

Connected Embedded Systems- Security Vulnerabilities, Attacks and Countermeasures



Abstract

Embedded Systems have been a driving force for making human lives easier in many domains, such as consumer electronics, automotive, healthcare, and industrial automation controls. Until not very long ago, most of these devices worked in Radio Frequency (RF) networks having a short range or in standalone node. In most of the cases, connectivity was limited to closed premise wired networks and the devices comprising of the controls systems did not have any exposure to the public networks. The advent of the internet brought massive and phenomenal changes in the way we interact and leverage technology in our day to day life.

It was next to impossible that embedded devices will stay untouched with the internet and connected ecosystem-based revolution. With the advent of cloud and IOT (internet of the Things), it was inevitable for the embedded device to leverage these technology advancements and touch the human lives in a more pervasive way. In the post-PC-Era and in this new mini and micro devices era, more and more embedded and computational systems are getting connected and getting exposed to the public networks. But on the flip side, connected and networked devices brings their own risks despite influencing the human race in big positive way. The purpose of this paper is not to discuss all the risks which connectivity brings in the equation. The paper solely to focuses on the cyber security attacks and risks which have become the issues of big focus lately.

TABLE OF CONTENTS

Overview	5
Basic Security Properties for the connected embedded Systems.	6
The cost of cyber-attacks	8
Assets, Threats, Vulnerabilities and Countermeasures	9
Classification of Vulnerabilities	10
Major Attacks	12
Remediation to the attacks	16
Short description of countermeasures and process recommendations	18
Summary	21
Conclusion	22
References	22

Abbreviations

CVE	Common Vulnerabilities and Exposures
CAPEC	Common Attack Pattern Enumeration and Classification
AVOIDIT	attack vector, operational impact, defense, information impact, and target
SCADA	Supervisory Control and Data Acquisition
	Deep Learning
IOT	Internet of Things
RF	Radio Frequency
CIA	Confidentiality, Integrity, Availability
MITM	Man in the Middle Attack
HIPPA	Health Insurance Portability and Accountability Act
PCI	Payment Card Industry
IEC	International Electrotechnical Commission
ISO	International organization for standards
PLC	Programmable Logic Controller
DoS	Denial of Service
DDoS	Distributed Denial of Service

Overview

Security is an important issue for the connected systems because these systems are the driving force behind many mission and safety-critical systems. The attacks on these systems do not result only into physical, financial, and reputational losses for the organization but also loss of human lives.

Nowadays, medical implants are also connected to some kind of external systems, and any kind of misconfiguration or tampering with the data might result in the precious loss of human lives. In the past, a lot of focus of the organization has been on protecting only their peripheral networks.

The Organizations involved in financial transactions were exceptions, and they invested heavily in cybersecurity because of the financial risks involved and stringent regulatory standards. The overall security for connected embedded systems in the other domains like healthcare and industrial automation domains becomes more important even than because of the risks to human lives. The regulatory compliance like HIPAA, PCI, and ISO/IEC standards for different business verticals are making it mandatory for the organization not to ignore cyber security risk because of huge regulatory financial penalties. Lately, there has been a lot of focus on securing connected embedded systems because of regulatory compliances and obvious risk to all kinds of assets, and more importantly, human lives.

According to the Barr Group 2018 Embedded Systems Safety & Security Survey that approximately 25% of all new products with internet connections could create hazardous effects for humans in case of being attacked or failure [1]

But a secure design of such systems vis-a-vis traditional IT systems come with complex and different challenges like resource constraints, design complexities, and difficulty in implementation, as well as expensive security controls.

This paper aims to get into high level overview of the existing threats and vulnerabilities based on the information available in the public domain. These findings can help design secure connected systems which fulfil the basic and essential properties CIA (confidentiality, Integrity, Availability) of a secure system.

Basic Security Properties for the connected embedded Systems

These 3 properties (CIA) in combination are called the security triad for a computational system.

The listed security properties are considered the most important and basic tenets of information security for any computational system.

Confidentiality

This deals with the authorization when it comes to access to any kind of data in the system. The proper authentication and authorization must be implemented in the system, and the proper security controls controlling the access to and kind of data or asset should be in place.

Since all the systems are connected nowadays, threats and vulnerabilities in the communication protocol compromising confidentiality should also be paid critical attention.

