Embedded Systems Threats: Security Threats and Solutions

[1]SIRATLA KAVITHA, [2]ANKISETTY BALAJI, [3]CHAITANYA ANUDEEP ORIGANTI

GAYATRI VIDYA PARISHAD TECHNICAL CAMPUSES,
AMRITA VISWAVIDYAPEETHAM

***ABSTRACT*: With the increasing use of embedded Systems in our daily life, security threats have also been increasing in a proportional rate. However, ensuring security in the embedded systems has become a great challenge not only for the embedded device experts but also for the manufacturers of devices. The problem especially arises because of the limited hardware and software implementation options for the embedded designers. At the same time, companies are trying to keep the vulnerabilities of the operating system of those embedded devices in secret and they are not relieving any necessary security updates quickly. It has become very urgent to ensure proper security of the embedded systems to save it from any major technological disaster near future. In this paper, we have broadly discussed the structures, characteristics and applications of different embedded devices in our daily life. Beside this, we have also discussed about the different causes of security threats and some of our suggested solutions to protect the systems from the attackers as well that we have found in our research.**

***KEYWORDS*: Peripheral, cryptography, firmware, hackers, microcontroller, real-time constraints, encryption, authentication**

.

INTRODUCTION:

I. INTRODUCTION

An embedded system can be defined as a special type of computer system that performs some specific pre-defined programs which is generally used within a larger scale of electrical or mechanical system. Generally, it is started from small MP3 players to largely complex hybrid vehicle systems. Some other examples of frequently used embedded systems in our daily life are keyboard, mouse, ATM, TV, PDA, cell phone, printer, elevator, smoke detector, DVD player, refrigerator, camera, GPS navigator, radio, TV remote, telephone, game controller, monitor, digital image processor, bar code reader, SD card, washing machine, anti-lock breaking system, blender etc. We use embedded systems especially because of its dependability, efficiency and it meets the real-time constrains. Examples of the embedded system show that it has become a part and parcel of our daily life in term of use. We are very familiar with the term 'Automation' because of the deployment of smart embedded system in our home. Now-a-days almost all of the embedded systems are connected with the internet. So security threats have become a major issue at present because most of the embedded systems lack security even more than personal computers. One of the reasons for this lack of security is the very limited hardware and software implementation options for the manufacturers of embedded system companies. Again they have to deal with the competitive market price of the other embedded manufacturer companies because they all have to keep the lowest possible price to maintain the customer satisfaction and at the same time they do not conduct any specific security research of their manufactured embedded products . This leads to the security threats for the embedded devices because ensuring advance security techniques for embedded systems means the higher cost of that embedded products. Customers also don't want to be more expensive usually when buying an embedded devices and they are not concerned also about the probable security threats of their products. Lack of security analysis and low-cost market product mentalities of the manufacturer companies lead the hackers the exact environment they are expecting for. Many embedded systems hacking tools are easily available in the internet. Hacking in the PDAs,mobile phones and modems are very common example of embedded systems hacking. Recent development trends of the embedded systems protocol are going to be convergence because of its applications in TCP/IP protocol for the purpose of inter-media interfacing. In this case, usingboth Cisco IPSec and IKEv2 will cost much more for the development of the embedded applications at least for the next few years. As a result IPv4 is going to dominate in the applications of embedded systems. This IPv4 is much more challenging for its internal security problems in terms of authentication, integrity confidentiality and cheap hardware.

II. STRUCTURE OF EMBEDDED SYSTEM

Although there are many types of applications, the principle of the embedded device structures is typically the same in terms of system components and design methodologies. Complex applications such as chemical plants may need standard I/O (Input / Output) devices but this is not mandatory for the most of the other embedded systems. At present, most of the embedded systems are microcontroller based that means memory and peripheral are integrated with the Central Processing Unit (CPU). In general it can be divided into three categories: small, medium and large. Small such as TV remote needs 4-bit microcontrollers.now a days TV remote are with 8-bit or 16-bit microcontrollers are well enough for medium size systems such as automated data acquisition systems and 32-bit or more needed for the high–end large scale computer system such as plant monitoring and central control system. Embedded systems are not standalone always rather than in the most of the time it is used as a part of a larger complex device. Here performance based real-time constrains must be met for the usability and safety of those devices. Graphical user interface is not always mandatory for the small scale device such as simple button or LED (Light Emitting Diode). But it is a must

for the bigger and complex devices such as nuclear power plant systems along with the networks, data bus connections, screen-edge systems etc.

