

BRUNEL UNIVERSITY LONDON

**Secure Expandable Communication
Framework for POCT System
Development and Deployment**

by

Sivanesan Tulasidas

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
College of Engineering, Design and Physical Sciences
Department of Electronic and Computer Engineering

November 2018

Declaration of Authorship

I, Sivanesan Tulasidas, declare that this thesis titled, '**Secure Expandable Communication Framework for POCT System Development and Deployment**' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"Knowledge is like a river. Its nature is to constantly flow. Wherever it can flow, it does so, nourishing culture. On the other hand, the same knowledge, if devoid of values becomes a source of destruction for the world. When values and knowledge become one, there can be no more powerful instrument for the welfare of humankind..."

Sri Mata Amritanandamayi Devi

Abstract

Health-care delivery in developing countries has many challenges because they do not have enough resources for meeting the healthcare needs and they lack testing lab infrastructures in communities. It has been proven that Point-Of-Care (POC) testing can be considered as one of the ways to resolve the crisis in healthcare delivery in these communities. The POC testing is a mission critical processes in which the patient conduct tests outside of laboratory environment and it needs a secure communication system of architecture support which the research refers as **POCT system**

Almost every ten years there will be a new radio access technology (RAT) is released in the wireless communication system evolution which is primarily driven by the 3GPP standards organisation. It is challenging to develop a predictable communication system in an environment of frequent changes originated by the 3GPP and the wireless operators. The scalable and expandable network architecture is needed for cost-effective network management, deployment and operation of the POC devices. Security mechanisms are necessary to address the specific threats associated with POCT system. Security mechanisms are necessary to address the specific threats associated with POCT system. The POCT system communication must provide secure storage and secure communication to maintain patient data privacy and security. The Federal Drug Administration (FDA) reports the leading causes of defects and system failures in medical devices are caused by gaps between the requirements, implementation and testing.

The research was conducted, and technical research contributions are made to resolve the issues and challenges related to the POCT system. A communication protocol implemented at the application level, independent of radio access technologies. A new methodology was created by combining Easy Approach to Requirement Specifications (EARS) methodology and Use Case Maps (UCM) model which is a new approach and it addresses the concerns raised by the FDA. Secure cloud architecture was created which is a new way of data storage and security algorithms models were designed to address the security threats in the POCT system. The security algorithms, secure cloud architecture and the communication protocol coexist together to provide **R**adio access technology **I**ndependent **S**ecure and **E**xpandable (**RISE**) POCT system.

These are the contributions to new knowledge that came out of the research. The research was conducted with a team of experts who are the subject matter experts in the areas such as microfluidics, bio-medical, mechanical engineering and medicine.

Acknowledgements

I am very grateful to the invisible force which resides in all human beings who have inspired to undertake the research work and provided me with the opportunity and the capability to proceed successfully. This thesis would not have been completed without help and support I received from a number of people inside and outside the School of Engineering and Design.

First of all, I would like to thank my first supervisor, Professor Wamadeva Balachandran, for his support and time he spent in guiding me throughout the research work. I am very grateful for having Professor Balachandran as my primary supervisor. I would like to thank my second supervisor, Dr Rajagopal Nilavalan, for his help and discussions on the interim assessments and processes.

I am indebted to many of my colleagues for their help and support. I would like to thank Dr Nadarajah Manivannan, Dr Ruth Mackay, Dr Branavan Nehru, Dr Chris Hudson and Dr Pascal Craw for their valued technical discussions, which have always been helpful to me.

I am also grateful for the assistance provided by Angel.Naveenathayalan at the department for valuable test data.

I am very appreciative of the document review service provided by Dr Jeanne McNett who helped to review my thesis.

I am very grateful to my business coach Vivek Agarwal for his coaching in managing the daily work, research work and life balance.

I am fortunate to have a supporting family, my spouse Jaspal and my aunt, Devi Mahadeva for moral and practical support.

I have to thank Jaffna Hindu College teacher, Francis master for allowing me to stay in his house during my research work during writing papers and developing communication protocols.

I am indebted to my Tabla music teacher Sri Arshad Syed for teaching me the complex compositions for developing encryption algorithms.

Finally, I thank my late mother Kamla Theavi for guiding me during trial and tabulation throughout the research work.

Contents

Declaration of Authorship	i
Abstract	iii
Acknowledgements	iv
List of Figures	x
List of Tables	xiii
Abbreviations	xiv
1 Introduction	1
1.1 Background	1
1.1.1 POCT devices in the industry	5
1.1.2 POCT devices developed by DoC Lab at Brunel University	7
1.1.3 POCT device system architecture	9
1.2 Aims and Objectives of Research	12
1.3 Summary of Methodology used	12
1.3.1 Example of an end-2-end system configuration	18
1.4 Thesis structure	21
1.5 Contribution to knowledge	23
1.6 List of Publications	28
2 Requirements for POCT system communication and Architecture	30
2.1 Introduction	30
2.2 POCT System Description	32
2.3 POCT System Requirements	33
2.3.1 Requirements for P-Node operation	33
2.3.2 UCM representation of P-Node operation requirements	34
2.3.3 Requirements for P-Node construction and system interconnection	35
2.3.4 Requirements for P-Node fail safe mechanism	36
2.3.5 Requirements for POCT system data storage	36
2.3.6 G-Node Requirements	38
2.3.7 P-Cloud Requirements	40

2.3.8	POCT System Security Requirements	43
2.4	P-Node Architecture Overview	44
2.4.1	M1: CMD module	45
2.4.2	Interface types:	46
2.4.2.1	CMD layer 1: Interface Module:	46
2.4.2.2	CMD layer 2-3: CMD interpreter and Interface Policy . .	47
2.4.2.3	CMD layer 4: Security validator	47
2.4.3	M2: CNTL module:	47
2.4.4	M3: Communication module:	47
2.4.5	M4: OMAP module:	48
2.4.6	M5: Error handling module:	48
2.4.7	M6: Local DB handler:	48
2.4.8	M7: External DB connectivity module:	48
2.4.9	M8: Reporting module:	49
2.4.10	M9: Security module:	49
2.4.11	M10: Device health monitoring module:	49
2.4.12	M11: Power Management module:	50
2.4.13	M12: Product Evolution module:	50
2.5	Protocol for POCT system communication Open Communication Protocol	50
2.5.1	PKT (Packet) type	52
2.5.2	Interface type	53
2.5.3	Operation Code	54
2.5.4	Type of Service (TOS)	54
2.5.5	Security Information	55
2.5.6	Data Length, Protocol Version, Data bits (bytes)	58
2.5.7	MSC for CMD communication	58
2.5.8	PROTOCOL EXPANSION STRATEGY	60
2.6	Experimentation with Android-based smartphone and Bluetooth	61
2.7	Summary	64
3	POCT Communication System Architecture Design Methodology and	
	Processes	68
3.1	Introduction	68
3.2	Pyramid Requirement of Eliciting Process	70
3.2.1	Issues with current process	70
3.2.2	Pyramid Requirement Model	72
3.3	System Design Process and Techniques	74
3.3.1	Basic Building Blocks Identification	74
3.3.2	Isolating System Module Responsibilities	76
3.3.3	Creating loosely coupled modules	77
3.3.4	Actuation Channels and Actuation Confirmation Channels	79
3.3.5	Security Principles for Designing Safety-Critical Software	84
3.4	P-Cloud Architecture for Delivering an Error Free System	86
3.5	Benefits of adopting IEC 62304 at early stages of development	90
3.6	Using Combinatorial Design Methodology for Generating Smart Testing	
Vectors	94
3.6.1	Application of combinatorial testing process in POCT system	96

3.7	Agile Project Management Approach in Developing POCT systems	100
3.7.1	Project Management in IEC62304	100
3.8	Summary	100
4	Security Framework for Managing Data Security within Point of Care	
	Tests	103
4.1	Introduction	103
4.2	POCT System Configurations	107
4.2.1	Architecture Layering	107
4.2.2	Security Layering	108
4.2.3	Definition of Asset in POCT system	110
4.2.4	Attack tree for POCT System	110
4.3	Security Compromising Scenarios	113
4.3.1	Processing Data in POCT devices	115
4.3.2	Access from multiple devices (Preventing Unauthorized Access) . .	116
4.3.3	User authentication	116
4.3.3.1	User validation and User Identification	116
4.3.3.2	Authentication using biometric	117
4.3.3.3	Authentication using Active Directory	117
4.3.3.4	Clinical and Surveillance data collection	117
4.3.3.5	Safely dispose of the testing device	118
4.3.3.6	Mechanical method	118
4.3.3.7	On device usage prevention using SW	119
4.3.3.8	More complex and fail-proof mechanism for preventing (or restricting) device usage	119
4.3.4	Web-based POCT System access	119
4.3.5	Cross-site scripting	120
4.3.6	SQL Injection	121
4.3.7	XML Injection	121
4.3.8	Client-side attacks	122
4.3.9	Malware attacks	123
4.3.10	Cookie and Attachment Threats	123
4.3.11	Denial of Service (DoS)	125
4.4	Security Framework	126
4.4.1	The protocol used between the G-Node and the P-Node	126
4.4.2	Security Mechanism -1 (challenge and response based)	127
4.4.3	Security Mechanism -2 (behavioural based)	128
4.5	M2M (Machine-to-Machine) Security in POCT System	131
4.5.1	Trust Relationships between POCT System entities	132
4.5.2	Core security requirements for POCT M2M	134
4.5.3	Bootstrapping requirements for POCT device deployments	135
4.6	Summary	135
5	P-Cloud and NAS Implementation	138
5.1	Introduction	138
5.2	Configuration database	142
5.3	Measurement database	143

5.4	Data mining (Analytics) database	144
5.5	Execution (operational) database	144
5.6	Deployment database	146
5.7	NAS implementation of private cloud	147
5.7.1	Need for Private Cloud	147
5.7.2	NAS Configuration	148
5.7.3	Validation Methodology	150
5.7.4	DDDNS SERVICE	153
5.7.5	Access POCT site from Internet (WAN side access)	153
5.7.6	Access POCT site via an internal IP address? (LAN side access)	154
5.7.7	Connectivity between G-Node and NAS	155
5.7.8	Strategies for data transfer from POCT sites	156
5.7.9	NAS QoS (Quality Of Service)	157
5.7.9.1	Traffic control process in NAS	157
5.7.9.2	Guaranteed Bandwidth	160
5.7.9.3	Maximum Bandwidth (BW)	160
5.7.9.4	Speed limits process in NAS	160
5.8	Conclusion	161
6	Network Models for POCT system deployment	163
6.1	Introduction	163
6.2	Basic connectivity model	165
6.3	Possibilities for connectivity	166
6.3.1	Direct connectivity between local server and both P-Node and G-Node	168
6.3.2	Direct connectivity between local server and P and G Nodes	169
6.3.3	Standalone model	171
6.3.4	Alternate Hybrid model	171
6.3.5	Standalone Hybrid model	172
6.3.6	Use of well-known secure enterprise server infrastructures	173
6.3.7	Using secure smartphone device as G-Node with secure enterprise server infrastructure	175
6.4	Definition of the POCT system and communities	176
6.4.1	Connecting POCT communities and P-Cloud	177
6.4.2	Realising POCT System	177
6.4.3	POCT Units	178
6.4.4	POCT Site	180
6.4.5	POCT Zone	181
6.4.6	POCT System	182
6.4.7	Capacity of POCT System - Honeycomb compatible communication links	183
6.4.8	Multiple POCT Systems	184
6.4.9	Interconnecting Multiple POCT sites to a P-Cloud	184
6.5	Development and maintenance strategy for POCT system	186
6.5.1	Interconnecting development sites	187
6.5.2	Business units needed for POCT development ecosystem	188
6.6	Connecting commercially available Cloud services with POCT	190

6.6.1	With commercial Cloud	190
6.6.2	With commercial Cloud and local private Cloud	191
6.6.3	Useful configuration of G-Node for connecting to commercial Clouds	191
6.7	Simulation of key network models	193
6.7.1	Simulation of Point-to-Point data traffic between P-Node and G-Node	193
6.7.2	Simulation details	194
6.8	Collaborative congestion control management	196
6.8.1	Congestion control algorithm	203
6.9	Summary	204
7	Conclusion and Future work	206
7.1	Conclusion	206
7.2	Future Work	212
7.2.1	Data aggregation of multiple P-Clouds	213
7.2.2	Architecture for POCT Mobile station unit and data privacy	215
7.2.3	Modification and expanding POCT protocol with USSD	218
7.2.4	Developing common data standard for multi-vendor P-Clouds	218
7.2.5	Security algorithms for dynamic adaptation	219
7.2.6	Connected homes and the POCT systems	219
	Bibliography	222

List of Figures

1.1	Schematic diagram of the Brunel POCT architecture [1]	9
1.2	High-level System abstraction view	13
1.3	Standard V-Model Overview [2]	14
1.4	V-Model developed for research	16
1.5	Functional Safety V-Model Overview	17
1.6	Security V-Model Overview	18
1.7	End-2-End Connection Topology	19
1.8	Technical contributions (indicated as <i>1: P-Node architecture, 2: G-Node to P-Node communication, 3:End-to-End communication and 4:P-Cloud</i>)	25
1.9	Hierarchical model for POCT system management	28
2.1	Basic building blocks	33
2.2	UCM representation of P-Node communication links	34
2.3	Identification of failure points	37
2.4	UCM for representing G-Node requirements	39
2.5	P-Cloud requirements	42
2.6	MUCM Example	43
2.7	Modularize framework	44
2.8	CMD Module in detail	45
2.9	Communication protocol	51
2.10	Core pseudocode for data TX and RX	57
2.11	Data communication Message Sequence Chart	59
2.12	Protocol evolution	60
2.13	System setup	61
2.14	Arduino Uno and BT module	62
2.15	TX Window and RX window for protocol development - QT based IDE (Protocol Development Utility Tool)	63
2.16	Protocol communication Data reception from POCT device)	64
3.1	Sources of errors and phases in development cycle [3].	71
3.2	Problem-Pyramid [3]	72
3.3	Basic building blocks of POCT System	75
3.4	Isolation of Functionalities and communication failure points	76
3.5	End-to-End Topology (external and internal communication links)	80
3.6	Algorithm for Peripheral Activation	81
3.7	Conceptual connection of ACCH and ACCH to stepper motor	83
3.8	P-Cloud data organization	87
3.9	IEC 62304 process [Standard: Introduction]	90

3.10	Architecture Partitioning and IEC62304 Classification	92
3.11	System under test	96
3.12	UI (User interface) of Pairwise tool [4]	96
3.13	Input data model [4]	97
3.14	Seeding Process [4]	98
3.15	Test Vector Generation [4]	99
3.16	Coverage Analysis [4]	99
4.1	Device usage categories	105
4.2	High-Level architecture layers	108
4.3	Secure layering of the basic architecture	109
4.4	Threat path tree (Attack tree)	110
4.5	Port scanning setup	112
4.6	Configuration of Port scanning tool	112
4.7	Scanned Results	113
4.8	Embedded Web server with POCT	120
4.9	Distributed Denial of Service (DoS) Attacks	125
4.10	Interaction of messaging mechanism	127
4.11	Interaction summary	128
4.12	Security Mechanism: Behavioural-based	129
5.1	Accessing P-Cloud via G-Node	138
5.2	Accessing P-Cloud via P-Node	139
5.3	Hybrid configuration model	140
5.4	Five types of P-Cloud Databases (with direct connection from P-Node)	141
5.5	Four types of human users	142
5.6	Configuration database segment - (part of P-Cloud, Fig 5.4)	143
5.7	Measurement database segment - (part of P-Cloud, Fig 5.4)	143
5.8	Analytics database segment - (part of P-Cloud, Fig 5.4)	144
5.9	Execution database segment - (part of P-Cloud, Fig 5.4)	144
5.10	Deployment database segment - (part of P-Cloud, Fig 5.4)	146
5.11	Architecture of NAS as Private Cloud	149
5.12	Physical topology for interconnecting NAS, G-Node, and P-Node	150
5.13	Test Data to be stored in the NAS	152
5.14	Accessing data from P-Node	152
5.15	Port forwarding setup at the router	153
5.16	Creation of Brunel-doc-lab.sinology.me	154
5.17	Login screen for accessing NAS outside of the POCT site via DDDNS	154
5.18	Login screen for accessing NAS using LAN address	155
5.19	One BAY was used for the experimentation	155
5.20	Multiple Bays system	156
5.21	Synology NAS	156
5.22	USB connection between NAS and G-Node	156
5.23	Control Panel Firewall and QoS	158
5.24	Specific port traffic with guaranteed and maximum bandwidth	159
5.25	Built-in applications	159
5.26	BW equation	160

5.27	Speed limit process	161
6.1	Fundamental connectivity Building blocks	165
6.2	Multiple ways to connect P-Cloud	166
6.3	Data send to local server	168
6.4	Data send to local server	169
6.5	P-Node as data gateway	170
6.6	Hybrid data gateways (Load balancing model)	172
6.7	G-Node as gateways and P-Node as gateways	173
6.8	BIS (Blackberry server for connecting to P-Cloud)	174
6.9	Using secure smartphone as G-Node	175
6.10	POCT Communities: Type 1, Type 2 and Type 3	176
6.11	Connecting to P-Cloud with multiple types of POCT systems	178
6.12	POCT System Hierarchy	179
6.13	Definition of POCT unit	180
6.14	Definition of POCT site	180
6.15	Definition of POCT Zone	181
6.16	Definition of POCT System	182
6.17	A complete POCT System (Capacity Model)	183
6.18	Multiple POCT systems map	185
6.19	MPLS based interconnection of POCT sites	186
6.20	Development and system maintenance ecosystem	187
6.21	POCT organization for Business	188
6.22	Commercial Cloud and G-Node	191
6.23	Commercial cloud and NAS as private cloud	192
6.24	Commercial cloud and NAS as private cloud	192
6.25	Throughput plot	195
6.26	Packets sent to G-Node	196
6.27	Packet loss (percentage)	196
6.28	Data and Control Channels	197
6.29	P-Cloud configuration in detail (expanded version of Figure 6.30)	198
6.30	P-Node configuration matrix	201
6.31	Message sequence chat for collaborative communication	202
6.32	Control Status Word for P-Nodes	202
6.33	Sending single transmission indication byte	202
6.34	Flow chart for congestion control	203
7.1	Policy framework engine	213
7.2	Lancet-2014 report	215
7.3	Bearer Services for POCT	216
7.4	IP address usage in African countries	217
7.5	NAT Based Solution	218

List of Tables

1.1	Mapping between chapters and specific research goals	24
2.1	EARS pattern [5]	32
2.2	Preamble and End Data Segments	55
2.3	Protocol Command mapping to OPCODES	58
3.1	Application of Pyramid Process	73
3.2	Example of ACH and ACCH	83
3.3	P-Cloud and Access Gateways	89
3.4	Benefits of Combinatorial Testing Process	95
4.1	Multiple Device Access	116
4.2	Clinical and Surveillance	117
4.3	Main stakeholders and security relationships	133
5.1	Architecture Artefacts	142
5.2	Connectivity paths	149
5.3	Sample Data from POCT device [1]	151

Abbreviations

LAH	List Abbreviations Here
3GPP	Wireless Modem Standard Organization
ACCH	Actuator Control Channel
AD	Active Directory
AKAMAI	Referring annual reports on Internet by AKAMAI
AI	Artificial Intelligence
AP	Wireless wireless Access Point
ASIC	application-specific integrated circuit
BAY	Slots for installing hard disk in NAS
BES	Blackberry Enterprise Server
BIS	Blackberry Internet Server
BW	Bandwidth
AD	Active Directory
BT	BlueTooth wireless connection
BlueTerm	Terminal mode Android application
CDMA	Code Division Multiple Access - Radio Access Technology
CSCN	Conference on Standards for Communications and Networking
CNTL	Control
DDDNS	DNS Service from NAS vendor
DDoS	Distributed Denial Of Service
DHCP	Dynamic Host Configuration Protocol
DM	Device Management
DOCRES	User defined protocol preamble
DSM	Disk Station Manager - NAS
EARS	Easy Approach to Requirement Syntax

EDGE	Radio Access Technology
EMI	Electromagnetic interference
End2End	End to End system
ETSI	European Telecommunications Standards Institute
FAM	Fluorescein dye for DNA sequencing
FCC	Federal Communications Commission
FDA	Food and Drug Administration
FEMA	Federal Emergency Management Agency
FR	Functional Requirements
GERAN	GSM/EDGE Radio Access Network
GPRS	General Packet Radio Service packet data service for 2G and 3G
GSM	2G Radio Access Technology
GP1	Security Group 1
GUI	Graphic User Interface
GW	Gateway
HIPAA	Health Insurance Portability and Accountability Act
HTTPS	Secure HyperText Transfer Protocol
I2C	Inter-integrated Circuit (I2C) Protocol
IDE	Integrated Development Environment
IEC62304	International standard for medical device software software life cycle processes
IMEI	International Mobile Equipment Identities
IoBT	Internet of biometric things
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
KPIs	key performance indices
LAN	Local Area Networks
LTE	4G Wireless Access Technology
M2M	Machine-to-Machine
MDM	Mobile Device Management
MODEM	Communication processor - Modulation and Demodulation
MPLS	Multiple Protocol Label Switching
MSC	Message Sequence Chart
MTC	Machine type Communication

NAS	Network-attached storage
NAT	Network address translation
NBIoT	Narrow Band IoT
NS3	Network Simulation Tool Version 3
OEMs	Other Equipment Manufacturers
OMA	Open Mobile Alliance
OPCODE	Operation Code
OS	Operating systems
OSI	Open System Interconnection
OTA	Over the Air
P2P	Point to Point Connectivity
P-Cloud	Cloud storage for POC Test data
PKT	Data Packet
P-Node	POCT Device
POCI	POCT Device as interface to other entities
POCM	POCT as monitoring device
POCT	Point of Care Testing
QoS	Quality Of Service
QT	Nokia's UI development framework
SCS	Safety Critical System
SCSP	Smart City Security and Privacy
SIT	System Integration Testing
SLA	Service Level Agreements
SOUP	Software with an Unknown Pedigree
SIM	Subscriber identity module
SYNOLOGY	NAS Vendor
TCP	Transmission Control Protocol
TOS	Type Of Service
UCM	Use Case Maps
UDP	User Datagram Protocol)
UICC	Universal Integrated Circuit Card
UMTS	3G / WCDM Radio Access Technology
UTRAN	Universal Terrestrial Radio Access Network

USSD	Unstructured Supplementary Service Data
WAN	Wide Area Network
WCDMA	3G / UMTS Radio Access Technology
WEM	wristband exercise communication module
XML	eXtensible Markup Language
XSS	cross-site scripting attack
ZigBee	an IEEE 802.15.4-based communication protocol specification

Dedicated to my loving late mother

Kamala Theavi Ramalingam. . .

Chapter 1

Introduction

1.1 Background

In the public health and medical communities, there has been a widely recognized need for monitoring and early diagnosis of infectious diseases in low-resource, low-income developing countries [6]. Such healthcare delivery strategies contribute to patient management by decreasing the testing and diagnosing times. In low-resource, low income environments, diagnostic infrastructures are not present. The analysis of the diagnosis needs be conducted at a site close to patients so that the analysis results can be communicated to the patient quickly, in order to manage cures for infectious diseases. The World Health Organization (WHO) has developed an operational terminology for quality healthcare delivery called Affordable, Specific, User-friendly, Rapid, Equipment-free, Delivered to those in need, (ASSURED). ASSURED guides health-care providers in creating healthcare delivery strategies that provide 85-95 percentage analytical performance values in sensitivity and specificity, which will increase the healthcare management in developing countries greatly [6, 7]. The monitoring and early detection of biological

entities causing the infectious diseases is vital for effective health-care management and saving lives. An autonomous and portable detection system that can process assays and analyze deficient concentration samples are needed to comply with the WHO's mandate for healthcare delivery in developing countries with low resources and many isolated communities. This critical issue—the need for diagnosis in such environments to control the spread of infectious diseases—is the main driver for developing a health delivery system that is capable of executing the assays to detect infectious diseases.

Commercially available testing tools for infectious diseases are insufficient to meet the needs of the limited-resource setting or poor infrastructure-based healthcare deployments. Improved health in such environments would help the country prosper economically and socially. In contrast to these developing country scenarios, diseases such as tuberculosis, malaria, HIV/AIDS and other sexually transmitted diseases are preventable and treatable in the developed world. Developing countries are still under enormous pressure to provide healthcare solutions for diagnosing diseases, in spite of technological advancements available in the developed world. In these environments, accessibility to laboratory facilities often is not possible for rural patients, who often lack access to basic testing tools and trained healthcare personnel [8].

Traditionally, rapid tests are often referred to as Point-of-Care (POC), because they are done by doctors or nurses in the clinic or the bedside. POC has been an evolving field of medicine and technology over the past few years. The most modern form of these diagnostic tests uses micro- or nanotechnologies which are in effect miniature laboratories that allow complex biological reactions, which usually take a long time in a full-sized laboratory, to happen very rapidly.

The Point-of-Care Testing (POCT) based diagnosis is a method for bringing medical laboratories to a patients home to conduct diagnostic tests (self-tests) so that the patient does not need to go to the doctor or laboratory in person. WHO envisions POCT as a way of meeting the complex and critical requirements for delivering health-care to remote sites where healthcare delivery has challenges. Such a revolutionary technology helps to provide affordable, accessible solutions. In order to be successful, the system must be available to local communities to conduct the tests for detection of infectious diseases. The collected test data then must be transmitted securely to a secure database for analysis by physicians that will lead to a diagnosis. An automated response can be set if all the stakeholders accept a hybrid diagnosis process. The secure test data must reach clinician/healthcare personnel without delay or inaccuracy. Such systems can be designed to provide low-cost detection of analytics even with low concentration samples, and they can run required assays faster than laboratory-based testing. These systems lead many who are concerned about health care in developing countries to hope that finally there is a way to provide quality, affordable health care for people in those countries.

The design and development of the POCT system and the associated security for communication infrastructure must be done with extreme rigour, as the system is mission critical and safety critical. Detection of mission anomaly is meant for ensuring the functional safety, an important aspect of the POCT. Diagnosis based on this testing is an essential process improvement for remote patient care management. With the security of all communications assured, physicians gain a tool to diagnose patients who previously could not afford or could not access reliable health care.

POCT device-based diagnostics is a medical tool composed of interconnected devices that can provide health care diagnostics in a patient's community environment, external to a hospital-based lab setting [8, 9]. This testing has many advantages compared to standard laboratory-based testing [10]. It provides faster diagnosis, reduces cost, and introduces automation, which provides the patient access. It plays a key role in e-health delivery. The advantages in microbiology and engineering, when linked with the advances in mobile computing technologies, have enabled the possibility of patients being diagnosed and treated away from the clinic. There is an increasing recognition that these m-health (mobile health) and e-health (electronic processes and communication) technologies will play a greater role in medical care. The present research will drive the implementation of digital health care.

The potential benefits of using the POCT device include access to robust data, a short turnaround time (TAT) for testing, portability (ideally handheld), affordability, accelerated clinical decision-making, and avoidance of the complex pre-processing of biological samples. It is less complicated to use than traditional testing, as it is controlled mostly by a wireless smartphone or a one-touch keypad. Unlike with Lateral Flow Immunoassay (LFIA) technology, with microfluidics-based molecular diagnostic technologies used in the POCT-device, required sensitivity and specificity can be achieved. Sample (or reagent) volume reduction is achieved because of the reduction in surface-to-volume ratios. Highly reproducible and quantitative results can be accomplished with the microfluidic-based system, as well. It is a closed system that can be operated with minimal wireless communication infrastructure, depending on the need [8], and expanded to a multilayer and multiuser system as needed.

1.1.1 POCT devices in the industry

The scope of the research is not developing a new device for Point of Care testing. The scope of the research study is to research into communication and data security mechanisms for the POCT systems. This includes working closely with bio-medical experts, POCT assays developers and mechanical engineers in the University research group and St. Joseph's hospital medical community who understands patient's needs.

The following paragraphs provide an overview of the existing POCT devices. NHS provides buyers guidelines for four kinds of POCT devices: Cholesterol measurement, Urine monitors for Diabetes, Blood Pressure Monitors and Blood Glucose systems [11]. The following POCT device manufacturers are discussed in the NHS report. Abbott Diabetes Cares is a medical company that is dedicated to developing sensor glucose monitoring and testing systems. They are word number one diabetes care providers [12]. ACON laboratories provide a wide range of medical solutions for POCT type of devices including Diabetes Care, Clinical Chemistry including urinalysis and Immunoassay Enzyme immunoassay (EIA)/enzyme-linked immunosorbent assay (ELISA) for the International market [13]. Bayer Diabetes Care is a medical service organisation for diabetes care management. They use devices manufactured by other medical device other equipment manufacturers (OEMs) [14]. BBI Healthcare provides products and services that are provided to the diagnostic, healthcare, research, defence, food and cosmetics industries globally [15]. Cambridge Sensors Limited designs and manufactures blood glucose test strips for use in blood glucose monitoring systems including sensors, microstructures, immunoassays, diagnostic tests,

and micro-circuits [16]. HemoCue - part of the Radiometer Group, are being used in more than 130 countries, provides POCT care for various clinical areas all over the world. They are actively working on solutions that can be classified as point-of-care testing solutions in the areas of diabetes management, centrifugation products, occult blood testing and data management and connectivity solutions [17]. LifeScan is a diagnostic systems manufacturer with products focusing on the diabetes market, specifically blood glucose monitoring systems [18]. Medtronic plc is a medical device company, offers a wide range of medical instrumentation solutions: Cardiac rhythm disease management, Spinal and biologics, Cardiovascular, Neuro-modulation (drug delivery system for chronic pain, common movement disorders and urologic and gastrointestinal disorders), and diabetes management[19]. Menarini diagnostics provide specific POCT system for veterinary health care providers and schools [20], [21]. Nova Biomedical is a world leader in the development and manufacturing of state-of-the-art, whole blood, point-of-care and critical care analysers, as well as providing the biotechnology industry with the most advanced instruments for cell culture monitoring. It gives blood testing analysers and diagnostic products for healthcare companies. Besides, it provides hospital glucose monitoring systems, hospital glucose/ketone monitoring systems, point-of-care whole blood creatinine and estimated Glomerular filtration rate (eGFR) testing products [22]. Roche Diagnostics Corporation offers support for healthcare providers in the prevention, diagnosis, and management of various diseases. The company provides assays/reagents, instruments/systems, laboratory systems and automation products, QC/calibration products, and biosensors. It gives the point of care

testing products, such as blood glucose, anticoagulation management, blood gas and electrolyte, cholesterol, urinalysis, and infectious disease products; and molecular products, including systems, donor screening products, and automation products. The company offers its products for various disease states, such as HPV, HIV, heart failures, and diabetes, as well as for other medical conditions [23]. Point Of Care Testing Ltd., distributes point of care diagnostic instruments and consumables for medical and research markets in the United Kingdom and Ireland. It offers portable blood and urine analysis systems for use in human patient-care setting to provide clinicians with blood constituent measurements [24].

1.1.2 POCT devices developed by DoC Lab at Brunel University

The POCT system research that is being undertaken by the Doc Lab research group [1] at the Brunel University London has developed handheld POC device which accepts raw samples from a person and provides the disease-specific diagnosis. Unlike the standard POCT device categories listed by the NHS [11], the research goals of the Doc Lab are to produce automated POC device that is capable of accepting multiple raw sample inputs (blood, urine, saliva, swabs), detection of various pathogens within predefined process time (less than 30 mins) [8]. These devices are typically equipped with the disposable and closed loop microfluidics cartridges.

The system has a modular architecture which allows the system partitioning based on the IEC 62304 standard [25] for medical device development [26]. Besides, the modular process enables any section to be replaced by an alternative method. The communication

methods can be selectable based on the available radio access technology during deployment [8]. The Doc Lab produced a robust, low-cost POCT platform for isothermal nucleic acid amplification on a microfluidic device. The bill of materials is kept inexpensive by using readily available materials locally and commercial off-the-shelf components[27]. Because of the use of natural construction methods, the construction of the device can be operated and deployed in resource-poor settings, especially in the development world. The platform uses a nucleic acid amplification test (NAAT) which offers a potential transformation for clinical and public health medicine. On the whole, the platform provides low-cost instrumentation for high-quality diagnostics.

The communication protocol used to trigger the assay process has the potential of adopting multiple assay types for detecting many infectious diseases in a single POC device. The cloud architecture supports the HIPPA compliance [28]recommendation. Also, the communication protocol has provisions to assign communication priorities for the assays used in the machine. The sample preparation, isothermal amplification and detection types can be altered at the device level. The standard POCT devices in the market are mostly considered as bench-top devices and require some hands-on sample preparation[8]. The cost of the benchtop devices is high, within this paper the development of a prototype, handheld, low-cost amplification and detection platform that cost less for the parts.

1.1.3 POCT device system architecture

The interdisciplinary Electronic System Research Group at Brunel University has developed a modularized system architecture for operating a POCT device using microfluidics technology [29]. The device uses lab-on-chip technologies for diagnosis of infectious diseases. A related project called the Electronic Self-Testing Instrument for Sexually Transmitted Infections (eSTI2) was funded by the Medical Research Council. eSTI2 consortium is a multidisciplinary, multi-institutional entity of scientists, clinicians, public health experts, engineers, software specialists and industry specialists who were working together to develop new types of tests for sexually transmitted infections and other infectious diseases, that can give accurate results immediately and also communicate seamlessly with hospital computing systems through wireless smartphone technology.

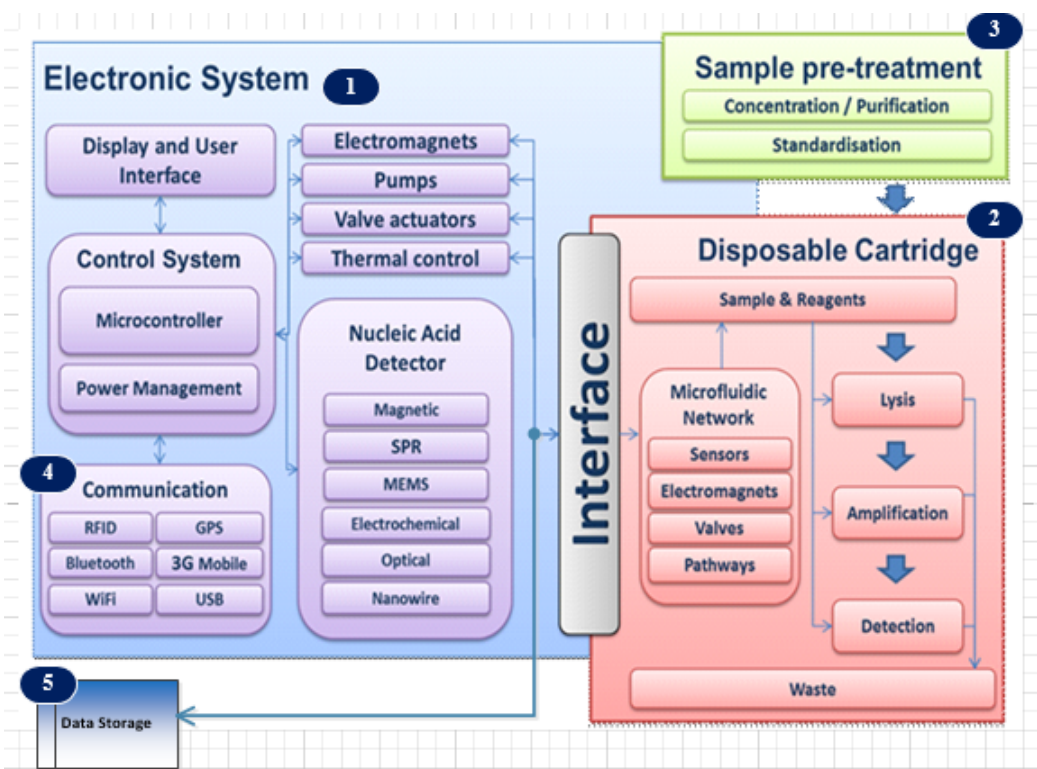


FIGURE 1.1: Schematic diagram of the Brunel POCT architecture [1]

Figure 1.1 [1] shows the generic device architecture needed for the POC testing that utilizes lab-on-a-chip technologies. The device operation is divided into three core functionalities. Sample pretreatment (or sample preparation (3)) is the first functionality for increasing the concentration, where the sample is introduced into the device with reagents (2). Reduction of surface to volume ratio enhances aspects of assay kinetics, per test [30]. After the first introduction of the sample into the device, it is be processed which may involve micro-mixing, heating and separation for extracting DNA molecules using cell lysis process. The released DNA molecules will be separated from the rest of the cell debris. The separated DNA molecule will go through an amplification process (second core function) to get the required concentration to facilitate the detection of the DNA molecules.

The next process is the nucleic acid detection (third core function) using bio-sensing technologies. Achieving sample-in-answer-out in the device is possible by having the capacity for inclusion of sequential sample preparation steps, nucleic acid amplification, and DNA detection in an integrated manner [30]. All three core functionalities, are aided by the microfluidic network. The microfluidic network operation is managed by the electronic control system (1). The electronic system enables user interface and display units as well. In addition these devices are capable of multiplexed analytic detection, because of the inherent design. Power management is necessary because the device needs uninterrupted operation during the process. All the processes are controlled by the smartphone, and the results are communicated in digital form to outside world. The smartphone communicates with the device via the communication module (4) within the device. The test data is tagged with a location attribute

for data analytics. The collected data is stored in a local storage unit such as an SD card.

The communication subsystem (4) that is part of the electronic system provides connectivity for transferring test data securely to either an internal private cloud in a hospital / clinic environment or an external public cloud. The locally stored data (5) can be sent to a central data cloud for actual diagnosis by health care professionals. The communication system provides short-range wireless radio access technologies such as Wi-Fi, Bluetooth, cellular wireless radio connectivity, and wired USB and LAN links. The cellular network radio access technologies such as 3G, 4G or 5G can help to establish connectivity between the device and other external entities such as the cloud.

The communication links within the device as well as external to the device are important to ensure the mission critical operations are carried out without any failure. The external communication links must be highly secure to preserve patient confidentiality. The test data collected must not be compromised by any security breaches. Therefore sophisticated, secure encryption schemes must be deployed for safeguarding the patient data at all cost. Accessing the device and initiating the detection tests must be possible only for authorized users. Unauthorized operations of all kinds must be prevented by the security mechanisms activated in the system. The test data can only be accessible via security-enforced communication links, including data moves within the device or external to the device such as system log dumps to a server and other types of data storage.

1.2 Aims and Objectives of Research

The goal of the project is to develop and evaluate communication pathways, data security mechanisms, and a scalable network for the system. This communication system will facilitates personalized medicine and clinical care pathway. This aim is achieved through the following objectives:

Specific objectives:

- To design a fail-safe communication system architecture suitable for POCT system deployment.
- To implement the system architecture independent of wireless technology evolution in communication.
- To develop, test and implement suitable mechanisms to maintain POCT data security in the communication system protocol.
- To investigate appropriate encryption schemes to identify a suitable method for data transfer while maintaining patient confidentiality.

1.3 Summary of Methodology used

A system approach was followed as a methodology for the research. As shown in Figure 1.2, there are three main data processing layers present in the system. The Data Collection and Analytics layer (Cloud Layer) is responsible for providing analytics mechanisms for a physician to develop diagnostics based on the collected data. The System Access Layer is responsible for authorizing access to system users to conduct the test as well as propagating the test data securely

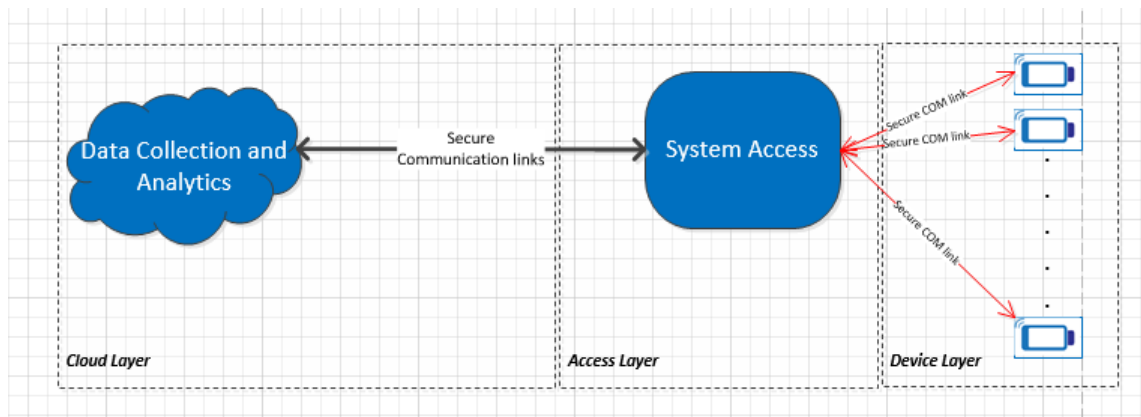


FIGURE 1.2: High-level System abstraction view

to the database. The Device Layer is the collection of the devices capable of carrying out the tests. All the system layers collaborate in providing error-free diagnostic results. A security violation in any one these layers will lead to a false diagnosis. The Access Layer represents a system component (or entity) that allows access to the device via communication links, mostly short-range connectivity such Wi-Fi, Bluetooth, or USB. The test data need to be sent to a data collection layer via cellular (or short range connectivity) communication links, where the data will be processed further to analyze the test results .

The research focuses on the communication links and mechanisms needed to transmit the test data securely from the device via the Access Layer to the Data Collection Layer. A customized V-model with agile product development methodology was used to develop the device and its internal and external communication paths. The standard V-Model (Figure 1.3 [2]) is a system engineering paradigm that represents an evolution of the waterfall model [31]. When this paradigm is internalized by a research organization, it provides a structured process for understanding research needs and linking them to requirements, design, and verification and to the final research

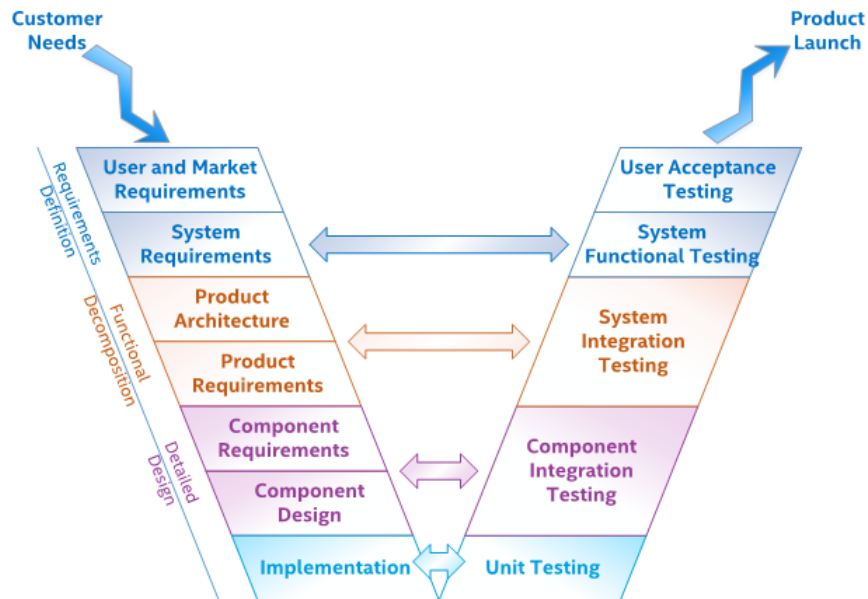


FIGURE 1.3: Standard V-Model Overview [2]

outcome and future work for the research. It facilitates the continuous integration of the system through the parallel development with other multi-disciplinary teams. This early collaboration with other researchers accelerates the research progress by asking critical questions around whether the right topics are considered before scope of the research is finalized. When these active feedback loops are established between the left and right sides of the V-model, flexibility and agility of the development teams result in a continuous release process where early patient feedback can shape the products delivered

Figure 1.3 shows a generic view of the standard system V-model [31]. There are different representations of this model used in the industry, which reflects how different business groups have tailored the model to their unique business needs. Starting at the top left, system engineers engage with user experience researchers and marketing and sales teams to elicit and clarify users and customers needs. These

needs are translated into system requirements by capturing and analyzing personas, usages, usage scenarios, and use cases that represent how users will interact with the system. Same concept is utilized in the development of the communication architecture for the deployment of POCT devices for infectious disease diagnostics.

During the requirements definition phase, Use Case Maps (UCM) methodology (Chapter-3) was used to perform an analysis of the data to negotiate and resolve conflicting requirements. The communication system architecture requirements establish contractual baseline between the research scope and the other functional teams for the architecture to be developed. This baseline is necessary because research needs will continue to evolve, so the requirements will also evolve. The baseline is used to provide a context for quantifying the schedule and cost impact of changes that will be imperative for completing a successful research project. There is a continuous negotiation throughout the analysis phase as requirements become better understood. If requirements are found to be infeasible, then further engagement with the stakeholders may be needed to resolve the conflict.

As shown in Figure 1.3, an important aspect of the V-model is early engagement with the quality and test efforts. In general, system functional test plans can heavily leverage the usage scenarios and use cases gathered during the requirements definition phase. The goal of functional testing is to verify that the functional, quality, and performance requirements have been met. During the system testing, missing functionality or performance requirements may be identified, which can be the basis of future research work. Integration test plans and test cases can be developed in parallel with the

communication system architecture requirements. The goal of integration testing is to verify that the components that are able to coexist and interact with each other as expected. Developing these plans simultaneously with the architecture provides the researcher crucial insight into how the architecture will be pieced together as components become available. Information derived from this activity influences the architecture and drives consideration of how the system will be tested. In the research, combinatorial design methodology was used to generate test cases for the system validation. The methodology is explained in detail in Chapter-2 .

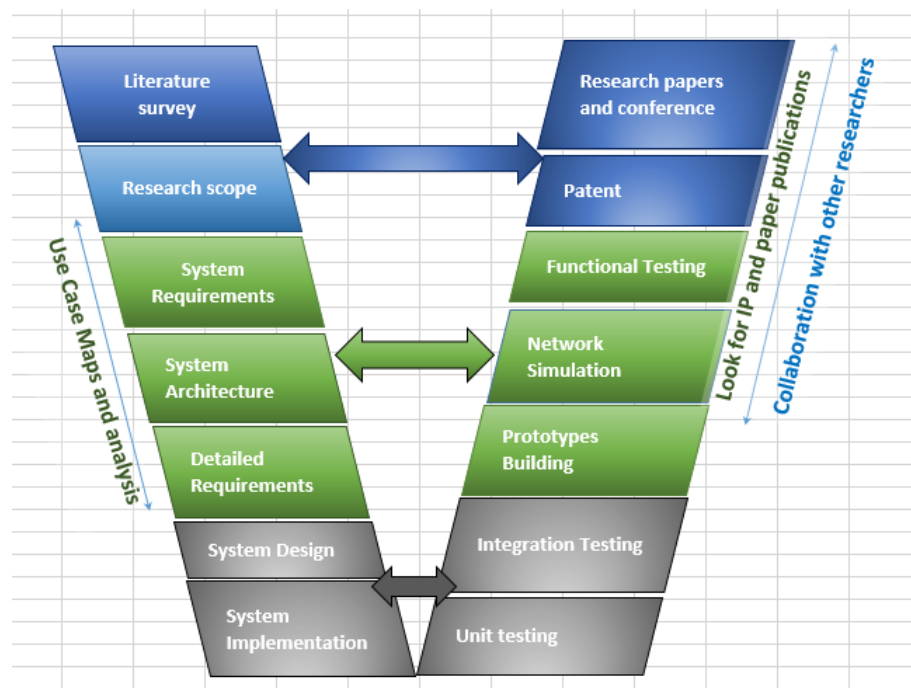


FIGURE 1.4: V-Model developed for research

Figure 1.4 shows customized V-model for the research work that has been undertaken. The customized model was derived from the standard V-model. The main differences between the models are that standard model is used for product development and the customized model is created for applying systems approach accomplishing research objectives. The new approach of introducing the UCM methodology for analysing requirements and combinatorial testing

are not part of the standard V-model. And the system requirements are derived from the research scope in the research V-model. Starting at the top left, the researcher conducts a literature survey, and he or she identifies the research scope. Then follows the creation of the systems requirements, the architecture development, and the creation of detailed requirements. The left side of the V-model is completed with the systems design and implementation. Starting from the lower right, unit testing and integration testing are done for the modules developed and configured. The unit level testing will identify issues with individual modules, such as hardware and software drivers. As illustrated in Figure 1.3, the integration testing helps to mitigate any interface related issues and it resolves coexistence issues. Then the process continues in collaboration with other research team members to create a prototype for functional testing. The simulation is done for point-to-point network configuration to decide the communication topologies needed.

The end-2-end system (device internal and external aspects of the communication) is concerned with functional safety and functional security, both of which overlay in the system V-model.

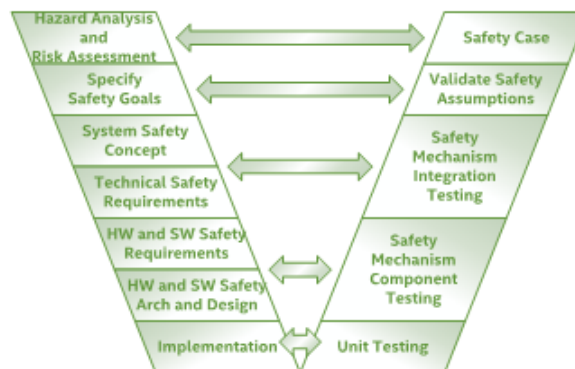


FIGURE 1.5: Functional Safety V-Model Overview

For functional safety, the IEC 62304 standard was adopted to develop the system communication architecture. Figure 1.5 shows the

functional safety V-model that corresponds to the general system V-model shown in Figure 1.3 and the research process V-model in Figure 1.4. More in-depth details of the IEC 62304 applied to the system design can be found in Chapter-2.

For the security analysis, the V-Model for the security system as shown in Figure 1.6 was used as a guiding model. Building a secure communications link for the system has different challenges, and the issues can be understood by creating thread models and threat mitigation strategies. The detailed description of the security analysis can be found in Chapter-4.

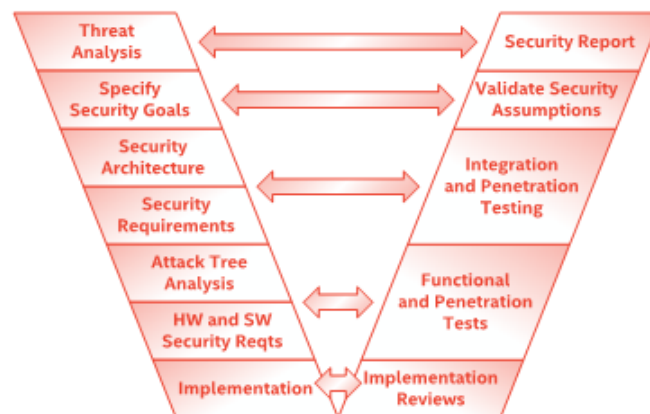


FIGURE 1.6: Security V-Model Overview

The next section shows an example of the end-2-end communication architecture developed in the research. Multiple configurations were generated, and the detailed explanation can be found in Chapter-6

1.3.1 Example of an end-2-end system configuration

The G-Node controls the device, and it acts as an intermediate layer for external communications. There are many possibilities to connect

the system entities based on the essential needs. One such configuration is shown in Figure 1.7. More discussion about the configurations can be found in Chapter-6.

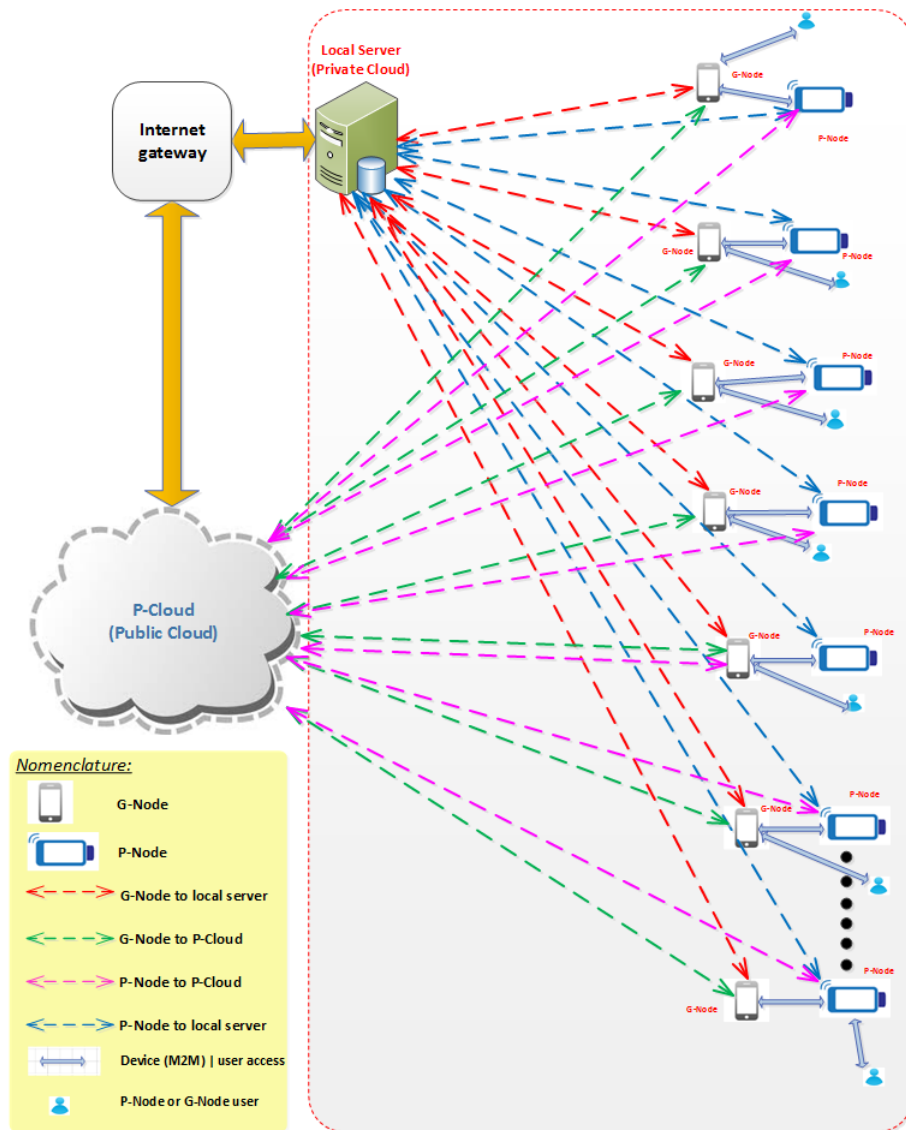


FIGURE 1.7: End-2-End Connection Topology

As shown in Figure 1.7, the system consists of multiple secure communication links between the network entities. P-Node represents the device. The communication system is responsible for supporting the topology connections shown in Figure 1.7. G-Node is a representation of the gateway functionality needed for communication of

multiple devices, while P-Cloud is the representation of cloud functionality required for storing test data and conducting diagnostics analytics (data storage and analytics cloud). There are five types of secure communication links that can be configured: G-Node to the local server, G-Node to P-Cloud, P-Node to P-Cloud, P-Node to the local server, a device to device communication (Machine to Machine (M2M) communication), and user to P-Node or user to G-Node. Some of the links shown in the figure are used for connecting network entities such as P-Cloud in the public domain, while the other links are used for establishing connectivity within a private sphere of connectivity (e.g. within a firewall of a hospital).

The communication system supports multiple cellular radio access technologies (RAT), 2G, 3G, and LTE. The device architecture was developed so that when the 5G standard is available commercially for deployment, it can be added to the device. In addition, location services module, Bluetooth, USB, Ethernet, and Wi-Fi connectivity paths are available in the communication sub-system.

There are other communication challenges from the communication system architecture itself. Modularized sub-systems are needed to provide decoupled architecture, to manage seamless data flow within the device. The internal data flow within the architecture will have an impact on the accuracy of data collection and further impact in transmitting the data to the P-Cloud. These challenges are addressed by having a protocol at the application layer of the design.

The P-Cloud has to support many salient features by means of a secure cloud structure within the cloud to ensure end-to-end connectivity of the communication. It contains of multiple sub-clouds

architectural components, namely, a measurement data cloud, a normalized data cloud for data mining and analysis, a configuration and provisioning data cloud, a system deployment data cloud, and a system operational data cloud. There are separate data access gateways to maintain data boundaries and data integrity. Distinct data access gateways and security groups are required in the P-Cloud to assure the security boundaries of users of the P-Cloud such as data provisioning agencies, health professionals, field technicians, and patients. Configurations such as those in the figure help to achieve successful deployments at the primary care level and are particularly amenable for use in remote settings with poor or no laboratory infrastructure [32].

1.4 Thesis structure

Chapter 1: Introduction and scope of research. This chapter introduces and explains the research idea, provides background information and explains why there is a need for the research. In addition, it lays out the organization of the thesis.

Chapter 2: Use Case Maps and Requirements for communication system development. Unambiguous requirements are key to successful product development. Various tools and methodologies exist to help with design, development, validation, and project management. This chapter discusses Use Case Maps (UCM) and EARS (Easy Approach to Requirements Syntax) methodology for creating communication system requirements.

Chapter 3: System Design process and Mission Critical System Design Process for the POCT system. This chapter

addresses the development cycle of the device. It begins with a description of the requirement gathering and moves on to outline the design process. From there, the discussion moves to the end-to-end system communication topology, the security principles of safety-critical communication software, and the development of a secure cloud architecture for the system. Then it addresses the process of system design compliant with the ICE 62304 standard, the validation methodologies for testing the communication links, and the Agile development methodology for project management.

Chapter 4: Security aspects of POCT system development.

Data security is an important feature of the system. This chapter explains the aspects that need to be protected, given the attack threat model for the device and the communication links. Use cases related to security are discussed. Two types of security frameworks have been developed, challenge and response based framework and system behavioral based framework. The discussion addresses security provisioning in the context of the M2M (e.g. P-Node to G-Node) communication model in relation to healthcare professional and the patients who use the system.

Chapter 5: P-Cloud and NAS Implementation. In this chapter, the internal components subsystem of the P-Cloud is explained. The internal architecture of the P-Cloud, which will provide error-free data storage for the device is shown in detail. In addition, a practical implementation of the private P-Cloud using NAS is described, along with the results.

Chapter 6: Network Models for POCT system deployment and simulation results. In this chapter, the various use cases of connecting the system components are explained. Simulation of

P2P connectivity representing the communication between P-Node and G-Node is shown with obtained results. For managing transmission control, a novel methodology was developed and implemented, collaborative congestion control, is introduced and explained.

Chapter 7: Conclusion and future work. In this chapter, achievements and contributions to knowledge are listed, and recommendations for further work are added to continue the research.

The traceability between the chapter contents and the specific goals are mapped in Table 1.1. The traceability shows the links between the chapters and the goals of the research in a matrix format for clear understanding. The row in the table labelled Chapter indicates the list of chapters in the thesis. The columns in the table labelled Research Scope show the research objectives. The traceability matrix shows the contents of the chapters relevant to the research objectives.

