IJSTA

International Journal of Scientific and Technical Advancements

ISSN: 2454-1532

Security Threats and Solutions in Embedded Systems: Review Paper

Manu Gupta, Danish Mukhtar
Department of Information & Technology, MIET, Jammu, J&K

Abstract**—**Embedded systems are being widely used and with the use the threats for security are increasing day by day. Many approaches have been proposed in the past to secure embedded systems but various facts such as deployment scale, resource limitations, the difficulty of physical protection, and cost consideration all make it very challenging to secure them. In this paper we have discussed the components, types, characteristics, applications of the embedded systems. Moreover, security threats and solution to the security threats in the embedded systems is also discussed.

Keywords**—** Embedded System, Security threats.

I. Introduction

An Embedded System can be defined as a system mainly built into a larger system that performs some specific pre- defined programs. Simply Embedded System consists of hardware and software embedded in it. optionally it may also contain mechanical part. Generally, embedded systems are devices used to control, monitor or assist the operation of equipment, machinery. Some examples of embedded systems which we use in our day today life are navigation systems, badge readers, crock pots, PDAs, battery chargers, cameras, elevators, MP3 players, patient monitoring systems, clocks, smoke detectors, TVs, coffee makers, thermostats, DVD players keyboards, ultrasonic toothbrushes, curling irons, vending machines, cell phones, internet servers, cash registers, cordless phones, printers, microwave ovens, ceiling fans, hot tubs, ATMs, garage door openers, parking meters, refrigerators etc. Processor is the main part of embedded system which could also be a generic microprocessor or a microcontroller and programmed to perform the specific tasks for which the integrated system has been designed.

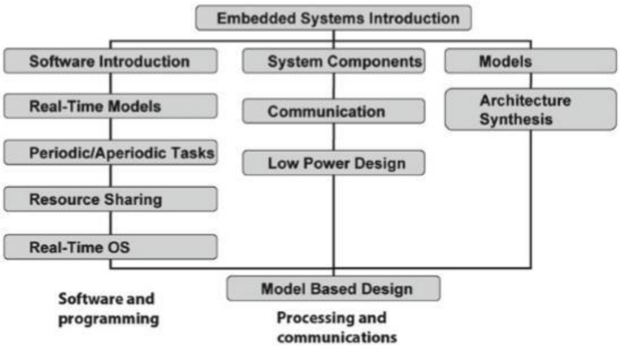

Fig. 1. Embedded system design.

Almost all of the embedded systems are connected with the internet due to which security threats have become a major issue because of lack of security in most of the embedded systems. Although many approaches have been proposed in the past to secure embedded systems but various facts such as deployment scale, resource limitations, the difficulty of physical protection, and cost consideration all make it very challenging to secure them, particularly for devices with remote control, maintenance and operation functions. Due to competition of market price between the embedded system manufacturer companies, these companies have to maintain the customer satisfaction which is possible by lowering the price of products. So, they manufacture less secure embedded systems and do not conduct any specific security research of their manufactured embedded products. Hackers take advantages of this scenario and easily attack the embedded system. Hackers simply get one system and with reverse engineering techniques find flaws and weaknesses in the design which can be exploited to create attacks against other system installed in the field. Due to internet connectivity in the embedded systems hackers use this capability to install their own code onto an embedded system and control it.

II. Components of Embedded System

The embedded system has mainly three components Hardware, Software, and Real-time operating system. Processor, Timers, Interrupt controller, I/O Devices, Memories, Ports comes under Hardware Component. The main part of an embedded system is the processor, which could be a generic microprocessor or a microcontroller which is programmed to perform the specific tasks for which the integrated system has been designed. Memory (RAM, ROM and Cache) is also an important part in embedded system. In RAM data is temporarily stored during execution of the system. In ROM input output routines are stored which are needed for the system at boot time. The cache is used as a temporary storage during the processing and transferring of data by the processor. The system clock (Hardware part) which is generally composed of an oscillator and some associated digital circuitry is used for all processes that are

Second Component is Application Software Which may perform concurrently the series of tasks or. Third Component is Real Time Operating System (RTOS) that defines the way the system work, supervise the application software. It sets the rules during the execution of the application program. A small scale embedded system may not need an RTOS.