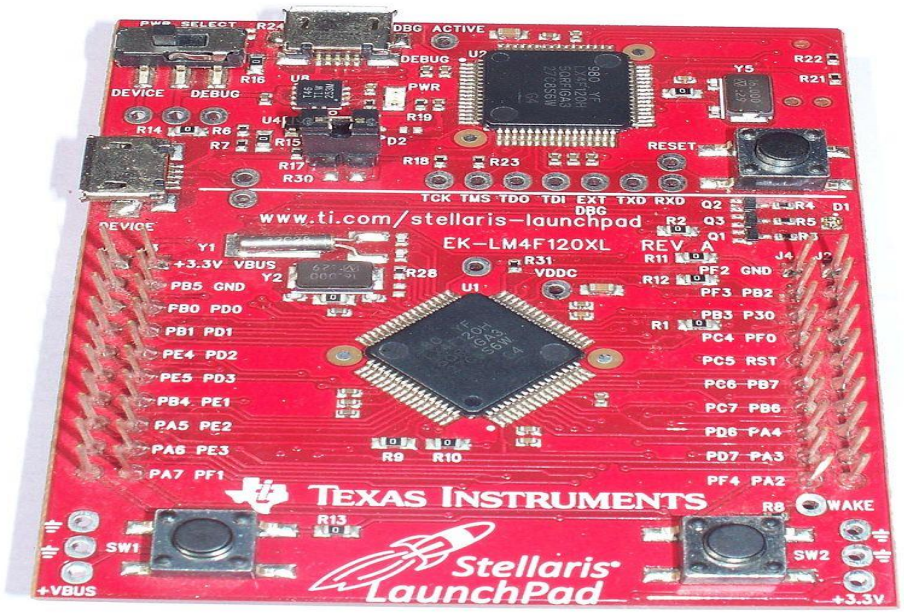

Figure: *LM4F120 TI Stellaris Launchpad Embedded ARM Board*

The term 'PSoC' stands for Programmable System on Chip. It is a programmable embedded design platform which integrates discrete, analog and programmable logic with a memory and a microcontroller. PSoC 6 is based on 32-bit ARM Cortex-M4 core with an added ARM CORTEX-M0+ core . It allows the designer to make flexible changes during design, validation and production. It is easy to reconfigure and implement using fewer system components. A single PSoC device can integrate about 104 peripheral functions. It also offers single-chip integration of multiple buttons, sliders, touch pads and proximity detectors with requiring no external components for sensing.

Characteristics of Embedded Systems:

In general, embedded systems are designed to perform any particular pre-defined task that must meet any real time constraint. The main difference between a computer and an embedded system is a computer is used to perform multiple tasks defined by the user. On the other hand, an embedded system is used to perform a specific task that is pre-defined by the manufacturers. Here, meeting all the real-time constraints is a very important characteristic of an embedded system. A real-time constraint is divided into two parts. One is hard real-time system and the other is soft real-time system. Hard real-time system means it must meet all its deadlines with a zero degree of flexibility and it is acceptable to be little flexible in the soft real-time system. It is not necessary to be standalone always for the embedded devices. Actually most of the embedded systems are integrated within a large computerized device. Devices such as MP3s, cameras and TV remotes are the example of standalone embedded devices. For the example of integrated embedded devices car and nuclear power plant are some good examples. GPS, fuel injection controller, anti-locking brake system, transmission controller, cruise control, active suspension, air- bag system, air-conditioner, display monitor-all the devices are integrated in a modern car system. The term 'firmware' is used to refer the program instructions written for embedded systems. It is stored in ROM (Read Only Memory) or in a flash memory chip. Resources like computer hardware do not need much to run. Another important characteristic of embedded systems is the dedicated user interface. It may range from no user interface to complex graphical user interface. For simple button and LED system, no user interface is needed. User interface means the task of button can change with the on-screen display and the selection depends on the user. Handheld device such as joystick which needs to be pointed with the screen is a good example user interface system. Size and weight should be less for an embedded device. For that reason, microcontrollers are used in embedded devices to deliver the best performance on demand. Generally, microcontrollers are required to perform repeated functions for long time without any failure. Beside this, it must be reliable and safe in case of some special systems such as car's anti-locking brake system and nuclear power plant controlling systems. Adding to those characteristics, embedded systems must be cost efficient also. Manufacturer companies try to keep the lowest price of their products. Using sensors and actuators it may be also connected with physical environment.

III. THREATS OF EMBEDDED SYSTEMS

This section lists some examples of attacks against embedded devices and systems and looks into the attackers' capabilities and their implications. Although not comprehensive, in our view, the examples are very representative and cover a broad range of application domains such as industrial systems, communications, and consumer devices. Presents a timeline for critical infrastructure. Noteworthy attacks date back to the 1982 and the number of attacks have been increasing since 2001. Presents

vulnerabilities and possible exploits of key management in wireless devices. For example, one of the devices is shipped with a graphical user interface with default values to configure the device. The implementation of the interface generates a passphrase which is later used to generate the AES key. However, the PseudoRandom Number Generator is seeded by the srand() function using the current time and generator itself is the rand() function. As a result, the attacker is capable of calculating the passphrase and the encryption key and can intercept all communication on the target wireless network. Demonstrates remote attacks against SCADA devices using the ModBus protocol. The vulnerability exploited is within the design of the protocol: it lacks encryption and authentication. As a result, a device exploitation can be easily achieved with a carefully crafted packet. RuggedCom devices can be attacked via hardcoded credentials in the operating system . The default account is present in the system to support password recovery, so can not even be disabled. However, attackers with the knowledge of the MAC address can use this account to connect to the device and take full control of it. Presented multiple attacks against satellite communication systems originating from the ground segment. In one of the attack scenarios, the man-machine interface of the airplane onboard SATCOM unit requires administrator password for restricted configurations and control mechanisms. The generation algorithm uses the device serial number (can be found printed on the device) plus a hard-coded string, which makes it easy to guess the password. Thus the attacker has access to all configurations and can disable critical parts related to the safety of the aircraft. Implemented a rogue carrier for satellite systems. Their method allows the attacker to become an illegitimate user of services provided. Firstly, the attacker must select its target, an artificial satellite. Then, the attacker point his antenna to the target and searches for unused, legal frequency for clients. If such a frequency is found, the attacker is free to transmit and receive as he wishes. However, the attacker still has to avoid detection: he has to sniff packets send by the operator to legitimate clients and do exactly as the operator packet asks. As stated in their talks, the method works because even if the satellite supports encryption, turning it on causes performance to drop significantly. As a result, operators turn it off because it is the service customers pay for, not the security of the service. Investigates the Automatic Dependent Surveillance-Broadcast (ADS-B) protocol and presents practical attacks using the vulnerabilities the protocol has: no authentication, no encryption and no challenge-response mechanisms. As a result, messages can be sniffed, spoofed or replayed. The attacker may confuse pilots and hinder them in performing their tasks. Presents an attack against a smart home automation device, the Nest Thermostat. Pressing a button for 10 minutes on the device initiates a global reset. Afterwards, there is a small time window during which the device accepts code from USB sticks connected to it and uses that code for booting without any cryptographic checks on the code. An attacker can use this vulnerability to install an SSH server and access the home network of the user. However, physical access is needed to the device to launch the attack, so either the attacker has to break into the house or compromise the device during transport. Presents physical and remote attack surfaces in cars. For example, the authentication protocol between the Telematics Unit and the center relies on a challenge response mechanisms. However, the random number generator is seeded with the same constant each time it is initialized. As a result, an observed response packet can be replayed by the attacker to authenticate himself as the Telematics Call Center, getting full control over the car. A possible attack against a wireless home automation device is presented in Which is used for controlling electrical outlets. The implementation of the Home Network Administration Protocol contains a buffer overflow which can be used to execute arbitrary code on the device. Since the device controls the power outlet to any device physically connected to it, the attacker has the ability to damage the connected device. The D-Link DIR-825 Wireless-N Dual Band Router contains a command injection vulnerability which allows the attacker to get remote access to the device as demonstrated by . The vulnerability lies with the packet parsing: strings inside backticks are considered commands and executed on the router. Discusses a case study of malicious firmware updates to a HP-RFU (Remote Firmware Update) LaserJet printer. The vulnerability which enables this attack comes from the fact that the printer has to accept printing jobs in an unauthenticated way (as dictated by the standard) and that the firmware is updated by printing to the memory. Thus, an attacker can send a printing job to the device, instructing it to update its firmware with the malicious code provided. Discusses attacks against a fireworks control system. The protocol used by the system provides no encryption, nor authentication, which allows the attacker to sniff packets and thus learn the addresses of each device. Now, the attacker might wait for the operator to arm the system, the attacker can immediately send the digital arm and fire commands. The system will immediately fire its pyrotechnics loads and may cause physical harm to the operator. The attack can be automated as well, since arbitrary Python code can be uploaded to the devices. Demonstrates multiple attacks against an automated external defibrillator. For example, the firmware upgrade software package shipped with the device contains a buffer overflow vulnerability which may result in arbitrary code execution. Another vulnerability is the use of CRC as a digital signature. Combining these two vulnerabilities allows the attacker to harm patients by setting shock protocols and shock strengths or launch a cyberattack against the IT system in which the device is deployed.