1.5 Contribution to knowledge

The research involved developing communication configurations for an end-2-end system architecture that supports device connectivity. The research also involved creating a design methodology for a robust, dependable, safe, and secure communication platform for interconnecting the devices. Unanticipated network infrastructure changes and latent coding errors lead to operation faults despite that usually a significant effort has been expended in the design, verification, and validation of the software system [32]. It is becoming increasingly more apparent that one needs to adopt different approaches that will guarantee that a complex system meets all safety,

Chapter Research Scope	Chapter -1: Introduction and scope of research	Chapter-2: Use Case Maps and Requirements for POCT system development	Chapter-3: POCT System Design process and Mission Critical System Design Process for POCT	Chapter 4: Security aspects of POCT system development	Chapter 5: pCloud (Cloud Implementation for POCT System) and realization using NAS.	Chapter 6: Network Models for POCT system deployment and simulation results	Chapter 7: Conclusion and future work
To design a fail-safe communication system architecture suitable for POCT system deployment.		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
To implement the system architecture independent of wireless technology evolution in communication.		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>	
To develop, test and implement suitable mechanisms to maintain POCT data security in the communication system protocol.			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
To investigate appropriate encryption schemes to identify a suitable method for data transfer while maintaining patient confidentiality.			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>

TABLE 1.1: Mapping between chapters and specific research goals

security, and reliability requirements, in addition to complying with standards. There are many initiatives taken to develop safety and security critical systems, at different development phases and in different contexts, ranging from the system infrastructure design to the device design. Various approaches are implemented to design error free software for the safety-critical system. The approach and methodologies adopted in the research can overcome the challenges

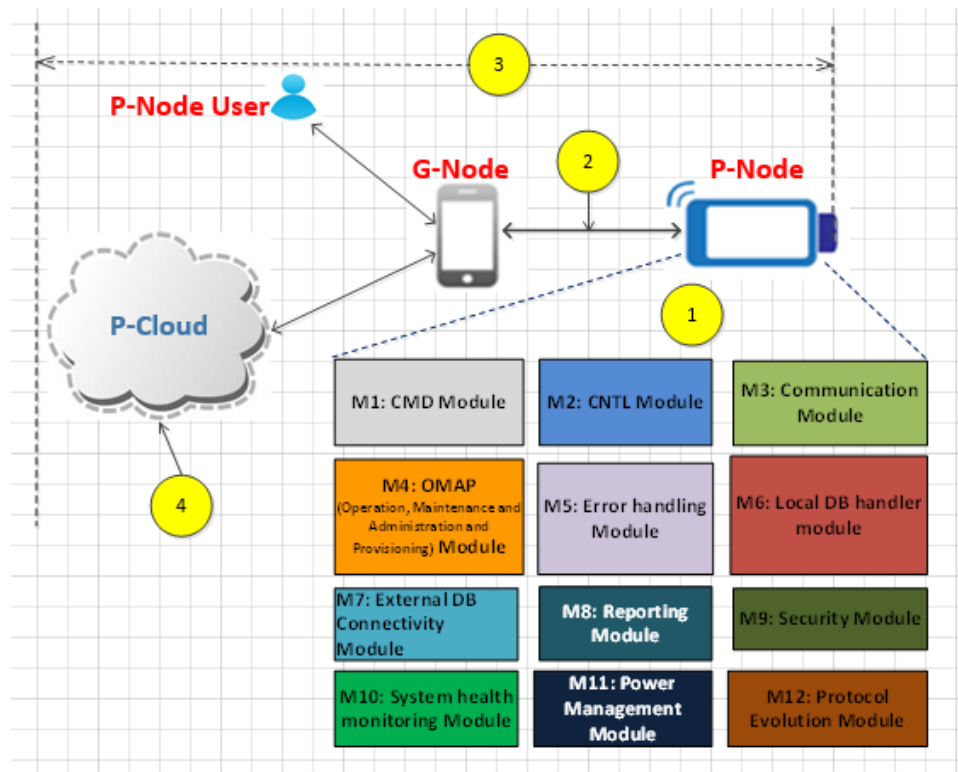


FIGURE 1.8: Technical contributions (indicated as 1: P-Node architecture, 2: G-Node to P-Node communication, 3: End-to-End communication and 4: P-Cloud)

in developing error control and communication software for communication and system architecture (see Figure 1.8).

This present research makes a unique contribution to the health-care system based on its incorporation of modern and evolving technologies such as mobile computing and secure and reliable cloud computing. Figure 1.8 shows the internal system modules needed for controlling the device from the G-Node (or smartphone). These subsystem modules can be used in the devices standalone mode of operation, when the G-Node is not present. As indicated in Figure 1-8, there are four key fields of contribution in this research, as listed below.

Contribution – 1

System requirement modelling and device architecture, Figure 1.8 ([1])

The research developed the required management approach for the communication links in order to deploy system securely, including a representation of requirements using Use Case Maps and EARS (Easy Approach to Requirement Syntax) methodology. Use Case Maps are used to express device communication requirements in a model form which is described in Chapter-3. These requirements provided the guidance for developing the communication system in detail. Security-aware architecture for data communication between the building blocks for the device system was developed. Internal communication between the sub-systems is vital for ensuring error-free external communications. The research developed a layered architecture with multiple functional modules. The model takes into consideration all the stakeholders (patients, healthcare personal, clinicians, POCT device manufacturers and network operators) requirements [33].

Contribution – 2

A secure communication protocol independent of radio technology, Figure 1.8 ([2])

A secure communication protocol with expandability that operates independent of radio access technology was developed. The radio technology independent protocol enables data transfer between P-Node, G-Node, and P-Cloud. The device communication system was implemented based on the IEC62304 standard, and it specifies the life cycle requirements for the medical device software and system based on the critical safety requirements. The communication subsystems were portioned to meet the IEC62304 risks classification classes. Communication between multiple components via connectivity (Bluetooth) and cellular protocols (2G and 3G) was demonstrated [32].

Contribution – 3

Contribution 3: An End-to-End Secure Communication, Figure 1.8 ([3])

Security mechanisms were created and implemented for the end-to-end communication system consisting of the P-Cloud, the G-Node, and the POCT device. Two kinds of security concepts (challenge-response based and behaviour based) have been developed to counteract security threats faced by the device and the communication system. A smart encryption model for transmitting secure data was developed. An encryption process of the system data storage and transmission was designed with encryption key management strategy for private cloud based on NAS [34].

Contribution – 4

Coordinating multiple devices and systems and Secure cloud architecture for POCT data storage, Figure 1.9 ([4])

The POCT system needs to support multiple devices in a practical network deployment. A configurable deployment model based on the capacity and capabilities of the network and the POCT device was designed and implemented. The system is made up of the hierarchical model to support large-scale systems with expandability based on the growth. It is divided into three logical systems: zone, site, and unit. A zone consists of multiple system sites. A system site contains multiple instances of POCT device units. Due to data transmission in three levels, there will be scenarios where the packet drops may occur, and they must be eliminated due to mission criticality of the data. Hence a congestion control algorithm for managing the devices communication has been developed.

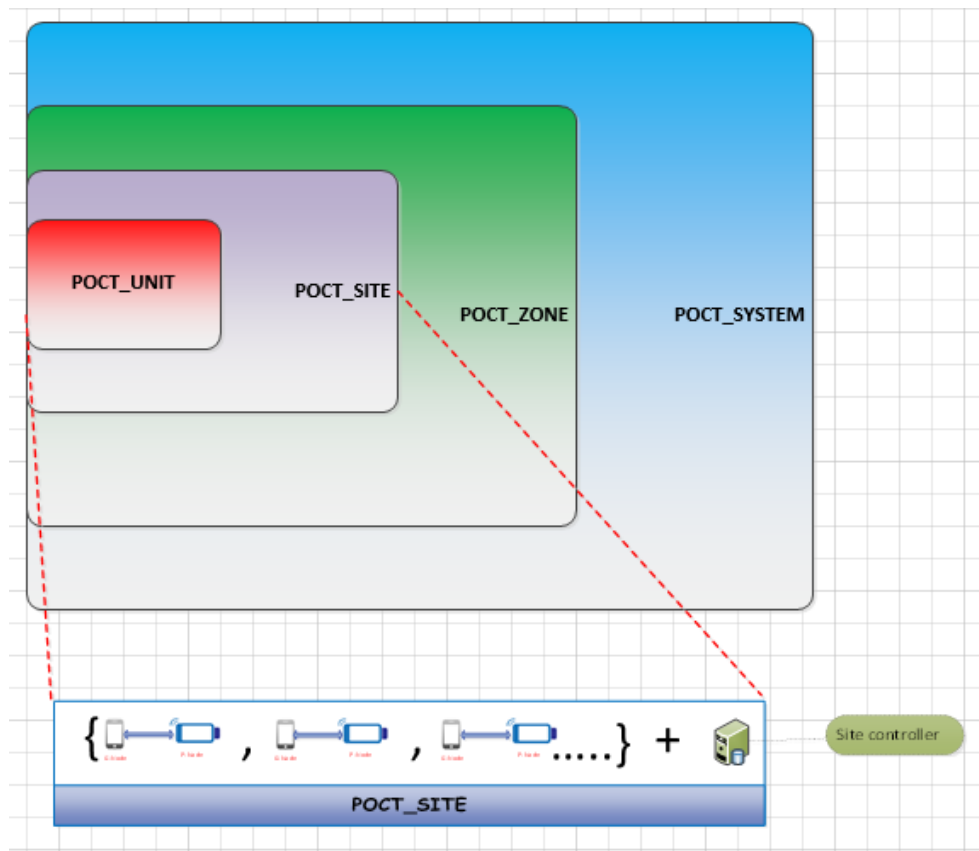


FIGURE 1.9: Hierarchical model for POCT system management

1.6 List of Publications

- Branavan. M., Mackay. R., E., Craw. P., Naveenathayalan. A., Ahern. J., Sivanesan. T., Hudson. C., Stead. T., Kremer. J., Garg. N., Balachandran. W., Modular development of a prototype point of care molecular diagnostic platform for sexually transmitted infections, *Med. Eng. Phys.*, vol. 38, no. 8, pp. 741748, Aug. 2016.
- Love. F., McMillin. B., Tulasidas. S., and Balachandran. W., Multiple security domain non-deducibility for point-of-care diagnostic technology: Work In Progress (WiP) abstract, in *Proceedings of the 7th International Conference on Cyber-Physical Systems*, 2016, p. 42.

-
- Tulasidas. S., Mackay. R., Craw. P., Hudson. C., Gkatzidou. V., and Balachandran. W., Process of Designing Robust, Dependable, Safe and Secure Software for Medical Devices: Point of Care Testing Device as a Case Study, *Journal of Software Engineering and Applications*, vol. 6, no. 9, pp. 113, Aug. 2013.
 - Tulasidas.S.,Hausner.J.,Terzakis.J.,Love.F.,Mattern.S.,Hudson.C., Manivannan.N., and Mackay. R., Requirements for Point of Care Devices using Use Case Maps, *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 3, no. 6, pp. 42844288, 2015.
 - Tulasidas. S., Mackay. R., Hudson. C., and Balachandran. W., Security Framework for Managing Data Security within Point of Care Tests Security Framework for Managing Data Security within Point of Care, *Journal of Software Engineering and Applications*, vol. 10, no. 10, pp. 174193, Feb. 2017.

Chapter 2

Requirements for POCT system communication and Architecture

2.1 Introduction

The design and development of the POCT system and the associated infrastructure must be done with extreme rigour, as it is a mission critical or safety critical system. The requirements for the creation and management of the system are the key processes for ensuring that a highly reliable system with a low defect is developed, since an accurate diagnosis is an essential process improvement for remote patient care management. The requirements state what the system must do and how well, in terms of performance. The requirements need to be specified accurately, completely, and without any ambiguity so that the POCT system can be designed and developed with minimum implementation defects. This process provides physicians a vehicle to diagnose patients with increased reliability. Requirement modelling can help to express the behaviour of the system in a visual format. Use of Case Maps (UCM) is one such modelling technique that sufficiently expresses the model of the required specifications for the POCT system. The UCM connects the end-2-end

system architecture and its behaviour in a visual format. The responsibilities of the architectural building blocks (components) and their interactions can be described effectively. With UCMs, user scenarios are drawn as exchanges of messages between the components. The UCM has notations that were developed at Carlton University, Canada [35, 36]. It needs informal data about the components of the system with their roles in the context of the system. It can express abstract level descriptions and multilevel decomposition of the functions. and is used to illustrate the functional requirements and security requirements of the POCT system. Some of the key requirements are represented with UCM notation The requirements were created using the EARS (Easy Approach to Requirement Syntax) syntax, which contains known patterns for particular types of functional requirements and was developed by Rolls-Royce system engineers [5]. The requirements are captured with a single imperative statement containing shall. The well-written requirements will provide a higher probability of implementing the system with low defects and better quality.

All the requirements will have an unique identification number or format. The identification format of the requirement followed in this chapter is as follows:

[FR] [sub-system-name] [a-three-digit-numerical-value]

where FR means a functional requirement, sub-system-name is the name of the associated functional module, and the three-digit is the unique identification of the functional requirements. Only the high-level requirements are discussed. Table 2.1 shows the patterns used in constructing the requirements.

Pattern Name	Pattern
Ubiquitous	The < System name> shall < system response>
Event-Driven	WHEN <trigger> <optional precondition> the <system name > shall <system response>
Unwanted Behavior	IF <unwanted condition or event>, THEN the <system name > shall < system response>
State-Driven	WHILE <system state>, the <system name> shall <system response>
Optional Feature	WHERE <feature is included>, the <system name> shall < system response>
Complex	Combination of all the above

TABLE 2.1: EARS pattern [5]

The requirements described here have three main attributes: *the name of the requirement*, *the description of the requirement*, and, when necessary, *the rationale* for the requirement. This structure is followed throughout the chapter.

2.2 POCT System Description

The POCT system is being developed at the Brunel DOC LAB [1] as described in Chapter-3. It consists of modular subsystems as described in reference [32]. The primary building blocks of the system are considered:

- **P-Node** the POCT device in the patients home
- **G-Node:** the gateway device in the patients home
- **P-Cloud:** the private cloud over which patient data that can be accessed by physicians is transmitted

These building blocks are shown in Figure 2.1.

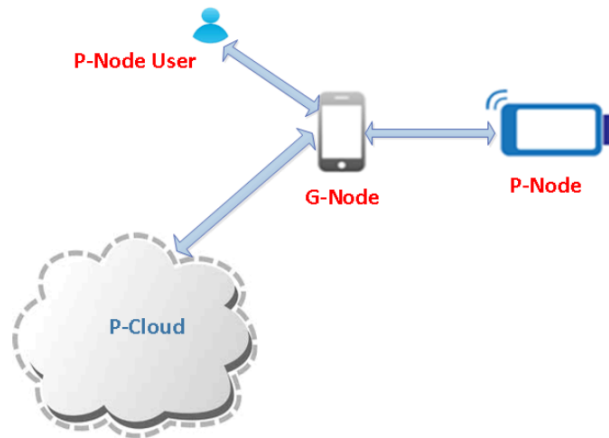


FIGURE 2.1: Basic building blocks

2.3 POCT System Requirements

The P-Node is the representation of the POCT device. The G-Node represents the gateway entity, which can be a smartphone or a laptop with internet access. A dedicated hardware device may play the role of the G-Node. Patient interacts with the P-Node for diagnostic testing via the G-Node. These devices send measurement data to the database server, as well as control the P-Node. The P-Cloud represents the secure private data cloud.

2.3.1 Requirements for P-Node operation

FR-PNODE-001: The P-Node shall have two modes of operation: user control mode and reporting mode.

The P-Node is initiated by the user interaction, which is called the user control mode. The measured data needs to be transmitted to the cloud database, which is called the reporting mode.

FR-PNODE-002: While in the user control mode, the P-Node shall receive commands from the G-Node for initiating the testing.

FR-PNODE-003:When completed test results are ready for transmitting to the G-Node, the P-Node shall switch to reporting mode.

FR-PNODE-004: While in reporting mode, the P-Node shall report the test data to the P-Node user (via the G-Node) and to the P-Cloud.

The requirements FR-PNODE-001,002, and 003 describing the basic behaviour of the P-Node. There are no other modes of operations are designed, other than user-control-mode and reporting-mode.

2.3.2 UCM representation of P-Node operation requirements

Figure 2.2 shows the UCM representations of the requirements FR-PNODE-001, FR-PNODE-002, FR-PNODE-003 and FR-PNODE-004.

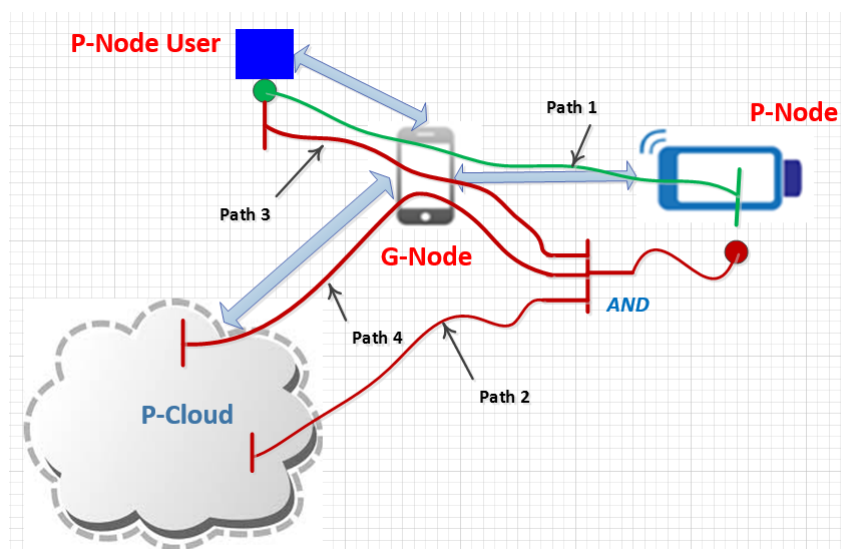


FIGURE 2.2: UCM representation of P-Node communication links

Path 1 conveys a message to the downstream implementation team that the P-Node user shall be able to control the P-Node via the G-Node. Path 1 shows the execution path for this use case (that the user can control the P-Node), which is the user control mode of the

device. Path 2 represents the reporting mode of the P-Node to the P-Cloud. Path 2 conveys the message to the development team that the data needs to be sent to the P-Cloud by some communication means. Note that Path 2 does not go through the G-Node, implying that the P-Node needs to have a communication subsystem. Path 3 conveys the message that during the reporting mode, the P-Node also needs to send data to the user via the G-Node. The AND UCM artefact [37] in Figure 2.2 (the vertical line with three horizontal lines on one side and one horizontal line on the other side) serves as a way of enforcing the AND condition that the report needs to be sent to both the user and P-Cloud. Path 4 conveys the intent of the requirement to the developer community or the testing community that the P-Node must send the data also via the G-Node to the P-Cloud for redundancy. The Paths 1,2,3 and 4 show the execution pattern that needs to happen with the system as a whole.

2.3.3 Requirements for P-Node construction and system interconnection

Each requirements will have a unique ID and a brief description following the EARS methodology.

FR-PNODE-005: The POCT system shall be designed as a collection of functional modules.

A modularized system encourages the design of mission critical system such as the POCT as a loosely coupled system [38].

FR-PNODE-006: The POCT subsystems shall interconnect using industry standard interface technology. The interconnect technology may include the I2C [39] interface but is not limited to it. A

standard technology for interconnecting hardware modules helps to use multivendor modules in the design.

FR-PNODE-07: If there is a fault detected in an isolated systems module, and then the POCT system shall contain that fault within the individual module.

FR-PNODE-008: The POCT modularized system shall meet the mandatory system design and development process as outlined in the international standard for medical device software and software cycle processes standard IEC 62304 [38].

FR-PNODE-009: The POCT system shall interconnect with a third-party system. Note: this interconnection can be via an industry standard interface such as USB or Wi-Fi.

2.3.4 Requirements for P-Node fail safe mechanism

FR-PNODE-010: The POCT system shall be implemented using multiple, independent hardware modules.

Note: The rationale for this requirement is to have a system that does not have a single point of failure. A distributed module architecture within the device was developed with minimal inter-dependency. Any failure in one module will not impact the functionality of the other modules. The communication radio access technologies are implemented on single radio modules.

2.3.5 Requirements for POCT system data storage

FR-PNODE-011: When the device completes a diagnostic test and the communication system is disabled, the device shall store the

test data in local non-volatile memory.

Note: Local non-volatile memory can include an SD card or a Flash drive.

FR-PNODE-012: If a communication link failure occurs, then the device shall identify the source of the failure.

FR-PNODE-013: The POCT system shall include an agent to monitor communication link failures.

FR-PNODE-014: The P-Node shall encrypt data transmitted to the P-Cloud using SSL revision 3.0.

FR-PNODE-015: All the data stored in the local storage shall be moved to the cloud when the communication links are available.

It is not advisable to leave the test data in the device indefinitely for security and privacy reasons. All the requirements stated (FR-PNODE-012 to FR-PNODE-015) must be implemented together.

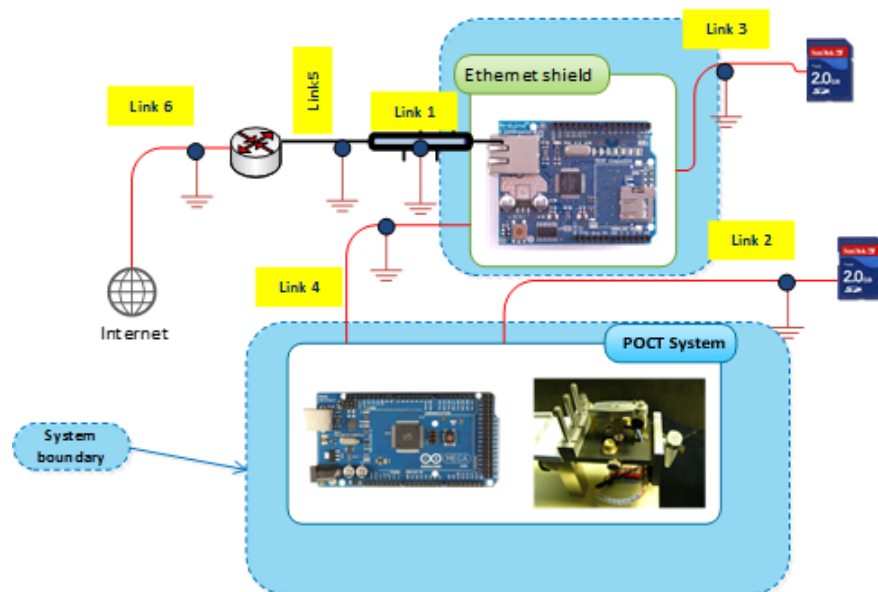


FIGURE 2.3: Identification of failure points

The links (Link 1,2,3,4,5 and 6) that are shown in Figure 2.3 are related to the local data storage when the SD card and Ethernet connection are used in the POCT system. Typical failure points are indicated by ground signals as per the UCM notation [40] and marked in the communication links. It is in addition to the text-based requirements (FR-PNODE-011 and FR-PNODE-012). The possible failure points are at the communication link between the core POCT system and the local SD card link (*link 2 failure*), the local SD card and the Ethernet card (*link 3 failure*), the Ethernet output link and the router input link (*link 1 and link 5 junction failure*) and the router output link to the Internet (*link 6 failure*). Both the textual requirements and the UCM representation diagram will provide an unambiguous way of conveying the intent to the development and testing organizations. This modelling paves the way for applying the failure mode analysis methodologies applicable to medical devices and the associated communication system [41, 41, 42].

2.3.6 G-Node Requirements

As explained previously, the G-Node represents the gateway entity that has the responsibility for controlling the device and collecting the test data from the device. The G-Node can be a smartphone or a computing device such as a PC. The following requirements (*FR-GNODE-XXX*) are some of the key features needed to build the G-Node.

FR-GNODE-001: The G-Node shall be subscribed for the test completion events for receiving test data from the POCT device.

FR-GNODE-002: The G-Node shall encrypt data transmitted to the P-Node using SSL (Secure Socket Layer) revision 3.0.

(Note: a corresponding P-Node requirement has been added under P-Node requirements: FR-PNODE-014)

FR-GNODE-003: The device shall utilize each of the following connectivity solutions for communication between the G-Node and the P-Node:

- Cellular radio (either 2G, 3G, 4G or 5G)
- Wi-Fi
- USB

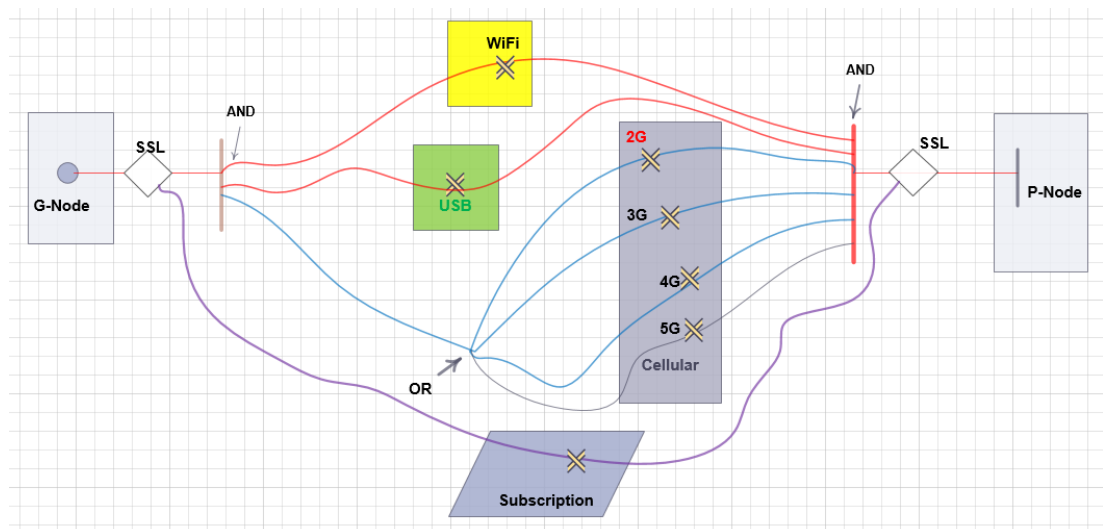


FIGURE 2.4: UCM for representing G-Node requirements

Figure 2.4 represents the G-Node requirements (FR-GNODE-001, FR-GNODE-002) and (FR-GNODE-003) using UCM notation. The SSL requirement is shown as a static stub (UCM terminology for special functionality) at both ends. The static stub informs the development team that SSL is needed in the G-Node as well as in the P-Node. The use of the AND junction conveys the intent that Wi-Fi, USB, and the cellular connectivity are needed as described in the requirement simultaneously. The OR junction indicates that any one (or all) of the technologies among the 2G /3G /4G /5G is active at

the time. The subscription process block shows that the G-Node is registered with the P-Node for an important event, such as completion of the assay test. When the test is completed, an indication will be sent to the G-Node with the measurement data.

The UCM map in Figure 2.4 shows (AND notation) another important consideration, in that the system design must look at the coexistence challenges [43] between the 4G and Wi-Fi. The wireless technologies continually evolve using newer frequency spectrum. Because of the use of new frequency spectrum, cross interference of two or many radio access technologies is unavoidable. But in the context of the POCT system, countermeasures for any interference due to the coexistence of the radio access technologies must be considered.

2.3.7 P-Cloud Requirements

The data collected from the POCT device is stored in a non-volatile memory device, such as an SD card, first. This data is transmitted to Cloud storage (P-Cloud) via the G-Node as shown in Figure 2.5. The following section lists the requirements pertaining to the P-Cloud. The P-Cloud provides a way of sharing the test data among various health-care organizations. The purpose of the P-Cloud can be divided into four domains (test data, device provisioning data, operational data and analytic data for data mining). The P-cloud is the storage for the raw data and the transformed data from the P-Node. The transformed data is needed for data mining analytics as an aid to diagnosis. The P-Node to P-Cloud connectivity link is implemented using M2M communication. Physicians use the transformed data for diagnosis. Therefore, the P-Node needs two separate access gateways. Provisioning data from POCT devices and deployment

data of the system installations need to use the P-Cloud. The access path for this usage can be managed by a separate access gateway. The POCT devices operational data also needs to use the P-Cloud, and it requires another access gateway to the P-Cloud. Based on the above concepts, the following requirements are formulated.

FR-PCLOUD-001: The P-Node shall provide secure access based on SSL 3.0 encryption for storing the POCT device provisioning data and the POCT system deployment data. Note: The connectivity between the P-Node and the P-Cloud shall be classified as M2M communication.

FR-PCLOUD-002: The P-Cloud shall provide secure access based on SSL 3.0 encryption for storing the POCT devices operational data.

FR-PCLOUD-003: The P-Node shall provide secure access based on SSL 3.0 encryption for a physician for diagnosing patient data.

FR-PCLOUD-004: The P-Cloud shall encrypt data transmitted to the P-Node using SSL revision 3.0.

Figure 2.5 shows the UCM representation of the POCT system indicating all the P-Cloud requirements stated. A similar approach is shown in reference [44]. The responsibilities (or the functionalities) required within each domain are displayed as X, representing the capabilities of each of the systems entities. All the capabilities are not shown. The G-Node has two Xs, indicating the two main functions, to collect data and to connect to the P-Node. The P-Cloud has two main functionalities, store data, and store diagnosis. The SSL connectivity shows all the players in the data collection to data storage process. Starting from the Patient, he or she initiates the

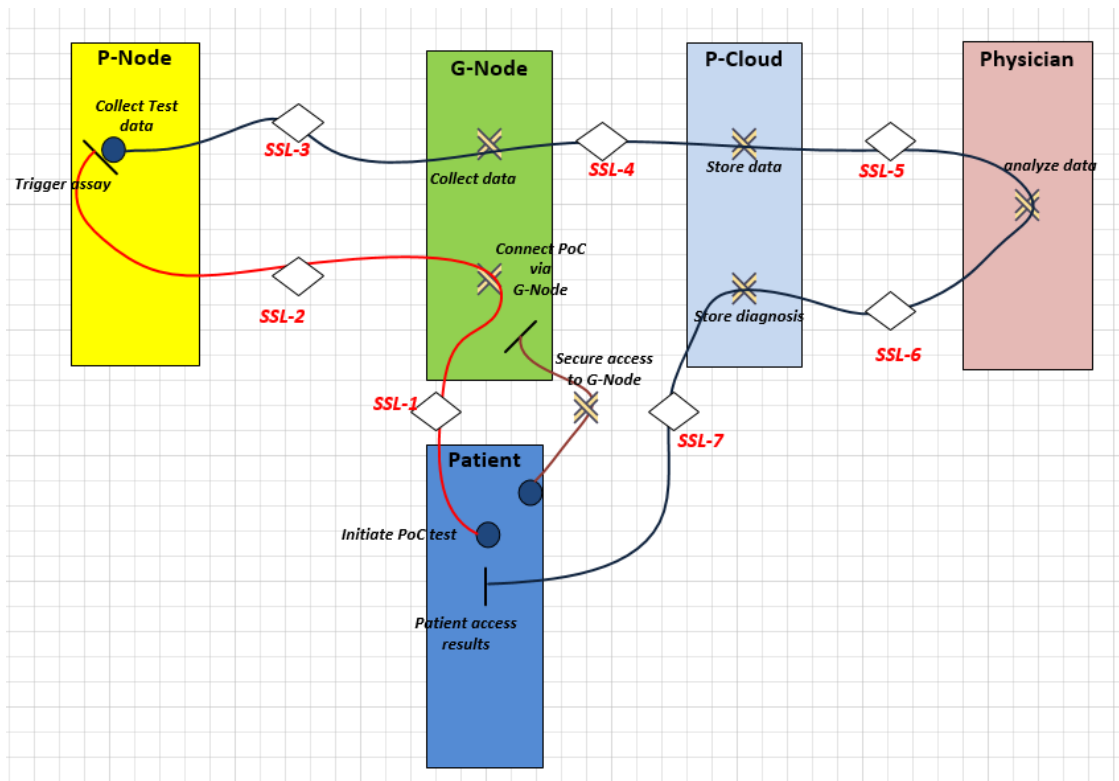


FIGURE 2.5: P-Cloud requirements

test via the G-Node (SSL-1). The G-Node responds to the patient and connects with the P-Node (SSL-2), triggering the corresponding assay on the device. When the data is ready, it is collected back to the G-Node (SSL-3). The collected data then is transmitted back to the P-Cloud (SSL-4) and it is securely stored. The transmitted data is available for a physician for analysis and data analytics (SSL-5, SSL-6). The diagnosis results is available for the patient (SSL-7). The SSL-7 links a web URL to view the results.

The communication path from the patient and ends at the P-Node (also UCM path ending). The other communication path starts from the P-Node, goes through the G-Node, P-Cloud, and physician, and ends with the patient. There is another communication path that starts from the patient and ends at the G-Node, which represents the secure communication access. Note that creating the UCM paths to describe the dynamic behaviour of the system is entirely up to

the system architect or designer. The success of the system design depends on the way in which the architecture is created.

2.3.8 POCT System Security Requirements

FR-POCT-DATA-001: Data stored on SD card shall be protected for authorized read and write activity. The following section

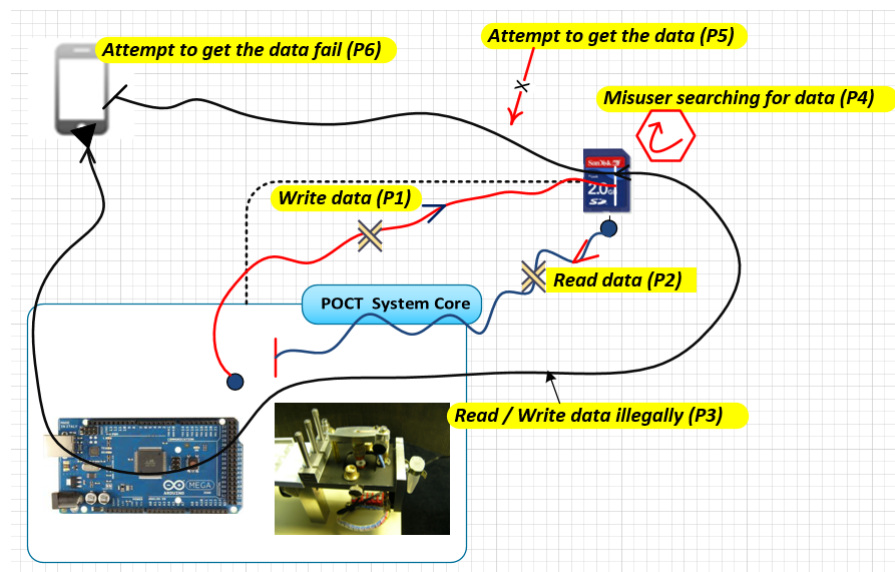


FIGURE 2.6: MUCM Example

discusses security requirements, using the technique called MUCM (missed use case maps) [45]. Security with respect to the data is elaborated.

Figure 2.6 shows the UCM and the MUCM paths for accessing data from and to the SD card. Since the initial data storage is on the SD card, it must be protected. The P1 path is a normal operation for writing data to the SD card during the test. Path P2 shows the reading of data from the SD card. These are the UCM representations for R/W data from/to with respect to the SD card.

The path P3 shows an intruder attempting to modify or read data on the SD card illegally, by connecting to the POCT System from an

unauthorized smartphone. The path P4 indicates that the misuser (unauthorized user) searches the SD card for the data. P5 indicates the misusers attempts to read the data. Attempting to access the data fails because of the closed-loop nature of the system and the security measures that are in place (Security measures are addressed in Chapter 4).

2.4 P-Node Architecture Overview

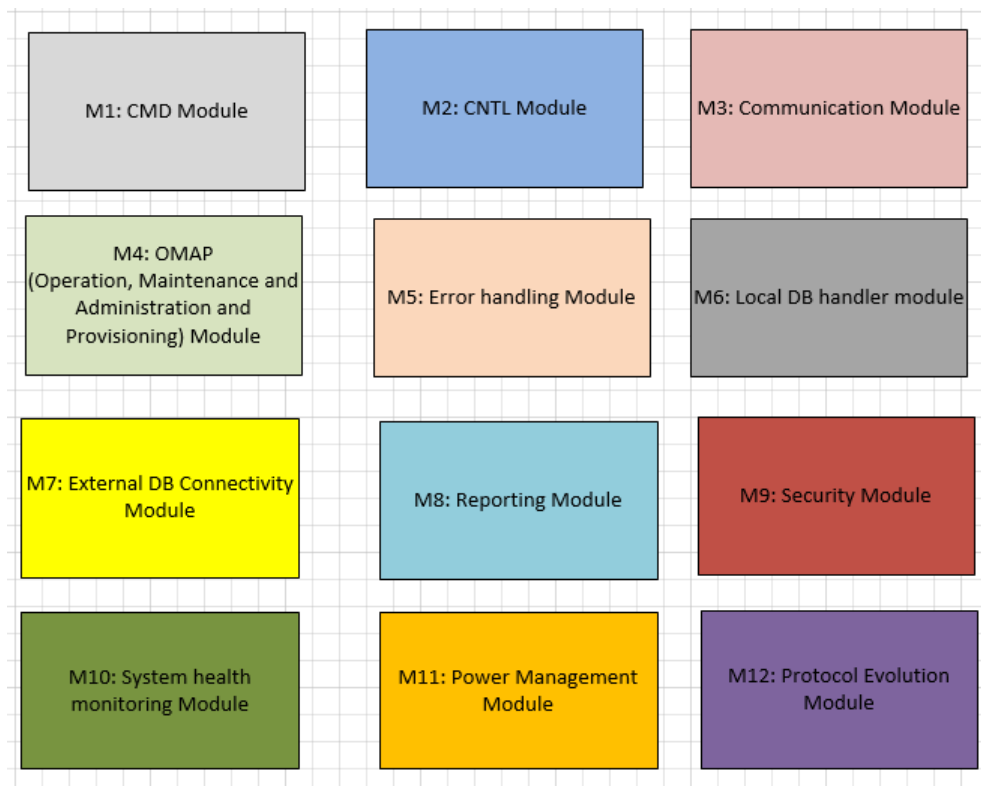


FIGURE 2.7: Modularize framework

The architecture of the P-Node is organized into 12 modules as shown in Figure 2.7. The P-Node system architecture will have the following SW and HW modules, forming a generic framework independent of interconnecting (hand-held or smartphones) devices. In this section, the framework is named the P-Node system framework. The interconnecting devices will use a communication protocol to send

data and retrieve information from the P-Node system framework. The communication protocol details are shown in the diagram Figure 2.8. All the modules that are described here will provide services that will be consumed by other modules to accomplish mission critical safety requirements for implementing the P-Node terminals. In other words, the proposed frame is a service-oriented system framework architecture for implementing P-Node systems, based on simple, functional calls or messaging. Because of the modularized architecture, the POCT system can be built using in-house developed modules and external libraries as a hybrid system.

2.4.1 M1: CMD module

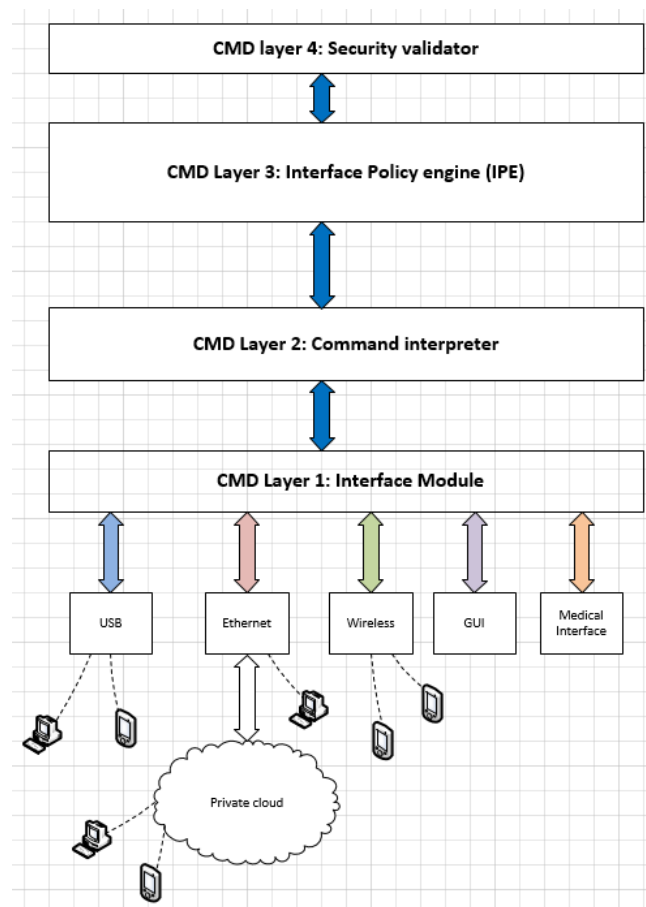


FIGURE 2.8: CMD Module in detail

The M1 module is responsible for interpreting commands that are sent from the five types of interfaces that are shown in Figure 2.8. The CMD module will incorporate security mechanisms that can be made to comply with the medical security directive specified by privacy and data protection agencies. The command modules role is to interpret the command data received via the user interface or other communication interfaces (wireless and USB based). The proposed command module has five layers. Each layer has a distinct responsibility in validating a command received by the five types of interfaces. These interfaces are: USB, Ethernet LAN, Wireless (cellular and connectivity), GUI and Medical Interface [46].

2.4.2 Interface types:

There are five types of interfaces considered. USB is the well-known connectivity mechanism for any IoT device. The Ethernet LAN connectivity is shown as connectivity for accessing data from a private cloud. The diagram in Figure 2.8 shows two ways of interfacing with the P-Node, via the private cloud and direct connection to a PC. It is also possible to have the same connectivity via the wireless interface as depicted in the Figure.

2.4.2.1 CMD layer 1: Interface Module:

The interface module provides driver support for all the predetermined interface paths for the five primary input interfaces shown in Figure 2.8. In practice, this is provided by stacked HW modules [47] such as the stackable Arduino HW, or integrated multi-input systems such as Intel Galileo [48]. An interface module based on

the Arduino system has been demonstrated, and the configuration details have been published [32].

2.4.2.2 CMD layer 2-3: CMD interpreter and Interface Policy

When the command is decoded and parsed, the contents are passed on to the interface policy layer. The interface policy layer looks at the interface type attribute and decides if this command is allowed or denied based on the policy rules configured.

2.4.2.3 CMD layer 4: Security validator

The role of security validator is to make sure that data integrity is preserved in accordance with the data encryption scheme.

2.4.3 M2: CNTL module:

The M2 module is responsible for providing control functionality for thermal control, magnetic control, and other implementations (realizations) of required control algorithms.

2.4.4 M3: Communication module:

The M3 module is responsible for all the communication (native and non-native) related to functionality. The native communication stacks such as TCP / IP, USB, and Ethernet are provided by this module. This module will provide features for GUI and medical interface connectivity. This module is responsible for location services as well. Self-contained MODEM solutions provided by Intel [49] or Qualcomm [50] are the examples of the M3.

2.4.5 M4: OMAP module:

The OMAP (Operations, Maintenance, and Provisioning) module is responsible for device operation, handling SW / HW maintenance, administrative functionalities (e.g. billing) such as interacting with health care providers (organizations), and provisioning (i.e. the configuration of device services).

2.4.6 M5: Error handling module:

The M5 module manages all type of errors and exceptions during operation and configuration of the device. The M5 module handles the communication errors in M3. The generation of error codes and responses to errors are covered by this module.

2.4.7 M6: Local DB handler:

The local DB handler module is responsible for providing local data storage services. One of the usages of this module is to handle the data storage securely during loss of communication link between the G-Node device and the P-Node device. This module uses the services provided by the security module for securing the data storage on the system SD card.

2.4.8 M7: External DB connectivity module:

Module M7 is responsible for establishing connectivity between external DB systems that are located in a private cloud, secure server farms, or data warehouses. The M7 module uses the services provided by the communication module and other modules. In addition,

it creates secure communication links to the external DB repository as needed.

2.4.9 M8: Reporting module:

The module M8 provides services that are required to produce reports and charts via the GUI interface on the device itself. This module may participate in concert with handheld or smartphone devices in the creation of required reports via the communication protocol.

2.4.10 M9: Security module:

The M9 module provides mechanisms (as functions for other modules to use) for protecting user data and user accounts associated with the devices. It provides access services for privileged access users. This module is responsible for managing security policies, as well. The security policies supports the required access to the device interface.

2.4.11 M10: Device health monitoring module:

The M10 module acts as the watchdog of functionality for the whole device. If the M10 determines that device behaviour is unexpected (i.e. not known behaviour) or a faulty situation, then the M10 attempts to take predetermined actions.

2.4.12 M11: Power Management module:

The M11 module provides power managing control for battery power saving. The module used in the POCT communication systems has its own power management capabilities.

2.4.13 M12: Product Evolution module:

The role of the M12 module is to provide compatibility information and a development path for expansion of the product. The module is usually accessible by the development community (such as Intel-based products [51]) who will help to develop applications and other related new developments for the P-Node. The new trends and user needs are captured and implemented as content in this module.

2.5 Protocol for POCT system communication Open Communication Protocol

An application specific communication protocol (Figure 2.9) is developed to run at the higher layer of the software stack. It is named an Open Communication Protocol because it is an open framework system, and it can be used without encryption, thereby enabling the communication design to exist independent of the communication and radio access technologies used. This is deliberate, to manage the continuing changes in the wireless access technology. In practice, the device can work with any communication system as a plug and play process where the core of the POCT communication does not impact due to the change in lower layer communication technologies. The protocol structure is a framework for implementing its

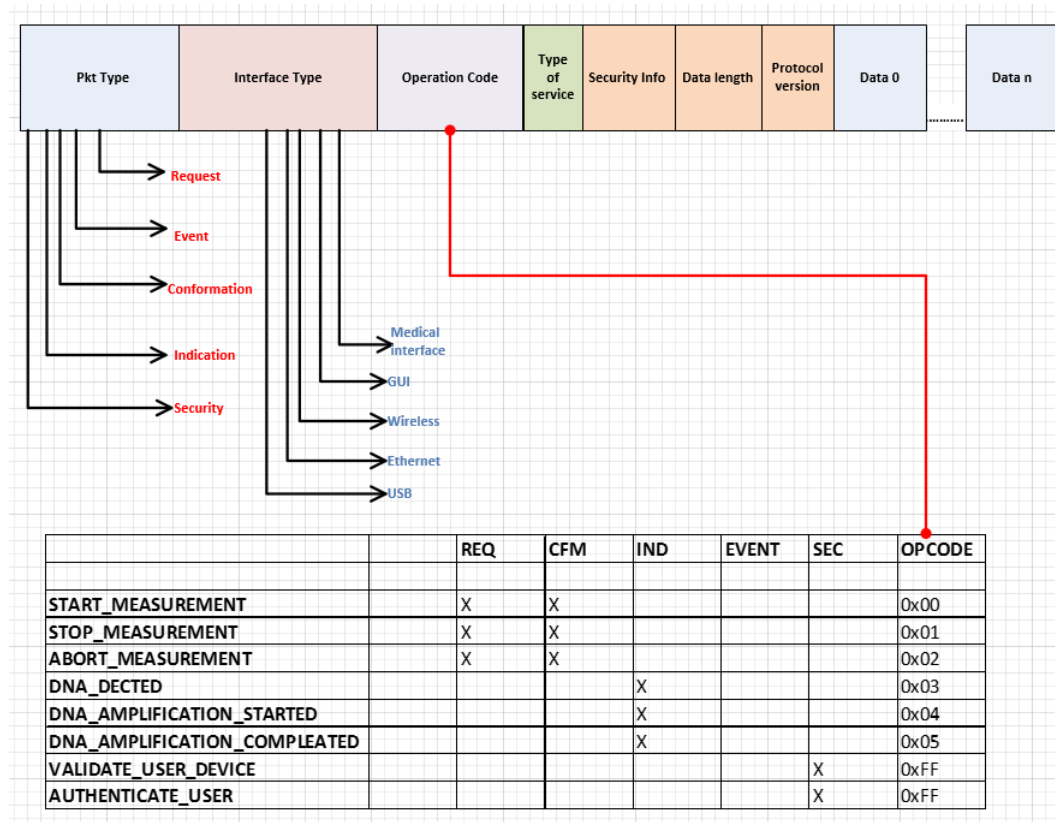


FIGURE 2.9: Communication protocol

commands in the POCT applications. The protocol is expandable and scalable as needed for processing the assays and it is independent of the type of POC testing application.

There are seven protocol data segments needed establish communication links between G-Node and P-Node is shown in Figure 2.9. Each protocol segment is organized as a byte (8 bits) or word (16 bits) format. The key to the usage of the protocol is the operation code (OPCODE). The OPCODE captures the behaviour or the sequence operation needed to execute an assay. Since this is a framework for using the protocol, the OPCODE is user a defined entity.

2.5.1 PKT (Packet) type

The packet type has five significant bits that indicate the type of command that the P-Node can understand. They are named: REQ (Request Bit), EVENT (Event Bit), CFM (Conformation Bit), IND (Indication Bit), and SEC (Security Bit). The REQ bit indicates that the P-Node receives a request to initiate an assay. It is up to the application developer to map the request to the assay test that pertains to the P-Node. The mapping function can be hardcoded, or it can be user configurable based on predefined process mapping. In other words, the REQ triggers a set of functions which are needed to run a particular assay. If the P-Node is capable of testing multiple infectious types, then the OPCODE can be defined to map relevant functionalities.

The EVENT bit indicates (and provides guidance) on what kind of event needs to be communicated. The event type is defined in the Operation Code data segment. The examples for the event type operation codes (VALIDATE-USER-DEVICE, AUTHENTICATE-USER) are shown in the table contained in Figure 2.9. One of the use cases is that the event occurs within the P-Node and the event information is needed to display a message on the G-Node. The OPCODE can be added for the other use cases pertaining to another type of test. Only the registered events will be reported to the G-Node. One of the bits in the OPCODE can be assigned to record the G-Node registration. There are no hard and fast rules regarding which bits will be used for capturing the functionalities.

The CFM bit indicates an acknowledgement (ACK) to the sender of the command. The sender will use the confirmation bit to implement further response or process the data in its domain, based

on the requirement. The IND bit segment is used to inform any connected devices about the status of the testing. It is up to the biologist to design the mapping information as shown in the table (e.g. DNA-DECTED, DNA-AMPLIFICATION-STARTED, and DNA-AMPLIFICATION-COMPLETED). In a practical system, the biologist will create the mapping table and hand it down to the application developer to code it.

The SEC bit can be adapted for many uses. One of the usages of the SEC bit is that it can indicate the type of encryption used and an OPCODE will be created for the type of encryption used. The combination of the security and the operation code (SEC-BIT, OPCODE) will inform the communication parties about the encryption scheme (standard and proprietary) in use so that at each end the decryption can be done as per the encryption scheme used. Note that if the security bit is not set then, there is no encryption is implemented. .

2.5.2 Interface type

The interface type supports five categories (mapped to individual bits) of information that needs to be communicated to the entities. The role of the interface type is to define user access to the P-Node. The medical interface is a particular interface type, and it informs the P-Node that the command sender supports the IEEE 11073 BT stack[52]. The sender device entity has certification of IEEE 11073, and it has an easy way to interface with sensor devices. The GUI interface type bit indicates that the command data is sent from the P-Node Graphical User Interface (if P-Node has built-in display unit). It informs the P-Node that the user dozent have access to a G-Node

at the time the command is sent from the GUI. Further, the test data must be kept within the device until it is sent to the P-Cloud.

2.5.3 Operation Code

The operation code defines the operations that are possible with the P-Node. The OPCODEs are user-defined as required for the device. The OPCODE can be set to 8 or 16 bits in length. The flexibility of having user-defined OPCODEs helps to customize the P-Node for a particular type of testing. An 8-bit OPCODE length will result in 256 types of operations for the device. A 16-bit OPCODE length will produce 65536 operation codes, which may be not be needed unless complex functionality is implemented. The OPCODE-based system is flexible, so that any future functionalities can be enabled with minimal changes to the implementation.

2.5.4 Type of Service (TOS)

The type of service field is an optional field that can be set depending on the application. For example, the TOS codes in the context of the testing can be defined by the P-Node device manufacturer and the associated ecosystems. The ecosystem for the testing would consist of many players, namely, P-Node vendors, POCT service providing communication carriers, POCT application developers (both for the device based UI and the applications written for the G-Node), and healthcare providers. Thus the TOS code provides bridges between the ecosystem (support system for providing wireless services) players for the system deployment. One of the uses of the TOS code is for billing among the ecosystems entities.

2.5.5 Security Information

The purpose of security information is to provide information regarding the security encryption method used in the device. If the security info field is set to 0x00, this means that the data is not encrypted. Note that it is possible to transmit data safely without any encryption because of the closed nature of the system. One should know the format of the data stream to understand the measured data from the POCT device. The SEC bit can be combined with the security info segment.

Whenever data is sent between the G-Node and P-Node, there is a preamble data segment (field) sent first. The detail of the preamble fields is shown in Table 2.2.

Data packet	Purpose	Direction of data flow
preamble-data-segment	Represent starting data segment of transmission	G-Node to P-Node
		P-Node to G-Node
end-data-segment	Represent end data segment of transmission	G-Node to P-Node
		P-Node to G-Node

TABLE 2.2: Preamble and End Data Segments

The same concept is explained in Figure 2.11. The first step is to start the transmission and decide the **preamble-data-segment** (*pd-segment*) and the transmission **end-data-segment** (*ed-segment*) for communication. In the context of the POCT device, POCT and POCTRES are chosen as the preamble data segment and the end data segment. The POCT is an acronym for Point of Care Test, and the POCTRES is an abbreviation for Point of Care Test Response. These are user-defined values, and they can be constructed as 8 or 16 bits (or user-defined data length), as needed by the ecosystem entities. Unless the values of the preamble data segment or the end

data segment are known, it is difficult for an intruding entity to sync-up with a transmission between the P-Node and the G-Node for the purpose of stealing the actual measurement data. This can be named as open communication without the need for any encryption schemes.

The values of the pd-segment and the ed-segment fields can be changed periodically for enhancing data security. The frequency of the period of change can be shared using the security info field. The algorithm shown below explains the management process of communication between the G-Node and the P-Node, while the values of the pd-segment and the ed-segment are changing periodically. Note that the algorithm segment needs to be implemented in only in the P-Node code. The key to the communication mechanism is that G-Node is the master that informs the P-Node of what needs to be done. Three type of encrypted communication processes are possible: standard security algorithms based encryption and proprietary encryption algorithms (explained in Chapter 4), open communication as explained in the Figure 2.10 and a hybrid of the two methods.

The high-level algorithm is shown in Figure 2.10. It starts with sending a POCT command to the P-Node (*line: 2*). The security-info is decoded and, if it is set for open communication, records the communication method in a status flag (*line: 4-5*). The communication reception starts with the preamble DOC (user defined) and ends with DOCRES (user defined). The reception of the DOC is noted as is the number of bytes expected by the receiving node to be determined (*line: 6-10*). This process will continue until the ed-segment (DOCRES) is received (*line: 11-19*). An error condition will be declared if all the test data is not received. The error will be

1	High level Algorithm				
2	Send DOC and the command to initiate assay process to P-Node				
3					
4	IF	(Encryption in Security-Info is set to 0x00){			
5		indicate that the communication is based on OPEN Communication Method			
6	IF	(OPEN Communication Method is in progress) {			
7		IF (DOC is received){			
8		IF (A tracking counter is not running) {			
9		Read Security-info // contains counter-value, timer-value			
10		Set counter to counter-value			
11		}			
12		ELSE-IF (counter-value is not zero) {			
13		counter-value = counter-value-1			
14		}			
15		ELSE-IF (DOCRES is received) {			
16		}			
17		ELSE-IF (Can wait for a defined time based on timer-value){			
18		Wait for DOCRES			
19		}			
20		ELSE {			
21		Declare ERROR condition			
22		Log the error			
23		Inform ERROR-CODE to all participants			
24		}			
25		END-IF			
26		}			
27		END-IF			
28		}			
29		END-IF			
30		}			
31		END-IF			
32		}			
33		END-IF			
34		}			
35	}	END-IF			
36	END-IF				

FIGURE 2.10: Core pseudocode for data TX and RX

logged, and also all the nodes in the communication path for providing a response to the user (*line: 20-24*) will be informed. The receiving node reassembles the data using order index value which is a variable part of the test data bytes. The test data bytes have the index value and the raw data. By the end of these processes, there will be new values for the pd-segment and ed-segment, pertaining to the next set of test data. The new values will be used until a new value is received in the security info field. Note the security info field is used for multiple purposes. Since this is a framework communication protocol, an application developer can use the field for any other user-defined propose.

2.5.6 Data Length, Protocol Version, Data bits (bytes)

The data length indicates the number of data segments (measurements points) involved in the testing. Note the number of data segments is not limited. The only limitation is the size of the user memory (storage size) in the POCT device. The protocol version is used to maintain interoperability within the ecosystem. The application developers must use the same version of the protocol for the communication. The data bit (or bytes) is the measurement point of the outcome of a particular assay. Usually, these are (time, data) pairs. The time (measurement intervals) are user-defined, and they can be set from the G-Node or predefined at the P-Node level. The OPCODE can be used to trigger the required measurement intervals, as needed.

Commands	REQ	CFM	IND	EVENT	SEC	OPCODE
START_MEASUREMENT	X	X				0x00
STOP_MEASUREMENT	X	X				0x01
ABORT_MEASUREMENT	X	X				0x02
DNA_DECTED			X			0x03
DNA_AMPLIFICATION_STARTED			X			0x04
DNA_AMPLIFICATION_COMPLETED			X			0x05
VALIDATE_USER_DEVICE					X	0xFF
AUTHENTICATE_USER					X	0xFF

TABLE 2.3: Protocol Command mapping to OPCODES

2.5.7 MSC for CMD communication

As shown in Figure 2.11, the message sequence chat describes interaction (after compatibility test). The compatibility test determines the protocol versions installed on both sides, i.e. the P-Node and the G-Node. If the compatibility test fails, the user will be informed using an appropriate error message, and the process of upgrading

the protocol versions is initiated. Once the compatibility issues are resolved, the user will be in a position to initiate the test needed. In

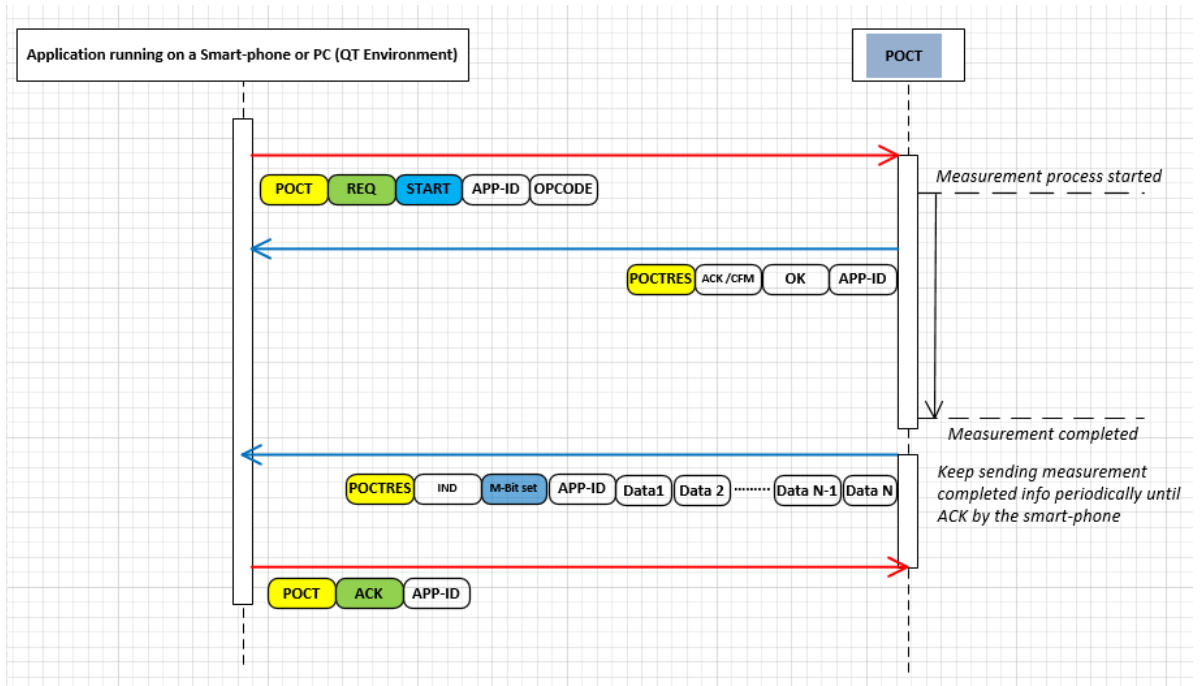


FIGURE 2.11: Data communication Message Sequence Chart

Figure 2.11, the command is sent to the device using the protocol table. This command represents a request (REQ) sent to the device, asking to start the assay process (by decoding the OPCODE). The request could have been initiated pressing a simple UI interface on the P-Node, as well. Once the measurement process starts, the device knows steps to carry out the requested assay. The device also sends back a confirmation flag (ACK /CFM) to the sender (G-Node). After the measurement process is completed, the device sends the data with indication flag (IND: M-Bit) with measurement bit enabled, along with the application ID (APP-ID). The APP-ID helps to track the requests between G-Node and the P-Node.

The G-Node sends back the acknowledgement (ACK) to the P-Node. The data is then transmitted periodically to the G-Node until the P-Node receives the ACK byte flag from the G-Node.

The user or the application developer can create a protocol table that is required for the test application. With this protocol framework, the application developers can build their implementation by creating additional entries in the table.