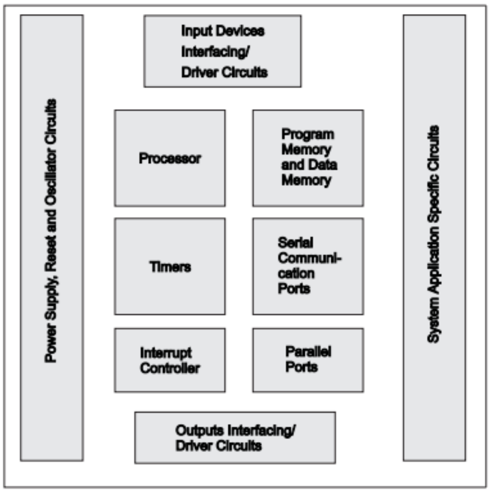
Input Devices Interfacing/ Driver Circuits
Processor
Program Memory and Data Memory
Timers
Serial Communi-cation Ports
Interrupt Controller
Parallel Ports
Outputs Interfacing/ Driver Circuits
Power Supply, Reset and Oscillator Circuits
System Application Specific Circuits

Fig. 2. Embedded system hardware.

III. Types of Embedded Systems

Embedded systems can be classified into different types based on performance, functional requirements and performance of the microcontroller.

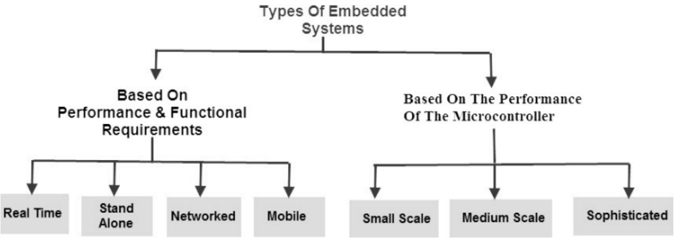
Types Of Embedded Systems
Based On Performance & Functional Requirements
Based On The Performance Of The Microcontroller
Real Time
Stand Alone
Networked
Mobile
Small Scale
Medium Scale
Sophisticated

Fig. 3. Types of Embedded systems.

Based on their performance and functional requirements: Real time embedded system gives a required o/p in a particular time. They follow the time deadlines for completion of a task.
Stand alone embedded systems do not require a host system like a computer, it works by itself. It takes the input from the input ports either analog or digital and processes, calculates and converts the data and gives the resulting data through the connected device-Which either controls, drives and displays the connected devices. Examples for the stand alone embedded systems are mp3 players, digital cameras, video game consoles, microwave ovens and temperature measurement systems.

Network embedded systems are related to a network to access the resources. The connected network can be LAN, WAN or the internet. The connection can be any wired or wireless. Example for the LAN networked embedded system is a home security system wherein all sensors are connected and run on the protocol TCP/IP

Mobile embedded systems are used in portable embedded devices like cell phones, mobiles, digital cameras, mp3 players and personal digital assistants, etc. Based on the performance of the microcontroller. Small Scale embedded systems are designed with a single 8 or 16-bit microcontroller, that may even be activated by a battery. For developing embedded software for small scale embedded systems, the main programming tools are an editor, assembler, cross assembler and integrated development environment.

Medium scale embedded systems are designed with a single or 16 or 32 bit microcontroller, RISCs or DSPs. These types of embedded systems have both hardware and software complexities. For developing embedded software for medium scale embedded systems, the main programming tools are C, C++, JAVA, Visual C++, RTOS, debugger, source code engineering tool, simulator and IDE. Sophisticated embedded systems have enormous hardware and software complexities, that may need ASIPs, IPs, PLAs, scalable or configurable processors. They are used for cutting-edge applications that need hardware and software Co-design and components which have to assemble in the final system.

IV. Characteristics of Embedded System

Embedded systems are designed to do some specific task, that is embedded systems are application specific & single functioned, rather than be a general-purpose computer which are designed for multiple tasks. Some systems have real-time performance constraints that must be met, for reasons such as safety and usability, some may have low or no performance requirements, allowing the system hardware to be simplified to reduce costs .Embedded systems are typically designed to meet real time constraints. For real time systems, right answers arriving too late (or even too early) are wrong.
Efficiency plays important role for embedded systems. They are optimized for energy, code size, execution time, weight & dimensions, and cost. For small scale devices such as simple button or Light Emitting Diode Graphical user interface is not mandatory but must for the bigger and complex devices such as nuclear power plant systems along with the networks, data bus connections, screen-edge systems etc.

