# Every Cloud has a Silver Lining

## Industry Standards, Best Practices, and Recommendations for Embedded System Security

Joe Grand (@joegrand)

GRAND IDEA STUDIO

# Every Cloud has a Silver Lining

- Embedded Security Concepts
- Standards
- Best Practices
- Product/Vendor Resources

GRAND IDEA STUDIO

# Embedded Security Concepts

# Security Overview

- Security is a process
  - Constantly changing to reflect "state of the art"
  - The attacker usually has the advantage
  - Treat security as an integral part of system design, continue to evaluate during development and revisions
- Given enough time, resources, & motivation, an attacker can break any system
  - Reduce risk to an acceptable level
  - Costs of a successful attack should outweigh potential rewards

GRAND IDEA STUDIO

# Threat Model/Risk Analysis

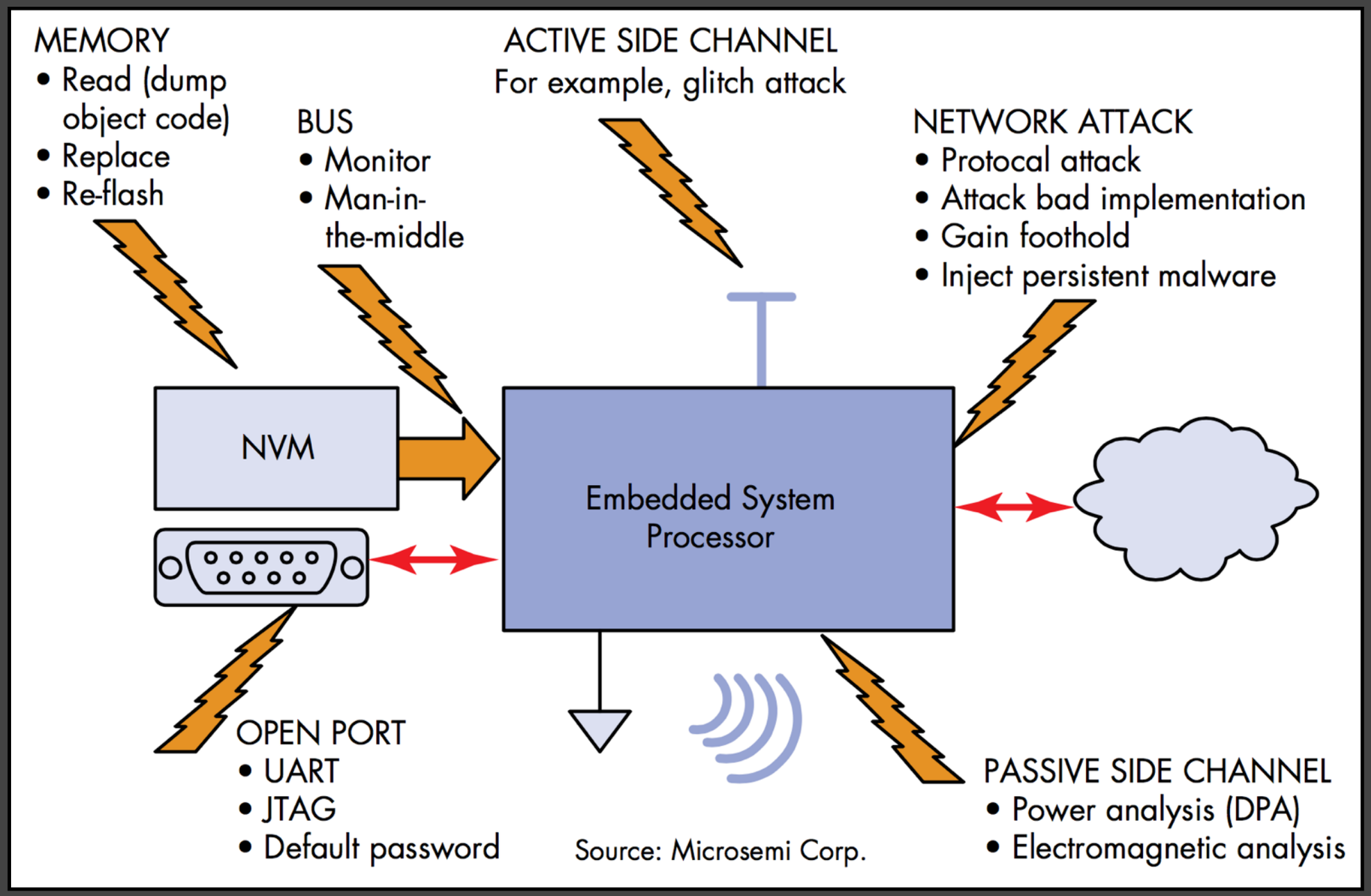
- You must understand your risk before you can protect yourself
  - What needs to be protected
  - Why it is being protected
  - Who you are protecting against (define your adversary)
- What features are absolutely necessary for system functionality?
  - Each new feature increases attack landscape
- Identify single points of failure across the lifecycle
  - Design, fabrication, integration/assembly, distribution, in-the-field