2.5.8 PROTOCOL EXPANSION STRATEGY

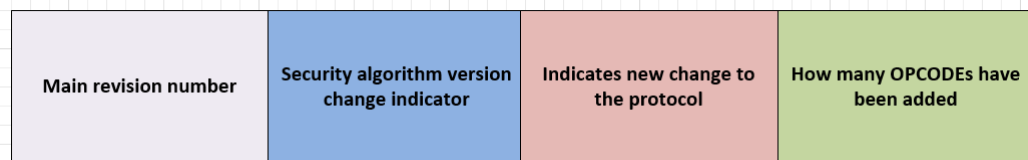


FIGURE 2.12: Protocol evolution

The protocol needs to be modified as new requirements are developed. The attributes shown in Figure 2.12 are proposed to manage the new changes such as the addition of new OPCODES for new types of assays that may be added in the future.

The G-Node and the P-Node must have a compatible protocol implementation. Before any testing is initiated, the compatibility of the two systems must be validated, as the testing process is a mission critical operation. The attribute Main revision number is to indicate that the SW in both G-Node and P-Node are compatible. The security is the essential functionality, and it must be kept in sync with any encryption key changes that may be needed due to evolving security threats. A change in security algorithms or encryption usually is sent to the POCT application installed on G-Node, assuming that the wide area connectivity (WAN) is provided by G-Node. It is also possible that the P-Node can receive the security alerts (via WAN), depending on the deployment scenarios (Chapter 6). In

any case, the attribute Security algorithm version change indicator reflects the security validation process.

New functional enhancements will be indicated by the attribute New change to the protocol. The other use of the attribute is to indicate any changes to the protocol structure. The variations in the OP-CODEs will be captured by the last attribute, How many OPCODEs have been added.

Before any testing related communication begins, the four attributes stated here can be used to verify if the G-Node and P-Node are compatible regarding SW. The user will be alerted by any mismatches. The tests will be allowed only if the attributes are compatible. The compatible mapping is left to the application developer to configure. Note that any human errors or configuration errors will result in data error, and the test data from such a system cannot provide valid data for diagnosis.

2.6 Experimentation with Android-based smartphone and Bluetooth

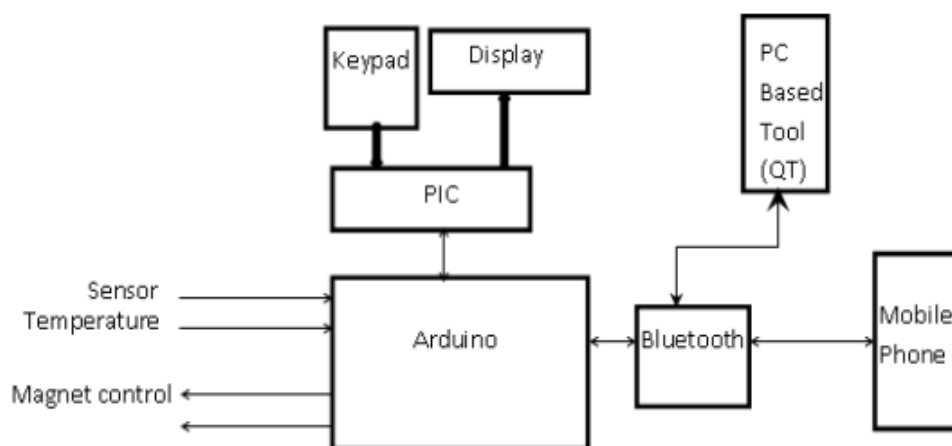


FIGURE 2.13: System setup

The system set-up for development is shown in Figure 2.13. The main components are the Arduino board and the Bluetooth board [47]. The actual modules used in the experiment with the Bluetooth module are shown Figure 2.14.

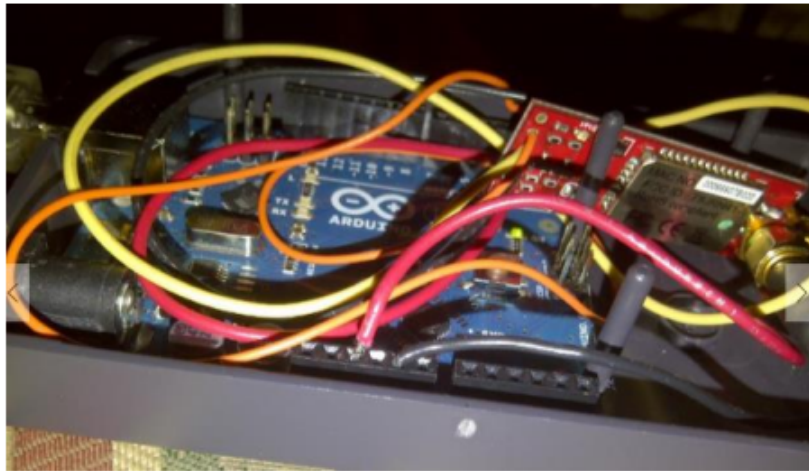


FIGURE 2.14: Arduino Uno and BT module

A PC based tool was developed based on QT (Nokias UI development platform) [53] technology (Figure 2.15). The QT IDE (Integrated Development Environment) represents the G-Node. The detail view of the IDE shows transmission and receive windows for testing the protocol (Figure 2.15). The QT-based tool is a development platform for modifying the protocol for future needs.

The Figure 2.15 shows an Integrated Development Environment (IDE) that was developed for experimenting with the protocol implementation. The IDE is based on the QT framework, an open source environment. It can be used for testing the POCT device components or peripherals such as motor (*LABEL-A*). *LABEL-B* indicates the transmission data window and *LABEL-C* indicates the receive data window. *LABEL-D* is the motor control instrumentation (speed set) for the POCT device. *LABEL-E* the controls for reading the communication ports and status indicators. *LABEL-F* is the mobile

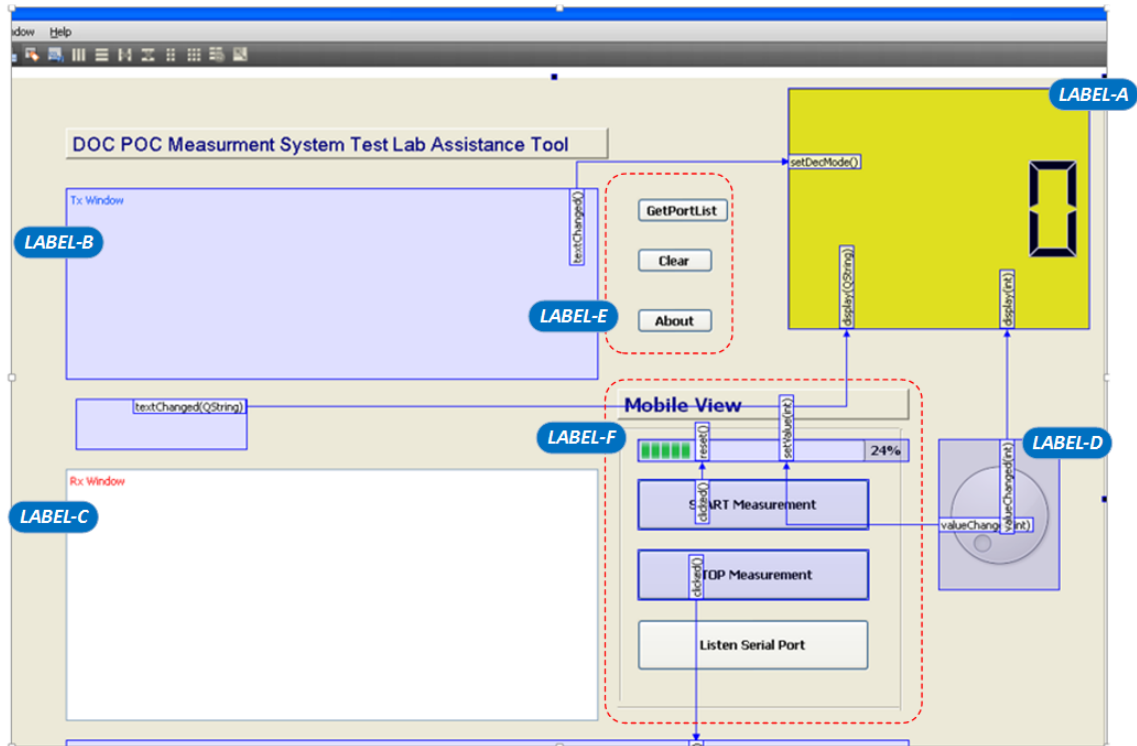


FIGURE 2.15: TX Window and RX window for protocol development - QT based IDE (Protocol Development Utility Tool)

device view for the application developers use. The purpose of the utility tools is to assist developers in creating POCT applications and testing changes to the communication protocol.

Figure 2.16 illustrates the reception of data from the P-Node. An Android-based smartphone was used as the G-Node. The BlueTerm application was used to receive the serial data via BT. Figure 2.16 indicates the data header POCTRES, the measurement bit, and the actual measured data. The application developer can use this data to create a smartphone application for the POCT. The POCTRES header shows the data header of the received data set from the POCT device. The Measured data availability is the bit indicating that availability of the test data. This can be used to trigger other application specific user interface controls on the smart-phone (which is G-Node). The Measurement data indicates a sample test data measurement sent from the POCT device.

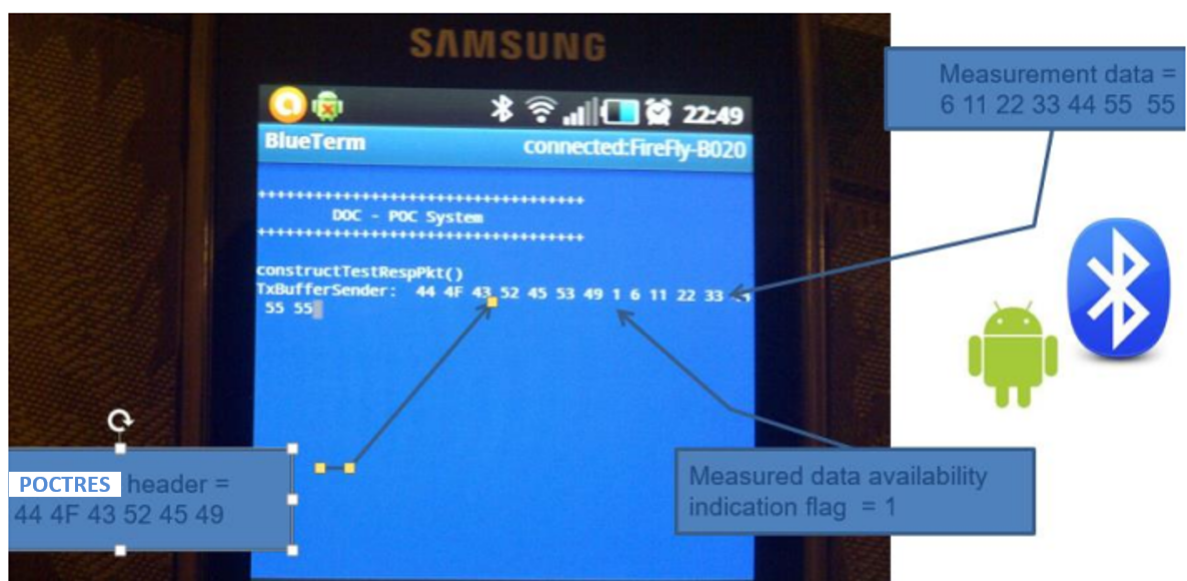


FIGURE 2.16: Protocol communication Data reception from POCT device)

2.7 Summary

The requirement defects are often the least expensive items to fix at the early stages of a project. The validated requirements form

the basis of many other work products in the project. Correcting defects earlier in the products development is more cost effective than attempting to correct them during the testing phase. The modelling techniques such as UCM and MUCM, facilitate the clear explanation of requirements and their textual descriptions. This chapter shows a novel way of using the Use Case Maps in requirements creation, while the MUCM is used to explain security requirements.

Based on the visual representation of the UCM, its adoption to express complex requirements will help to communicate requirements in addition to their text description. It helps to manage the project at all stages of product development: requirement elicitation, analysis and validation, product specification creation, verification, and management. An ambiguous requirement can be expressed in terms of the UCM visual representation. It is safe to state that if a requirement is difficult to express in the UCM, then the implementation of the requirement will be difficult for the downstream development teams.

The UCM will help to narrow gaps between the design and requirements because of the visual representation. Using UCM to express requirements will provide a clear traceability between test cases and the requirements. Hence, the miscommunication between the testing and design teams can be minimized. A new development project can be managed with low risk because the ambiguity in requirements is less with the use of UCM. A way of measuring the success of using the UCM for expressing the requirements can be done by gathering defect containment rates and defect finding rates.

P-Node architecture as explained in this chapter outlines twelve fundamental system functionalities for realizing a POCT device, including wireless communication interfaces. In a practical system design, the P-Node will have shared responsibilities between the communication processor (MODEM) and the application processor. The architecture described can be implemented in any Open Source hardware platform. Arduino is used for the research work, while Round-Robin software architecture is used for the experimentation. The Round-Robin looping is adequate for supporting the functionalities in the P-Node. P-Nodes are typical embedded systems with wireless connectivity. They usually have some kind of embedded operating systems such as Linux, which is capable of providing task-oriented software architecture. If an operating system is used, then it needs to be certified by the FDA.

The protocol developed for communication stays at the application level of the system software stack. It works with all types of RAT (radio access technologies): 2G, 3G, 4G, and 5G. It is independent of the radio layer, which makes it portable to all the current and future RATs. The protocol uses an OPCODE, which is a user-defined variable. The OPCODE is mapped to various events in the P-Node, and it can adopt all the events related to steps in the assays (user-defined). It supports multiple interfaces: USB, Ethernet, Wireless, User interface, and Medical interface. It is a protocol framework which can be modified by the application developer. Thus the protocol is scalable and expandable based on the need.

In this chapter, all the end-to-end aspects of systems development are reviewed. The key to successful product development is to have requirements that are understood across the all the functional teams.

The UCM methodology is an effective modelling process that expresses the system requirements in a visual format. The RAT independent protocol gives a stable and mature framework for system development, and it is flexible to meet the needs of the application.

Chapter 3

POCT Communication System Architecture Design Methodology and Processes

3.1 Introduction

In the medical devices industry, system failures can cost lives and result in fatal consequences. The faults can arise due to the interaction between the software, the hardware, and the operating environment. Unexpected environmental changes lead to software abnormalities that may have a significant impact on the overall success of the system operation. Latent coding errors can surface at any time during the system operation and trigger faults despite significant effort having been expended in verification and validation (V and V) of the software system. Extra efforts and spending considerable time in the V and V is not enough to guarantee that a complex software system meets all safety, security, and reliability requirements [54]. Medical devices are particularly partial to interoperability issues due to incompatible data formats that lead to fatal consequences in the interconnected network environment [55].

Point of Care Testing (POCT) is becoming an increasingly popular method compared to standard laboratory testing for the diagnosis of infectious and genetic diseases. This testing methodology has shown many advantages such as diagnosis time reduction, cost reduction, portability and better process control due to the automated nature of the POCT device in a hand-held or bench-top formats. This reduces the risk of human error. The decrease in diagnosis time and rapid onset of treatment has been shown to improve patient outcome in emergency settings [56]. In certain instances, diagnostic errors such as false positive results can occur, which could lead to unnecessary treatment or a delay in therapy [57]. Therefore, it is vital that a good quality assurance process is needed for the POCT.

The POCT devices use the automated process control for measuring parameters pertaining to medical diagnosis (assays administered in laboratories). It is deployed in a network connected environment and it meets the basic definition of the Safety Critical System (SCS). In this context, this chapter shows how inherent complexity of software development process cycle for implementing the system can be project managed, designed and verified, using the methodologies, tools and system design concepts that are followed in prototyping the device and its communication system. These software systems developing processes will help in eliminating (or minimizing) fatal consequences of errors made in software coding. In other words, a holistic strategy of developing software (SW) is needed.

The chapter contents have been divided into different sections. Each section addresses the communication design development process with various aspects of the development cycle to produce the end-2-end system. In section 3.2, a requirement eliciting methodology called Value-Based Requirements Gathering (Pyramid Requirement

Eliciting Processes) is described. This process eliminates ambiguity in requirements. This is essential for developing any safety critical system. In section 3.3, a system design process and technique that is used in the development of the system end-to-end architecture is described. The system identification process will provide a basic abstraction view of the system that needs to be designed and to be implemented. In section 3.4, the role of security design in the development of the system is explained. In section 3.5, operational scenarios used have been discussed. In section 3.6, a cloud-based architectural strategy for the de-vices has been presented. In section 3.7, justification and benefits for adopting recommendations from IEC 62304 at the early stage of the development cycle are given. In section 3.8, a process generating testing strategy based on combinatorial design methodology has been described. This methodology shows how less number of test scenarios guarantees 99 percentage functional coverage as op-posed to spending time and resources on exhaustive testing. In section 3.9, a recommendation for project management methodology that will potentially help to eliminate software errors is discussed. Finally, in section 3.10, a summary is presented.

3.2 Pyramid Requirement of Eliciting Process

3.2.1 Issues with current process

The FDA has analysed 3140 medical device recalls between 1992 and 1998. These reveal that 242 of them (7.7 percentage) are traceable to software failures related to device and associated communication links. Those software related recalls, 192 (or 79 percentage) were linked to software defects that were introduced when changes were made to the software after its initial distribution [58]. One of the

reasons for the failures was that the requirements were not fully understood by the down-stream development teams when changes were made.

Figure 3.1, shows the traceability relationship between requirements and the downstream teams: design, software coding, testing and deployment teams. The Figure also shows those portions of the system with errors increase as the development cycle moves from the requirement phase to the implementation phase. Note that the errors portion at the design and SW coding level is greater than the errors portion at the requirement level. Less error at the requirement level will have a cascade effect at the last stage. It is very clear that by controlling and eliminating the errors at the requirement phase can be minimized at the implementation level. At any given phase of development, the domain experts concentrate on their main deliverables and have limited awareness of information from previous phases.

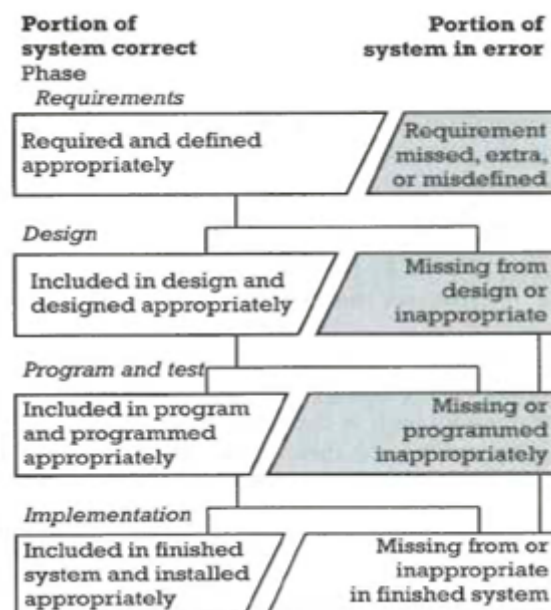


FIGURE 3.1: Sources of errors and phases in development cycle [3].

3.2.2 Pyramid Requirement Model

In order to eliminate the errors that originate from the requirement phase, a methodology is presented here. It is known as Problem Pyramid modelling and is amalgamated prior to creating any requirements. The requirements become unclear because they are not tied to key measures. The pyramid modelling provides a way to associate the key measures that the requirement needs to fulfil as shown in Figure 3.2. Each block is identified with a flow, which will help to create a meaningful and error-free requirement. The requirement can be refined methodically by having the iterative refinement process (via stages 1 to 2 to 3 to 4 to 5) until all the stakeholders are satisfied.

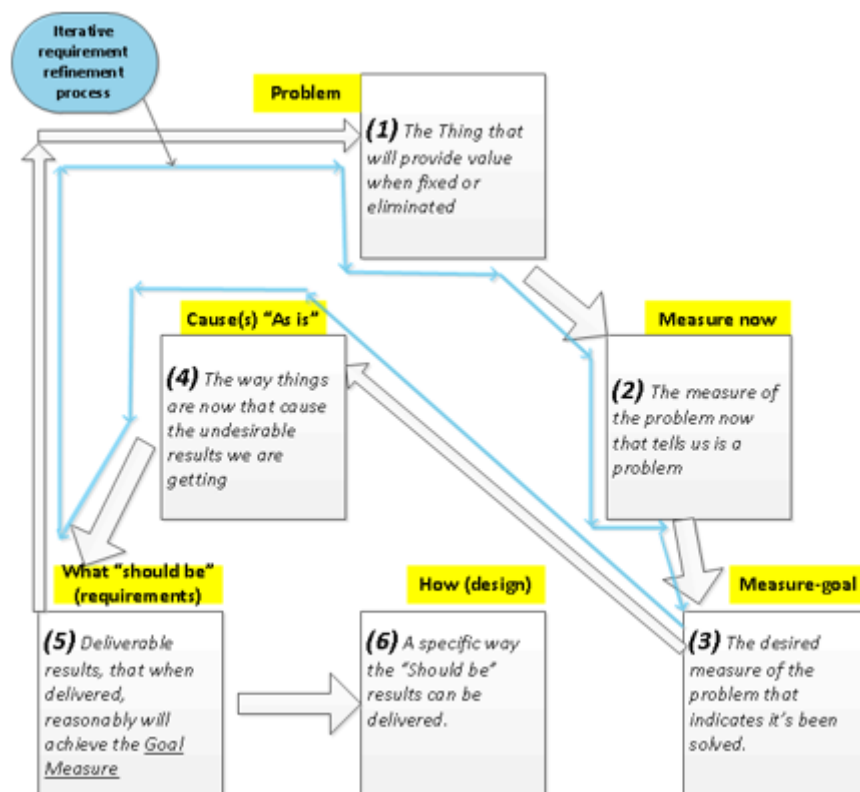


FIGURE 3.2: Problem-Pyramid [3]

At stage (1) the problem is defined. At stage (2) a measurement value that confirms the existence of the problem is stated. At stage

(3) the goal is defined in quantifiable terms to indicate the desired measured value. At stage (4) causation analysis to understand the problem domain is done. As shown in the Figure stages (2) and (4) are at the same level. Also stage (4) reveals the rationale needed in creating the requirement. At stage (5) the requirement is formulated. At stage (6) the implementation options are stated. It is very unlikely that a System Engineer who focuses on the Program and Test phase (Figure 3.1) will detect an error propagating from the Requirement phase (Figure 3.1). However, the use of the Pyramid Model would prevent error propagation and help create error-free requirements. In other words this methodology gives more importance to technical and business values that the requirement brings to the device development [3].

An example of formulating an error-free requirement for the device communication operation is shown in Table 3.1.

(1)	<i>Problem:</i>	POCT Device user needs to input appropriate communication parameters when configuring the device
(2)	<i>Current Measure:</i>	Allows to enter only cellular RAT selection and unable to select WiFi or BT
(3)	<i>Goal Measures:</i>	Must be able to select both the cellular as well as other networks
(4)	<i>Causes:</i>	Keypad entry does not differentiates entry
(5)	<i>Should be:</i>	Keypad should differentiate the entry for any of the identified communication links
(6)	<i>Design:</i>	Add universal select function to the keypad

TABLE 3.1: Application of Pyramid Process

In this example, items (1) and (2) represent the problem domain; item (3) states the desired outcome; item (4) identifies the problem causes; item (5) states the requirement to accomplish the desired goal; item (6) states the design view of how the requirement can be implemented. With this kind of information it is very hard to make software related coding errors because the requirement is tied

to a measurable goal. This process increases visibility and access to actual requirements for the downstream team (software development and verification teams). This will encourage all the teams who are in the various vertical layers to collaborate effectively to produce error-free software for the medical devices.

3.3 System Design Process and Techniques

3.3.1 Basic Building Blocks Identification

Analysis of medical devices recalls reports, in the FDA database from 2005-2006 shows 109 software related recall cases. The main recall reason for the high-risk device is that the device contains many system design defects. Though the ratio of Class I devices with high risk was declining in 2006 compared to 2005 [59], the FDA data still shows the repeatable occurrence of device recalls. Directives and legislation have been proposed for software vendors for preventing product failure [60]. The design defects are the results of failure to understand the intent of the requirements by the design team and the lack of architectural view of the end-to-end system. Formulating the basic building blocks of the system that is being developed will help to design a fundamental DNA of the system architecture. Once the basic building blocks are understood, they can be organized or connected in such a way that a service (functionality) can be implemented to meet user requirements.

The basic building block view of the POCT end-2-end communication system consists of P-node, G-node, local server and P-cloud (Figure 3.3). The P-Node is the representation of the POCT device. The G-Node is the representation of the gateway entity, which can

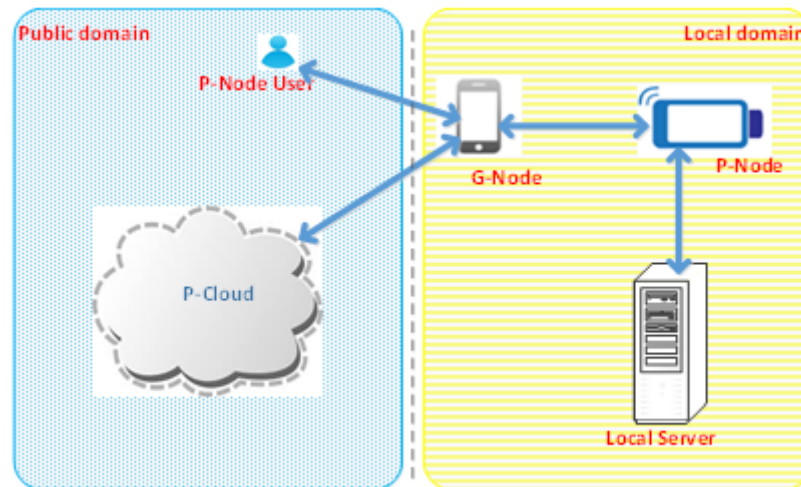


FIGURE 3.3: Basic building blocks of POCT System

be a smartphone or a laptop which is used to send measurement data to the database server. The P-Cloud is the representation of the secure private data cloud which has special interface access requirements such as UKs NHS digital [61] cloud infrastructure, USA Cloud Standard Customer Council [62] and Canadian Info way [63]. The P-Node user is the patient who is authorized to run tests using the G-Node.

In general, there are two main root classes of the mode of operation of the system. When a request for testing is initiated from the G-Node to P-Node, the mode of operation is defined as user functional mode. When the data is sent from the P-Node to the P-Cloud after the test, the mode of operation is defined as reporting mode. In other words, in the User Functional (Control) mode the request from G-Node and response from P-Node are communicated to the P-Node user (patient / health-care worker / clinician). In the Reporting mode, data is sent back from P-Node to G-node. These are two main abstracts of the high-level communication scenarios for the system.

3.3.2 Isolating System Module Responsibilities

Once the basic building blocks have been identified, isolation of the functionalities and creation of functionality boundaries are done by assigning responsibilities to system components or architectural modules. This process creates responsibility zones within the system architecture. A similar idea is encouraged in UCM (USE-CASE MAPS) methodology [35] which is a popular system modelling process for the complex architecture of interconnected systems. More in depth discussion on the UCM can be found in Chapter-3.

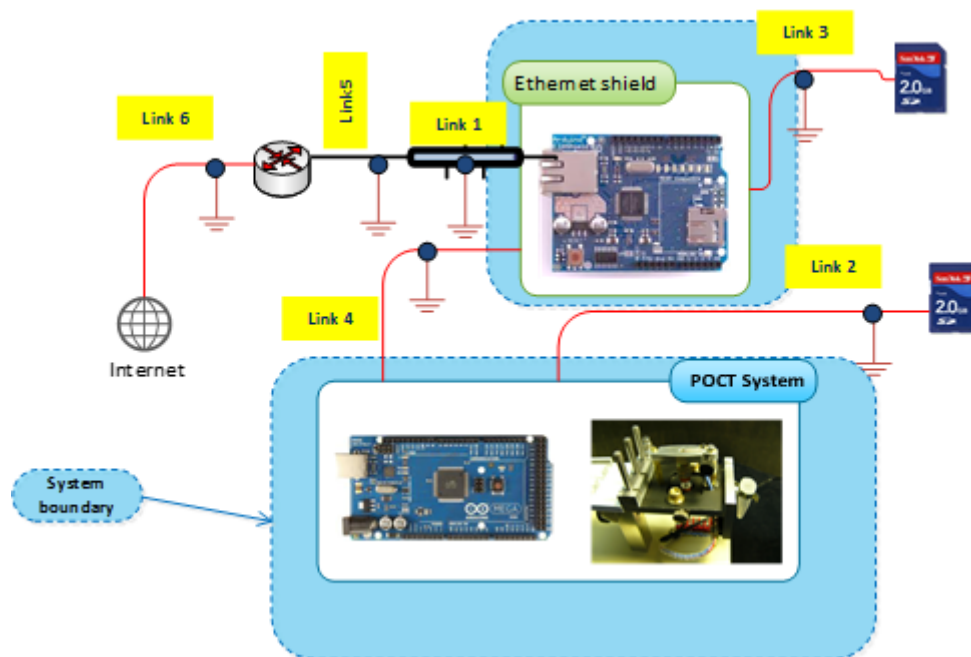


FIGURE 3.4: Isolation of Functionalities and communication failure points

Figure 3.4 shows a typical setup for the POCT communication links, using the Arduino open-source modules. POCT device module is the core of the testing processes which executes as-says based on the commands received from the G-Node. The resulted test data will be transferred to the P-Cloud via the Ethernet module. There are internal communication links shown as Link 2, 3 and 4. When the data is

transmitted, the data path will be *Link-4 (internal)...Ethernet-Link...Link-1 (external)...Link-5 (external)...Link-6 (external)*. Link 2 (internal) and Link 3 (internal) are used for interfacing with the SD cards. Failure points are indicated as ground symbols. If the complete system is built in one single module, then any of the failures will shut down the entire module. By having isolated modules as shown in the Figure, link failures can be managed effectively and this system will not lead to a single point failure. System functional boundaries are indicated by dotted lines. In other words, the functionalities are independent of each other with internal communication links. Upgrading system software also becomes easy with the isolated and interconnected module architecture. Functional testing is also effective with the isolated functional architecture.

3.3.3 Creating loosely coupled modules

The loosely coupled system is a product of the isolated and interconnected architecture. The loosely coupled system has many advantages in service-oriented architecture [64–66]. This helps to implement the system independently because of the nature of the loosely coupled system. In practical terms, this kind of system architecture allows more stable and dependable products as the faults or fatal errors in one subsystem will not impact the other system components. The interaction between loosely coupled systems is vital for producing desired system behaviour as per the system requirements. This communication between the loosely coupled system components should be of asynchronous messaging to maintain non-interdependency. The following paragraphs explain how this is implemented in the system architecture.

The loosely coupling implementation in the devices is used both in software and hardware components. The internal communication in the Arduino stackable system uses multiple technologies depending on the vendors of the module. In general they use SPI (Serial Peripheral Interface Bus), I2C (Inter-Integrated Circuit), and Serial (Serial Communication) (or similar interface which is available in open source hardware) to share common data. This will provide the following advantages:

- Independent systems operation (i.e., each module can function on its own)
- The system has high fault tolerance, or the faults can be contained within the subsystem boundaries
- The safety classifications (system partitioning) can be applied easily as per the IEC 62304, the standard for the medical device software life cycle process.
- The system can be easily validated and verified as sub-system units and single End-2-End system.
- This type of system design paves the way for the expandable system (encouraging scalability) whenever requirements changes or the device needs to interface with 3rd party systems.

When designing mission critical systems such as medical devices, the couplings (hardware based or software based) need to be minimized in order to eliminate single point failures which will impact the whole system. This includes assigning hardware architectures to multiple hardware modules or platforms. The loosely coupled systems help internal communication within the device.

Figure 3.5 shows the complete end-to-end topology of the communication system (using Arduino stackable modules) which was developed [32]. The principle of isolating the functionalities and assigning the functionalities to multiple hardware modules is followed in developing the topology. Link 1 connects the router to the Ethernet interface, Link 2 and Link 3 enable SD storage, Link 4 provides Ethernet connectivity, Link 5 connects router to the MODEM, Link 6 enables Internet link, Link 7 provides touchscreen access, Link 8 provides Bluetooth connectivity, Link 9 enables 2G cellular communication, Link 10 provides external 3G cellular connectivity and GPS data, Link 11 enables user with keyboard interface, Link 12 creates internal communication path with the POCT core module and provides access to the P-Cloud via cellular connectivity. Thus the internal communication links are identified within the HW components. This isolated architecture helps to recover errors quickly during system operation (i.e. POC testing process) as well as maintenance of system modules. This also creates an environment for testing the functionalities during the system integration tests, which are done by verification and validation teams.

3.3.4 Actuation Channels and Actuation Confirmation Channels

G-Node needs to communicate with P-Node to control the device as well as external links. Following paragraph shows the implementation of the communication of the internal functions of the device.

There are external peripherals (e.g., coils and actuators) used in the POCT device. Cross verification of activation of control outputs signals (via feedback) that drives the external peripherals needs to be verified for all the outgoing actuation scenarios such as enabling coils

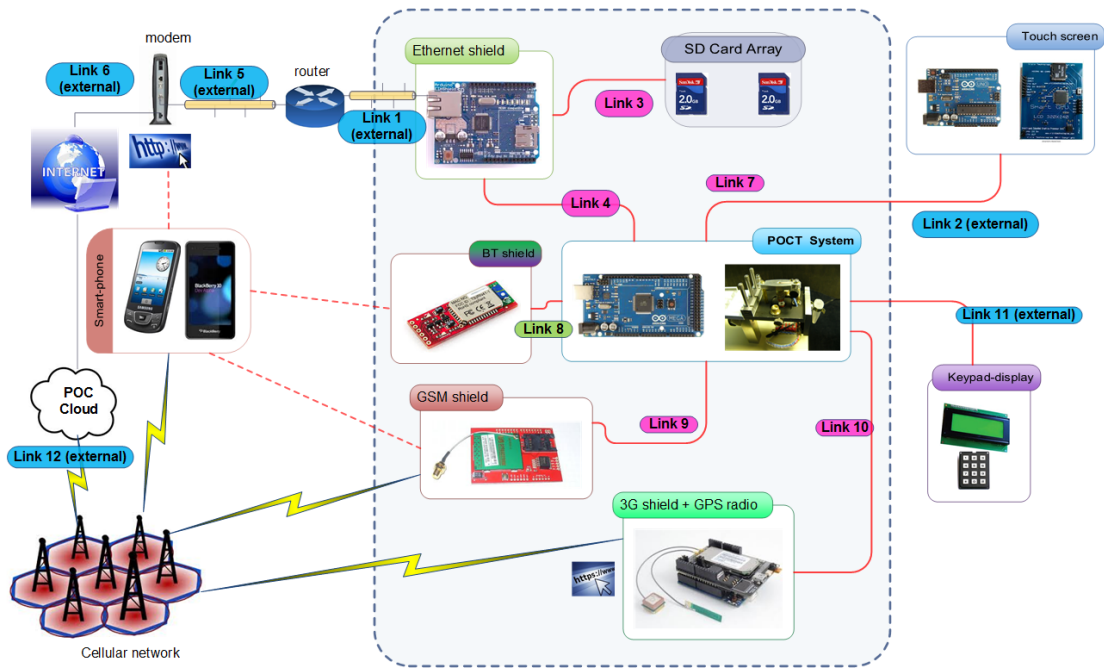


FIGURE 3.5: End-to-End Topology (external and internal communication links)

or opening electromagnetic valves. The cross verification is used in automated train control systems or autonomous cars (cite) - invariably in all the mission critical and safety systems. This concept is applied to POCT system. For the POCT system, an actuation / feedback cross verification mechanism is developed. There are two groups of control channels, ACH (actuation channel) and ACCH (actuation confirmation channel) used. The actuation channel is responsible for activating (e.g., setting a voltage to an actuator) a peripheral. The actuation confirmation channel is responsible for verifying if the actuation was successful. In the POCT system design, this will be accomplished by an actuation control word which is 16 bits variable and each bit (or group of bits) corresponds to a peripheral. This is because some of the peripherals will need multiple input signals to activate them. The actuation confirmation channel has a corresponding word (a 16 bits entity) maps to the actuation control word.

The flowchart in Figure 3.6 shows a way of implementing the concept

in the software. The description of each block and the data flow are shown.

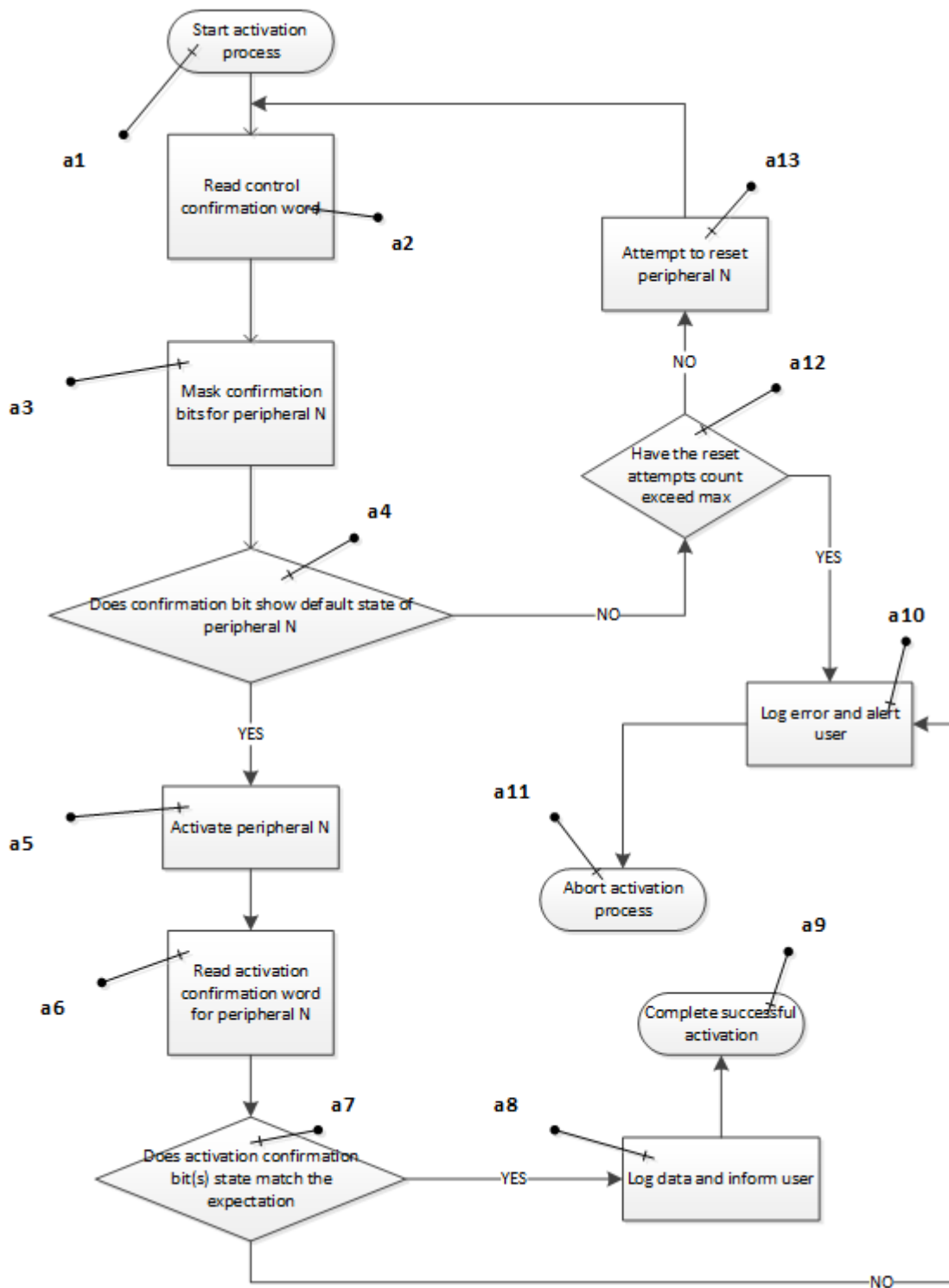


FIGURE 3.6: Algorithm for Peripheral Activation

The explanation for each actions flowchart is described below:

[a1, a2, a3]: Default state of the peripheral is read - which needs to be activated, before the activation process function is called. The control confirmation word which represents the default state of the peripheral is acquired via the ACCH. Information pertaining to the peripheral of interest is masked out from the control confirmation word (also known as a status word).

[a4]: The control confirmation word is verified for the expected state of the peripheral.

[a12]: Reset attempts will be made if the expected state of the control confirmation word is not correct. This process will be repeated for a predefined number of times.

[a5]: If the status word is read and confirmed, then the peripheral will be activated (to support a biological process, e.g., a coil is turned on to generate required temperature).

[a6, a7, a8, a9]: The activation confirmation word will be read, and it will be verified that the activation was successful. And the process will continue.

[a7, a10, a11]: If the activation confirmation word indicates that the peripheral was not activated correctly, then the process will be aborted, an error log will be created and the user will be notified.

[a10, a11]: If the reactivation attempts failed, then an error log will be created, the user will be alerted, and the process will be aborted.

Table 3.2 shows an example of an 8-bit control (ACH control) word and an 8-bit confirmation word (ACCH confirmation). The 1 indicates logic high and 0 indicates logic low. The X means, it is a dont care logic state with respect to the associated peripherals. There

are three peripherals in the system (stepper motor, micro-pump and micro-valve).

Actuation WORD		Peripheral					
		Stepper Motor		Micro-pump		Micro valve	
ACH (control)	ACCH (confirmation)	Input	Output	Input	Output	Input	Output
ACH-0	ACCH-0	1	1	X	X	X	X
ACH-1	ACCH-1	1	X	X	X	X	X
ACH-2	ACCH-2	X	X	1	X	X	X
ACH-3	ACCH-3	X	X	0	1	X	X
ACH-4	ACCH-4	X	X	X	X	X	X
ACH-5	ACCH-5	X	X	X	X	X	X
ACH-6	ACCH-6	X	X	X	X	1	1
ACH-7	ACCH-7	X	X	X	X	1	1

TABLE 3.2: Example of ACH and ACCH

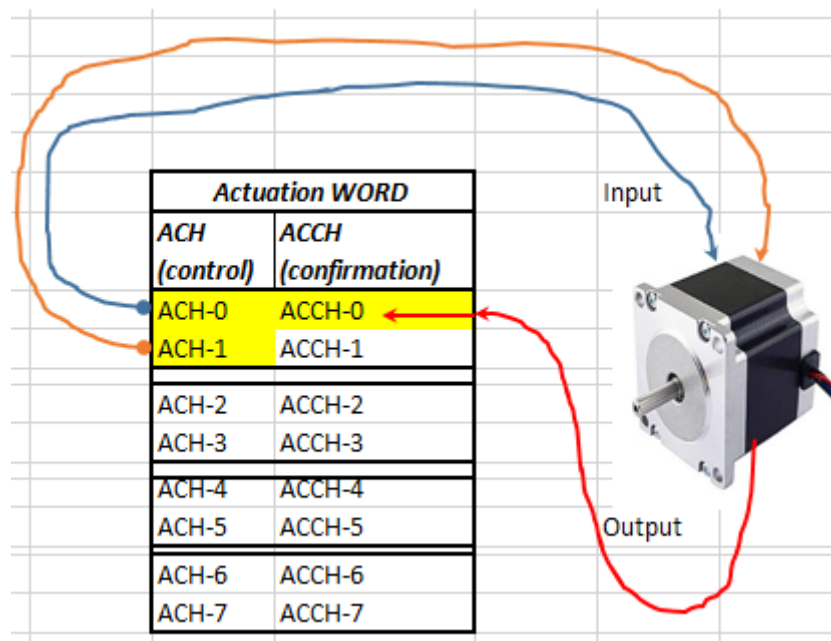


FIGURE 3.7: Conceptual connection of ACCH and ACH to stepper motor

Figure 3.7 shows logic connections between the actuation words which are derived from the controller of the POCT device and the stepper motor driver. For the discussion here, both ACH-0 and ACH-1 are set to logic 1 (Table 2-2). These logic levels drive the stepper motor. Feedback from the stepper motor is sent back to the POCT

controller via the ACCH word (BIT ACCH-0). The ACCH-0 informs the POCT controller that the stepper motor has responded to the controls ACH-0 and ACH-1. Thus the mission criticality of the testing process is closely monitored to preserve the integrity of the POCT process. Two other such mappings are shown in the table. For mi-cro-pump, both ACH-2 and ACH-3 are used and the drive logic levels are set to 1 and 0. The feed-back is received by ACCH-3. The ACCH-3 will be set to 0 if the micro-pump operates properly. For micro-valve, both ACH-6 and ACH-7 drive its inputs and ACCH-6 and ACCH-7 receive confirmation from it. The signals marked X are not used in the context of the connected peripheral. There are many benefits of having this strategy: detecting component failures before starting the tests; automated way of having component maintenance; a postmortem of success or failure of the device operation. It also provides a guarantee in collecting accurate test measurement data information and hence accurate diagnosis.

3.3.5 Security Principles for Designing Safety-Critical Software

Communication within the device boundary and external to the device must be governed by security principles. All access to user data (measurement data and configuration data) must be approved by the user and minimally a protected password must be enforced on all interfaces that can access this data, including hardware interfaces. This includes protection of data removed or copied from the device: memory dumps, cloud server access and other data storage. The device is linked to a smart-phone (via Bluetooth) only when the smart-phone access is authorized by the user. When the measurement data is sent over to the P-Cloud, the database link (M2M Machine to Machine communication link) access must be password

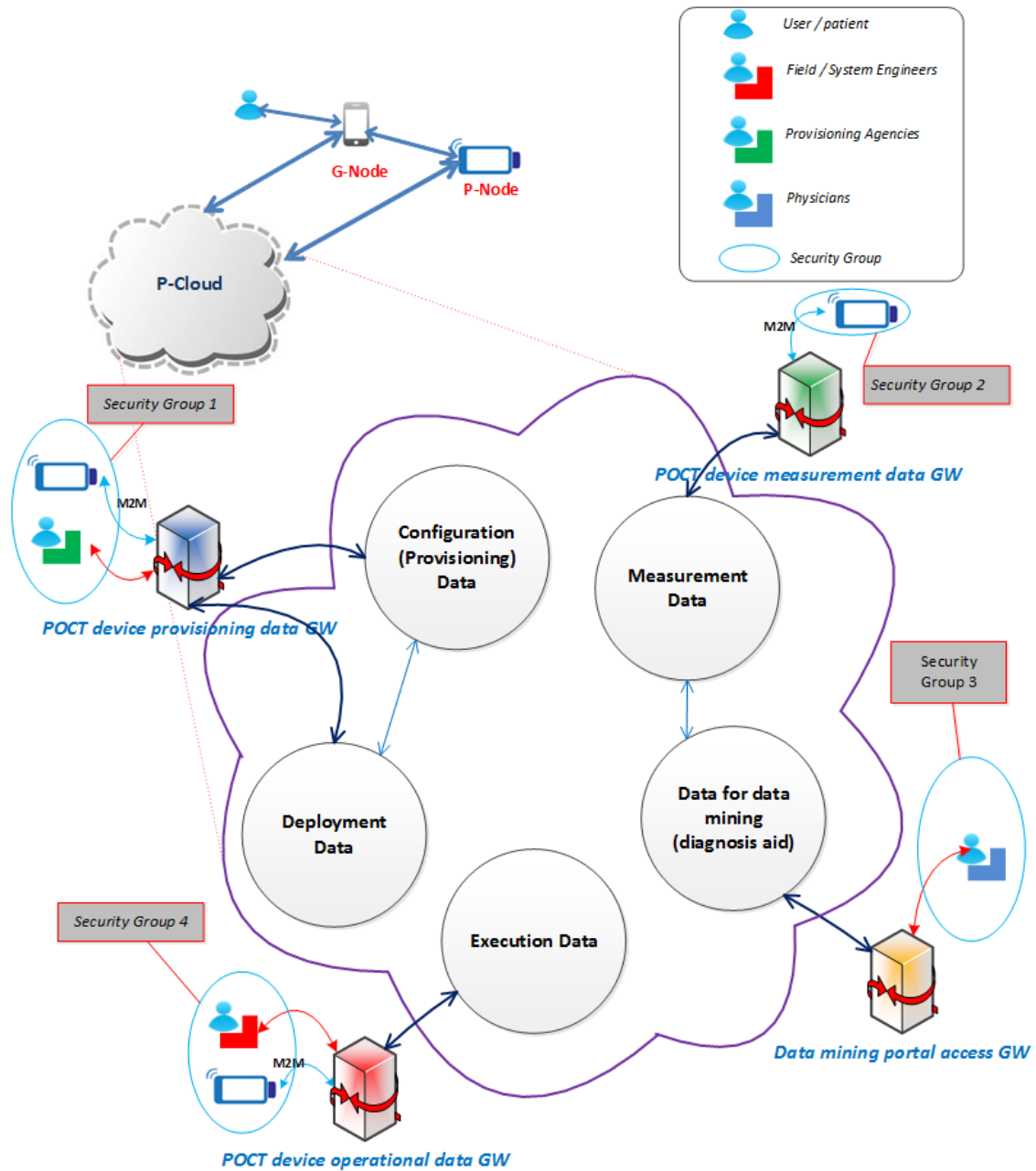
protected. At any time, if there is an opportunity to improve security at the: end-2-end system level; subsystem level; software and hardware component level; software API level; data structure level - required changes should be implemented. Relevant security requirements must be updated and they must be shared among all other stakeholders during the life cycle of the device. If a design team finds an opportunity for improving security, that information must be documented. After appropriate impact analysis, all the impacted teams must be informed about the changes needed. The opportunity for improving security must be encouraged regardless of the bottom-up or top-down direction of the team hierarchy.

Some security experts believe that security through obscurity is one of the security measures, which creates system design ambiguity to confuse security attacks. However, this process of dis-closing system design as insecure systems can lead to catastrophic security failure [67]. Automation systems have relied on security through obscurity to solve computer attacks problems. The tools needed to conduct these attacks are easily obtainable for free, and the potential consequences of an attack are large [68]. A system relying on security through obscurity may have theoretical or actual security vulnerabilities. Therefore, obscurity should not be relied upon as part of the security design, but can (and should) be used as a mitigation strategy against unknown software bugs in the implementation. In general, anything that would divert any attempts of hackers in turning a software bug into exploitation must be avoided. In-depth details on the security topic are discussed in Chapter-4.

3.4 P-Cloud Architecture for Delivering an Error Free System

The following section explains the concepts of the P-Cloud. The principle here is that the division of data boundaries is based on the system usage model. It will guarantee creation of an error-free computing environment with respect to data handling. Each data repository can be accessed via separate gateways (Figure 3.8). The gateways will connect to their partner entities in the G-Node which is usually a smartphone. In the case of the POCT, there are four gateways to get the access to appropriate databases in P-Cloud. These gateways have machine to machine (M2M) interfaces and user interfaces. The smartphone application developers will create applications which will provide access to the gateway entities in the P-Cloud.

The P-Cloud consists of mainly five classes of database repositories. They are noted as configuration (provisioning) data, measurement data, data mining data, execution data and deployment data. Each of these classes of data are accessible via four different logical gateways: provisioning data gateway, measurement data gateway, operational data gateway and data mining data gateway. The measurement data is the POCT data sent from the P-Node via the POCT measurement data GW (gateway). The communication links between the P-Node and the GW is normally categorized as M2M type. Data mining database will be used by the clinicians for diagnosing. This database is an aggregated version of the measurement database. The data mining database will be accessible to data mining application developers (data reporting tools) to provide suitable data portals for the clinicians. Access to the data mining database



Cloud Base Framework For Handling POC Data

FIGURE 3.8: P-Cloud data organization

is provided by the separate logical gateway. The measurement data and data mining data segments are linked internally (can be done by creating database table views or data warehouse).

Execution database will collect operational data. One of the key needs for this database is to track the assay processes per device.

The execution database will help to debug field issues and it will help to log critical execution paths in the POCT device. The concept here is that the logged data will help to identify any computational deviation for a particular POCT process. It is essential that all the processes and their execution sequences in the device (which is a mission critical device) should be documented and assigned labels. In the POCT, a particular process or a set of processes will provide measurement data that pertain to a test. There will be a set of documented execution paths in the design that will accomplish test results. If the documented execution paths differ during a POCT, the test will not be valid. The execution path data which will be logged in the database will reveal any execution deviations against the expected execution for a given test. This process will not only help to remove any errors in software coding during system acceptance testing (before launching the system), but will also be an essential tool for the field engineering team. By analyzing the execution data logs, the field engineering team will be able to pinpoint the root cause of any issues during actual operation of the system. The execution log database will be accessible to the field team and the POCT devices in the system.

Configuration database is a place for storing POCT device provisioning data. This will be accessible to the device (via M2M link) and reachable by various health care providers and lab associates. This database will have all the configured data of the devices. The deployment database will have the information about the association of the device and the G-nodes that can access the device. It is possible that the association between the device and the G-Node can be one-to-one or one-to-many (one device and many kinds of G-Nodes such as smart-phones, PC and any user computing devices). The

configuration database will only have the configuration that pertain to the device and the deployment database will only have the associated data. The configuration and deployment databases share a common device provisioning gateway.

The principles here are to segregate the data boundaries for different databases and to avoid any logical errors and execution errors. The access privileges to each gateway provide adequate security boundaries for data privacy. These databases are independent of each other with the minimal association. The commercially available cloud services can be configured to provide a P-Cloud as discussed in the above sections. A summary of the P-Cloud structure and access is also shown in Table 2-3.

Database Class	Gateway Access				Users					Security Grouping
	Provisioning data GW	Measurement data GW	Operational data GW	Mining portal data GW	P-Node	Patient	Field / System Engineers	Provisioning Agencies	Physicians	
Measurement data		✓			✓					GP2
Data mining data				✓					✓	GP3
Configuration data	✓				✓			✓		GP1
Deployment data	✓				✓			✓		GP1
Execution data			✓		✓		✓			GP4

TABLE 3.3: P-Cloud and Access Gateways

Security groups are used to control the access of the users. As shown in the Table, P-Node is a member of security group GP2. It has access to Measurement data via the Measurement data GW. Physicians (member of security group GP3) have access to Data mining data via Mining portal GW. P-Node and the provisioning agencies (both are part of security group GP1) have access to Configuration data and Deployment data via Provisioning GW. P-Node and the Field /

system engineers (both the users are part of the security group GP4) have access to Execution data via Operational GW.

Thus, the security groups play a major role in providing the access to the data in P-Cloud. The security grouping provides segregation of data segments for maintaining data boundaries. An in-depth discussion on the data security is discussed in Chapter-4.

3.5 Benefits of adopting IEC 62304 at early stages of development

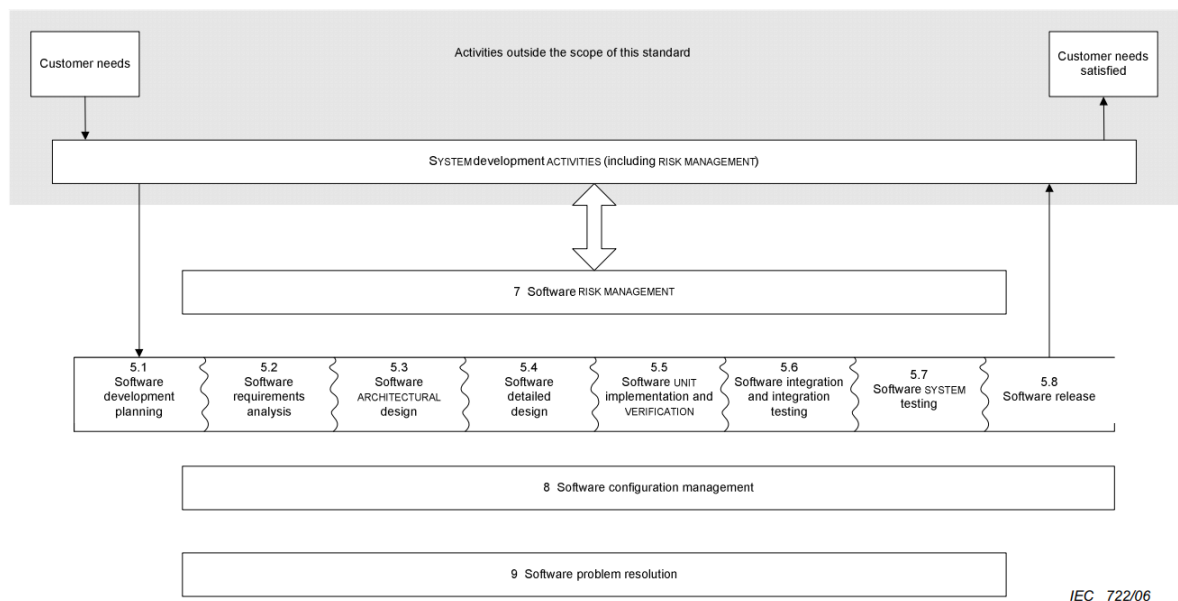


FIGURE 3.9: IEC 62304 process [Standard: Introduction]

POCT system communication software is the key part of the system. The software must fulfil the requirements for realizing control and communication functionalities with acceptable risks. IEC62304 is a framework for developing and maintaining mission critical software for medical devices. It was initially released in 2006, and it specifies the life cycle requirements for the software based on the safety criticality of system functionalities. It defines the life cycle requirements

for software. Currently, it is a harmonized EU standard, which is approved by the FDA as a recognized standard for safety-critical system development. It uses ISO 14971 to do risk analysis. The development process described by the standard is shown in Figure 3.10 [IEC 62304 standard: introduction].

Sections 5.1 - 5.8 describe a general guideline for the software system development activities. The Standard can be referred to for specific items.

There are three classes (Class A, B and C) of the software safety [IEC62304: section 4.3] based on the level of risks or hazard presented to the patient and users. This is based on the level of severity concerning the injury or damage to the individual. The Standard defines the term SERIOUS INJURY (wrong diagnosis by the device due to malfunction of communication links) as a life threatening situation, which will result in permanent damage to the bodily functions, and needs a medical intervention to prevent any impairment of the body. If there is no injury or damage to health, the system software is classified as Class A. If there is a possibility of NON-SERIOUS INJURY, then it is classified as Class B. In the case of death or SERIOUS INJURY, then it is classified as Class C. Class C is the highest risk and serious situation as defined by the Standard. Rigorous development processes need to be followed as defined by the Standard [IEC62304: chapter 5.0].

In Figure 3.10 the subsystems of the POCT device are identified using the safety classifications based on the definitions provided in the Standard. The core module is classified as class C because the POCT process is done with various constraints and controls. Table

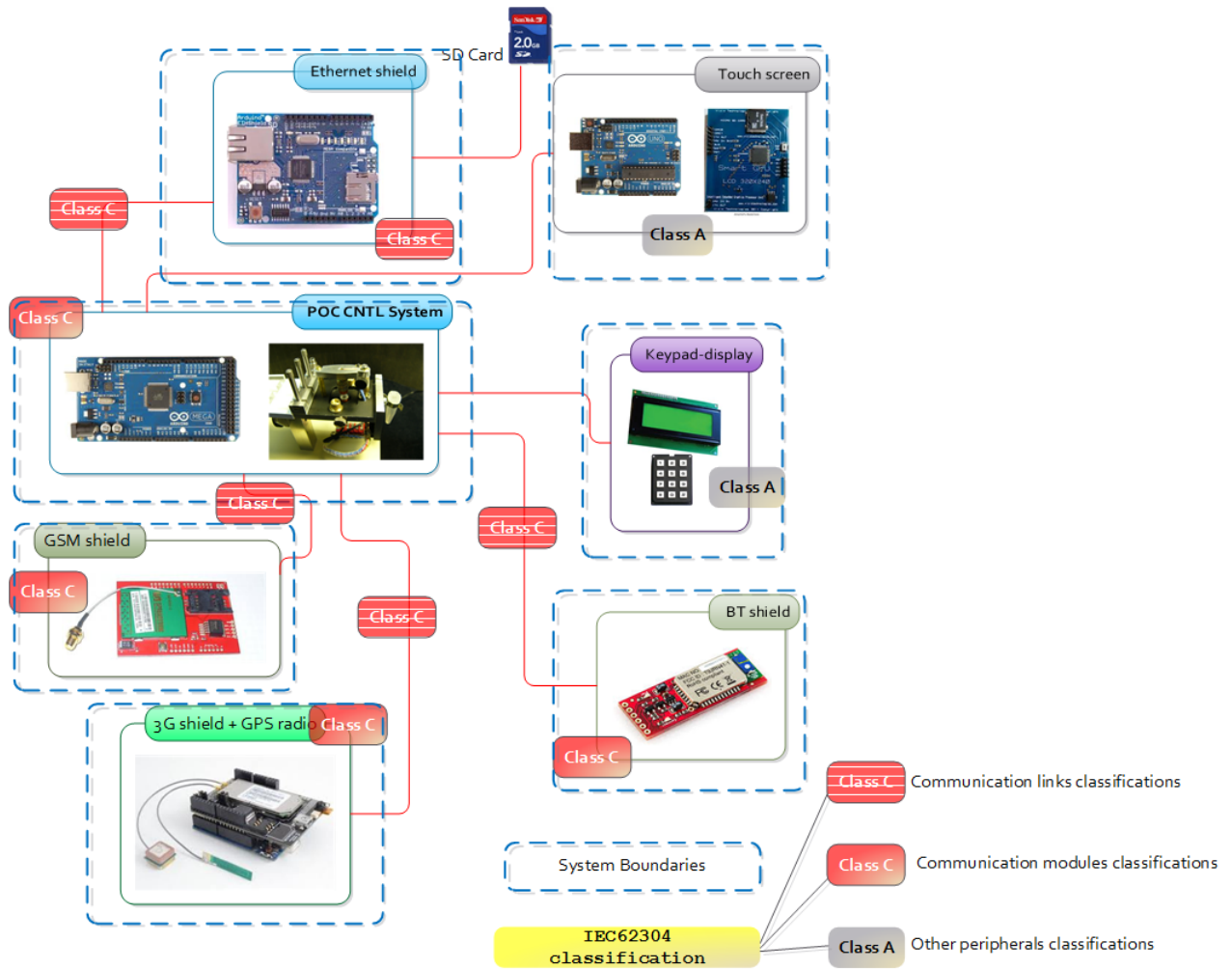


FIGURE 3.10: Architecture Partitioning and IEC62304 Classification

A.1 [ISO 62304: Annex A.2] shows the development process requirements by the safety classification. Class C has additional process requirements compared to Class A and B. Any failure in the steps will lead to wrong generated data and could lead to a false diagnosis.

