**INTRODUCTION OF PROJECT UNDERTAKEN**

**CHAPTER 1**

* **Company’s Vision and Mission**

**Vision:**

**We are the link between the real and digital world**

Our vision describes the future we envision and want to achieve for our organization. It is the ultimate goal that inspires and motivates us to keep going forward.

Internet of Things, artificial intelligence, Industry 4.0, Deep Machine Learning, Intuitive Sensing – digital transformation is becoming reality, and the physical world will be connected with the virtual world like never before.

Semiconductors are the backbone of these technological developments: They link the real world to the digital one. Sensors turn electronic impulses into data, actuators turn data into electronic impulses, microcontrollers control entire systems with a single chip, security controllers protect devices and data integrity, and smart power ICs regulate the production, distribution and consumption of energy.

Our customers strive for new business and service models, enhanced system performance and reliability, and shorter time to market. To achieve this, they need a semiconductor partner that can combine technology leadership with cutting-edge system understanding and core capabilities spanning sensing, cross-application control, power efficiency and security. As an enabler of future technologies, Infineon is the right partner to capitalize on the opportunities of advancing digitalization.

**Mission:**

**We make life easier, safer and greener**

Our mission describes how we are helping to solve the technological, economic and social challenges the world is facing today. With our everyday work, we are doing our part to create an easier, safer and greener life for generations to come– with technology that achieves more, consumes less and is accessible to everyone. Microelectronics from Infineon is the key to a better future.

* **We make life easier**

Smart functions such as speech recognition, gesture control and 3D applications (augmented/virtual reality) make everyday items like speakers, wearables and smartphone apps more convenient and easier to use.

Whether based on MEMS, radar or sensor chips, semiconductors from Infineon make life easier. Thanks to our technologies, smartphones and tablets have increasingly compact and lighter adapters, shorter charging cycles and longer battery lives.

* **We make life safer**

As web-based services are becoming the norm, it is increasingly important to protect our data, devices and electronic documents against misuse. Our security solutions use innovative encryption technologies to safeguard identities and data.

We also help make our roads safer – with active safety systems like pedestrian detection and blind spot alerts that correct driver errors to prevent accidents. Our solutions also make premium-class automotive safety systems affordable in mid-range and compact car classes.

* **We make life greener**

Our world’s energy needs keep growing and growing. Today, we already need to produce, transmit and use energy more efficiently. Our semiconductors are used to effectively generate electricity from solar and wind sources, and transmit energy with almost no losses.

Our technologies reduce energy consumption in cars, trains, industrial plants, consumer electronics and household appliances.

**Values:**

**We commit. We partner. We innovate. We perform.**

Not just what we do, but how we do it makes us a successful company. Our values summarize the attitude we adopt and the approach we take in our work. They shape our relationships with customers, partners and colleagues – and hence our corporate culture.

As guiding principles for our behavior and actions, they distinguish us from the competition and form the basis for our sustained success.

* **We commit**

Our customers can trust in our commitment to the highest standards of quality in our products, services, manufacturing and supply chain.

* **We innovate**

We are a key driver of semiconductor innovations, giving our customers a game-changing lead in their markets.

* **We partner**

We build on long-term, reliable relationships with our customers, distributors and ecosystem partners to ensure optimal results for all players.

* **We perform**

We are dedicated to delivering outstanding results and creating sustainable value for our customers, shareholders and employees.

