



**5G Mobile Network Architecture**  
for diverse services, use cases, and applications in 5G and beyond

**Deliverable D3.2**

*Final resilience and security report*

<b>Contractual Date of Delivery</b>	2019-03-31
<b>Actual Date of Delivery</b>	2019-04-08
<b>Work Package</b>	WP3 – Resilience and Security
<b>Editor(s)</b>	Diomidis MICHALOPOULOS (NOK-DE)
<b>Reviewers</b>	Xiaowei ZHANG (DT), Amina FELLAN (UNIKL), Dimosthenis IOANNIDIS (CERTH)
<b>Dissemination Level</b>	Public
<b>Type</b>	Report
<b>Version</b>	1.0
<b>Total number of pages</b>	112

**Abstract:** This report provides the final results on the resilience and security concepts and developments carried out in the framework of the 5G-MoNArch project. It reflects the work conducted in the respective work package 3 of the project and focuses on complementing and evaluating the concepts initially proposed in Deliverable D3.1. Specifically, the considered concepts include macro diversity via data duplication and network coding, root cause identification of faults in sliced network environments, controller scalability, context-aware VNF migration, security trust zones, as well as a joint study between resilience and security in virtualised network slicing environments. In addition to Deliverable D3.1, this report includes an analysis of a graph-based anomaly detection method for identifying potential threats, as well as a study of the effect of security threats on the main 5G network components, with direct application to the Hamburg Smart Sea Port use case.

**Keywords:** Resilience, RAN reliability, Telco cloud reliability, Fault management, Security threat evaluation, Joint resilience and security study

## Executive Summary

This is the final deliverable of work package 3 (WP3) of the 5G-MoNArch project, reporting on the project's final conceptual and development results regarding network resilience and security. The focus of this deliverable is on the analysis and evaluation of the fundamental techniques proposed in the 5G-MoNArch Deliverable D3.1 [5GM-D3.1], which are further augmented with complementary concepts towards a resilient and secure operation of a 5G network. In this respect, the three major topics of WP3 of 5G-MoNArch – namely, i) *Radio Access Network (RAN) reliability*, ii) *telco cloud resilience*, and iii) *security* – are put forward. Their ability to provide an adequate performance is assessed, as well as their novelty compared to existing techniques.

Specifically, in this document the concepts of macro diversity via data duplication and network coding are evaluated, through both analysis and numerical simulations, demonstrating their ability to provide sufficient levels of network reliability. These two techniques reflect the set of functionalities applied within the RAN for increasing the reliability levels, i.e., for ensuring a larger percentage of transmitted packets to be flawlessly delivered at the target device within a given period of time. Numerical results are provided for each of such techniques separately, demonstrating the achievable level of RAN reliability under certain topological, propagation as well as deployment assumptions. Furthermore, capitalising on the fact that network coding has the overall potential to achieve higher reliability levels by proper combination of data packets, while data duplication can outperform network coding under certain assumptions on latency, this document presents a hybrid approach between data duplication and network coding. The proposed hybrid scheme can in fact combine the advantages of both techniques by switching between the two accordingly, based on given rules. In this respect, numerical results show that the achieved reliability level outperforms data duplication and network coding when used individually.

Besides RAN reliability aspects, this document elaborates on the concepts of availability and resilience of the telco cloud, thus treating the concept of resilience not only at the RAN, but also at the telco cloud domain. In this regard, telco cloud resilience is enhanced via a solution for root cause identification of faults in sliced network environments, which complements the concept of fault management cognitive functions (FM CFs) described in [5GM-D3.1]. In a similar context, the initial concepts of controller scalability and context-aware VNF migration proposed in [5GM-D3.1] are extended. In particular, novel controller scalability solutions are presented, together with a scalability analysis pertaining to the Open Network Operating System (ONOS) framework and its survivability from network failures. Moreover, the concept of 5G Islands presented in [5GM-D3.1] is extended, including a simulation-based analysis that accounts for the migration cost as well as the outage loss associated with a slice-aware virtual network function migration from central to edge clouds.

Together with resilience, security plays an important role in maintaining the operation of critical network functionality. In this respect, this deliverable provides an additional analysis of the security threats associated to a 5G testbed deployment in industrial environments, which complements the initial results of [5GM-D3.1]. Such analysis pertains to the main 5G components, including devices, 5G network elements and network slicing related topics. Together with a theoretical study, simulations are provided where the security effect of the Security Trust Zones described in [5GM-D3.1] is assessed. In parallel to the trust zone analysis, a graph-based anomaly detection method is put forward, along with an extension that is based upon machine learning approaches.

Finally, besides assessing and extending the three major components of WP3 of 5G-MoNArch, namely RAN reliability, telco cloud resilience, and security, this deliverable highlights the relationship between telco cloud resilience and security. To this end, a joint study is presented, which identifies synergies as well as common virtual resource allocation considerations relevant to a telco cloud deployment. In this regard, the joint resilience and security study addresses the impact of security threats to the network functionality, with particular focus on the effect on the network fault management approach considered in WP3. This study sheds light onto an approach where resilience and security solutions interact with each other in a common network slice consideration, towards a more efficient use of telco cloud resources.

## List of Authors

Partner	Name	E-mail
NOK-DE	Diomidis Michalopoulos Borislava Gajic Dimitrios Schoinianakis	<a href="mailto:diomidis.michalopoulos@nokia-bell-labs.com">diomidis.michalopoulos@nokia-bell-labs.com</a> <a href="mailto:borislava.gajic@nokia-bell-labs.com">borislava.gajic@nokia-bell-labs.com</a> <a href="mailto:dimitrios.schoinianakis@nokia-bell-labs.com">dimitrios.schoinianakis@nokia-bell-labs.com</a>
DT	Jakob Belschner	<a href="mailto:jakob.belschner@telekom.de">jakob.belschner@telekom.de</a>
NOK-FR	Gopalasingham Aravinthan Bessem Sayadi Fred Aklamanu Frederic Faucheux	<a href="mailto:gopalasingham.aravinthan@nokia-bell-labs.com">gopalasingham.aravinthan@nokia-bell-labs.com</a> <a href="mailto:bessem.sayadi@nokia-bell-labs.com">bessem.sayadi@nokia-bell-labs.com</a> <a href="mailto:fred.aklamanu@nokia-bell-labs.com">fred.aklamanu@nokia-bell-labs.com</a> <a href="mailto:frederic.faucheux@nokia-bell-labs.com">frederic.faucheux@nokia-bell-labs.com</a>
HWDU	Onurcan Iscan Yunyan Chang Oemer Bulakci	<a href="mailto:onurcan.iscan@huawei.com">onurcan.iscan@huawei.com</a> <a href="mailto:yunyan.chang@huawei.com">yunyan.chang@huawei.com</a> <a href="mailto:oemer.bulakci@huawei.com">oemer.bulakci@huawei.com</a>
ATOS	Beatriz Gallego Nicasio-Crespo Susana Gonzalez Zarzosa Ruben Trapero	<a href="mailto:beatriz.gallego-nicasio@atos.net">beatriz.gallego-nicasio@atos.net</a> <a href="mailto:susana.gzarzosa@atos.net">susana.gzarzosa@atos.net</a> <a href="mailto:ruben.trapero@atos.net">ruben.trapero@atos.net</a>
CERTH	Eleni Ketzaki Anastasios Drosou Stavros Papadopoulos Athanasios Tsakiris	<a href="mailto:eketzaki@iti.gr">eketzaki@iti.gr</a> <a href="mailto:drosou@iti.gr">drosou@iti.gr</a> <a href="mailto:spap@iti.gr">spap@iti.gr</a> <a href="mailto:atsakir@iti.gr">atsakir@iti.gr</a>
UNIKL	Bin Han	<a href="mailto:binhan@eit.uni-kl.edu">binhan@eit.uni-kl.edu</a>

## Revision History

Revision	Date	Issued by	Description
0.1	28.02.2019	NOK-DE	Reviewer clearance
1.0	08.04.2019	NOK-DE	Final version

## List of Acronyms and Abbreviations

2G	2nd Generation mobile wireless communication system (GSM, GPRS, EDGE)
3G	3rd Generation mobile wireless communication system (UMTS, HSPA)
4G	4th Generation mobile wireless communication system (LTE, LTE-A)
5G	5th Generation mobile wireless communication system
3GPP	3rd Generation Partnership Project
5G-PPP	5G Public Private Partnership
ABD	Anomaly Based Detection
ANN	Artificial Neural Network
AGW	Access Gateway
ARQ	Automatic Repeat Request
AUSF	Authentication Server Function
CAPEX	CAPital Expenditure
CC	Chase Combining
CF	Cognitive Function
cMTC	critical Machine Type Communications
CSMF	Communication Service Management Function
CU	Central Unit
DD	Data Duplication
DU	Distributed Unit
DOS	Denial of Service
E2E	End-to-end
ECF	Event Correlation Function
EID	Event Identifier
FTP	File Transfer Protocol
HH	Hansestadt Hamburg
HPA	Hamburg Port Authority
HSM	Hardware Secure Module
HSS	Home Service Subscriber
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoT	Internet of Things
FM	Fault Management
KPI	Key Performance Indicator
LoS	Line of Sight
LSTM	Long-Short-Term Memory
LTE	Long Term Evolution
MAC	Medium Access Control
MANO	Management and Orchestration
MCS	Modulation and Coding Scheme
mMTC	massive Machine Type Communications
MNO	Mobile Network Operator
MS	Mobile Station
NBA	Network Behaviour Analysis
NC	Network Coding
NDI	New Data Indicator
NE	Network Element
NF	Network Function
NGMN	Next Generation Mobile Networks

---

NM	Network Management
NR	New Radio
NSI	Network Slice Instance
NSMF	Network Slice Management Function
NSSI	Network Slice Subnet Instance
OPEX	Operational Expenditure
ONOS	Open Network Operating System
PAN	Personal Area Network
PDCP	Packet Data Convergence Protocol
PDSCH	Physical Downlink Shared Channel
PGW	Packet Gateway
ODC	Open Daylight Control
QoE	Quality of Experience
QoS	Quality of Service
RACH	Random Access Channel
RAN	Radio Access Network
RAT	Radio Access Technology
RLC	Radio Link Control
RRC	Radio Resource Control
RSRP	Reference Signal Received Power
S&R	Security and Resilience
SDN	Software Defined Network
SMm	Security Monitoring manager
SON	Self-Organised Network
SLA	Service Level Agreement
SthD	Security Threat Detection
SthP	Security Threat Prevention
SthR	Security Threat Reaction
STZ	Security Trust Zone
STZm	Security Trust Zone manager
SUMO	Simulation of Urban MObility
TB	Transfer Block
ThIntEx	Threat Intelligence Exchange
TTI	Time Transmission Interval
UE	User Equipment
URLLC	Ultra-Reliable Low-Latency Communication
UDM	Unified Data Management
VIM	Virtualised Infrastructure Manager
VM	Virtual Machine
VNF	Virtual Network Function
VPN	Virtual Private Network

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>11</b>
1.1	<i>Resilience and security as an end-to-end concept.....</i>	<i>11</i>
1.1.1	End-to-end availability enabled by RAN reliability and telco cloud resilience .....	12
1.1.2	Security as an end-to-end concept.....	13
1.2	<i>Resilience and security as part of project-wide study and evaluation .....</i>	<i>13</i>
1.2.1	WP3 enablers in 5G-MoNArch architecture: interaction with WP2 .....	14
1.2.2	Project-wide evaluation of WP3 enablers: interaction with WP6 .....	15
1.3	<i>Structure of the document.....</i>	<i>16</i>
<b>2</b>	<b>RAN reliability approaches .....</b>	<b>17</b>
2.1	<i>Data duplication as a RAN reliability approach .....</i>	<i>17</i>
2.1.1	On the considered data duplication scheme .....	17
2.1.2	Simulation analysis.....	19
2.1.2.1	<i>Architectural setup .....</i>	<i>19</i>
2.1.2.2	<i>Simulation setup .....</i>	<i>20</i>
2.1.3	Obtained results .....	21
2.1.3.1	<i>Investigation of the offered load.....</i>	<i>22</i>
2.1.3.2	<i>On the performance limits of data duplication.....</i>	<i>27</i>
2.2	<i>Performance and suitability assessment of network coding based multicasting approach .....</i>	<i>28</i>
2.2.1	Integration and suitability.....	28
2.2.2	Performance evaluation .....	30
2.3	<i>The hybrid data duplication / network coding approach .....</i>	<i>31</i>
2.3.1	The hybrid approach.....	32
2.3.2	Simulation methodology .....	33
2.3.3	Simulation results .....	34
2.3.3.1	<i>URLLC air interface.....</i>	<i>34</i>
2.3.3.2	<i>Medium air interface .....</i>	<i>35</i>
2.3.3.3	<i>Low reliability air interface.....</i>	<i>36</i>
2.3.3.4	<i>Impact of correlated links.....</i>	<i>37</i>
2.3.4	Concluding remarks on the hybrid approach.....	38
<b>3</b>	<b>Telco cloud resilience .....</b>	<b>39</b>
3.1	<i>Root cause identification of faults and applying redundancy for higher availability at telco cloud.....</i>	<i>39</i>
3.1.1	Advanced fault management event correlation in slicing enabled network .....	39
3.1.1.1	<i>Event correlation function, event notification message and its distribution area.</i>	<i>41</i>
3.1.1.2	<i>Event correlation function – deployment and benefits .....</i>	<i>42</i>
3.1.2	Applying redundancy for higher resilience .....	42
3.1.2.1	<i>Selection of suitable redundancy scheme .....</i>	<i>43</i>
3.2	<i>Augmented resilience via increased controller scalability .....</i>	<i>44</i>
3.2.1	Scalability solution analysis .....	44
3.2.1.1	<i>OpenDayLight .....</i>	<i>44</i>
3.2.1.2	<i>ONOS.....</i>	<i>49</i>
3.2.2	Scalable controller framework .....	53
3.3	<i>5G Islands: evaluating migration cost and outage loss for context-aware NF migration.....</i>	<i>55</i>
3.3.1	Central-to-edge VNF migration .....	55
3.3.2	Migration cost versus outage loss.....	56
3.3.3	Estimating outage loss.....	56
3.3.3.1	<i>Stateful VNF .....</i>	<i>56</i>
3.3.3.2	<i>Stateless VNF .....</i>	<i>56</i>

3.3.4	Simulation analysis.....	57
3.3.5	Neural network assisted 5G Island for stateful VNFs .....	58
<b>4</b>	<b>Security on 5G networks.....</b>	<b>60</b>
<b>4.1</b>	<b><i>Threat analysis on main 5G components</i> .....</b>	<b>60</b>
4.1.1	Device security .....	60
4.1.2	Security in 5G networks .....	61
4.1.3	Network slicing security .....	62
4.1.4	General remarks .....	64
<b>4.2</b>	<b><i>On the suitability of security trust zones</i> .....</b>	<b>64</b>
4.2.1	Suitability analysis .....	65
4.2.2	Process for defining STZs within a 5G infrastructure .....	66
4.2.3	Templates based deployment of STZs .....	68
4.2.4	Changing security requirements of a STZ.....	69
<b>4.3</b>	<b><i>Simulated threats and corresponding detectors</i>.....</b>	<b>69</b>
4.3.1	Security simulation campaign for monitoring 5G network slices .....	69
4.3.1.1	<i>Simulation of attacks against an STZ</i> .....	72
4.3.1.2	<i>Detection of attacks at SMm</i> .....	74
4.3.2	Network behaviour analysis .....	78
4.3.2.1	<i>A graph-based anomaly detection method</i> .....	79
4.3.2.2	<i>An extension of the anomaly detection method based on machine learning</i> .....	83
4.3.2.3	<i>Behaviour of attacked users and effect on the throughput performance</i> .....	91
<b>5</b>	<b>Resilience and security on common infrastructure: synergies and resource allocation issues.....</b>	<b>94</b>
<b>5.1</b>	<b><i>Interaction between fault management and security</i> .....</b>	<b>94</b>
5.1.1	Commonalities of fault and security management in 5G networks.....	94
5.1.2	Security & Resilience (S&R) cross domain / cross slice management entities.....	95
5.1.3	Joint security and fault management considerations for resource optimisation .....	95
<b>5.2</b>	<b><i>The Hamburg Smart Sea Port use case scenario: the effect of security threats to resources</i> .....</b>	<b>100</b>
<b>6</b>	<b>Summary .....</b>	<b>109</b>
<b>7</b>	<b>References .....</b>	<b>111</b>
<b>Appendix A</b>	<b>.....</b>	<b>114</b>

## List of Figures

Figure 1-1: Typical 5G network architecture .....	12
Figure 1-2: Main security areas in a 5G network .....	13
Figure 1-3: 5G-MoNArch architecture enriched with resilience and security .....	14
Figure 1-4: Interaction of WP3 enablers in the Management and Orchestration .....	15
Figure 2-1: Coordination of duplicated packets across different distributed units .....	18
Figure 2-2: Exemplary view of the architecture considered in the data duplication .....	19
Figure 2-3: Two-dimensional visualisation of the considered simulation setup .....	21
Figure 2-4: Three-dimensional visualisation of the considered setup .....	22
Figure 2-5: Low Load Scenario: Percentage of lost PDCP packets .....	23
Figure 2-6: Low Load Scenario: CDF of packet delivery delay at the application layer .....	23
Figure 2-7: Low Load Scenario: CDF of throughput for single connectivity .....	24
Figure 2-8: Downlink resource occupancy, measured in percentage of PRB allocation .....	24
Figure 2-9: Medium Load Scenario: Percentage of lost PDCP packets .....	25
Figure 2-10: Medium Load Scenario: CDF of throughput for single connectivity .....	25
Figure 2-11: Medium Load Scenario: CDF of packet delivery delay .....	25
Figure 2-12: High Load Scenario: Percentage of lost PDCP packets .....	26
Figure 2-13: High Load Scenario: CDF of packet delivery delay .....	26
Figure 2-14: High Load Scenario: CDF of throughput for single connectivity .....	26
Figure 2-15: The restricted area of the simulation scenario where the KPIs .....	27
Figure 2-16: Performance in terms of the KPIs of interest within a restricted area .....	28
Figure 2-17: Performance of the presented network coding approach .....	30
Figure 2-18: Performance of the presented network coding approach .....	31
Figure 2-19: Improving RAN reliability by multi-connectivity in combination .....	32
Figure 2-20: Simulation setup for the hybrid approach .....	33
Figure 2-21: Simulation of the hybrid approach: Lower layer / air interface performance .....	33
Figure 2-22: Simulation results for bursty traffic and URLLC air interface .....	35
Figure 2-23: Simulation results for uniform traffic and URLLC air interface .....	35
Figure 2-24: Simulation results for bursty traffic and medium air interface .....	36
Figure 2-25: Simulation results for uniform traffic and medium air interface .....	36
Figure 2-26: Simulation results for bursty traffic and air interface with low reliability .....	37
Figure 2-27: Simulation results for uniform traffic and medium air interface .....	37
Figure 2-28: Simulation results for correlated links and bursty traffic .....	38
Figure 2-29: Simulation results for correlated links and uniform traffic .....	38
Figure 3-1: Interdependencies between FM CFs at NSI and NSSI levels .....	40
Figure 3-2: Distribution area of NSSI C .....	41
Figure 3-3: Overall availability of the network given different redundancy schemes .....	44
Figure 3-4: Module-based shard [ODLSHARD] .....	45
Figure 3-5: ODL topology synchronisation .....	46
Figure 3-6: ODL install features .....	46
Figure 3-7: Example of akka.conf .....	47
Figure 3-8: Example of module-shards.conf .....	47
Figure 3-9: OpenDayLight curl data .....	48
Figure 3-10: OpenDayLight curl data output .....	48
Figure 3-11: OpenDayLight web GUI .....	49
Figure 3-12: OpenDayLight cluster monitor tool - election procedure of shared leaders .....	50
Figure 3-13: Data partitions and replication set .....	51
Figure 3-14: Cluster with 5 nodes .....	52
Figure 3-15: Cluster with 1 node failure .....	52
Figure 3-16: One node is down from partition perspective .....	52
Figure 3-17: Four nodes are down .....	53
Figure 3-18: ONOS clustering .....	53
Figure 3-19: Scalable controller framework .....	53
Figure 3-20: Scalable controller framework – evaluation scenario .....	54