The architectural partitioning of the POCT device is developed, based on class definition of the Standard. The functional modules are segregated as Class A, B and C based on the health risks to the user of the system. The user interfaces: keypad display and touch screen are classified as Class A. The communication links and the associated modules: Ethernet, GSM, 3G and GPS radio are classified as Class C. The core of the system is classified as Class A. There

are two main approaches that can be used to segregate the system: hardware approach and software approach. The software approach is tied directly to operating system parameters. The important operating system parameters are as follows: task isolation, restricted memory access, CPU / time protection, exception handling and virtualization. If an embedded OS is used in the design, the first three parameters can be set easily. The exception handling needs to be done at the application code level. The virtualization helps to create a separate partition over the functionalities.

Securing the compliance with the IEC62304 development process has many benefits which include the development of error-free software. During the project cycle, a software development plan must be maintained, and it should reflect the changes, and it must provide a real picture of the activities. All the team members must be aware of the plan, and they must follow the plan. Software risk assessment must be done based on the ISO 14971 and risk should be mitigated. The risk management activities should be given importance and must be performed during the design phase.

Hazard identification is developed at the early stage of the development cycle by looking for opportunities to isolate and contain critical elements of the system to reduce the number of critically classed components. The effective system design (error-free system) and software partitioning are driven by two main goals: building a safe and effective system and minimizing development complexity and cost. Thus, the architecture segregation is key to risk isolation. The software architecture breaks the medical device software down into smaller modules. These modules have a defined function and can be classified based on their functional role in the system. The

system integration test must be conducted at the early stages of the project to mitigate any interface issues.

The Standard specifies the degree of the software development process rigour based on the established classification. It places additional focused attention on the use of third-party software, known as Software with an Unknown Pedigree (SOUP). The device manufacturer must take full responsibility for the entire software stack (including SOUP) [IEC62304: sections: 5.1.1, 5.1.5, 5.1.7, 5.2.2, 5.3.3-3.4, 5.3.6, 6.1, 7.1.2, 7.1.3, 7.4.1-4.2 8.1.1-1.2]. All the known bugs in the SOUP must be enumerated, and the intended use and conditions for SOUP must be stated. The risks need to be identified by analyzing the failure modes and mitigated for the SOUP portion of the software stack. The configuration management for the SOUP must be implemented. The importance of evaluating SOUP modules is mentioned in the Standard very strongly.

With these recommendations, delivery of the error-free software for the medical device and its maintenance after the first product release can be managed.

3.6 Using Combinatorial Design Methodology for Generating Smart Testing Vectors

The combinatorial design methodology will help to identify the basic and essential test vectors that are required for verifying the system [69–74]. There will be many possibilities that can be validated. However, it is not possible to test all the possibilities within a given short cycle of testing available. Combinatorial testing is an adaptable methodology which is useful in a wide range of situations to uncover

Manual Process	Combinatorial Design Process
Use more scenarios than needed	Right number of scenarios
Forget an important combination	Additional scenarios
Repeating the same scenario	Specialized cases
	Unexpected combinations
	Notable combinations
	Rare combinations that may occur

TABLE 3.4: Benefits of Combinatorial Testing Process

software defects. It is based on the statistical process that while the behaviour of a software system may be affected by a large number of factors, only a few factors are dominant in inducing software failures [75].

There are two main reasons for the large number of software defects in the design and implementation of Safety critical systems: a system with complex requirements and absence of software verification tools. Exhaustive system testing which covers all the use case scenarios is hard within the allocated testing time. Manually identifying the highly probable complex combination of inputs of the system under test is impossible because of the combinatorial explosion in the great number of states that an SCS can reach when it executes [76].

Table 3.4 shows a comparison between manual and combinatorial testing process. The combinatorial design-based approach offers a way of testing the system with fewer key inputs. This process will build structured variation into testing scenarios. The structured variation focuses on creating almost all the key subsystem behaviours. Most of the issues are triggered by one or two entities in the system, and this process will engage all possible two-way interactions

(or combinations).

3.6.1 Application of combinatorial testing process in POCT system

The combinatorial testing methodology was used to identify the critical test cases for validating the communication system. Having the minimal number of critical test cases will provide early detection of communication failures.

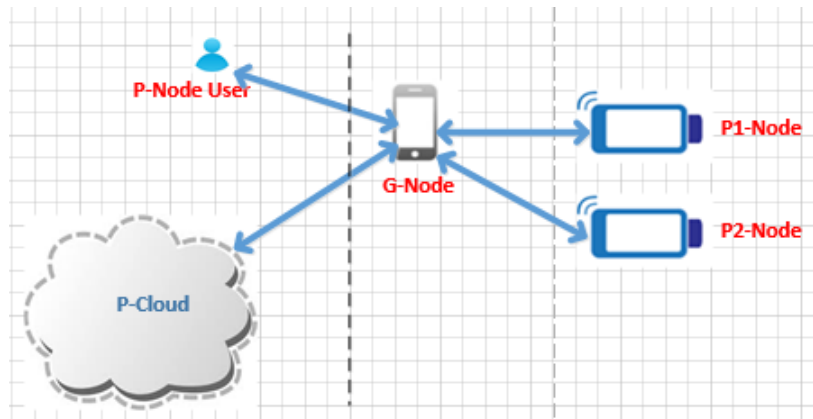


FIGURE 3.11: System under test

There are numerous open source tools available for combinatorial testing. Pairwiser is a combinatorial design tool developed by Inductive, which is based in Norway [1]. A system configuration shown in Figure 3.11 was considered for demonstrating test case generation. There are two P-Nodes, a G-Node and a P-Cloud are configured for POCT operation. The test generation focuses on the communication links between the nodes.

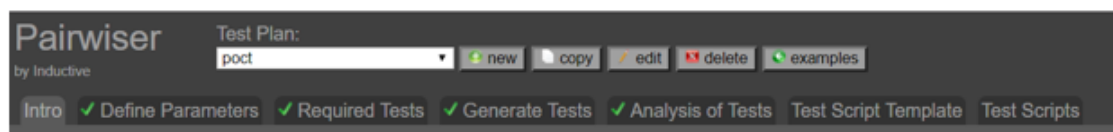


FIGURE 3.12: UI (User interface) of Pairwise tool [4]

Figure 3.12 shows the UI (user interface) of the tool which has multiple sections: Define Parameters, Required Tests, Generate Tests and

Analysis Tests. The Test Script Template and Test Scripts are not required for the current discussions. They are meant for developing test scripts depending on the testing environment.

The screenshot shows a software interface for defining parameters and constraints. At the top, there is a toolbar with buttons for "Define Parameters", "save", "parameter", "constraint", and "Import from CSV". Below the toolbar, the interface is divided into two main sections: "Parameters:" and "Constraints:".

Parameters:

1. **P1-Node_G-Node_Link** (parameter icon) with values: WLAN, BT, USB, ZigBee.
2. **G-Node_P-Cloud_Link** (parameter icon) with values: 2G, 3G, 4G, 5G.
3. **P1-Node_P-Cloud_Link** (parameter icon) with values: 2G, 3G, 4G, 5G.
4. **P2-Node_G-Node_Link** (parameter icon) with values: WALN, BT.

A "parameter" button is located below the parameter list.

Constraints:

1. **if** (dropdown) with two conditions:
 - Condition 1: **P1-Node_G-Node_Lir** (dropdown) is (dropdown) **WLAN** (dropdown) then (dropdown) (parameter icon) (close icon).
 - Condition 2: **P2-Node_G-Node_Lir** (dropdown) is not (dropdown) **WALN** (dropdown) (parameter icon) (close icon).

A "constraint" button is located below the constraint list.

FIGURE 3.13: Input data model [4]

Figure 3.13 shows possible (configured) radio technologies for the communication links under consideration as shown in Figure 3.13. This is also known as input data model of the system under test. It defines all the configurable radio technologies possible in each link. The constraint segment defines that when the WLAN is used by P1-Node, P2-Node should not use WLAN in order to avoid coexistence issues.

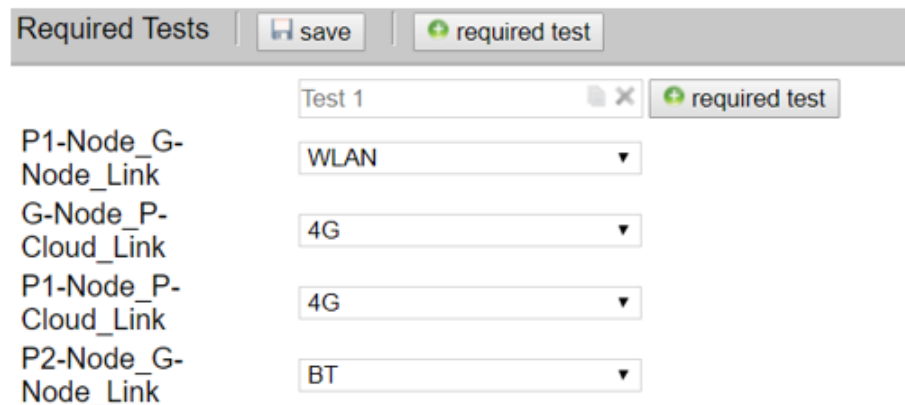
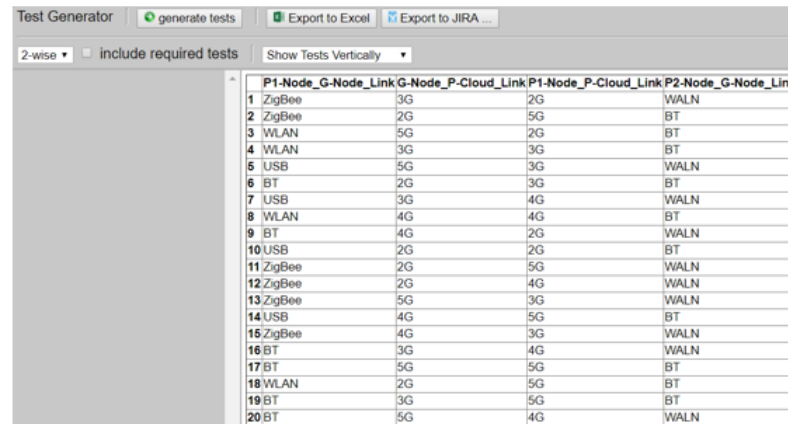


FIGURE 3.14: Seeding Process [4]

Figure 3.14 shows the seeding process (seeding of required input data combinations) of the system under test. This process helps the testing team to assert any known input combinations that need to be verified. The net result is that Test 1(WLAN, 4G, 4G, and BT) will be added to auto-generated test vectors in the next step. The identified input combinations are then used with the generated test vectors as must test cases.

The tests case (or test vector) generation step (Figure 3.15) lists the test vectors. This is generated by the tool using the input data model provided. Note that an option is available to change combinatorial coverage dimension to 3, 4 or mixed mode. The figure shows the 2-wise coverage dimension. The 2-wise coverage dimension is quite

adequate in identifying the test vectors that give at least 95 percent-
age functional coverage.



	P1-Node_G-Node_Link	G-Node_P-Cloud_Link	P1-Node_P-Cloud_Link	P2-Node_G-Node_Link
1	ZigBee	3G	2G	WALN
2	ZigBee	2G	5G	BT
3	WLAN	5G	2G	BT
4	WLAN	3G	3G	BT
5	USB	5G	3G	WALN
6	BT	2G	3G	BT
7	USB	3G	4G	WALN
8	WLAN	4G	4G	BT
9	BT	4G	2G	WALN
10	USB	2G	2G	BT
11	ZigBee	2G	5G	WALN
12	ZigBee	2G	4G	WALN
13	ZigBee	5G	3G	WALN
14	USB	4G	5G	BT
15	ZigBee	4G	3G	WALN
16	BT	3G	4G	WALN
17	BT	5G	5G	BT
18	WLAN	2G	5G	BT
19	BT	3G	5G	BT
20	BT	5G	4G	WALN

FIGURE 3.15: Test Vector Generation [4]

The test analysis (Figure 3.16) predicts cumulative functional cov-

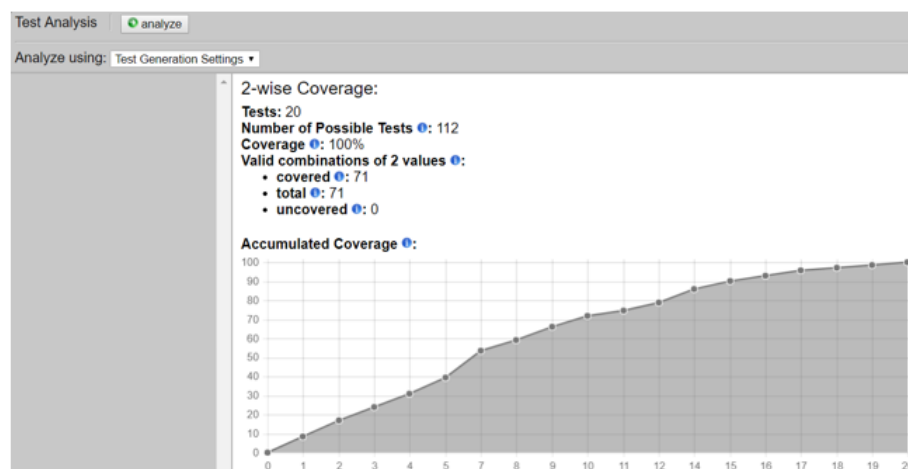


FIGURE 3.16: Coverage Analysis [4]

erage of 100 percentage with twenty test cases. Note that there is
no need to run all the possibilities (120 test cases) to accomplish 100
percentage coverage. The number of possible tests grows rapidly
as more parameters and values are added to the parameter defini-
tion. For 2-wise test vector generation, use of the number of valid
and possible pairs of values, resulted in 71 test vectors. The valid
number of tests indicate that the possible combination of tests with
2-wise (pair-wise) selection exist among the input parameters. The

combinations pairs must be chosen from two different input parameters. It is invalid to select the values from the same parameter. On the whole, the process of test case generation using combinatorial methodology does safe test execution time and 100 percentage coverage is possible, which is important for POCT device system validation.

3.7 Agile Project Management Approach in Developing POCT systems

3.7.1 Project Management in IEC62304

The requirements are elicited from the stakeholders: bio-medical experts, mechanical engineers, user interface engineers, physicians and system engineers. The scope of the requirements were allowed to change during the research because of a new finding or new outcomes. In the POCT system development, Agile Project Management [77–79] allows for flexibility of delivering interim software. The system was developed using stackable open source hardware modules. Effective integration of the modules is possible because of the multiple incremental phases of development by adding communication modules.

3.8 Summary

In this chapter, a system design methodology is shown which describes the benefit of creating the basic system building blocks for the end to end system. This process highlights the importance of having a clear architectural view of the system. There are some

key strategies such as isolation of system modules; need for creating a loosely coupled system and its benefits; actuation and actuation confirmation channel and an algorithm for implementation are presented. The goal of these approaches is to develop an error-free POCT communication software system that can be used in medical diagnosis environments as an independent system. These strategies help to produce a product with highly predictable behaviour.

The role of security in safety critical system (in the context of the POCT) is presented. Some of the operational scenarios pertaining to the operation of the POCT device may cause security risks and the process of mitigating those risks is presented. The security aspects will be discussed in chapter 4.

A cloud-based architecture is presented to manage five types of data that is relevant to the POCT system. A gateway for data access concept is described. This will separate the implementation of the database design and the associated applications. The segregation will provide data access boundaries that will eliminate any software implementation errors; hence the system will be deemed dependable for the POCT operations.

The strategies and methods discussed in this chapter will guarantee the delivery of high-quality software for safety and mission critical systems, in particular the development of POCT device system communication software. The processes: requirement elicitation methodology, Agile Project Management method, combinatorial testing process and the development strategies presented in this chapter have many advantages over the traditional software development and system testing processes. The Combinatorial testing process will help

to contain any software defects before the product launch or maintenance upgrades. This can be done with fewer resources and relatively less testing time than the traditional testing processes.

The Value-Based Requirement eliciting process provides key-value attributes which will directly resolve real requirements needed for the POCT system. Due to the fact that it provides measurable attributes to the requirement, the implementation can be validated without any ambiguities. In other words, the behaviour of the system can be described with measurable requirement attributes. The importance of having measurable requirements attributes has been highlighted as a key factor to the development of highly predictable systems such as POCT. There is a need to link system architecture to the system requirements of the POCT system because of the importance of traceability in the architecture design. In the next chapter, a methodology for system architecture design, Use Case Maps with links to the requirements are presented.

Chapter 4

Security Framework for Managing Data Security within Point of Care Tests

4.1 Introduction

With the exponential rise in clinical devices such as POCT instruments [32], clinical network security has become a major issue for biomedical teams and healthcare organizations [80]. Because of the need for multiplexed detection of viral and bacterial infections without easy access to a lab, management of future outbreaks will become more involved with the compact portable point of care test devices [81]. Based on the State of the Internet report published by Akamai, Port 80 was the top targeted port by advisories in the US [82]. In addition, Port 443 [83] based attacks were seen primarily in Indonesia [82]. Port 80 is for the application based on HyperText Transfer Protocol (HTTP), and use of port 443 is for the applications based on HTTPs (Secure-HTTP). Targeting ports 80 and 443 implies that the advisories were targeting web-based applications (both HTTP and

HTTPs), which are very popular among smartphone users. Smartphones use medical applications based on the web and it is device-specific. Clinical instruments and devices such as POCT instruments that connect via the smartphone grew more than 60 percentage in 2012 [80]. As more devices are added, there is an increasing concern with respect to security of the data transmitted in a clinical setup.

The POCT systems can be classified into three broad categories as shown in Figure 1: testing, healthcare monitoring and alert, and interfacing with other existing health monitoring devices. All classes of devices need to have strategies and processes to deal with security concerns.

The current ways of providing security are specific to applications, e.g., the process for IoT related security proposals are available. The authentication process that is applicable to IoT is described in Srinivas, Mukhopadhyay, and Mishra [84]. A proposal for secure communication protocol specifically for the healthcare IoT also has been discussed in reference [85]. A general security framework to address IoT security issues has been outlined in Huang et al [86]. The need to authenticate the user, the system, methods, and processes are evolving [87]. A method for preventing energy depletion security attacks with ZigBee is explained in Cao et al [88]. A review of security challenges and existing architectures in the fast-growing IoT system and it has been documented [89]. The important need for having a holistic security framework is mentioned in [90]. Though cloud-based storage for medical data is convenient, accessing data from the cloud has security issues, and the data must be accessed securely [91]. The IoT security can be accomplished in many ways, including using the HW and biometrics defence [92]. The challenges of developing m-Health securely for defending the privacy of the user

community is one of the key aspects of interconnected medical systems [93]. It is crucial to have security systems able to work with multiple interconnected systems that may use multiple communication technologies [94]. There is a need to enhance the security of communication links for any IoT type of system, in order to preserve the patients anonymity [95].

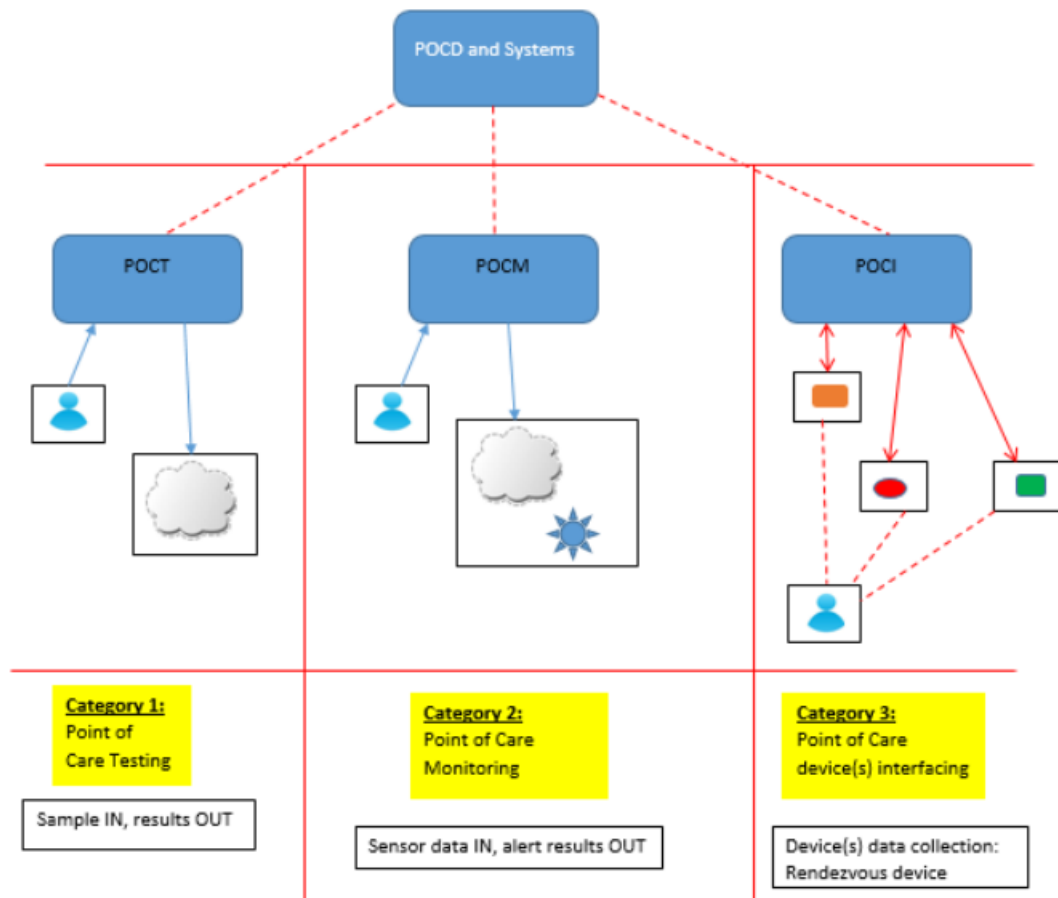


FIGURE 4.1: Device usage categories

Therefore, having a security framework independent of communication technologies and medical applications that is flexible enough to use universally is important. The security framework presented may be used within a hospital network scenario or any other healthcare clinical establishment where POCT is carried out remotely. This chapter provides a security framework that can be used in the context of the POCT devices, system, and networks. The framework is

independent of SW and HW of the POCT device.

POCT devices and systems can be categorized into three broad classes (CAT 1, CAT 2, and CAT 3) based on the context of operation and usage as shown in Figure 4.1. The categories are defined to address certain usage models of the POCT device. The devices that are used for POC testing and diagnostic applications (POCT) are defined as CAT 1 devices; the devices that are used for patient monitoring are defined as CAT 2 devices (POCM); the devices that are used for interfacing with other devices are defined as CAT 3 devices (POCI). The POCI devices provide an interface gateway for collecting and aggregating data from other medical devices. In all categories, data security is an important aspect.

As mentioned above, Point of Care devices and systems can be categorized into three main classes, based on their operation and usage contexts. The use of the point-of-care (POC) for testing and diagnostic applications (POCT), category one; use of the POC for patient monitoring (POCM), category two; and use of the POC as an interface gateway (POCI) for collecting and aggregating data. In all categories, data security is an important aspect. This chapter presents a security framework concept that is applicable to any of the categories of POCT operation. It outlines the concepts and framework for preventing security challenges in unauthorized access of data, unintended data flow, and data tampering during communication, both between system entities and between the user and the POCT system. This framework includes secure layering of the basic POCT system architecture, protection of POCT device in the context of application, and network and attack tree (threat model) for the system. A proposal for a low-level security protocol is discussed. This protocol is independent of communication technologies, and it

is elaborated in relation to providing security. A mechanism that can be used to overcome the threat challenges has been shown using the elements in the protocol. The chapter further discusses a vulnerability scanning process for the systems interconnected network. Finally, the chapter provides a summary highlighting the M2M security point of view and discusses the main players in a real deployment of the POCT system and its security challenges.

4.2 POCT System Configurations

4.2.1 Architecture Layering

There are three high-level layers, as shown Figure 4.2. The P-Node represents a POCT device; the G-Node represents a smartphone or another device used for accessing the POCT device, and the P-Cloud represents the data collection endpoint. The three layers outlined here form the basis for security partitioning.

Consider the following POCT system configurations or operational scenarios: The P-Node communicates with the P-cloud without any intermittent gateway entities such as smartphones.

Integration of the P-Node and the G-Node as one single unit; i.e. the G-Node, which is a smartphone, has all the required sensors and control built into, conduct the POCT processes.

In the first case, the POCT device and associated infrastructure combined provide security functionality. In the 2nd case, the G-Node together with the associated infrastructure assures security. In both instances, a layered model requires a partition strategy that protects all components in the network (POCT devices, network equipment,

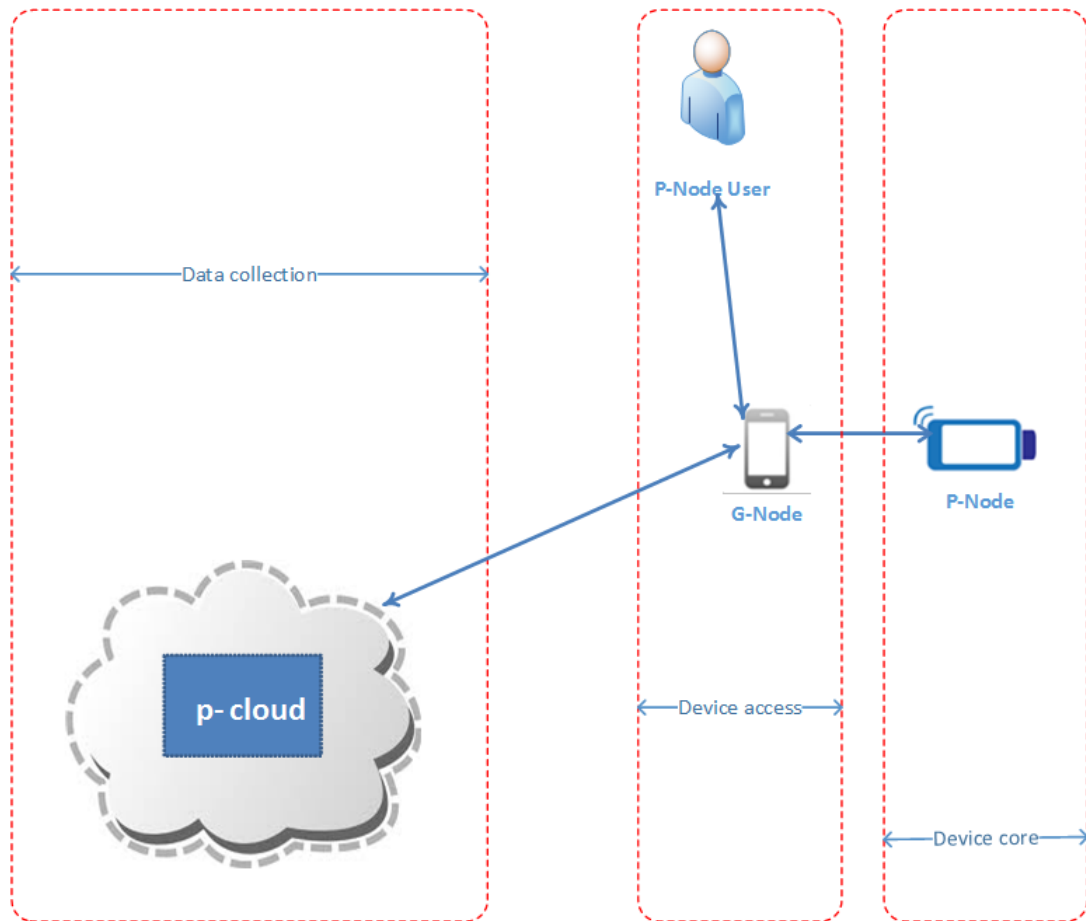


FIGURE 4.2: High-Level architecture layers

secure gateway and secure clouds). This layering approach provides physical security for the end- to -end system. The layering approach architecture needs to be developed dependent upon the POCT system deployment.

4.2.2 Security Layering

A balanced approach to ensuring security should be in place for the system. The POCT system is inter-connected, and all the elements in the configured network must have individual security boundaries (or layers) as shown in Figure 4.3. All secrets stored on the device are unique to that device. This will prevent any compromises to the

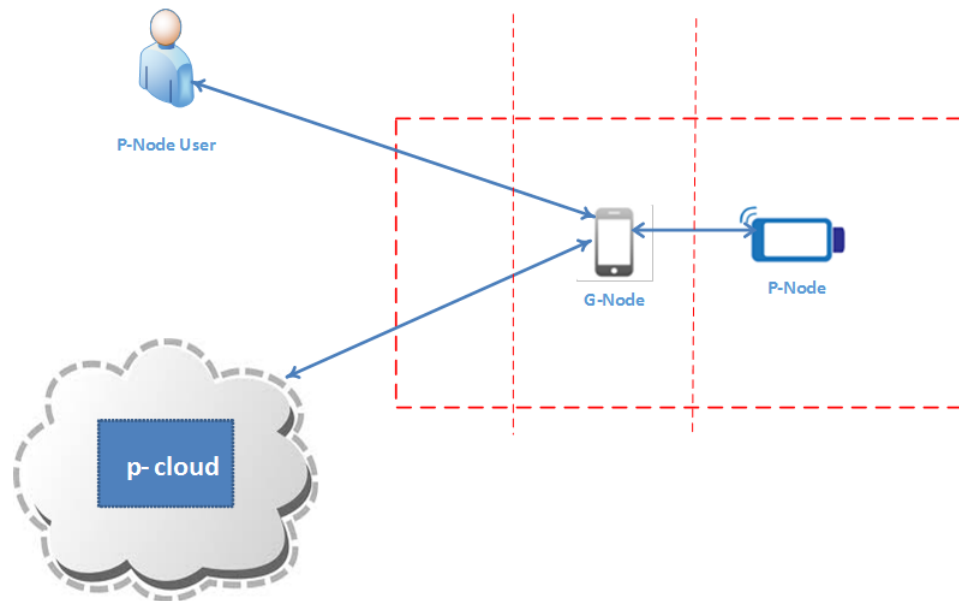


FIGURE 4.3: Secure layering of the basic architecture

whole system occurring in one device or a sub-system within the device. Development effort and cost should not be factors in deciding security protection strategies. A less costly device and an expensive device must be treated equally with respect to security protection. The designers and architects provide the layering security architecture for the POCT device and system infrastructure. The system implementers of the POCT device and the service providers must ensure that updated security technologies and security products are used to secure data on the POCT device and its associated network infrastructure. A management layer for administrative functions and organizational policies ensures that the patient uses the POCT system deployment securely. Service operators involved in the system deployment use the management layer to configure security and privacy policies on the system infrastructure. As shown in Figure 4.3, all the partitioned layers of security coexist to provide secure data transfer between key entities (P-Node, G-Node, and P-Cloud).

4.2.3 Definition of Asset in POCT system

Within the system domain, an asset is the value of data collected from patients during POCT. A threat model [96, 97] aids understanding of any potential threat scenarios and threat agents who are deemed likely to carry out a threat. The model shows the threat paths for the POCT system.

4.2.4 Attack tree for POCT System

The security threat model needs to be reviewed and evaluated on a continual basis, as good security countermeasures applicable yesterday may not be applicable tomorrow. An attack tree [98, 99] has been built for analyzing the POCT system security implications (Figure 4.4).

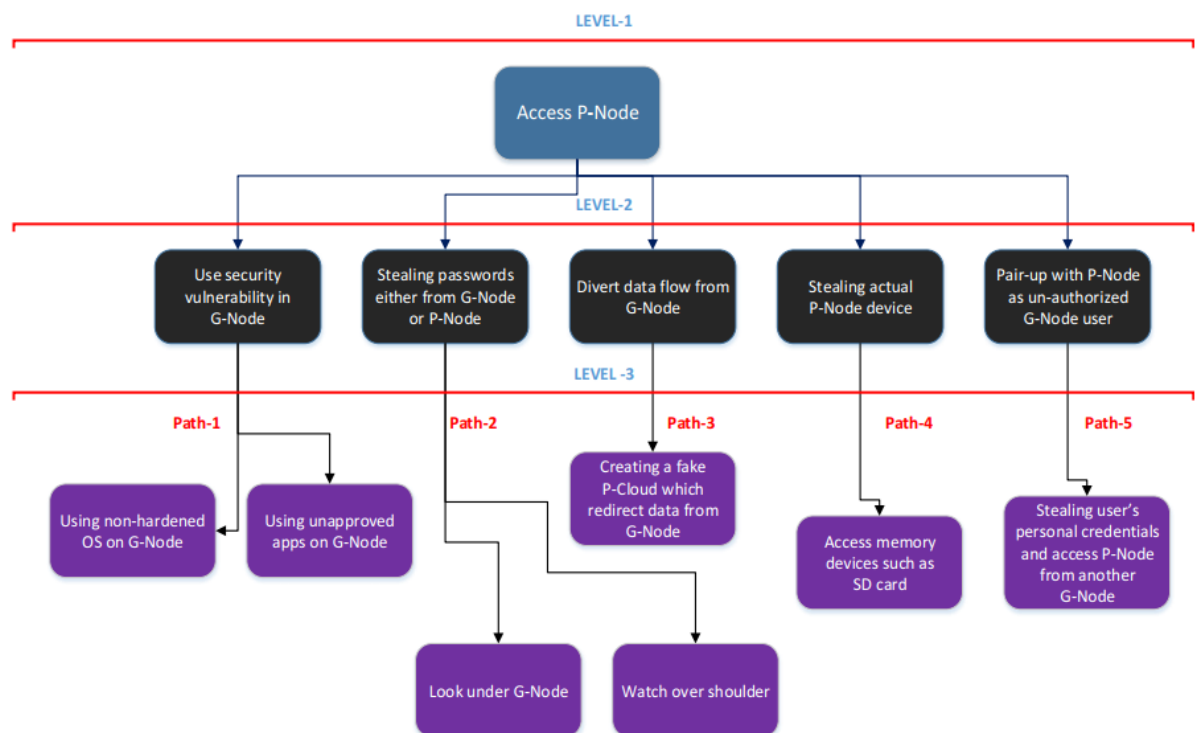


FIGURE 4.4: Threat path tree (Attack tree)

The threat path tree is divided into three levels. Level-1 represents accessing P-Node. Level-2 and 3 are illegal paths to gain access to the P-Node. Path-1 shows two ways: users G-Node is loaded with the non-hardened operating system and use of an unapproved application on G-Node, either of which can compromise security on G-Node, which creates an unauthorized access to P-Node. Path-2 shows two ways: accessing G-Node by stealing the password and watching over the authorized user when he or she logs on to G-Node, which will then provide an illegal entry to P-Node. Path-3 shows that an intruder can redirect the data collected to a fake P-Cloud server for accessing user data. Path-4 shows the possibility that P-Node can be stolen by an unauthorized user for accessing the measurement data stored on the P-Node. Path-5 shows that an illegal G-Node can access the P-Node to steal users personal credential data. And there are more threat paths possible, so appropriate countermeasures need to be in place to mitigate the risks of unauthorized access. Section 4.4 shows the processes for creating such countermeasures for POCT system.

The Freeport scanner shows that the ports have been configured as filtered (using open source tool [100]: Network Mapper, also known as Nmap). The state is defined either as open and filtered or closed and unfiltered. Open means that an application on the target machine is listening for connections or packets on that port. Filtered means that a firewall, filter, or another network activity is blocking the port. Therefore, Nmap cannot determine whether the port is open or closed. The closed ports have no application listening to them (they are available to use), and an application can open them at any time [101]. Ports are identified as unfiltered when they are responsive to the Nmap probes. However, the Nmap cannot determine

whether they are open or closed [101].

The setup configuration used to experiment with the Nmap port scanning tool is shown in Figure 4.5.

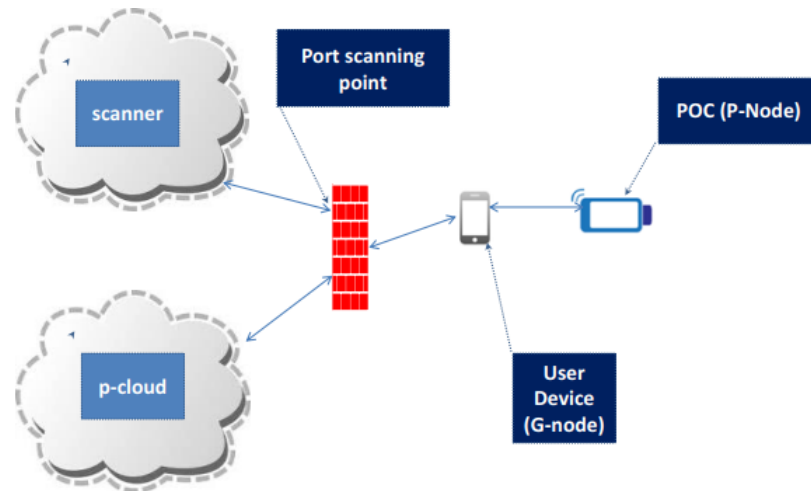


FIGURE 4.5: Port scanning setup

Figure 4.6 displays the configuration of the Nmap tool.

The screenshot shows the configuration interface for the TCP Port Scan with Nmap tool. The interface includes the following elements:

- Hostname / IP address:** 24.5.245.122 (ex. pentest-tools.com)
- Options:**
 - Ping host to check if it's alive
 - Detect operating system
 - Detect service version (slow)
 - Do Traceroute
- Port range:** 80 - 200 (e.g. 1 - 1024)
- Most common ports:**
 - 80 http
 - 443 https
 - 8080 http-proxy
 - 8000 http-alt
 - 8443 https-alt
 - 21 ftp
 - 22 ssh
 - 23 telnet
 - 3389 ms-term
 - 5900 vnc
 - 1723 pptp
 - 25 smtp
 - 110 pop3
 - 143 imap
 - 995 pop3s
 - 993 imaps
 - 465 smtps
 - 3306 mysql
 - 1433 mssql
 - 1720 h.323
 - 5060 sip
 - 179 bgp
- Buttons:** [Check all](#), [Uncheck all](#), **START**, **ReSTART**
- Calculation:** 32 + 5 = 37

FIGURE 4.6: Configuration of Port scanning tool

Figure 4.7 shows the output of a sample scan run

```
Starting query... [2014-05-01 07:51:06] Stay on this page for results!

Starting Nmap 6.00 ( http://nmap.org ) at 2014-05-01 10:51 EEST
Initiating Ping Scan at 10:51
Scanning 24.5.245.122 [4 ports]
Completed Ping Scan at 10:51, 0.23s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 10:51
Scanning c-24-5-245-122.hsd1.ca.comcast.net (24.5.245.122) [121 ports]
Completed SYN Stealth Scan at 10:51, 6.21s elapsed (121 total ports)
Nmap scan report for c-24-5-245-122.hsd1.ca.comcast.net (24.5.245.122)
Host is up (0.23s latency).
All 121 scanned ports on c-24-5-245-122.hsd1.ca.comcast.net (24.5.245.122) are filtered

Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 6.53 seconds
Raw packets sent: 246 (10.800KB) | Rcvd: 4 (148B)

Query finished [2014-05-01 07:51:13]
```

FIGURE 4.7: Scanned Results

4.3 Security Compromising Scenarios

In the POCT domain, an intruder targets applications run on the P-Node or G-Node that may range from web application attacks to client-side attacks and buffer overflow attacks [102]. Physical security of the on-board data must be safeguarded carefully, as the POCT device will contain clinical test data that belongs to the user. If a device is opened physically, it should not provide any more access to the data contained within the device than if it is logically accessible through the user interface or external ports without opening the device. Circuit level attacks on the device should be unfeasible, and the minimum bar to a hardware-based attack on the user data should be a silicon level attack.

The POCT devices can be misused for their access capabilities by attackers pretending to be the back-end application server. A scenario in which such an attack could be of some economic value to

the attack-er is the sale of POCT data or the use of the POCT device to gain control over other devices or systems. By pretending to be the POCT application server, attackers can penetrate other barriers and reach other business applications (such as mobile banking applications) on the user's devices. The POCT devices (G-Node and P-Node) must be protected from unauthorized entities trying to establish communications to and from them.

The data collected from the M2M-enabled POCT devices are sensitive in nature. For example, the data may contain information that can be used against the user by insurance organizations. Thus, the M2M POCT system security solution must be designed such that it is not possible to acquire information about the stored data by eavesdropping at any point within the network. The security framework in section 4.4 ensures prevention of any such attempts to access the network in an unauthorized way.

Identity information can be correlated with other data, such as the networks location data elements from which the identification data is retrieved, to discern some patterns. In the case of POCT system the identity of the end-customer not be available from a public database is critically important [41]. Therefore, the device should not transmit unencrypted data relating to the user identity [41]. Well-known, robust encryption mechanisms must be used, rather than reinventing new algorithms [103].

Low-cost health care devices such as heart rate monitors are required to send data collected by a server in a single data connection session. This design can be easily compromised by clever adversaries [41]. The proposed architecture model in Section 5 encourages multiple

communication sessions to be established with the main network entities before the data access is granted.

The POCT device is a non-mobile entity in the system. Moreover, physically accessing the device without authorization causes three main problems. Firstly, the data can be taken from the SD card. Secondly, the credentials from the UICC (Universal Integrated Circuit Card or SIM card) can be taken if the device has a cellular access interface. Thirdly, since there is no need for the hand-offs (device mobility is not required) to different base stations or radio access network, the intruder can easily identify the associated network and hence the device identity.

It is essential that appropriate SLA (service level agreements) between all stakeholders involved be in place prior to system deployment, and this must include security as one of the main items. The integrity of data stored in the POCT device after the testing process can be tampered. It is possible to masquerade as another POCT device and upload incorrect data to the P-Cloud.

4.3.1 Processing Data in POCT devices

Scenario: Where will the data process/generate results stage take place? The options are the POCT device or the server (P-Cloud).

The data process needs to take place on the server (the other end of the SSL tunnel) domain where the computing power can be more than the POCT device.

4.3.2 Access from multiple devices (Preventing Unauthorized Access)

Scenario: How can we ensure a one-to-one relationship/connection between a particular user and the testing device? The one-to-one relationship can be guaranteed using logic similar to that illustrated in Table 4.1.

<pre> IF (an active session is present) AND (the user is not owner) THEN (Reject any new session creation) ENDIF </pre>
<p>Note that this can be configured as to how many concurrent sessions can be established by the owner. This can be configured using the services provided by OMAP (Operation, Maintenance and Provisioning) Module.</p>

TABLE 4.1: Multiple Device Access

4.3.3 User authentication

4.3.3.1 User validation and User Identification

There are two methodologies that can be used for user validation and user identification.’ User validation is a process of comparing two data sets, while user identification is a process of using other data sets (databases) to identify the user. The choice of authentication method is based on the environment in which the device is used. This selection can be set in the OMAP module. The result of this would be to set security information data (in the communication protocol) sent between the POCT device and G-Node, to reflect this configuration.

4.3.3.2 Authentication using biometric

The biometric authentication process uses the user validation methodology. In this method, the user presents his or her credentials, and the device compares what is presented with the user profile information stored in the device.

4.3.3.3 Authentication using Active Directory

The active directory (AD) technology is an authentication mechanism developed by Microsoft.

When the user tries to log on to the device, a security challenge is communicated from the AD server to authenticate the user. The user can also log on to the device with the cached data (user profile info in the device) when there is no connection to the AD. For the AD to work, there needs to be packet data network connectivity via cellular network or WLAN to the AD server.

4.3.3.4 Clinical and Surveillance data collection

Scenario: Will the data (clinical and surveillance) be collected before the results of the test are presented to the user?

<pre> IF ((test completion bit set) AND (smart-phone has subscribed to the test completion event)) </pre>
<pre> THEN { 1- inform smart-phone about test completion 2 – provide smart-phone with a pointer to access the test data memory } </pre>
<pre> ENDIF </pre>

TABLE 4.2: Clinical and Surveillance

The data collection will happen when the test completion bit is set. This will be the trigger event for the smart-phone application to initiate the data collection. This means that the POCT device must have the capacity to store the collected data during the testing process, which will simplify the SW implementation. This is shown in Table 4.2 .

4.3.3.5 Safely dispose of the testing device

Scenario: Will the user be advised to safely dispose of the testing device once the test is complete?

An approach to accomplish this requirement is to have the smart-phone app (or the apps running on G-Node) register with the POCT device for the completion event of the testing. This will enable the POCT device to send the testing event complete type packet to the smart-phone. Then the smartphone app can indicate the test completion to the user by a visual or sound indication via UI (User Interface).

4.3.3.6 Mechanical method

If the user attempts to use the device cartridge again, a mechanism can be implemented to prevent this by mechanical means. Some of the options that can be used are using material properties of the cartridge and with paper microfluidic devices, it is only one-time usage.

4.3.3.7 On device usage prevention using SW

If the whole POCT device needs to be destroyed after one-time use, then this can be done in SW; a usage indication bit can be stored to indicate that the device has been used once. This needs to be stored in the ROM area (non-volatile memory). Whenever the user attempts to use the disposable device for the second time, the SW will prevent the user using the device.

4.3.3.8 More complex and fail-proof mechanism for preventing (or restricting) device usage

Another alternative is to compare the device ID for the history (in a database) before the device is used. For this, we need to have the network connectivity at the beginning of the test. The FDA provides guidelines on usages that are outlined for the device identification process [17]. This will also prevent any illegal resale of the POCT devices.

4.3.4 Web-based POCT System access

Web-based applications are one of the ways in which application developers create smartphone applications that will be used to control the P-Node. One of the mechanisms to secure web applications is to apply well-known security hardening patches for web servers and provide adequate network protection.

Figure 4.8 shows the application of the Web server that was running on the POCT device. The POCT device was situated behind a firewall and a router. This configuration is an example of a protected network behind a firewall.

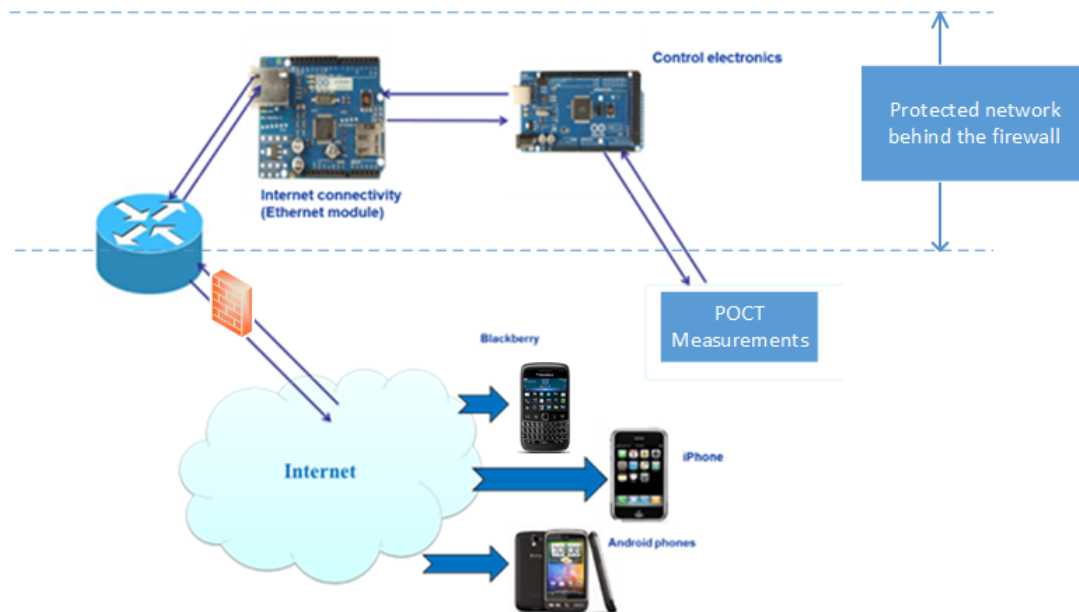


FIGURE 4.8: Embedded Web server with POCT

Web browsers are a readily available feature on any smartphone. An embedded web server was implemented in the POCT device, and a public Internet IP address was provided (via port forwarding at the router) for access to the web server. The smartphone was able to obtain the IP address to look at the POCT measurement data using the embedded web browser.

Common web application attacks such as cross-site scripting, SQL injection, XML injection, and command injection are also applicable to POCT systems. Any HTTP request from the smartphone may be subject to these common web attacks.

4.3.5 Cross-site scripting

In the cross-site scripting (XSS) attack in the context of POCT system access, malicious instructions can be sent to the smartphone browser. A standard browser cannot distinguish between valid code and a malicious script, and it accepts user input without validation. The main goal of the XSS is to steal information that is retained by

the browsers. Therefore, it is necessary to have an approved browser that is configured to run POCT web applications. The standard browsers available on the smartphones are not advisable for accessing medical applications. There are secure frameworks available for developing secure mobile embedded browsers [104]. There are many techniques developed to secure browsers for data transfer [105]. Applications such as MedCheck [106] are needed on the smartphones for POCT applications. For developing embedded secure web servers, an architecture similar to the Sizzle platform is required to secure POCT web servers [107].

4.3.6 SQL Injection

By entering an incorrectly formatted e-mail address, an attacker attempts to analyze whether the input is being validated. Then the attacker will use SQL statements to collect data from the database. This illegal, unauthorized access to data can be prevented if all the fields are validated in the HTTP website code. To ensure that the data field validation is implemented, a mandatory requirement must be created. This requirement will be implemented and tested before the deployment of the web application.

4.3.7 XML Injection

HTML instructs the browser to display text in a particular format. XML carries data instead of indicating how to display it using a pre-defined set of tags, mostly defined by the user application. If the website does not filter user data, it will be prone to XML tag injection. This process will modify data stored in the P-Cloud database.

By implementing the requirement for data filtering, the XML injection attacks can be prevented.

A compromised G-Node will lead attacks on the web server that resides within the POCT device. An attacker can gain access to the operating system on the POCT device via the infected G-Node. One of the HTTP header fields, the referrer field, indicates the site that generated the web page. Attackers can modify this field to hide that it came from another website (a website similar to POCT device web server), a modified web page hosted from an attackers computer. The accept-language field is another HTTP header; some web applications pass contents of this field directly to the database (P-Cloud). This field could be used to inject SQL commands to get patient data.

4.3.8 Client-side attacks

So far, the attacks related to the web applications have been discussed. Server-side attacks and client-side attacks also target vulnerabilities that exist in client applications. Examples of a client-side attack are that the client application interacts with a compromised server or the client initiates a connection to the server, which could result in an assault.

The security of the G-Node, i.e. the client computer, can be compromised simply by viewing a web page. Attackers can inject content into the vulnerable web server and gain access to servers operating system.

4.3.9 Malware attacks

Malware is software [108] that enters a computer system without the owners knowledge or consent. These are spread through computer viruses and worms [109]. Trojans, rootkits, logic bombs and backdoors are all forms of malware. Malware with a profit motive includes botnets, spyware, adware, and keystroke loggers.

Social engineering [110] is a means of gathering information for an attack from individuals. Types of social engineering approaches include phishing [111], impersonation [112], dumpster diving [113], and tailgating.

Malware can be downloaded to the G-Node without the knowledge of the user. Attackers develop a zero pixel frame to avoid visual detection and embed an HTML document inside the main document. When the browser used by the G-Node downloads a malicious script, it instructs the G-Node to download malware. Therefore, it is critical that the G-Node must be loaded with suitable anti-malware software to detect any malware downloads.

4.3.10 Cookie and Attachment Threats

Cookies store user-specific information on the G-Node. They are used to identify repeat visitors; e.g., travel websites would store users travel itinerary and personal information provided when visiting the site. Only the website that created the cookie can read it.

There are a number of types of cookies. Website users create a first-party cookie when they visit a web-site. Website advertisers use a third-party cookie to record user preferences. A number of cookies will also be used when a web-based POCT application is accessed

on the G-Node. A few scenarios that involve the use of cookies are outlined; a session cookie is stored in the RAM and expires when the browser closes. The G-Node records a persistent cookie on its drive, and the persistence cookie does not expire when the browser closes. A secure cookie is used when a browser visits the server over a secure connection, which is always encrypted. A flash cookie uses more memory than a traditional cookie, and it can-not be deleted through browser configuration settings. Given this wide range of cookie types, cookies pose security and privacy risks and, if stolen, can be used to impersonate a user and can, therefore, be exploited by attackers to steal data from the G-Node.

Session hijacking is a malicious process used by an attacker to impersonate a user by stealing or guessing the session token when the G-Node communicates with the web server. To prevent these kinds of attacks, the G-Node to P-Node and the G-Node to P-Cloud links must be encrypted using well-known encryption algorithms.

Buffer overflow is an anomaly in the software code where the buffer boundaries are not checked during data writing. An attacker uses any buffer overflow to steal data by attempting to store data in RAM beyond the boundaries of a fixed-length storage buffer, which cause data overflow into adjacent memory locations. This attack may cause the G-Node or the P-Node to stop functioning and open an unintended pathway in which the attacker can change the return address,” to redirect to an address containing malware code.

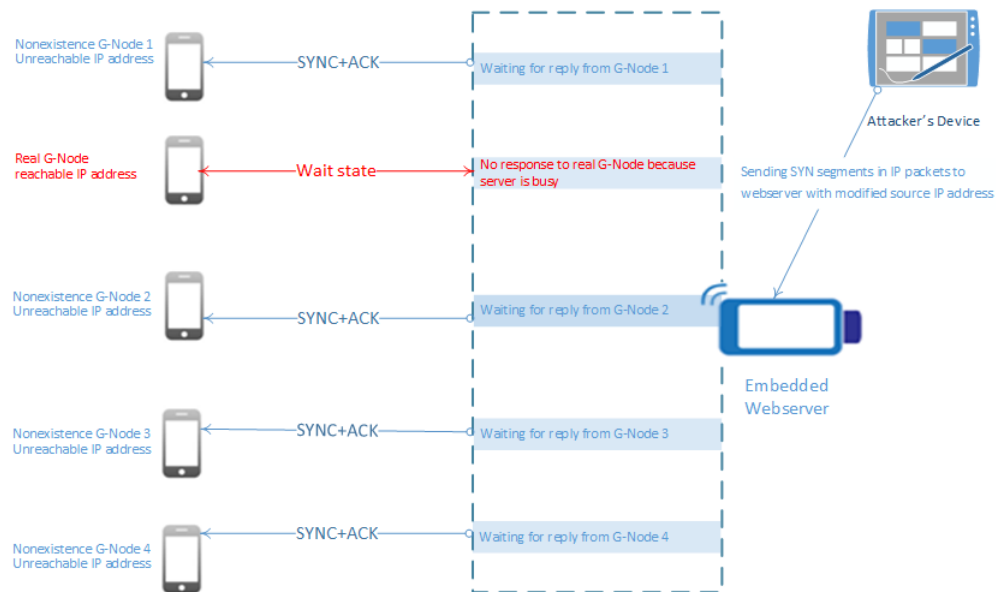


FIGURE 4.9: Distributed Denial of Service (DoS) Attacks

4.3.11 Denial of Service (DoS)

The DoS attempts to prevent the system from performing normal functions by ping flooding, therefore, sending a large number of echo request messages, which in turn overwhelms the web server. It is possible to send a ping request and alter the original IP address, thus mimicking the target G-Node; therefore, an attacker can acquire specific responses from all the devices connected to the network. Dangerous attack types include the SYNC flood attack and the DDoS (Distributed DoS [114]) attack (Figure 4.9). In the SYNC flood attack, the attacker takes advantage of procedures for establishing a connection.

In the DDoS, the attacker uses many zombie G-Nodes (G-Nodes connected to the Internet that has been compromised by a hacker) to flood a device with requests from non-existent IP addresses. The source of the attack is impossible to identify and, therefore, cannot be blocked.

In order to prevent any of the security attacks described above, the POCT system network requires a separate entity for security management [115, 116]. The separate entity is a network element (S-Node) for monitoring security. The S-Node is a specific kind of node that needs to be updated with all the latest security signatures of known vulnerabilities. DoS attacks can be prevented [117] by utilizing a static IP address (that is known to an administrative domain) plan for POCT system deployments. Even if an attacker imitates the DoS attack, this can be easily detected with the static IP configuration.

4.4 Security Framework

There are many approaches available for providing security (mainly for identity and data protection) for the POCT system environment. Methodologies and techniques already exist for providing information protection in the industry. A list of few notable methodologies are listed here; a biometric-based identification, user identification based on behavioural analytic process (including the Big Data approach), challenge and response mechanism and the multiattribute access process. The method is to establish a trust relationship with the POCT system network nodes, especially the P-Node, G-Node, and P-Cloud. The two, main approaches, a communication protocol and security mechanisms, are outlined in the following sections.

4.4.1 The protocol used between the G-Node and the P-Node

A communication protocol can be used to help accomplish data security. In a closed system, the protocol format can contribute to

providing security protection for the asset, which is the data collected during the testing. Any violators will not be able to determine the data unless they have access to the protocol format (refer Chapter-3).

4.4.2 Security Mechanism -1 (challenge and response based)

All device security measures originate from the device. Activation of any security agent, security sub-system, or security service must start from the device. It will then use the security services provided by other collaborating entities in the system. For example, data transport security is a function of the infra-structure. Whenever the data is sent to the P-Cloud from the G-Node, its encryption must be implemented using the security services provided by the infrastructure.

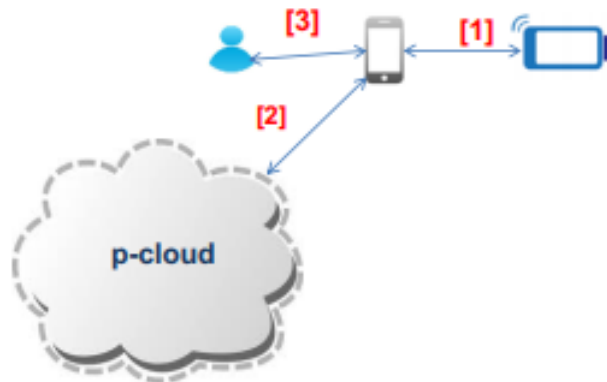


FIGURE 4.10: Interaction of messaging mechanism

As shown in Figure 4.10, it is assumed that the network connectivity between the three main nodes has been established. At this point, the G-Node attempts to start a POCT process using a native application or a web-based application. The OPCODE for START (refer Chapter 3) is used to begin the assay process by the user. A security challenge is sent to the G-Node from the POCT device to list the operational capabilities of the device. These interactions are shown

by [1] in Figure 4.10. The user device (which is the G-Node) will respond with a list of known operational codes that are provisioned by the operator. The POCT device selects a subset of OPCODEs from the received list and queries the G-Node for the last known list of OPCODEs. This particular information must come from the P-Cloud. If the G-Node has the authorization to retrieve information from the P-Cloud, then it can obtain the requested OPCODEs. This interaction is depicted by [2] in Figure 4.10. User interaction is shown as [3].