* **Origin and Growth of the Company**

**The Infineon Technologies AG was founded in 2000, originated from the Siemens semiconductor division. In the same year, the facility and the production area were purchased from CVEM in Hungary.**

**Text

Description automatically generated**

****

Chart, pie chart

Description automatically generated



Timeline

Description automatically generated



A picture containing diagram

Description automatically generated

Diagram

Description automatically generated

* **Various Departments and their functions**
* **Automotive**

The ATV division is shaping the future of mobility by enabling clean, safe and smart cars. Its product and solution offering is powering the transition to hybrid and purely electric vehicles. It is supporting the next stages of automated driving as well as higher levels of connectivity, digitalization and security in today’s cars. ATV drives safety, digital cockpit, infotainment, comfort and lighting innovations. Its portfolio integrates sensors, microcontrollers, high-performance memories for specific applications, power semiconductors based on silicon (Si) and silicon carbide (SiC) as well as components for human-machine interaction and vehicle connectivity. Infineon is the world leader in automotive semiconductors

* **Industrial Power Control**

The IPC division delivers leading semiconductor solutions for the smart and efficient generation, transmission, storage and use of electrical energy. Its broad application spectrum includes photovoltaic installations, wind turbines, high-voltage DC transmission and energy storage systems, industrial power supplies, trains, electric commercial vehicles, home appliances, and charging infrastructures for electric vehicles. Infineon is the global number one for IGBT-based power semiconductors. With its SiC solutions for industrial applications, IPC is also in a worldwide leading position. The division is also advancing integration and digitalization with innovations such as energyefficient Intelligent Power Modules (IPM). Building on Infineon’s extensive portfolio spanning sensors, microcontrollers and connectivity technologies, IPC provides innovative solutions for smart home and industrial IoT applications.

* **Power & Sensor Systems**

The PSS division delivers a wide range of power, connectivity, RF and sensor technologies designed to reduce the size and weight of chargers, power tools and lighting systems, while also increasing energy efficiency. The next generation of silicon and wide-bandgap solutions (SiC and gallium nitride) provide unparalleled performance and reliability for 5G, big data and renewable energy applications. Highly precise XENSIV™ sensor solutions are bringing “human” senses to IoT devices, enabling them to react intuitively to their surroundings. Audio amplifiers complement the PSS portfolio, enabling smart speakers and other audio applications offering exceptional sound experiences.

* **Connected Secure Systems**

The CSS division provides end-toend systems for a connected, secured world – building on trusted, game-changing microcontrollers as well as wireless and security solutions. In recent decades, CSS has delivered microcontrollers plus WiFi, Bluetooth and combined connectivity solutions (known as connectivity combos) along with hardware-based security technologies to power the broadest application spectrum spanning consumer electronics, consumer and IoT devices, cloud security, IT equipment, home appliances, connected cars, credit and debit cards, electronic passports, ID cards and more. The division is at the forefront of computing, wireless connectivity and trusted technologies that are helping to securely connect the networked systems of today and tomorrow.

**INTRODUCTION OF PROJECT UNDERTAKEN**

**CHAPTER 2**

* **Objectives of the work undertaken**

The main objective of the project work from the industry perspective was to create an APDU fuzzer to fuzz test their product. There were several other objectives and expectations included in conjunction to the main one as the APDU fuzzer was to be a new tool or methodology to test out their product.

I was expected to understand the testing concepts and techniques that are being used in the company for me to apply it in best use. Testing concepts would let me be clear about why we test a component and why we do not. It would help me in efficient time division per SUT and not waste it on a less important component. Testing technique would help me to choose the best suitable testing approach as per the given SUT. Example: Choosing random fuzzing method over grammar-based approach for a protocol-based message would lead to very less coverage per test cases generated which would eventually lead to loads of wasted time.

Another objective was to not only create an efficient APDU fuzzer but also the fuzzer must respect the requirement analysis and support it. In an industry, there could be a lot of variation in requirements per head and a tool must perform the best for all of it. Thus, the APDU fuzzer framework should also be performing as per the requirements.

The APDU fuzzer was expected to do most of the tasks by its own with very little user interventions. It should be very user friendly and must have proper documentations for anyone to easily get started with it.

The final goal of the project was to integrate the APDU fuzzer framework to Infineon test framework and demonstrate its working on one of the Infineon products. This would mark the success of the tool that it follows all the protocols as expected and thus could be used for fuzzing purpose for the product inside Infineon.