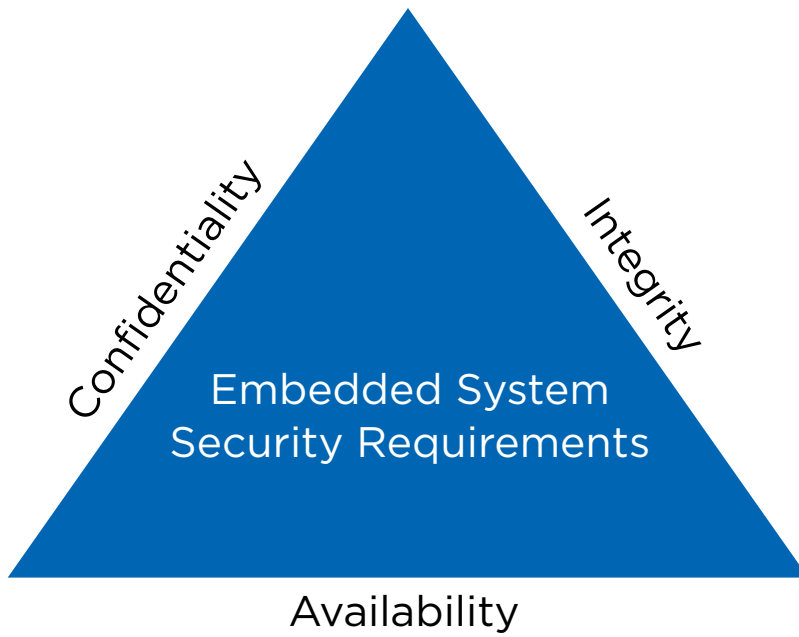
Integrity

No tampering or modification of the data by an external and non-authorized actor should be allowed. The security controls should make sure that the hackers do not have any kind of access to any kind of data, 'data in transit' as well as 'data at rest'.

Any kind of unauthorized modification in data will result in an undesirable functionality of the system. A very common use case can be sensitive information, like credit card data travelling over the wire. The exposure of this kind of data might result in financial as well as reputational loss for the organization.

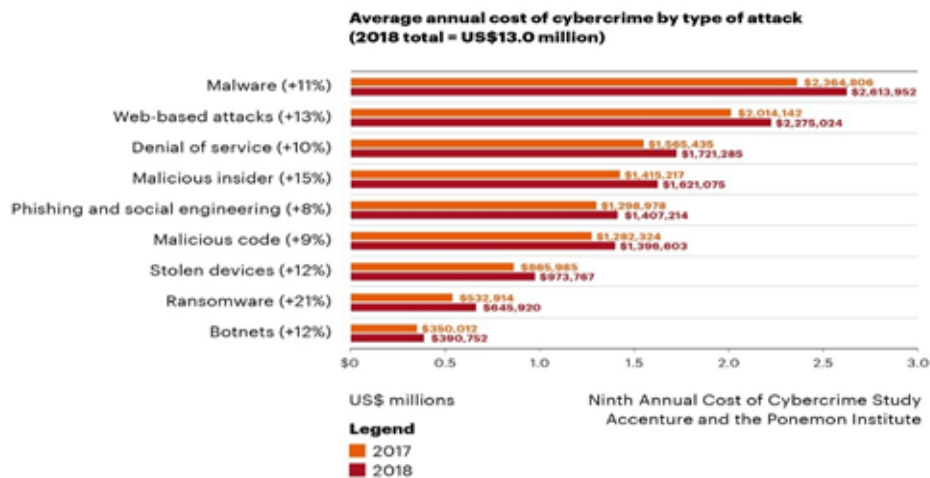
Availability

It is one of the critical and key necessities of any kind of embedded system. It means that the system should be available when it is needed by the user. In the healthcare domain, offline safety critical systems can put human life in danger. Devices like pacemaker are expected to be available all the time. The offline E-commerce website would result in revenue and customer loss for the organization.



The cost of cyber-attacks

Before we delve further into types of cyber-attacks, we should pay attention to the cost of cybercrime for the latest years. That will help us appreciate why cyber security has become so important. The following infographics has been taken from a report compiled by Accenture and Ponemon institute. As per this report the average cost of cybercrime per organization had gone up to USD 13M in 2018 [2]



Graphics : www.businesswire.com[3]

As per the findings:

In 2018, surveyed companies each recorded an average of 145 cyberattacks — resulting in the infiltration of a company’s core networks or enterprise systems — an 11 percent increase over 2017 and 67 percent higher than five years ago.

Malware is the most expensive type of attack, costing companies US\$2.6 million, on average, followed by web-based attacks, at US\$2.3 million.

The number of organizations experiencing ransomware attacks increased by 15 percent in 2018, with the costs increasing 21 percent, to approximately US\$650,000 per company, on average. The number of ransomware attacks more than tripled in the past two years.

Assets, Threats, Vulnerabilities, and Countermeasures

As already mentioned, this paper tries to list the consolidated information about cyber threats, vulnerabilities, and counter measures based on the distributed information available in the public domain. One of the commonly followed vulnerabilities database is CVE (<https://cve.mitre.org/>). As per CVE site -

CVE® is a list of entries—each containing an identification number, a description, and at least one public reference—for publicly known cybersecurity vulnerabilities.