Once the history of the operational codes is retrieved from the P-Cloud, the G-Node informs the POCT with the OPCODEs list. If the data matches the records in the POCT device, then the user is allowed to continue the interaction with the POCT for testing. Figure 4.11 shows a summary of the interactions.

POC: [1] Tell me about my operational abilities
User device: [1] Responds with list of known OPCODEs
POC: [1] Will chose a subset of OPCODEs and ask when was the last time the OPECOE (s) was (were) used
User device: [2] Connects to p-cloud to get the last known time info for the requested OPCODE(s)
User device: [1] Informs POC with requested <u>OPCODE-usage time</u> record.
POC: [3] Will allow the operation from the user if the <u>record s match</u> the stored values

FIGURE 4.11: Interaction summary

4.4.3 Security Mechanism -2 (behavioural based)

In this method, the user is first authenticated with the G-Node by the standard methods such as the G-Node device password and network password in order to access services. **Step 1:** shows the process of user authentication with the gateway (G-Node) in Figure 4.12.

The next step (shown as **Step 2:** Application authorization) involves establishing a security relationship with the POCT application

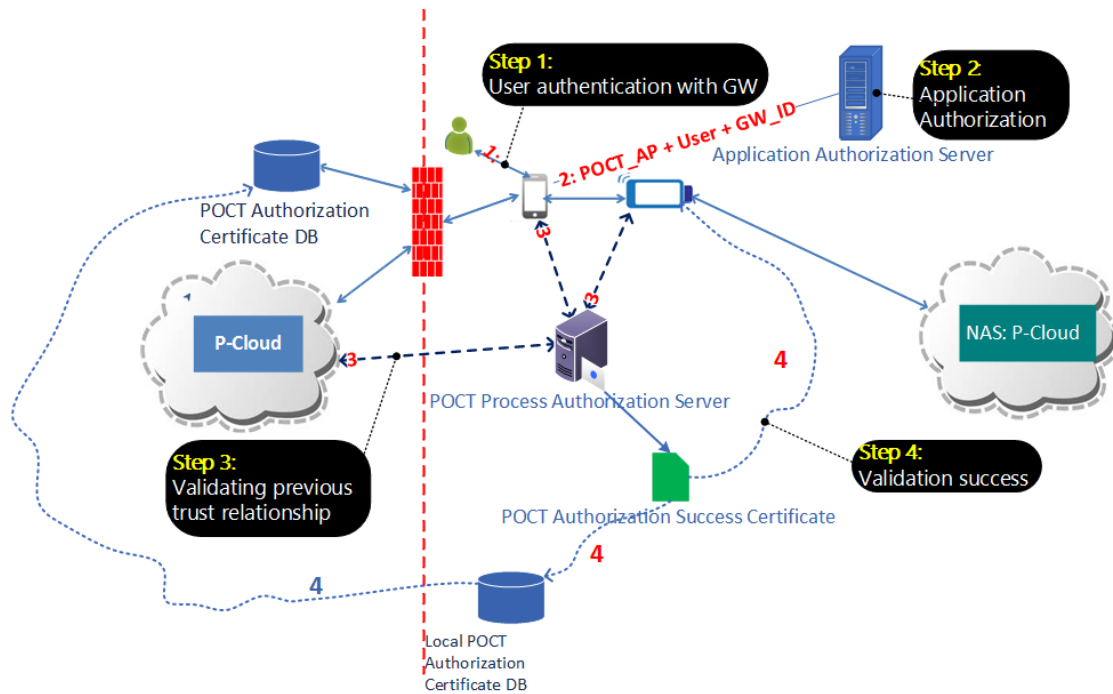


FIGURE 4.12: Security Mechanism: Behavioural-based

authorization server. The POCT application authorization server is responsible for providing approvals for the G-Node users to use a particular application or a set of POCT applications. The application authorization process is triggered when the user attempts to start a POCT application from the G-Node (via G-Node user interface). The authorization process requires few metadata attributes that are related to actual data. There are metadata attributes available that can be used for the authorization process (e.g. an identifier for the POCT application, a user identification parameter such as SIM card and MEID [118] or IMEI [119] of the G-Node). These data attributes must be validated by the POCT application server prior to the testing.

The purpose of **Step 3**: (validation of previous trust relationship) is an assertion and validation of the relationships that have existed between P-Cloud, G-Node, and POCT device, prior to the current test. This process is similar to the security mechanism in section

4.4.2. An entity called the POCT process authorization server is introduced, whose purpose is to manage who can execute certain assay types on the POCT. The server is responsible for creating a POCT authorization success certificate, which is an outcome of the record of the validation process for step 3. Step 3 can be further enhanced using Big Data techniques to identify the user. **Step 4:** is the final step for successful validation.

To validate the security mechanisms shown in Figure 4.12 an experiment setup is needed with a sample of users who can use the devices to initiate testing. Using any simulation tools will not help to validate the concept, because the interactions are based on user behaviour. The validation of the mechanism at best can be done as a system level test including some aspects of user acceptance testing (UAT). It is possible to derive many test scenarios due to the parameters involved in the design. The goal of the validation is to verify if all the steps 1,2,3 and 4, are adequate to prevent any data breach and security compromises. A poorly designed user interface can cause the failure on the G-Node, or it can be a real rejection from the application authorization server. The recommended relationship between G-Node and P-Node is 1:1, i.e. only one POCT device able to connect to a specific G-Node. The application authorization server has a database that contains a mapping of users and authorized applications. The validation at this level is to test how to avoid wrong entries the application authorization database. If the user gains access because of the bad data entries, there must be another defence to prevent the user from executing the test. The steps described in the architecture can be done with user involvement or without the user. In the case of automatic validation, the user may not be in a position to influence the validation results, but

privacy considerations need to agree among all parties involved. All the possible communication links must be included in the validation processes. The recommended and efficient way to validate the security mechanism is to apply the combinatorial methodology [75] to come up with test scenarios which will guarantee as high as 90 percentage of functional coverage, because of the large number of scenarios.

4.5 M2M (Machine-to-Machine) Security in POCT System

Deployment and operation of POCT systems are encompassed by cellular or short-range communication M2M [120] technology. The cellular M2M system differs from current cellular networks in three important ways.

First, the cellular network services today are typically offered by a single service provider who owns the distribution of devices with the SIM card distribution, device provisioning, network infrastructure, and service delivery for voice and data services. In contrast, with the cellular M2M, multiple operators and network vendors offer services. These players have limited business relationships among them.

For example, the entities involved in providing M2M solutions for power metering are the utility company that provides application, the cellular access network provider, the meter manufacturer, and the end-user. These entities do not necessarily trust each other. Hence, providing a security implementation in this environment is very challenging.

Secondly, M2M communication does not have intensive data transmission that could lead to lower financial earnings for the players involved. Fewer economic outcomes lead to less interest in implementing a security related layer that is not cost effective.

Thirdly, unlike cellular phones, smartphones, or wireless-enabled laptops, M2M devices are often unattended and are subjected to a higher risk of malicious mischief and misuse. However, in the case of POCT, there is a user who has ownership of the POCT device.

These dynamics among the main stakeholders suggest that the current security mechanisms in place today for mobile devices in the cellular networks are not appropriate for M2M applications.

The device manufacturer and the M2M service providers may not have a business relationship or predefined mutual trust. The users may buy devices on the open market. In the case of POCT market, there are authorized drug stores or authorized medical device dealers who are involved in the sales and marketing of the devices and services. In medical applications such as the POCT, the user may not even be aware of the existence of a virtual network operator who is providing the service on behalf of the owner of an organization that owns a POCT application. However, securing usage of the application is critical to all stakeholders.

4.5.1 Trust Relationships between POCT System entities

The following Table 4.3 shows the business relationships involved in M2M service delivery for a POCT system. This scenario is applicable for a POCT device at home as well as in the hospital. The health-care or medical institution acquires POCT devices from a vendor

company and distributes these to the end-users (end-consumers, the patients) who have subscribed to the service. The healthcare institution owns and deploys the POCT devices in the homes or hospitals of the end-consumers. The medical institution subscribes to the M2M service from an M2M service provider for POCT data collection and device management. The M2M service providers are usually the telecom carriers (operators, e.g. ATT [121]). The M2M service provider, in turn, has business relationships with network providers (e.g. Ericsson [122]) for the use of bandwidth for transmission of data. The table below shows the various entities and the relationships between them. A similar table is used to describe the metering service [41].

	What role does this entity play	POCT Device Users	POCT Device	Network Providers	M2M operators	POCT Application Providers (Health Care service providers)
POCT Device Users	Subscribe to health care service providers	-				
POCT Device	POCT capable of wireless connectivity	POCT device is configured with user's smartphone				
Network Providers	Provides wireless communication (cellular and connectivity)	No direct relationship	Certifies wireless modules in the POCT device			
M2M operators	Home network for POCT devices and interface with multiple transport network providers	No direct relationship	No direct relationship	Roaming?		
POCT Application Provider	Health Care service providers	POCT device user subscribes to POCT service	POCT Device ownership and deployment	No direct relationship	POCT Application Provider gets wireless services from M2M operator	
POCT device vendor	Makes POCT device with wireless module	No direct relationship	Makes POCT device with wireless module	No direct relationship	No direct relationship	Delivers POCT device with wireless module

TABLE 4.3: Main stakeholders and security relationships

Given the above summary, note that the complexity of the POCT M2M ecosystem is strongly characterized by diverse business and trust relationships, which cannot be accurately predicted during the design of security solutions. For this reason, the security protocol

design for M2M systems has to assume inherently that the M2M service provider may not have trust relationships with other stakeholders in the ecosystem. Therefore, a collection of suitable security strategies for the case of M2M is required along with design recommendations for appropriate security solutions in the context of the POCT system design.

4.5.2 Core security requirements for POCT M2M

Based on the scenarios discussed above, a list of core security requirements can be formulated [123].

- Authentication: Mutual authentication procedures need to be carried out by the POCT device and the POCT system operator network before initiating the testing and the data transfer.
- Confidentiality: Unauthorized data eavesdropping must be prevented between the application server and the POCT device.
- Data Integrity: Unauthorized data manipulations or modifications must be prevented between all entities in the POCT system.
- Exclusive Access: The POCT device must use only authenticated POCT applications that are available from the apps providers apps marketplace, and the network operator should prevent any other use.
- Identity: The identity of the POCT device or the user must not be revealed to any intruders in the event of security compromises.

4.5.3 Bootstrapping requirements for POCT device deployments

The bootstrapping process can be defined as initial processes that must be run before the POCT device can be used for the testing [124]. The POCT device ecosystem is complex in that it involves security relationships that must be established between the four key stakeholders (POCT Device Users, POCT Device, Network Providers, M2M operators, and POCT Application Provider). During the bootstrapping process, the trust relationships between the four main entities must be established successfully. The scalability of the POCT devices deployment is a major factor, as the ecosystem expands with the growth of the POCT device users. Registration of the POCT device on a network needs to comply with the 3GPP standards [125] (including specialized requirements specified by the network providers and the operators [126]). The bootstrapping process will start after the successful network registration. Due to business reasons, the application provider or the POCT user may change operators, and the bootstrapping process must ensure forward and backward compatibility between the network operators, keeping the network boundary agreements intact.

4.6 Summary

In this chapter, security mechanisms that can be used for three main POCT configurations (POCT for testing, POCT as the patient monitoring device, and POCT as an interfacing device) are discussed. It should be noted that the security mechanisms discussed here do not depend on the mentioned configurations; this applies to POCT operation and systems in general.

An attack tree model is presented, considering the main assets that are required to be secured, including the collected data. Port scanning results are presented (using Nmap process) with the constructed asset tree in a real scenario where the POCT is used within a secure in-house environment. The purpose of showing the process is to emphasize the need for the port scanning process for continuous security validation of the POCT system. Any applications within the POCT system potentially can create security violations if they are conducted without security guidelines for developing POCT applications. A rigorous process for accepting any such applications must be in place along with a security monitoring centre for POCT installations.

The security concerns that are applicable to any web-based system are very much applicable to POCT web-based systems. The use of an embedded web server within the POCT is discussed with potential security vulnerabilities such as cross-site scripting, SQL injection, XML injection, and command injection.

Given that the vulnerability issues in the POCT system are unavoidable, methods (security framework) for managing security risks have been discussed. The communication protocol developed (close communication) for the POCT system is the first defence process for securing the data asset. Two kinds of security mechanisms have been discussed, challenge and response-based and data behavioural-based. All the methods discussed here are independent of communication technologies and Radio Access Technologies such as 2G, 3G, and 4G. Finally, M2M security that applies to the POCT system discussed. It is a good example of the M2M communication environment. The

POCT system deployment depends on a successful working relationship among multiple stakeholders or entities ; the POCT device users, the POCT device, the network providers, the M2M operators, the POCT application providers, and the POCT vendors (OEMs). The key M2M requirements for successful POCT system deployment have been reviewed. POCT M2M security is a complex issue that needs not only technology collaboration but also trust among the main organizations involved.

Chapter 5

P-Cloud and NAS Implementation

5.1 Introduction

A brief introduction for the P-Cloud was provided in Chapter-2, section 2.4. This chapter attempts to explain more details that are not covered in section 2.4. In addition, a practical realization of the P-Cloud is shown with the help of NAS (Network Attached Storage). The connection to P-Cloud can be configured three ways as shown in Figures 5.1, 5.2 and 5.3.

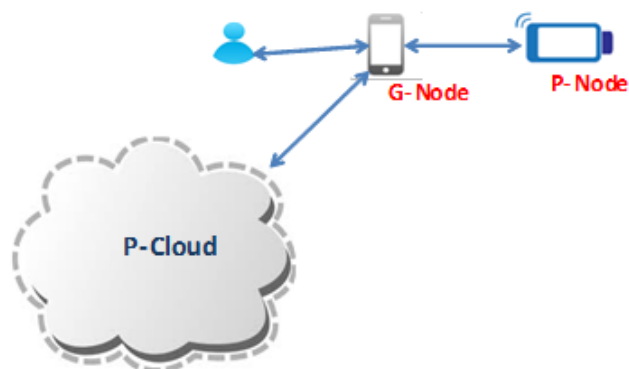


FIGURE 5.1: Accessing P-Cloud via G-Node

In Figure 5.1, G-Node plays a role in forwarding the POCT data from P-Node. In this configuration, P-Node is capable of connecting

to G-Node using non-cellular connectivity such as Bluetooth or WiFi or similar radio access technologies. The configuration leverages the security mechanisms provided by the G-Node for transmitting data to P-Cloud.

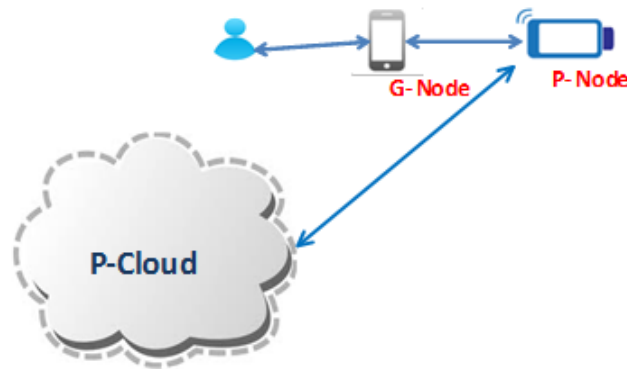


FIGURE 5.2: Accessing P-Cloud via P-Node

In Figure 5.2, P-Node has direct connectivity to P-Cloud. In this scenario, P-Node may have a user interface which can be used for initiating an assay. Then the data is collected, and it is sent to P-Cloud directly. Providing a secure communication link is overhead in this configuration because the security provided by the G-Node wireless infrastructure is not utilized. If the G-Node is a Blackberry device, it would be advisable to leverage the security support infrastructure, Blackberry Enterprise Software(BES) used by the G-Node. There are alternatives configurations such as Mobile Device Management (MDM) solutions can be used in place of the BES. Multiple implementations of the MDM are available from mobile OEM vendors such as Apple and Samsung.

Figure 5.3 shows a hybrid configuration. Both G-Node and P-Node are capable of sending the data to P-Cloud. For instance, G-Node can start the testing process and moves away from P-Node, which forces P-Node to send the test data to P-Cloud. It is possible to implement an appropriate response to a scenario in P-Node . If the

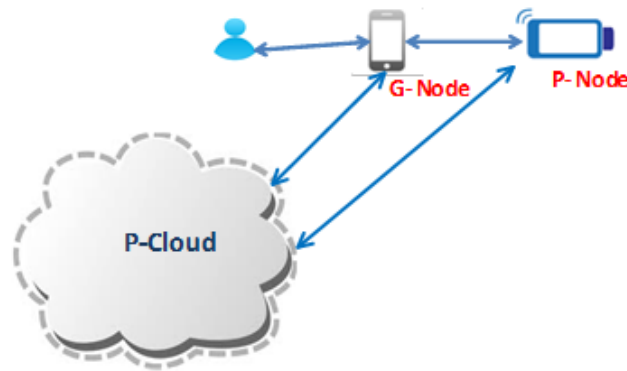
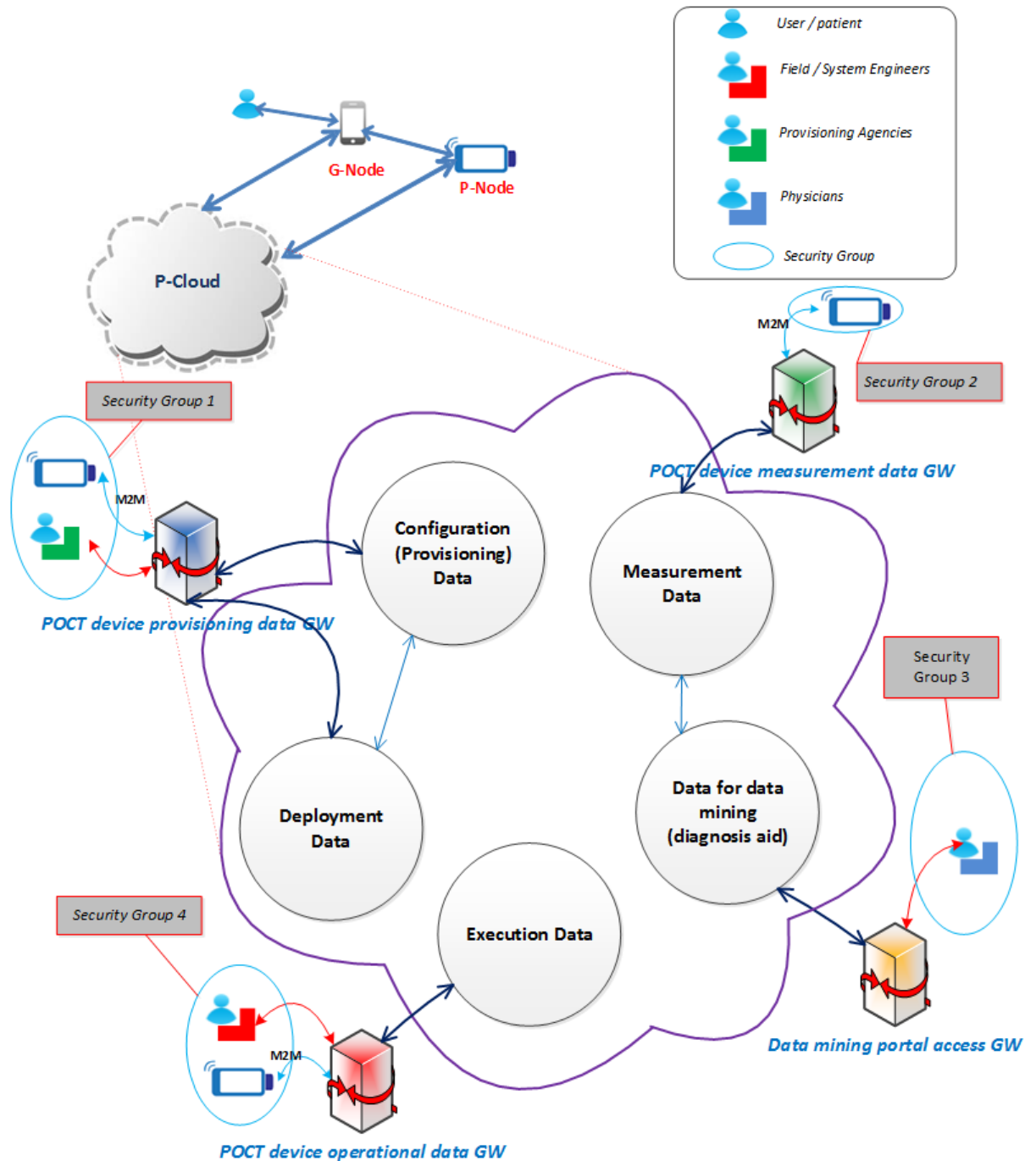


FIGURE 5.3: Hybrid configuration model

connectivity between P-Node and G-Node was lost for a predetermined time, then P-Node can be instructed to send the data directly to P-Cloud. How long the P-Node can tolerate the loss of connectivity to G-Node can be predetermined. Thus the hybrid configuration model provides some flexibility in which the operations can be carried out.

There are no hard and fast rules that which entity needs to be connected to the P-Cloud. The P-Cloud is capable of providing cellular, or non-cellular (BT, WiFi, or similar access technologies) links to the G-Node or the P-Node. It is entirely up to the deployment cost-effectiveness to select a configuration for the usage.

As explained in Chapter-2, P-Cloud has five independent databases conceptually (Figure 5.4). The smartphone application developers will create applications which will provide access to the gateway entities in the P-Cloud. The P-Cloud is conceptualized to meet the HIPAA [28] privacy compliance requirements. In P-Cloud the privacy compliance is designed by data segregation. Security groups are employed to separate data boundaries. The P-Cloud consists of four kinds of architectural artefacts as shown in Table 5.1. The Table shows the purpose of each element in the system.



Cloud Base Framework For Handling POC Data

FIGURE 5.4: Five types of P-Cloud Databases (with direct connection from P-Node)

There will be four types of human users: Patient, Engineers (Field and Systems), Provisioning agencies and Physicians, who interact with P-Cloud. Their access will be managed via security groups (please refer Table 2-3, Chapter 2 for details).





Architectural Component Name	Purpose of the component / Explanation
	This indicates a gateway which provides access protection for P-Cloud. There are four gateways linked to the databases configured in P-Cloud.
	This symbol is used to depict the databases within the P-Cloud. There are five databases shown in Figure 5.4.
	This indicates the M2M communication link between P-Node and P-Cloud
	This shows the connectivity between the type users and the P-Cloud.

TABLE 5.1: Architecture Artefacts

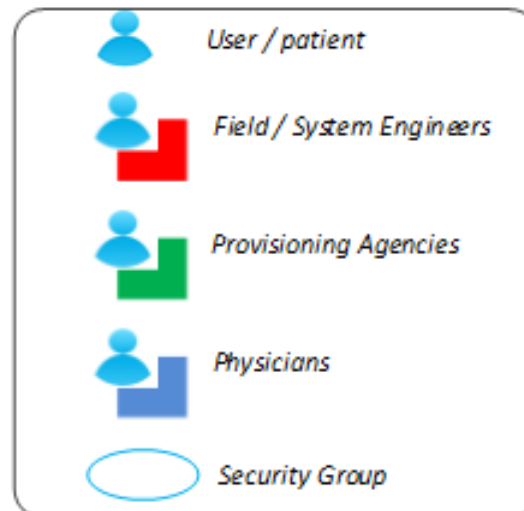


FIGURE 5.5: Four types of human users

5.2 Configuration database

A configuration database (Figure 5.6) is a place for storing POCT devices provisioning data. The configuration database will be accessible to all the provisioned POCT devices (M2M link). Also, the

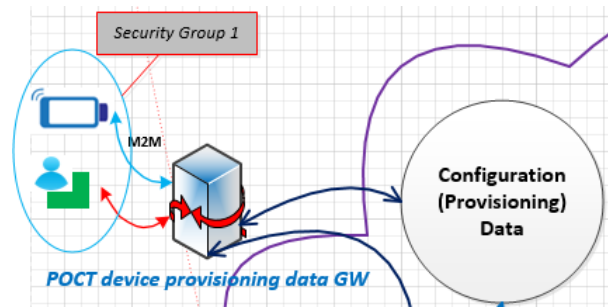


FIGURE 5.6: Configuration database segment - (part of P-Cloud, Fig 5.4)

configuration database is available to various health care providers and lab associates. The provisioned parameters will be pushed to the devices from P-Cloud, using suitable device management technologies such as OMA DM (Open Mobile Alliance OMA, Device Management-DM) standard processes.

5.3 Measurement database

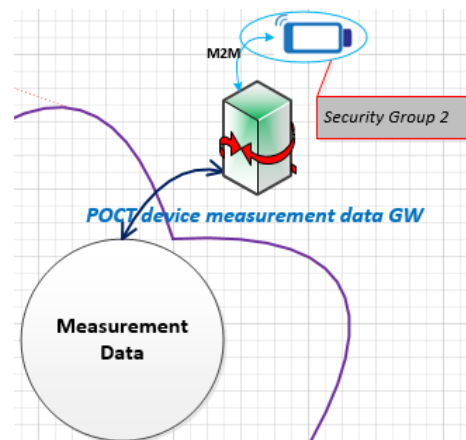


FIGURE 5.7: Measurement database segment - (part of P-Cloud, Fig 5.4)

The measurement database (Figure 5.7) will have the data collected during testing, and it will only be accessible to active and authorized POCT devices in the system. The devices use M2M link to connect to the gateway and the gateway forward the data to measurement database of P-Cloud.

5.4 Data mining (Analytics) database

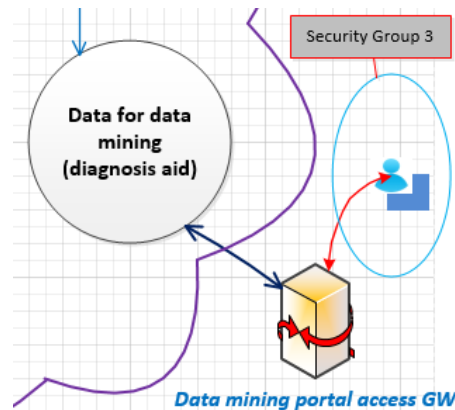


FIGURE 5.8: Analytics database segment - (part of P-Cloud, Fig 5.4)

The purpose of the data mining database (Figure 5.8) is to provide a platform for clinicians to analyze patient data. The data mining database is an aggregated version of the measurement database. The data mining database will be accessible to data mining application developers (for interfacing data reporting and analytic tools) to provide a suitable information portal for the clinicians. The clinicians will access data mining portion of P-Cloud via the data mining portal access gateway.

5.5 Execution (operational) database

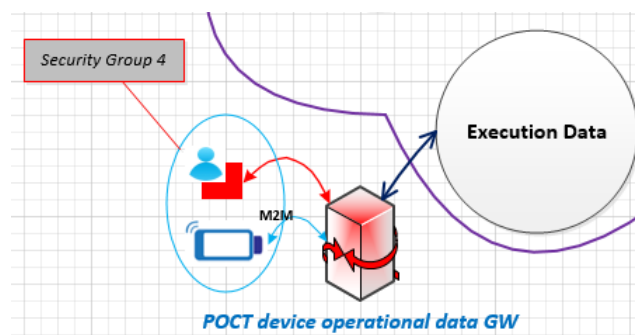


FIGURE 5.9: Execution database segment - (part of P-Cloud, Fig 5.4)

As shown in Figure 5.9, the execution database will collect system operational data. The operational database will help to debug field issues, and it will enable to log critical system execution paths in the provisioned POCT devices in the system. Field engineers and P-Node are the primary users of the execution database. The operational data gateway provides access to the execution portion of the database. The concept here is that the logged data will aid to identify any computational deviation for a particular POCT process. It is essential that all the processes and their execution sequences in the device should be clearly documented with assigned labels. P-Node is designed to initiate a particular process, or a set of processes that will provide measurement data pertains to a diagnostic test. There will be a set of documented system execution paths in the design that will accomplish the result. A diagnostic test will not be valid if the documented execution paths differ during testing. The execution path data which will be logged in the database will reveal any execution deviations against the standard system execution for a given test. This process will not only help to remove any errors in software coding during system acceptance testing (before launching the system) but will also be an essential tool for field engineering team to understand any device related issues. By analyzing the execution data logs, the field engineering team will be able to pinpoint any cause of any issues during actual operation of the system. This process will help to resolve device software issues in a timely manner. The execution log database will be accessible to the field team and the devices in the system.

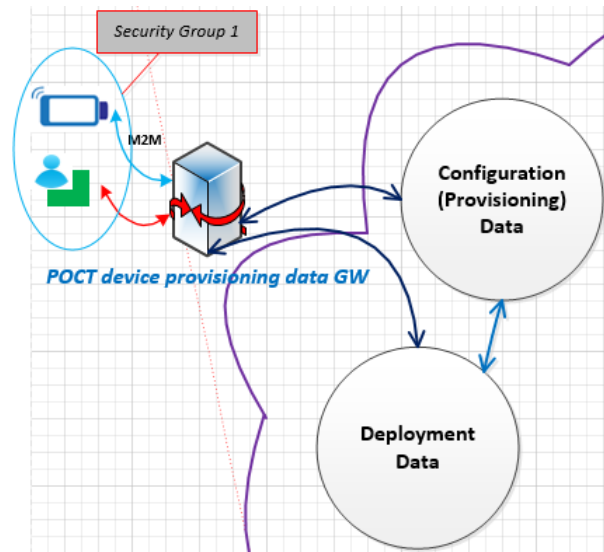


FIGURE 5.10: Deployment database segment - (part of P-Cloud, Fig 5.4)

5.6 Deployment database

As shown in Figure 5.10, the deployment database will have the information about the association of the devices and the G-Nodes that can access the devices in the system. There are multiple deployment scenarios are possible with POCT system. All the tested deployment configurations will be available from the deployment database. It is possible that the association between the POCT device and the G-Node can be one to one association or one to many associations (i.e. one POCT device and many kinds of G-Nodes such as smartphones, PC and any user computing devices). The configuration database will only have the configuration that pertains to the devices in the system, and the deployment database will only have the association data. The principle here is to segregate the data boundaries in different databases and hence avoid any software defects (bugs) and execution errors.

5.7 NAS implementation of private cloud

Network-attached storage (NAS)[127], is a file-level computer data storage server connected to a computer network providing data access to a heterogeneous group of clients. NAS is specialized for serving files either by its hardware, software, or configuration. The NAS can be accessed via services provided by the NAS vendor. POCT Site (indicated in Figure 5.11) access is provided via wired or wireless connection between the device and the NAS. If a public cloud is available, then the NAS (Private Cloud) can be synchronized with the public cloud, based on data sharing policies accepted at the POCT site organization.

5.7.1 Need for Private Cloud

By definition of NIST (National Institute of Standards and Technology)[128], Private Cloud is defined as a bank cloud resource controlled by an organization and used only by that organization. Public Cloud is defined as cloud resources managed by a service provider who is based in a data centre and shared by other customers. There are also other types of deployment of cloud models, community cloud: provided for a certain community and provisioned to meet those communities needs, public cloud, which is provisioned for open use by general public, and hybrid cloud, which is a combination of two or more types of cloud models. The objections to using the public cloud services are, they have inadequate SLA (service level agreements) for PoC type of applications, they are less secure than private cloud deployments. Also complying with regulatory requirements with public clouds can be very challenging[128].

On the other hand, the public cloud is catered for a multi-tenant environment, whereas the private cloud is used for a single tenant (or community) environment. The private cloud provides high security because this type of cloud is dedicated to the single organization. Compliance with medical directives such as HIPAA compliance is impossible with the public cloud hosting. The private cloud deployments are customizable. Any hybrid requirements such as running some of the servers at a higher speed than other servers is possible with private clouds because there is only one organization decides the private cloud deployment strategies [129].

5.7.2 NAS Configuration

NAS configured as a private P-Cloud is shown in Figure 5.11. The NAS differs from a standard file server in term of providing services with those that are not available normally on a server. The NAS was purchased from SYNOLOGY [130], a popular manufacturer of NAS technology. The NAS will satisfy the data security and privacy requirements needed for storing POCT data.

As shown in Figure 5.11, there are two sections to the private cloud configuration, POCT Site and Public-site. These two sites are separated by a firewall configured at the boundary of both sites.

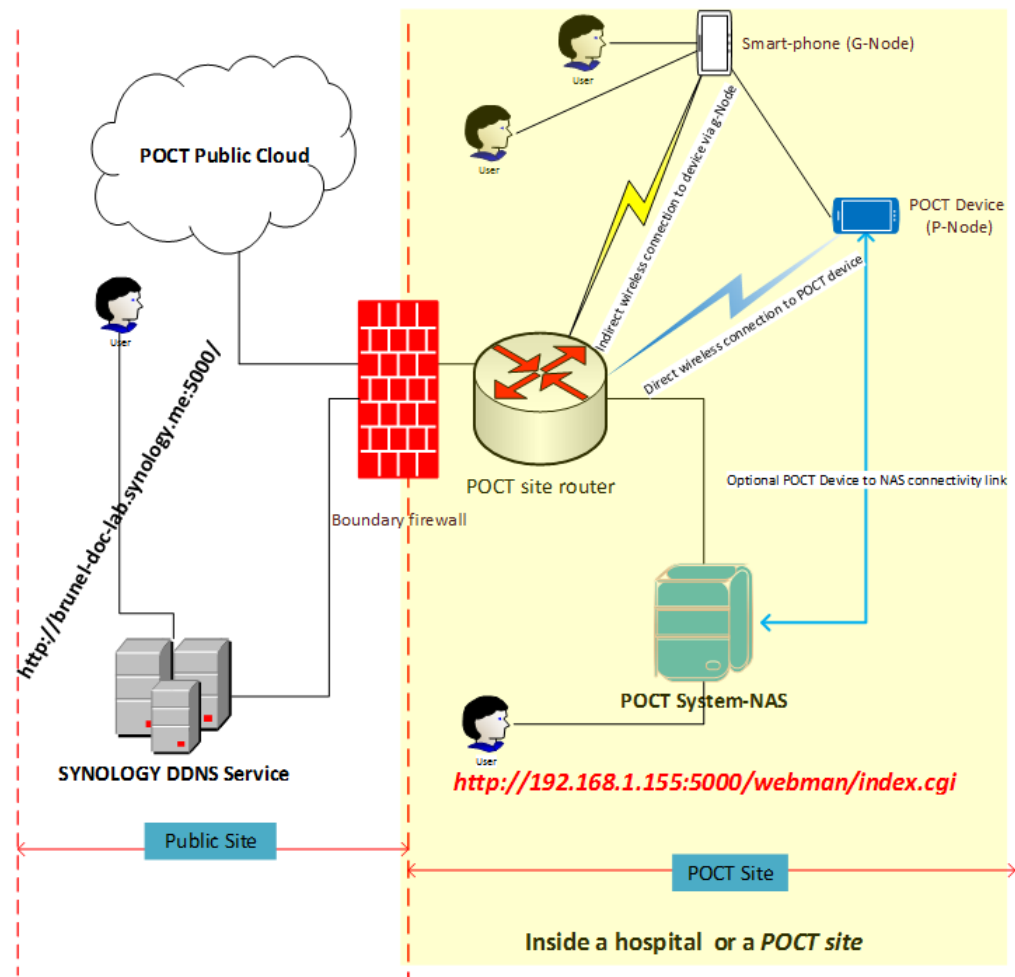


FIGURE 5.11: Architecture of NAS as Private Cloud

There are a few data connectivity paths (Table 5.2) possible with this architecture model.

Connectivity path	Connectivity link	Recommended Access technology
P-Node >> G-Node >> POCT Site router >> POCT System -NAS	P-Node >> G-Node	Wi-Fi, ZigBee, Bluetooth, NFC, USB connection (normally Wi-Fi, BT, USB)
	G-Node >> POCT Site router	Same as above (normally Wi-Fi)
	POCT Site router >> POCT System -NAS	Wired LAN
P-Node >> via POCT Site router >> POCT System -NAS	P-Node >> via POCT Site router	Wi-Fi, ZigBee, Bluetooth, NFC, USB connection (normally Wi-Fi, BT, USB)
Optional P-Node >> POCT System -NAS	POCT Site router >> POCT System -NAS	Wired LAN, Wi-Fi, ZigBee, Bluetooth, NFC, USB connection (normally wired LAN)

TABLE 5.2: Connectivity paths

The actual topology used during the experimentation is shown in

Figure 5.12. A laptop was used as G-Node. USB is used as a connection between the P-Node and the G-Node.

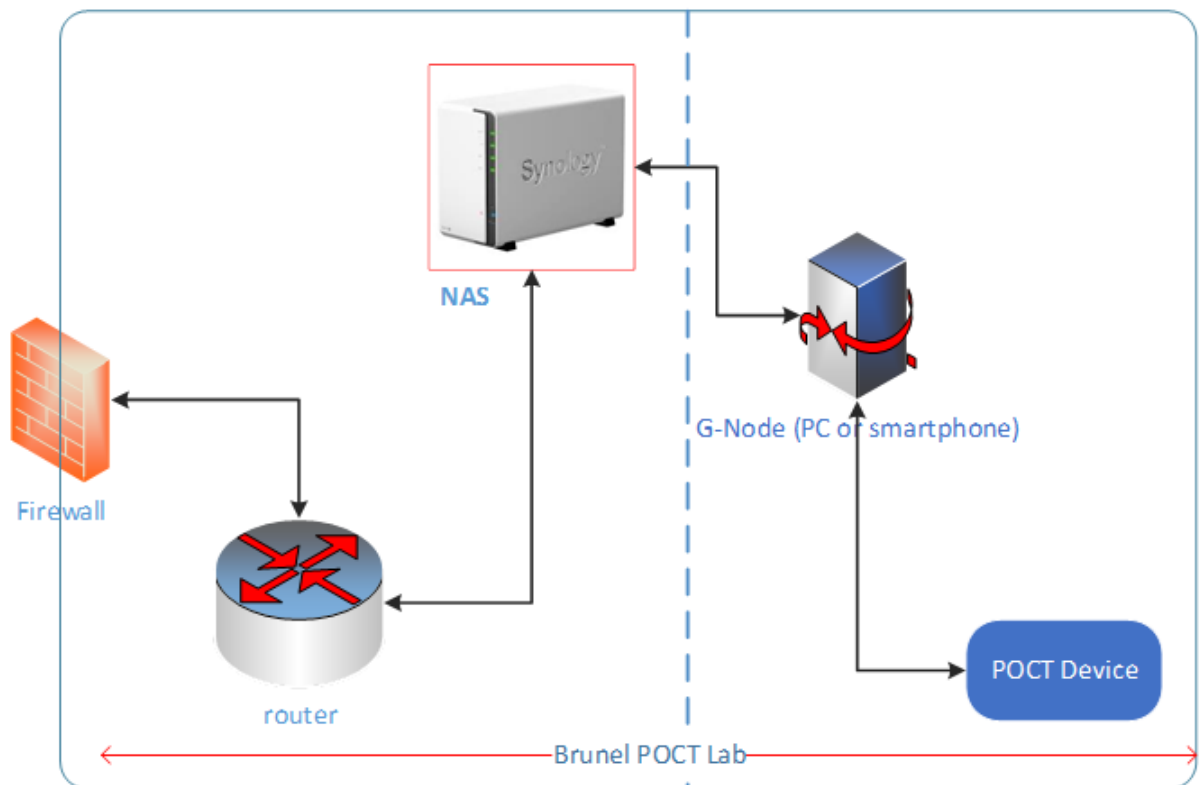


FIGURE 5.12: Physical topology for interconnecting NAS, G-Node, and P-Node

5.7.3 Validation Methodology

The system model that has been configured to represent POCT system consists of a POCT device, G-Node which can be a smartphone or a PC, and P-Cloud which contains cloud data storage which is the test data. As shown in Figure 5.12, the right side of the environment was set up at the Brunel POCT Lab, where the data collection was done using a set of assays. The Data output was captured at the G-Node (a laptop). The G-Node shown in Figure 5.12 was a laptop. The G-Node was connected to the device via USB.

An example sample data collected is shown in Table 5-3. This is an actual test data obtained from an experiment conducted in the

Time	FAM Results											
	TUBE 1	TUBE 2	TUBE 3	TUBE 4	TUBE 5	TUBE 6	TUBE 7	TUBE 8	TUBE 9	TUBE 10	TUBE 11	TUBE 12
0:00:00	211.6	110.3	103.5	100.6	106.9	103.4	103.8	101.6	112.1	111.9	110.6	112.1
0:00:32	137.1	110.3	103.6	101	107.3	103.5	103.9	102.3	112	112	110.4	113.5
0:01:03	107.2	110.3	103.5	100.7	107.3	103.4	104.1	102	111.8	112	110.7	113.7
0:01:34	106.3	110.4	103.6	100.7	107.3	103.5	104.2	101.7	112	112	110.6	114.1
0:02:04	109.6	110.2	103.6	100.8	107.4	103.5	104.3	102	111.7	112.4	111	114.8
0:02:36	106.6	110.2	103.6	101	107.3	103.5	104.2	101.7	111.5	112.1	111.1	115.2
0:03:06	106.7	110.2	103.6	101	107.7	103.5	104.4	102.3	111.5	112.4	111.3	115.3
0:03:37	107.2	110.1	103.9	101	107.6	103.8	104.4	102	111.5	112.6	111.2	116.1
0:04:07	107	110.2	103.9	101.2	107.9	103.8	105.1	101.9	111.4	112.8	111.5	115.8
0:04:38	107	110.1	104.1	101.1	107.8	104	104.7	101.9	111.7	112.8	111.4	115.6
0:05:08	107.2	110	104.4	101.3	107.8	104.1	104.8	101.9	111.8	112.8	111.5	115.7
0:05:39	107.5	110.1	104.4	101.3	107.8	104.1	105.2	101.9	111.5	113	111.7	115.8
0:06:10	107.4	110.4	104.5	101.3	108.1	104	105	102.4	111.6	113.1	111.8	115.9
0:06:41	107.3	110.5	104.7	101.5	108	104.3	105.1	102.3	111.8	113.6	111.9	115.7
0:07:13	108.2	110.5	104.7	101.7	108.1	104.4	105.1	102.5	111.7	113.2	111.8	116.3
0:07:44	107.4	110.4	104.8	101.7	108.2	104.4	105.3	102.7	112.1	113.4	111.9	116
0:08:15	108.9	110.4	104.8	102	108.3	104.4	105.3	103.5	111.8	113.6	112	115.9
0:08:46	107.3	110.8	105	101.9	108.4	104.5	105.5	102.9	112	113.8	111.9	116
0:09:17	107.8	110.7	105.3	102	108.5	104.8	105.2	102.7	112.2	113.7	112.1	116.1
0:09:49	107.5	110.9	105	102.1	108.5	104.7	105.5	102.9	112.7	113.7	112.3	116
0:10:19	107.8	110.8	105.3	102.2	108.6	104.8	105.7	103.2	111.8	114.1	112.3	116.2
0:10:51	107.7	110.9	105.3	102.2	108.8	104.9	105.9	103	112.2	113.9	112.5	116.4
0:11:23	108.8	110.9	105.4	102.4	108.9	105	105.9	103.4	112.2	113.9	112.6	116.1
0:11:53	107.8	111.2	105.4	102.5	108.9	105	106.2	103.1	112.8	113.8	112.5	116.3
0:12:24	108.1	111.2	105.6	102.4	108.9	105	106.2	103.2	112.6	114	112.9	116.5
0:12:56	107.9	111.2	105.5	102.6	109	105.2	106.5	103.3	112.4	114.1	112.6	116.3

TABLE 5.3: Sample Data from POCT device [1]

Doclab, Brunel University. The experiment was related to DNA sequencing using fluorescein dyes: ROX, FAM and HEX [131]. Basically, HEX, ROX and FAM are the fluorescein levels for the LED readings recorded by the POCT device used. T16 desktop Axxin model was used as POCT device. The Axxin device was used as a gold standard. The performance of the assay on the Axxin device was being monitored. For the data derived from the Axxin, the only values that are considered is each tube value for FAM. Rox and Hex are discarded as fluorescein dyes were not in use in the assay. Results obtained for this dyes were default measurement from the device itself. The assay used contained a dye (Eva Green) labelled with FAM. The Axxin was the standard device used to run the assays and then these results were compared with the Felix platform, which was prototyped POCT device by Brunel. In the Figure 5.13, the x-axis represents the time and the y-axis representing the measurement value of the LED light through the dyes. In an actual POCT

measurement, such techniques are used in DNA detection. The plot shows the data collected from the device. In a practical measurement scenario, the collected data will be interpreted by physicians and healthcare experts for diagnosing the medical condition.



FIGURE 5.13: Test Data to be stored in the NAS

There are many ways to bring the data into the NAS. An easy way to access data from the G-Node is to create shared folders both in the G-Node and P-Node and create a soft-link between them. The data collected is linked to the NAS using a soft link creation process as described Figure 5.14. Once the data is copied into the NAS, the data in G-Node was purged for security and privacy compliance.

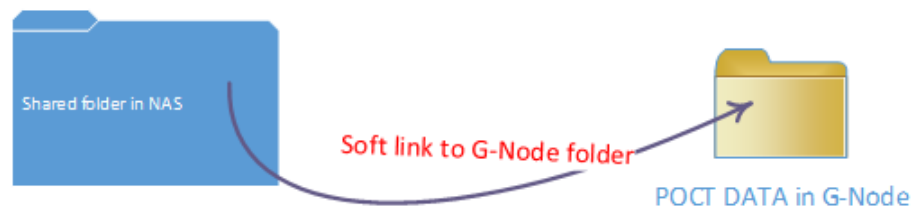


FIGURE 5.14: Accessing data from P-Node

By the process of port forwarding configuration (Figure 5.15) at the edge router, the NAS data can be viewed securely beyond the firewall boundaries, using a standard https link that is created before.

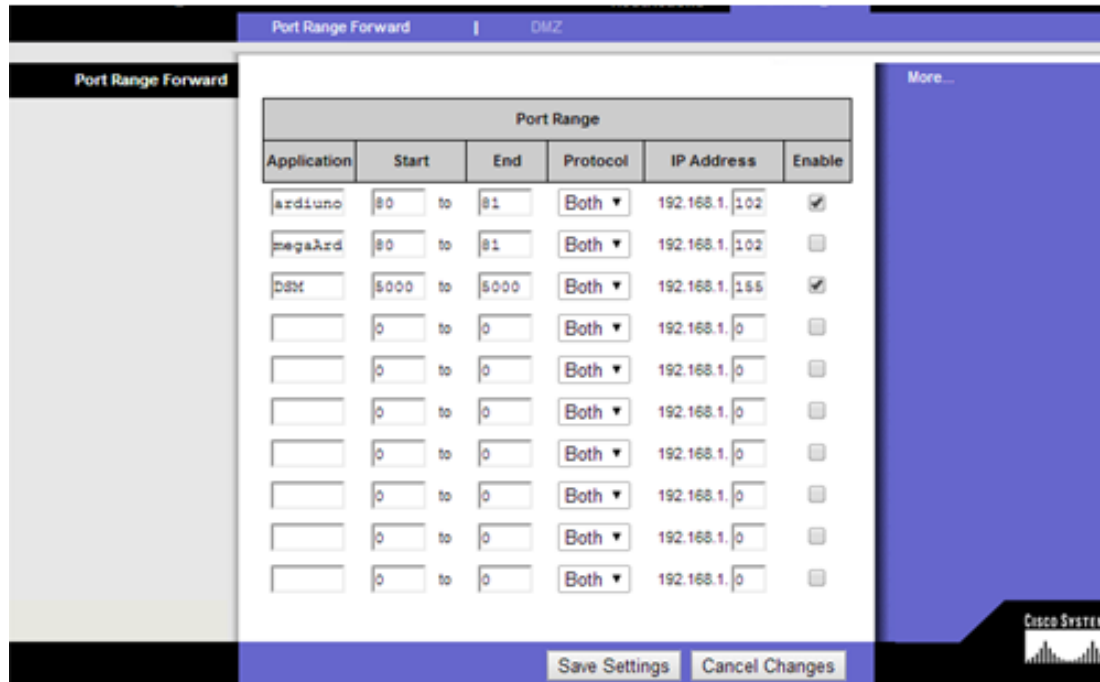


FIGURE 5.15: Port forwarding setup at the router

5.7.4 DDDNS SERVICE

Dynamic Domain Name Service (DDNS) is used to connect the NAS via the Internet. The DDNS helps connecting the NAS to the Internet by assigning a name to IP address of the NAS [130]. The name (Host Name), Brunel-doc-lab was created (Figure 5.16).

5.7.5 Access POCT site from Internet (WAN side access)

The port forwarding process will provide NAS accessibility via the <https://Brunel-doc-lab.sinology.me> (Figure 5.17). This is how the P-Cloud is accessible outside of the firewall. The NAS vendor allows having the DNS service configured as defined by the user.

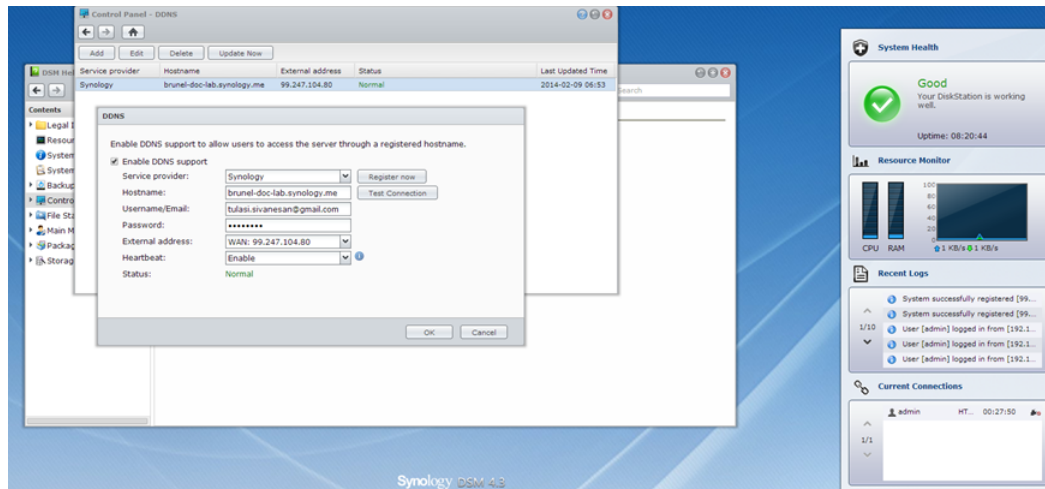


FIGURE 5.16: Creation of Brunel-doc-lab.synology.me

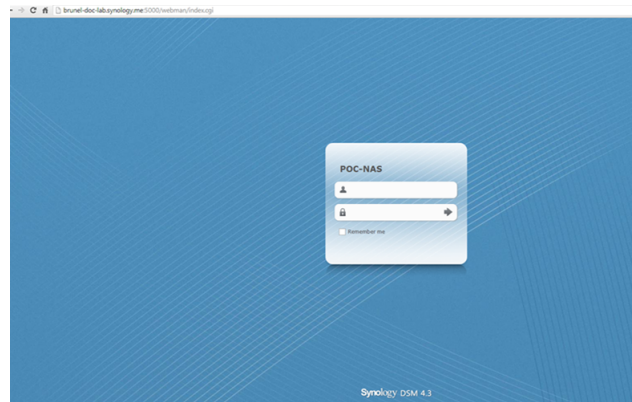


FIGURE 5.17: Login screen for accessing NAS outside of the POCT site via DDDNS

The user needs to provide the user-id and the password configured to access the POCT data.

5.7.6 Access POCT site via an internal IP address? (LAN side access)

The local LAN address can be used to access the NAS within the POCT site (Figure 5.18). This is how the data can be accessed using the 192.168.1.155 (local LAN IP-address) within the firewall boundaries.

Note that the external (WAN side) and the internal (LAN side) user interface (login screen) is the same look alike.

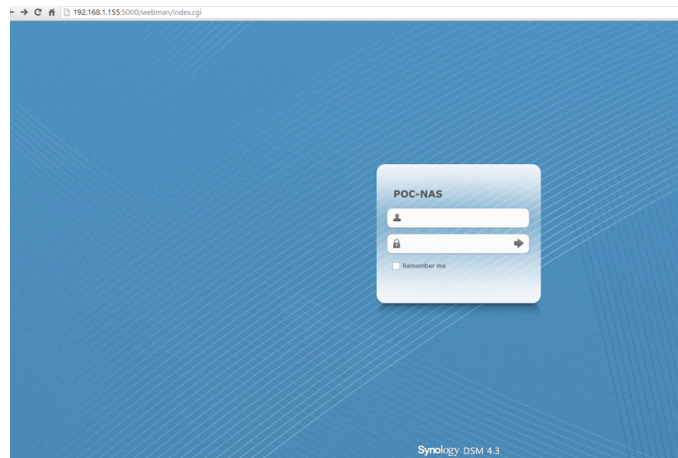


FIGURE 5.18: Login screen for accessing NAS using LAN address

Figures 5.19-21 shows the NAS hardware (Synology manufacturer) used in the experimentation. The hard disk is purchased separately. Usually, a typical NAS hardware can support two standard hard disks.



FIGURE 5.19: One BAY was used for the experimentation

5.7.7 Connectivity between G-Node and NAS

The connection between the device and the P-Cloud (NAS) can be wired LAN or wireless. For the experimentation, the NAS access is provided via Ethernet (wired LAN) link (Figure 5.22).



FIGURE 5.20: Multiple Bays system



FIGURE 5.21: Synology NAS



FIGURE 5.22: USB connection between NAS and G-Node

5.7.8 Strategies for data transfer from POCT sites

There are many ways the data can be shared among the user of the system. The POCT data can be uploaded to an SD card. Then the SD card needs to be encrypted. The encrypted data then can be

transmitted as a file to the NAS. It is possible to update the NAS directly as shown in Figure 5.14, by creating a soft link connection. If the P-Cloud is available only via the G-Node, then an application on the G-Node (Smartphone or PC) can upload the data to the P-Cloud.

The data sharing outside of the POCT site can be accomplished by imposing firewall policies. This will ensure that only required metadata is shared with a 3rd party or public cloud.

5.7.9 NAS QoS (Quality Of Service)

There are two main ways in which QoS can be configured for the NAS Disk Stations. They are called Traffic Control and Speed Limits [132]. NAS provides dedicated QoS management tools to support desired BW. The traffic control helps to ensure a predetermined bandwidth for the users. The speed limit process helps to control bandwidth usage by P-Cloud groups.

5.7.9.1 Traffic control process in NAS

Disk Station Manager (DSM) provides all the required parameters for managing QoS. DSM is provided by the NAS vendor. Traffic control tab of the DSM is shown in Figure 5.23. Traffic control aims to manage the outgoing traffic of services running on the NAS. Multiple rules can be created to set the guaranteed and maximum bandwidth of specific ports. The ports provide certain services to the users.

The Figure 5.23 shows the user interface for the application based BW configuration. A guaranteed value and a maximum value of the

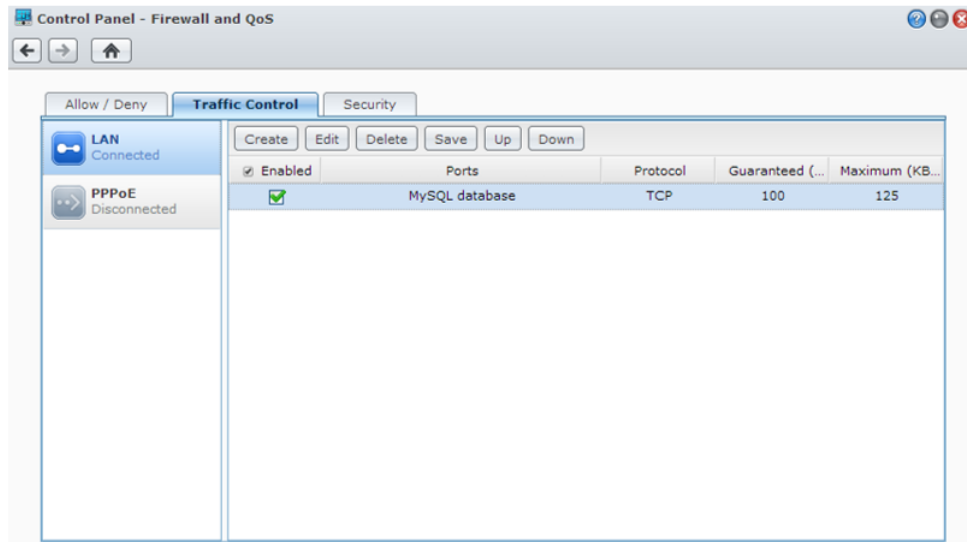


FIGURE 5.23: Control Panel Firewall and QoS

BW can be assigned. The application based BW assignment helps to manage the POCT applications in the system effectively. The NAS administrator can differentiate the QoS based on the protocols used by the POCT applications.

A maximum and guaranteed BW (Kb /s) can be configured per application (e.g. MySQL DB).

A typical BW value can be in the order of a data rate for Narrow Band IoT (NB-IoT) devices. Usually, it is in the order of 20 Kb / s in the case of single tone system and 250 Kb /s in the case of multi-tone system[133][134].

QoS can be configured based on a specific protocol and specific port(s) or ports range as shown in Figure 5.24. The figure shows the user interface for configuring BW based on the port numbers. The port number based BW control is another process of managing the POCT applications. The port numbers based BW management provides more granularity to assign BW for the POCT applications. Figure 5.25 shows some of the lists of protocols with the port numbers.

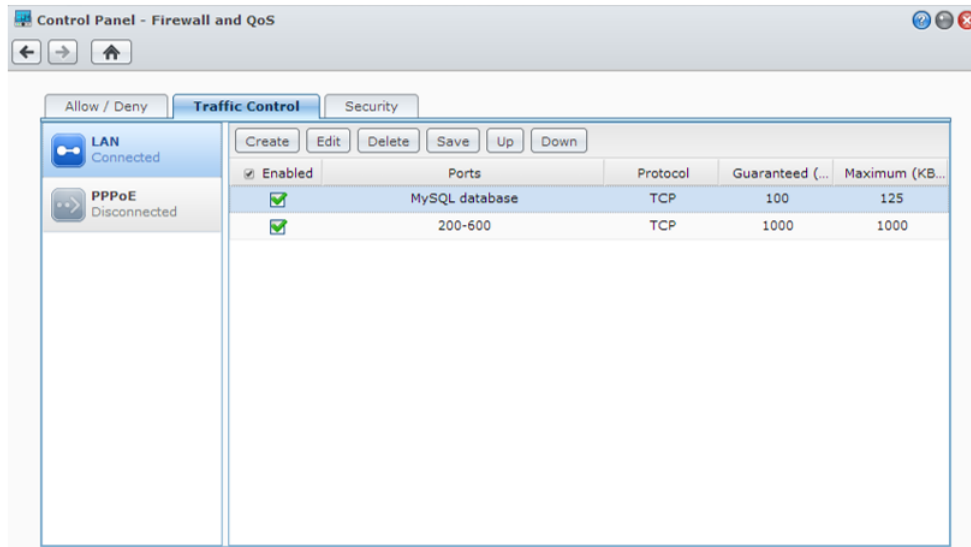


FIGURE 5.24: Specific port traffic with guaranteed and maximum bandwidth

Select Built-in Applications			
<input type="checkbox"/> Enabled	Applications	Ports	Protocol
<input type="checkbox"/>	NTP Service	123	NTP
<input type="checkbox"/>	Network Printer, AirPrint	631	IPP
<input type="checkbox"/>	Network Printer	515	LPR
<input type="checkbox"/>	VisualStation	19999	Search VisualStation
<input type="checkbox"/>	Bonjour	5353	Bonjour Service
<input type="checkbox"/>	iSCSI Target	3260,3261	iSCSI Target
<input type="checkbox"/>	Network MFP	3240-3259	Network MFP
<input type="checkbox"/>	UPnP Service	55900-55910	UPnP SSDP
<input type="checkbox"/>	SNMP service	161	SNMP
<input type="checkbox"/>	FTP file server	21	FTP
<input type="checkbox"/>	Share files with Mac	548	Mac File Service
<input type="checkbox"/>	Windows file server	137,138,139,445	CIFS
<input type="checkbox"/>	Mac/Linux file server	111,2049,892	NFS
<input type="checkbox"/>	WebDAV	5005	WebDAV

FIGURE 5.25: Built-in applications

Traffic control rules can be created for the individual applications. A list of built-in applications is shown in Figure 5.24. The outgoing traffic from the NAS will be throttled to meet the conditions set by the rules per application. There will be an overhead due to the processing of the rules which is negligible with use of higher grade network processors. The definitions of these parameters used in the

rules are defined as follows [135, 136].