# Types of Hackers/Attackers

| Resource | Curious Hacker | Academic | Organized Crime | Government |
|---|---|---|---|---|
| Time | Limited | Moderate | Large | Large |
| Budget ($) | < $1000 | $10k - $100k | > $100k | Unknown |
| Creativity | Varies | High | Varies | Varies |
| Detectability | High | High | Low | Low |
| Target/Goal | Challenge | Publicity | Money | Varies |
| Number | Many | Moderate | Few | Unknown |
| Organized? | No | No | Yes | Yes |
| Release info? | Yes | Yes | Varies | No |

P. Kocher, Crypto Due Diligence, RSA Conference 2002

# Common Attack Surfaces




The Biggest Security Threats Facing Embedded Designers, Electronic Design, June 2016

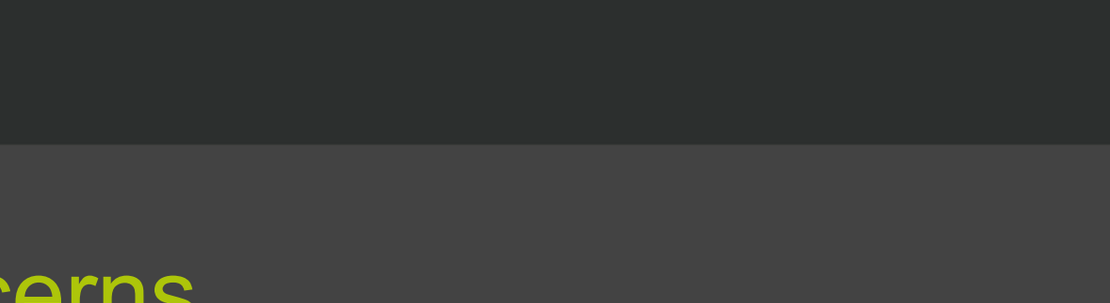
GRAND IDEA STUDIO

# Security Concerns

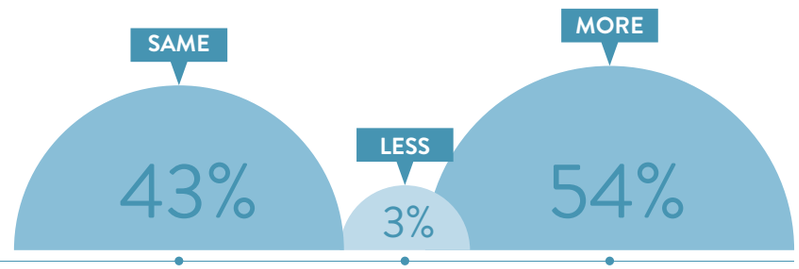
SALARY SURVEY 2015 ELECTRONIC DESIGN

## PREPARING FOR THE INTERNET OF THINGS

### Importance of security in products



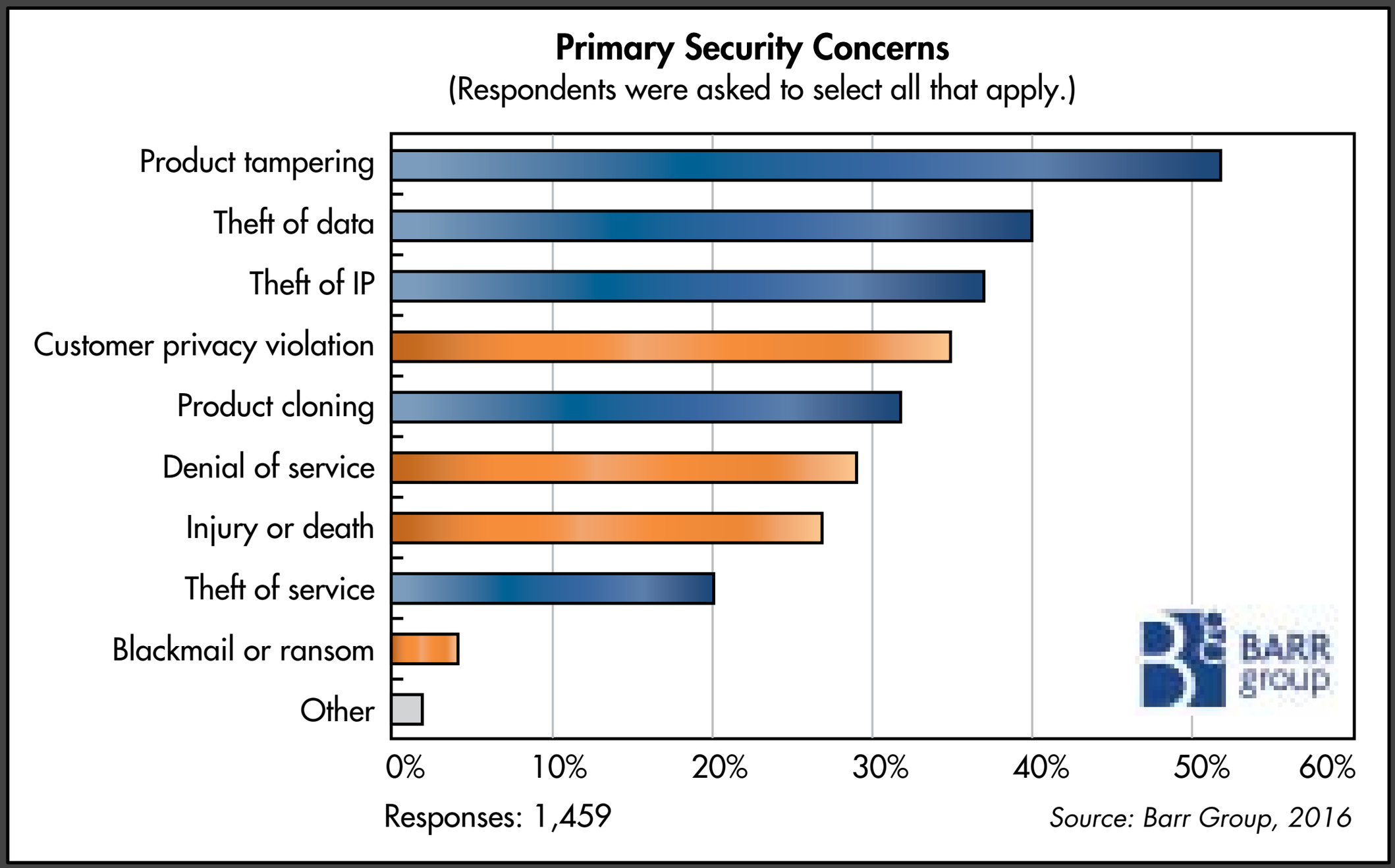

**How important willsecurity be in future products ?**




### Companies that will produce connected products

43%

# Security Concerns 2



# Easier Said Than Done

- Challenge of cost v. security v. convenience
- Implementation is product specific
  - No one-size-fits-all solution
- However, security solutions/techniques/resources becoming more accessible
  - Still requires some level of security competency
  - Be sure to independently verify what you're implementing

GRAND IDEA STUDIO

# Standards

# Standards

- Can be used as a checklist/starting point
  - Usually consists of *what* to do, not *how* to do it
- Some markets require full compliance to specific standards
  - Arguably a detriment to security if standard is too strict (e.g., only allow a specific process or encryption algorithm)
- Just because a device conforms doesn't make it impenetrable