CVE Entries are used in numerous cybersecurity products and services from around the world,

including the U.S. National Vulnerability Database (NVD)

Before we delve more into the attacks on embedded connected systems, we should define the target asset surfaces of the in-scope connected computational systems on a broad level.

S.no.	Target	Henceforth referred to as
1	Hardware/Silicone	HW/SC
2	Firmware	FW
3	Operating Systems	OS
4	Applications	App
5	Web based Applications/Cloud applications	Web/Cloud
6	Communication interfaces (Wired, SPI, Wireless etc.)	Comm
7	Combination for more than two above	Device

Most of the times, more than two asset surfaces get targeted, so securing only one or two assets cannot be enough.

Classification of Vulnerabilities

We are going to classify the major common vulnerabilities types which can be exploited by attack vectors as per the publicly available vulnerabilities databases. Most of the vulnerabilities are going to fall into one of the below listed categories. Please note this classification is more on a macro level, and this paper does not claim to cover all kinds of vulnerabilities in the below listed classification.

Ignorance to the best programming practices:

Most of the systems do not follow the best programming practices. The kind of programming errors might include ignorance of input/output validation, improper dynamic memory management, and use of deprecated functions. This may result into attacks like command injection attacks, buffer overflow attack, and SQL injection besides others.

Implementation of Unsecure communication channel:

This class pertains to not applying enough security to 'Data in Motion.' It means that the communication devices and protocols carrying the information are not properly secured.

Unsecured device lifecycle management:

Other vulnerabilities may result because of not following the secure lifecycle management practices during the lifecycle of the device. Unsecure S/W updates and lack of proper monitoring are some of the types of these kinds of vulnerabilities. The non-secure S/W updates may result in unauthenticated access to the devices or even the control of the devices.

Web/Cloud-based vulnerabilities:

Nowadays, connected embedded devices either have a web-based control interface and or these devices communicate with public clouds for many purposes like data storage and data visualization etc. This exposes the device or its part to all the vulnerabilities as a particular web or cloud application would have. Some of the examples are cross-site scripting and broken authentication [4].

Weak authentication:

Sometimes, passwords are hardcoded, or strong password rules are not enforced, and this makes those devices susceptible to brute-force kind of attacks.

Hardware-based vulnerabilities:

The lapses in the security controls of the hardware in the device allows an attacker to exploit the system through remote or physical access to the system. These kinds of vulnerabilities allow some common attacks like CPU side channel attack and DRAM memory Rowhammer attacks. [5]

Weak cryptography implementation:

Sometimes 'fit for the purpose' cryptography algorithms are not used. Other cases might be the storage of keys is not in a secure place. This is going to result in both weak authentication and communication attacks if someone can get hold of the cryptographic keys.

Internal sabotage:

These are quite abstract, and all kind of vulnerabilities may fall into it. A disgruntled employee having all the access to the system can compromise the security of the system, or make it vulnerable to the external attacks.

Major Attacks

The following table lists the major attacks which can exploit the vulnerabilities in the system.

S No	Attack	Target Surface	Class of Vulnerability Exploited	Affected Security tenets
1	Control hijacking attacks	OS/FW/SW	Ignorance to the best programming practices.	CIA
2	Reverse Engineering	FW/SW	Unsecured Device life cycle management	CIA
3	Malware	FW/SW/OS	Web/Cloud based vulnerabilities.	CIA
4	Injecting crafted packets or input	Comm/SW/OS	Ignorance to the best programming practices, Implementation of Unsecure communication channel	CIA
5	Eavesdropping	Comm	Implementation of Unsecure communication channel	CIA
6	Brute Force Search attack	OS/FW/SW	Weak Authentication	CIA
7	DNS poisoning	Comm	Implementation of Unsecure communication channel	CIA
8	Man in the Middle Attack	Comm	Implementation of Unsecure communication channel	CIA
9	DoS (Denial of Service Attack)/ DDos	Comm/OS	Multiple	A

10	Silicone Memory Attack	SC/HW/FW	Ignorance to the best programming practices	CIA
11	Cold Boot Attack	HW/SC	Hardware based vulnerabilities	CIA
13	Invasive Physical Attacks	HW/SC	Hardware based vulnerabilities	CIA
14	Non Invasive Physical Attacks	HW/SC	Hardware based vulnerabilities	CIA
15	Protocol Attack	Comm	Weak cryptography implementation:	CIA
16	Normal use	Device	Internal Sabotage	CIA
17	Session High jacking	Comm	Implementation of Unsecure communication channel	CIA

The following section tries to elaborate more on the attacks:

Control hijacking attacks:

In this type of attack, the hackers try to take over the target device by executing arbitrary code on target by hijacking control flow. Some of the examples are Buffer overflow attacks, integer overflow attacks, and format string vulnerabilities.