Figure 3-21: Scalable controller framework – performance measurement .....	55
Figure 3-22: Map of mobility simulation .....	57
Figure 3-23: Markov chain to simulate the central cloud VNF outage .....	58
Figure 3-24: Cost/loss per day: simulation results .....	58
Figure 3-25: An example output of UE trace online prediction with LSTM network .....	59
Figure 4-1: General architecture of the Hamburg Sea Port use-case .....	61
Figure 4-2: Process for defining STZs within a 5G infrastructure .....	66
Figure 4-3: Process for selecting STZs based on templates .....	68
Figure 4-4: Security Trust Zone approach for protecting 5G network slices .....	70
Figure 4-5: Detailed STZ and data flows .....	70
Figure 4-6: Complete 5G-MoNArch security simulation testbed .....	72
Figure 4-7: Testbed deployment for simulating attacks .....	73
Figure 4-8: SthD configured at the STZm (using the Atos XL-SIEM GUI).....	73
Figure 4-9: DoS attack simulated with hping3 in Kali Linux .....	74
Figure 4-10: Network scanning attack simulated using Nmap in Kali Linux .....	74
Figure 4-11: Brute-force attack simulated using ncrack in Kali Linux .....	74
Figure 4-12: Script to simulate several attacks.....	74
Figure 4-13: Denial of Service events received by the SMm.....	75
Figure 4-14: Network scan events received by the SMm.....	75
Figure 4-15: Brute-force attack events received by the SMm.....	76
Figure 4-16: Alerts for attacks created with Kali Linux tools.....	76
Figure 4-17: Events received from the SthD to the SMm sent by different simulated sensors.....	77
Figure 4-18: Alerts generated by the SMm after correlating events from simulated sensors .....	77
Figure 4-19: First application – results of the proposed approach .....	81
Figure 4-20: First application – results based on four features .....	82
Figure 4-21: Second application – results for different behavioural groups .....	83
Figure 4-22: Architecture of the proposed methodology for anomaly detection .....	84
Figure 4-23: Frequency of each type of attack in the UNSW-NB dataset .....	85
Figure 4-24: The Architecture of the proposed anomaly detection methodology procedure.....	86
Figure 4-25: Heat maps of coefficient correlation prove the existence.....	87
Figure 4-26: The ROC curve per each type of attack.....	90
Figure 4-27: The throughput value on PDCP level for the first simulation .....	92
Figure 4-28: The throughput value on PDCP level for the second simulation.....	93
Figure 5-1: use cases derived and considered by x-domain and x-slice S&R.....	97
Figure 5-2: x-domain and x-slice S&R Management: actions performed.....	98
Figure 5-3: Message sequence chart for joint fault management and security management .....	99
Figure 5-4: Process for evaluating incidents and estimate the most convenient reaction .....	107
Figure 5-5: General process for mitigating security incidents .....	108

**List of Tables**

Table 3-1: Basic motion speed of different mobility classes..... 57  
Table 3-2: Mobility penalty factors in different areas..... 57  
Table 4-1: Network slicing security risks in Hamburg Sea Port use case ..... 63  
Table 4-2: Analysis of STZs vs 5G infrastructures ..... 66  
Table 4-3: Security Probes integrated in the 5G-MoNArch security testbed..... 71  
Table 4-4: Description of sub graph features that constitute the inputs ..... 84  
Table 4-5: Comparison between state-of-the-art and the proposed method..... 86  
Table 4-6: Coefficient of correlation among features with value close to 1 ..... 87  
Table 4-7: Overview of ANN model architecture, accuracy and precision ..... 89  
Table 4-8: Comparison of the experimental results in terms of precision % (and recall %)... 90  
Table 4-9: Corresponded attributes and estimated parameters..... 92  
Table 5-1: Incidents considered in the study ..... 100  
Table 5-2: Example of attacks analysis, mitigations, impact on resilience and on ..... 101

## 1 Introduction

This deliverable summarises the work conducted within the framework of the work package 3 (WP3) of 5G-MoNArch. It represents the continuation of the work presented in Deliverable D3.1 of the project [5GM-D3.1], in the sense that the contents provided in this deliverable are based upon the preparatory work conducted in the first year of the 5G-MoNArch project and reported in D3.1.

In light of the above, the focus of this deliverable is two-fold: i) It concerns the *evaluation of the concepts* proposed in D3.1 towards a *resilient and secure* operation of the network; ii) It includes *extending the concepts* proposed initially in D3.1 and complementing them with new approaches that provide a more effective and resource efficient performance. Such targeted network design spans across multiple domains and focuses in particular to the topics of i) Radio Access Network (RAN) reliability; ii) Telco cloud resilience; iii) Security. These three topic areas are in principle studied separately, due to the different nature of the techniques and analyses involved. However, besides individual investigations, WP3 also encompasses joint studies of the above elements within its framework. These joint studies particularly refer to common approaches towards network fault and security management, leading to interesting synergies between telco cloud and security.

In the context of assessing and extending the initial concepts provided in 5G-MoNArch D3.1 [5GM-D3.1] pertaining to the three study areas of WP3 of 5G-MoNArch, the following actions are taken.

- The conducted RAN reliability analysis concerns the concepts of macro diversity via data duplication and network coding. In the evaluation framework of this deliverable, these are assessed via simulations and analytical calculations, with respect to their ability to provide sufficient levels of resilience. In addition, a hybrid approach is proposed that is able to switch between data duplication and network coding depending on the requirements on reliability and latency, thereby combining the benefits of both techniques for a tailored application use.
- As regards the investigation on the telco cloud domain, additional concepts are integrated into the analysis of telco cloud resilience presented in D3.1. Such additional concepts pertain to correlating the root causes of network faults in slice-aware environments, extending the initial controller scalability work, as well as evaluating the cost of context-aware NF migration, as part of the “5G Islands” approach introduced in D3.1.
- With reference to the security domain, the initial concepts presented in D3.1 are extended towards a threat analysis on the main 5G components, complemented by a simulation-based analysis where the concepts of security trust zones and network behaviour analytics are assessed.
- Finally, the concepts of telco cloud resilience and security are studied in a joint framework. In this regard, synergies are identified when such concepts are jointly deployed in a telco cloud environment, along with respective virtual resource allocation considerations.

### 1.1 Resilience and security as an end-to-end concept

As its name implies, the two major pillars of WP3 of 5G-MoNArch are resilience and security. As also explained in [5GM-D3.1], these two conceptual pillars are treated in a common framework in 5G-MoNArch due to the common deployment, service and application characteristics they are associated with, resulting in common design approaches in network slicing environments. The notion of resilience in this sense is treated as a major conceptual element that enables a reliable operation of two major network components, namely RAN and telco cloud. In this regard, the term resilience in WP3 is used to refer to the technical work towards both a reliable operation of the RAN (which represents the first out of the three major topics of WP3, as described above) and a resilient operation of the telco cloud (which represents the second major topic of WP3).

The above consideration renders resilience an end-to-end concept, due to the multiple network domains it comprises as well as their interdependencies for providing an overall resilient service. Indeed, it is generally expected that the corresponding services of RAN reliability, telco cloud resilience are typically seen from an end-to-end perspective. This implies that the developments and analysis carried out in a domain-specific fashion (that is, in certain parts of the network such as RAN and telco cloud) should be studied together, to the largest possible extent, paving thus the way for an end-to-end approach. It is also

important to note that such end-to-end approach is conceptually related to the notion of *availability*, which refers to the percentage of time a specific service (seen from the end-to-end perspective) is available to the end user. In other words, when dealing with a combined notion of a reliable operation of both the RAN and the telco cloud, it is the time percentage of the provided error-free service that counts, since this is the Key Performance Indicator (KPI) measure that can better capture such end-to-end effect.

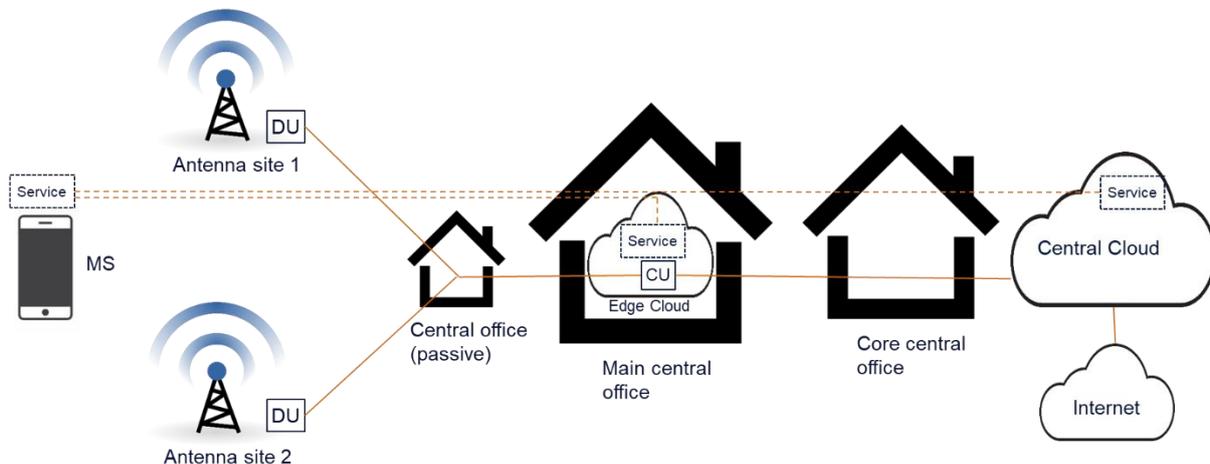
Besides resilience, the concept of security is also highly related with an end-to-end consideration. Such end-to-end aspect involves a detailed security analysis for the respective network elements, spanning across the deployed end-devices, network elements specifically used in 5G deployments, as well as an analysis pertaining to slicing-specific issues.

In the following, the notions of end-to-end availability (as this is enabled via RAN reliability and telco cloud resilience as described above) and end-to-end security are further elaborated in the respective sections 1.1.1 and 1.1.2. Then, a brief description of the role of resilience and security in the overall 5G-MoNArch architecture follows, along with a discussion on the inter-relation between WP3 and WP6 of 5G-MoNArch, pertaining to extending the evaluation of the WP3 concepts to a wider scale.

### 1.1.1 End-to-end availability enabled by RAN reliability and telco cloud resilience

Availability is an important property of 5G networks, as documented in [3GPP 22.261]. In a rough definition, it refers to the time that a particular service is provided uninterrupted to an end-user device. Such end-device could be, for example, a sensor, a smartphone or a car, which sees a particular service from an end-to-end perspective in the sense that if such service is interrupted it does not make any difference to the device if the cause lies within the RAN or telco cloud domain. As a result, the 5G network must ensure that all individual components required to access this service operate in a reliable manner.

In more detail, Figure 1-1 shows a typical 5G network architecture. A Mobile Station (MS) accesses a service, which is hosted at either an edge or a central cloud.



**Figure 1-1: Typical 5G network architecture**

The communication between the MS and the service takes place via the following entities:

- 1) The wireless radio channel towards one or multiple antenna sites
- 2) The radio equipment installed at the antenna sites - the so-called Distributed Units (DUs)
- 3) A fibre-optical network connecting the DU to the edge cloud
- 4) The Central Unit (CU) of the radio network, which runs as a Virtualised Network Function (VNF) within the edge cloud
- 5) The VNFs of the 5G core network and the service itself, residing either
  - a. at the edge cloud as well,

- b. at the central cloud (in this case an additional fibre optical network towards the central cloud is used) or
- c. distributed over edge and central cloud

Within this list, the entries 1 to 4 represent the RAN. Achieving a reliable communication via the wireless radio channel is a challenging task. Solutions for RAN reliability are presented in Chapter 2. Entry 5 is referred to as telco cloud, as it represents a telecommunication network in a virtualised and cloudified environment. Approaches to enable a reliable operation of the telco cloud are subject of Chapter 3.

### 1.1.2 Security as an end-to-end concept

In our effort to define an end-to-end (E2E) security approach that applies to 5G networks, various security considerations for the main as well as peripheral 5G components should be involved. In this regard, an overview of the main security areas involved in 5G networks is presented in Figure 1-2.



Figure 1-2: Main security areas in a 5G network

With reference to Figure 1-2, the devices refer to any type of network peripheral used as a transceiver, ranging from typical handheld devices such as smartphones and tablets, to devices placed in fixed locations such as sensors. The term “5G network” is a broad term that denotes all such elements of the 5G network that are susceptible to potential threats. Finally, the last term refers to all components that are associated with a slice-specific network operation, where the concepts of network virtualisation and software-defined networking are also taken into consideration.

The above combined analysis consists the major element for a holistic security study that applies in principle to every 5G network. In the context of 5G-MoNArch, such security analysis is tailored for the Hamburg Smart Sea Port use case, where certain devices, network elements as well as network slicing aspects are deployed. An elaborated security analysis of the Hamburg Smart Sea Port use case is provided in Chapter 4 of this deliverable.

## 1.2 Resilience and security as part of project-wide study and evaluation

WP3 of 5G-MoNArch captures the technical effort conducted towards a resilient and secure operation of 5G networks, by means of the respective RAN reliability, telco cloud resilience and security enablers. Nevertheless, such analysis is conducted not in a standalone fashion but instead in a project-wide study. In particular, such project-wide study refers to the fact that the **WP3 enablers are developed as part of the overall 5G-MoNArch architecture**, as this is defined in WP2 of 5G-MoNArch and documented in [5GM-D2.2] and [5GM-D3.1].

In the following, the resilience and security enablers considered in the framework of WP3 are studied with respect to their mapping to the 5G-MoNArch architecture, introducing thus interactions with the WP2 of 5G-MoNArch. Besides the first year of the project, however, where such architectural interactions were established, in the second year of 5G-MoNArch a project-wide evaluation campaign has been additionally addressed. Such extension of the assessment level of the WP3 enablers allows that they span *beyond the typical short-scale scenarios and are thus suitable for a project-wide assessment*. This implies that, in conjunction with their integration in the overall architecture, WP3 enablers fit the

scope of 5G-MoNArch in terms of their ability to meet the project's main KPI requirements. This broad activity further allows for an interaction between WP3 and WP6, by means of direct **exploitation of the WP3 results in wider, more realistic scenarios considered in WP6**, and their further assessment in project-wide evaluation campaigns.

These two project-wide aspects of WP3 enablers are elaborated separately in the ensuing two sections. In the first part, the potential architectural interaction between the enablers considered in WP3 and their inter-relation with other enablers developed in other work packages of 5G-MoNArch is discussed. Such architectural integration comprises part of joint work between WP2 and WP3 of 5G-MoNArch, including the common approach between resilience and security. In the second part, a wide-scale evaluation of the proposed enablers is put forward. Such work is carried out jointly with WP6 of 5G-MoNArch, such that the evaluation of WP3 enablers fits to a project-wide evaluation concept.

### 1.2.1 WP3 enablers in 5G-MoNArch architecture: interaction with WP2

Figure 1-3 depicts the 5G-MoNArch architecture, modified such that the role of the WP3 enablers is highlighted. Figure 1-3 provides an aggregated view of WP3 along its entire duration since the start of 5G-MoNArch, in the sense that all enablers discussed so far are included. This Figure is used as reference point when referring to the role of WP3 in the overall 5G-MoNArch architecture, hence it is used extensively throughout this document, particularly when a detailed explanation of the technical WP3 enablers is provided in the subsequent sections.

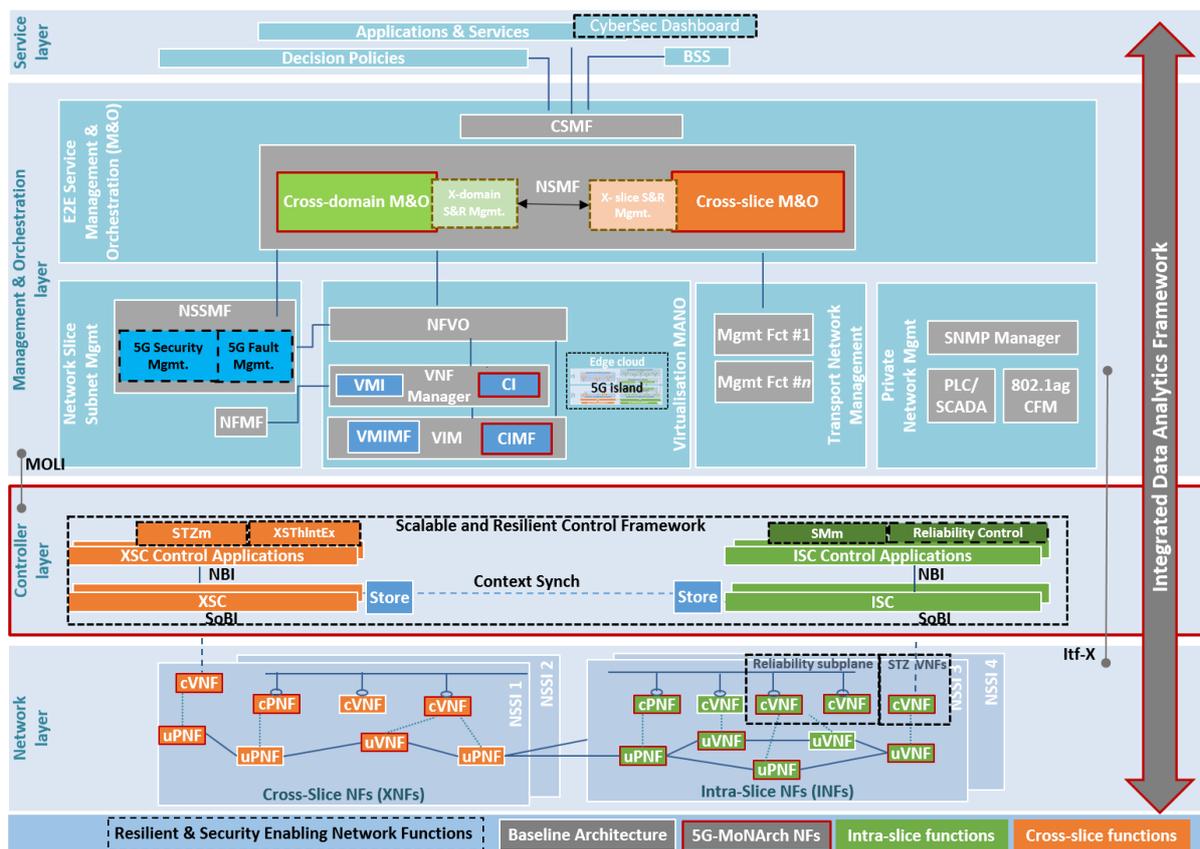


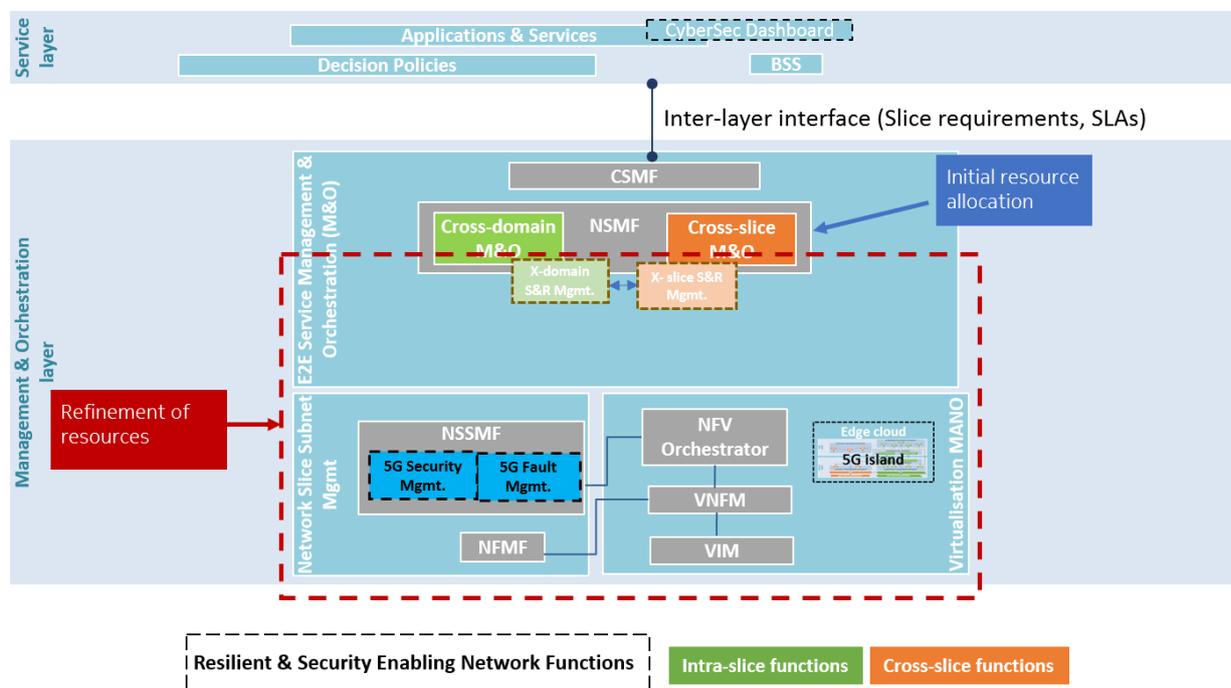
Figure 1-3: 5G-MoNArch architecture enriched with resilience and security (WP3) enablers

It is important to note that Figure 1-3 represents an elaborated version of the architecture picture defined in [Figure 2-2, 5GM-D2.3]<sup>1</sup>, emphasising thus the fact that the **WP3 enablers represent an instantiation of the 5G-MoNArch architecture**, as this is defined in WP2 of 5G-MoNArch. In fact, Figure 1-3 provides the reader with an overview on how the WP3 technical modules are built on top of the WP2

<sup>1</sup> 5G-MoNArch deliverable D2.3 [5GM-D2.3] is under preparation stage at the time this deliverable is finalised. Specific reference to [5GM-D2.3] material (e.g., figures, tables) may be subject to editorial amendments.

architecture, clarifying thus their role in the architecture as well as potential interactions with other 5G-MoNArch modules. The WP3 enablers depicted in Figure 1-3 include not only those discussed in this document, but also enablers which were reported in [5GM-D3.1]. As can be seen, the WP3 enablers span across all four considered architecture layers, namely service, management and orchestration, control and network layer. The network functions corresponding to the WP3 enablers are placed within the appropriate boxes, indicating their role in the overall architecture. Moreover, the involved interfaces are marked with respective signs (i.e., connecting lines and arrows), underlying thus the fundamental architectural aspects. Further details on the architecture role of the WP3 enablers are available in [Section 6, 5GM-D3.1].