Thinking inside the box: system-level failures of tamper proofing, UCAM-CL-TR-711, Feb. 2008

# Standards 2

- National Institute of Standards and Technology (NIST) Computer Security Resource Center
  - Guidelines/recommendations/references for many aspects of secure systems
  - SP 800: Computer Security
  - SP 1800: Cybersecurity Practice Guides
  - SP 500: Computer Systems Technology
  - http://csrc.nist.gov/publications/PubsSPs.html

# Standards 3

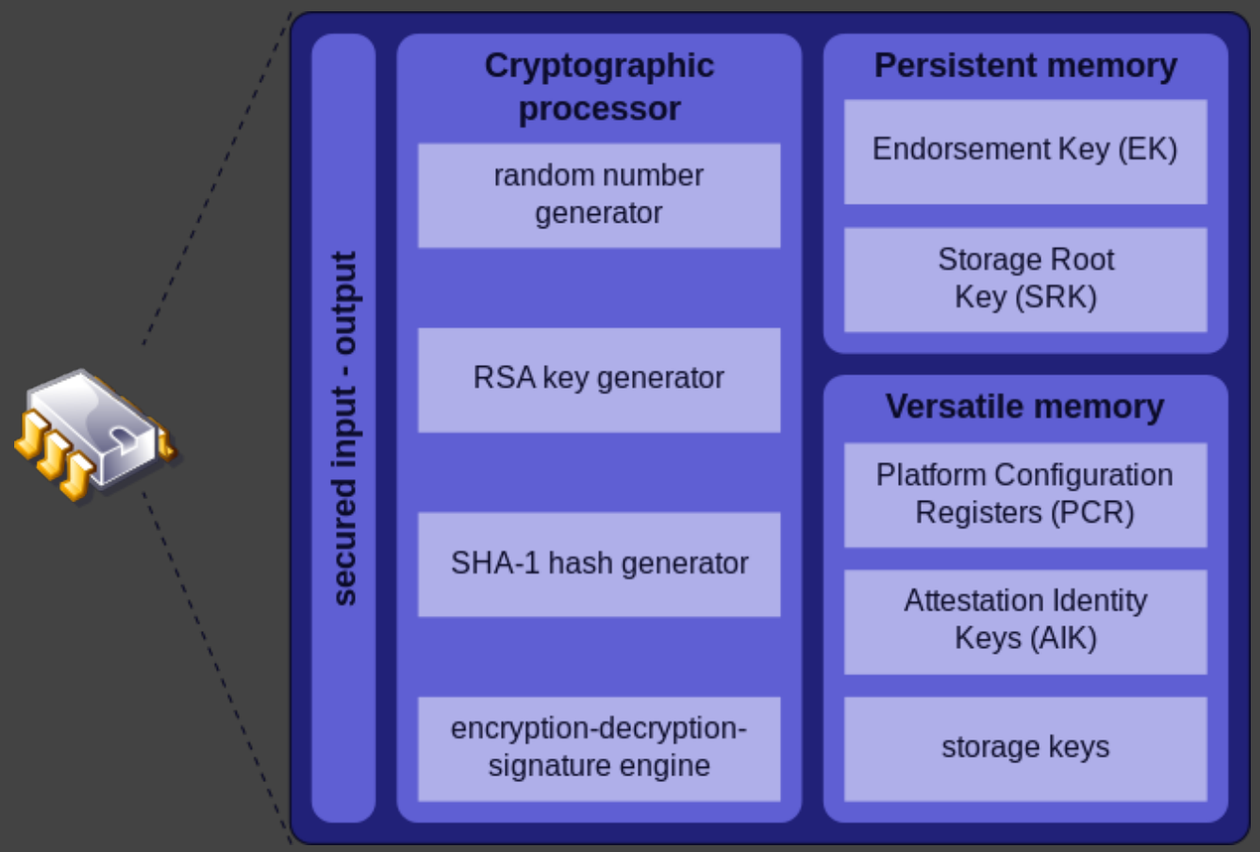
- Underwriters Laboratories Cybersecurity Assurance Program
  - UL 2900 Outline of Investigation for Software Cybersecurity for Network-Connectable Products
    - Part 1: General Requirements
    - Part 2-1: Healthcare Systems
    - Part 2-2: Industrial Control Systems
  - Standards only available for purchase
  - www.ul.com/cybersecurity/
  - https://standardscatalog.ul.com/standards/en/outline_2900-1_2