DNS Poisoning:

The attacker may corrupt the local Domain Name System (DNS) server. As a result of the same, legitimate data will be directed to the attacker's website for misuse. The attacker may host a fake web site to steal the credentials of the legitimate users.

Reverse engineering:

By reverse engineering the Software code on the device, the attacker can find sensitive information stored, any programming loophole, and get hold of cryptographic keys. This helps him find other vulnerabilities also which can be used to execute other kind of attacks. This is one kind of social engineering attack.

Malware:

The target device can be attacked by injecting other kind of potentially harmful and unwanted software. This may result in the modification of the behavior of the system. The infamous Stuxnet attack is one of the examples of this kind of attack [6]. In this attack, the malware reprogrammed a PLC in a nuclear facility. That led to the physical destruction of the centrifuges controlled by the infected PLCs.

Injecting crafted packets or input:

Such an attack involves crafted packet injection in communication protocol or the tampering with the inputs to a device software. to so that injection of crafted packets is an attack method against protocols used by embedded devices. Both of these types of attacks can result in communication failure or a change in application behavior .

Eavesdropping:

This is one kind of passive attack where a hacker observes the data passing through a communication channel. If there is no proper and secure enough encryption implemented, the hackers can read the sensitive information passing through the communication channel.

Brute-force search attacks:

These attacks exploit vulnerabilities like weak passwords and weak cryptography. In this type of attack, the attacker tries different combinations of user names and passwords, until he gets the right combination to get into the device. Once he gets into the device, he can do all sort of malicious activities.

Normal use by employees:

This attack is executed on the device by people with malicious intentions or the internal people (poorly trained) who have complete access to the systems. The disgruntled internal employees may have mala fide intentions and may manipulate systems to expose the systems to the external attacks.

Man in the Middle Attack:

In cryptography and computer security, a man-in-the-middle attack (MITM) is an attack where the attacker secretly relays and possibly alters the communications between two parties who believe that they are directly communicating with each other [7]

DoS (Denial of Service):

A DoS (Denial of Service) attack affects the availability of the system, thus, making it unavailable for functional use to legitimate users. This can be executed by exhausting the capacity of the data carrying components or by exhausting the computing resources on the target systems. DDos is another such kind of attack.

Silicone Memory Attack:

In case of physically and insufficiently secure hardware, it may be possible to read the contents of memory. RAM and ROM are both susceptible to this kind of attack. To avoid this, all the static code and data residing as a part of firmware should be encrypted using some kind of mechanism.

Cold Boot Attack:

In computer security, a cold boot attack (or to a lesser extent, a platform reset attack) is a type of side channel attack in which an attacker with physical access to a computer performs a memory dump of a computer's random access memory by performing a hard reset of the

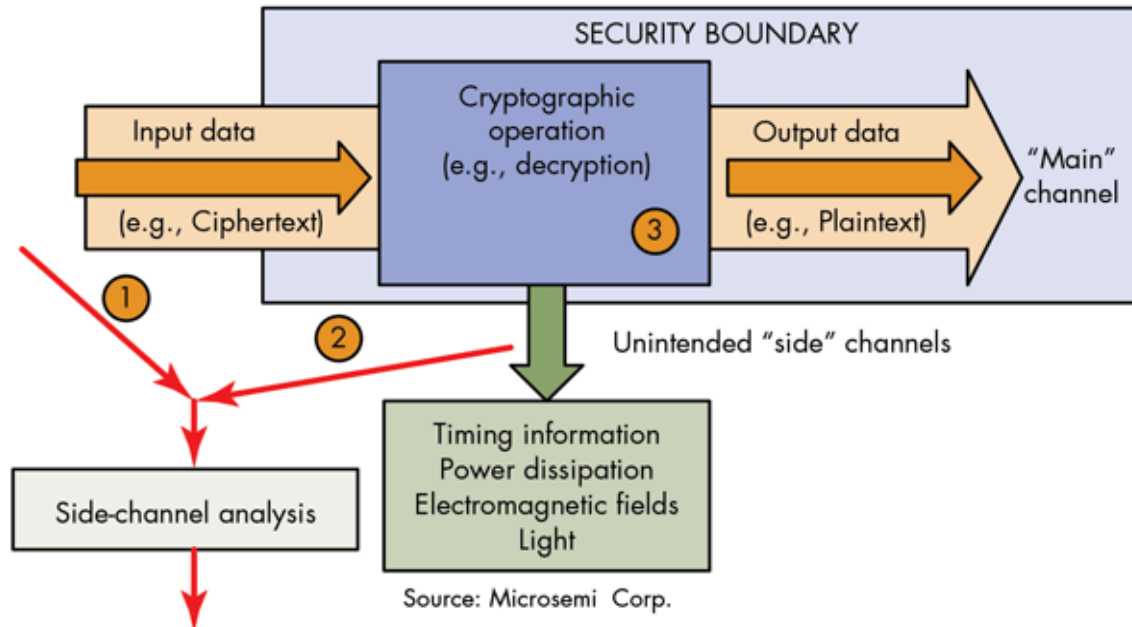
target machine. Typically, cold boot attacks are used to retrieve encryption keys from a running operating system for malicious or criminal investigative reasons. The attack relies on the 'data remanence'[8] property of DRAM and SRAM to retrieve memory contents that remain readable in the seconds to minutes after power has been removed. [9]