A particular example of interaction of WP3 with the 5G-MoNArch architecture and the modules developed in WP2 is depicted in Figure 1-4. The figure highlights the network functions involved in the joint security and fault management considerations for resource optimisation, elaborated in WP3 and described in detail in Section 5.1. Such resource optimisation requires strong interaction between functions developed within WP3, such as x-domain and x-slice S&R Management and the rest of the 5G-MoNArch architecture, specifically *Virtualisation MANO* and *Network Slice Subnet Management* functional blocks.



**Figure 1-4: Interaction of WP3 enablers in the Management and Orchestration layer and corresponding processes involved**

### 1.2.2 Project-wide evaluation of WP3 enablers: interaction with WP6

As indicated in the first paragraph of this section, the focus of this deliverable is on providing further details on the enablers for resilience and security as well as the insights on their evaluation. We refer to the evaluation of enablers done within WP3 as small-scale evaluation. This type of evaluation considers the performance of each enabler individually in the context of the WP3. Such enabler-specific evaluations can be regarded as building blocks of the overall end-to-end large-scale evaluations that will be performed within WP6 of 5G-MoNArch. The insights of the enabler-specific evaluations are intended to be fed to WP6, such that they are utilised as a baseline for building the large-scale evaluation methodologies.

It is noted that by the end of the WP3 framework the interrelation between WP3 and WP6 has been established, and the initial exchange of evaluation insights from selected WP3 enablers has taken place. In particular, the data duplication technique for increased RAN reliability and impact of redundancy on

telco cloud availability has been evaluated and the insights have been incorporated into the large-scale evaluation development in WP6. In addition, the input from the telco cloud availability analysis has been introduced to WP6, leading to large-scale evaluation results that apply to wider, more realistic scenarios. In this respect, such large-scale evaluation considers, besides the technical evaluation, economical evaluation aspects as well. Further details on such project-wide evaluation of WP3 enablers are anticipated to be available in the project's final report on architectural verification and validation, documented in deliverable D6.3.

### ***1.3 Structure of the document***

The remainder of this document is structured as follows.

Chapter 2 contains an evaluation of the RAN reliability approaches conducted within the framework of WP3 of 5G-MoNArch. Specifically, such evaluation refers to a simulation analysis of data duplication, including the architectural implications of the considered approach, as well as to a simulation-based analysis of network coding approaches designed to increase the RAN reliability levels.

In Chapter 3, an analysis of the approaches directed towards telco cloud resilience is presented. In particular, the effect of redundancy in the form of spare telco cloud resources is evaluated, along with a root case identification of faults. In addition, Chapter 3 contains an analysis of the solution that leads to augmented scalability levels of the controller, by facilitating the adding and removing the number of nodes in the controller cluster. Furthermore, Chapter 3 includes an evaluation analysis pertaining to the concept of 5G Islands, where the migration cost and outage loss for context-aware network function migration is assessed.

A security analysis that relates to the 5G-specific characteristics of 5G networks is presented in Chapter 4, along with an elaborated view on the threat analysis of the Hamburg Sea Port testbed. Such 5G threat analysis spans across the main elements of a 5G network, namely the devices, network infrastructure elements, along with slice-specific aspects. Chapter 4 additionally provides a report of a simulated study on the potential threats of the 5G network, together with the corresponding detection mechanisms, thereby allowing for an assessment of the security trust zone approach. In a similar context, a graph-based network behavioural analysis is also presented as part of Chapter 4, thereby accounting for a complementary method for identifying behavioural anomalies within a 5G network setup.

Chapter 5 contains a joint analysis between the concepts of resilience and security in 5G networks. Specifically, Chapter 5 focuses on common resource allocation issues resulting from the co-existence of resilience and security features within a common network slice. Network synergies are identified, focusing on the interaction between fault management procedures and security management approaches. In this framework, resource optimisation considerations pertaining to such joint approach are put forward. Moreover, in Chapter 5 the effect of security threats to the 5G resources is discussed, with special focus on network security aspects pertaining to the Hamburg Smart Sea Post use case scenario. Finally, Chapter 6 summarises the deliverable and puts the contributions of WP3 into the overall framework of 5G-MoNArch.

## 2 RAN reliability approaches

As introduced in Section 1, a high availability is a key requirement for 5G networks. More specifically and as described in Section 1.1.1, this requires a resilient operation of the Telco Cloud (which is the objective of the methods and approaches that are part of Chapter 3) and a reliable operation of the Radio Access Network (RAN) that is subject of this section.

In this direction, and as documented in [5GM-D3.1], within the framework of 5G-MoNArch two approaches for increasing the reliability of a Radio Access Network (RAN) are proposed and studied: Data Duplication and Network Coding. Within to the overall architecture shown in Figure 1-3, each of these schemes consists of an intra-slice Network Function (NF) in the reliability subplane and a corresponding control layer application (within “Reliability Control”).

Data duplication uses the redundant transmission of duplicate packets over the radio, by means of transmitting the same message via two transmitting nodes, resulting in a reduced packet error probability. The concept of this scheme is described in detail in [5GM-D3.1] whereas Section 2.1.1 of the present document provides a concise overview.

In contrast to data duplication, Network Coding (NC) is a broad concept which can be utilised in different ways. Section 2.2 shows how it can be applied to send re-transmissions with an increased efficiency, which can then be converted into an increased reliability. Section 2.3 introduces how NC can be used in a similar manner as data duplication (i.e. to reduce the packet error probability by adding redundancy), thereby discussing and evaluating a hybrid scheme which exploits the benefits of both schemes in certain performance regions. Such hybrid scheme therefore can be seen as a combination of data duplication as described in Section 2.1 and network coding for increased redundancy, as introduced in Section 2.3.

### 2.1 Data duplication as a RAN reliability approach

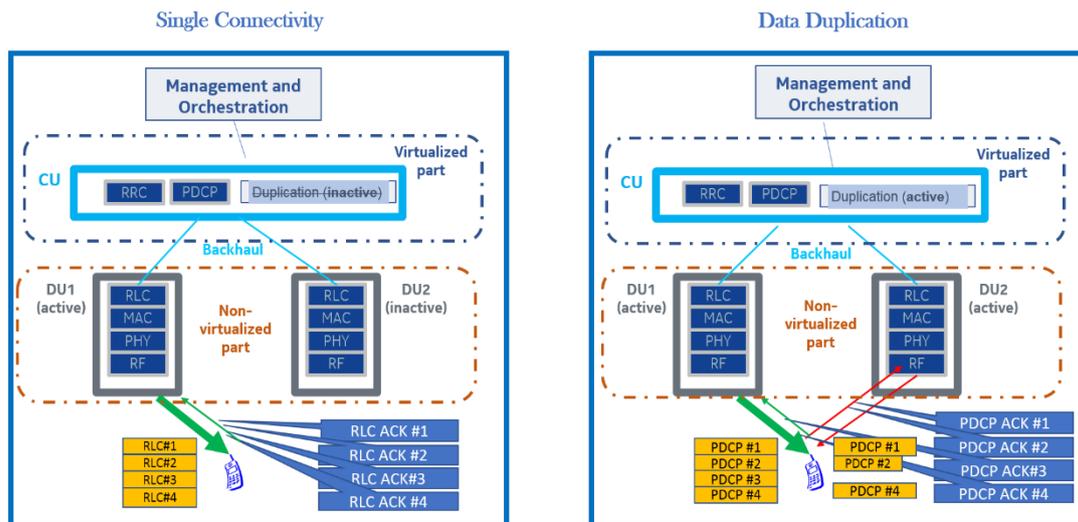
Data duplication is a relatively recent technique which has been proposed as a means to increase the RAN reliability of 5G communication networks [A18], [RV18]. The main principle of data duplication is the enabling of redundant transmissions at the air interface of the RAN, such that the detrimental effects of fading are tackled and thereby the probability of correct packet delivery to the terminals is increased.

Nevertheless, the application of data duplication in 5G networks brings about design challenges related to the coordination of duplicate packets at the RAN. In this regard, a data duplication approach has been proposed in the 5G-MoNArch framework [5GM-D3.1], where the benefits of such approach in specific implementation environments were discussed. In the following, the studied data duplication scheme is revisited for the sake of completeness. Then, the studied scheme is evaluated via a simulation-based analysis.

#### 2.1.1 On the considered data duplication scheme

In short, data duplication involves the redundant transmission of duplicate packets over the radio, by means of transmitting the same message via two transmitting nodes, resulting in a reduced packet error probability. More specifically, the considered scheme applies to the Central Unit (CU) – Distributed Unit (DU) architecture, which represents the architecture considered in 5G-MoNArch. It involves a special coordination scheme that handles the acknowledgments from the packets correctly received at the UE [5GM-D3.1].

The objective of this coordination scheme is, on the one hand, to ensure that duplicate packets are delivered to the UE, and on the other hand, to minimise the additional overhead of excessive duplicate transmissions. In order to achieve this goal, the use of Packet Data Convergence Protocol (PDCP) acknowledgments was proposed in [5GM-D3.1]. Specifically, PDCP level acknowledgments are introduced as a means to inform the respective radio transmission entities (i.e., the DUs) that a packet waiting at their buffer has been already delivered to the UE via another DU. Then, a DU receiving an indication that a given packet has been successfully delivered via another DU can discard that packet. An example of such process is illustrated in Figure 2-1: In this example a PDCP packet #3 is discarded from DU2, after an indication has been received by DU2 that this packet has been correctly delivered to the specified UE by DU1.



**Figure 2-1: Coordination of duplicated packets across different distributed units by means of PDCP acknowledgments**

It is important to note that this mechanism is not trivially applicable with existing technologies, which involve an acknowledgment feedback mechanism up to the Radio Link Control (RLC) layer, since the RLC packet sequences of different DUs are not necessarily identical to one another. In other words, the RLC packet numbers of, e.g., DU1 cannot be interpreted by DU2 (c.f. Figure 2-1), due to the different physical links involved in both cases. As a result, such coordination is handled by an upper network layer which can directly translate its packet sequence with that of the respective DUs. On the basis of the 5G-MoNArch architecture involving the split of network functions to the CU and DU network units, and in line with the 3GPP developments on network architecture [3GPP 38.801], the network layer handling such coordination is the PDCP layer located at the CU. This motivates the use of PDCP acknowledgments.

With reference to Figure 2-1, data duplication involves a modification of the RAN functionality when the system switches from the single connectivity mode (i.e., the traditional mode of operation involving a single transmitting node) to the data duplication mode. Besides the duplicate flow of the packets from the CU to the respective DUs, such modification is associated with a change on the acknowledgment messages exchanged between the UE and the network. Specifically, for the reasons mentioned above, in the data duplication mode PDCP acknowledgments are introduced, thereby replacing the RLC acknowledgments used in the single connectivity mode. It is worth noting that replacing the RLC acknowledgments finds also application to services with low latency requirements where RLC needs to operate in the unacknowledged mode for excluding the Automatic Repeat Request (ARQ) latency from the overall transmission delay (see, e.g., [3GPP 38.300]).

The activation of the data duplication and thereby the switching from the single connectivity to the duplication mode is assumed to follow the commands arriving from the management and orchestration layer, c.f. Figure 2-1. The CU handling the coordination of duplicate packets is assumed to occupy virtualised resources, in accordance with a cloud-based RAN deployment. At the non-virtualised part of the RAN, an activated duplication mode implies additional resource consumption as well as modified scheduling rules, which stem from the additional introduced traffic. That is, besides the additional computational resources occupied at the CU for handling the coordination of duplicate packets, the lower and non-virtualised layers of the RAN need to deal with an increased traffic in the data duplication mode. Such additional traffic practically equals double the traffic of the UEs with services requiring data duplication.

In technical terms, the additional overhead caused by the data duplication mode is anticipated to cause a degradation of some KPIs, those listed in [5GM-D3.1]. Overall, one would expect that there exists a trade-off between some of the project's relevant KPIs associated with the activation of data duplication. In particular, KPIs related to reliability such as packet error rate and overall latency are anticipated to improve with data duplication, whereas KPIs related to data rate transmission are expected to deteriorate, owing to the less efficient use of the resources. This leads to the need for an evaluation campaign of data

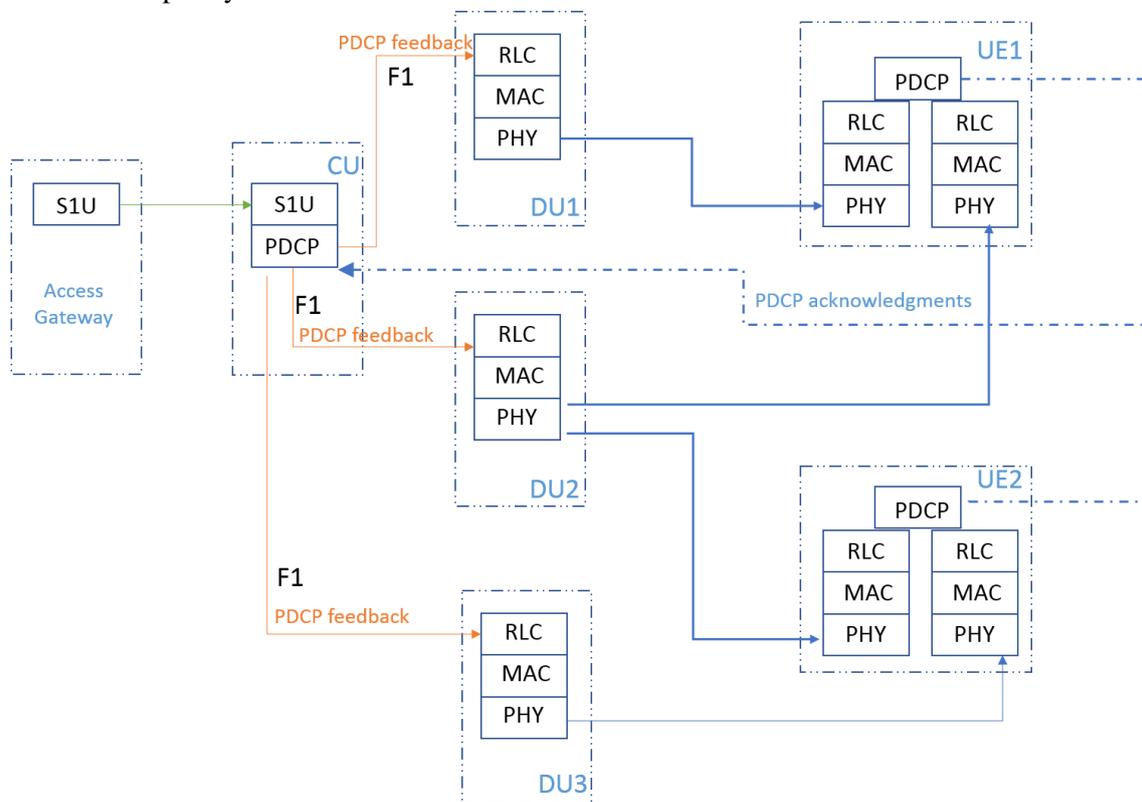
duplication, where the benefits and drawbacks with respect to the above KPIs would be quantified. To this end, a simulation analysis of data duplication was conducted, which is analysed in the remaining part of this section.

## 2.1.2 Simulation analysis

The conducted simulation analysis involved the data duplication scheme described above and analysed in [5GM-D3.1] in detail. In this section, the simulation setup is explained, followed by an analysis of the obtained results in the ensuing section.

### 2.1.2.1 Architectural setup

The architecture considered in the simulation campaign is according to the CU – DU model, which represents the architecture considered in 5G-MoNArch. This architecture involves the use of two separate network entities, namely the CU and the DU, where different layers of the protocol stack are carried out. These two entities are connected to each other via an interface which is referred to in the 3GPP standards as the F1 interface (see, e.g., [3GPP 38.470]). The F1 interface is in principle configurable with respect to capacity and delay. However, it should be noted that in the initial simulation campaign considered in this deliverable, the capacity and delay values of the F1 interface were assumed constant for simplicity.



**Figure 2-2: Exemplary view of the architecture considered in the data duplication simulations**

Figure 2-2 depicts the protocol stack considered in the simulations. In particular, the use of multiple CUs and respective DUs has been included, where the multiple DUs are connected per CU and the CUs are directly connected to the Access Gateway (AGW) at the core network. The PDCP functionality is carried out at the CU, while the RLC, MAC, and PHY functionalities are executed at the respective DUs.

Every time the CU receives a downlink packet from the AGW for a given UE, it directs a replica of this packet to all DUs which are connected to this UE. The DUs then apply the respective RLC layer processing to the replicas they are handling and transmit the packets independently from one another. At the UE receiver, the packets are received separately and are passed to the receiver PDCP entity,

where the replicas are decoded. Then, if a PDCP packet has been successfully received, in the sense that the decoding was correct, an acknowledgement message is generated and fed back to the CU, which then informs the RLC entities at the corresponding DUs to proceed to the next packet.

In fact, the use of the PDCP level acknowledgments here is to account for coordinated duplicate transmissions. This is particularly useful for imbalanced links, i.e., for the case where the links involved are substantially different in terms of the received signal strength. As analysed in [5GM-D3.1], the proposed data duplication method minimises unnecessary transmissions of packets which have already been received via alternative links, and “pushes” towards transmitting packets which have still not been received.

For a better understanding of the benefit of the proposed method for enhancing the efficiency of data duplication, let us consider the following example. Suppose that the UE is simultaneously connected to DU1 and DU2; DU1 has a strong link to the UE, while the corresponding link from DU2 to the UE is relatively weak. This implies that different modulation and coding schemes (MCS) are deployed in the PHY layer of the two links, such that the link between DU1 and the UE conveys more information per unit time than the link from DU2 to the UE. This further implies that packets which have been already correctly delivered to the UE via DU1 are still under process in DU2, i.e., they can only be delivered to the UE at a future time instance, via DU2. The proposed method that involves the use of PDCP packet acknowledgments increases the efficiency of data duplication in utilising the available resources. As such, a PDCP packet which has been correctly received by the UE via DU1, will generate an acknowledgment message to the CU, which will then notify DU2 to discard such packet from the corresponding RLC entity. This in fact means that the weak PHY link (i.e., the PHY link between DU2 and UE) will only be used for those packets which failed to be transferred via the strong link (i.e., between DU1 and UE).

Such advantage of the proposed efficient duplication technique as described above is reflected into the overall delay in delivering PDCP packets correctly to the UE, as will be manifested in the ensuing section where the respective simulation results are shown. Of course, data duplication is associated with an inherent robustness against fading, which results in lower packet error rates as well as fewer radio link failures in scenarios with mobility, when compared to single connectivity approaches. The above two features of the proposed data duplication approach are highlighted by means of the respective KPIs, namely the delay on packet delivery and percentage of lost packets, as shown below.