5.7.9.2 Guaranteed Bandwidth

This parameter defines the outgoing traffic service, and it guarantees to serve when the whole system bandwidth is wide enough.

5.7.9.3 Maximum Bandwidth (BW)

Maximum BW defines the outgoing traffic. This service can be triggered when the entire system bandwidth is sufficient, and there is still system BW remaining.

The algorithm used in the NAS calculates System Output Bandwidth first and then make sure the sum of Guaranteed Bandwidth for each service is not greater than System Output Bandwidth.

$$\begin{aligned} \text{System Output Bandwidth} &= \text{Sum of Guaranteed Bandwidth of each service} \\ &+ \\ &\text{System Remaining Bandwidth} \\ \text{Guaranteed Bandwidth for each service} &\leq \text{Maximum Bandwidth for each service} \end{aligned}$$

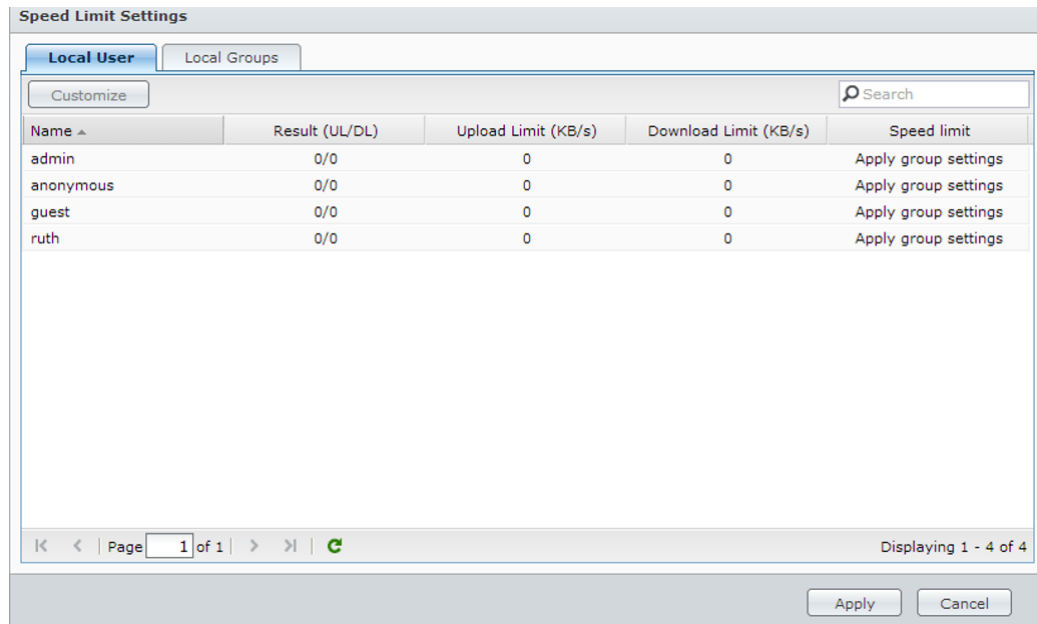
FIGURE 5.26: BW equation

In summary, the traffic control is managed by creating traffic rules via the NAS control panel.

5.7.9.4 Speed limits process in NAS

One of the limitations of Traffic Control is the rules are only applied to individual ports or services. The users also need a fine-grained QoS control on user-level when they transfer files over the Internet [130].

As shown in Figure 5.27, the user groups configured (Name): admin, anonymous, guest and ruth. These users can configure up-link and downlink speeds. This mechanism will provide more granular control of traffic in addition to the port based bandwidth control.



The screenshot shows a web interface titled "Speed Limit Settings" with two tabs: "Local User" (selected) and "Local Groups". Below the tabs is a "Customize" button and a search field. The main content is a table with the following data:

Name ▲	Result (UL/DL)	Upload Limit (KB/s)	Download Limit (KB/s)	Speed limit
admin	0/0	0	0	Apply group settings
anonymous	0/0	0	0	Apply group settings
guest	0/0	0	0	Apply group settings
ruth	0/0	0	0	Apply group settings

At the bottom of the interface, there is a pagination control showing "Page 1 of 1" and a "Displaying 1 - 4 of 4" status. There are also "Apply" and "Cancel" buttons at the bottom right.

FIGURE 5.27: Speed limit process

5.8 Conclusion

P-Cloud implementation as a private cloud, using NAS technology is discussed in the chapter. The security group concepts can be implemented in a NAS platform or a commercially available cloud platforms, by applying system provided privilege mechanisms at data folders or data segment levels. The privilege mechanisms can be used as a way of implementing QoS. But the more advanced process of ensuring QoS is necessary as explained in the chapter. There are two mechanisms: traffic control and speed control are used to provide service level QoS and user-level QoS. The five types of databases within the P-Cloud can be assigned to individual folders in the NAS. The

folders can be assigned to security groups (or user groups) to provide independent bandwidth allocation and user group based traffic control as needed. Realization of P-Cloud using the NAS is a cost-effective way of experimenting new configurations in P-Cloud, which is capable of providing functionalities that are expensive if they are deployed using commercially available cloud vendors NAS solution is scalable as the hard disks can be added as deployment needs grow. The current QoS implementation of NAS by the vendor is linked to WAN (Internet) traffic and LAN traffic. The NAS DSM provides an application called, security adviser for accessing POCT data security. It scans for network connections, user and system configurations, malware software and NAS software updates and highlights any potential security risks. In addition, it can provide a report on security rules configured and their validity. Appropriate action can be taken based on the report [137]. If the FTP is used for transferring data from the testing device, it is possible to have 32 concurrent connection sessions, depending on the NAS model[138].

Chapter 6

Network Models for POCT system deployment

6.1 Introduction

The Point-of-Care Testing (POCT) system can be considered an AI-based system or a Cloud-based computing system, depending on the system boundaries. If the Public Cloud (P-Cloud) is configured as a private cloud within firewall boundaries and all the computation is done with G-Node and P-Node, then the system can be described as an AI-based platform. The AI-based computing systems are independent of public cloud-based computing, standalone systems with interfaces capable of connecting to public clouds. The cloud-based computing systems use the cloud services for most of the computing activities. This chapter presents network models that can be used to deploy POCT systems. These network models apply to both the AI and the Cloud-based platforms. The network models provide connectivity configurations for the POCT usage scenarios. The network model also provides a process for deploying such practical systems. It is a typical application in M2M / domain topics. The

IoT deployment techniques can be used in deploying the POCT systems as well. In the M2M system deployment network infrastructure plays a critical role [139]. It is necessary to provide a good user experience by having appropriate IoT application-sensitive networks [140]. This is a mission-critical application because it deals with a medical diagnosis related to the wellness of humans. It is vital that the network architecture be interoperable at all levels because of the mission criticality of the application [141]. Interoperability is one of the key parameters for successful system operation. The POCT operation demands the offloading of computing powers to the cloud and servers [142]. This chapter explains the goal of making sure that the resources and computing power are shared. Black routing and node obscuring are phenomena with IoT type of networks. It is stated that token-based routing is one of the ways to mitigate packet delays due to black routing [143]. In section 6.4, a similar approach is discussed using a Multiple Protocol Label Switching (MPLS) routing scheme, and it shows the use of MPLS for connecting networking entities within the POCT system. Section 6.2 provides the basic building blocks of the POCT system. Section 6.3 shows the use cases of connectivity for the POCT network nodes. Managing Quality Of Service (QoS) is complex in the IoT type of network, a condition that requires special considerations [144]. In section 6.4, a network topology is presented that will resolve the QoS related challenges in the POCT network deployments. In addition, section 6.4 discusses ways of classifying the connectivity models to simplify the deployment complexities and process of deploying practical systems. Section 6.5 shows a parallel ecosystem that is needed for development and maintenance of the POCT system. Section 6.6 shows several scenarios where usage of commercially available cloud

services and other means for establishing communication paths for POCT systems. Section 6.7 discusses a novel process of managing congestions, called collaborative congestion management.

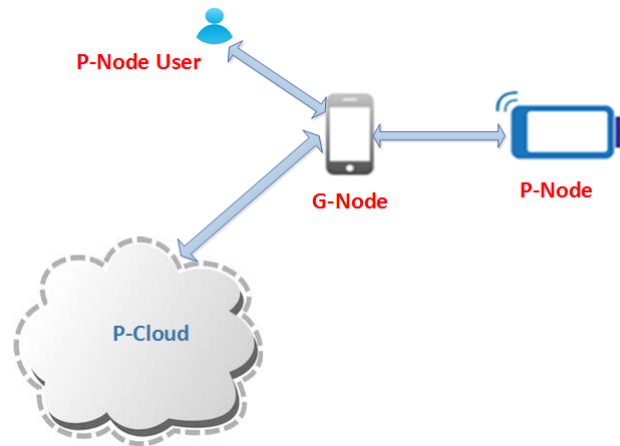


FIGURE 6.1: Fundamental connectivity Building blocks

(This figure is the same as figure 3.3, but repeated here to aid the reader)

6.2 Basic connectivity model

The connectivity links between the nodes can be cellular communication links or short-range type communication links, depending on the user scenarios. The following sections outline important user scenarios concerning wireless connectivity. The P-Node and the G-Node can have both cellular and short-range wireless access technologies integrated into a single module. For discussion purposes, it is assumed the cellular and the technologies like Wi-Fi are enabled in both the P and G Nodes. For the purpose of creating user scenarios, a system consists of multiple basic building blocks; sections 6.3.1-6.3.7 considers them. The seven P-Node and G-Node configuration scenarios are discussed.

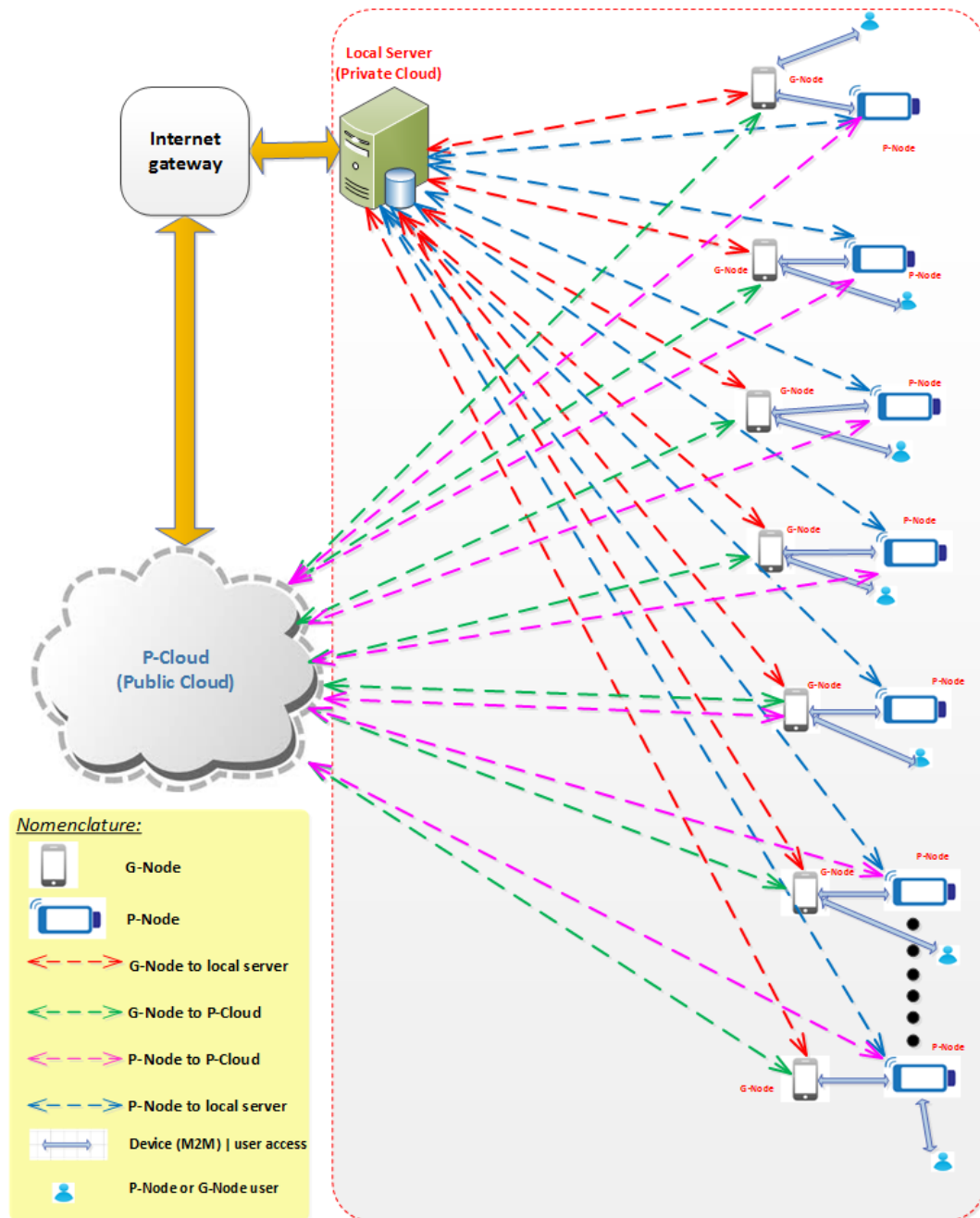


FIGURE 6.2: Multiple ways to connect P-Cloud

6.3 Possibilities for connectivity

Figure 6.2 shows the multiple ways of establishing communication links between the fundamental architectural entities. In the diagram, an internal server is shown within the local location network boundary.

The P-Node and the G-Node are linked via USB or any short-range radio access technologies such as BT or Wi-Fi. The P-Cloud is shown as outside entity of the local network domain. Note that an internal server is a form of internal private cloud that is described in detail in this chapter. The data from the internal server can be shared with P-Cloud via an Internet gateway interface.

The green lines show that the G-Node can be connected directly to the P-Cloud, based on the configuration. The green link is probably a cellular connection (2G, 3G or 4G). Also, the G-Node can be connected (red lines) to the internal server using radio access technologies such as Wi-Fi. The violet lines indicate the connectivity links between the P-Node and the P-Cloud. The violet link is the same kind (mostly cellular connections) as the green link. The blue links indicate short-range radio access between the P-Node and the local server.

All the connectivity links can be operated in parallel. But at least one link must be active in order to transmit measured data from any POCT device. In a practical deployment, all the links may not be activated. The number of links needed will depend on the capacity of the network providers and the Service Level Agreements (SLA) policies that are in place.

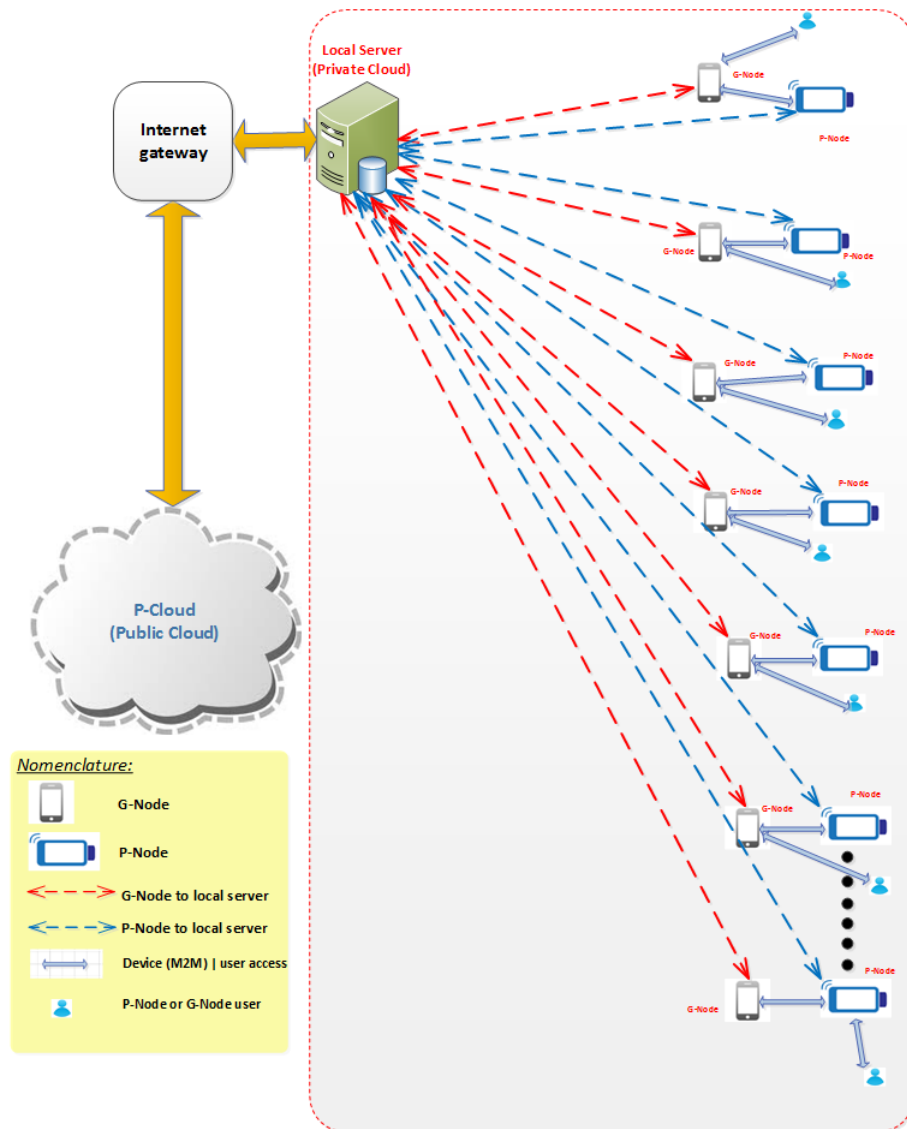


FIGURE 6.3: Data send to local server

6.3.1 Direct connectivity between local server and both P-Node and G-Node

The local server within the local network domain shown in Figure 6.3 is a private cloud. The local data must be transmitted to the P-Cloud for sharing the test data for diagnosis with other institutions. Appropriate server data export (share) policies can be deployed to control data flow. The main advantage of having the local server is to provide data privacy for the patients. All the links shown in the diagram (red and blue links) are mainly short-range connectivity

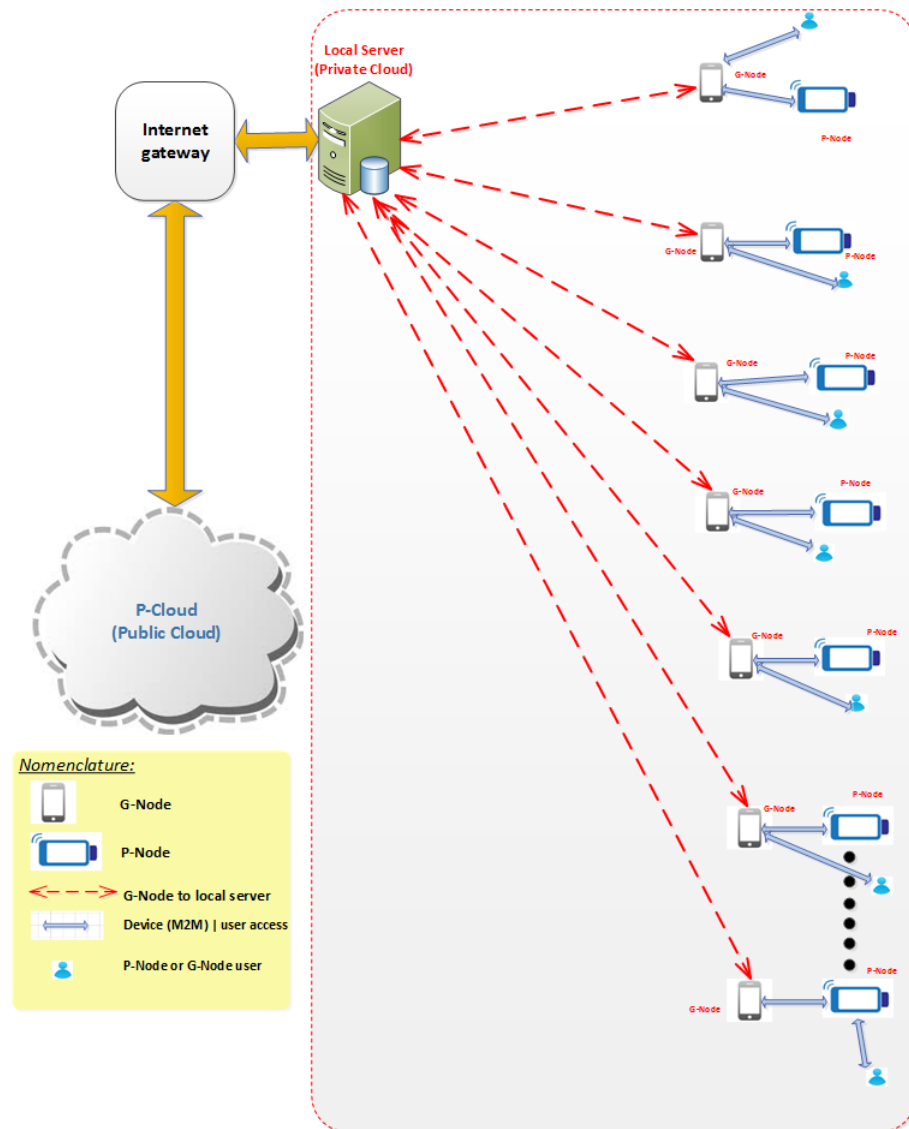


FIGURE 6.4: Data send to local server

radio technologies such as Wi-Fi and ZigBee. This configuration is most useful in a hospital environment where the diagnosis is possible using the local data server.

6.3.2 Direct connectivity between local server and P and G Nodes

In this configuration, the G-Node can play a role as a secure data gateway that transmits test data from the P-Node as shown in Figure 6.4. In this configuration, the communication functionality of the G-Node is reused, which eliminates the need for developing wireless

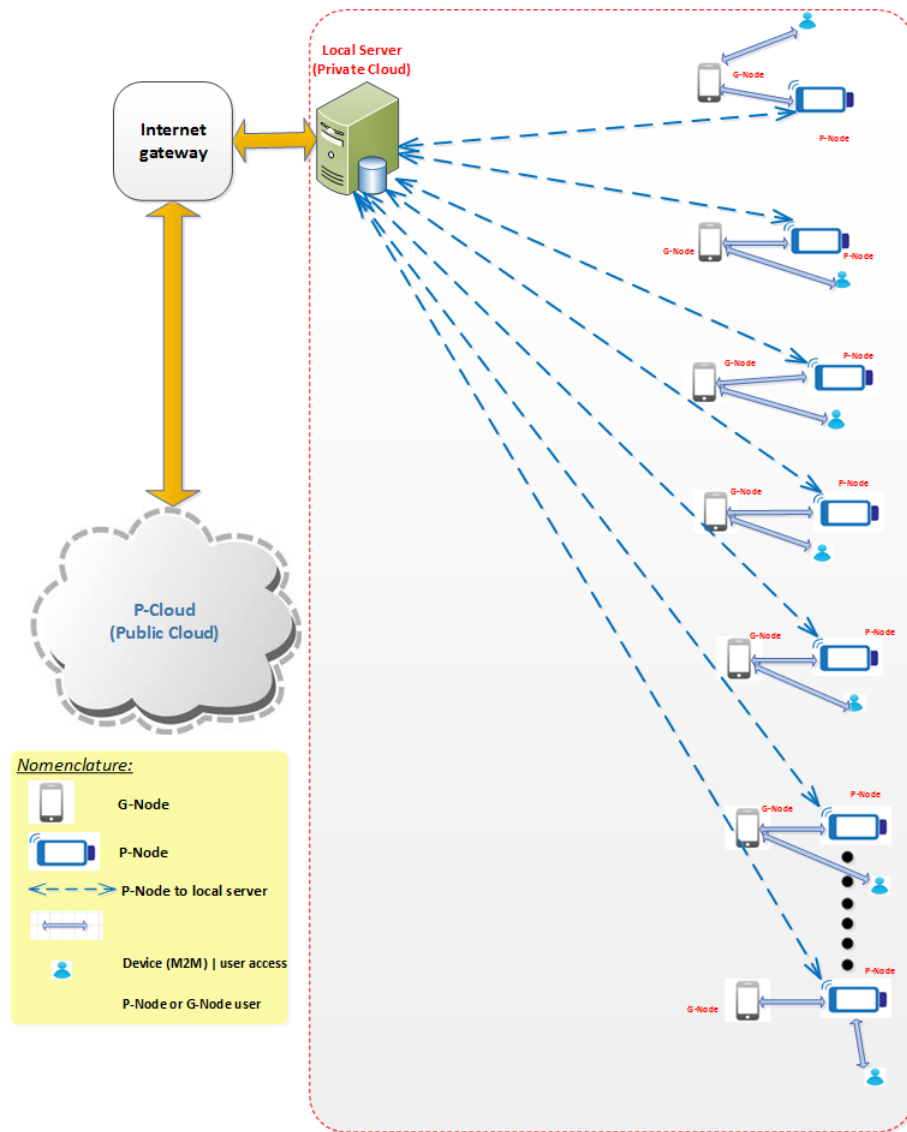


FIGURE 6.5: P-Node as data gateway

capability within the P-Node. P-Node and G-Node can be linked via USB interface or BT interface. The links between G-Node and the local server would be short-range radio access technologies such as Wi-Fi or ZigBee. This configuration is useful for low-cost rural operations. The security is leveraged based on the Wi-Fi network availability, which will include a standard secure Wi-Fi deployment.

6.3.3 Standalone model

The standalone configuration is defined based on the mode of operation of the P-Node. As shown in Figure 6.5, the P-Node has two main roles, data gateway for communication and testing control for execution of relevant assays. The P-Node needs to have the communication modules in addition to the control modules. The role of the G-Node is to initiate the testing process in P-Node. It is also possible that P-Node can be operated with its own user interface, such as a touchscreen. The radio access technologies would be short-range wireless connectivity such as Wi-Fi. The P-Node in this configuration can be considered a benchtop mode of operation if it is used without a G-Node.

6.3.4 Alternate Hybrid model

In the alternate hybrid model, the G-Node and the P-Node are configured to transmit test data to the server. This configuration is named the alternative hybrid model because G-Node and P-Node are used alternatively to provide wireless connectivity. The use case for the configuration is to balance the communication pathways (shown in red and blue in Figure 6.6) within the system. With this type of configurations, P-Nodes with both wireless access and non-wireless access can be deployed. The blue links can be set as Wi-Fi, and the red links can be set as Bluetooth, or even different channels within the Wi-Fi, depending on the availability and loading of the channels. The allocation of the access technologies needs to be selected on a case-by-case basis.

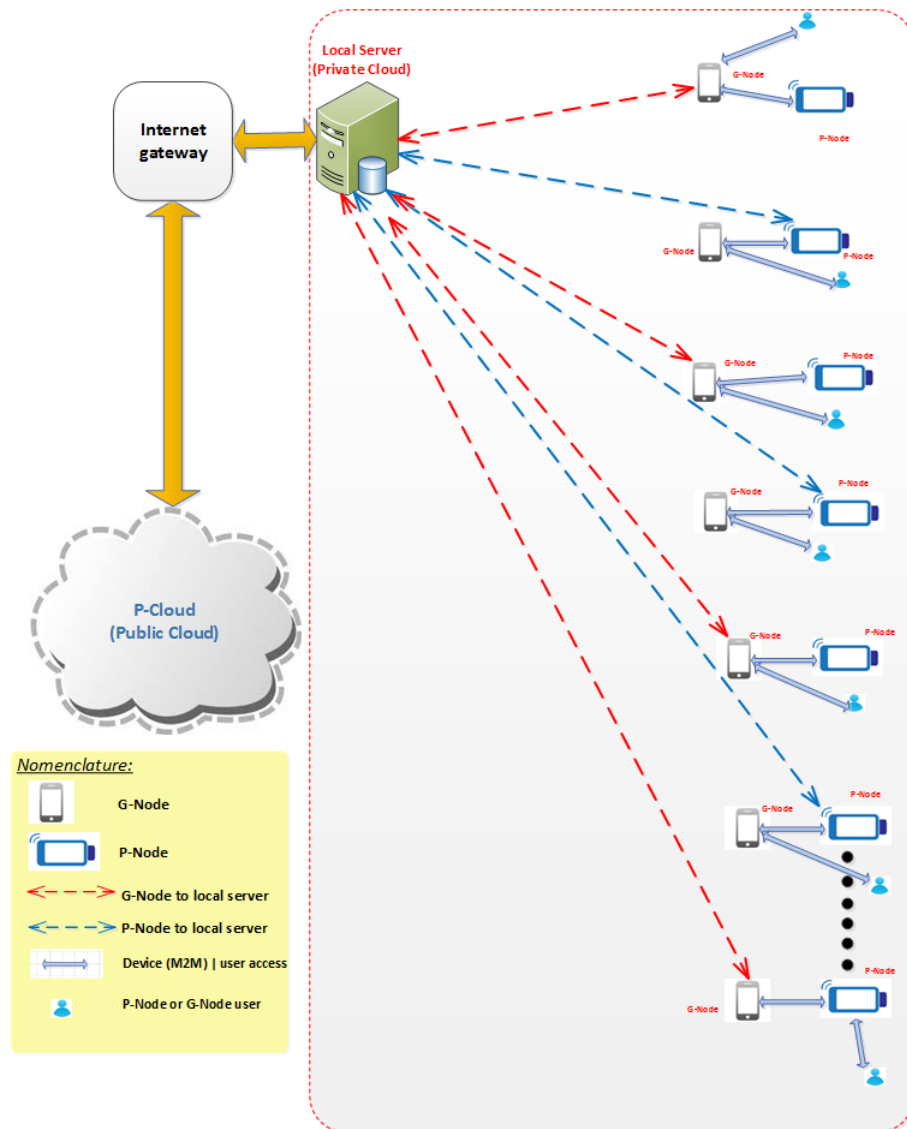


FIGURE 6.6: Hybrid data gateways (Load balancing model)

6.3.5 Standalone Hybrid model

In the standard hybrid model, the data path is shared between the G-Node and P-Node. The use case for the standard hybrid configuration is for managing the availability of access technologies at the location where the system is deployed. Either the two scenarios use the access technology spectrum heavily and cause interference, or the spectrum (bands) is not adequate to meet all the demands of the network users. The standard hybrid configuration provides data collection assurance in both scenarios.

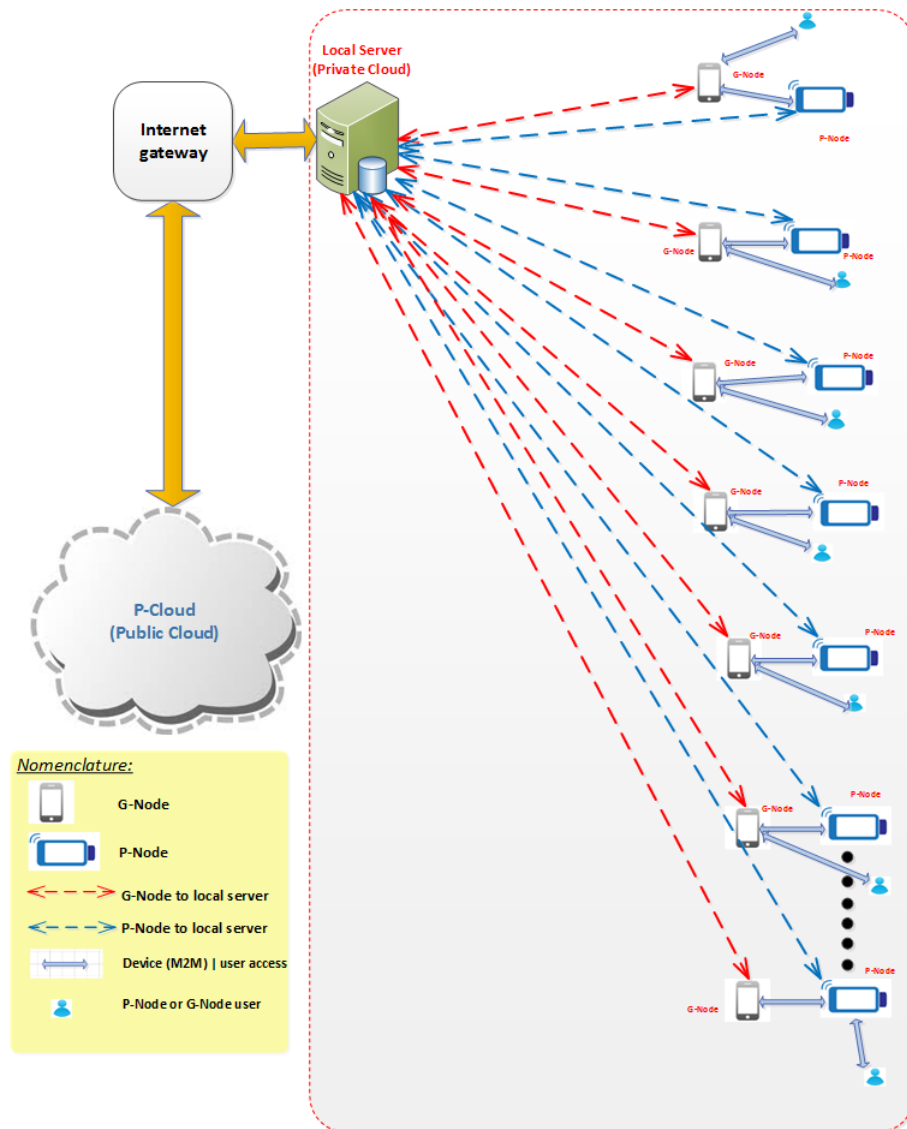
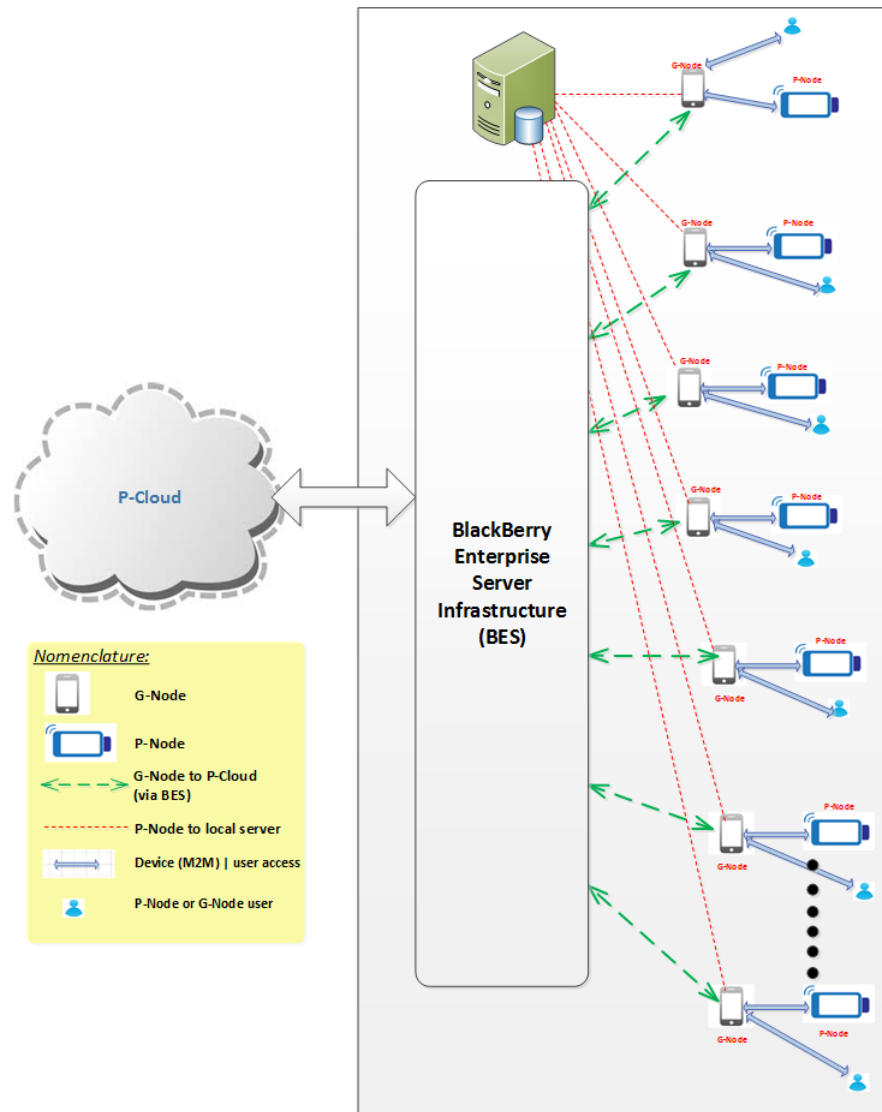


FIGURE 6.7: G-Node as gateways and P-Node as gateways

6.3.6 Use of well-known secure enterprise server infrastructures

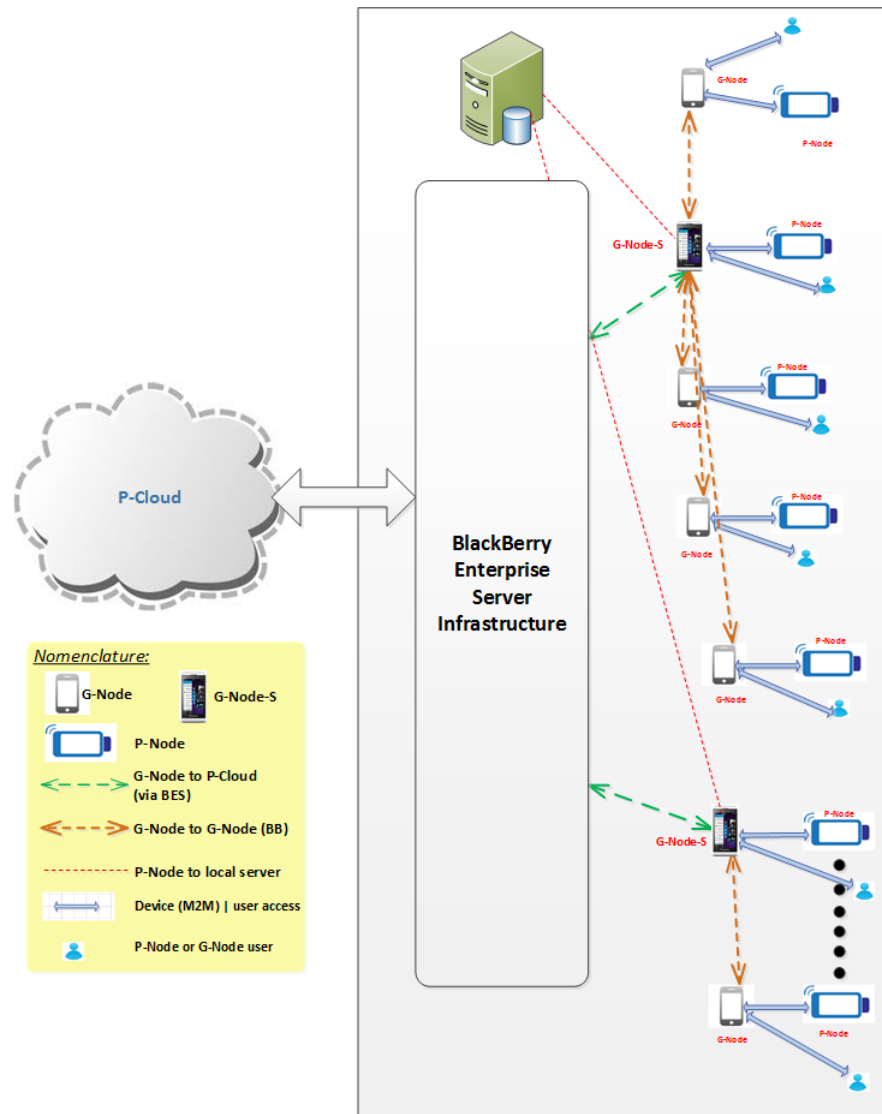
In all the configurations, the POC test data need to be sent to the local server or the P-Cloud securely. Using any well tested secure infrastructure is a smart strategy to send the data securely. Figure 6,8 shows the well-known secure connectivity model Blackberry Enterprise Server (BES) (cites as an example) available for transmitting secure data. It is assumed that the G-Nodes have access to the BES infrastructure via carrier subscription. The BES secure connection is made via the cellular links provided by the carriers. This is shown



**G-Node-BlackBerry Enterprise Server Infrastructure – P_Cloud :
Secure Model using BlackBerry infrastructure**

FIGURE 6.8: BIS (Blackberry server for connecting to P-Cloud)

in the green links. The G-Nodes are connected to the internal server as well, shown as red dotted lines in the diagram. Because it uses an industry-tested secure BES infrastructure, the POCT system can depend on the BES for secure data transmission.



**G-Node-BlackBerry Enterprise Server Infrastructure – P_Cloud :
Secure Model using BlackBerry infrastructure**

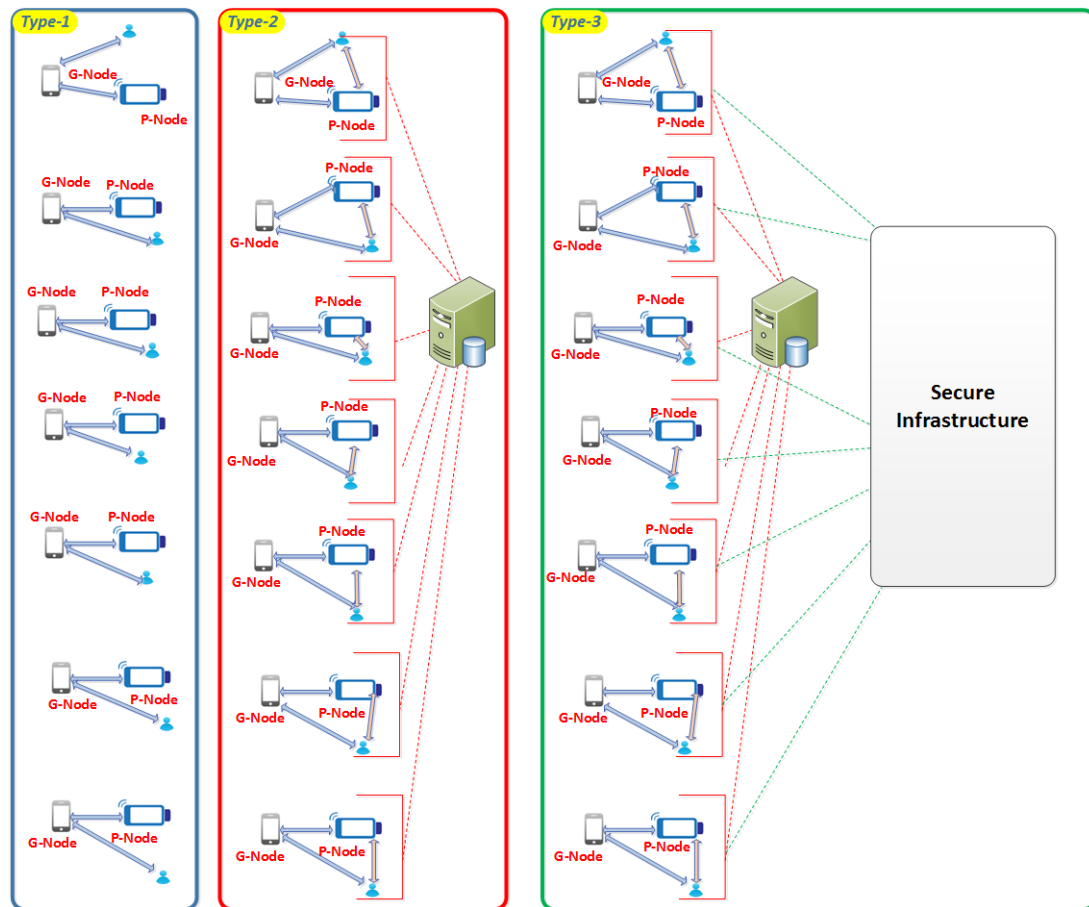
FIGURE 6.9: Using secure smartphone as G-Node

6.3.7 Using secure smartphone device as G-Node with secure enterprise server infrastructure

If all the G-Nodes do not have the secure BES access, it is possible that the G-Nodes that have access can play a role as rendezvous nodes (shown as G-Node-S) to provide secure transmission paths for other G-Nodes that do not have access to the BES. Figure 6.9 shows only two of the G-Nodes can connect to the BES. The two BES enabled G-Nodes are connected to the P-Cloud via the BES

services (as shown in Figure 6.9 with green lines). The selection of the configurations depends on the individual situations and available deployment budget.

6.4 Definition of the POCT system and communities



POCT Community types

FIGURE 6.10: POCT Communities: Type 1, Type 2 and Type 3

A deployment strategy for the POCT system is needed to maintain the quality of service (QoS) or service level agreements for POCT device users. The following section explains a method of accomplishing QoS in POCT-based interconnected systems.

In Figure 6.10, the concept of POCT communities is depicted. The POCT community type-1 consists of separate P-Node and G-Node

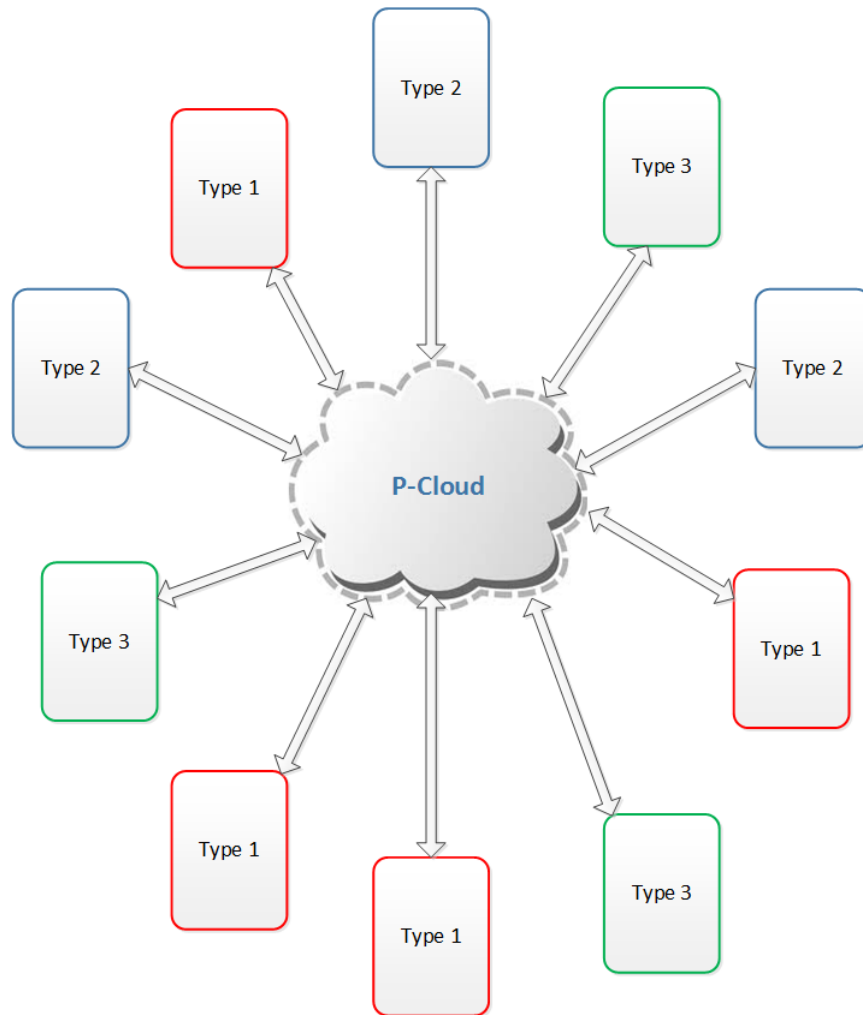
units. The QoS depends on the capabilities of the P-Node or G-Node. In the type-2 community, the P-Node — G-node pair is linked to a local server. The QoS for the type-2 community can be defined at the server level. The type-2 community is meant for using the system within local boundaries of a hospital or a patient testing centre. The type-3 community is meant to serve locally and also extend the data transmission to an external cloud via secure infrastructure. The QoS for the type-3 communities can be defined at the secure infrastructure level and the local server level. Having the three types of POCT communities helps to define Service Level Agreements from the service providers, thus providing guaranteed and desired QoS.

6.4.1 Connecting POCT communities and P-Cloud

Figure 6.11 shows three types of POCT communities or systems that are connected to a central P-Cloud. The QoS levels for the connectivity can be set based on the type of system that is connected to the P-Cloud. The idea of having the classification helps to assign QoS parameters individually.

6.4.2 Realising POCT System

In this section, the idea of building the POCT system is explained. The POCT System is presented as a hierarchical model. The hierarchical approach helps to define access permissions. By having hierarchical boundary policies, the data flow between various levels can be controlled. There are four independent regions of control shown in Figure 6.12, POCT system, POCT zone, POCT site and



POCT-Community based QOS and SLAs

FIGURE 6.11: Connecting to P-Cloud with multiple types of POCT systems

POCT units. The POCT-System is made up of multiple POCT-Zones. The POCT-Zones are deployed with multiple POCT-Sites. The POCT-Site is a collection of multiple POCT-Units.

6.4.3 POCT Units

The POCT unit can have two kinds of configurations, (G-Node and P-Node) and (standalone P-Node), as shown in Figure 6.13. The advantage of the G-Node and P-Node configuration is that the security

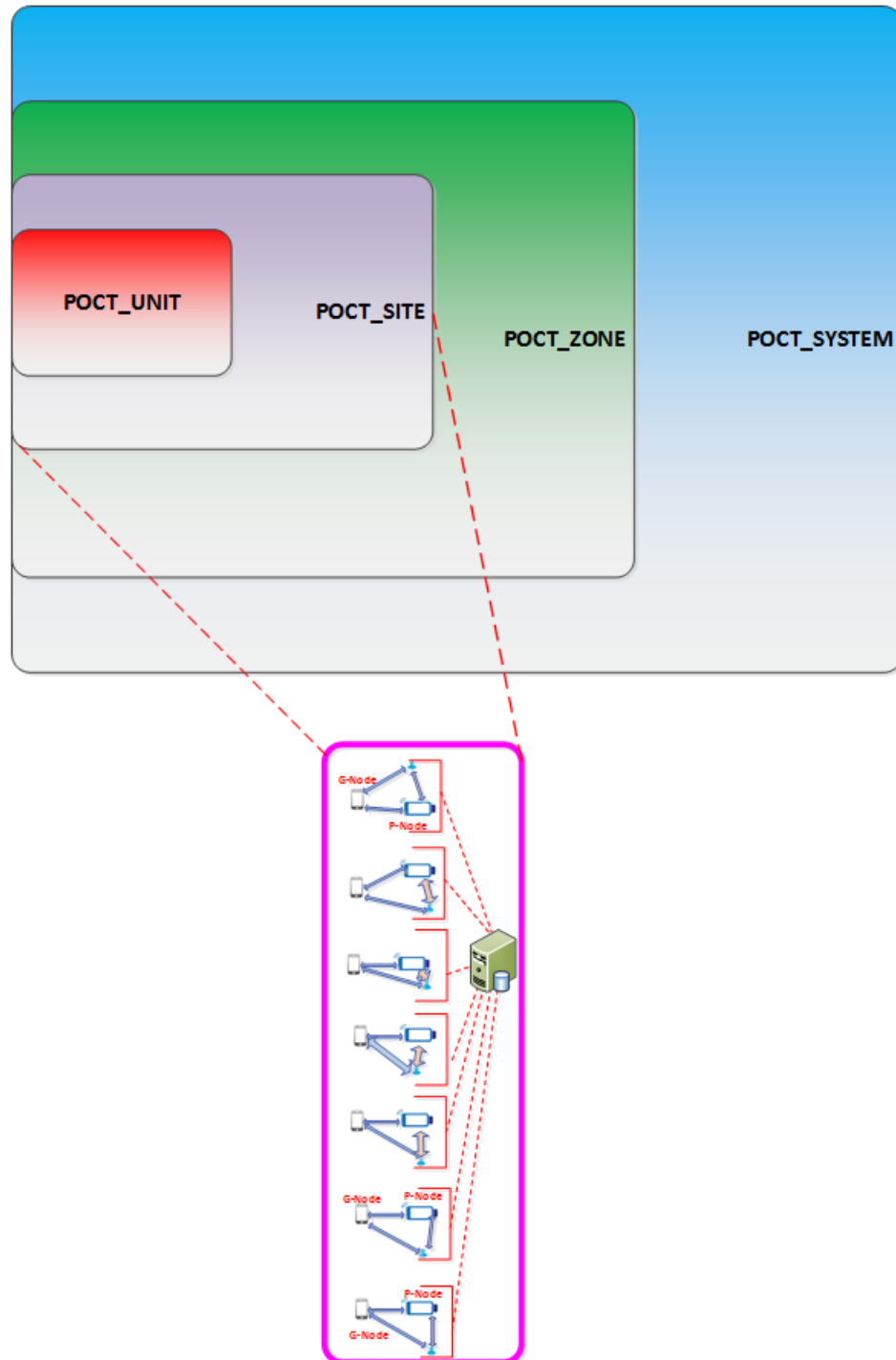


FIGURE 6.12: POCT System Hierarchy

requirements can be met by the G-Node alone (or leveraged by the G-Node) and the P-Node can be designed for the POCT testing, using the security mechanisms provided by the G-Node. The G-Node and P-Node pair configuration provides a cost-effective solution because

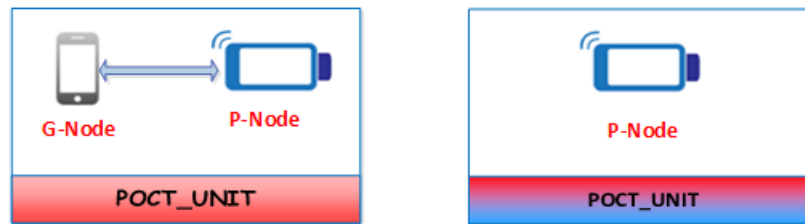


FIGURE 6.13: Definition of POCT unit

the patient can leverage the existing data communication functionality from the G-Node.

On the other hand, the standalone P-Node configurations provide user with a display and keypad interface to conduct POC testing. After the completion of the testing the P-Node data needs to be sent over to P-Cloud for diagnostics analysis. The standalone configuration is useful during communication outages.

6.4.4 POCT Site

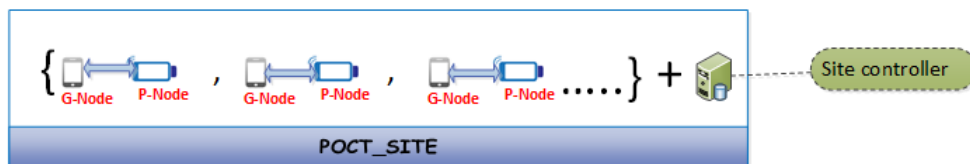


FIGURE 6.14: Definition of POCT site

A POCT site is constructed with multiple POCT units and a server as a site controller. The site controller includes data storage. A POCT site is an example of a testing location in a hospital. The communication links within the site normally are deployed using a short-range connectivity radio access technology such as Wi-Fi.

The number of POCT Units in a POCT Site is determined by the capability of the site controller. The performance parameters are to be looked at based on the site needs.

6.4.5 POCT Zone

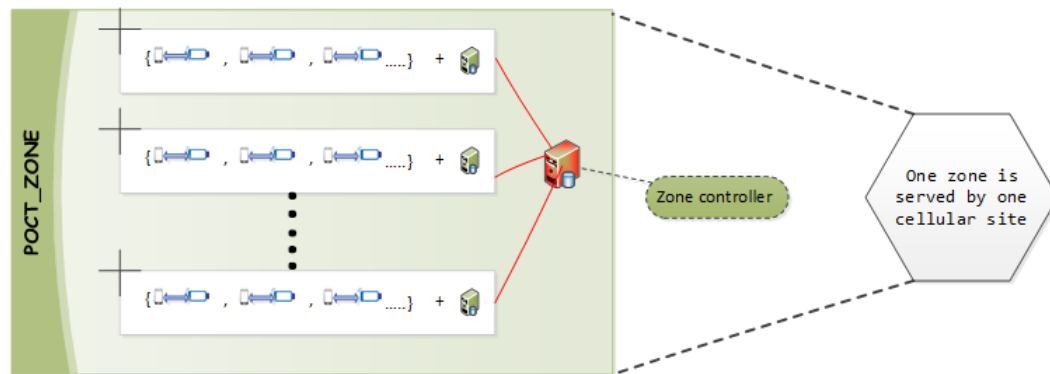


FIGURE 6.15: Definition of POCT Zone

The construct of zone architecture helps to connect multiple sites. The zone will have a server, called the zone controller. This zone controller links multiple site controllers via cellular radio access technology. In Figure 6.15, the zone is shown as a cellular site. But there are many configurations possible with available network operators. Since mobility is not part of the POCT zone, the usual challenges such as handover from a cellular RAT to another cellular RAT are not applicable. Certain cellular bands are allocated by the network operators for the kind of operations that are expected from the POCT zone scenarios. The communication between POCT zones to POCT sites can be viewed as extensions of the IoT.

The zone and site controllers normally are developed by the OEMs or module vendors, such as Telit or Foxconn. The process of deploying carrier (operators) compliance equipment is lengthy and complex. It starts with the modem vendors, such as Intel or Qualcomm, who get involved in the chip level certification with the network operators, such as ATT (other well-known network operators: Orange, T-Mobile, Verizon, Telstra, Sprint, China Mobile). The module vendors (e.g. Telit, Sierra Wireless, u-blox) conduct the device or zone

controller level certification with the operators. The modem and the server module need to be certified by the network operators.

6.4.6 POCT System

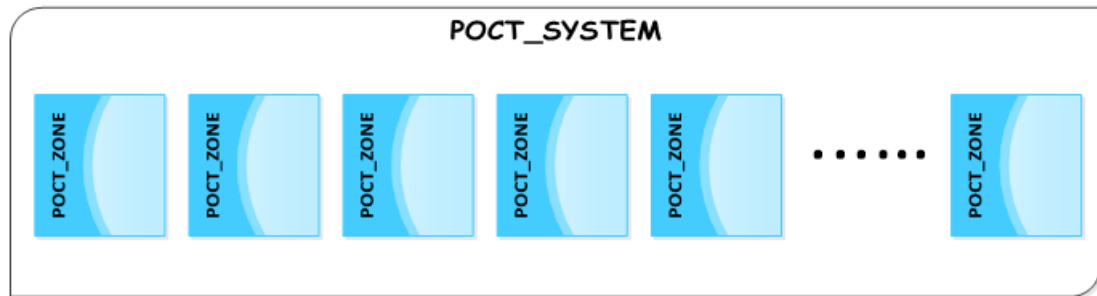


FIGURE 6.16: Definition of POCT System

The POCT system consists of multiple POCT zones. The number of POCT zones in a POCT system, in other words, the capacity of the POCT system, depends on the capabilities of the individual components. It is not possible to predict a value for the capacity of POCT zones.

Figure 6.16 shows multiple POCT zones within a POCT system as an example. Having the hierarchical system development approach helps to determine the capacity of the system as an aggregated value. The hierarchical architectural view differs from traditional approaches that exist currently. In the hierarchical architecture, each level in the system works independently, coexisting as part of the complete system, based on the need. A POCT system can have several units or a single unit. There is no need to deploy the complete system at the beginning, as the system can be expanded as demand increases, based on the needs of the community or users.

6.4.7 Capacity of POCT System - Honeycomb compatible communication links

In a complete system, there are multiple POCT zones that need to be linked. The following section provides a deployment planning rules process by which the multiple zones can be linked. The POCT zone is depicted as a hexagon with six faces. Each face represents the communication interfaces available in a POCT zone. The length of the face represents the capacity of the communication link. Since the lengths of the hexogen arms are equal, the hexogen model represents a POCT zone that has capacity or bandwidth for six equal links. A process of building the connectivity follows.