* **Scope of the work**

The initial scope of the APDU fuzzer framework was just limited to our department CSS ESS

in Infineon. Also, it was being targeted for just a single product of our department initially.

Since, the APDU fuzzer tool was first one of its kind, it was being focussed at a small scale and then if successfully merged, it would be scaled for wider applications.

The reason it was targeted for a single product is that product would efficiently determine whether the APDU fuzzer tool follow the expected protocols and how much of a coverage it gives. Also, it helps in integrating the APDU fuzzer to the Infineon test framework which will further help it in scaling.

The product that was targeted is of our department and thus APDU fuzzer was initially focused for a single department i.e. CSS ESS in Infineon.

* **Importance and Applicability**

A brand should deliver what they have promised which should have minimum flaws and security parameters has been kept in mind. Therefore, it’s incredibly important to ensure that the brand is performing regular testing of its products to keep up with requirements and promises.

General importance of product testing:

* Measure the effects of aging on products. Accelerated life testing makes sure products can withstand the test of time and it is especially important for technology-based businesses.
* It can also monitor potential threats from competing products – testing can find any flaws in the product and compare its use in a variety of ‘real’ settings and environments.
* Product testing can also be applied to customers. Market testing falls under the same umbrella and with this, firms can predict consumer acceptance of any new products. This should determine whether an item will sell well or whether it still needs work before it hits the mass market.
* Safety. Product testing makes sure that an item is safe for general use. Consumers need to be able to rely on the company to deliver a high standard of safety in all sectors.

Testing has two main approaches. One is White-box testing and the other is Block-box testing.

Fuzz testing approach is a Black-box testing and thus APDU fuzzer is a Black-box test tool which would determine bugs and vulnerabilities that were not known before (if any) in Infineon’s product.

Black box testing is a technique of software testing which examines the functionality of software without peering into its internal structure or coding. The primary source of black box testing is a specification of requirements that is stated by the customer.

In this method, tester selects a function and gives input value to examine its functionality and checks whether the function is giving expected output or not. If the function produces correct output, then it is passed in testing, otherwise failed. The test team reports the result to the development team and then tests the next function. After completing testing of all functions if there are severe problems, then it is given back to the development team for correction.

**White Box Testing** is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers, so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing and Glass box testing.

The testing can be done at system, integration and unit levels of software development. One of the basic goals of Whitebox testing is to verify a working flow for an application. It involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output, you have encountered a bug

Product development takes time, creativity, resources and more time. One vital piece of the development process is product testing for quality, reliability and durability. You need to find out whether your product will do what it's supposed to do — that's quality. You also need to learn whether it will do what it's supposed to do repeatedly, even when conditions aren't ideal — that's reliability and durability.

Product testing is something that may be done either in-house (you or your manufacturer do the tests) or it's outsourced to an independent product testing agency. As much as you might think you want to be in control of the product testing, trust us: there are good reasons to outsource testing — chiefly cost. Maintaining a testing lab is expensive. It requires dedicated space, equipment (which must be maintained and repaired), upgrades, labour, training and certifications.

Types of testing include quality assurance, product reliability and lifecycle testing.

Testing is usually done during the development and manufacturing phases of product development. The outcome criteria are based on realistic expectations of customer use. Questions you need to ask include:

* How will this part or product be used?
* What sorts of pressures does it need to withstand - thermal, weight, etc.?
* How long should this product last?

The components that make up the product are selected based on the answers to the customer use profile. Perhaps the electronics or the raw material needs to be a certain grade or quality. Those items will be taken into account. Let’s say you’re making a medical device, for example. Components in medical devices must meet a higher standard than those going into a non-medical consumer electronics product.

During the manufacturing process, it’s part of the quality process to test a product’s functionality off the line. The quantity of items tested is determined in advance. It’s also wise to test your product at the system level as the components are only as good as the sum of all the parts working together. Determine what the end results should look like and design the test(s) to rigorously examine how well the product/part performs.