### 2.1.2.2 Simulation setup

A RAN protocol layer simulator was developed, which involves simulating the PDCP, RLC, MAC and PHY layers of the protocol stack, using the architecture shown in Figure 2-2. The application layer is also included in the simulator, comprising of traffic sources and sinks of a given type. The considered MCS schemes are adopted from release 15 specification of new radio (NR) [3GPP 38.211]. A transmit time interval (TTI) length of 0.2ms was assumed, with 14 OFDM symbols per TTI. The carrier frequency was set to 3.5GHz, with a system bandwidth of 100MHz. The number of physical resource blocks (PRBs) was set to 10, with 132 subcarriers per PRB and a subcarrier spacing of 75kHz. The guard period was set to 0.87μs.

The setup involves simulating three outdoor cells, where the transmit power is set to 30dBm each. Within the coverage area of those cells, 56 UEs are assumed to move in a wrap-around fashion. Whenever the UEs reach the coverage area of neighbouring cells and if certain handover conditions are satisfied<sup>2</sup>, the UEs perform handovers, i.e., they switch their connection to the strongest cell. In case the UE remains for a sufficiently large amount of time without any sufficiently strong connection to the cells<sup>3</sup>, a Radio Link Failure (RLF) is declared. The considered propagation model is the model adopted in the standards [3GPP 38.901]. This includes the urban micro and urban macro propagation models (c.f. [3GPP 38.901, Table 7.2-1]), while outdoor line of sight (LoS) and non LoS (NLoS) conditions are

<sup>2</sup> The handover conditions involve a difference on the reference signal received power (RSRP) from neighboring cells at least 3dB and a certain time-to-trigger mechanism, however such mobility-related analysis is out of the scope of this document, hence such parameters are adopted here unaltered from the state-of-the-art.

<sup>3</sup> Similarly, as above, the conditions for declaring a radio link failure are out of the scope of this analysis and are adopted from state-of-the-art approaches.

selected on the basis on whether the link between the UE and the access point is blocked by an obstacle (for instance, a building, a tree, etc).



*Figure 2-3: Two-dimensional visualisation of the considered simulation setup*

Figure 2-3 provides a snapshot of the considered simulation scenario in two dimensions. Specifically, the three cells are depicted with distinct colours, and are assumed to extend to areas that resemble streets in an urban environment. The shade of the respective colours denotes the RSRP level, such that areas with, e.g., strong blue presence correspond to areas where the respective cell is the strongest cell. In this regard, the areas in dark colour represent buildings which cause attenuation [3GPP 38.901], [CEL10], as well as NLoS propagation characteristics for the links between a UE and a cell across them. Moreover, trees are assumed to be included in the streets (not visible in the 2-dimensional view), which cause additional NLoS effects.

The considered UEs are grouped into two major categories, namely pedestrians (marked with light blue cell-phone symbols in Figure 2-2), and vehicles (marked with orange car symbols in Figure 2-2). The pedestrians are assumed to move with a speed of 3Km/h, whereas the speed of cars is set to 30Km/h in the respective models. The traffic associated with such UEs is a constant bit rate traffic that corresponds to 200Kbits/sec.

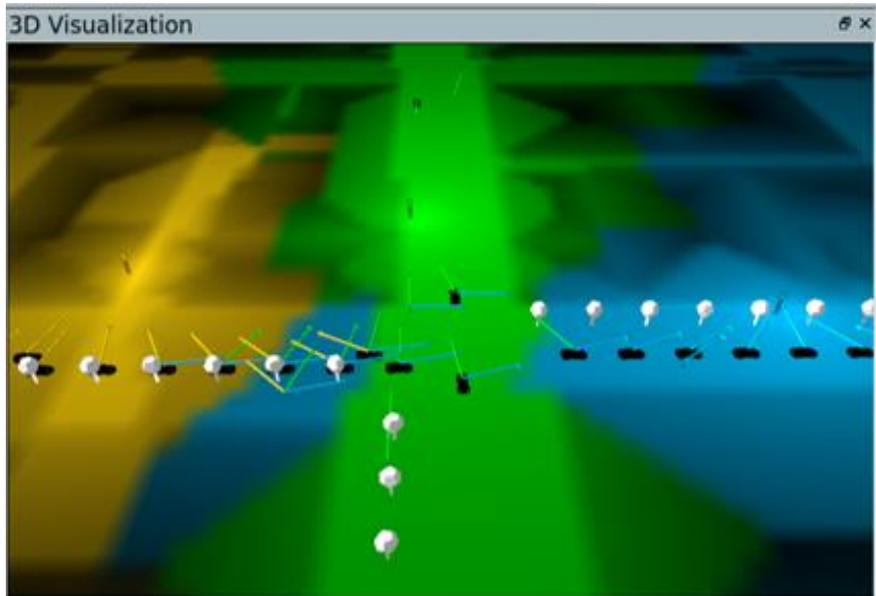
### 2.1.3 Obtained results

The obtained results focus on showcasing the performance of the proposed data duplication approach, on the basis of the aforementioned reliability-related KPIs, namely the percentage of lost PDCP packets and the delay on packet delivery. In addition to these KPIs, the simulation campaign provides insights on the overhead of the proposed approach to the throughput, as well as to the overall occupancy of the resources.

For the case of data duplication, an additional link selection mechanism was assumed, which compares the RSRP values of the nearby cells with that of the serving cell. As such, cells are added into the data duplication mode only if they are associated with an RSRP measurement which is at least as large as the RSRP from the serving cell minus a given offset value.

It is worth mentioning that the link imbalance threshold determines the conditions for activating data duplication and thereby which and how many links are used. A snapshot of the simulation campaign for the case where the link imbalance threshold is set to 9dB is provided in Figure 2-4. As can be seen, at the time this snapshot was taken, most of the UEs are connected to a single cell, some UEs are connected to two cells in data duplication mode, while few UEs are simultaneously connected to three cells. By

configuring such an offset value, which is dubbed here “link imbalance threshold”, interesting insights on the performance of data duplication are obtained, as will be shown in the following paragraphs.



**Figure 2-4: Three-dimensional visualisation of the considered setup, showing the simultaneous connections to the access points**

### 2.1.3.1 Investigation of the offered load

Since the performance of data duplication highly depends on the level of offered load to the simulated system, we distinguish between three different scenarios, namely the *low*, *medium*, and *high* load scenarios, which are analysed below.

#### 2.1.3.1.1 Low load scenario

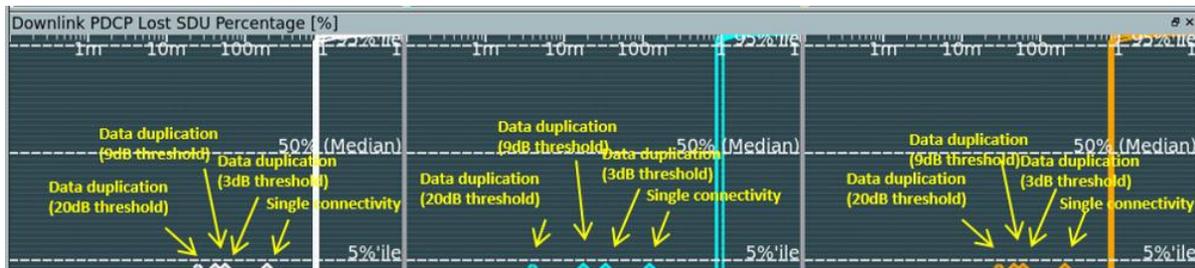
We first concentrate on the scenario where the generated traffic of the served users corresponds to a relatively low load. Specifically, the traffic in all 56 UEs is assumed to be exponential with an average of 128Kbps per device. The average burst duration equals 5sec and the idle duration equals 15sec. This corresponds to an overall load of  $128Kbps \times \frac{56}{4} = 1.8Mbps$  across the entire simulated area. The packet size has been set to 32 bytes to match the assumptions of [3GPP 38.913].

#### *Performance in terms of PDCP packet loss*

The anticipated benefit of data duplication with respect to reliability at the RAN level is reflected into the percentage of PDCP packets which fail to be successfully transmitted to the UE. This is illustrated in Figure 2-5, where the cumulative distribution function (CDF) of the lost PDCP service data units (SDUs) is depicted<sup>4</sup>. In Figure 2-5, the light blue lines correspond to the pedestrian UEs’ performance, the orange to the vehicle UEs’ performance, while the white colour corresponds to average performance values across all UE types. Moreover, the x-axis is depicted in logarithmic scale, using the “mili-” notation (e.g., “10m” denotes “ $10 \cdot 10^{-3}$ ”). It should be noted that since this figure refers to a random variable that reflects the percentage of lost packets, which in principle yields a large number of zero samples, the depicted lines overlap with one another on the zero value of the y axis. However, the mean distribution values per group are highlighted and marked with the respective symbol (triangle) per line. In Figure 2-5, the single connectivity case as well as data duplication with different values of the link imbalance threshold (namely 3dB, 9dB and 20dB) have been considered. As can be seen, increasing the

<sup>4</sup> Figure 2-5 and subsequent figures illustrate the percentage of lost PDCP packets, which is formulated by counting the percentage of binary variables (“ones or zeros”), indicating whether a packet is lost or not. This results in discontinuous CDF plots, with the respective lines showing a discontinuous jump from zero to hundred percent level.

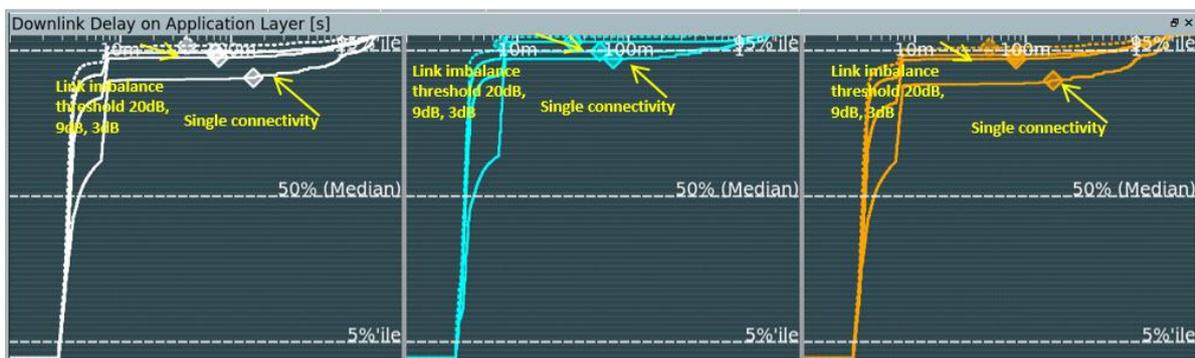
link imbalance threshold results in fewer lost packets, since for this case the inclusion of additional links in the multi-connectivity setup is facilitated. In particular, it is observed that without data duplication (single connectivity) approximately 0,25% of PDCP packets are lost, while with data duplication a loss percentage of 0,07% to 0,03% can be achieved, depending on the threshold value (20dB). Nonetheless, as will be shown later, this reduction on the lost packets comes at the cost of decreased throughput, since more resources are utilised for transmitting replicas of the same packet, decreasing thus the overall spectrum utilisation efficiency.



**Figure 2-5: Low Load Scenario: Percentage of lost PDCP packets for single connectivity (no duplication) and data duplication, under different assumptions on the link imbalance threshold**

### Performance in terms of delay of packet delivery

Similar observations related to the performance of data duplication are obtained from the application layer packet delivery delay, as depicted in Figure 2-6. In particular, it is noticed that a considerable reduction in the packet delivery delay is attained with the activation of data duplication. As expected, such reduction increases with the link imbalance threshold, since an increased value of such threshold results in higher chances that additional links are included, which leads to an overall faster packet delivery.



**Figure 2-6: Low Load Scenario: CDF of packet delivery delay at the application layer**

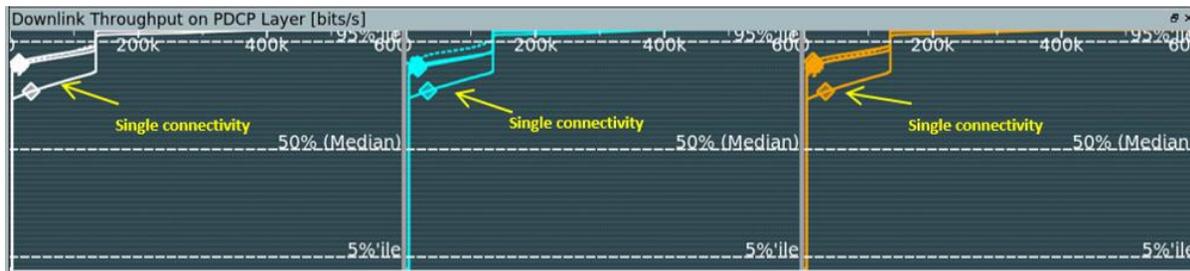
The observed average values (white marks) of packet delivery are in the range of 170ms for single connectivity, while such values drop to approximately 80ms to 40ms for a link imbalance threshold ranging from 3dB to 20dB. That is, by activating the data duplication mode a decrease on the packet delivery delay of approximately 50% can be achieved, even with relatively small values of the link imbalance threshold.

### Throughput performance

As expected, data duplication introduces a throughput overhead. Such overhead stems from the utilisation of redundant radio resources for the sake of reliability, thereby leaving less resources for new data transmission, which ultimately reduces the overall throughput.

The throughput reduction caused by data duplication is quantified in Figure 2-7. The main observation from Figure 2-7 is that the use of data duplication decreases the throughput by approximately 50% (that is, a decrease from 32KBps to 15KBps on average). It is further observed that such reduction does not highly depend on the link imbalance threshold. This is anticipated, since the low load scenario implies

that network resources are scarcely fully occupied, and hence plenty of resources are available to transmit duplicate packets. For the same reason, one may notice from Figure 2-7 that the non-zero throughput values are restricted to only a limited percentage of the simulation runtime.



**Figure 2-7: Low Load Scenario: CDF of throughput for single connectivity and data duplication, for variable values of the link imbalance threshold**

### Resource occupancy

Figure 2-8 depicts the resource occupancy of the simulated cells for the low load scenario. It illustrates the cases of single connectivity and data duplication, where for the latter the link imbalance threshold was set to 9dB. In fact, Figure 2-8 provides the following information:

- 1) Left part of Figure 2-8: In the left part of the picture, the average PRB allocation percentage (across the simulated time) is shown per cell. That is, the orange bars correspond to first cell; the blue bars to the second cell; the green bars to the third cell. The white bars correspond to the average resource allocation of the three cells (that is, the per-cell average of the per-time average of the PRB allocation percentage). In each category, the first bar corresponds to the case of single connectivity, while the right bar to the case of data duplication with link imbalance threshold equal to 9dB.
- 2) Right part of Figure 2-8: In the right part of Figure 2-8, the four lines correspond to the allocation of the cells (with the respective colours) plus the average PRB allocation (shown in white). All such lines show the time-specific resource allocation, for the time shown in the x-axis. Such time-specific allocation is used to extract the per-time average information given in the left part of Figure 2-8 for a sufficiently large time window. The vertical black line corresponds to the time when the switching from the single connectivity (i.e., no duplication) case to the case of data duplication takes place.

As can be seen, data duplication results in an increase of the overall usage of resources, as was initially anticipated. Depending on the cell deployment configuration, the increase on the resource occupation can vary. For instance, a larger increase for cell 2 is observed, whereas such increase for cell 1 is smaller. On average, a switch from the single connectivity to the case of data duplication with 9dB link imbalance threshold results in an increase from 12% to 21%, as indicated by the white part of the left graph in Figure 2-8.



**Figure 2-8: Downlink resource occupancy, measured in percentage of PRB allocation, for the cases of single connectivity and data duplication with link imbalance threshold equal to 9dB**

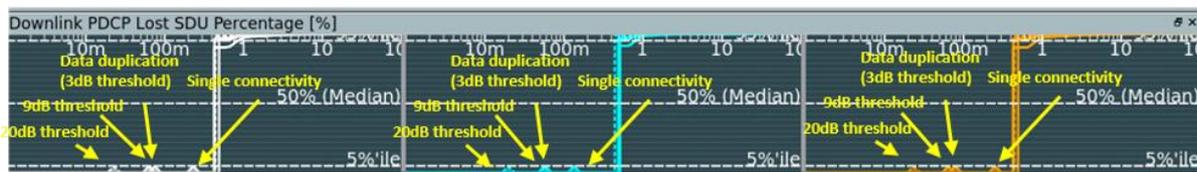
It should be noted, however, that such resource occupancy highly depends on the network traffic. In the considered example, the assumed traffic is relatively low, and corresponds to a constant bit rate of 200kbps. An increase of the considered traffic is expected to lead to higher levels of resource occupancy. An analysis that contains larger volumes of considered traffic is presented below.

### 2.1.3.1.2 Medium load scenario

In this and the following sections, the performance of data duplication under higher load assumptions is examined. This is expected to lead to a deteriorated throughput performance, since in a highly loaded system the additional resource consumption caused by duplicate transmissions has a stronger impact on system performance. In particular, the medium load scenario corresponds to a constant bit rate traffic of 200Kbps per device in all 56 devices, corresponding to an overall system load of 11.2Mbps.

#### *Performance in terms of packet recovery, packet delivery delay, and throughput*

Figure 2-9, Figure 2-11, and Figure 2-10 illustrate the percentage of lost PDCP packets, the delay at the application layer and the mean throughput, respectively, in the medium load scenario. In principle, as regards the relative performance of data duplication with respect to single connectivity, similar observations can be made as with the case of low load, in the sense that higher threshold leads to better packet loss and delay performance, yet to higher throughput.

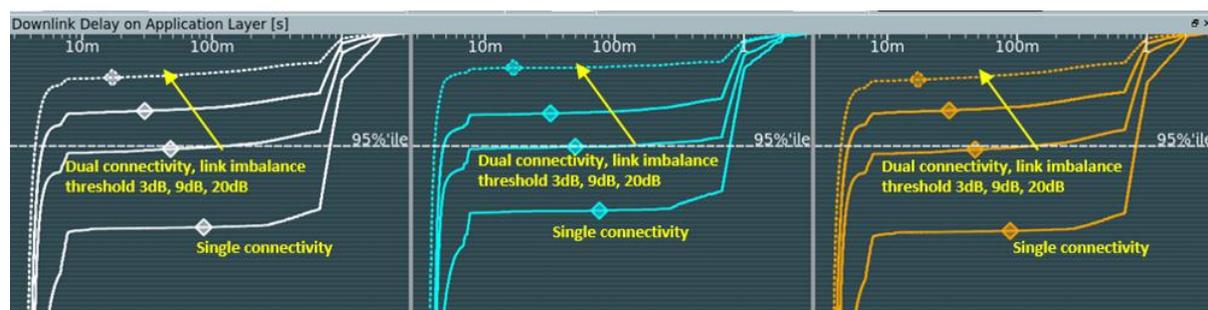


**Figure 2-9: Medium Load Scenario: Percentage of lost PDCP packets for single connectivity (no duplication) and data duplication, under different assumptions on the link imbalance threshold**

Specifically, Figure 2-9 shows that, with the exception of the 20dB threshold case, the medium load scenario leads to a larger percentage of lost packets than the low load scenario. Interestingly, we observe a high dependence of the mean throughput (c.f. Figure 2-10) as well as of the application layer delay (c.f. Figure 2-11) on the link imbalance threshold. Such effect is less visible in the low load scenario (c.f. Figure 2-6 and Figure 2-8), since in that case that the additional resources used for duplication rarely lead to a saturation of the available resources.



**Figure 2-10: Medium Load Scenario: CDF of throughput for single connectivity and data duplication, for variable values of the link imbalance threshold**



**Figure 2-11: Medium Load Scenario: CDF of packet delivery delay at the application layer**

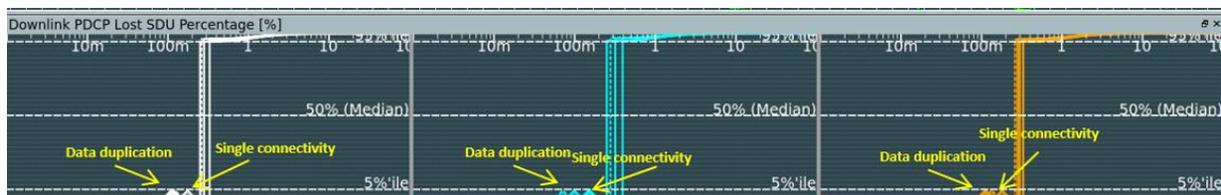
### 2.1.3.1.3 High load scenario

The high load scenario is examined here in an attempt to investigate the performance of data duplication in very highly loaded traffic scenarios. From another viewpoint, the analysis of this section pertains to a test of data duplication in situations where it is not anticipated to provide the desired performance. This is because in scenarios where the available resources are already saturated: the additional resource consumption overhead will severely deteriorate the overall performance.