Watchdog is the component that resets the computer in regular time intervals with the use of an internal timer, which can be added to the embedded system in order to make the system capable of self-reset in case of data loss or corruption, as embedded system may not be physically accessible for example space systems. Hence Embedded system must have a self-resetting capability. Embedded devices can be categorised in Standalone Embedded Devices and Integrated Embedded Devices

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.
Embedded systems must be cost efficient. Manufacturer companies try to keep the lowest price of their products. But it is obvious, the device which is less costly will be less secure. So it is a challenge for the manufacturing companies to make the product less costly and much secure. In the Embedded

systems, the useful software to manage them is “Firmware”. It is the type of software which is stored in ROM or Flash memory chips.

V. Applications of Embedded Systems

Embedded systems are being widely used and day by day its usage goes on increasing as it makes our life easy. The various applications of embedded systems are as under:
Home Appliances – Washing Machine, Microwave, HV AC System, DVD, Dishwasher, Microwave Oven, Set-Top Box, Home Security System. Office Automation – Fax, Copy Machine, Smartphone systems, Printers, Scanners.
Security – Face Recognition, Finger Recognition, Eye Recognition, Building Security Systems, Alarm Systems, etc.
Academia – Smart Board, Smart Room, OCR, Calculator, Instrumentations – Signal Generator, Signal Processor, Power Supplier, Process Instrumentations, .
Telecommunications – Router, Hub, Cellular phone, Web Camera,.
Auto mobile – Fuel Injections Controller, Air-bag Systems, GPS, Cruise Control, .
Aerospace – Navigation systems, Automatic Landing Systems, Space Explorer, Space Robotics, .
Industrial Automation – Data Collection Systems, Monitoring SystemsIndustrial Robotics. .
Medical – CT Scanner, ECG, EEG, EMG, MRI, Blood Pressure Monitor, Medical Diagnostic Devices,etc.
Banking and Finance – ATM, Smart Vendor Machines, Cash Registers, etc.
Entertainment – Video games, Robot, MP3, Mind Storm, Smart Toy.

VI. Causes of Security Threats of Embedded Systems

As earlier discussed that almost all of the embedded systems are connected with the internet due to which security threats have become a major issue. It is well known fact that embedded devices are being more connected to our life and its usage is increasing day by day whereas on the other hand its security threats are also increasing as a proportional rate. Various causes of embedded system security threats are explained below:

One main cause for security threat is due to competition of market price between the embedded system manufacturer companies, these companies have to maintain the customer satisfaction which is possible by lowering the price of products. So, they manufacture less secure embedded systems and do not conduct any specific security research of their manufactured embedded products. Hackers take advantages of this scenario and easily attack the embedded system. Hackers simply get one system and with reverse engineering techniques find flaws and weaknesses in the design which can be exploited to create attacks against other system installed in the field. Due to internet connectivity in the embedded systems hackers use this capability to install their own code onto an embedded system and control it.

One reason may be the use of more popular programming languages such as C and C++ as they are very efficient for embedded systems but they cannot protect against the simple kinds of attacks such as buffer overflows. The processing capabilities of many embedded systems are easily overwhelmed by the computational demands of security processing, leading to failures in sustaining required data rates or number of connections. Battery-driven systems and small form-factor devices such as PDAs, cell phones, and networked sensors are often severely resource constrained. It is challenging to implement security in the face of limited battery capacities, limited memory, and so on. An ever increasing range of attack techniques for breaking security, such as software, physical, and side-channel attacks, require that the system be secure even when it can be logically or physically accessed by malicious entities. Countermeasures to these attacks need to be built in during system design.

Embedded devices have to perform same task again and again usually by using loop. Here, speed can easily reach to 20 loops in every single seconds with strong real-time constraints. Hence a single delay of even 0.01 second can cause a loss of control loop stability which means the system can be vulnerable to attack that is designed to destroy the system timing. In the most of the time, embedded systems have no real administrator by which an internet connected device can be easily launched by distributed denial of-service (DoS) attacks by the hackers. Many embedded systems are designed and developed by the small development teams even by the single engineer that cannot afford any embedded system security specialist.