Unlike quality testing, the purpose of which is to find defects early in the product development process, the main reason for reliability testing is to measure performance and endurance over the long haul — preferably in similar operational environments.

Stress testing from the earliest product development phase is meant to find design weaknesses and specific assembly and material problems. **Highly Accelerated Life Testing (HALT)** and**Highly Accelerated Stress Test (HAST)**introduce the types of real-life problems a product may encounter during its lifecycle, without waiting the life of the product. For example, if corrosion could occur during the life of the product, the test would introduce corrosive material (salt water, for example) to the part/product to see if or how much the performance is affected.

Reliability testing is done for three specific reasons:

* To identify the failure rates of the product.
* If possible, to determine preventative measures that can increase the product's reliability and lifespan.
* To examine a product's endurance under certain conditions.

Typically, reliability testing will examine five areas:

* Vibration
* Temperature and humidity
* Time duration
* Number of test cycles
* Duration

* **Role and Profile**

My role was an intern in CSS ESS where I worked on APDU fuzzer tool project. There was only one intern per project being allocated which was same for me. I under the guidance of my mentor was working on the project. The profile was a mix of software and hardware. The understanding of hardware was at top priority and the implementation was done in software. The tool had to be created in a way that it easily integrates and follows the protocols of the hardware which is to be tested.

**BRIEF DESCRIPTION OF THE WORK DONE**

**CHAPTER 3**

* **Position of Internship and Roles**

My role was an intern in CSS ESS where I worked on APDU fuzzer tool project for one of their products. There was only one intern per project being allocated by the company which was the same case for me. I, under the guidance of my mentor was working on the project alone.

The profile was a mix of software and hardware. The understanding of hardware was at top priority and the implementation was done in software. The tool had to be created in a way that it easily integrates and follows the protocols of the hardware which is to be tested.

The work was properly time lined by my mentor. I was given work on that basis and a daily meeting were taken to check the progress. My role solely, was to report daily to my mentor and bi-weekly to the manager with the progress that I have made.

After the completion of the project, I had to demonstrate the demo of the work done to the member of the department and take up any feedbacks that they give. Also, if some requirement changes had to be made in the APDU fuzzer tool then they would tell it.

**BRIEF DESCRIPTION OF THE WORK DONE**

**CHAPTER 4**

* **Activities/Equipment handled**

There were a series of activities and training that were done in order for me to get started with the protocols and tool creation.

1. **Agile**

The Agile is one of the popular set of principles for product development and also a common standard for CSS ESS department in Infineon. Thus, I was expected to develop the tool following the same set of principles.

I was given a training of around a week on Agile, Scrum, Business Conduct Guidelines, Clean Desk Policy, POSH, and Intellectual Property. All these offer a professional support to carry out work responsibly in such security-concerned environment.

Agile and Scrum was the primary objective that was to be understood to continue work in the department.

1. **Test SW Architecture**

The very next thing that I had to get familiar with was the Test SW Architecture. This was the architecture that included the flow of commands, protocols and hardware. It discussed about what all components was to be taken care of and what was not accessible to me.

As per the discussion with my mentor, I had to understand the comms layer and crypto layer that was mention in the architecture. Software part was to be done from home because of the lockdown situation. The hardware and the related DLLs would be provided once I reach the office.

1. **OPTIGATM  Trust M**

A brief introduction to it was given by my mentor with its related information. Introduction like its architecture, endurance rating, security parameters, and communication protocols.

This is the specific product that was targeted initially for the APDU fuzzer framework.

I had to get familiar with the applications and components of the product. Provided that this product was targeted so understanding the protocols and various commands becomes important. Thus, the same was done from the solution reference manual.

I learnt about the format of APDU commands in it, I2C and IFX I2C protocols, various data structure in it, and how do they work. I had to learn about different Data format including TLV and CBOR.