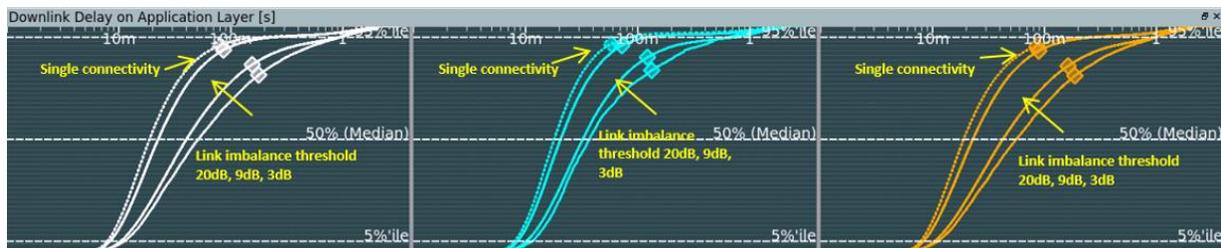
In this regard, the high load scenario investigated in this section corresponds to a traffic profile that is generated from a File Transfer Protocol (FTP) traffic of 5Mbytes every second per device, for all 56 devices, which leads to an overall load of 2.2Gbps.

#### *Performance in terms of packet recovery, packet delivery delay, and throughput*

Figure 2-12, Figure 2-13, and Figure 2-14 depict the percentage of lost PDCP packets, the delay at the application layer and the mean throughput, respectively, of the high load scenario. The main observations are as follows. First, data duplication demonstrates a limited capacity to recover lost packets, corresponding to a packet loss drop from approximately 0.2% to 0.1%. As shown in Figure 2-12, this effect hardly depends on the value of the applied link imbalance threshold.

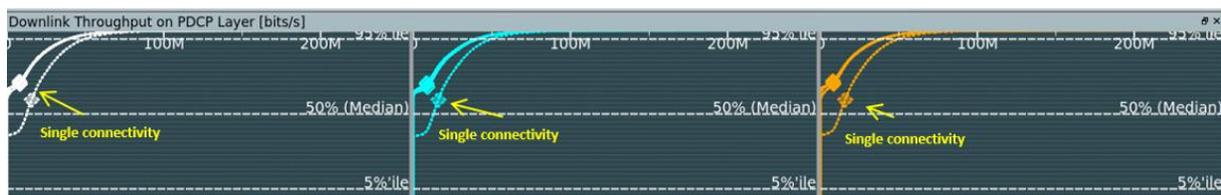


**Figure 2-12: High Load Scenario: Percentage of lost PDCP packets for single connectivity (no duplication) and data duplication, under different assumptions on the link imbalance threshold**



**Figure 2-13: High Load Scenario: CDF of packet delivery delay at the application layer**

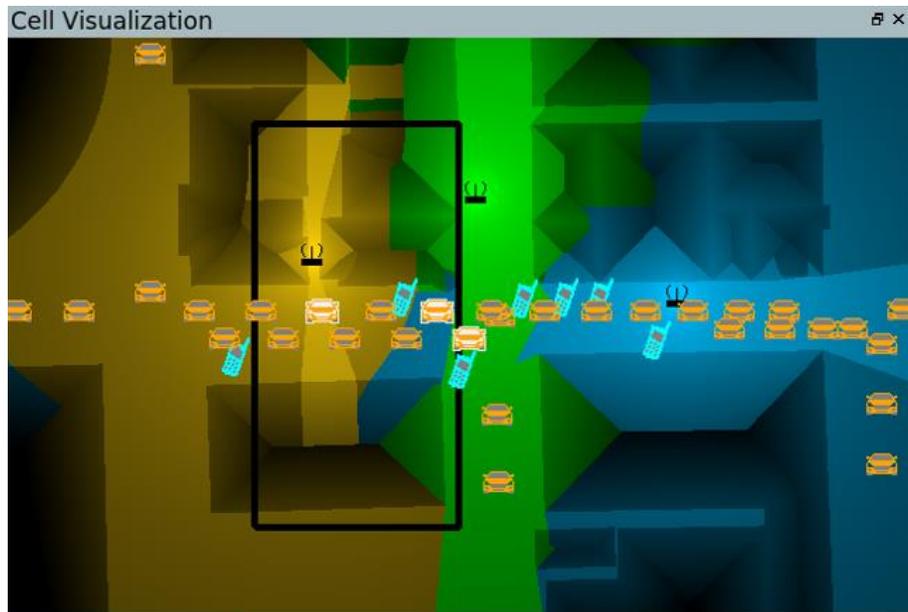
More importantly, the observed application layer delay does not improve with the use of data duplication in the high load scenario (c.f. Figure 2-13); it is further deteriorated as the value of link imbalance threshold grows large. This effect is explained by the resource saturation due to the high load, resulting in an inefficient use of resources when data duplication is activated. In a similar context, the mean throughput drops when data duplication is active (c.f. Figure 2-14), yet the effect of the link imbalance threshold is less visible.



**Figure 2-14: High Load Scenario: CDF of throughput for single connectivity and data duplication, for variable values of the link imbalance threshold**

### 2.1.3.2 On the performance limits of data duplication

So far, the simulation results correspond to the entire simulation area, as depicted in Figure 2-3 and Figure 2-4. From a close observation of the obtained results, one can infer that the strict requirements of ultra-reliable services in 5G, corresponding to 99.999% of correct packet reception, are not met. Nevertheless, given that such strict requirements correspond to mission critical services, it is natural to consider that such services are supported in limited areas only.



*Figure 2-15: The restricted area of the simulation scenario where the KPIs of interest are captured*

In view of this, the simulation campaign is repeated such that the performance of the UEs located within a restricted geographical area is captured. This is illustrated by the black box in Figure 2-15. For the same reason, only the low load scenario is considered, as an attempt to investigate the performance limits of data duplication in special areas. The results pertaining to the considered KPIs are depicted in Figure 2-16 and explained as follows.

**Packet Recovery via Data Duplication:** In certain restricted areas with sufficient coverage, data duplication leads to a substantial reduction of lost packets. The corresponding reliability levels can even exceed the target of 99.999% with proper configuration of the link imbalance threshold, as shown in Figure 2-16.<sup>5</sup>

**Delay Reduction at Application Layer:** Similar to the packet loss KPI, a proper configuration of data duplication can lead to a considerable reduction of the application layer delay as compared to the single connectivity (no duplication) case. As demonstrated in Figure 2-16, the 95%ile of the delay CDF can be as low as 3ms to 4ms for the case of 20dB link imbalance threshold. It is noted that while this is a relatively low value, it is still beyond the ambitious target of 1ms for 32-byte packets, as set in [3GPP 38.913].

**Throughput Overhead:** It is observed from Figure 2-16 that the relative throughput reduction due to data duplication is at approximately the same levels as with the case of non-restricted simulation area, shown in Figure 2-7, Figure 2-10, and Figure 2-14.

<sup>5</sup> For pedestrian UEs (blue lines) and link imbalance threshold 20dB, the number of lost packets was smaller than the measurement capability of the deployed simulation. This case is therefore not included in Figure 2-16.

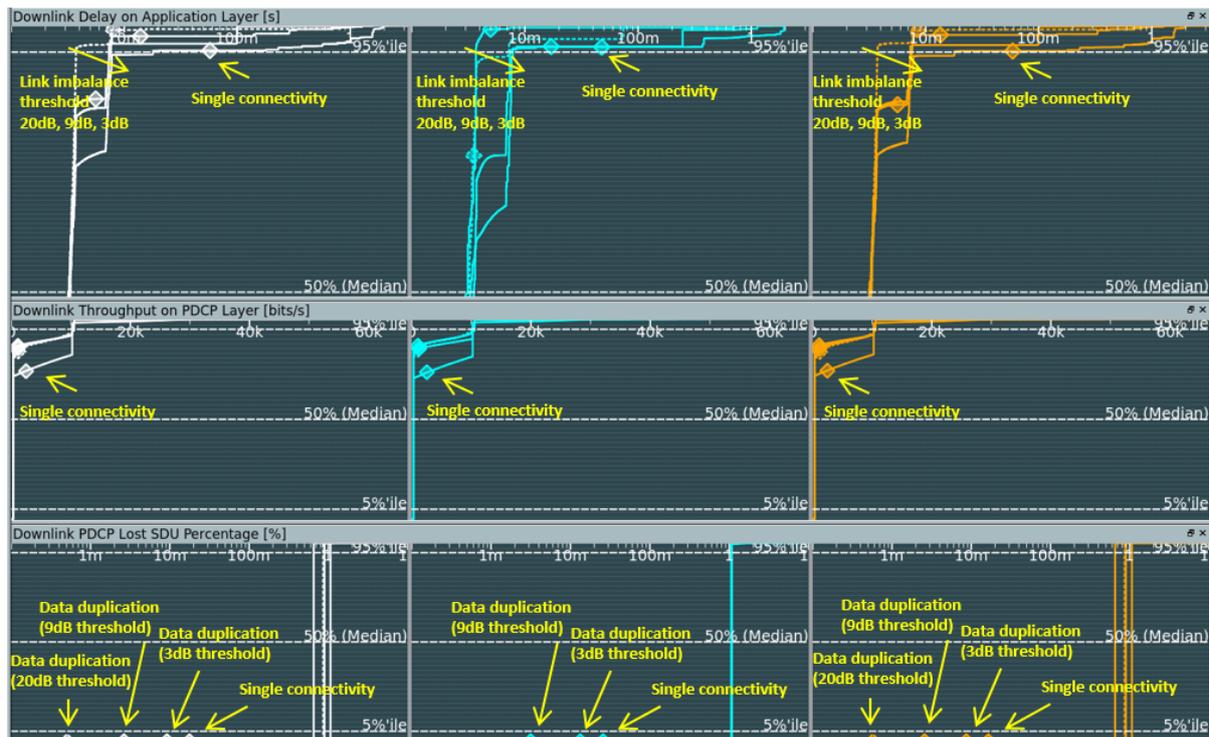


Figure 2-16: Performance in terms of the KPIs of interest within a restricted area

## 2.2 Performance and suitability assessment of network coding based multicasting approach

In [5GM-D3.1] a network coding approach was presented, which is suitable for downlink communications, particularly for multicasting and broadcasting scenarios. The main idea is to generate network coded packets depending on the ACK/NACK feedback, similar to [GT09] and [SI12]. The network coded packets are combinations of packets that are previously transmitted but erroneously received by at least one of the UEs. If a network coded retransmission is received by a UE, it can revert the network coding operation by using its previously error-free received packets, resulting in a reduced number of retransmissions. Theoretically achievable rates with this method are also given in [5GM-D3.1]. In the following, we will first discuss the suitability and integration of the presented approach to existing systems, and later we will show performance evaluations.

### 2.2.1 Integration and suitability

For the proposed approach to work, the following requirements have to be fulfilled:

- (R1) A multicasting setup needs to be available where the transmitted transport blocks (TBs) are decodable by at least two UEs.
- (R2) A feedback channel between UEs and the DU should exist.
- (R3) A buffer at the UE is needed, where the UE stores the received TBs for using them to decode the network coded packets.
- (R4) The packet IDs of the combined TBs need to be signalled to the receivers.

Moreover, according to the theoretical analysis in [5GM-D3.1] the improvement by this network coding approach becomes more visible for links with high error probability.

As a new multicasting service, LTE Rel. 13 introduced Single-Cell Point-to-Multi-Point (SC-PTM) technology. Fortunately, SC-PTM already fulfils some of the above requirements that we discuss next.

- SC-PTM uses the physical downlink shared channel (PDSCH) to transmit messages to multiple UEs, where the UEs use a group-radio network temporary identifier (group-RNTI) to decode the messages. Thus, a multicasting setup (R1) is already supported.
- It was shown in [3GPP 36.890] that SC-PTM could also exploit the unicast feedback for advanced link adaptation, if the number of UEs is small. This feature was finally not standardised in Rel.13, but this study shows the feasibility of a feedback channel for SC-PTM, i.e. (R2) is feasible and can be fulfilled.
- The presented scheme performs coding operation on the TBs, for which Hybrid-ARQ (HARQ) soft buffers already exists. We will show that by small modifications, the HARQ buffer can also be used for the purposes of network decoding, and thus (R3) is also fulfilled.
- In order for the receivers to determine which TBs are combined, new fields in the downlink control information should be included, such that (R4) can be fulfilled.

To sum up, the SC-PTM scheme can be taken as a baseline, and certain modifications can be made such that the requirements for the proposed scheme are fulfilled. In the following, we discuss the transmitter (DU) and the receiver (UE) side operations as an example with two UEs.

### ***Operations at the DU***

The DU starts encoding and transmitting TBs to a group of two UEs with the same group-RNTI in the conventional way. Let us call two of the TBs as  $TB_1$  and  $TB_2$ , and their respective HARQ processes as  $HARQ_1$  and  $HARQ_2$ . After transmission, the DU waits for the feedback of both transmissions (or continues with transmitting new TBs). If both TBs are not decoded by both UEs (if NACKs are received from both UEs), then the HARQ processes continues as in the unicast case. In case  $UE_1$  decodes only  $TB_1$  and  $UE_2$  decodes only  $TB_2$ , the DU generates a network coded packet for retransmission, which contains  $enc(TB_1) \oplus enc(TB_2)$ . Here,  $\oplus$  denotes the element-wise modulo two sum and  $enc(.)$  denotes the channel encoding (i.e. LDPC encoding in 5G NR) operation. Moreover, the DU also informs the UEs about the HARQ process IDs and the redundancy versions of both encoding operations using the Downlink Control Information (DCI). Moreover, the New Data Indicator (NDI) field within the DCI for both HARQ processes is set to 0 to avoid clearing the soft buffer at the UEs, such that the information in the buffer can be used for network decoding later.

### ***Operations at the UEs***

After reception of a signal, both UEs perform demodulation and de-mapping to obtain the log-likelihood ratio values (L-values) which are usually the input for the channel decoders. L-values are real numbers that represent the probability of each bit being zero or one. A positive L-value usually corresponds to a bit value of zero, and a negative L-value corresponds to a bit value of one. The magnitude of an L-value is related to the reliability of the decision.

If the received signal does not correspond to a network coded packet, the UEs write the L-values to the respective HARQ buffer as usual, i.e. for each received bit, the de-mapper produces an L-value, which is added to the value in the respective HARQ buffer. In case of a network coded packet, the L-values may not be written to the HARQ buffer directly. As explained before, a network coded packet contains the modulo-two sum of two-bit sequences, where one of the sequences is known to the receiver. Therefore, the modulo-two summation should be reverted in the L-value domain before the L-values are written to the HARQ buffer. Fortunately, this is a relatively simple task: one can basically change the sign of the L-values of the bits, for which the corresponding bit in the known bit sequence is a one. The L-values of the rest of the bits (corresponding to zeros in the known bit-sequence) can be left unchanged. Note that  $a \oplus b = a$ , if  $b=0$ , hence if the known bit is a zero, there is no need to change the sign of the L-values. After this inversion, the L-values can be written to the respective HARQ buffer, which is then used as the input for the channel decoder.

This reversion of the modulo-two addition in the L-value domain allows us to use the existing HARQ buffer for network decoding easily. Note that this operation supports both incremental redundancy and chase combining based retransmissions.

As a result, we can summarise the main required modifications as follows:

- A new mechanism at the DU to perform linear combinations on the packets depending on the ACK/NACK feedback.
- New DCI fields, indicating the HARQ process IDs and the redundancy versions of multiple TBs.
- A modified buffer management at the UEs, which reverts the modulo-two addition in the L-value domain before writing them to the HARQ buffer.

In the following, we show performance evaluations of the proposed network coding approach.

## 2.2.2 Performance evaluation

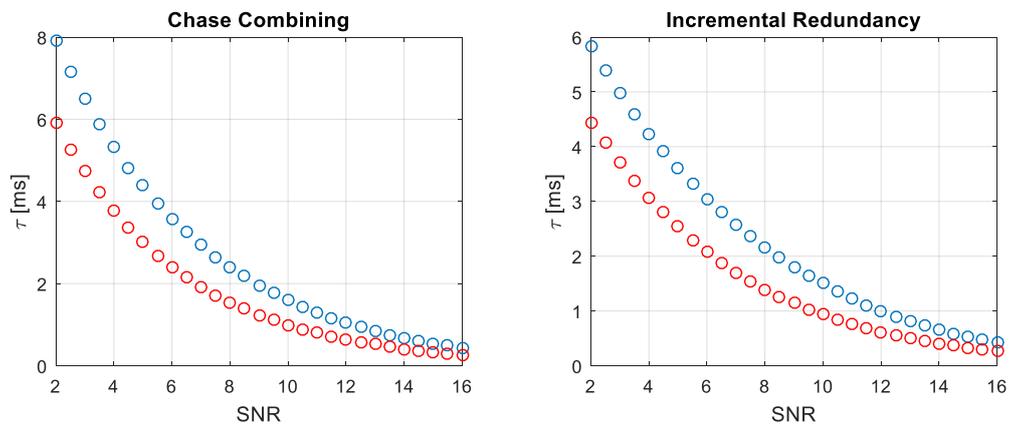
We evaluate the performance of the presented network coding scheme in terms of the average number of retransmissions (and latency) on Rayleigh block fading channels. To this end, we evaluate the outage probability  $P_m$  after  $m^{\text{th}}$  retransmission of the same packet, if the packet carries  $R$  bits of information per channel use. In [C06], a method is presented to obtain  $P_m$  both for Chase Combining (CC) and Incremental Redundancy (IR) based retransmissions. Accordingly, one can evaluate the theoretical outage probability independent of the used channel coding scheme.

$$P_m^{CC} = Pr \left[ R > C \left( \sum_i^m \gamma_i \right) \right]$$

$$P_m^{IR} = Pr \left[ R > \sum_i^m C(\gamma_i) \right]$$

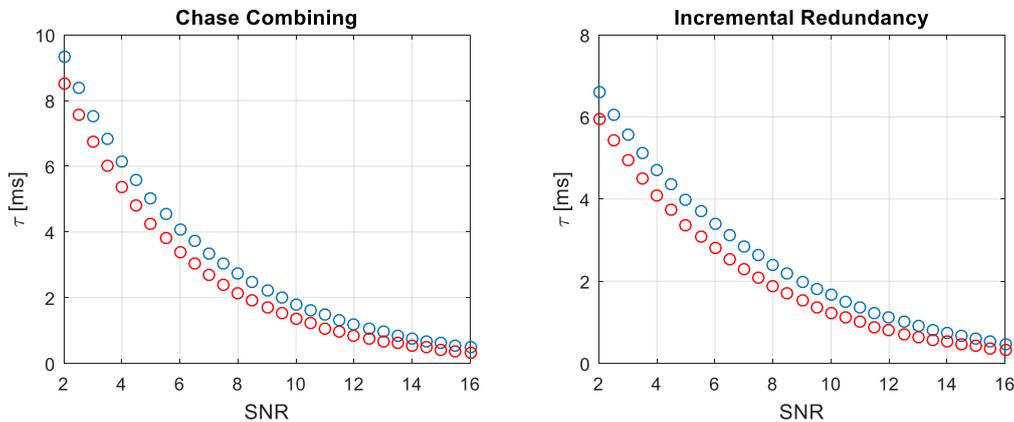
Here,  $C(\gamma) = \log_2(1 + \gamma)$  is the capacity formula, and  $\gamma_i$  is the SNR of the  $i^{\text{th}}$  retransmission. By using these formulas, we evaluate the average number of retransmissions with and without network coding by means of Monte-Carlo simulations, until no outages occur (i.e. by generating Rayleigh distributed channel realisations randomly for each transmission and checking how many retransmissions are needed until the summation in the equations becomes larger than the rate).

For an enhanced Mobile Broadband (eMBB) scenario with 15kHz sub-carrier spacing, we can assume that the time between each retransmission is roughly 3 slots, where each slot corresponds to 1ms. This allows us to translate the average number of retransmissions to average time in milliseconds between the first and last transmissions with and without network coding. Figure 2-17 depicts the performance of CC and IR for  $R=2$  bits/channel use in a balanced scenario, where both UEs experience the same average SNR, but are subject to different independent fading coefficients. We observe that for both IR and CC, network coding (as expected) reduces the number of retransmissions, resulting in a reduced average latency, as network coding combines multiple retransmission to a single packet. Note that this gain can be interpreted as lower latency for a target reliability, or better reliability at a given latency.