Invasive physical attacks:

Attackers need physical access to the device to execute these kinds of attacks. These are hardware and Silicone-based attacks. Attackers might cut open the chip packaging and examine the internal contents with micro-probing techniques. These may be used to eavesdrop the communication between components and read the security-sensitive information.

Non-invasive physical attacks:

These attacks are also known as side channel attacks. The active side channel attack may be one type of glitch attack. One of the examples is the infamous attack on the Sony PlayStation. This attack and resultant chain of attacks cost Sony almost \$1 billion [10]. This attack used voltage glitches on the power supply of the processor resulting in a malfunction at a critical juncture in the execution of the embedded program. It allowed the attacker to enter a privileged processor state, enabling him to dump all the code. The attacker found a cryptographic implementation flaw, and that is what he needed to execute his malicious intentions. A passive type of side channel attack is depicted in the graphics on the next page. Without going too much into the details of differential and simple power analysis inside a processor or FPGA may be exploited as side channel attacks if proper security counter measures are not implemented.

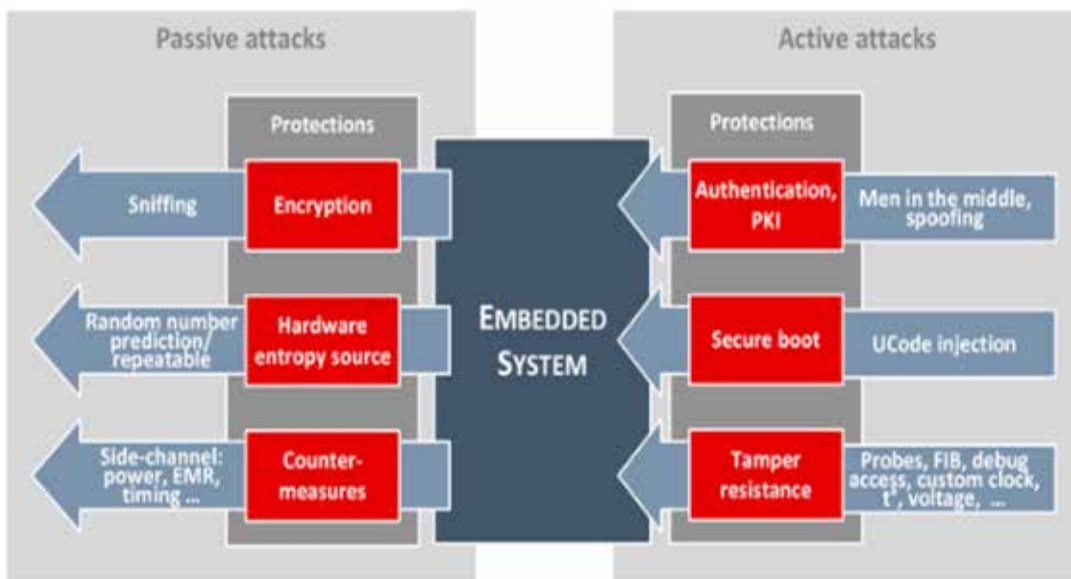


Session Hijacking:

In computer science, session hijacking, sometimes also known as cookie hijacking, is the exploitation of a valid computer session—sometimes also called a session key—to gain unauthorized access to information or services in a computer system.[11][12]

Remediation to the attacks

On a broader level, the following graphic depicts the types of attacks and counter measures for a connected embedded system using hardware based solutions. Please see the reference article [13] for complete details. Hardware based solutions are hard to crack and much more robust.



Graphics: <https://www.chipestimate.com> [13]

The Security controls can vary depending upon the type of embedded system being used. The type of attack may also vary according to the device lifecycle stage, environment, and type of users. The industry vertical and device lifecycle phase-specific controls provide strong countermeasures for that particular device. The following table lists the well-known attacks and possible countermeasure.

S No	Attack	Counter Measures
1	Control hijacking attacks	Best programming practice. E.g. MISRA coding guidelines for C/C++ [14] Buffer overflow protection
2	Reverse Engineering	Obfuscation of strings, resources, entry points, and memory. Encryption using H/W based crypto systems.

