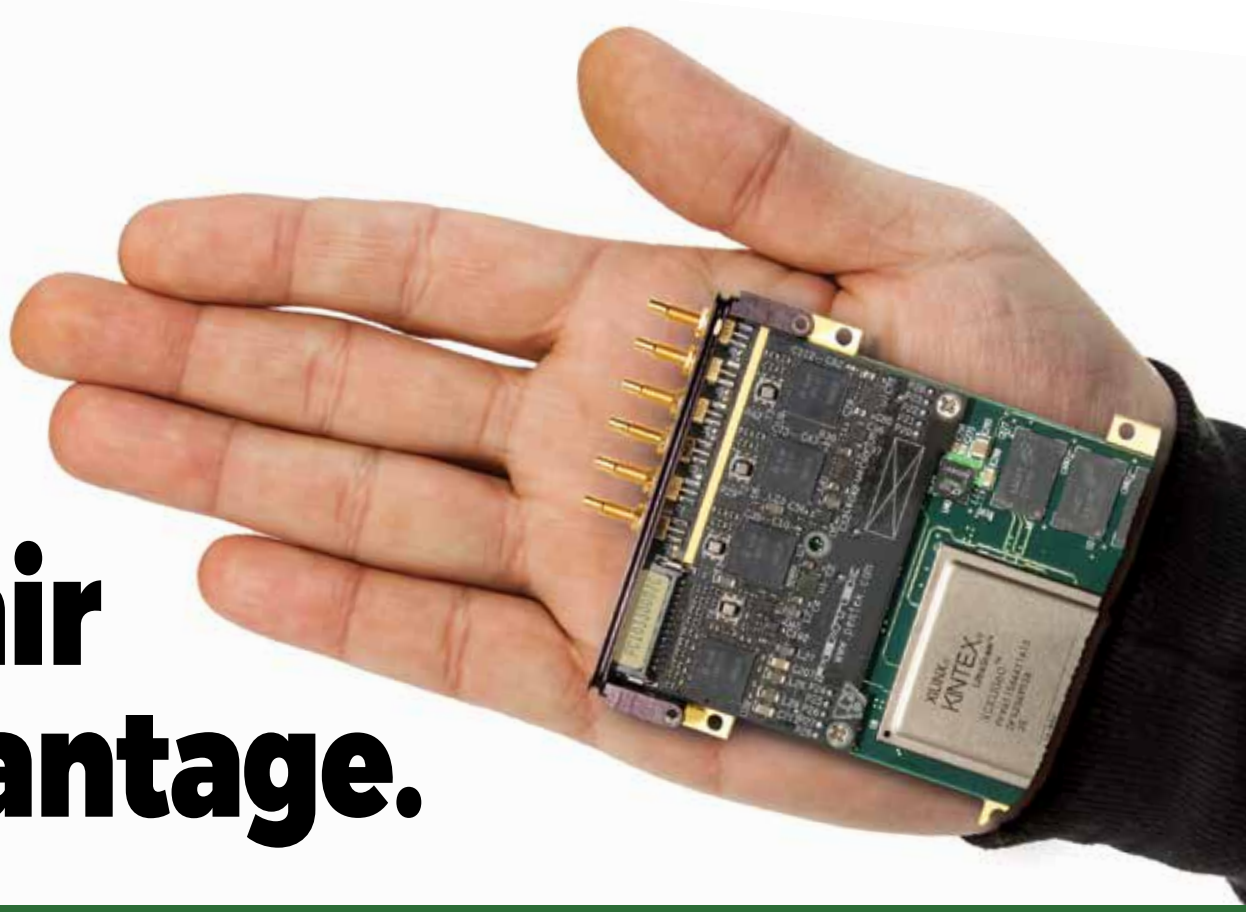
*starting at

$425 USD

*Taxes & shipping not included

» GET MORE GENIE NANO DETAILS AND DOWNLOADS:
www.teledynedalsa.com/genie-nano

TELEDYNE DALSA
Everywhereyoulook™



Unfair Advantage.

2X **HIGHER** performance | 4X **FASTER** development

Introducing Jade™ architecture and Navigator™ Design Suite, the next evolutionary standards in digital signal processing.

Pentek's new Jade architecture, based on the latest generation Xilinx® Kintex® Ultrascale™ FPGA, doubles the performance levels of previous products. Plus, Pentek's next generation Navigator FPGA Design Kit and BSP tool suite unleashes these resources to speed IP development and optimize applications.

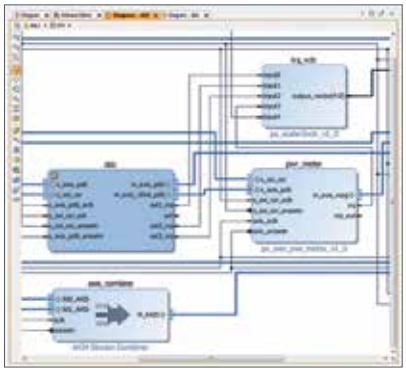
- **Streamlined Jade architecture** boosts performance, reduces power and lowers cost
- **Superior analog and digital I/O** handle multi-channel wideband signals with highest dynamic range
- **Built-in IP functions** for DDCs, DUCs, triggering, synchronization, DMA engines and more
- **Board resources** include PCIe Gen3 x8 interface, sample clock synthesizer and 5 GB DDR4 SDRAM
- **Navigator Design Suite** BSP and FPGA Design Kit (FDK) for Xilinx Vivado® IP Integrator expedite development
- **Applications** include wideband phased array systems, communications transceivers, radar transponders, SIGINT and ELINT monitoring and EW countermeasures

All this plus FREE lifetime applications support!

Jade Model 71861 XMC module, also available in VPX, PCIe, cPCI and AMC with rugged options.

JADE

Kintex Ultrascale FPGA



Navigator FDK shown in IP Integrator.

NAVIGATOR Design Suite

See the Video!
www.pentek.com/go/rtcjade or call 201-818-5900 for more information

PENTEK
Setting the Standard for Digital Signal Processing

Pentek, Inc., One Park Way, Upper Saddle River, NJ 07458
Phone: 201-818-5900 • Fax: 201-818-5904 • email: info@pentek.com • **www.pentek.com**

30 YEARS