**Figure 2-17: Performance of the presented network coding approach (red curves) and the conventional multicasting approach (blue curves) on Rayleigh fading channels with balanced links with incremental redundancy and Chase combining based retransmissions**

As a second example, we consider a scenario with imbalanced links, where the average SNR between both UEs differ by 3dB, as depicted in Figure 2-18. We observe that the presented approach still shows improvements, however the performance gain compared to the conventional approach without network coding is reduced. These results indicate that the presented network coding approach is more powerful especially in scenarios with balanced links. Note that in a multicasting scenario, the DU may have the freedom to choose pairs of UEs (out of multiple UEs) for which network coded transmissions are performed. Picking UEs with relatively balanced links would be a good choice to obtain most gains from the presented network coding approach.

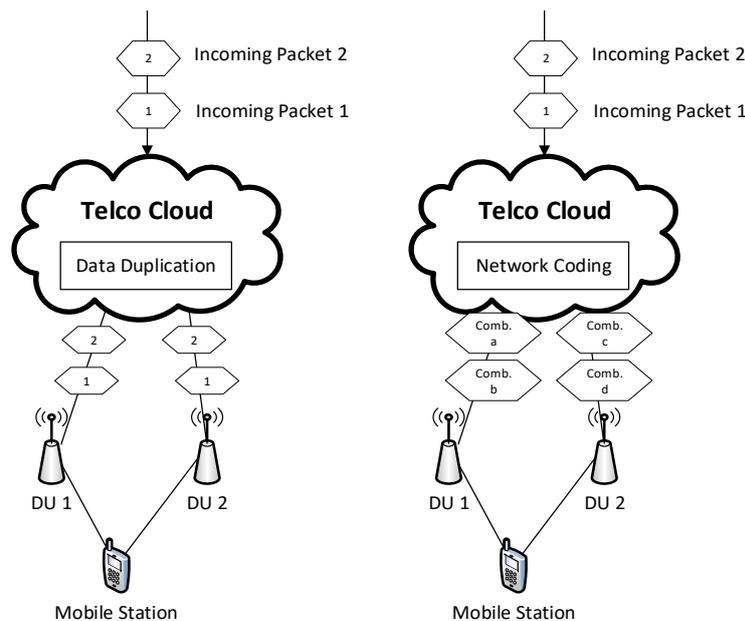


**Figure 2-18: Performance of the presented network coding approach (red curves) and the conventional multicasting approach (blue curves) on Rayleigh fading channels with imbalanced links, where the average SNR of the first UE is 3dB larger than the average SNR of the second UE**

### 2.3 The hybrid data duplication / network coding approach

As analysed above, there exist two techniques associated with an enhancement of RAN reliability, namely Data Duplication (DD) and Network Coding (NC). DD achieves an increased reliability by duplicating data and sending it via two independent links (exploiting multi-connectivity). NC introduces an additional degree of freedom by generating multiple linear combinations out of a group of packets. Section 2.2 showed how this can be used to send re-transmissions with an increased efficiency. In general, NC can be used to increase reliability by sending an additional amount of linear combinations to increase the decoding probability [TB11].

In the remainder of this section, the aforementioned RAN reliability techniques are studied in a combined manner, leading to a *hybrid approach that applies to multi-connectivity setups*. Figure 2-19 shows how both schemes can be applied in a multi-connectivity scenario. In the case of DD (left side of the figure), incoming packets are duplicated within a RAN reliability Virtual Network Function (VNF) in the Telco Cloud. Duplicates of the packets are forwarded towards the MS via two DUs. A loss of one duplicated packet can be compensated by a successful reception of the same packet through the second link. NC can be used as shown in the right side of the figure. In this case, two (or more) incoming packets are combined. A set of different linear combinations (four in case of the figure) is generated and sent via the two links. Even if two out of the four linear combinations are not successfully received, there is a high probability that both packets can be decoded at the MS [TB11].



**Figure 2-19: Improving RAN reliability by multi-connectivity in combination with Data Duplication (left side of the figure) and Network Coding (right side of the figure)**

### 2.3.1 The hybrid approach

This section describes a hybrid approach which makes use of the advantages of both schemes. In particular, the hybrid scheme is designed such that it can *switch between DD and NC depending on the given requirements on reliability and/or latency*. The hybrid scheme is introduced in the following and then evaluated by means of simulations.

Comparing both approaches, it can be seen that there are advantages and disadvantages for both of them, as listed below:

- NC has the potential to achieve a higher reliability compared to DD. Taking the example of Figure 2-19, NC could compensate the loss of combinations a and b, if combinations c and d are received (or in general any two combinations). In the case of DD, the same event (the loss of both duplicates of packet one) would lead to a packet loss.
- DD has advantages in terms of latency: Packets that arrive at the DD VNF can be processed immediately. In the case of NC, a first packet might have to be queued to combine it with a second or third packet. A corresponding effect occurs at the UE: If a linear combination is delayed, it might cause other linear combinations to be queued until the decoding can take place.

Combining the advantages of both schemes is the motivation for creating a hybrid approach. This hybrid approach is assumed to reside in the Telco Cloud as a VNF and flexibly adjust the RAN reliability strategy by switching between NC and DD.

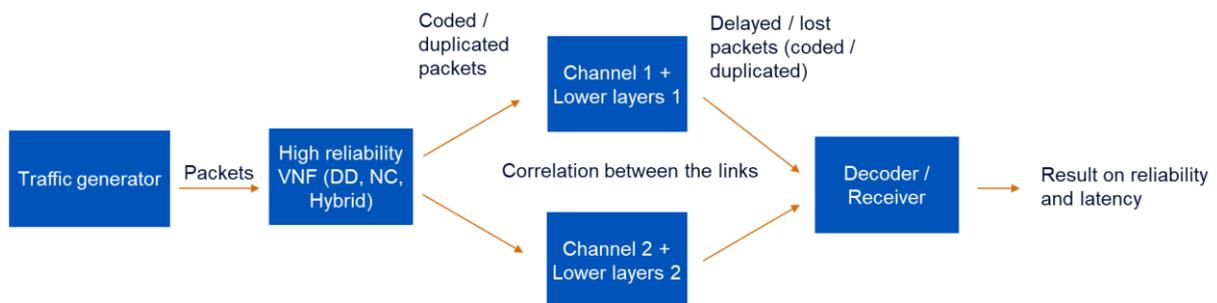
This flexible operation of the hybrid approach follows the following rules:

- 1) If multiple packets arrive at the RAN reliability VNF simultaneously, they are combined using NC to exploit the previously mentioned reliability gain. This happens based on the NC generation size, which is fixed value. It determines the number of packets which form one generation out of which the linear combinations are derived. If, for instance, four packets arrive and the generation size is set to two, packets one and two are combined as well as packets three and four.
- 2) If there are remaining packets which were not combined with other packets (e.g. packet five in the case of five packets arriving and a generation size of two), they are queued for a short time according to a configuration parameter. If no further packets arrive within this duration, the queued packets undergo DD and are sent out.
- 3) If a single packet arrives, it undergoes the same procedure described under point 2.

### 2.3.2 Simulation methodology

A simulation was executed to evaluate the performance of the hybrid approach compared to DD and NC.

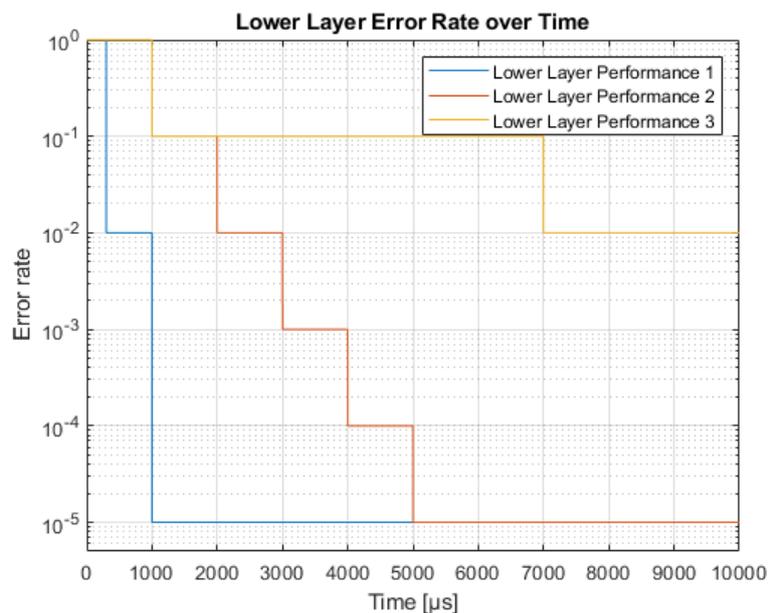
Figure 2-20 shows the simulation setup. A traffic generator creates packets and forwards them to the VNF for RAN reliability. The VNF generates coded or duplicated packets according to the selected scheme (DD, NC, hybrid). The coded / duplicated packets are then sent via two links. For each link assumptions on its performance (reliability versus delay) are made, which are introduced in the following sections. It is also possible to correlate the behaviour of both links, i.e. to increase the probability of simultaneous errors. Moreover, it is assumed that both links terminate at the same decoder which reconstructs the original packets.



**Figure 2-20: Simulation setup for the hybrid approach**

It is emphasised that the performance of the underlying links heavily influences the resulting reliability at the decoder. To study this, *three different air interfaces* are studied below. Their performance is depicted in Figure 2-21.

Specifically, the blue curve represents an Ultra-Reliable Low-Latency Communication (URLLC) air interface [PPM18]. With this air interface, it is able to deliver packets after 0.3ms with an error rate of  $10^{-2}$ . After 1ms, packets can be delivered with an error rate of  $10^{-5}$ .



**Figure 2-21: Simulation of the hybrid approach: Lower layer / air interface performance**

The yellow curve represents the behaviour of LTE with 1ms latency for an error rate of  $10^{-1}$ . It should be noted that the performance of an URLLC air interface can only be achieved under the constraint of a

significantly lower spectral efficiency [NGMN18, SWD+18]. Therefore, a third air interface is additionally assumed (red curve), which is targeted to achieving a compromise between reliability and spectral efficiency. To achieve high spectral efficiency, the third air interface targets a relatively low reliability ( $10^{-1}$  error rate) and uses a 1ms TTI, but at the same time performs retransmission more quickly than LTE. With a corresponding parameterisation, this performance should be achievable with 5G NR.

Besides the lower layer performance, also the traffic model, i.e. the timing of the incoming packets has a significant impact on the performance of NC and the hybrid approach. Two traffic models have been simulated:

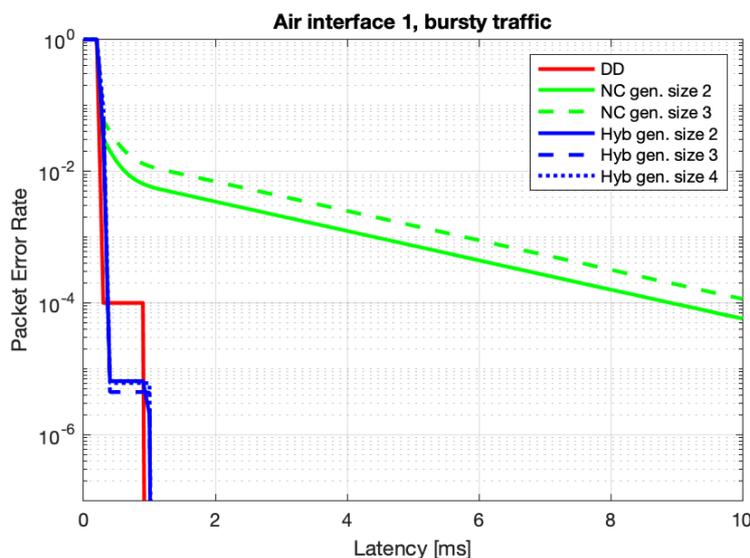
- A traffic model with uniform inter-arrival time of the packets. In this case, a packet is followed by the next one after a fixed time. This traffic model represents industrial fieldbus traffic [WMW05]. To create a best-case scenario for NC, a large number of packets (10000 packets per second) was assumed here, such that the queuing time for combining one packet with another is low.
- In contrast, a bursty traffic model was assumed, which represents e.g., file transfers. In this case, bursts of in average 50 packets (with a standard deviation of 4 packets) were generated. One burst spans over 1ms and in average 500 bursts per second are generated.

### 2.3.3 Simulation results

In the following, simulation results for the three different lower layer performances and the two traffic types are presented. Then, simulation results which study the impact of correlated links are also provided.

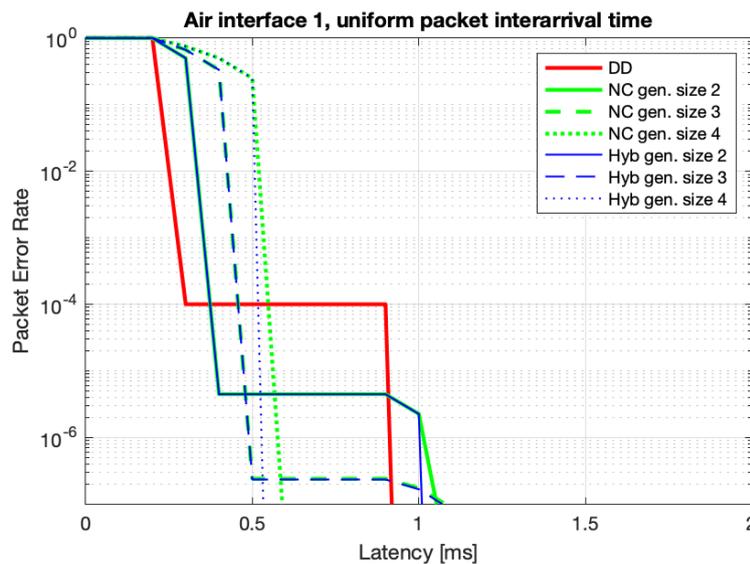
#### 2.3.3.1 URLLC air interface

Figure 2-22 shows the performance in the case of the URLLC air interface and burst traffic. By observing Figure 2-22, a significant drawback of the NC approach can be seen: due to the queuing effect described in Section 2.3.1, some packets are delayed, which influences the overall performance negatively. In this respect, one should note that, for instance, for a packet error rate of  $10^{-4}$  one delayed packet out of 10,000 packets affects the overall performance. DD achieves the expected performance: as a single link achieves an error rate of  $10^{-2}$  after 0.3ms (blue curve in Figure 2-21) it achieves an error rate of  $10^{-4}$  after 0.3ms by exploiting two uncorrelated multi-connectivity links in Figure 2-22. The hybrid approach thus achieves a significant reliability increase compared to DD, by combining most of the packets via NC while simultaneously avoiding the queuing problems of NC. For the hybrid approach as well as NC, different generation sizes, i.e., the number of packets that were combined in one group, were simulated. This had low impact under the simulated conditions.



**Figure 2-22: Simulation results for bursty traffic and URLLC air interface** Figure 2-23 shows the performance for the case of the URLLC air interface and uniform traffic. The scaling of the x-axis is changed compared to that in Figure 2-22 to allow for more insights on the performance at low latency.

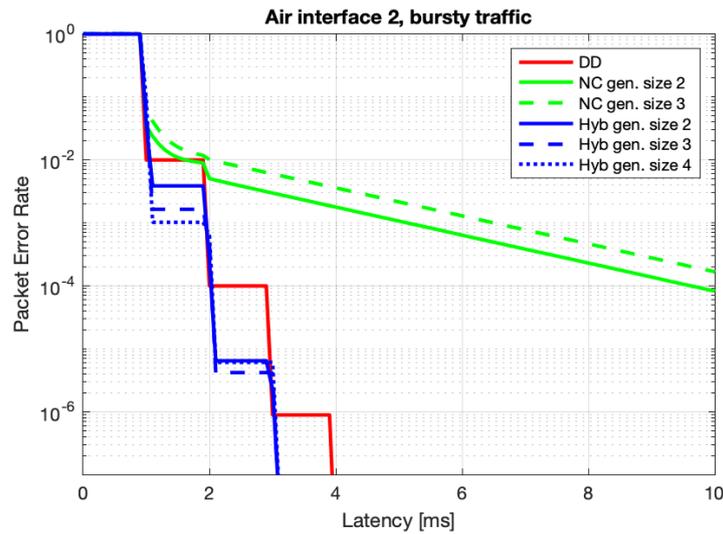
NC and the hybrid approach in this case achieve the same performance, such that the green and the blue curve coincide (NC generation size 2 achieves the same performance as the hybrid approach with generation size 2, NC generation size 3 achieves the same performance as the hybrid approach with generation size 3 and so on). Uniform traffic with a high packet rate is the best case for NC, as a low queuing delay is required until a second or third packet arrives. Therefore, NC achieves a significant increase in reliability, with a low penalty in terms of latency. Higher generation sizes lead to higher reliability at the cost of latency. The hybrid approach in this case was configured via the configuration parameter (described in Section 2.3.1) such that it queues the packets until one NC generation can be created. It therefore achieves the same performance as NC.



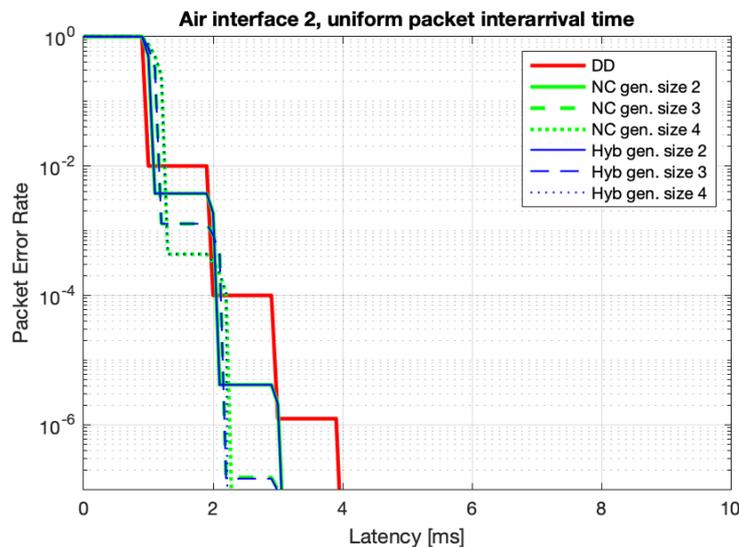
**Figure 2-23: Simulation results for uniform traffic and URLLC air interface (the green and blue curves coincide)** **Medium air interface**

Figure 2-24 and Figure 2-25 show the simulation results for the air interface with reduced reliability (red curve in Figure 2-21). In principle, the same trends as observed for the URLLC air interface can be seen here. Specifically:

- NC has a significant drawback in the case of bursty traffic.
- The hybrid approach achieves the best performance in the bursty traffic case; in the case of uniform traffic it achieves the performance of NC (the green curves are again coinciding with blue ones).