# Standards 4

- Federal Information Processing Standards (FIPS)
  - FIPS 140-2 Security Requirements for Cryptographic Modules
  - http://csrc.nist.gov/groups/STM/cmvp/standards.html
- Common Criteria
  - International standard for computer security certification (ISO/IEC 15408)
  - Verified by independent testing laboratories
  - www.commoncriteriaportal.org

# Standards 5

- Trusted Platform Module (TPM 1.2/2.0)
  - Standard/specification for secure cryptoprocessor
  - On-chip encryption/decryption/signing/key storage/RNG
  - https://en.wikipedia.org/wiki/Trusted_Platform_Module
  - www.trustedcomputinggroup.org/tpm-main-specification/



GRAND IDEA STUDIO

# Best Practices

GRAND IDEA STUDIO

# Best Practices

- Proper design principles can go a long way
  - If implemented correctly...
- Remove the low-hanging fruit
  - Increase difficulty of attack
- Strive for simplicity
  - Each security feature should support a defined goal

GRAND IDEA STUDIO

# Best Practices 2

- Avoiding the Top 10 Security Flaws
  - http://cybersecurity.ieee.org/center-for-secure-design/avoiding-the-top-10-security-flaws.html
- U.S. Dept. of Homeland Security (DHS) Strategic Principles for Securing the IoT
  - High-level guidelines/recommendations
  - www.dhs.gov/securingtheIoT
- Online Trust Alliance (OTA) IoT Trust Framework
  - Guidelines/recommendations for user privacy/security
  - https://otalliance.org/initiatives/internet-things

GRAND IDEA STUDIO

# Best Practices 3

- Global System for Mobile Communications Association (GSMA) IoT Security Guidelines
  - Guidelines/recommendations for endpoint devices/service providers/network operators
  - www.gsma.com/connectedliving/gsma-iot-security-guidelines-complete-document-set/
- Industrial Internet Security Framework (IISF)
  - www.iiconsortium.org/IISF.htm

# Best Practices 4

- FDA Cybersecurity
  - Principles/considerations for managing security in medical devices
  - Also involved in assessing security threats in released products
  - www.fda.gov/MedicalDevices/DigitalHealth/ucm373213.htm
- National Highway Traffic Safety Administration (NHTSA) Cybersecurity Best Practices for Modern Vehicles
  - Guidelines/recommendations for managing security in automotive electronic systems/communication networks/control algorithms
  - www.nhtsa.gov/Research/Crash-Avoidance/Automotive-Cybersecurity

# Best Practices 5

- Compartmentalization
  - Distribute design documentation on a need-to-know basis
  - Be aware of where/how documentation appears online (firmware update packages)
- Identifiers/Markings
  - Remove all non-necessary information
  - PCB silkscreen (designators, fab markings, logos)
  - Component/IC markings (part numbers, logos)
  - Increases reverse engineering time

# Best Practices 6

- Security Fuses
  - Prevents full read-out or access to a specific memory area
  - Most commonly used on MCU internal memory
  - Easy to enable during code compilation or device programming
  - May still be exploited via brute force, glitch, die attack, off-shore services
- On-Chip Debug/Program/Diagnostic Interfaces
  - Disable or remove completely for production units
  - Implement password/authentication mechanism (may not be part of standard interface)
  - Possibly inconvenient for legitimate personnel (manufacturing, service/repair)

# Best Practices 7

- Coding
  - Take care to handle undefined behavior, memory leaks, buffer overflows/bounds checking, invalid data structures, off-by-one, etc.
  - Remove debug symbols/tables, enable optimization
  - Mechanism to update/patch vulnerable code/OS (if needed)
  - Couple w/ source code review, static analysis
- Network Configuration
  - Don't use default login credentials (username/password)
  - Don't add backdoors for future use
  - Close unused ports/daemons/configuration/management interfaces
  - Learn about common network/OS exploits