A picture containing text, electronics

Description automatically generated

Diagram

Description automatically generated

1. **Testing and Security Implications**

This session taught me a lot of new stuffs. I got to know about different approaches of testing, both white-box and black-box testing and also had a practical implementation of it. How NUnit is used in Visual Studio was introduced and several exercises on Equivalence class partitions and Boundary Value Analysis was done.

The security implications of testing is solely to find bugs and vulnerabilities. One may think that software testing is done to make sure the product is working fine. This is just half true as per the white box approach which tests only one available set of known bugs.

But as per the Black box approach, the unknown or new bugs and found out. Which means software testing is done and should be done to find new bugs in order to avoid any vulnerability and loss to the company.

Thus, product testing technique becomes and efficient part of product life cycle and should not be skipped at any point. Testing is usually done during the development and manufacturing phases of product development. The outcome criteria are based on realistic expectations of customer use.

1. **SW and HW crypto**

When one think of cryptography, what comes in their mind is encoding and decoding. I was one of these people until this day of session. In a semiconductor industry hardware crypto is way more important that SW crypto as it is very tough to tamper it.

In this session, I basically learnt about how HW crypto works and what are its major benefits over the software crypto.

1. **Software Testing process**

This was the in-detail session that talked about minor to major points of software testing. This included reading research papers for the types fuzz testing, white-box testing, black-box testing and a scenario of PNG file fuzzer where the efficiency of two fuzzer were compared.

All these studies would be helping me in future to take correct decisions while coding the Fuzz framework or even choosing the correct technique to fuzz a target.

1. **Case Study**

A case study plays an important role when we want to understand a field in a practical way. In my case where I had to keep security in mind, I did 3 case studies to have a practical understanding of theorem and mistakes by big names.

* **Infinite Monkey Theorem**

This theorem said that if infinite monkeys are typing on a typewriter for infinite time then there is probability of 1 that whatever one has said, written or thought about will be present in the typed script.

* **Goto fail**

This was an epic mistake by the tech giant Apple Inc. This vulnerability lead bypassing certificate validation in Apple devices because of a single line of mistake: “goto fail;”.

* **Heartbleed**

The Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software library. This weakness allows stealing the information protected, under normal conditions, by the SSL/TLS encryption used to secure the Internet. SSL/TLS provides communication security and privacy over the Internet for applications such as web, email, instant messaging (IM) and some virtual private networks (VPNs).

The Heartbleed bug allows anyone on the Internet to read the memory of the systems protected by the vulnerable versions of the OpenSSL software. This compromises the secret keys used to identify the service providers and to encrypt the traffic, the names and passwords of the users and the actual content. This allows attackers to eavesdrop on communications, steal data directly from the services and users and to impersonate services and users.

1. **OOPs Concepts**

OOPs concepts and way more secure method of programming. This is much respected in Infineon and coding practices are done in this manner. A comprehensive training was given for about 3 weeks with different scenario-based questions to be implemented using OOPs concepts.

**Object-Oriented Programming System (OOPs)** is a programming concept that works on the principles of abstraction, encapsulation, inheritance, and polymorphism. It allows users to create objects they want and create methods to handle those objects. The basic concept of OOPs is to create objects, re-use them throughout the program, and manipulate these objects to get results.

1. **C#**

C# is the language that was chosen to continue with. I had to create the APDU Fuzz testing tool in this language. Along with the OOPs concepts, training of C# and clean-code practise was given.

After the OOPs implementations, various questions were given to implement using a combination of both OOPs and C#.

C # is pronounced "C-Sharp".

It is an object-oriented programming language created by Microsoft that runs on the .NET Framework. C# has roots from the C family, and the language is close to other popular languages like C++ and Java. The first version was released in year 2002. The latest version, **C# 8**, was released in September 2019.

1. **Crypto introduction**

In this session a detailed introduction to cryptography methods, tools and techniques were discussed. Not only cryptography but also, digital certificates and digital signatures.