3	Malware	Configurable firewall for embedded systems [16] Machine-learning-based Anti-Malware protections
4	Injecting crafted packets or input	IPsec [21] Program input and output validation as per defined design. Best programming practices.
5	Eavesdropping	IPsec [21] SSL, SSH, TLS Solutions
6	Brute Force Search attack	Limiting the number of attempts and increasing the period before the next attempt to authenticate can be made.
7	DNS poisoning	DNSSEC [22]
8	Man in the middle attack	IPsec [21]. SSL, SSH, TLS Solutions.
9	DoS (Denial of Service Attack)/DDoS	IDS (Intrusion Detention Systems) [23]. IPS (Intrusion Protection System) [24]. Distributed Denial of Service Attacks: Four Best Practices for Prevention and Response [25].
10	Silicone Memory Attack	HW/SW Defender technique [20].
11	Cold boot attacks	Restricting Physical access to the system. Full memory encryption. Secure erasure of memory when not in use.
12	Invasive Physical Attacks	Secure decommissioning, Restricted physical access. Use Biometric authentication.
13	Non Invasive Physical Attacks	Masking [28] Random clock technique[5]
14	Protocol Attack	Shielding techniques [26] Asynchronism [26]
15	Normal use	Security awareness for users. At least two users should authenticate, to avoid the possibility of a disgruntled user breaking into the critical system.
17	Session High jacking	Encryption

Short description of countermeasures and process recommendations

In this section, this paper tries to recommend the remediation controls and best practices in a clear and concise form to the common threats.

Secure Boots

Secure boot is a security standard to help make sure that a device boots using only software that is trusted by the Original Equipment Manufacturer (OEM). No one should be able to reprogram or bypass these instructions. It is always recommended to build their trust in hardware roots in the form of ROM (Read Only Memory) embedded code/security certificate and non-accessible private keys. There are H/W based external secure solutions, which provide best in class anti-tamper techniques and cryptographic mechanisms in the form of chips.

Secure Update

Software and firmware should be updated using the binaries only with trusted signatures that only the device S/W developers or owners can produce. This often involves 3rd party CAs (certificate authorities) issuing the certificates and private keys. Any means delivering the S/W to update the software should be highly secured and trusted.

Monitoring and self-healing

The secure system should be accompanied by preferably H/W based solutions capable of detecting attacks, raising the immediate alerts, healing themselves, and upgrading themselves to keep pace with every new threat reported with the passage of time. If H/W based solution cannot itself provide these solutions, then we should explore S/W based solutions. Sometimes, small embedded devices may not have resources to accomplish this. In such a case, a balanced trade-off between the cost of resources and minimum required security needs to be made.

Always adopt the best practice for coding

The following 3 time-tested practices can help you develop a more secure device. These are not expensive and do not involve any security experts. The developers themselves can make it a habit of their day to day coding work.

- Use the best and approved (if available) coding standards.
- Regular code reviews.
- Static analysis of the code using well-known and industry proven tool.

Defense in Depth

Wikipedia defines the “Defense in Depth” [18] as follows -

Defense in depth (also known as Castle Approach[20]) is an information assurance (IA) concept in which multiple layers of security controls (defense) are placed throughout an information technology (IT) system. Its intent is to provide redundancy in the event a security control fails, or a vulnerability is exploited that can cover aspects of personnel, procedural, technical, and physical security for the duration of the system’s life cycle. The idea behind the defense in depth approach is to defend a system against any particular attack using several independent methods. It means that failure of one or more layer does not allow the intruder access to the system until he has broken through all the layers.

Continuous Education about threats and Remediation:

The security expert needs to keep himself updated about the latest threats and security controls implemented about that.

Reduce Attack Surface:

Software architecture and security controls around that architecture go hand in hand. It is always recommended to keep the system architecture small and as much as abstract as possible to reduce the attack surface. It is not always easy to do so, but it is not a good practice from a security perspective to add configuration options to change the behavior of the system at the runtime. If we cannot design a small and least exposed system, then it is better to divide the same into manageable, simple systems and apply security controls. While doing so, we should always follow ‘least privilege’ and ‘segregation of privileges’ principles.

Prefer H/W based security solutions:

Software-based security controls are relatively cost-effective and easy to update. But Software-based solutions are much more vulnerable to attacks like malware etc. Hardware-based security controls are immutable and integrate specialized functionality not to be tempered with. These controls can protect against side channel analysis and other threats. Hardware based solutions also address the limitations of resource constraints of the software-based solution when it comes to complex cryptographic computations. The only drawback of hardware-based solution is that these solutions tend to be expensive.

Secure Storage:

Secure data storage encompasses the following:

- Encrypt personal data
- Avoid silent data corruption - Data corruption refers to errors in computer data that occur during writing, reading, storage, transmission, or processing, which introduce unintended changes to the original data [25]
- Safe disposal of the devices and prevent of deleted data.