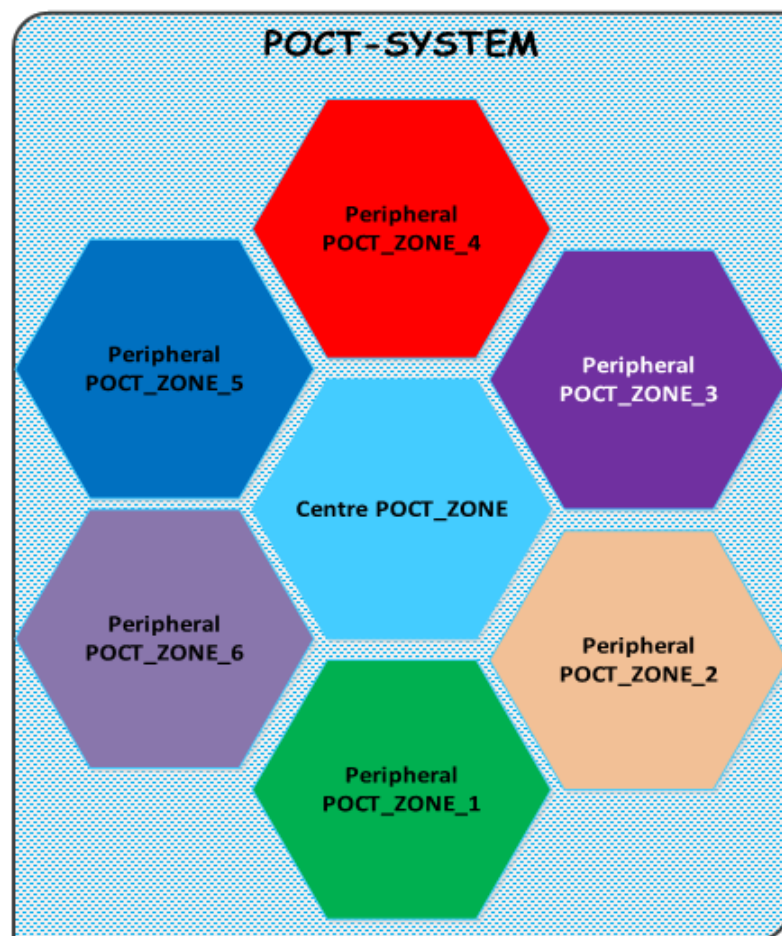


FIGURE 6.17: A complete POCT System (Capacity Model)

The centre POCT zone has six ways of connecting to other peripheral POCT zones. The six connections can be combinations of cellular wireless links and Wi-Fi type links or one kind of communication link. The six communication linkages imply that there are six other POCT zones compatible with communication links provided by the centre POCT zone. It is possible that the peripheral POCT (POCT-Zone-1) zones will then connect with two other neighbouring peripheral POCT zones (PoCT-Zone-2 and POCT-Zone-6), in addition to connecting with the centre POCT zone. Thus, a complete system will have seven POCT zones, as shown in Figure 6.17. This process can be used to plan both the POCT-Zones and the POCT-Sites.

6.4.8 Multiple POCT Systems

Figure 6.18 shows expanded systems with multiple POCT systems. This figure also shows the number of active POCT zones. The POCT systems map reveals the availability of the POCT-Zones at an abstracted level for all the parties involved in deploying the system. The model shown in Figure 6.18 will provide many benefits: the decision to deploy equipment from same vendors or different vendors, system capacity study of individual POCT systems, interoperable experiments with multiple vendors, and deployment cost analysis.

6.4.9 Interconnecting Multiple POCT sites to a P-Cloud

In this section, an approach for interconnecting the POCT sites is explained. Multiple Protocol Label Switch (MPLS) is a protocol that uses existing layer -3 routing protocols, and it is placed between layer-3 and layer-2 in the 7-layer networking model. It is also called layer-2.5 protocol technology.

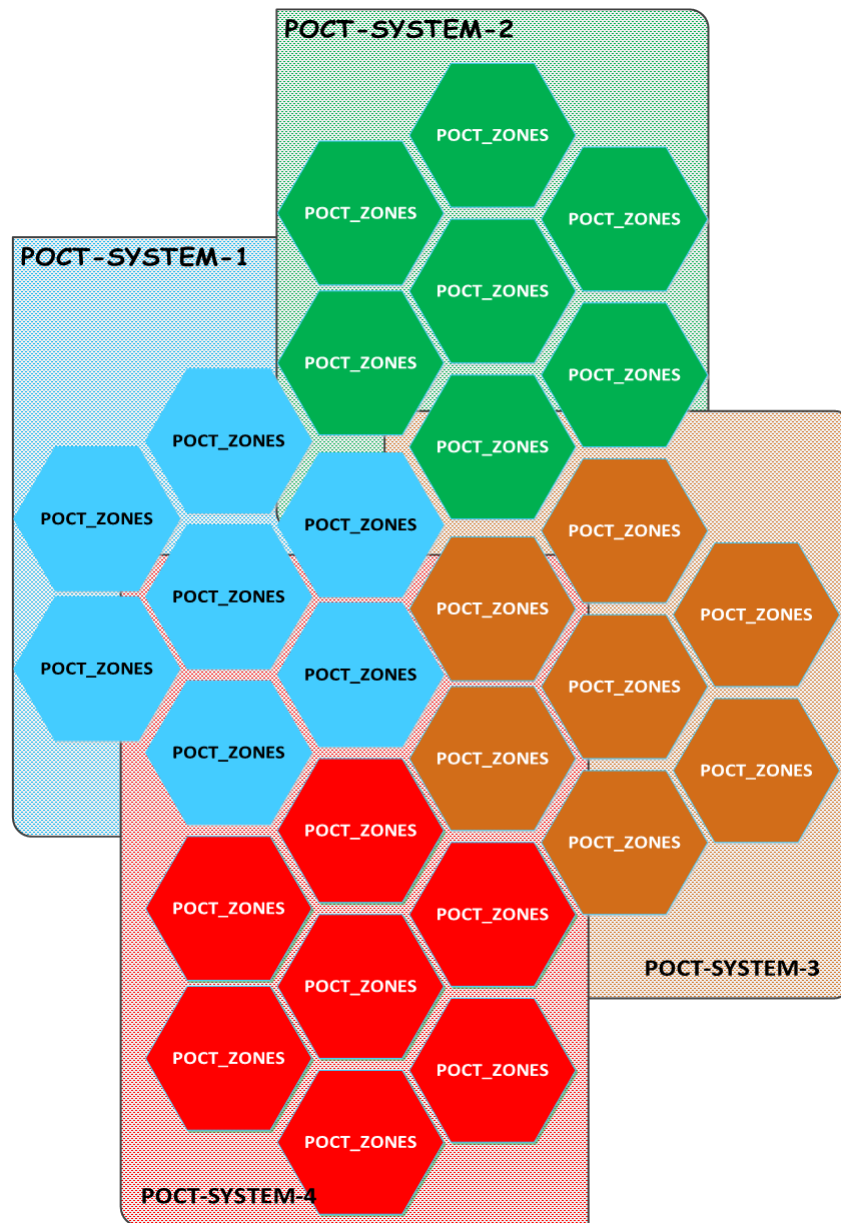


FIGURE 6.18: Multiple POCT systems map

The MPLS is known for multi-vendor systems interconnections in providing faster network connectivity. One of the characteristics of the MPLS is that the number of hops in the network path is less, compared to standard layer-3 connectivity. As shown in Figure 6.19, it will take Router-1, Router-2, and Router-3 to transmit from POCT-Site-1 to POCT-Site-3, which will be three hop counts. But with MPLS, virtually the POCT sites situate them next to each other with respect to the network hop count. The fewer hop counts

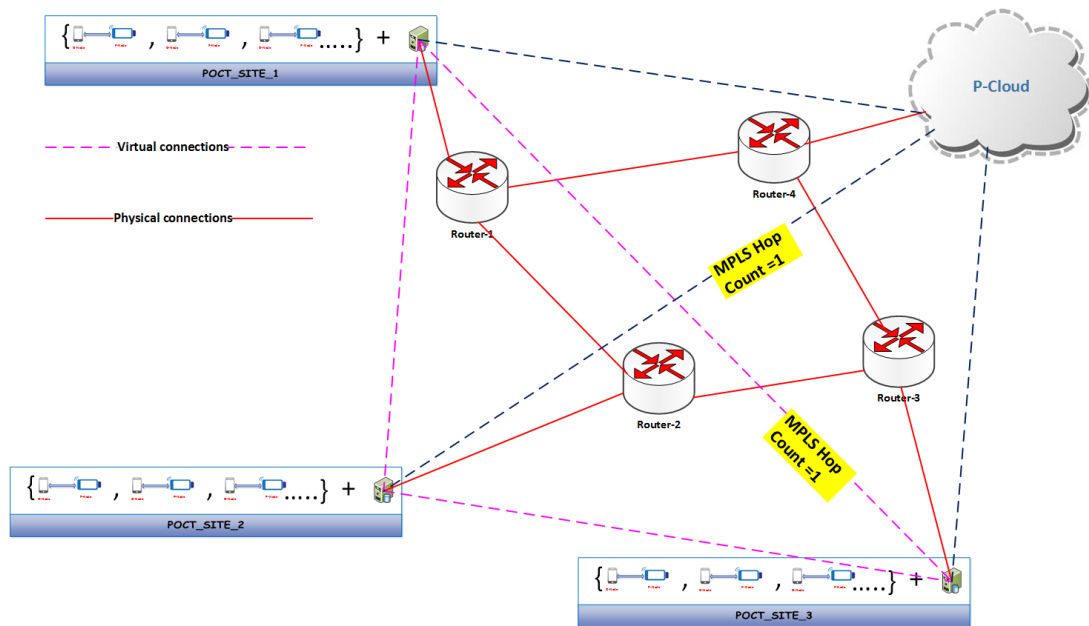


FIGURE 6.19: MPLS based interconnection of POCT sites

will translate into less delay between the site communications.

The routers are assumed to be MPLS enabled devices (i.e., they have MPLS stack in addition to standard TCP / IP protocol stack). With the MPLS enabled, the reachability of the P-Cloud from the POCT sites is one hop.

6.5 Development and maintenance strategy for POCT system

Ongoing development and maintenance are planned to support the installations of the POCT systems. An ecosystem that provides the development and maintenance needs is important to the success of the deployed POCT system operations. Figure 6.20 shows the required ecosystems needed to support the development and maintenance activities after the initial deployment of the POCT sites. The activities include network operator compliance certifications, implementation of new standard rules, upgrading of system components,

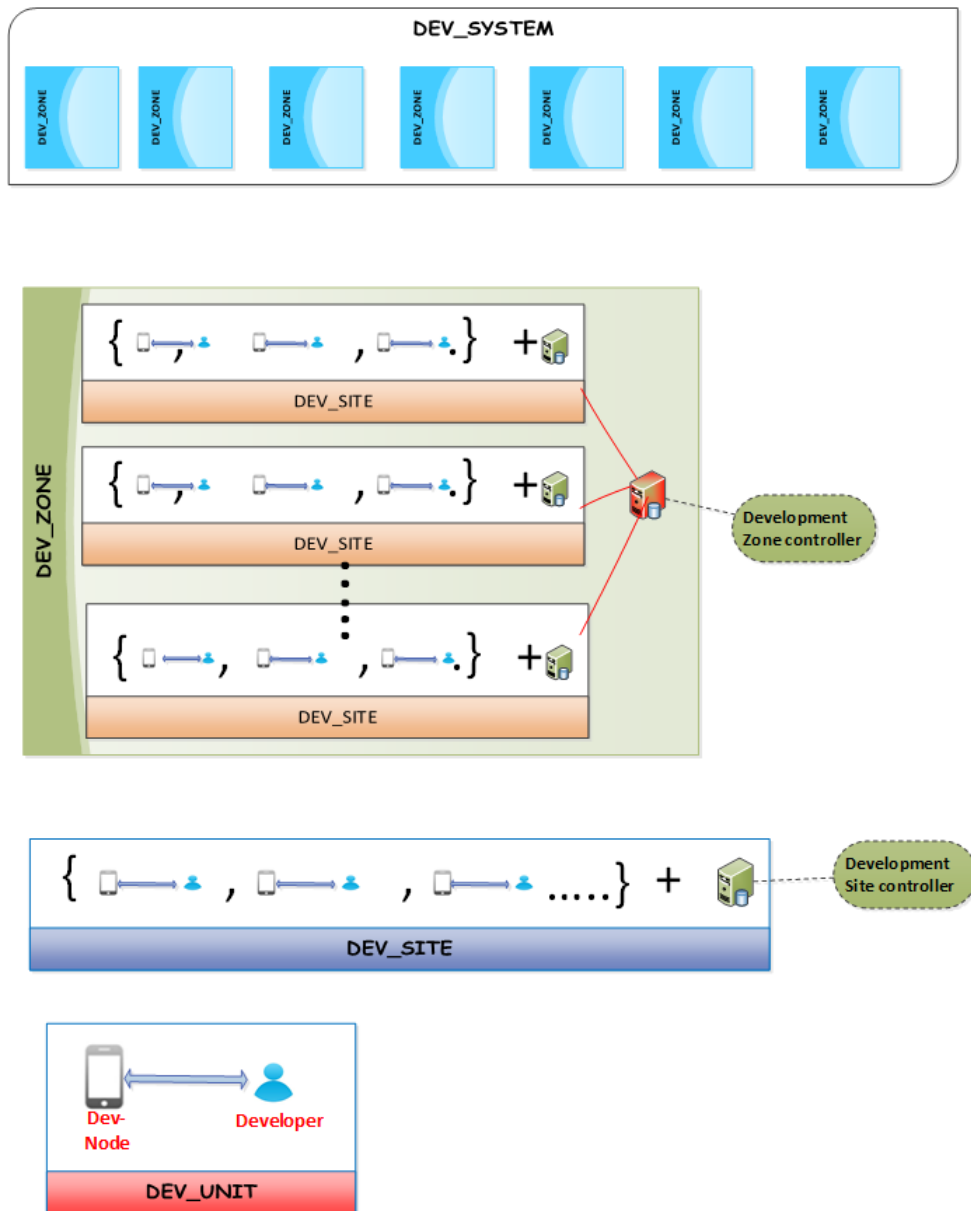


FIGURE 6.20: Development and system maintenance ecosystem

POCT applications development and the introduction of new devices to meet new requirements. Failure to have a supportive ecosystem will impact POCT system operations.

6.5.1 Interconnecting development sites

As mentioned in section 6.4.8, development and maintenance sites can be interconnected via the MPLS connectivity (as shown in Figure

6.19).

6.5.2 Business units needed for POCT development ecosystem

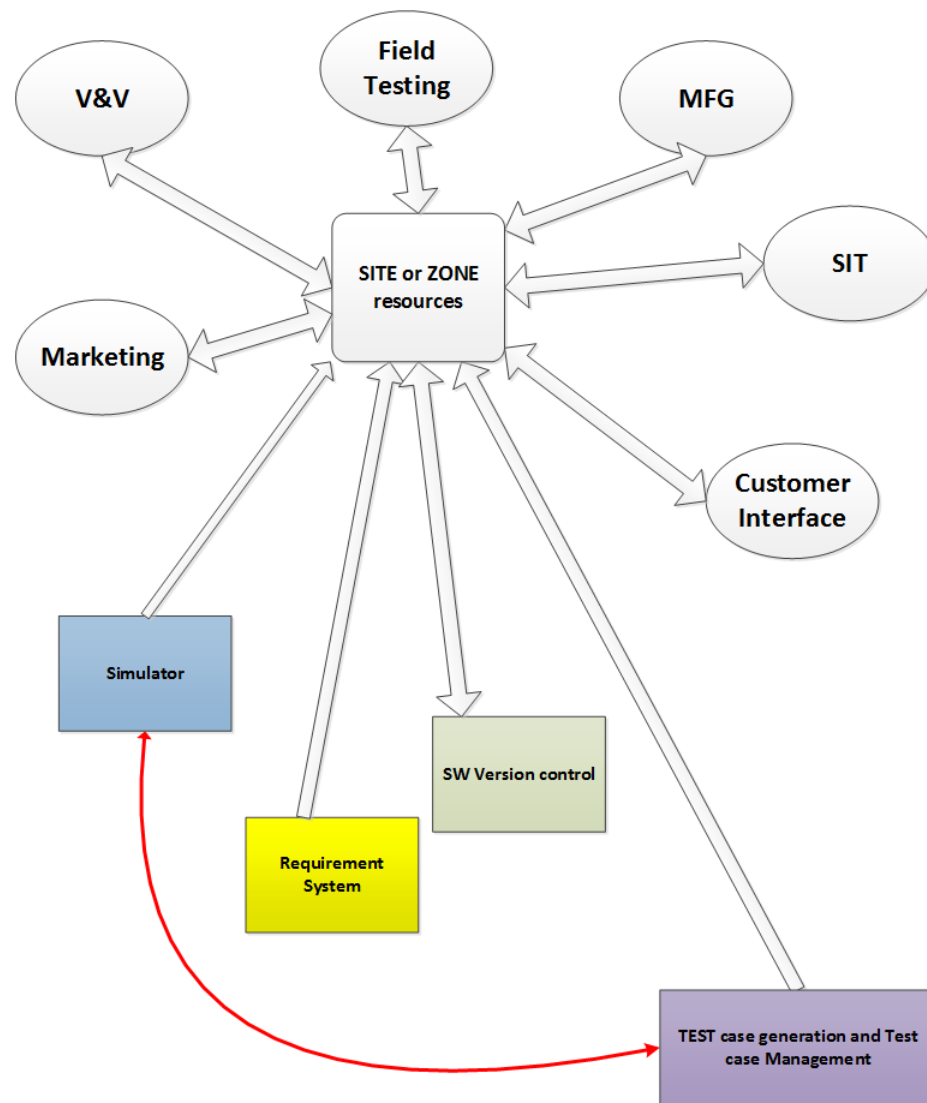


FIGURE 6.21: POCT organization for Business

For business development, testing, and selling the POCT systems, there are few business organizations needed. These business units are shown in Figure 6.21. It starts from the requirement management (system specification), SW / HW development, Validation and Verification (V and V), System Integration Testing (SIT), Manufacturing (MFG), Field testing, Customer care, and Marketing business units.

Also, simulation and modelling, along with test case generation and management, are required to run a successful POCT business.

The requirement system is a database of all the specifications of the developed system, and it is the main driver of any system development activities. The requirement management is usually a part of the system engineering discipline. One of the activities of requirement management is to understand the technical specification requested by various stakeholders of the POCT system, identify any functional gaps and plan development effort to close the gaps. The software version control system maintains all versions of the system deployed and in development. The versions history will reveal functionalities in the working system. This is an important system when dealing with maintenance system releases and in maintaining high-quality systems. Customer interface is an operational business unit where the daily customer interactions are handled. V-and-V is verification and validation unit where the test cases are developed and traced to the system requirements. This is a critical business unit that helps to contain any critical system issues before the product is released. The V-and-V unit plays a role as the main gatekeeper before the system is deployed, with the test case generation unit a part of it. But sometimes it is an independent system modeling unit that develops test scenarios to cover maximum functional coverage (usually 99 percentage of the functionalities), using modelling and simulation methodologies such as combinatorial design techniques.

The marketing business unit researches the future system needs and conducts competitive studies to provide market analysis data for the systems engineering business unit. The manufacturing organization works in collaboration with the development team and produces

POCT devices for the planned deployments. In some cases, the manufacturing is done by a third party organization. The field testing unit is an absolute necessity for understanding the deployment environments, and the field team helps fix any issues that may surface after deployments.

Note that all the business organizations will use the shared resources from the sites and the zones. The site/zone resources include the clinical research community who provide their expertise in the POCT system component development. A dedicated development ecosystem must exist for sustaining the POCT systems deployments.

6.6 Connecting commercially available Cloud services with POCT

The purpose of this section is to provide some interesting configurations for realizing the POCT system using commercially available system components.

6.6.1 With commercial Cloud

Figure 6.22 shows the use of commercially available cloud services such as AWS (Amazon Cloud Services) and a Wi-Fi enabled router connected to the POCT device, which is a very simple approach to transmit test data from the POCT devices to the cloud. The security of the system is accomplished by leveraging the standard security provided by the Internet Service Providers (ISPs) and cloud service providers.

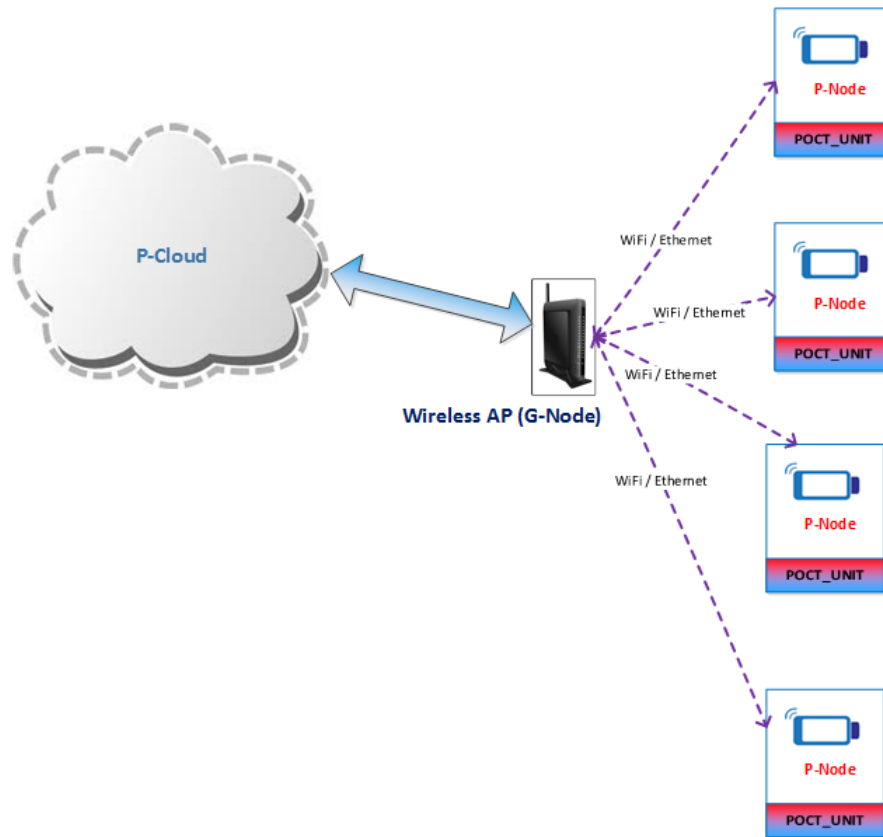


FIGURE 6.22: Commercial Cloud and G-Node

6.6.2 With commercial Cloud and local private Cloud

Figure 6.23 shows a POCT device deployment configuration that is created by the commercial cloud and Network Attached Storage (NAS) as a private cloud for transmitting test data. Because of the NAS, the user has the option to store the data locally as well. Detailed information about the NAS is found in Chapter-5.

6.6.3 Useful configuration of G-Node for connecting to commercial Clouds

The role of the G-Node is to provide a gateway to POCT data. As shown in Figure 6.24, the G-Node is defined as a combination of a wireless Access Point (AP) and a smartphone. There are two sets of data pathways to transmit the test data to the P-Cloud, based

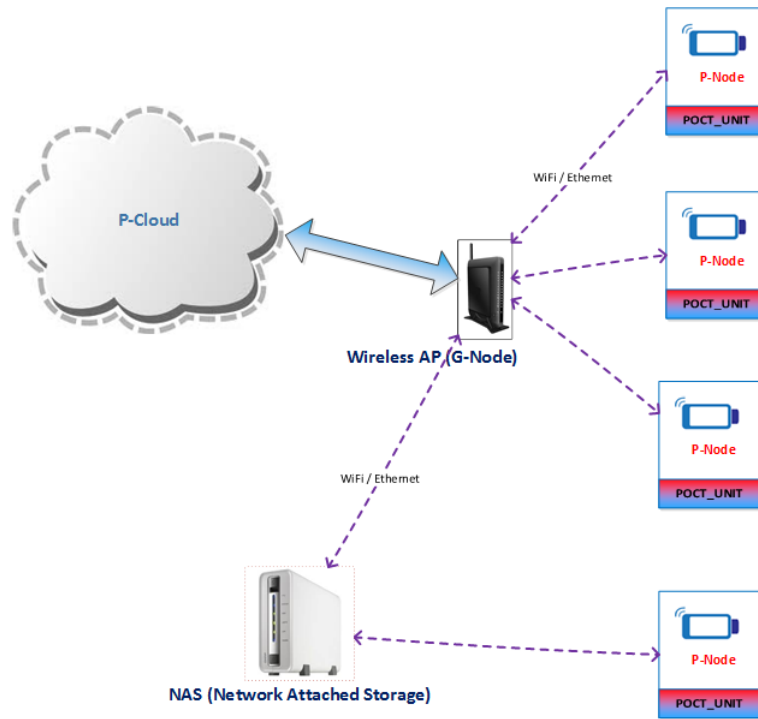


FIGURE 6.23: Commercial cloud and NAS as private cloud

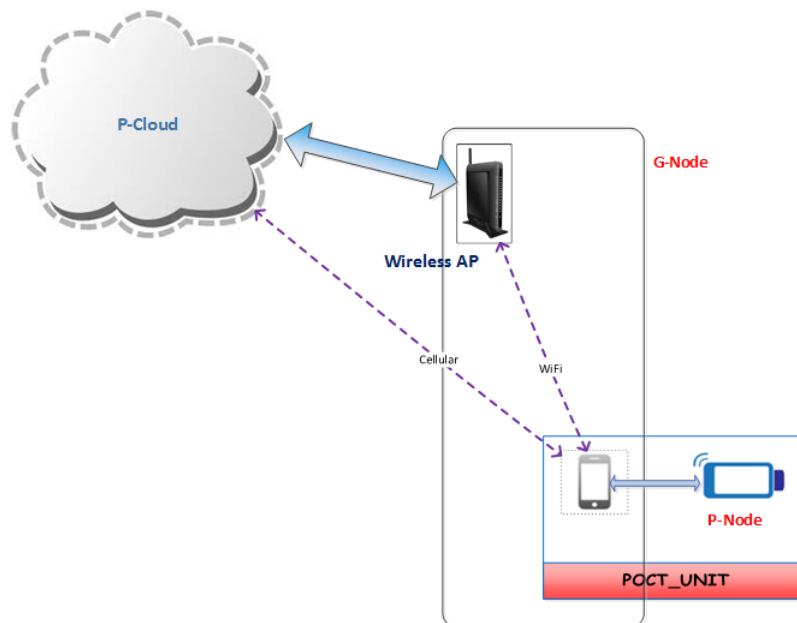


FIGURE 6.24: Commercial cloud and NAS as private cloud

on user preference. The security of the data depends on the Wi-Fi access provided by the AP.

6.7 Simulation of key network models

The common characteristics of the network topologies for the POCT system can be stated as a combination of point-to-point (P2P) and star configurations. The star configurations are considered as multiple P2P connections. The NS3 simulation system is used to model some of the scenarios[145]. The concepts associated with the NS3 are explained[146].

6.7.1 Simulation of Point-to-Point data traffic between P-Node and G-Node

One way of managing the data traffic is to manage the buffer size at the receiver. Buffer bloating [147] is known as a method of increasing the buffer size of the queue to manage data reception. If the buffer is increased to accommodate a queuing space to a maximum allowable possible, the packet drop can be brought down to zero. But the QoS may not be controllable in the state of maximum buffer size.

The simulation tool used was NS3. The NS3 has some basic abstraction concepts related to network simulation [148]. Node is a basic protocol stack entity defined in the NS3. The P-Node and the G-Node are the basics entities in the context of the POCT system.

An application runs on the nodes in the NS3 simulation framework. The UDP packets or TCP packets represent the data that are transmitted and received by the nodes. The POCT application runs over the TCP / IP layer.

The abstraction of the channel provides a means of connecting nodes that are configured in multiple configurations. The purpose of simulating the basic entities in the POCT system, point to point connection (Point-To-Point-Channel) is considered because regardless of the number of P-Nodes, the data is sent to the G-Node via the point-to-point connectivity.

The net device is an abstraction of software driver control which connects the node to the selected channel. The P-Node and the G-Node are modelled as the net devices (Point-To-Point-NetDevice), using the point-to-point connection functionality. Also, the net device represents the layer-2 (data link layer) functionalities in TCP / IP network model[149].

6.7.2 Simulation details

The network topology is simulated as a point-to-point (P2P) connection between P-Node and G-Node. In a standard POCT system, there will be multiple P2P connections. These connections can be categorized as narrowband links (NB-IoT) [150], because of the nature of the POCT system. There will be bursts of data traffic in the links.

By changing the packet size throughput of the point-to-point, connection is recorded. The traffic control layer is a concept in the NS3 in which the TC layer lies between the L2 and the protocol stack [151]. The throughput is defined by the following formula.

$$throughput = \frac{totalPacketsReceived * 8}{simulationTime * 1000000.0} Mbits/sec \quad (6.1)$$

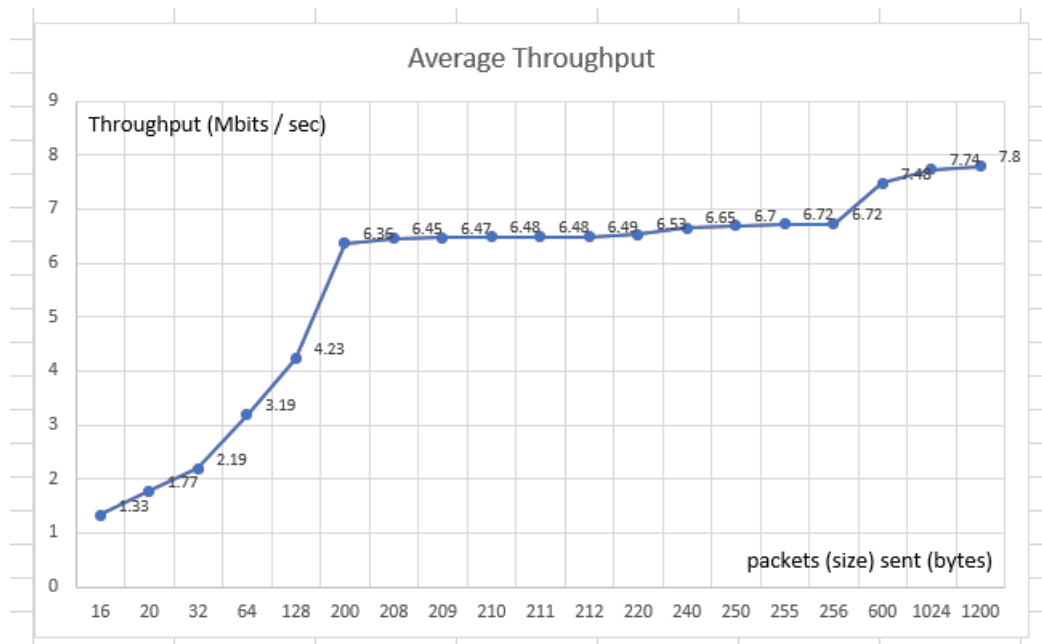


FIGURE 6.25: Throughput plot

The traffic controllers role is to manage the internal packet queues based on the number of packets received. Note that, as shown in Figure 6.25, at 200 bytes onwards, the throughput change is minimal. The trend of minimal change in throughput goes up to 256 bytes packets. Therefore, the desired operating region can be assumed between 200 to 256 bytes. The number of bytes used in POCT measurements can differ based on the assays and processes defined by individual test measurements. There is another throughput increment period starting from 600 bytes packet size. The number of packets depends on type assays used. Some assays will produce more data than others. Accordingly, the transmission of data can be planned based on the throughput plot of the whole system.

Figure 6.26 and Figure 6.27 show the packet loss profile of the G-Node (or the P2P link). Note that the packet size of 200 bytes is the key knee point for getting the stable packet transfer. The knee point is due to packet loss management characteristics of the TC implementation of the NS3. The TC of the G-Node can be developed

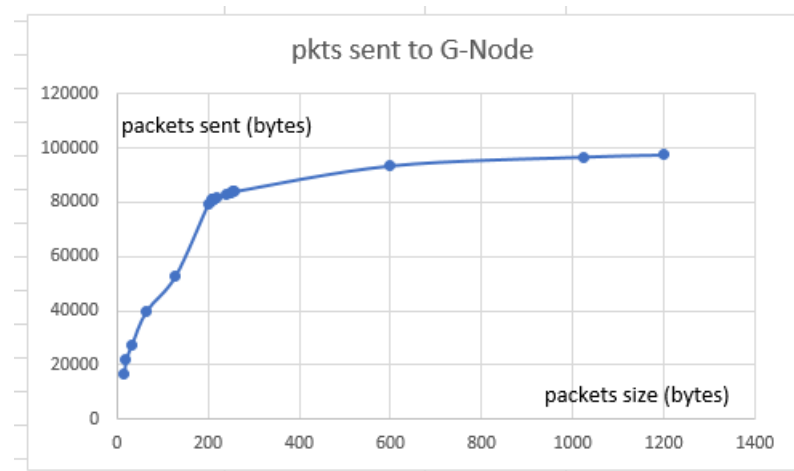


FIGURE 6.26: Packets sent to G-Node

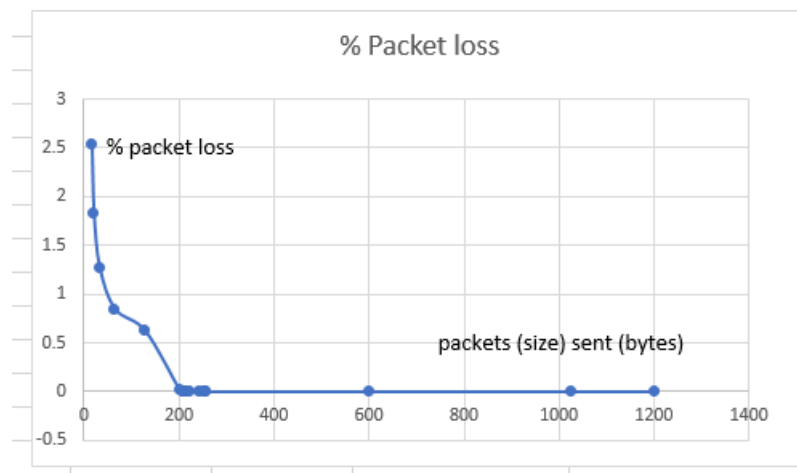


FIGURE 6.27: Packet loss (percentage)

in such a way that the interface (i.e., the interface used by the P-Node P2P connections) achieves the desired knee point reference to the P-Node dataset as configured. Practical application of the simulation is to apply padding so that the data packets can reach the 200 bytes packet size which will provide a constant throughput for a given P-Node.

6.8 Collaborative congestion control management

In the previous section, it was stated that the buffer manipulation plays one of the key roles in managing data link performance. In

this section, a new way of managing data congestion during transmission is explained. The idea is to have sixteen bytes of data as a base for transmitting measurement data by any active P-Node in the system. In a POCT system, there will be multiple P-Nodes with different assays for measurement. The P-Nodes will be configured to handle a different kind of assay. Measurement data capacity required by each P-Node needs to be set per the requirement provided by the clinician. Using the sixteen bytes as the baseline measurement element and multiples of sixteen bytes will be allocated for the P-Nodes, depending on the assays requirements.

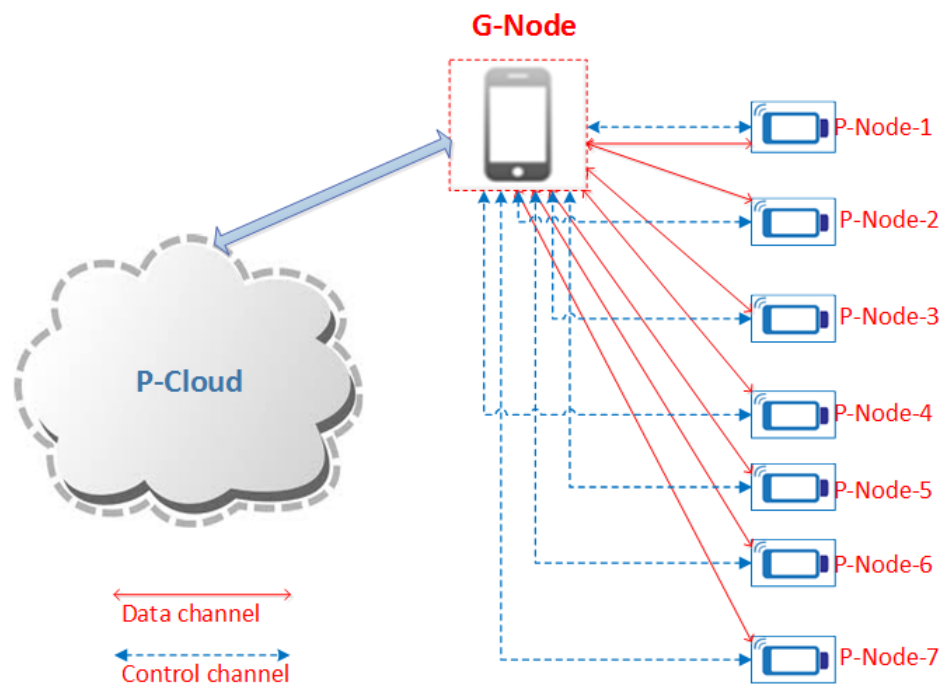


FIGURE 6.28: Data and Control Channels

In Figure 6.28, the network setup for multiple P-Nodes connected to the G-Node is shown. The connection between the G-Node and each P-Node is governed by the point-to-point connectivity. Control channels are shown in addition to the data channels. The control channel is used to having collaborative congestion control among all connected entities. The concept is to inform the P-Nodes regarding

the availability of the data channels between the G-Node and the P-Nodes. Figure 6.29 shows configuration matrix of the P-Node. The configuration matrix control the communication rules of the P-Node. Figure 6.29 is zoomed in version of configuration matrix of Figure 6.30.

Byte Count	TxStpNode-1	TxStpNode-2	TxStpNode-3	TxStpN
1	1	1	1	1
2	1	1	1	1
3	1	1	1	1
4	1	1	1	1
5	1	1	1	1
6	1	1	1	1
7	1	1	1	1
8	1	1	1	1
9	1	1	1	1
10	1	1	1	1
11	1	1	1	1
12	1	1	1	1
13	1	1	1	1
14	2	2	2	2
15	1	1	1	1
16	1	1	1	1
17	0	1	1	1
18	0	1	1	1
19	0	1	1	1
20	0	1	1	1
21	0	1	1	1
22	0	1	1	1
23	0	1	1	1
24	0	1	1	1
25	0	1	1	1
26	0	1	1	1
27	0	1	1	1
28	0	1	1	1
29	0	1	1	1
30	0	2	2	2
31	0	1	1	1
32	0	1	1	1

FIGURE 6.29: P-Cloud configuration in detail (expanded version of Figure 6.30)

As shown in Figure 6.29, *Byte Count* field indicates the number of bytes that can be configured. *TxStpNode-1* field indicates the

measurement capacity for P-Node-1. ***TxStpNode-2*** field indicates the measurement capacity for P-Node-2. The 1s are the bytes available for storing the measurement data. The 1s indicate enabled byte positions. The 0s indicate the disabled byte positions. The 2s indicate the byte positions used for transmission control. The 14th byte is reserved for the transmission status indicator. There are 32 bytes available in P-Node-2. The 14th and 30th bytes are reserved for indicating transmission status (***Transmit end indication data***). The transmission status indicators are marked as 2 in the diagram. Similar logic is applied for configuring other higher capacity measurement nodes.

The Figure 6.31 shows six P-Nodes and a G-Node in the network. Note-1 indicates the configured data capacity for each P-Node. The P-Node-1 is configured to collect test data of sixteen bytes, and the other P-Nodes are configured as 32, 48, 64, 80, and 96 bytes shown in the diagram. Note that the measurement capacity is set in multiples of 16 bytes, as 16 bytes is the basic configuration. These configurations are shown in the configuration matrix (Figure 6.29).

The use of the transmission status indicator is to inform the G-Node of the two bytes before the end of the 16 bytes group transmission. In the case of 16 bytes, G-Node will get an indication when the 14th byte is transmitted. In the case of the 32 bytes node, G-Node will get a transmission status indication of the 14th and the 30th bytes, which is shown in Note-2 in Figure 6.31. Note-3 shows the status byte transmission from the P-Node-6, which has the capacity of 96 bytes. Following the same process, the P-Node-6 will transmit the status byte six times as shown in the Note-3. The G-Node will inform the P-Nodes when it receives a transmission status byte from any P-Nodes, as shown in Figure 6.31, Note-4 and Note-5. The information

sent to the P-Nodes contains a control word as shown in Figure 6.32.

There are six 16 bytes spaces of data storage available in the P-Node-6, which means that the application program can send a transmission status indication six times (byte locations: 14, 30, 46, 62, 78 and 94) to the G-Node. Also, the application can send a single transmission status byte as shown in Figure 6.33.

Thus, the framework structure provides a flexible way of implementing the transmission status byte. The transmission status byte is the byte position at $(N-2)$, where $N = (16, 32, 48, 64, 80, 96 \text{ and so on})$.

In the beginning, the G-Node will create a subscribed nodes list that indicates the type of P-Nodes attached to the G-Node. The subscribed nodes list is formed by a control word or control vector that informs the P-Node of the status of data transmissions from each of the P-Nodes. In addition to sending the control word, G-Node also recommends the channel capacity in terms of the number of P-Nodes. Using these two basic parameters, the P-Nodes that have completed the measurement process can collaborate in using the available channel to transmit the data to the G-Node. The control word will be sent periodically to all the subscribed P-Nodes. The detailed algorithm is shown in Figure 6.34.

ByteCount	TxStpNode-1	TxStpNode-2	TxStpNode-3	TxStpNode-4	TxStpNode-5	TxStpNode-6
1	1	1	1	1	1	1
2	1	1	1	1	1	1
3	1	1	1	1	1	1
4	1	1	1	1	1	1
5	1	1	1	1	1	1
6	1	1	1	1	1	1
7	1	1	1	1	1	1
8	1	1	1	1	1	1
9	1	1	1	1	1	1
10	1	1	1	1	1	1
11	1	1	1	1	1	1
12	1	1	1	1	1	1
13	1	1	1	1	1	1
14	2	2	2	2	2	2
15	1	1	1	1	1	1
16	1	1	1	1	1	1
17	0	1	1	1	1	1
18	0	1	1	1	1	1
19	0	1	1	1	1	1
20	0	1	1	1	1	1
21	0	1	1	1	1	1
22	0	1	1	1	1	1
23	0	1	1	1	1	1
24	0	1	1	1	1	1
25	0	1	1	1	1	1
26	0	1	1	1	1	1
27	0	1	1	1	1	1
28	0	1	1	1	1	1
29	0	1	1	1	1	1
30	0	2	2	2	2	2
31	0	1	1	1	1	1
32	0	1	1	1	1	1
33	0	0	1	1	1	1
34	0	0	1	1	1	1
35	0	0	1	1	1	1
36	0	0	1	1	1	1
37	0	0	1	1	1	1
38	0	0	1	1	1	1
39	0	0	1	1	1	1
40	0	0	1	1	1	1
41	0	0	1	1	1	1
42	0	0	1	1	1	1
43	0	0	1	1	1	1
44	0	0	1	1	1	1
45	0	0	1	1	1	1
46	0	0	2	2	2	2
47	0	0	1	1	1	1
48	0	0	1	1	1	1
49	0	0	0	1	1	1
50	0	0	0	1	1	1
51	0	0	0	1	1	1
52	0	0	0	1	1	1
53	0	0	0	1	1	1
54	0	0	0	1	1	1
55	0	0	0	1	1	1
56	0	0	0	1	1	1
57	0	0	0	1	1	1
58	0	0	0	1	1	1
59	0	0	0	1	1	1
60	0	0	0	1	1	1
61	0	0	0	1	1	1
62	0	0	0	2	2	2
63	0	0	0	1	1	1
64	0	0	0	1	1	1
65	0	0	0	0	1	1
66	0	0	0	0	1	1
67	0	0	0	0	1	1
68	0	0	0	0	1	1
69	0	0	0	0	1	1
70	0	0	0	0	1	1
71	0	0	0	0	1	1
72	0	0	0	0	1	1
73	0	0	0	0	1	1
74	0	0	0	0	1	1
75	0	0	0	0	1	1
76	0	0	0	0	1	1
77	0	0	0	0	1	1
78	0	0	0	0	2	2
79	0	0	0	0	1	1
80	0	0	0	0	1	1
81	0	0	0	0	0	1
82	0	0	0	0	0	1
83	0	0	0	0	0	1
84	0	0	0	0	0	1
85	0	0	0	0	0	1
86	0	0	0	0	0	1
87	0	0	0	0	0	1
88	0	0	0	0	0	1
89	0	0	0	0	0	1
90	0	0	0	0	0	1
91	0	0	0	0	0	1
92	0	0	0	0	0	1
93	0	0	0	0	0	1
94	0	0	0	0	0	2
95	0	0	0	0	0	1
96	0	0	0	0	0	1

FIGURE 6.30: P-Node configuration matrix

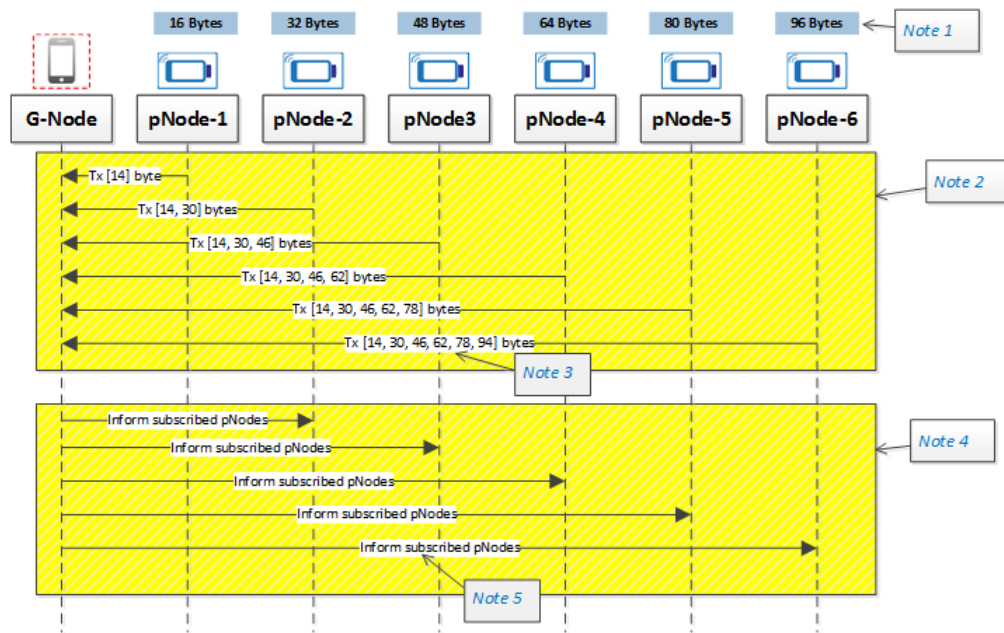


FIGURE 6.31: Message sequence chat for collaborative communication

Subscribed node list:	16	32	64	48	64	96
Transmission Byte received ? (received =1, not received = 0)	1	0	0	1	0	1

FIGURE 6.32: Control Status Word for P-Nodes

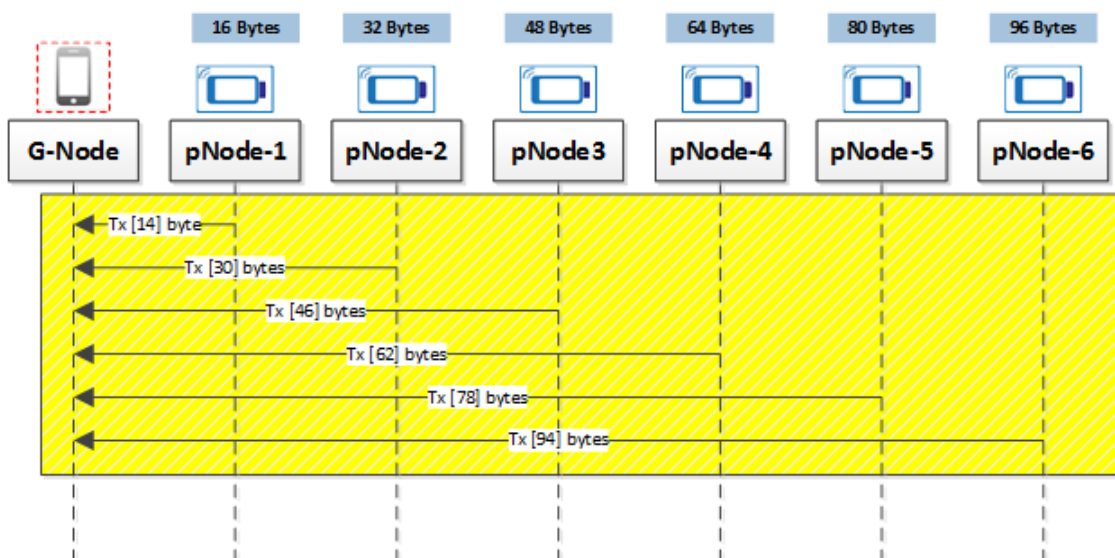


FIGURE 6.33: Sending single transmission indication byte

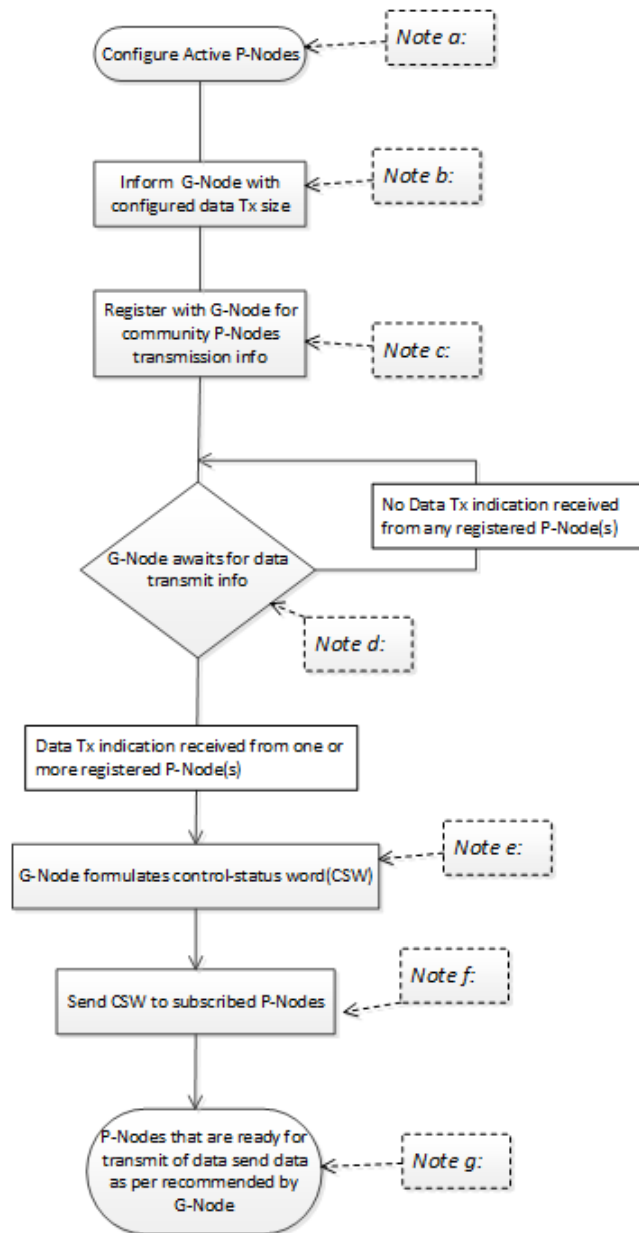


FIGURE 6.34: Flow chart for congestion control

6.8.1 Congestion control algorithm

The algorithm for managing congestion control is shown in Figure 6.34. *Note a:* The P-Nodes are configured based on the assays requirements. The master configuration matrix is shown in Figure 6.29 and Figure 6.30. The data capacity is allocated in chunks of 16 bytes. *Note b:* The configuration matrix is shared with G-Node. *Note c:* The P-Nodes register (or subscribe) for updates transmitted from

G-Node. *Note d:* The G-Node waits for measurement data from the P-Nodes. *Notes e and f:* When a data transmission indication byte is received, the G-Node formulates the control word (an example is shown in Figure 6.33 and transmits to the subscribed P-Nodes. It is expected that all the active P-Nodes will register for receiving GNodes updates. *Note g:* Based on the P-Node configuration matrix and the control word, the G-Node provides a recommendation regarding which of the P-Nodes can attempt to transmit. And accordingly, the P-Nodes will use the data channel to transmit the measured data without any loss. The algorithm explained in this section can be scalable to multiple numbers of P-Nodes.

6.9 Summary

In this chapter, the basic connectivity model is described, which consists of the G-Node, the P-Cloud, and the P-Node. Once the basic connectivity model is identified as the core communication system unit, other possible connectivity types can be developed. They include multiple ways to establish connectivity with the P-Cloud, the local P-Cloud connectivity model, the G-Node as a gateway model, a standalone configuration, and a hybrid configuration, and then the use of well-known secure servers such as Blackberry Enterprise Server and a model using the G-Node as a secure gateway to the P-Cloud. The POCT system definition, based on the connectivity model, was developed as type 1, type 2, and type 3. The type classifications help to define the QoS levels as required by the user community. For ease of deployment of the POCT system, a hierarchical model view was presented. The hierarchical model starts from the basic POCT unit

and the developing POCT sites, POCT zones, and the POCT systems. The hierarchical model view presents the practical approach to linking multiple POCT systems, which include connectivity using the established layer 2.5 protocol MPLS.

A concept of creating an ecosystem for ongoing maintenance and development of the POCT system was developed. The interaction of multiple business units needed in developing the POCT system was developed. Configuration models for connecting with the commercially available cloud were developed by mapping the basic core system model, which consists of P-Node, G-Node, and P-Cloud. Simulation of key network models was carried out, and the results obtained are shown and discussed. An innovative process of managing the network traffic congestion was developed, and the algorithm framework was designed. This framework can be used by application developers to incorporate congestion control in the P-Node application.

Chapter 7

Conclusion and Future work

7.1 Conclusion

This section summarizes the fulfillment of the research objectives established for this project and concludes with a summary and a description of additional research that will extend the solutions suggested by the present project.

The objectives set at the beginning of the research have been accomplished by creating development methodology processes (requirement modelling to an actual working system) that validate the implementation of functionalities for the POCT system. The requirement elicitation process used to identify the functionalities was a non-traditional method of representing the requirements. The pyramid requirements process, along with the use of the Use Case Maps (UCM), was used to identify the communication systems key requirements. This is the only methodology that provides measurable attributes to the requirements, which helps to create the predictable system. The requirements methodology ensures that the key performance indices (KPIs) and required user experiences are met.

For the architecture design, the UCM modelling methodology was used, which links the necessary requirements to the system architecture. The UCM is a powerful system modelling tool with the ability to generate test cases. It provides actual use cases and data flow related to the requirements and helps to provide an end-to-end view of work packages needed for the project, which assists the researcher and the team in providing a realistic plan for project execution.

Building a communication design for securely interconnecting the system building blocks had many challenges: scoping the requirements, choosing the development of the methodology and testing process, and the managing of the end-to-end development. The system design strategy was developed in order to implement the communication system, from the concept to a functional system.

A modularized system architecture was developed to realize the communication of the system. The architecture consists of independent subsystems, developed using open source hardware modules. The modularized architecture allows the system to be developed using in-house components or modules available from a 3rd party vendor, depending upon the need and scope of the device. The key assumption is that the modules provide services via well-defined application-specific interfaces (APIs) for other modules using the services.

Because of the fast pace of changes in the medical world, it is important that the changes in the assay processes are reflected in the system, so that the patient can benefit from the latest changes by using up-to-date software systems. One way to reduce the shipment time or time to market is to have a testing process that finds any critical system software bugs quickly, with a minimum set of smart test vectors. The process of generating the intelligent test vectors is

called Combinatorial Design methodology, which is explained with sample results in Chapter 2.

The standard IEC 62304 helped to classify the key architectural systems to be developed based on the three classes for software safety as defined in the standard. The key accept in the standard is the traceability concept: the requirements, development and testing must be aligned so that any risks can be mitigated. Compliance to the standard has many advantages, including faster approval of certification from FDA and subsequent recertification for software upgrades. For accomplishing IEC 62304 compliance, it is vital that the requirements for developing the system have no ambiguities. The EARS methodology discussed in Chapter 3 provides a solid foundation for the requirements needed to build a complete system. It is recommended that any future development or enhancement requests should start with proper requirements statements based on the EARS methodology and supported by the UCM diagrams. Having the requirements analysis at the beginning of the process will save project costs by 30-40 percentage, and eliminate any risks with the system development.

The wireless communication world is always changing. Yet the communication system for the devices requires great stability, due to the mission critical functionality. With the ongoing discussion of 5G deployment, it is predicted that there will be 1.4B 5G connections by the year 2025 [152]. The actual definition of 5G itself is still evolving. The existing radio access technologies such as LTE (4G) also is still developing. In light of these constant changes in wireless communication technologies, it is important to have the communication protocol developed independently of the radio access technologies. The developed protocol is at the application layer level of the OSI

model. The communication protocol details have been stated in Chapter 3.

Security plays an important role, an essential aspect of the communication system architecture. Several salient operational use cases for securing communication between multiple system components in the context of system communication were presented. Data integrity must be maintained at all times: accessing the device by the authorized user, operating the tests process by the authorized user, accruing measurement data during the tests, data transmission after the test and data storage in the P-Cloud. Chapter 4 outlines two kinds of security methodologies: challenge and response-based and behavioural-based. The challenge and response-based methodology uses the OPCODEs defined in the communication protocol (Chapter 3) to establish the authentication and authorization access of the device. The behavioural-based methodology makes use of secondary attributes or metadata of the measured data. Both the methods can be used in accordance with an encryption scheme commercially available or proprietary schemes. The POCT systems are characterized as Machine-to-Machine (M2M) communication systems. Providing security in the context of the M2M is complex because multiple stakeholders are participating. All the stakeholders, MODEM (communication chip) vendors, device makers, network providers, application developers and device users must be coordinated for the successful operation of system deployment.

A P-Cloud architecture was developed with multiple types of segregated data that are relevant to the communication system. The architecture is agnostic to multiple vendor cloud technologies. Using the NAS technology is one way of constructing a private P-Cloud. It

needs to be designed to meet the HIPAA conformance for data privacy. The multiple types of data in each class are handled separately in the proposed architecture. Managing QoS of the data was shown in the context of the NAS, to accomplish bandwidth requirements. The details are in Chapter 5.

Chapter 6 explains network models for the communication system. The possibilities of connectivity configurations were developed. There are three types of communities of system configurations designed to accomplish QoS SLAs. All the three types of configuration connect to the P-Cloud. A hierarchical system was designed for controlling information flow based on network policies. Concepts of the unit, sites, zones and system were created to manage the data privacy and related policies. Development and system maintenance strategy was formulated to manage large-scale deployments. Simulation of the network connectivity was carried out using the NS3 simulator. A new congestion control algorithm for managing data transmission was developed.

Agile project management methodology was used to manage the ongoing system development with multiple stakeholders. This provides the faster adaptability for meeting the requirements in the system, as POCT is becoming a standard of care in many diagnostic situations when the patient is unable to get treatment on time due to inaccessibility of labs in their locations.

Designing of the POCT requires a different approach than a usual system development methodology, as it is a mission critical communication system. A new development process using the IEC62304 standard has been established, which includes: requirement eliciting, system partitioning, software coding process and a smart way of

testing the system. Following these processes results in an error-free system. Unambiguous system requirements are vital in managing the development of the POCT communication system. A new approach to combining the UCM and the EARS methodologies for understanding the system specification was developed successfully. This approach captures the intention of the specifications for multiple stakeholders: project managers, implementers and system verification and validation teams, in terms of easily understandable artefacts. The communication protocol developed is independent of the wireless radio access technologies. It is scalable for use with modification to existing assays and future assay processing.

Security of data transmission is a critical need for safeguarding the patients test data. The security of the POCT devices communication needed a threat modelling process to develop a suitable security mechanism in the context of the POCT system. Multiple stakeholders are involved in a practical system deployment scenario: device users, network vendors, wireless operators, application developers and device manufacturers. Coordination of all the key parties is needed to build a secure communication system. Two new security mechanisms were developed to meet the challenges of complex security needs. The secure data storage for the patient data is a necessary element in the system. The storage needs to be used for multiple purposes: diagnosing the patient condition, data analytics, device status, system deployment and device operation. A new way of creating cloud storage was developed. Also, there is the need to have a private cloud system with low cost components. The NAS was developed to demonstrate the private cloud application.