IV. SOLUTIONS OF SECURITY THREATS IN EMBEDDED SYSTEMS

Security requirements of embedded devices can vary from Situation to Situation. As an example of a cell phone system, end user may be concerned about his private data protection while content provider may be concerned about copy protection of the multimedia contents delivered to the embedded devices and manufacturers may be concerned about the firmware that has been used in that cell phone. Here the system of attack may also vary for users, content providers, manufacturers etc. We have already described different challenges of embedded systems in term of security and in this section we will describe some probable solutions also to get rid of those problems also. Modern cryptography techniques provide strong defiance against the conventional attacks. However, much more effort and care is still required in the software design to make the system more protected from bugs and design flaws. Designers should be emphasizing more on Software Development Life Cycle (SDLC). Different secure level practices should be applied which can be classified into three. They are the design level, the implementation level and the testing level. Tamper-resistance techniques should be strengthening more to protect the system against different software and hardware attacks. These techniques can be used for attack detection, recovery and prevention as well. To prevent side-channel attacks, different hardware and software level approaches have been proposed to identify symptoms that allow the leak of the system's side-channel

information like power dissipation, timing and electromagnetic radiations. Software based countermeasures include randomization instruction sequence, introducing dummy instructions, bit splitting and balancing hamming weights of internal data. Randomization can also be applied on the clock signal or the power consumption. It has been experimented that software based countermeasures are most efficient although they slightly decrease the performance of cryptographic algorithm in terms of memory, energy and execution time.Security solution in the architectural level should also be improved that means consider the mapping of adopted algorithms and protocols more efficiently. One solution to overcome the limitation of software based efficiency is to implement the resource-greedy cryptographic computations on a dedicated hardware using Application Specific Integrated Circuits (ASICs). Therefore 'hardwired algorithm' approach may be followed for its proven performance although it's costly. Beside those solutions, some extra added modules such as SSL and SSH may also be implemented. It would be the best solution to protect many attacks such as denial of service (DoS) attack, spooling, hijacking and sniffing although implementation of such value added module is not mandatory because of the lacking of hardware resources available.