RAID (Redundant Array of Inexpensive Disks) systems of solid state drives, with integrated security based on SLC (Single Level cell) flash memory technology can be one of the solution. However, the trade-off amongst security, resources, and financial cost is required.

Safe Access:

This directly pertains to physical access to the embedded systems. We do not wish anyone to have unauthorized access to the systems. The big industrial automation controls systems are the example where we would strictly like to control physical access to only authenticated persons. Instead of using PIN or password-based access controls, it is always recommended to have biometric sensor-based access. It could be anything like face recognition system or fingerprint. This will also help prevent side channel attacks.

Safe Communication:

When systems talk to each other, it should always be a rule to make sure that all 3 tenets (CIA) for security are religiously followed. Again, hardware root of trust is going to be an ideal solution. All new communication protocols (wired and wireless) support TLS based security, and TLS based security is fairly easy to implement. If possible, we should always go with the latest version of these security controls in the industry.

Side Channel Attacks:

Both hardware and software-based approaches have been suggested to identify symptoms to detect the leak of system's power dissipation, timing, and EMR (electromagnetic radiations). Randomized instruction sequence and bit splitting are some of the solutions. Clock signal and the power consumption can also be secured using randomization.

Summary

When we talk of connected device security, the solutions must be:

- Both H/W and S/W based solutions can be deployed depending upon the scenarios. The trade-off between the level of security and financial costs/saving should always be considered.
- It is always recommended to have CPU and OS independent solutions for modularity, if possible.
- Performance, simplicity, the required level of security and costs should always be kept in the mind while deciding an appropriate security control.
- In a short sentence, authentication between two communicating entities, message integrity, confidentiality, replay attack protection [29], and anticipated future security challenges should always be kept in mind. That will help a security architect design better security control.
- Always go for modularity and try to keep components separate. A bug in printer driver should not impact the printer security.

In addition to the above, we should adhere to the following in terms of some of the followed processes-

- There should be a crisp and clear articulation of the security requirement depending upon the products, its lifecycle stages, and environmental factors.
- Threat identification should be done in the design phase itself. Nowadays, a lot of threat modelling tools are available in the market. One can find both free and commercial versions.
- Always prepare a risk management plan and risk ranking matrix to do a cost-benefit analysis.
- It is always recommended to have CPU and OS independent solutions for modularity if possible.
- We should use the most secured and expert-recommended runtime environment. If it is COTS, then its security concerns can always be addressed at a short notice, and the vendor can deliver security fixes at a short notice.
- We should never ignore security testing.
- We should always keep in mind 3 principles. Secure by Default; Secure by Design and Secure by Deployment.
- Product vendor should always be encouraged to go for industry-specific cyber security certificate if there is already one available in the market by a standardized body.
- A security response plan should always be in place.

Conclusion

This paper provides a comprehensive overview of cyber security applicable to Connected Embedded Systems by describing attacks, vulnerabilities, and remediation. It provides information on how a connected embedded system can be attacked. Moreover, the presented ideas and knowledge can assist the organization developing and delivering the secure products.

References

1. <https://www.businesswire.com/news/home/20180221005893/en/Barr-Group's-2018-Embedded-Systems-Safety-Security>
2. <https://www.accenture.com/us-en/insights/security/cost-cybercrime-study>
3. <https://www.businesswire.com/news/home/20190306005218/en/Malware-Malicious-Insiders-Accounted-One-Third-Cybercrime-Costs>
4. <https://owasp.org/www-project-top-ten/>
5. https://en.wikipedia.org/wiki/Row_hammer
6. <https://en.wikipedia.org/wiki/Stuxnet>
7. https://en.wikipedia.org/wiki/Man-in-the-middle_attack
8. https://en.wikipedia.org/wiki/Data_remanence
9. https://en.wikipedia.org/wiki/Cold_boot_attack
10. https://en.wikipedia.org/wiki/2011_PlayStation_Network_outage
11. https://en.wikipedia.org/wiki/Session_hijacking
12. Global Journal of Computer Science and Technology: E Network, Web & Security Volume 16 Issue 1 Version 1.0 Year 2016 by Parves Kamal, Saint Cloud State University, United States
13. <https://www.chipestimate.com/Why-do-you-need-a-hardware-solution-to-secure-your-embedded-system/Silex-Insight/Technical-Article/2014/01/14>
14. https://en.wikipedia.org/wiki/MISRA_C
15. <https://www.csoonline.com/article/3410046/31-hardware-and-firmware-vulnerabilities-a-guide-to-the-threats.html>
16. 16. "Protection of embedded processing systems with a configurable, integrated, embedded firewall." By Peikari, Cyrus. U.S. Patent Application No. 10/346,956.
17. "Customized machine learning-based hardware-assisted malware detection in embedded devices." By Sayadi, Hossein, et al.