Deploying multiple POCT systems is crucial to the success of reaching out to the beneficiaries of the system, such as patients and health-care providers. The system must be scalable as the demand grows. Therefore, a scalable communication system architecture is needed that is capable of configurable QoS attributes at various system usage points. A hierarchical model interconnecting network architectures was developed to meet the needs of the cost-effective system with scalability. There is no limitation to scale a basic system unit to the large-scale system using the hierarchical model developed. This method also helps to analyze communication interface compatibilities using simple hexogen diagrams. Congestion control is one of the key features during system expansion. Congestion control algorithms were developed for use when multiple POCT devices attempt to transfer test data to the P-Cloud.

The new processes and systems developed during this research project have an impact on key ideas presented: technology independent communication protocol, system requirement gathering, system testing, secure communication links specific to the POCT context, securely partitioned data storage (P-Cloud), unlimited scalable network communication model, and congestion control.

7.2 Future Work

There are multiple applications, challenges and opportunities in the area of POCT communication which are explained in the following section. These items form the future work area for system communication and data security.

7.2.1 Data aggregation of multiple P-Clouds

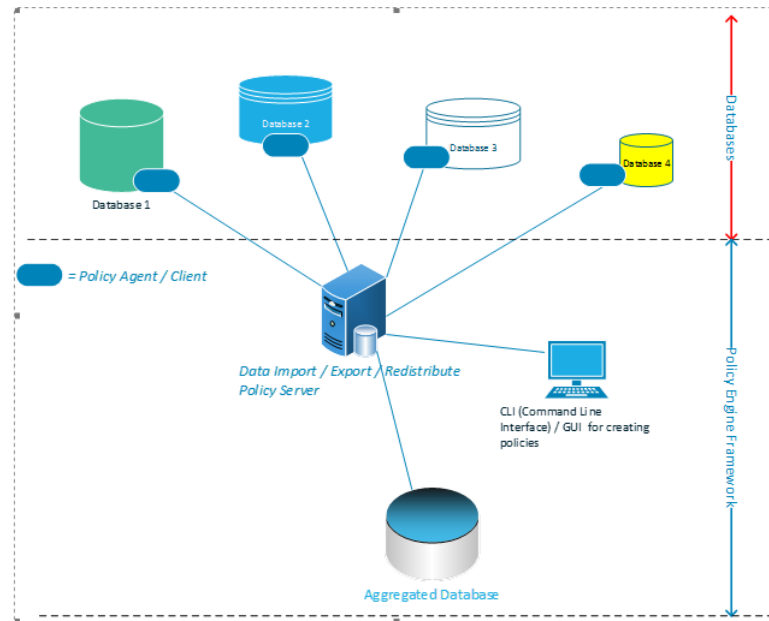


FIGURE 7.1: Policy framework engine

As shown in Figure 7.1, the aggregation of P-Clouds in multiple sites is beneficial in analyzing the collected data for building an understanding of the spread of certain health conditions, including improving the design of future systems, using machine learning-based data analytics.

Consider a scenario that is depicted in Chapter 5, where the P-Cloud was realized using the NAS unit. It is located within the firewall as a private cloud. Multiple system sites will have multiple P-Clouds that need to be interconnected to have aggregated information, as shown in Figure 7.1. For example, if there are four independent databases from four different agencies, they will face the issue of linking some or all the attributes from each of the four databases. A simple solution would be to create a policy engine framework that helps to aggregate data attributes from the four databases.

A policy engine framework is shown in Figure 7.1, which may be a possible strategy for solving data aggregation from multiple data sources. The policy engine framework consists of a policy server, a user interface for creating and modifying policies and an aggregated database. The aggregated database can be developed as a Hadoop container (big data) where the big data search engine algorithms can be applied to generate stories around the data from the four data sources.

The goal is to get the desired data attributes from the four databases into a common aggregated database. The policy server controls information flow (i.e., only the attributes that are allowed by configured policies at interface level) from the databases to the aggregated database. Each database is associated with a policy agent or a policy client that consults the policy server before the data is transmitted to the aggregated database. The policy agents get updated policy information from the policy server, based on the policies configured. Note that the policy server is a centralized entity in the architecture. Investigation of suitable policy agents or some other methods can be taken up as an extension of this work.

7.2.2 Architecture for POCT Mobile station unit and data privacy

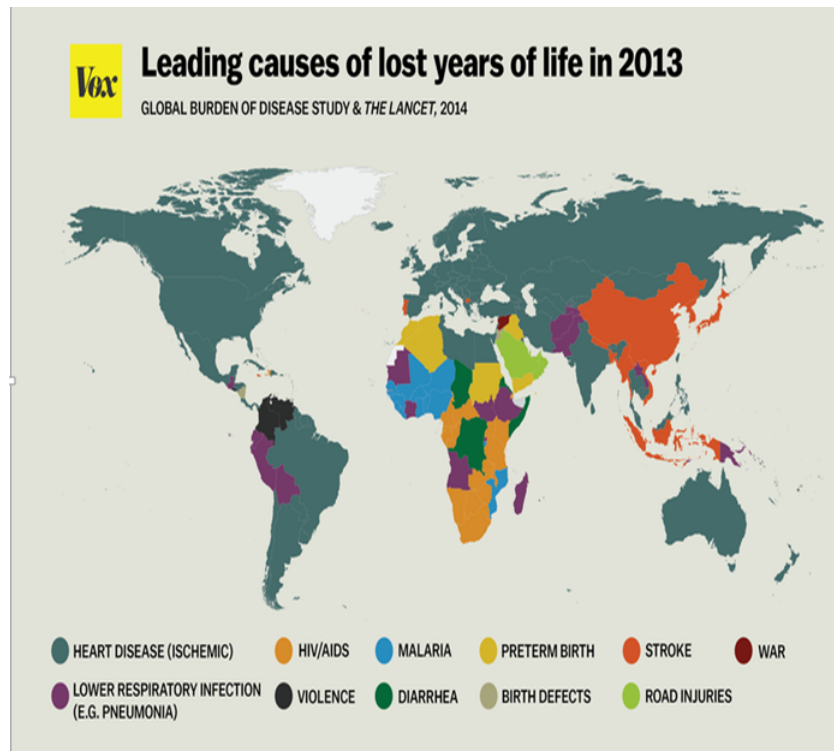


FIGURE 7.2: Lancet-2014 report

The Lancet (2014) has published a study on the leading causes of lost years of life, worldwide. According to the study, the primary reason for the loss of life in African countries is HIV (AIDS) and malaria diseases [153]. The World Health Organization reports that almost 45.8 percentage of member states have less than one doctor per 1000 population [154]. Most of the member states are in African countries (Figure 7.2). Given the situation, testing offers an efficient path to reach a higher doctor-patient percentage population.

The deployment of the POCT system in countries such as those in Africa helps to provide efficient healthcare services to compact loss of lives due to spreading diseases. The POCT system is a data-centric system, and it needs Internet connectivity (IPv4 / IPv6 connectivity) for sending test data to a secure cloud or smartphone. The IPv4 / IPv6 depends on bearer services (rules and policies for providing

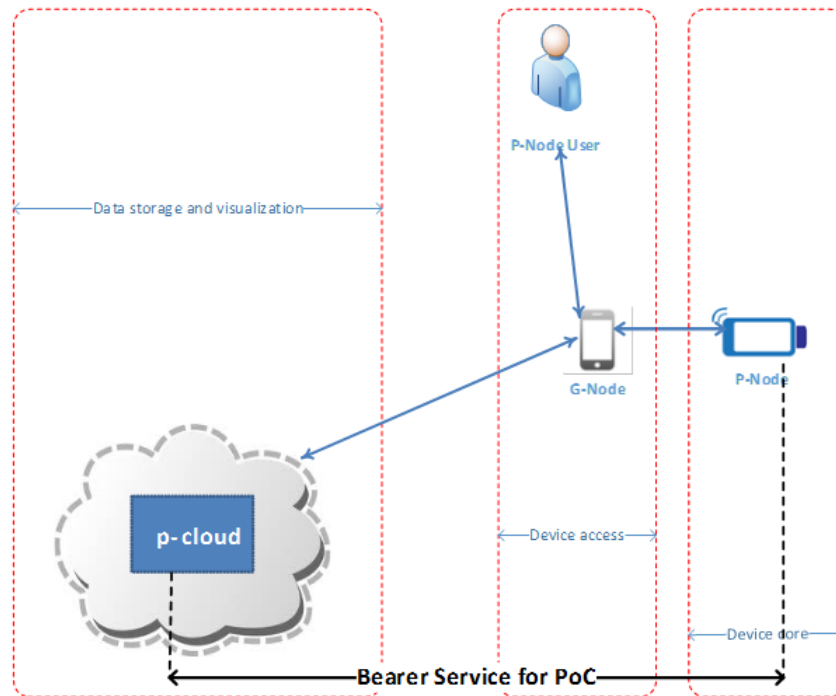


FIGURE 7.3: Bearer Services for POCT

testing service, including QoS attributes), which are needed for the system deployments. The bearer has an endpoint IP address (Figure 7.3). The system deployment operation assumes that all the installations support private IP addresses. This is one of the minimum requirements needed for system deployment. Lack of IP addresses will create practical IP address sharing issues which will prevent successful deployments.

The IPv4 address space is insufficient in African countries (e.g., In Nigeria, 80 people share one IP address, and in Congo, 92 people share one IPv4 address) (Figure 7.4) [155]. The address spaces have been consumed by the growth in mobile data traffic, deployment of data session-hungry applications consuming multiple IP connectivities and growth in M2M communication.

These factors contribute to the limitation issue with private IPv4

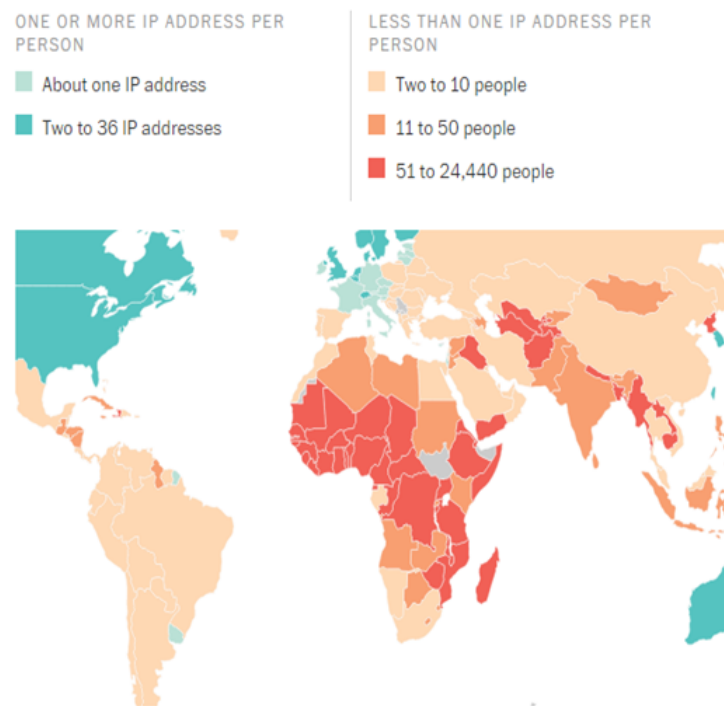


FIGURE 7.4: IP address usage in African countries

addresses. It will take few years to deploy IPv6-based services. Management of coexistence between IPv4 and IPv6-based services can be a challenge, too, because of the network carrier infrastructure configurations. Therefore an intermittent solution, which is scalable and coexists with future IPv6 deployments, is needed. Network address translation (NAT)-based solutions may be considered to manage the IP address space crisis, which helps multiple users sharing one single public IPv4 address (Figure 7.5).

Future research may consider modifications to the deployment scenarios modelled in Chapter 6 and study the impact due to the solutions such as NAT. A Deep Learning network model could be the next step in the process of solving the challenges such as lack of an IP address.

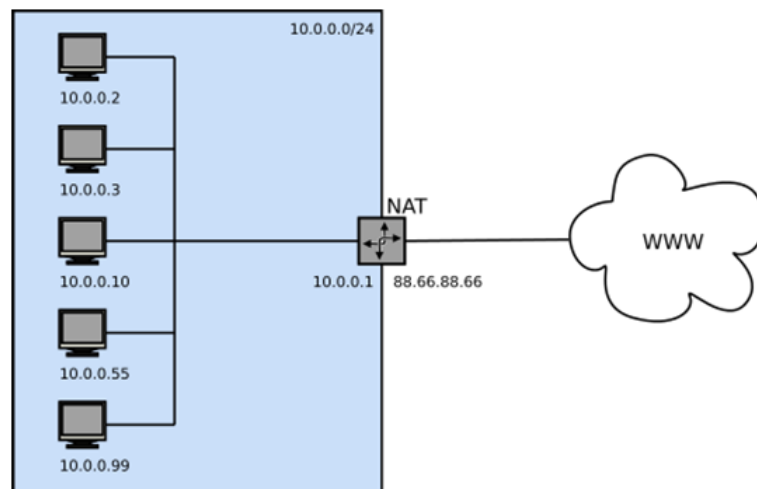


FIGURE 7.5: NAT Based Solution

7.2.3 Modification and expanding POCT protocol with USSD

Unstructured Supplementary Service Data (USSD) technology is a set of functional codes which are used by the GSM phones[156]. The USSD can be adopted for many services that are offered by the network providers. The POCT can be treated as a service provided by the network operator in setting up a secure connection from the device via the G-Node to the cloud server. The protocol described in Chapter 3 could be modified to incorporate the USSD for communication at the application layer.

7.2.4 Developing common data standard for multi-vendor P-Clouds

There are three primary vendors for the cloud platform today, Amazon, Microsoft, and Google. A cloud architecture with data segregation can be implemented using the commercially available cloud technologies. There are propriety cloud infrastructures used across the industry. It is essential to develop a standard data format for having end-to-end data usage and interoperability. Use of well-known

industry standard cloud infrastructures provide trusted and secure P-Clouds. The AI-based algorithms may be developed to find correlations between the multi-vendors cloud deployments for interfacing the system with all the cloud vendors seamlessly.

7.2.5 Security algorithms for dynamic adaptation

The security algorithm models described in the research may have weaknesses in compacting the security threats surfacing almost daily in the cybersecurity world. It is possible to develop a dynamically evolving threat model that shows the weakness against the current security mechanisms and models implemented. Investigating the reuse of other known security countermeasures in such a way that mitigation of any new threats can be managed with the dynamic threat model is a topic for future research.

The security mechanism developed needs to be tested with a full set of devices, and the existing thread model can be evaluated with new security information. The impacts due to the recent findings of security violations within the processing units such as Spectre Attacks (Exploiting Speculative Execution [157]) are to be investigated. Any dependency on the low layers of the communication protocol stack could be considered.

7.2.6 Connected homes and the POCT systems

Interfacing of the POCT system with connected home technologies consisting of an external 5G communication backhaul with an internal Internet router connectivity could be investigated. The POCT can be a special application in these environments with special performance indices. This may impact the network models analyzed in

the research. This investigation would be a value-added project for all stakeholders.

The research has come up with a framework uses the communication protocol which is independent of radio access technology (RAT), meaning the communication protocol will work with future RAT technologies. The research came out processes and methodologies: creating meaningful requirements to solve the problems, system design methodology for building loosely coupled systems, security principles needed for data privacy, applying the IEC62304 for system partitioning, the combinatorial methods for a smart way of testing the system and using Agile project development processes. These are significant outcomes in the methodology and procedures for developing the POCT device communication but also other medical devices in general. Expressing the requirements without any ambiguity is vital to have an error-free system. The new methodology which combines the UCM model and the EARS requirement syntax provides a unique way of expressing the requirements for mission critical system not only suitable for POCT development but also applicable to other mission critical system development in medicine. The communication protocol that evolved from the research is ideal for medical devices which need to execute multiple assays or procedures. The cloud architecture conforms to HIPPA compliance, and the medical systems that need to store medical data can adopt this architecture. The existing commercially available cloud systems can be configured to model the architecture developed. The hierarchical network system designed in the research can be used for managing all kind of IoT systems and medical devices with limitless types of communication technologies.

This research aims to impact the monitoring and early diagnosis

of infectious diseases in low-resource, low-income developing countries. This critical issue depends on the collaborative efforts of many, healthcare specialists, technologists, physicians, and others who are committed to improving healthcare delivery in low-resource, low-income developing countries. This researchers desire is that this project provides forward momentum in the development of a health delivery system that is capable of executing the assays in distant locations to detect infectious diseases.

Bibliography

- [1] Brunel DOC LAB. URL <http://bruneldoclab.com/>.
- [2] Naeem M Zhu W Memon A Khalid A. Using v-model methodology, uml process-based risk assessment of software and visualization. *Proceedings of 2014 International Conference on Cloud Computing and Internet of Things*, (1), January 2014. URL ieeexplore.ieee.org/document/7062536/.
- [3] Discovering Real Business Requirements for Software Project Success. URL <https://dl.acm.org/citation.cfm?id=983756>.
- [4] Pairwiser - Pairwise Testing and Test Generation Tool. URL <https://inductive.no/pairwiser/>.
- [5] Alistair Mavin, Philip Wilkinson, Adrian Harwood, and Mark Novak. Easy Approach to Requirements Syntax (EARS). In *2009 17th IEEE International Requirements Engineering Conference*, pages 317–322. IEEE, 8 2009. ISBN 978-0-7695-3761-0. doi: 10.1109/RE.2009.9. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5328509>.
- [6] Rosanna W Peeling and Ruth McNerney. Emerging technologies in point-of-care molecular diagnostics for resource-limited settings. *Expert Review of Molecular Diagnostics*, 14(5):525–534, 6 2014. ISSN

- 1473-7159. doi: 10.1586/14737159.2014.915748. URL <http://www.ncbi.nlm.nih.gov/pubmed/24784765><http://www.tandfonline.com/doi/full/10.1586/14737159.2014.915748>.
- [7] A. N. Abou Tayoun, P. R. Burchard, I. Malik, A. Scherer, and G. J. Tsongalis. Democratizing Molecular Diagnostics for the Developing World. *American Journal of Clinical Pathology*, 141(1):17–24, 1 2014. ISSN 0002-9173. doi: 10.1309/AJCPA1L4KPXBJNPG. URL <https://academic.oup.com/ajcp/article-lookup/doi/10.1309/AJCPA1L4KPXBJNPG>.
- [8] Manoharanehru Branavan, Ruth E. Mackay, Pascal Craw, Angel Naveenathayalan, Jeremy C. Ahern, Tulasi Sivanesan, Chris Hudson, Thomas Stead, Jessica Kremer, Neha Garg, Mark Baker, Syed T. Sadiq, and Wamadeva Balachandran. Modular development of a prototype point of care molecular diagnostic platform for sexually transmitted infections. *Medical Engineering and Physics*, 38(8):741–748, 8 2016. ISSN 18734030. doi: 10.1016/j.medengphy.2016.04.022. URL <http://linkinghub.elsevier.com/retrieve/pii/S1350453316300789>.
- [9] Gabrielle Tiven, Lisa Frist, Michael Chang, and Yechiel Engelhard. Management for the World Module: Assessing point-of-care diagnostics for resource-limited settings. 2011.
- [10] 5 Things You Need To Know About The Point-Of-Care Technology Market. URL <http://www.meddeviceonline.com/doc/things-you-need-to-know-about-the-point-of-care-technology-ma>
- [11] NHS Choices. NHS Choices Home Page.

- [12] Meryl Brod, Betsy Pohlman, Michael Wolden, and Torsten Christensen. Non-severe nocturnal hypoglycemic events: experience and impacts on patient functioning and well-being. *Quality of Life Research*, 22(5):997–1004, 6 2013. ISSN 0962-9343. doi: 10.1007/s11136-012-0234-3. URL <http://link.springer.com/10.1007/s11136-012-0234-3>.
- [13] Home - ACON Labs, . URL <https://www.aconlabs.com/>.
- [14] Ascensia Diabetes Care Holdings AG: Private Company Information - Bloomberg. URL <https://www.bloomberg.com/research/stocks/private/snapshot.asp?privcapId=303565574>.
- [15] BBI Healthcare. URL <http://www.bbihealthcare.com/>.
- [16] Cambridge Sensors Limited: Private Company Information - Bloomberg. URL <https://www.bloomberg.com/research/stocks/private/snapshot.asp?privcapid=115036415>.
- [17] Solutions - HemoCue. URL <https://www.hemocue.us/en-us/solutions/>.
- [18] Home — LifeScan, Inc., . URL <http://www.lifescan.com/home>.
- [19] Medical Technology, Services, and Solutions Global Leader — Medtronic. URL <http://www.medtronic.com/us-en/index.html>.
- [20] Features, . URL <https://www.menarinidiagnostics.com/en-us/Home/Laboratory-products/Point-of-Care-Testing/V-Sight/Features>.

- [21] Features, . URL <https://www.menarinidiagnostics.com/en-us/Home/Laboratory-products/Point-of-Care-Testing/BC-5300-Vet/Features>.
- [22] Nova Biomedical Corporation: Private Company Information - Bloomberg. URL <https://www.bloomberg.com/research/stocks/private/snapshot.asp?privcapId=836724>.
- [23] Overview - Point of Care Testing — Roche Diagnostics USA. URL <https://usdiagnostics.roche.com/en/point-of-care-testing/overview.html>.
- [24] Products available from Point Of Care Testing Ltd (POCT Ltd). URL <http://www.poct.co.uk/products.cfm>.
- [25] Christoph Gerber. Introduction into IEC 62304 Software life cycle for medical devices. (September), 2008.
- [26] Sivanesan Tulasidas, Sivanesan Tulasidas, Ruth Mackay, Pascal Craw, Chris Hudson, Voula Gkatzidou, and Wamadeva Balachandran. 1 Process of Designing Robust, Dependable, Safe and Secure Software for Medical Devices: Point of Care Testing Device as a Case Study. URL <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.426.7039>.
- [27] Pascal Craw, Ruth Mackay, Angel Naveenathayalan, Chris Hudson, Manoharanehru Branavan, S. Sadiq, Wamadeva Balachandran, Pascal Craw, Ruth E. Mackay, Angel Naveenathayalan, Chris Hudson, Manoharanehru Branavan, S. Tariq Sadiq, and Wamadeva Balachandran. A Simple, Low-Cost Platform for Real-Time Isothermal Nucleic Acid Amplification. *Sensors*, 15(9):23418–23430, 9 2015. ISSN 1424-8220. doi: 10.3390/s150923418. URL <http://www.mdpi.com/1424-8220/15/9/23418>.

- [28] HIPAA Security Requirements. URL <http://www.hipaasurvivalguide.com/hipaa-regulations/164-312.php>.
- [29] <https://bruneldoclab.com/>. URL <https://bruneldoclab.com/>.
- [30] Simmons G.b Wong J.ac McDevitt J.T McRae, M.P.a. Programmable Bio-nanochip Platform: A Point-of-Care Biosensor System with the Capacity to Learn. *American Chemical Society*, 49(7):1359–1368, 2016. doi: 10.1021/acs.accounts.6b00112.
- [31] Muhammad Rashid Naeem, Weihua Zhu, Adeel Akbar Memon, and Adeel Khalid. Using V-Model methodology, UML process-based risk assessment of software and visualization. In *Proceedings of 2014 International Conference on Cloud Computing and Internet of Things*, pages 197–202. IEEE, 12 2014. ISBN 978-1-4799-4764-5. doi: 10.1109/CCIOT.2014.7062536. URL <http://ieeexplore.ieee.org/document/7062536/>.
- [32] Sivanesan Tulasidas, Ruth Mackay, Pascal Craw, Chris Hudson, Voula Gkatzidou, and Wamadeva Balachandran. Process of Designing Robust, Dependable, Safe and Secure Software for Medical Devices: Point of Care Testing Device as a Case Study. *Journal of Software Engineering and Applications*, 06(09):1–13, 8 2013. ISSN 1945-3116. doi: 10.4236/jsea.2013.69A001. URL <http://www.scirp.org/journal/PaperInformation.aspx?PaperID=36489&#abstract>.
- [33] Sivanesan Tulasidas, Josef Hausner, John Terzakis, Fred Love, Stefan Mattern, Chris Hudson, Nada Manivannan, and Ruth Mackay. Requirements for Point of Care Devices using Use Case

- Maps. *International Journal on Recent and Innovation Trends in Computing and Communication*, 3(6):4284–4288, 2015. URL <http://www.ijritcc.org/download/1436848521.pdf>.
- [34] Sivanesan Tulasidas, Ruth Mackay, Chris Hudson, and Wamadeva Balachandran. Security Framework for Managing Data Security within Point of Care Tests Security Framework for Managing Data Security within Point of Care. *Journal of Software Engineering and Applications*, 10(10):174–193, 2 2017. ISSN 1945-3116. doi: 10.4236/jsea.2017.102011. URL <http://www.scirp.org/journal/jsea%5Cnhttps://doi.org/10.4236/jsea.2017.102011>.
- [35] Daniel Amyot. Use Case Maps Quick Tutorial. See <http://www.usecasemaps.org/pub/UCMtutorial/UCMtutorial.pdf>, (September), 1999. URL <http://cserg.site.uottawa.ca/ucm/pub/UCM/VirLibTutorial99/UCMtutorial.pdf>.
- [36] Bridging the Requirements/Design Gap in Dynamic Systems with Use Case Maps (UCMs). URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.29.1221&rep=rep1&type=pdf>.
- [37] jUCMNav Tutorial 1: Creating a simple path, components, stubs and plug-in maps. URL <http://www.youtube.com/watch?v=kuXvxmcfzh8>.
- [38] IEC 62304 Ed. 1.0 b:2006 Medical device software - Software life cycle processes. URL http://webstore.ansi.org/RecordDetail.aspx?sku=IEC%2062304%20Ed.%201.0%20b:2006&source=google&adgroup=iec&gclid=CjwKEAiA68WnBRCJxZr5qoaL3iMSJAAXIrr3_muvjitFt8NGj6Lk8EEFigpeQCZLSfQVL1IoI44s5hoCShfw_wcB.

- [39] IC - Wikipedia, the free encyclopedia. URL <http://en.wikipedia.org/wiki/I%C2%B2C>.
- [40] Daniel Amyot and Gunter Mussbacher. Minitutorial - UCM Minitutorial - UCM Table of Contents to Use Case Maps (UCMs) and the UCM Notation Minitutorial - UCM Minitutorial - UCM Use Case Maps Web Page. 2001.
- [41] *Standard Practice for System Safety*. U.S. Department of Defense, 1998. URL https://assist.daps.dla.mil/quicksearch/basic_profile.cfm?ident_number=36027.
- [42] The Use and Misuse of FMEA in Risk Analysis — MDDI Medical Device and Diagnostic Industry News Products and Suppliers. URL <http://www.mddionline.com/article/use-and-misuse-fmea-risk-analysis>.
- [43] Zhenping Hu, Riikka Susitaival, Zhuo Chen, I-Kang Fu, Pranav Dayal, and Sudhir Baghel. Interference avoidance for in-device coexistence in 3GPP LTE-advanced: challenges and solutions. *IEEE Communications Magazine*, 50(11):60–67, 11 2012. ISSN 0163-6804. doi: 10.1109/MCOM.2012.6353683. URL <http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6353683>.
- [44] Craig Kuziemsky Xia Liu, Liam Peyton, University of Ottawa, Canada. 550 Cumberland St. Ottawa, Ontario, and Kuziemsky@telfer.uottawa.ca xliu044@uottawa.ca, lpeyton@site.uottawa.ca. A Requirement Engineering Framework for Electronic Data Sharing of Health Care Data between Organizations. URL <http://lotos.site.uottawa.ca/ucm/pub/UCM/VirLibMceTech09EDS/MCETECH2009-EDS.pdf>.

- [45] Peter Karpati, Guttorm Sindre, and Andreas L. Opdahl. Visualizing cyber attacks with misuse case maps. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6182 LNCS(7491):262–275, 2010. ISSN 03029743. doi: 10.1007/978-3-642-14192-8{-}24.
- [46] Cheng I. Rui Shen Basu A. Rossol, N. Touchfree medical interfaces. URL <https://www.scopus.com/record/display.uri?eid=2-s2.0-84943744362&origin=resultslist&sort=plf-f&src=s&st1=%22Medical+Interface%22&nlo=&nlr=&nls=&sid=0315CF6E9EABC032FC7FF69F377B71E8.ZmAYsXCHIBxxTXbnsoe5w%3a400&sot=b&sdt=cl&cluster=scopubyr%2c%222016%22%2ct>.
- [47] Stacking Arduino Shields — Freetronics. URL http://www.freetronics.com.au/pages/stacking-arduino-shields#.VycD4_krKM8.
- [48] Intel® Galileo Gen 2 Development Board, . URL <http://www.intel.com/content/www/us/en/embedded/products/galileo/galileo-overview.html>.
- [49] Intel® Mobile Modem Solutions, . URL <http://www.intel.com/content/www/us/en/mobile/modem-solutions.html>.
- [50] Qualcomm Cellular Modem Product Catalog — Qualcomm. URL <https://www.qualcomm.com/products/modems>.
- [51] IoT - Home — Intel® Developer Zone, . URL <https://software.intel.com/en-us/iot/home>.

- [52] IEEE SA - Healthcare IT Standards. URL https://standards.ieee.org/findstds/standard/healthcare_it.html.
- [53] Qt - Product — The IDE. URL <https://www.qt.io/ide/>.
- [54] Ashok N. Srivastava and Johann Schumann. Software health management: a necessity for safety critical systems. *Innovations in Systems and Software Engineering*, 9(4):219–233, 12 2013. ISSN 1614-5046. doi: 10.1007/s11334-013-0212-0. URL <http://link.springer.com/10.1007/s11334-013-0212-0>.
- [55] Shaohui Wang, Anaheed Ayoub, Radoslav Ivanov, Oleg Sokolsky, and Insup Lee. Contract-based blame assignment by trace analysis. In *Proceedings of the 2nd ACM international conference on High confidence networked systems - HiCoNS '13*, page 117, New York, New York, USA, 2013. ACM Press. ISBN 9781450319614. doi: 10.1145/2461446.2461463. URL <http://dl.acm.org/citation.cfm?doid=2461446.2461463>.
- [56] Robert H Birkhahn, Elizabeth Haines, Wendy Wen, Lakshmi Reddy, William M Briggs, Paris A Datillo, C.J. Lindsell, V. Anantharaman, D. Diercks, et al., R.J. Ryan, C.J. Lindsell, J.E. Hollander, et al., W.B. Gibler, J.P. Runyon, R.C. Levy, et al., M.A. Gomez, J.L. Anderson, L.A. Karagounis, et al., L.G. Graff, J. Dallara, M.A. Ross, et al., E. Braunwald, E.M. Antman, J.W. Beasley, et al., E.M. Antman, D.T. Anbe, P.W. Armstrong, et al., K. Thygesen, J.S. Alpert, H.D. White, F.S. Apple, A.Y. Chung, M.E. Kogut, et al., E. Lee-Lewandrowski, D. Corboy, K. Lewandrowski, et al., C.A. Parvin, S.F. Lo, S.M. Deuser, et al., A.J. Singer, P. Viccellio, H.C. Thode, et al., T.J. Ryan, J.L. Anderson, E.M. Antman, et al., F.S.

- Apple, R.H. Christenson, R. Valdes, et al., S.M. Ng, P. Krishnaswamy, R. Morrissey, et al., J. McCord, R.M. Nowak, P.A. McCullough, et al., T.J. Clark, P.H. McPherson, K.F. Buechler, A.L. Blomkalns, W.B. Gibler, M.J. Schull, D.A. Redelmeier, A.B. Storrow, C.J. Lindsell, S.P. Collins, et al., T.E. Caragher, B.B. Fernandez, F.L. Jacobs, and et al. Estimating the clinical impact of bringing a multimarker cardiac panel to the bedside in the ED. *The American journal of emergency medicine*, 29(3):304–8, 3 2011. ISSN 1532-8171. doi: 10.1016/j.ajem.2009.12.007. URL <http://www.ncbi.nlm.nih.gov/pubmed/20825823>.
- [57] Laura van Dommelen, Frank H van Tiel, Sander Ouburg, Elfi E H G Brouwers, Peter H W Terporten, Paul H M Savelkoul, Servaas A Morré, Cathrien A Bruggeman, and Christian J P A Hoebe. Alarming performance in Chlamydia trachomatis point-of-care testing. *Sexually transmitted infections*, 86(5): 355–9, 10 2010. ISSN 1472-3263. doi: 10.1136/sti.2010.042598. URL <http://www.ncbi.nlm.nih.gov/pubmed/20876754>.
- [58] J.M.S.a Alonso and D.M.M.b Pereira. Medical software requirements at the new Cuban regulations for evaluation and state control of medical devices. In *IFMBE Proceedings*, volume 33 IFMBE, pages 433–435, 2013. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-84875013855&partnerID=40&md5=87f6ae4b2499064dd7d8d1c98651b930>.
- [59] Yi Yan, Shenglin Liu, Qiang Zhang, and Hanxi Wu. Analysis of medical device recall reports in FDA database in 2005-2006. In *IFMBE Proceedings*, volume 39 IFMBE, pages 766–769, 2013. ISBN 9783642293047. doi: 10.1007/978-3-642-29305-4{_}201.

- [60] Sylvia Kierkegaard and Patrick Kierkegaard. Danger to public health: Medical devices, toxicity, virus and fraud. *Computer Law and Security Review*, 29(1):13–27, 2013. ISSN 02673649. doi: 10.1016/j.clsr.2012.11.006.
- [61] Health and Social Care Network - NHS Digital. URL <https://digital.nhs.uk/services/health-and-social-care-network>.
- [62] Cloud Standards Customer Council — CSCC. URL <http://www.cloud-council.org/>.
- [63] Digital Health in Canada — Canada Health Infoway. URL <https://www.infoway-inforoute.ca/en/>.
- [64] Gerald Holl, Daniel Thaller, Paul Grünbacher, and Christoph Elsner. Managing emerging configuration dependencies in multi product lines. In *Proceedings of the Sixth International Workshop on Variability Modeling of Software-Intensive Systems - VaMoS '12*, pages 3–10, New York, New York, USA, 2012. ACM Press. ISBN 9781450310581. doi: 10.1145/2110147.2110148. URL <http://dl.acm.org/citation.cfm?doid=2110147.2110148>.
- [65] Benedikt Ostermaier, Matthias Kovatsch, and Silvia Santini. Connecting things to the web using programmable low-power WiFi modules. In *Proceedings of the Second International Workshop on Web of Things - WoT '11*, page 1, New York, New York, USA, 2011. ACM Press. ISBN 9781450306249. doi: 10.1145/1993966.1993970. URL <http://portal.acm.org/citation.cfm?doid=1993966.1993970>.

- [66] Cristian Ruz, Franoise Baude, and Bastien Sauvan. Component-based generic approach for reconfigurable management of component-based SOA applications. In *Proceedings of the 3rd International Workshop on Monitoring, Adaptation and Beyond - MONA '10*, pages 25–32, New York, New York, USA, 2010. ACM Press. ISBN 9781450304221. doi: 10.1145/1929566.1929570. URL <http://portal.acm.org/citation.cfm?doid=1929566.1929570>.
- [67] Stephen Bono, Aviel Rubin, Adam Stubblefield, and Matthew Green. Security through legality. *Communications of the ACM*, 49(6):41, 6 2006. ISSN 00010782. doi: 10.1145/1132469.1132499. URL <http://portal.acm.org/citation.cfm?doid=1132469.1132499>.
- [68] Murray McKay. Best practices in automation security. In *IEEE Cement Industry Technical Conference (Paper)*, 2012. ISBN 9781467302869. doi: 10.1109/CITCON.2012.6215678.
- [69] Martin F Johansen, Oystein Haugen, and Franck Fleurey. Bow tie testing: a testing pattern for product lines. *Proceedings of the 16th European Conference on Pattern Languages of Programs*, pages 9:1–9:13, 2012. doi: 10.1145/2396716.2396725. URL <http://doi.acm.org/10.1145/2396716.2396725>.
- [70] Jayaram Natarajan, Joshua Wells, Abhijit Chatterjee, and Adit Singh. Distributed comparison test driven multiprocessor speed-tuning: Targeting performance gains under extreme process variations. In *Proceedings of the Asian Test Symposium*, pages 154–160, 2011. ISBN 9780769545837. doi: 10.1109/ATS.2011.84.

- [71] I Vilajosana and M Dohler. 19 - Machine-to-machine (M2M) communications for smart cities. In Carles Antn-HaroMischa Dohler, editor, *Machine-To-machine (M2m) Communications*, pages 355–373. Woodhead Publishing, 2015. ISBN 978-1-78242-102-3. doi: <http://dx.doi.org/10.1016/B978-1-78242-102-3.00019-8>. URL <http://www.sciencedirect.com/science/article/pii/B9781782421023000198>.
- [72] Valdivino Alexandre de Santiago Júnior and Nandamudi Lankalapalli Vijaykumar. Generating model-based test cases from natural language requirements for space application software. *Software Quality Journal*, 20(1):77–143, 2012. ISSN 15731367. doi: 10.1007/s11219-011-9155-6.
- [73] Changhai Nie and Hareton Leung. The Minimal Failure-Causing Schema of Combinatorial Testing. *ACM Transactions on Software Engineering and Methodology*, 20(4):1–38, 2011. ISSN 1049331X. doi: 10.1145/2000799.2000801. URL <http://dl.acm.org/citation.cfm?doid=2000799.2000801>.
- [74] Itai Segall, Rachel Tzoref-Brill, and Eitan Farchi. Using binary decision diagrams for combinatorial test design. In *ISSTA '11 Proceedings of the 2011 International Symposium on Software Testing and Analysis*, page 254, 2011. ISBN 9781450305624. doi: 10.1145/2001420.2001451. URL <http://portal.acm.org/citation.cfm?doid=2001420.2001451>.
- [75] Raghu N. Kacker, D. Richard Kuhn, Yu Lei, and James F. Lawrence. Combinatorial testing for software: An adaptation

- of design of experiments. *Measurement: Journal of the International Measurement Confederation*, 46(9):3745–3752, 2013. ISSN 02632241. doi: 10.1016/j.measurement.2013.02.021.
- [76] M.I. Capel and L.E.M. Morales. A formal compositional verification approach for safety-critical systems correctness: Model-Checking based methodological approach to automatically verify safety critical systems software. In *ICEIS 2012 - Proceedings of the 14th International Conference on Enterprise Information Systems*, volume 2 ISAS, 2012. ISBN 9789898565105.
- [77] Kevin Gary, Andinet Enquobahrie, Luis Ibanez, Patrick Cheng, Ziv Yaniv, Kevin Cleary, Shylaja Kokoori, Benjamin Muffih, and John Heidenreich. Agile methods for open source safety-critical software. *Software - Practice and Experience*, 41(9): 945–962, 2011. ISSN 00380644. doi: 10.1002/spe.1075.
- [78] Masoumeh Taromirad and Richard F. Paige. Agile requirements traceability using domain-specific modelling languages. In *Proceedings of the 2012 Extreme Modeling Workshop on - XM '12*, pages 45–50, 2012. ISBN 9781450318044. doi: 10.1145/2467307.2467316. URL <http://dl.acm.org/citation.cfm?doid=2467307.2467316>.
- [79] Kent Beck, Mike Beedle, Arie Van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas. Agile Manifesto, 2001. ISSN 10708588. URL <http://agilemanifesto.org/>.
- [80] Securing networks through the Internet. *The Health Management Technology*. URL [http:](http://)

[//www.healthmgttech.com/articles/201309/securing-networks-through-the-internet.php](http://www.healthmgttech.com/articles/201309/securing-networks-through-the-internet.php).

- [81] Aurel Ymeti, Paul H. J. Nederkoorn, Alma Dudia, Vinod Subramaniam, and Johannes S. Kanger. *Rapid, ultrasensitive detection of microorganisms based on interferometry and lab-on-a-chip nanotechnology*. In Craig S. Halvorson, rka O. Southern, B. V. K. Vijaya Kumar, Salil Prabhakar, and Arun A. Ross, editors, *Proceedings of SPIE - The International Society for Optical Engineering*, volume 7306, pages 73060J–73060J–7, 5 2009. ISBN 9780819475725. doi: 10.1117/12.818466. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-69949112823&partnerID=tZ0tx3y1>.
- [82] AKAMAIS STATE OF THE INTERNET Q1 2014 REPORT — VOLUME 7 NUMBER 1. URL <http://www.akamai.com/dl/akamai/akamai-soti-q114.pdf>.
- [83] TCP and UDP Ports. URL http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers.
- [84] Jangirala Srinivas, Sourav Mukhopadhyay, and Dheerendra Mishra. Secure and efficient user authentication scheme for multi-gateway wireless sensor networks. *Ad Hoc Networks*, 54: 147–169, 2017. ISSN 15708705. doi: 10.1016/j.adhoc.2016.11.002.
- [85] Kun-Hee Han and Woo-Sik Bae. Proposing and verifying a security-enhanced protocol for IoT-based communication for medical devices. *Cluster Computing*, 19(4):2335–2341, 12 2016. ISSN 1386-7857. doi: 10.1007/s10586-016-0669-3. URL <http://link.springer.com/10.1007/s10586-016-0669-3>.

- [86] Xin Huang, Paul Craig, Hangyu Lin, and Zheng Yan. Se-cIoT: a security framework for the Internet of Things. *Security and Communication Networks*, 9(16):3083–3094, 11 2016. ISSN 19390114. doi: 10.1002/sec.1259. URL <http://doi.wiley.com/10.1002/sec.1259>.
- [87] Nima Karimian, Paul A. Wortman, and Fatemeh Tehrani-poor. Evolving authentication design considerations for the internet of biometric things (IoBT). *Proceedings of the Eleventh IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis - CODES '16*, pages 1–10, 2016. doi: 10.1145/2968456.2973748. URL <http://dl.acm.org/citation.cfm?doid=2968456.2973748>.
- [88] Xianghui Cao, Devu Manikantan Shila, Yu Cheng, Zequ Yang, Yang Zhou, and Jiming Chen. Ghost-in-ZigBee: Energy Depletion Attack on ZigBee-Based Wireless Networks. *IEEE Internet of Things Journal*, 3(5):816–829, 10 2016. ISSN 2327-4662. doi: 10.1109/JIOT.2016.2516102. URL <http://ieeexplore.ieee.org/document/7377010/>.
- [89] Vctor Custodio, Francisco J Herrera, Gregorio López, and Jos Ignacio Moreno. A review on architectures and communications technologies for wearable health-monitoring systems. *Sensors (Basel, Switzerland)*, 12(10):13907–46, 10 2012. ISSN 1424-8220. doi: 10.3390/s121013907. URL <http://www.ncbi.nlm.nih.gov/pubmed/23202028><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3545599>.

- [90] Nancy Cam-Winget, Ahmad-Reza Sadeghi, and Yier Jin. INVITED: Can IoT be Secured: Emerging Challenges in Connecting the Unconnected. doi: 10.1145/2897937.2905004.
- [91] ShuQi Wang, Thirupathiraja Chinnasamy, Mark A. Lifson, Fatih Inci, and Utkan Demirci. Flexible Substrate-Based Devices for Point-of-Care Diagnostics. *Trends in Biotechnology*, 34(11):909–921, 2016. ISSN 01677799. doi: 10.1016/j.tibtech.2016.05.009.
- [92] Zimu Guo, Nima Karimian, Mark M. Tehranipoor, and Domenic Forte. Hardware security meets biometrics for the age of IoT. In *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1318–1321. IEEE, 5 2016. ISBN 978-1-4799-5341-7. doi: 10.1109/ISCAS.2016.7527491. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7527491>.
- [93] Ding Ding, Mauro Conti, and Agusti Solanas. A smart health application and its related privacy issues. In *2016 Smart City Security and Privacy Workshop (SCSP-W)*, pages 1–5. IEEE, 4 2016. ISBN 978-1-5090-2924-2. doi: 10.1109/SCSPW.2016.7509558. URL <http://ieeexplore.ieee.org/document/7509558/>.
- [94] Adamko Attila, Abel Garai, and Istvan Pentek. Common open telemedicine hub and infrastructure with interface recommendation. In *2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, pages 385–390. IEEE, 5 2016. ISBN 978-1-5090-2380-6. doi: 10.1109/SACI.2016.7507407. URL <http://ieeexplore.ieee.org/document/7507407/>.

- [95] Seungsoo Seo. Preserving patient s anonymity for mobile healthcare system in IoT environment. 5. ISSN 15501477.
- [96] SDL Threat Modeling Tool. URL <http://www.microsoft.com/security/sdl/adopt/threatmodeling.aspx>.
- [97] Hassan El-Hadary and Sherif El-Kassas. Capturing security requirements for software systems. *Journal of Advanced Research*, 5(4):463–472, 7 2014. ISSN 20901232. doi: 10.1016/j.jare.2014.03.001. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-84901693833&partnerID=tZ0tx3y1>.
- [98] AttackTree+ - Isograph. URL <http://www.isograph.com/software/attacktree/>.
- [99] ThreatModeler Archives — MyAppSecurity. URL <http://myappsecurity.com/threatmodeler/>.
- [100] Nmap - Free Security Scanner For Network Exploration & Security Audits. URL <http://nmap.org/>.
- [101] Chapter15.Nmap Reference Guide. URL <http://nmap.org/book/man.html>.
- [102] Application and Network Attacks. URL <http://sl.sierracollege.edu/cis147/TextBook/CIS147-TextbookChapter3.pdf>.
- [103] Jiangtao Wen, Mike Severa, Wenjun Zeng, Max Luttrell, and Weiyin Jin. A format-compliant configurable encryption framework for access control of multimedia. In *2001 IEEE Fourth Workshop on Multimedia Signal Processing*, pages 435–440, 2001. ISBN

0780370252. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-0035791314&partnerID=tZ0tx3y1>.
- [104] Muneer Malik and Dharma P. Agrawal. Secure web framework for mobile devices. In *2012 IEEE Globecom Workshops*, pages 781–786. IEEE, 12 2012. ISBN 978-1-4673-4941-3. doi: 10.1109/GLOCOMW.2012.6477674. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-84875655699&partnerID=tZ0tx3y1>.
- [105] Proceedings of the 2009 ACM Workshop on Cloud Computing Security, CCSW '09, Co-located with the 16th ACM Computer and Communications Security Conference, CCS'09, 2009. ISSN 15437221. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-74249109576&partnerID=tZ0tx3y1>.
- [106] Saul N Weingart, Hope E Hamrick, Sharon Tutkus, Alexander Carbo, Daniel Z Sands, Anjala Tess, Roger B Davis, David W Bates, and Russell S Phillips. Medication safety messages for patients via the web portal: the MedCheck intervention. *International journal of medical informatics*, 77(3):161–8, 3 2008. ISSN 1386-5056. doi: 10.1016/j.ijmedinf.2007.04.007. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-38549147032&partnerID=tZ0tx3y1>.
- [107] Vipul Gupta, Matthew Millard, Stephen Fung, Yu Zhu, Nils Gura, Hans Eberle, and Sheueling Chang Shantz. Sizzle: A standards-based end-to-end security architecture for the embedded internet. In *Proceedings - Third IEEE International Conference on Pervasive Computing and Communications, PerCom 2005*, volume 2005, pages 247–256, 2005. ISBN

0769522998. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-33646569085&partnerID=tZ0tx3y1>.
- [108] Malware - Malicious Virus Code Detection - Trojan - Trojan Horse — Symantec. URL http://us.norton.com/security_response/malware.jsp.
- [109] Mark Ciampa. *Guide to Network Security*. URL https://books.google.com/books/about/Security+_Guide_to_Network_Security_Fund.html?id=CIHYWBrg9JQC.
- [110] What is social engineering? - Definition from WhatIs.com, . URL <http://searchsecurity.techtarget.com/definition/social-engineering>.
- [111] What is phishing? - Definition from WhatIs.com, . URL <http://searchsecurity.techtarget.com/definition/phishing>.
- [112] Impersonation. URL <https://technet.microsoft.com/en-us/library/cc961980.aspx>.
- [113] What is dumpster diving? - Definition from WhatIs.com, . URL <http://searchsecurity.techtarget.com/definition/dumpster-diving>.
- [114] Understanding Denial-of-Service Attacks — US-CERT. URL <https://www.us-cert.gov/ncas/tips/ST04-015>.
- [115] Moti Geva, Amir Herzberg, and Yehoshua Gev. Bandwidth Distributed Denial of Service: Attacks and Defenses. *IEEE Security & Privacy*, 12(1):54–61, 1 2014. ISSN 1540-7993. doi: 10.1109/MSP.2013.55. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-84896462209&partnerID=tZ0tx3y1>.

- [116] Tero Rontti, Anna-Maija Juuso, and Ari Takanen. Preventing DoS attacks in NGN networks with proactive specification-based fuzzing. *IEEE Communications Magazine*, 50(9):164–170, 9 2012. ISSN 0163-6804. doi: 10.1109/MCOM.2012.6295728. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-84866283429&partnerID=tZ0tx3y1>.
- [117] DHCP Consumption Attack and Mitigation Techniques White Paper. URL http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white_Paper_C11_603833.html.
- [118] Mobile equipment identifier - Wikipedia, the free encyclopedia. URL http://en.wikipedia.org/wiki/Mobile_equipment_identifier.
- [119] *3GPP TS 22.016: International Mobile Equipment Identities (IMEI)*. 2009. URL http://www.3gpp.org/ftp/Specs/archive/22_series/22.016/22016-900.zip.
- [120] *M2M Communications: A Systems Approach*. John Wiley & Sons, 2012. ISBN 1119940966. URL <https://books.google.com/books?id=bVaqAFpH6EgC&pgis=1>.
- [121] AT&T M2M Solutions — Machine to Machine — M2M Communications. URL http://www.business.att.com/enterprise/Family/mobility-services/machine-to-machine/#fbid=-70Uq7Bu73_.
- [122] Machine-to-Machine Products - Ericsson. URL <http://www.ericsson.com/ourportfolio/products/machine-to-machine-products?nav=productcategory002>.
- [123] M2M Communication: A System Approach. page 233. .

- [124] M2M Communication: A System Approach, Section 8.3.5. .
- [125] Compliance Guides for Small Businesses — FCC.gov. URL <https://www.fcc.gov/encyclopedia/compliance-guides-small-businesses>.
- [126] Safe For Network Certification - Verizon Wireless. URL https://opennetwork.verizonwireless.com/content/dam/open-development/files/LTE_3GPP_Band13_Device_Reqs.pdf.
- [127] What is network-attached storage (NAS)? - Definition from WhatIs.com, . URL <http://searchstorage.techtarget.com/definition/network-attached-storage>.
- [128] Peter Mell and Timothy Grance. The NIST Definition of Cloud Computing Recommendations of the National Institute of Standards and Technology. doi: 10.6028/NIST.SP.800-145.
- [129] Public vs. Private Cloud Computing. URL <http://www.onlinetech.com/resources/references/public-vs-private-cloud-computing>.
- [130] Synology NAS User's Guide Based on DSM 4.3. .
- [131] Axxin - Innovation to Impact. URL <http://www.axxin.com/>.
- [132] QoS: Traffic Control and Speed Limit. URL <https://blog.synology.com/?p=1554>.
- [133] Editing Narrowband IoT - Wikipedia. URL https://en.wikipedia.org/w/index.php?title=Narrowband_IoT&action=edit.

- [134] *IoT now : how to run an IoT enabled business.*
. URL <https://www.iot-now.com/2016/06/21/48833-cat-m1-vs-nb-iot-examining-the-real-differences/>.
- [135] Synology Inc., . URL <https://www.synology.com/en-us/>.
- [136] FAM (fluorescein), HEX, JOE, ROX, TAMRA, TET, Texas Red® and others.
URL <https://www.atdbio.com/content/33/FAM-fluorescein-HEX-JOE-ROX-TAMRA-TET-Texas-Red-and-others>.
- [137] NAS Security, . URL https://www.synology.com/en-us/knowledgebase/DSM/help/DSM/Tutorial/secure_your_nas.
- [138] NAS Maximum Shared folders, . URL https://www.synology.com/en-us/knowledgebase/DSM/tutorial/Backup_Restore/How_many_Shared_Folder_Sync_tasks_does_Synology_Product_support.
- [139] P. Martigne. *Machine-To-machine (M2m) Communications*. 2015. ISBN 9781782421023. doi: 10.1016/B978-1-78242-102-3.00002-2. URL <http://www.sciencedirect.com/science/article/pii/B9781782421023000022>.
- [140] Tyler Nicholas Edward Steane and P J Radcliffe. An enhanced implementation of a novel IoT joining protocol. In *2016 26th International Telecommunication Networks and Applications Conference (ITNAC)*, pages 22–25. IEEE, 12 2016. ISBN 978-1-5090-0919-0. doi: 10.1109/ATNAC.2016.7878776. URL <http://ieeexplore.ieee.org/document/7878776/>.
- [141] Claudio Savaglio, Giancarlo Fortino, and Mengchu Zhou. Towards interoperable, cognitive and autonomic IoT systems: An

- agent-based approach. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 58–63. IEEE, 12 2016. ISBN 978-1-5090-4130-5. doi: 10.1109/WF-IoT.2016.7845459. URL <http://ieeexplore.ieee.org/document/7845459/>.
- [142] Farzad Samie, Vasileios Tsoutsouras, Lars Bauer, Sotirios Xydis, Dimitrios Soudris, and Jorg Henkel. Computation offloading and resource allocation for low-power IoT edge devices. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 7–12. IEEE, 12 2016. ISBN 978-1-5090-4130-5. doi: 10.1109/WF-IoT.2016.7845499. URL <http://ieeexplore.ieee.org/document/7845499/>.
- [143] Shaibal Chakrabarty, Monica John, and Daniel W. Engels. Black routing and node obscuring in IoT. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 323–328. IEEE, 12 2016. ISBN 978-1-5090-4130-5. doi: 10.1109/WF-IoT.2016.7845477. URL <http://ieeexplore.ieee.org/document/7845477/>.
- [144] Association for Computing Machinery and Institute of Electrical and Electronics Engineers. *2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS) : October 2-7, 2016, Pittsburgh Marriott City Center, Pittsburgh, PA*. ISBN 9781509035908. URL <http://ieeexplore.ieee.org/document/7750969/>.
- [145] ns-3. URL <https://www.nsnam.org/>.
- [146] Conceptual Overview ns-3 vns-3.10 documentation, . URL <https://www.nsnam.org/docs/release/3.10/tutorial/html/conceptual-overview.html#key-abstractions>.

- [147] Jiancheng Ye, Ka-Cheong Leung, and Victor O. K. Li. Optimal delay control for combating bufferbloat in the Internet. In *2016 IEEE International Conference on Communication Systems (ICCS)*, pages 1–6. IEEE, 12 2016. ISBN 978-1-5090-3423-9. doi: 10.1109/ICCS.2016.7833560. URL <http://ieeexplore.ieee.org/document/7833560/>.
- [148] Conceptual Overview ns-3 vns-3.10 documentation, . URL <https://www.nsnam.org/docs/release/3.10/tutorial/html/conceptual-overview.html>.
- [149] H. Zimmermann. OSI Reference Model—The ISO Model of Architecture for Open Systems Interconnection. *IEEE Transactions on Communications*, 28(4):425–432, 4 1980. ISSN 0096-2244. doi: 10.1109/TCOM.1980.1094702. URL <http://ieeexplore.ieee.org/document/1094702/>.
- [150] Narrow Band Internet of Things (NB-IoT) — Internet of Things. URL <https://www.gsma.com/iot/narrow-band-internet-of-things-nb-iot/>.
- [151] Traffic Control Layer Model Library. URL <https://www.nsnam.org/docs/models/html/traffic-control-layer.html>.
- [152] 1.4B 5G Connections by 2025? Analysts Say Yes — Light Reading. URL http://www.lightreading.com/mobile/5g/14b-5g-connections-by-2025-analysts-say-yes/d/d-id/735454?_mc=RSS_LR_EDT.
- [153] Joseph L Global Burden of Disease Health Financing Collaborator Network, Annie Haakenstad, Angela Micah, Mark Moses, Cristiana Abbafati, Pawan Acharya, Tara Ballav Adhikari, Arsne Kouablan Adou, Aliasghar Ahmad Kiadaliri, Khurshid

Alam, Reza Alizadeh-Navaei, Ala'a Alkerwi, Walid Ammar, Carl Abelardo T Antonio, Olatunde Aremu, Solomon Weldegebreal Asgedom, Tesfay Mehari Atey, Leticia Avila-Burgos, Ashish Awasthi, Rakesh Ayer, Hamid Badali, Maciej Banach, Amrit Banstola, Aleksandra Barac, Abate Bekele Belachew, Charles Birungi, Nicola L Bragazzi, Nicholas J K Breitborde, Lucero Cahuana-Hurtado, Josip Car, Ferrn Catalá-López, Abigail Chapin, Lalit Dandona, Rakhi Dandona, Ahmad Daryani, Samath D Dharmaratne, Manisha Dubey, Dumessa Edessa, Erika Eldrenkamp, Babak Eshrati, Andr Faro, Andrea B Feigl, Ama P Fenny, Florian Fischer, Nataliya Foigt, Kyle J Foreman, Nancy Fullman, Mamata Ghimire, Srinivas Goli, Alemayehu Desalegne Hailu, Samer Hamidi, Hilda L Harb, Simon I Hay, Delia Hendrie, Gloria Ikilezi, Mehdi Javanbakht, Denny John, Jost B Jonas, Alexander Kaldjian, Amir Kasaeian, Jennifer Kates, Ibrahim A Khalil, Young-Ho Khang, Jagdish Khubchandani, Yun Jin Kim, Jonas M Kinge, Soewarta Kosen, Kristopher J Krohn, G Anil Kumar, Hilton Lam, Stefan Listl, Hassan Magdy Abd El Razek, Mohammed Magdy Abd El Razek, Azeem Majeed, Reza Malekzadeh, Deborah Carvalho Malta, George A Mensah, Atte Meretoja, Ted R Miller, Erkin M Mirrakhimov, Fitsum Weldegebreal Mlashu, Ebrahim Mohammed, Shafiu Mohammed, Mohsen Naghavi, Vinay Nangia, Frida Namnyak Ngalesoni, Cuong Tat Nguyen, Trang Huyen Nguyen, Yirga Niriayo, Mehdi Noroozi, Mayowa O Owolabi, David M Pereira, Mostafa Qorbani, Anwar Rafay, Alireza Rafiei, Vafa Rahimi-Movaghar, Rajesh Kumar Rai, Usha Ram, Chhabi Lal Ranabhat, Sarah E Ray, Robert C Reiner, Nafis Sadat, Haniye Sadat Sajadi, Joo Vasco

- Santos, Abdur Razzaque Sarker, Benn Sartorius, Maheswar Satpathy, Miloje Savic, Matthew Schneider, Sadaf G Sepanlou, Masood Ali Shaikh, Mehdi Sharif, Jun She, Aziz Sheikh, Mekonnen Sisay, Samir Soneji, Moslem Soofi, Henok Tadesse, Tianchan Tao, Tara Templin, Azeb Gebresilassie Tesema, Subash Thapa, Alan J Thomson, Ruoyan Tobe-Gai, Roman Topor-Madry, Bach Xuan Tran, Khanh Bao Tran, Tung Thanh Tran, Eduardo A Undurraga, Tommi Vasankari, Francesco S Violante, Tissa Wijeratne, Gelin Xu, Naohiro Yonemoto, Mustafa Z Younis, Chuanhua Yu, Maysaa El Sayed Zaki, Lei Zhou, Bianca Zlavog, and Christopher J L Murray. Spending on health and HIV/AIDS: domestic health spending and development assistance in 188 countries, 1995-2015. *Lancet (London, England)*, 391(10132):1799–1829, 4 2018. ISSN 1474-547X. doi: 10.1016/S0140-6736(18)30698-6. URL <http://www.ncbi.nlm.nih.gov/pubmed/29678342>.
- [154] WHO — Density of physicians (total number per 1000 population, latest available year). *WHO*, 2018. URL http://www.who.int/gho/health_workforce/physicians_density/en/.
- [155] Mapping the worlds 4.3 billion Internet addresses - The Washington Post. URL https://www.washingtonpost.com/news/the-switch/wp/2015/01/07/mapping-the-worlds-4-3-billion-internet-addresses/?utm_term=.19e70f6d4c67.
- [156] Egemen Taskin. GSM MSC/VLR Unstructured Supplementary Service Data(USSD) Service. 2012. URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.884.9083&rep=rep1&type=pdf>.

- [157] Paul Kocher, Daniel Genkin, Daniel Gruss, Werner Haas, Mike Hamburg, Moritz Lipp, Stefan Mangard, Thomas Prescher, Michael Schwarz, and Yuval Yarom. Spectre Attacks: Exploiting Speculative Execution *. URL <https://spectreattack.com/spectre.pdf>.