VII. Solutions of Security Threats in Embedded Systems

Earlier we discussed about different challenges of embedded systems in term of security .This section will describe some probable solutions for few of those problems. Modern cryptography is best techniques which provides strong defiance against the conventional attacks but still, more efforts are still needed at higher levels to protect the embedded software from a large diversity of attacks which exploit their development defects essentially caused by implementation bugs or design flaws. Proper protocols needs to implement for the manufacturer companies for installation of security channel in the system. This should be done up to that level, which the customer can afford. Moreover, the designers should be well aware and should emphasize more on Software Development Life Cycle (SDLC). Security solutions in the architectural level consider the mapping of adopted algorithms and protocols within a layer of software and hardware specializations.

One approach that overcomes the processing inefficiency of the software-based solutions is to completely implement the resource-greedy cryptographic computations on a dedicated hardware using ASICs (Application Specific Integrated Circuits). With this manner, the embedded processor can offload the cryptographic computations, through high speed bus for example, to the custom hardware designed to guarantee higher processing speed and lower energy consumption. Various companies offer Cryptographic Hardware Accelerators that implement diverse asymmetric and symmetric ciphers for systems ranging from low-power mobile appliance and smartcards to high performance network routers.

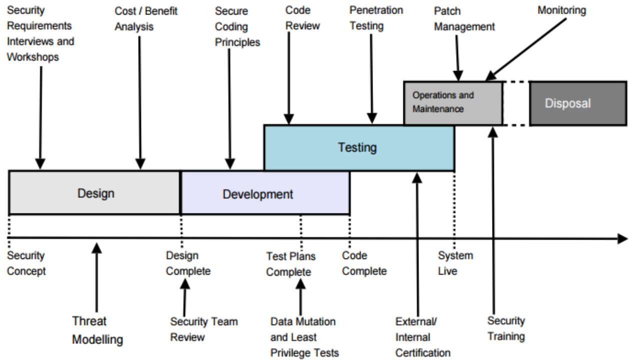

Fig. 4. Secure software development life cycle.

Although this ”hardwired-algorithm” approach has proved its excellent performance level, it’s much less effective in term of cost and flexibility, when several cryptosystems are required in order to support multiple security protocols and emerging standards for better interoperability with other systems. In an attempt to achieve a trade-off between processing performance, flexibility and design cost, a third generation solutions have been proposed, they can be classified into two subcategories: processors enhancement with (i) special purpose extensions, and (ii) general purpose extensions. With this manner, the embedded processor can offload the cryptographic computations, through high speed bus for example, to the custom hardware designed to guarantee higher processing speed and lower energy consumption.. 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. SSL and SSH may also be implemented which 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.

VIII. Conclusion

From the above discussion it is clear that the embedded systems and very useful in day to day life as they make the life easy. With the increasing usage the security threats are also going on increasing. Attackers have made their path to pass the security level and are working more 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 the unauthorized access from the unsecured third party. Cryptography, tamper-resistance techniques, advanced microcontroller and algorithms can mostly make the embedded devices secure enough. Their should also be implementation of proper rules and regulations for the manufactures in order to make every system more secure possible.

References

[1] A. Barua, M. M. Hoque, R. Akter, “Embedded Systems: Security Threats and Solutions Department of ICT,” Mawlana Bhashani Science and Technology University, Bangladesh.

[2] S. Singh, P. Mor and G. Singh, “Application of Embedded Systems in Modern Society,” VSRD International Journal of Electrical, Electronics & Communication Engineering, 2 (6), 2012, 373-384

[3] J. Lizarraga, R. Uribeetxeberria, U. Zurutuza, M. Fernández, “Security in Embedded Systems,” Computer Science Department, Mondragon University, Spain

[4] D. Papp∗†, Z. Ma†, L. Buttyan∗, “Embedded Systems Security: Threats, Vulnerabilities, and Attack Taxonomy, (∗CrySyS Lab Budapest University of Technology and Economics, Hungary and †Digital Safety & Security Department AIT Austrian Institute of Technology, Austria)

[5] S. Ravi and A. Raghunathan, Nec laboratories america paul kocher cryptography research and sunil hattangady texas instruments inc, Security in Embedded Systems: Design Challenges

[6] L. Khelladi, Y. Challal, A. Bouabdallah, and N. Badache, “On Security Issues in Embedded Systems: Challenges and Solutions.

[7] Embedded systems: www.wikipedia.org

[8] Classification of embedded systems with classifications: www.efxkits.us