# Best Practices 8

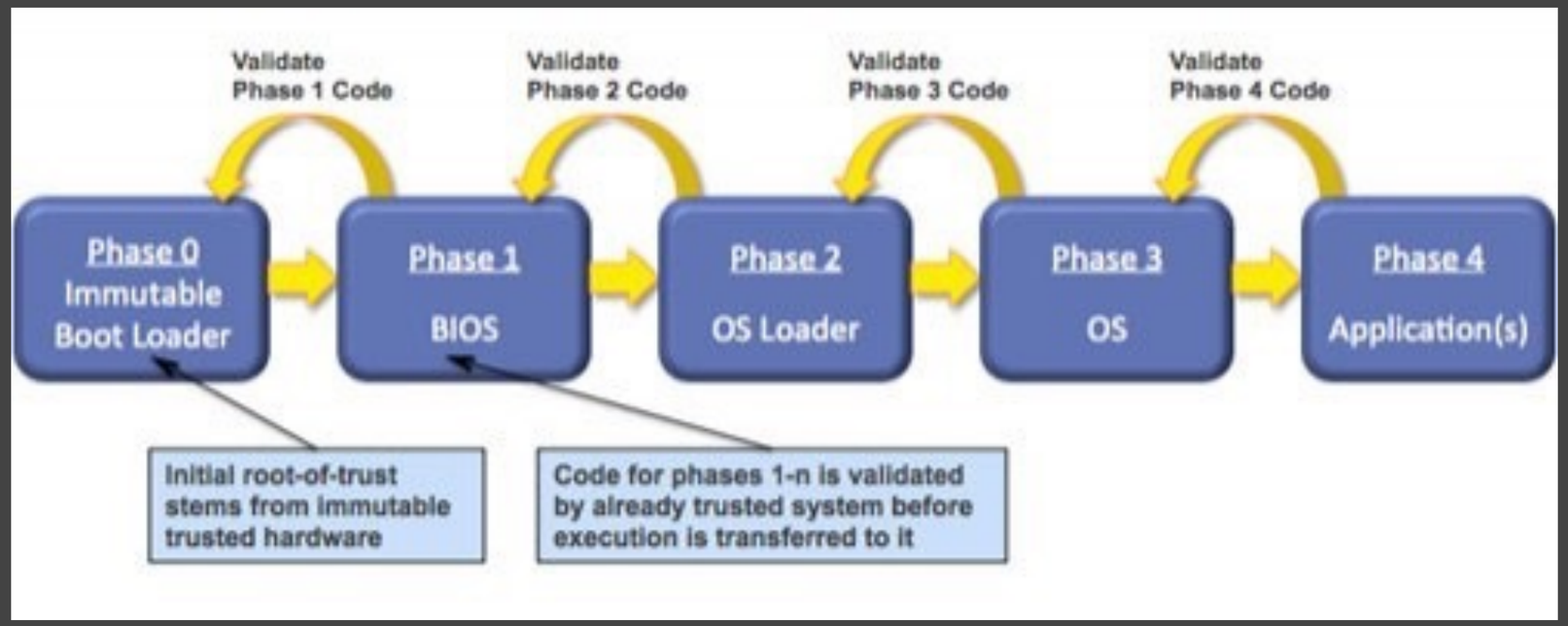
- Anti-Tamper
  - Prevent/deter/detect physical access or tampering of embedded system
  - Resistance, evidence, detection, response
  - See Physical Security Devices for Computer Subsystems: A Survey of Attacks and Defenses, Weingart, CHES 2000
- Run-Time Diagnostics/Failure Modes
  - Ensure device is fully operational at all times (watchdog, periodic system/memory checks)
  - Detect when system is being operated outside of defined conditions (voltage, timing, thermal, optical glitching)
  - Determine how product handles failure (halt/shutdown system, erase critical memory areas)

# Best Practices 9

- Encryption
  - For both data at rest and in motion (including firmware, if possible)
  - Consider key management/storage, cipher type
  - Many vendors offer on-chip support for encrypted memory areas
  - Beware of how unencrypted data could be accessed during operation (chip-to-chip communication, debug interface to RAM)
  - For wireless systems, use available security features (check if protocol has already been broken)
  - Use industry standard, publicly scrutinized/analyzed/proven ciphers
    - Don't roll your own!