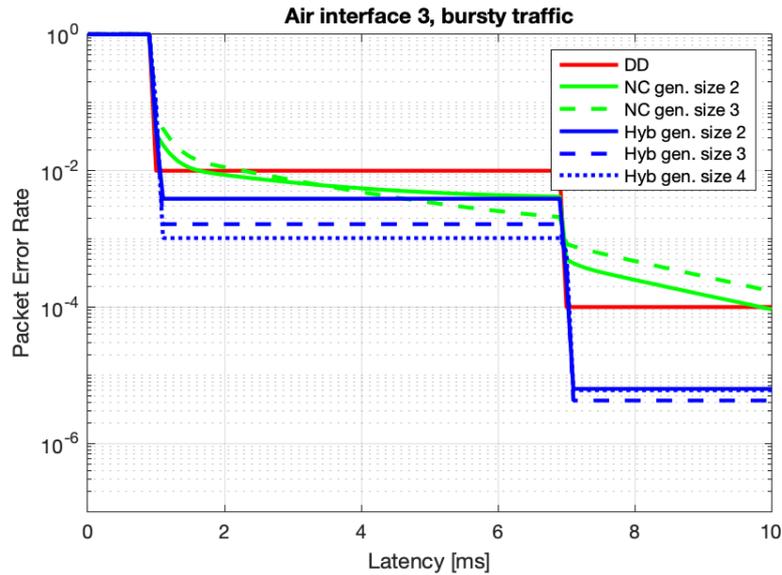
*Figure 2-24: Simulation results for bursty traffic and medium air interface*



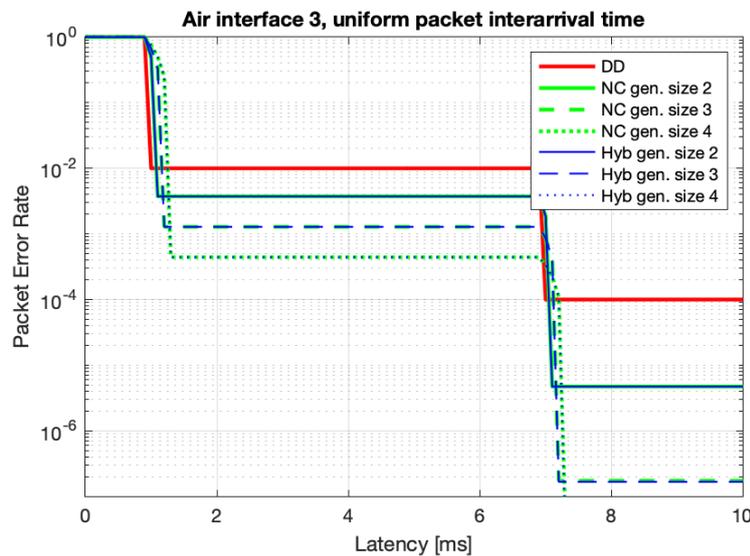
*Figure 2-25: Simulation results for uniform traffic and medium air interface (the green and blue curves coincide)*

Figure 2-26 and Figure 2-27 show the simulation results for the air interface with low reliability (yellow curve in Figure 2-21). The main observation from Figure 2-26 and Figure 2-27 relates to the following trend:

- Due to the lower overall reliability, NC can compensate the drawbacks in case of bursty traffic and achieve a performance similar to DD.
- For the uniform traffic the hybrid approach again achieves the same performance as NC. Due to combining packets and multi-connectivity, this leads to the fact that even with the relatively unreliable air interface, a packet error rate of  $10^{-5}$  or less can be achieved if a higher latency is tolerated.



**Figure 2-26: Simulation results for bursty traffic and air interface with low reliability**



**Figure 2-27: Simulation results for uniform traffic and medium air interface (the green and blue curves coincide)**

The results shown so far assumed two uncorrelated links towards the MS, which is a best-case assumption. In the following the impact of correlated links is studied for the example of the air interface of medium reliability. For this purpose, a correlation factor is introduced:

- A correlation factor of 0 means that both links cause independent packet losses.
- A correlation factor of 0.5 means that half of the errors occur simultaneously at both links, whereas the other half occurs uncorrelated.
- A correlation factor of 1 means that only simultaneous errors on both links occur.

Figure 2-28 and Figure 2-29 show the corresponding results pertaining to correlated links. It can be seen that, as expected, correlated links have a strong negative impact on reliability. More specifically, in the extreme case of fully correlated links, DD is not a suitable means for increased reliability and achieves the same performance as a single link. NC and the hybrid approach can compensate also for simultaneous errors (as introduced in Section 2.3.1) but on the expense of the reliability they achieve.

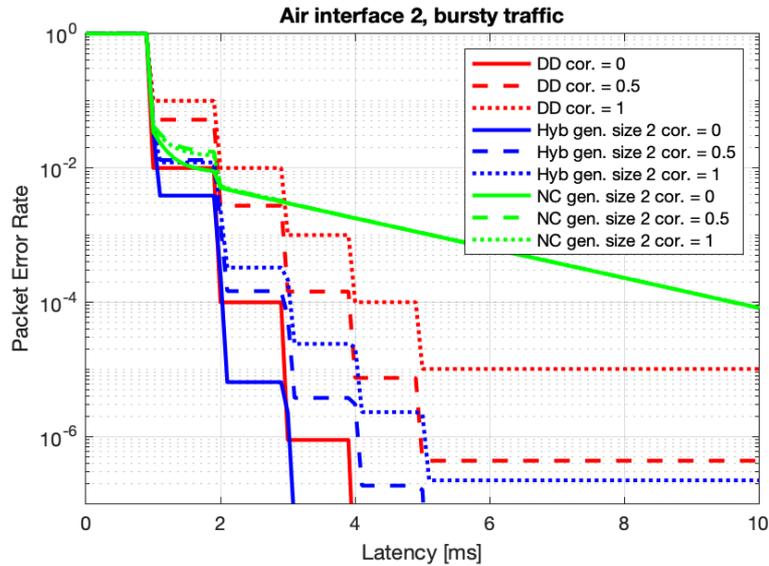


Figure 2-28: Simulation results for correlated links and bursty traffic

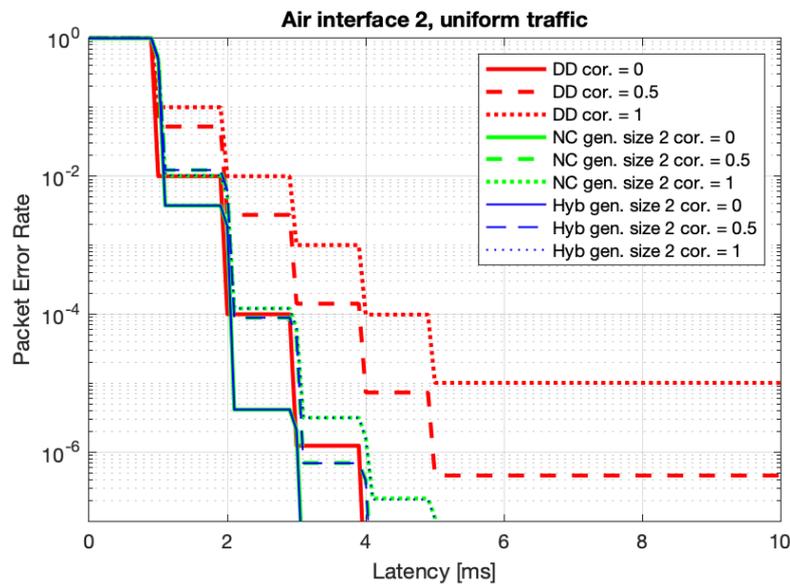


Figure 2-29: Simulation results for correlated links and uniform traffic (the green and blue curves coincide) **Concluding remarks on the hybrid approach**

The simulation results show that the proposed hybrid approach can combine the advantages of DD and NC. It achieves the highest reliability in the case of bursty traffic and equal performance compared to NC in the case of uniform traffic. It is also shown that by combining packets, such as in the case of NC and the hybrid approach, the negative impact of correlation in the case of multi-connectivity can be significantly reduced compared to DD.

### 3 Telco cloud resilience

The telco cloud resilience represents one of the two fundamental pillars of WP3 of 5G-MoNArch as described in Section 1.1. It comprises the approaches for increasing the robustness of the telco cloud through redundancy, augmented controlled scalability and autonomous VNF migration. The functions that enable resilience of telco cloud which are described within Chapter 3 are integrated into the overall 5G-MoNArch architecture, as depicted in Figure 1-3. Specifically, the telco cloud resilience enablers developed in WP3 lie within the *Management and Orchestration layer* and *Controller layer* of the 5G-MoNArch architecture and are marked with the respective outer frames in Figure 1-3.

Section 3.1 elaborates on the fault management, enhanced root cause analysis and resource redundancy techniques towards telco cloud resilience. This section also presents the approach for selection of suitable redundancy scheme in the telco cloud by considering the availability requirements of the slices, type of deployed virtual network functions (NF) and also the inter dependencies between those functions. The fault management approaches described in Section 3.1 may be applicable to different network functions. However, a specific network functions such as network controllers may require additional mechanisms in order to achieve required level of resilience. With respect to network controllers the resilience requirements include the ability to adapt to load increase, i.e. ability to seamlessly scale.

Section 3.2 presents in detail the problem of controller scalability and the current state of the art solutions. Furthermore, the Section 3.2 describes the drawbacks of the current solutions and the details of the proposed scalable controller framework developed within 5G-MoNArch in order to improve the control plane resiliency in the telco cloud. Furthermore, the outages in backhaul connectivity require specific approaches and algorithms for achieving the required level of resilience and the common fault management framework may not be sufficient to fulfil such requirements. For example, in the case of backhaul connection outage the VNF migration from the central cloud to the local edge cloud, need to be performed. Such approaches are discussed in the framework of autonomous VNF migration, referred to also as “5G Islands” in [5GM-D3.1], and are detailed in Section 3.3.

#### 3.1 Root cause identification of faults and applying redundancy for higher availability at telco cloud

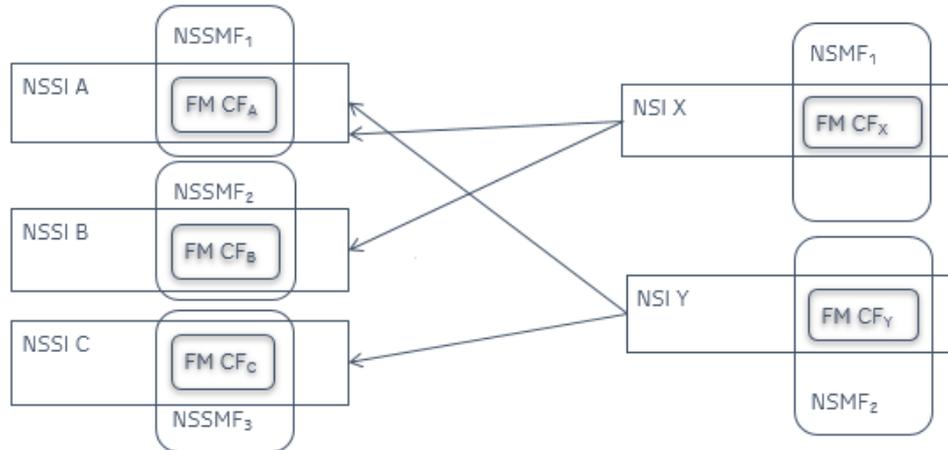
In the following section we provide a description on fault management (FM) enhancements needed to handle the network faults in slicing enabled networks. Hereby we especially focus on the correlation of events coming from different network management entities that may reside on different domains and network functions. Furthermore, we highlight the importance of redundancy in achieving higher network availability, providing the analytical results on network availability when selected redundancy schemes are used.

##### 3.1.1 Advanced fault management event correlation in slicing enabled network

The concept of cognitive network management and Fault Management (FM) cognitive functions described in previous deliverable [D3.1] proposes approach for improving flexibility and adaptability of SON functions based on requirements of network slices as well as based on the network context. The Fault Management Cognitive Functions (FM CFs) focus on troubleshooting of a network slice instance, network slice subnet instance or its individual network functions and deployment. However, the FM CFs cannot act independently as the network entities and resources they are responsible for can have many interdependencies. Furthermore, as the parts of the network slices can even belong to different administrative and/or management domains, the visibility of information across FM CFs can be additionally limited.

Figure 3-1 illustrates the case where limited visibility of slice subnet components by different management entities, i.e. FM CFs at NSI and NSSI level, can result in inability of proper root cause analysis. The Figure 3-1 shows two network slices, NSI X and NSI Z. The arrows in the Figure 3-1 illustrate which slice subnets are building blocks of NSI X and NSI Y, i.e. NSI X is composed out of NSSI A and NSSI B, whereas NSI Y is composed out of NSSI A and NSSI C. Given the interdependencies in Figure 3-1, the problem at NSSI C followed by reconfiguration performed by FM

$CF_C$  may have negative impact/problem at NSI Y. The FM  $CF_X$  knowing that the original problem is in NSSI C, would try to mediate the problem by issuing the reconfiguration request at NSSI A level, performed by FM  $CF_A$ . Such reconfiguration at NSSI A level might cause the problem at NSI X level and consequently at NSSI B level. However, as the FM  $CF_B$  does not have direct visibility of events at NSSI C level, it is not able to perform adequate root cause analysis directly, i.e. by processing received information on visible events. Consequently, this might lead into difficulties in the problem recovery.



**Figure 3-1: Interdependencies between FM CFs at NSI and NSSI levels**

Generally, the capability of high resource multiplexing in virtualised environment implies more complex dependencies between NSSIs and NSIs, and corresponding management entities such as FM CFs. E.g., the NSSIs can be shared among multiple NSIs, thus, it is very likely that the problem or re-configuration of one NSSI will affect possibly multiple NSIs and associated NSSIs, leading to a “chain reaction” in propagation of effects among NSSIs. Without having the full visibility on all network entities and their interdependencies, as the simplest solution to the problem, some of the FM CFs may attempt to “undo” the NSSI reconfigurations. This may lead to “ping-pong” effect, i.e. bouncing back the actual problem, where the remedy of one problem would be considered as its root cause and will be reverted. Apart from being inefficient, such approach may result in instabilities of the system. In the example shown in Figure 3-1. The FM  $CF_X$  being unable to detect the actual root cause of the problem and not being aware of re-configuration events that have followed, might attempt to undo the re-configurations done in NSSI A, and thus create “ping-pong” effect. On the other hand, by re-configuring the NSSI B it might prolong the chain reaction and propagate further the effects of the original problem. Both effects should be minimised as they might have negative influence on the network performance. The event correlation among FM CFs provide the means for minimising such effects by discovering the overall picture on the network events and their interdependencies.

In more complex sliced network deployments, the NSIs, NSSIs and corresponding management functions may be operated even by different organisations. In such cases there might not exist a single network management function with an overview on the complete network. Therefore, without event correlation the dependencies between the FM CFs necessary for self-healing use cases will be hard to manage. Furthermore, the network functions can produce a large amount of information that can be processed by FM CFs on different levels, but not all such information is relevant to all FM CFs and ultimately, not all of such information is important for all FM CFs. Thus, the processing of all such information is not only costly but also unnecessary.

Therefore, there is a need of implementing the event correlation system that utilises the information related to a large number of events and extracts the information on few events that have the highest relevance and importance. This is usually done by finding the relationships between events and analysing such relationships. In mobile networks, the Network Management (NM) entity has the capability of correlating alarm events raised by different network functions. Apart from alarms also other types of events can be correlated e.g., anomalies detected by an anomaly detection function. Additionally, the data from multiple sources can be combined in order to derive more complex events or patterns. This can be used to for example to improve the quality of network management processes.

### 3.1.1.1 Event correlation function, event notification message and its distribution area

The approach followed by the 5G-MoNArch aims at correlating the information on events that is detected by the FM CFs. Such correlation is performed by distributed Event Correlation Function (ECF) which can detect related events reported by other FM CFs in different NSMFs or NSSMFs including causality and temporal context. The FM CFs communicate the Event Notification (EN) messages including the available information about the events, e.g.

- ID of the reporting management function (e.g. NSSMF or NSMF ID)
- An Event Identifier (EID) within the reporting management function
- ID of the reporting FM CF (FMCFFID)
- Timestamp of the event
- Event Type (ET) (for example, but not limited to: alarm type, anomaly root cause label)
- Event Lifecycle info, indicating e.g. if the event is new or it is an end of the event
- An indicator, if the reporting FM CF is acting on the event (for coordination purposes)

The exchange of such event notification messages can be performed in different ways, e.g., using the publish/subscribe paradigm, or the message can be sent between FM CFs within a dedicated distribution area. The distribution area containing the set of FM CFs that need to exchange the event notification messages, can be determined based on the location of the FM CF that is sending the notification message, e.g.:

- If the EN is sent by a FM CF in NSSI the relevant FM CFs, i.e. distribution area may comprise
  - all remaining FM CFs of that NSSI
  - all FM CFs in directly associated NSIs to that NSSI
  - all FM CFs in NSSIs that are building blocks of the directly associated NSI– indirect association between NSSIs
- If the EN is distributed by a FM in NSI the distribution area may comprise of:
  - all remaining FM CFs of that NSI
  - all FM CFs in NSSIs that are building blocks of that NSI
  - all FM CFs in NSIs that are sharing (directly associated to) building blocks NSSIs– indirect association between NSIs

One example of distribution area of NSSI C is marked in Figure 3-2:. This implies that the events related to the NSSI C need to be communicated to the FM CF<sub>Y</sub> as the NSI Y is composed out of NSSI C. Furthermore, as the NSSI A is another building block of the NSI Y, the events need to be communicated to its FM CF, i.e. FM CF<sub>A</sub>. The distribution area can be further dynamically adjusted. Larger distribution area would enable better information availability, thus more valuable information extraction. However, this implies more complexity in event processing, and potentially the delays in obtaining the correlation results.

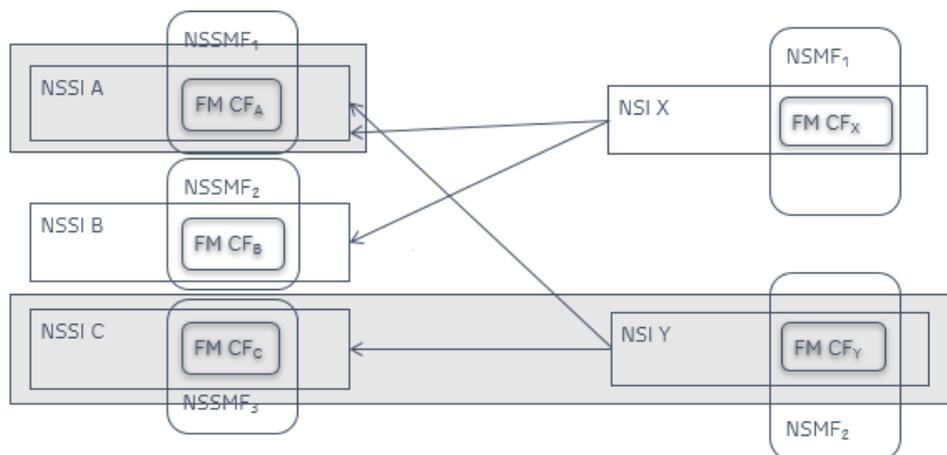


Figure 3-2: Distribution area of NSSI C

### 3.1.1.2 Event correlation function – deployment and benefits

The Event Correlation Function (ECF) can be deployed either at subnet or slice level. ECF will gather the information on relevant events, e.g. those issued within distribution area, and correlate them in order to derive more complex event which can be useful for overall diagnosis. E.g., ECF can be deployed as a component of NSSMF<sub>3</sub> in Figure 3-2:, or as an integral part of FM CF<sub>C</sub> and it can receive the events from marked distribution area. Based on such information, it can derive that the problems at NSSI A are caused by the initial reconfiguration of NSSI C. This complex event can be further signalled to related FM CFs, e.g. FM CF<sub>X</sub>. The FM CFs (such as FM CF<sub>X</sub> in this example) can subscribe to such complex events or can be configured to receive complex events from FM CFs with which they have only transitive dependencies, in order to receive an earlier warning for potential problems.

The main benefits of such an approach where advanced correlation of FM events if used can be summarised as follows:

- The FM CFs may utilise a distributed ECF, which can detect related events reported by other FM CFs in different NSMFs or NSSMFs including causality and temporal context
- The FM CFs may coordinate their corrective actions such that they minimise the impact on those functions that have the most dependencies as indicated by the event correlation.
  - As an example, corrective actions on a NSSI dedicated to a single NSI are preferred over changes in a shared NSSI.
- Using the results from ECF the aforementioned “chain reaction” and “ping-pong” effects in event propagation can be minimised

### 3.1.2 Applying redundancy for higher resilience

High availability of the 5G network is tightly coupled with the high availability of the telco cloud as its integral component. In order to achieve high availability of the telco cloud the different mechanisms for improving its resilience can be applicable. One of approaches for improving the resilience is applying the redundancy in telco cloud deployment. This approach is also a prerequisite for efficient operation of other mechanisms for improving telco cloud resilience discussed in 5G-MoNArch, such as 5G Islands, or enhanced fault management. Increased redundancy allows shorter failure recovery time, and thus improves overall network availability. However, the increased redundancy comes with increased costs and operational complexity. Such a trade-off in applying redundancy needs to be carefully considered in system design.

There are different redundancy schemes that can be applied in the telco cloud, leading to different levels of telco cloud availability and consequently cost and complexity. In general, a number of components (N) is backed up with a certain number of additional components (M), forming the N+M redundancy approach. There are different modes in which N components are interacting with M redundant components. As an illustration, in following we briefly describe some of the representative redundancy modes [H+16], [AVA18], however, further redundancy schemes are applicable to the telco cloud e.g. as discussed in [AMF16], [AD13], [AVA19].