The things that I learnt from this session:

* RSA encryption
* ECC encryption
* Brainpool curve
* Symmetric Cryptography
* Asymmetric Cryptography
* X509 Digital Certificate
* Digital Signature
* Key size and digital signature relation in terms of RSA and ECC
* Implementing digital signature calculation and verification using OpenSSL command line utility
* Created Digital Certificate using XCA tool
* Viewed Digital Certificate in ASN.1 Editor
* Created a certificate chain in XCA tool

1. **Bouncy Castle**

The Bouncy Castle library is an open source library for various purpose.

Example: certificate generation, certificate parsing, generation of keys, signing and verification of digital signature and also can be used to verify certificate chain.

Hands on this library was done to get familiar with open-source and how to use them for our purpose. Bouncy Castle was expected to be integrated to our APDU fuzz tool to verify the digital signature.

1. **PKI**

Public Key Infrastructure (PKI) is a technology for authenticating users and devices in the digital world. The basic idea is to have one or more trusted parties digitally sign documents certifying that a particular cryptographic key belongs to a particular user or device. The key can then be used as an identity for the user in digital networks.

The users and devices that have keys are often just called entities. In general, anything can be associated with a key that it can use as its identity. Besides a user or device, it could be a program, process, manufacturer, component, or something else. The purpose of a PKI is to securely associate a key with an entity.

The trusted party signing the document associating the key with the device is called a certificate authority (CA). The certificate authority also has a cryptographic key that it uses for signing these documents. These documents are called certificates.

Diagram

Description automatically generated

1. **APDU format**

This was one of the essential parts of the project as understanding the OptigaTM Trust M APDU format and IFX I2C protocol was the most complex ones. It took fair amount of time to get used to it. Also, various exercises were included on a daily basis for me to understand every part of it clearly.

1. **APDU Fuzzer framework creation**

I started creating the APDU fuzzer framework by using peach framework as a fuzz engine which is a grammar-based fuzzer. The work was started with a known set of requirements and some more requirements were added as the work proceeded and I analysed several fuzz engines for our Internship project work.

Considering the date of Viva i.e. 19th Dec ’20, a lot have been achieved in this framework and a lot is yet to be done.

1. **Demonstration**

The demonstration part to the department members is still pending as the framework is still being worked on.

**BRIEF DESCRIPTION OF THE WORK DONE**

**CHAPTER 5**

* **Challenges faced and how those were tackled**

Every new learning has some challenges involved but it is we who decide how to tackle them.

I have described mine below:

* The first challenge in Infineon as an intern was to get used to agile and scrum and sticking to it. Because of the help of my mentor and manager I got used to some parts of it. The other, I believe will be gained through experiences.
* There was a lot of learnings involved in the process till date. Everyday came up with a new challenge. The major one I remember is to decide about the fuzz engine to be used as a core in the APDU fuzzer framework. Within a period of 2-3 weeks, I tested 3 fuzz engines, read several articles and a couple documentations to reach to a conclusion.
* Working from home in this lockdown was another big challenge for both me and people at Infineon. As this was totally a new thing for both of us, things like poor internet and power cut sometimes made conversation and explanations difficult. But as days went by, we got used to it by not thinking about unable to access the in-person work environment.

**BRIEF DESCRIPTION OF THE WORK DONE**

**CHAPTER 6**

* **Outcomes**

Hard work surely pays off after the time has come. The similar is the case with me. The APDU Fuzzer framework was targeted for a single product and a small set of members of a department initially. But, being able to be used for other applications, the APDU Fuzzer framework will have applications more that just APDU fuzzing. It will be accepted throughout Infineon and everyone will be using it once it gets approved from higher officials.

The learning outcomes that I had from these 4 months are:

* Learned C# and OOPs concepts in conjunction with it
* Learning crypto and its several components
* Learned to use git for more purposes
* Learned to use open-source library like Bouncy-Castle
* Learned a lot of testing techniques
* Learned about APDU and IFX I2C protocol