# Best Practices 10

- Secure Boot/Authentication Process
  - Each stage verifies subsequent stage
  - Only execute trusted code (verified origin/integrity)
  - Prevents arbitrary code execution (unless defeated, commonly done via glitch or patch of hash compare)




Protecting networked designs from intrusion w/ secure FPGAs, Signal Processing Design, Oct. 2013

GRAND IDEA STUDIO

# Best Practices 11

- Side-Channel Prevention
  - Unintentional leakage from system
  - Consider power/EM, timing, thermal
  - See Rambus DPA Countermeasures
  - Many compilers generate side channels unintentionally

# Product/Vendor Resources

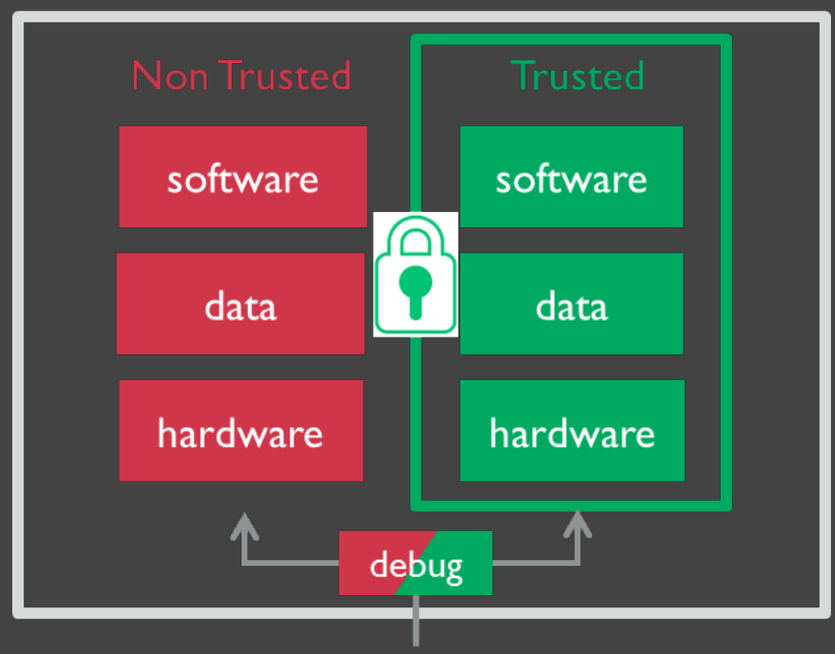
GRAND IDEA STUDIO

# Product/Vendor Resources

- No endorsement given!
- Evaluate before implementation
  - Some versions may already have been broken
    - Security Failures in Secure Devices, Tarnovsky, BH DC 2008
    - Hacking the Smartcard Chip (TPM), Tarnovsky, BH DC 2010
- Many vendors require NDA for data sheet
- Just a sampling of what's available for embedded systems

# Product/Vendor Resources 2

- Altera (Intel)
  - Secure programmable logic (FPGA, SoC)
  - Root key storage, encrypted bitstream, glitch protection, HW crypto
  - www.altera.com/solutions/technology/security/overview.html
- ARM TrustZone
  - Security extensions/kernel added to ARM architecture
  - Open source reference implementation
  - www.arm.com/products/security-on-arm/trustzone



# Product/Vendor Resources 3

- Atmel
  - CryptoAuthentication, TPM, CryptoRF, CryptoMemory
  - www.atmel.com/products/security-ics/
- Broadcom
  - Secure Applications Processors (ARM + TPM)
  - BCM5880, BCM5882, BCM5892, BCM5830x family
- Cypress
  - Secure MCUs/PSoC (HW crypto, WiFi security features)
  - SecureNAND Flash Memory (Block protection capabilities)