V. CONCLUSION

Embedded devices have made our life more easy and comfortable by meeting almost all the real-time constraints. Although it is very popular among the people but they are quite unconscious about the probable security threats till now even the manufactures and the engineers associated with embedded devices. Expert hackers from the different parts of the world have already found many security vulerablities of the embedded devices and they are further working on it. So, it is very clear that it could create a huge blow in near future for the technological industry if the engineers and the manufactures do not take the necessary security solutions as proposed in this article to protect the unauthorized access from the unsecured third party. We heartily believe that more concentration on cryptography, tamper-resistance techniques, advanced microcontroller and algorithms can mostly make the embedded devices secure enough. At the same time, it is also important for the manufacturer companies to design and implement the whole embedded system with much more security concern.

REFERENCES

[1] Sudhakar Singh, Prashant Mor and Gajendra Singh, Application of Embedded Systems in Modern Society, VSRD International Journal of Electrical, Electronics & Communication Engineering, 2 (6), 2012, 373-384

[2] Jesús Lizarraga, Roberto Uribeetxeberria, Urko Zurutuza, Miguel Fernández, Security in Embedded Systems, Computer Science Department, Mondragon University, Spain

[3] "Hacking the d-link dsp-w215 smart plug," http://www.devttys0.com/ 2014/05/hacking-the-d-link-dsp-w215-smart-plug/, /DEV/TTYS0, May 2014.

[4] James O. Hamblen, Introduction to Embedded Systems Using Windows Embedded CE, School of Electrical and Computer Engineering, Georgia Institute of Technology, USA, 2007

[5] Philip Koopman, Embedded System Security, Associate Professor,Department of Electrical and Computer Engineering, Carnegie Mellon University, 2004

[6] Bergman, P., Berman, S., The Criminal Law Handbook: Know Your Rights, Survive the System

[7] Paul Kocher, Ruby Lee, Gary McGraw, Anand Raghunathan and Srivaths Ravi, Security as a New Dimension in Embedded System Design, NEC Laboratories America, Princeton, NJ

[8] Bleichenbacher, D., and Nguyen, P. Q. 2000, Noisy polynomial interpolation and noisy Chinese remaindering. Advances in Cryptography, EUROCRYPT 2000

[9] P. Kocher, R. Lee, G. McGraw, A. Raghunathan, and S. ModeratorRavi, "Security as a new dimension in embedded system design," in Proceedings of the 41st annual Design Automation Conference. ACM, 2004, pp. 753–760

[10] Morrose, F., Reiter, M., and Wetzel, S (1999), Password hardening
based on keystroke dynamics. ACM Conference on Computer and Communication Security

[11] National, R. C., (2002), Cyber Security Today and Tomorrow: Pay Now or Pay Later, National Academy Press, Washington, D.C.

[12] Massey, J. L. 1969, Shift register synthesis and bch decoding. IEEE Transactions on Information Theory vol. 15, no. 1, pp. 122- 127

[13] Smeulders, AWM 2000, Content based image retrieval at the end of the early years', IEEE Transactions on pattern analysis and machine intelligence, pp. 1349-1380

[14] Phrack Magazine 2014, Hacking with Embedded Systems, www.phrack.org

[15] Embedded Insights Inc. 2012, Security and Encryption, www.embeddedinsights.com

[16] Dorottya Papp, Zhendong Ma, Levente Buttyan Embedded Systems Security: Threats, Vulnerabilities, and Attack Taxonomy, 2015 Thirteenth Annual Conference on Privacy, Security and Trust (PST)

[17] Anik Barua , Mohammad Minhazul Hoque , Rubina Akter: Embedded Systems: Security Threats and Solutions, American Journal of Engineering Research (AJER)

[18] Embedded Insights Inc. 2012, Security and Encryption, www.embeddedinsights.com

[19] Embedded Systems, www.wikipedia.com