18. [https://en.wikipedia.org/wiki/Defense_in_depth_\(computing\)](https://en.wikipedia.org/wiki/Defense_in_depth_(computing))
19. "Security protection and checking for embedded system integration against buffer overflow attacks via hardware/software." By Shao, Zili, et al. IEEE Transactions on Computers 55.4 (2006): 443-453.
20. <https://medium.com/@sbwoodside/defence-in-depth-the-medieval-castle-approach-to-internet-security-6c8225dec294>
21. <https://searchsecurity.techtarget.com/definition/IPsec-Internet-Protocol-Security>
22. <https://www.icann.org/resources/pages/dnssec-what-is-it-why-important-2019-03-05-en>
23. https://en.wikipedia.org/wiki/Intrusion_detection_system
24. <https://searchsecurity.techtarget.com/definition/intrusion-prevention>
25. https://insights.sei.cmu.edu/sei_blog/2016/11/distributed-denial-of-service-attacks-four-best-practices-for-prevention-and-response.html
26. "Electromagnetic analysis (ema): Measures and counter-measures for smart cards." By Quisquater, Jean-Jacques, and David Samyde. International Conference on Research in Smart Cards. Springer, Berlin, Heidelberg, 2001.
27. One-time cookies: Preventing session hijacking attacks with disposable credentials by Dacosta, Italo, et al. Georgia Institute of Technology, 2011.
28. "Cryptographic architecture with random instruction masking to thwart differential power analysis." By Shu, David B., Lap-Wai Chow, and William M. Clark Jr.
29. https://en.wikipedia.org/wiki/Replay_attack
30. <https://www.copyscape.com/?s=437440929842273>
31. <https://www.copyscape.com/?s=800166536842274>

Author Info



Sachin Kumar

Sachin Kumar has more than 23 years of experience in IT industry. He is part of HCL ERS Engineering & Technology office - Security COE. He has diverse experience in security and worked for long time in Fintech domain for a big Swiss bank in Europe. He holds bachelor in Electrical engineering. He is an alumnus of IIM Kozhikode from where he completed his executive MBA with specialization in IT and Finance.

ABOUT HCL TECHNOLOGIES

HCL Technologies (HCL) is a leading global IT services company that helps global enterprises re-imagine and transform their businesses through digital technology transformation. HCL operates out of 46 countries and has consolidated revenues of US\$ 9.9 billion, for quarter ending September, 2019. HCL focuses on providing an integrated portfolio of services underlined by its Mode 1-2-3 growth strategy. Mode 1 encompasses the core services in the areas of Applications, Infrastructure, BPO, and Engineering & R&D services, leveraging DRYICE™ Autonomics to transform clients' business and IT landscape, making them 'lean' and 'agile'. Mode 2 focuses on experience-centric and outcome-oriented, services such as Digital and Analytics Services (BEYONDigital™), IoT WorkS™, Cloud and Security, utilizing DRYICE™ Orchestration to drive business outcomes and enable enterprise digitalization. Mode 3 strategy is ecosystem-driven, creating innovative IP-partnerships to build products and platforms business.

HCL leverages its global network of integrated co-innovation labs, and global delivery capabilities to provide holistic multi-service delivery in key industry verticals including Financial Services, Manufacturing, Telecommunications, Media, Publishing, Entertainment, Retail CPG, Life Sciences Healthcare, Oil & Gas, Energy & Utilities, Travel, Transportation & Logistics and Government. With 120,000 professionals from diverse nationalities, HCL focuses on creating real value for customers by taking 'Relationships Beyond the Contract'. For more information, please visit www.hcltech.com.

ABOUT HCL ENTERPRISE

HCL is a \$8.6 billion leading global technology and IT enterprise comprising two companies listed in India - HCL Technologies and HCL Infosystems. Founded in 1976, HCL is one of India's original IT garage start-ups. A pioneer of modern computing, HCL is a global transformational enterprise today. Its range of offerings includes product engineering, custom & package applications, BPO, IT infrastructure services, IT hardware, systems integration, and distribution of information and communications technology (ICT) products across a wide range of focused industry verticals. The HCL team consists of over 150,000+ ideapreneurs of diverse nationalities, who operate from 46 countries including over 500 points of presence in India. HCL has partnerships with several leading global 1000 firms, including leading IT and technology firms. For more information, please www.hcl.com



www.hcltech.com

Hello there! I am an Ideapreneur. I believe that sustainable business outcomes are driven by relationships nurtured through values like trust, transparency and flexibility. I respect the contract, but believe in going beyond through collaboration, applied innovation and new generation partnership models that put your interest above everything else. Right now 120,000 Ideapreneurs are in a Relationship Beyond the Contract™ with 500 customers in 32 countries.

How can I help you?

HCL