# Product/Vendor Resources 4

- Infineon
  - OPTIGA family (Trust, TPM, Mobile)
  - Authentication, secure MCUs
- Maxim
  - Authentication, secure MCUs (DeepCover), secure memory/ managers
  - IoT Embedded Security Reference Design (MAXREFDES143)
  - www.maximintegrated.com/en/products/digital/embedded-security.html
- Microchip
  - CEC1302 Crypto Embedded Controller (ARM Cortex-M4)
  - PIC Microcontrollers w/ Cryptographic Engines
  - www.microchip.com/design-centers/embedded-security

# Product/Vendor Resources 5

- Microsemi
  - Secure FPGAs (root of trust, on-chip cryptographic support)
  - SmartFusion2 SoC (ARM Cortex-M3), IGLOO2
  - www.microsemi.com/products/fpga-soc/security
- NXP (Freescale)
  - Kinetis K8x Secure MCU family (ARM Cortex-M4)
  - On-the-fly AES decryption/execution from external Flash, boot ROM for encrypted FW updates, HW crypto, tamper detection (temperature, voltage, clock)

# Product/Vendor Resources 6

- Qualcomm
  - Snapdragon/SecureMSM
  - Secure boot, trusted execution environment, HW crypto, authentication
  - www.qualcomm.com/products/snapdragon/security
- Renesas
  - Secure MCUs (RS-4, AE-5)
  - www.renesas.com/en-us/products/secure-mcus.html
- Samsung
  - UFS (Universal Flash Storage) Shark SED
  - Secure NAND Flash w/ on-the-fly encryption/decryption
  - http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140sp/140sp2380.pdf

# Product/Vendor Resources 7

- STMicroelectronics
  - ST23, ST31, ST32, ST33 Secure MCU families
  - STSAFE-A authentication
  - www.st.com/en/secure-mcus.html
- Texas Instruments
  - Secure MCUs, HW crypto, protected memory regions
  - www.ti.com/ww/en/embedded/security/
  - Secure FW update reference design
  - www.ti.com/lit/wp/slay041/slay041.pdf

# Product/Vendor Resources 8

- Xilinx
  - Secure programmable logic (FPGA)
  - Root key storage, encrypted bitstream, HW crypto, anti-tamper, DPA countermeasures
  - www.xilinx.com/products/technology/design-security.html
- Zilog (IXYS)
  - eZ80F91 MCU w/ TCP/IP stack & embedded firewall (ZGATE)
  - www.zilog.com/ZGATE

# Product/Vendor Resources 9

- CHIPSEC: Platform Security Assessment Framework
  - Test suite for analyzing security of PC platforms (HW, system firmware, platform components)
  - https://github.com/chipsec/chipsec
- SparkFun CryptoShield
  - Open source hardware security reference/experimentation shield for Arduino and compatible
  - Real-time clock, TPM, encrypted EEPROM, authentication chips
  - www.sparkfun.com/products/13183
- CrypTech Alpha
  - Open source Hardware Security Module (HSM) reference design
  - Cryptographic engine and key storage (ARM + FPGA)
  - https://cryptech.is/

# Product/Vendor Resources 10

- Cryptography Research (part of Rambus)
  - CryptoManager Security Platform
  - HW/SW root-of-trust, secure provisioning/key storage/authentication solutions
  - www.rambus.com/security/cryptomanager-platform/
- Armored Things
  - Cloud-centric endpoint and management layers for IoT devices
  - Controlled data sharing, attack prevention, adoptable settings/rapid enrollment
  - Co-founded by Joe Grand, Marcus Ranum, Charles Curran
  - In development!

GRAND IDEA STUDIO

# What Now?

- Learn from history/prior attacks
- Enable security by default (if possible)
  - Don't rely on the customer to make the product secure
- Proactive security means safer products for all
  - Invest in proper design from the beginning
  - Allocate time for white/black box product security analysis/testing
  - Bug bounty programs, accept/reward outside discoveries
  - Don't let your product be the launching point for larger attack

GRAND IDEA STUDIO

# Think Like an Attacker

GRAND IDEA STUDIO

Thanks for watching!