- N+M redundancy scheme (N active-M standby) is designed in a way that one telco cloud instance, e.g. VM, container is processing the load, i.e. being active instance, whereas additional instance is prepared to take over the processing from active instance, once it fails. The procedure of taking over the processing load may incur the considerable delay. Such delay depends mainly on the level of readiness of the standby instance (e.g. being in a cold or hot standby) to take over the processing load. E.g. in the case of stateful network function failure, the processing states need to be synchronised beforehand among the active and standby instance. Better preparation of redundant instance, e.g. by synchronisation between active and standby instances, decreases the failover time but increases the resource utilisation and consequently resource costs and wastage.
- Load sharing scheme allows sharing of processing load among instances. This scheme follows the N+M redundancy approach, where only N instances would be needed to handle the peak processing load, but additional M instances are used in parallel, thus the processing load is distributed among N+M instances. This redundancy scheme provides a good trade-off between

the telco cloud availability that can be achieved and the amount of redundant resources/cost. However, it is mainly suitable for processing the tasks without major interdependencies, where parallelisation of processing can be achieved.

- Full redundancy (2N) where to N active instances an equal amount of redundant instances (N) is associated. Such redundant instances may have different level of readiness (e.g. being in a cold or hot standby) or can even perform the processing in parallel to active instances. This approach provides the highest availability at the cost of largest resource wastage. Therefore, it is suitable mainly to cases with extreme availability requirements.

The resulting availability of the telco cloud depends on the availability of the single instance, type of redundancy scheme, as well as the amount of redundancy applied. Furthermore, the time to detect the actual fault as well as to recover from it e.g. by using redundancy influence the resulting availability of the telco cloud.

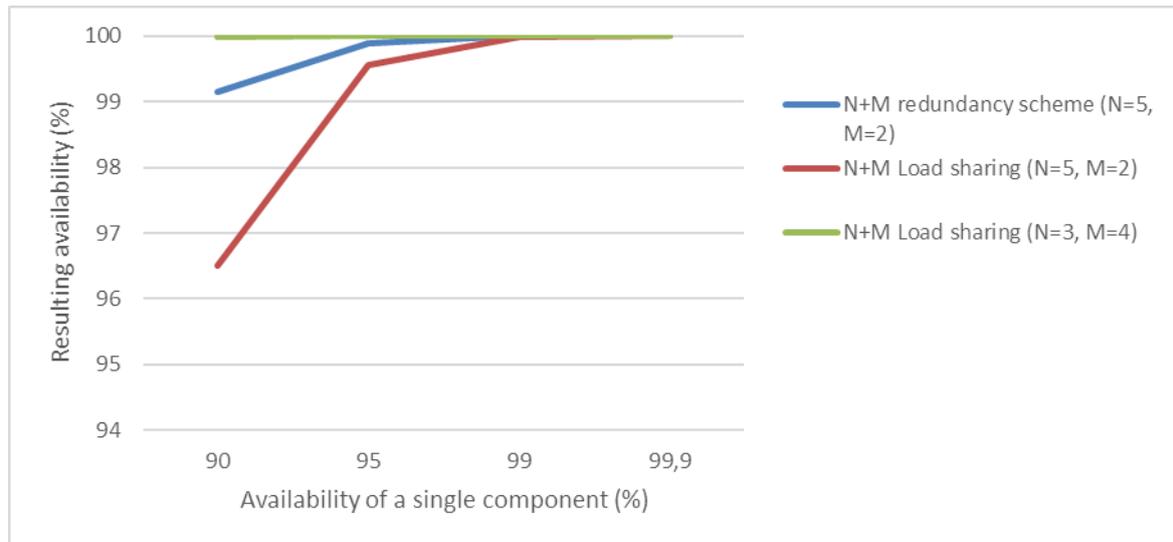
### 3.1.2.1 Selection of suitable redundancy scheme

As indicated above, the different redundancy schemes might be suitable for different use cases. In order to select the most appropriate scheme for particular context in which the telco cloud is applied the 5G-MoNArch elaborates on the most important inputs to be taken into account, which are:

- Information regarding the required availability level of the telco cloud, given the required E2E availability requirements of the service/slice, e.g. 4-nines, 5-nines availability (that is, 99,99% and 99,999%, respectively).
- Type of network functions deployed on the telco cloud with respect to processing state, i.e. stateful or stateless network function (NF).
- The recovery of stateful network functions requires the information about the operational state in the moment of failure, thus adequate preparation of redundant instance by a priori state synchronisation between active and redundant instances is required. This is not the case for stateless network functions.
- Consideration on interdependencies among network function and their processing tasks.
- This input determines how processing tasks can be handled, i.e. in parallel or serial way. Thus, different redundancy schemes may be suitable.

Due to its cost efficiency the load sharing redundancy scheme can be seen as the most reasonable approach for the cases where the processing tasks can be executed to a large extent in parallel. Furthermore, the resulting availability that can be achieved by load sharing scheme depends on the current load in the network. E.g. if N instances can handle the peak load and M are used in parallel (resulting in N+M load sharing scheme) in the case of lower load where only N-P instances are needed to handle the lower load, the resulting redundancy scheme would be (N-P, M+P) which significantly increases the overall availability of the network.

Figure 3-3 illustrates this effect for N=5, M=2, P=2, for different assumptions on availability of a single component. Additionally, Figure 3-3 shows the comparison between the load sharing approach and generic N+M redundancy scheme without load sharing. Such generic redundancy scheme provides better results in terms of overall availability, at the cost of more resource usage and no flexibility with respect to the traffic load. Note that results in Figure 3-3 take into account only the assumption on the availability of a single instance and the redundancy scheme and amount applied, without consideration on further impacting factors such as fault detection time, time needed to recover from the fault using the redundant instances, etc.



**Figure 3-3: Overall availability of the network given different redundancy schemes and assumptions on availability of a single component.**

### 3.2 Augmented resilience via increased controller scalability

Controller is a key element in the Telco cloud for implementing various functionalities such as programmability of VNFs, VNF chaining and networking between VNFs deployed across physically distributed Infrastructure. Telco cloud controller need to be designed to support augmented resiliency as well as scalability to have reduced control plane latency, fast recovery during failure, and improved performance. When deciding on which software defined network (SDN) controller(s) to use in production, there are plenty of features to be examined such as programming language, the performance, the time to learn to develop applications, the protocols of southbound API, performance of centralised and distributed approaches, etc. For instance, a single SDN controller would represent a single point of failure for the entire network and the solution in this case is to use a cluster of controllers running in parallel instead. In the typical SDN use case, the default action of an SDN capable device is to forward new packets to the controller if they do not correspond to any of the entries found in the SDN devices' flow tables. The SDN controller decides what to do with the packets.

In a network with a large number of nodes, forwarding traffic to a single controller would lead to a bottleneck in performance. Multiple controllers will be the answer to assure high availability and scalability, so if one controller fails, the others would be available to take over the role. Such method raises the issue of distributed state management, where synchronisation is indispensable to have uniformity in the network. However, the available open-source and commercial controllers have various issues related to the scalability of controller functions; i.e., the number of controller nodes in a cluster cannot scale automatically in response to the underlying network traffic. This work mainly targets the performance improvement of the controller framework to be auto-scalable and better at supporting load balancing.

#### 3.2.1 Scalability solution analysis

In this section, we study and experiment with various SDN controllers with special focus on their scalability solution.

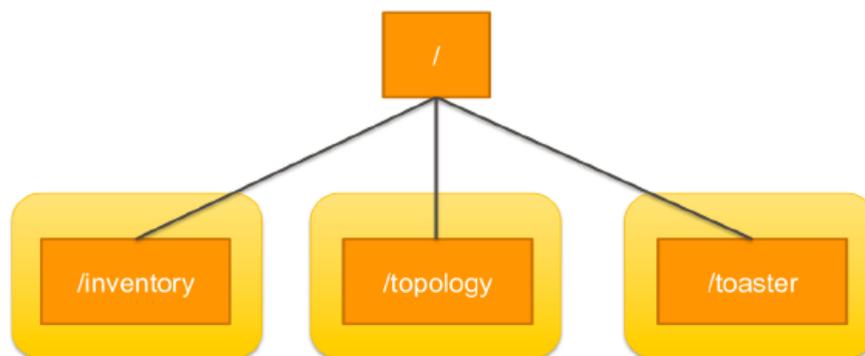
##### 3.2.1.1 OpenDayLight

OpenDayLight [ODL18] from Linux foundation is one of the mainstream open source SDN controllers. The main features that ODL offers include: clustering, to ensure high availability by implementing the Raft algorithm, and employing AKKA framework [Raft18] to manage multiple controllers and their states. Diverse experiments have been conducted to measure the performance and the effectiveness of clustering controllers in ODL. Model driven-service abstraction layer (MD-SAL) clustering allows multiple controllers in ODL to form a cluster, where each controller executes an identical set of network

services. MD-SAL clustering in ODL provides a Network State Database (NSDB) called the distributed data store. It enables network services to store their network states into several partitions and select which YANG module to be contained in the partition. This data is referred to as a shard. Shards can be replicated and placed into multiple ODL controllers.

### **Shard:**

There exists one special partition called the default shard, which contains all data except the data defined by the selected YANG modules set by the administrator. For example, ODL models the topology state, that includes a set of devices and links, as a network-topology YANG module. If the administrator selects the network-topology YANG module to be contained in a partition, the total number of partitions becomes two (i.e., default shard and topology shard), since all other network states except the topology state are contained in the default shard. Each partition is replicated into R replicas, where R is configurable by the administrator. Each replica is assigned to the controller which has the least number of replicas. When the number of replicas is smaller than the number of controllers in a cluster, the amount of network states that can be handled by the distributed data store increases with the number of controllers in the cluster. Meanwhile, when each partition has multiple replicas, and as synchronisation is performed on a per partition basis, the synchronisation overhead in the distributed data store increases with the number of partitions.



**Figure 3-4: Module-based shard [ODLSHARD]**

### **Raft in OpenDayLight:**

ODL uses the Raft protocol for synchronisation between replicas of a partition, which provides strong consistency at the cost of inferior read/write performance. The Raft protocol in ODL elects one leader replica for a partition and thus all states within distributed data store of ODL guarantee strong consistency. Meanwhile, read/write requests from ODL controllers which do not contain a leader replica are handled remotely, which increases latency for their requests. Also, in order to commit write requests to the leader replica, the agreement among most replicas is mandatory, and therefore additional latency occurs. When a controller which contains a leader replica for a partition fails, data access to the partition is prohibited during the absence of the leader for strong consistency.

Figure 3-5 shows an example of topology state synchronisation in ODL. In ODL, topology state is stored into distributed data store, which uses Raft protocol for synchronisation. As shown in Figure 3-5, each ODL controller manages a subset of the topology and read/write requests to the topology state can only be handled by the leader replica in ODL 1 controller, which increases latency. For instance, when ODL 2 controller receives read requests to Topology B, it fetches the corresponding topology state stored in the leader replica in ODL 1 controller and replies with the fetched state. Also, when ODL 3 controller receives topology update events from Topology C, the requests are forwarded to the leader replica in ODL 1 controller. After that, the leader replica asks for the agreement on the updates among most of the follower replicas and, if successful, it commits the updates. In this manner, the consistency between topology state replicas is guaranteed all the time in ODL.

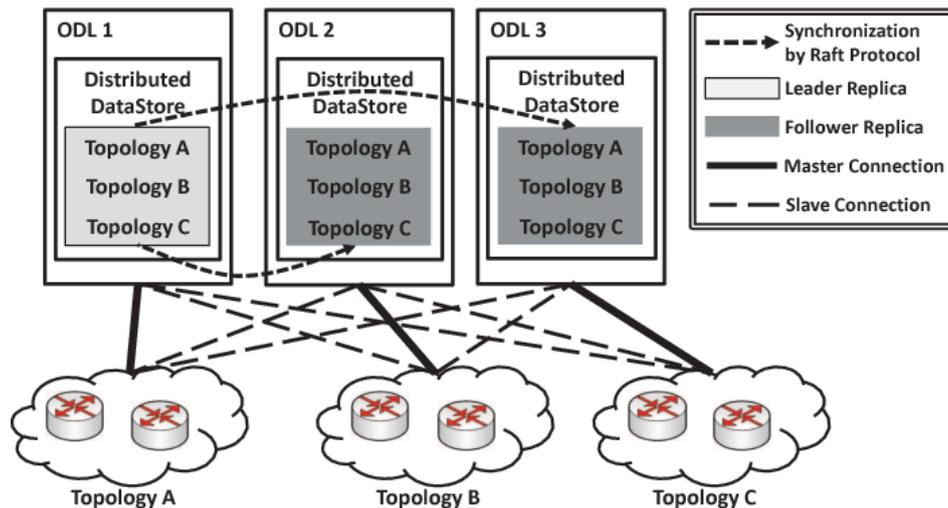


Figure 3-5: ODL topology synchronisation

### Scalability Analysis:

In order to examine the functionality of ODL, we form a cluster of 3 controllers. For our experiment, we used the Beryllium version of ODL running in VM. We started with 3 VM's with 2GB RAM and 10GB ROM and deployed ODL inside each VM. Since this experiment uses VirtualBox, it will be easier to create a secure shell (SSH) connection to each VM and access the VM through the command line interface (CLI) from the host. The CLI of ODL is shown in Figure 3-6.

```

root@dingzg-ubuntu: /home/dingzg
Hit '<tab>' for a list of available commands
and '[cmd] --help' for help on a specific command.
Hit '<ctrl-d>' or type 'system:shutdown' or 'logout' to shutdown OpenDaylight.

opendaylight-user@root>feature:list -i
Name          | Version | Installed | Repository      | Description
-----
standard     | 3.0.7  | x         | standard-3.0.7 | Karaf standard feature
config       | 3.0.7  | x         | standard-3.0.7 | Provide OSGi ConfigAdmin support
region      | 3.0.7  | x         | standard-3.0.7 | Provide Region Support
package      | 3.0.7  | x         | standard-3.0.7 | Package commands and mbeans
kar support  | 3.0.7  | x         | standard-3.0.7 | Provide KAR (KARaf archive)
ssh         | 3.0.7  | x         | standard-3.0.7 | Provide a SSHd server on Karaf
management  | 3.0.7  | x         | standard-3.0.7 | Provide a JMX MBeanServer and a set of MBeans in Karaf
opendaylight-user@root>

```

Figure 3-6: ODL install features

Mininet is used to create a network topology. Feature installation and file modifications should be done for every controller. Cluster configuration defines the members (nodes) of the cluster and the replicas of the shard. The configuration can be defined in a number of configuration files, which can be placed in the ODL distribution. When the ODL controller is started, the configuration file can be passed to it. When the MD-SAL clustering service bundle comes up, it can look at which specific configuration needs to be loaded, reads it from disk, and initialises itself. In order to enable the clustering feature, there are 2 files to modify in /configuration/initial folder: akka.conf and module-shards.conf. When the odl-mdsal-clustering is installed, it creates those two files. When each shard is defined in every member, this means that it replicates and stores each shard, such as: inventory, topology, toaster, and default, in each controller. An example of akka.conf and module-shards.conf is provided in Figure 3-7 and Figure 3-8, respectively.

```

odl-cluster-data {
  akka {
    remote {
      netty.tcp {
        hostname = "192.168.56.101"
        port = 2550
      }
    }
    cluster {
      seed-nodes = ["akka.tcp://opendaylight-cluster-
data@192.168.56.101:2550","akka.tcp://opendaylight-
cluster-data@192.168.56.102:2550","akka.tcp://
opendaylight-cluster-data@192.168.56.103:2550"]
      roles = ["member-1"]
    }
    persistence {
      # By default the snapshots / journal directories live
      # in KARAF_HOME. You can choose to put it somewhere
      # else by modifying the following two properties.
      # The directory location specified may be a relative
      # or absolute path. The relative path is always
      # relative to KARAF_HOME.
      # snapshot-store.local.dir = "target/snapshots"
      # journal.leveldb.dir = "target/journal"
      journal {
        leveldb {
          # Set native = off to use a Java-only
          # implementation of leveldb
          # Note that the Java-only version is not
          # currently considered by Akka to be
          # production quality

          # native = off
        }
      }
    }
  }
}

```

**Figure 3-7: Example of akka.conf**

```

module-shards = [
  {
    name = "default"
    shards = [
      {
        name = "default"
        replicas = [
          "member-1"
          "member-2"
          "member-3"
        ]
      }
    ]
  },
  {
    name = "topology"
    shards = [
      {
        name = "topology"
        replicas = [
          "member-1"
          "member-2"
          "member-3"
        ]
      }
    ]
  },
  {
    name = "inventory"
    shards = [
      {
        name = "inventory"
        replicas = [
          "member-1"
          "member-2"
          "member-3"
        ]
      }
    ]
  },
  {
    name = "toaster"
    shards = [
      {
        name = "toaster"
        replicas = [
          "member-1"
          "member-2"
          "member-3"
        ]
      }
    ]
  }
]

```

**Figure 3-8: Example of module-shards.conf**

The module-shards.conf file describes which shards live on which members (nodes of a cluster) and the cluster members on which replicas of those shards exist. The replica primarily depends on the order of the replica list. The clustering service is responsible for the discovery of all nodes (controllers) which form the cluster and all related functions. It depends on AKKA clustering to identify the members of a cluster. When the clustering service comes up, it first checks for the state of the cluster. It looks up all the members in the cluster and verifies that all the roles defined in the cluster-configuration are fulfilled by the cluster membership. Once all the members with the required roles are up and running, the clustering service notifies its listeners that the controller is open for connection. We installed jolokia to access the information of the shard by JSON format. Using curl command, we can request some information on a specific shard by defining the IP and member ID as shown in Figure 3-9. The output is shown in Figure 3-10.

```
curl -s
http://192.168.56.102:8181/jolokia/read/org.opendaylight.controller:Category=Shards,name=member-2-shard-topology-config,type=DistributedConfigDatastore | python -m json.tool
```

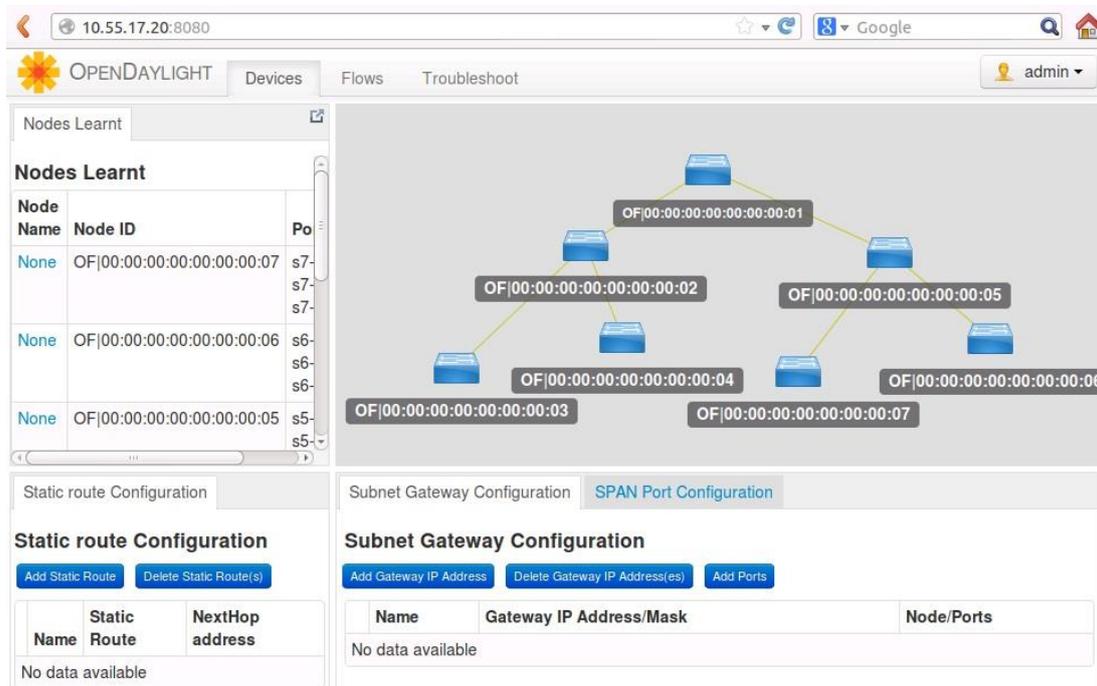
**Figure 3-9: OpenDayLight curl data**

```
{
  "request": {
    "mbean": "org.opendaylight.controller: Category=Shards,name=member-2-shard-topology-
config,type=DistributedConfigDatastore",
    "type": "read"
  },
  "status": 200,
  "timestamp": 1522843142,
  "value": {
    "AbortTransactionCount": 0,
    "CommitIndex": -1,
    "CommittedTransactionsCount": 0,
    "CurrentTerm": 443,
    "FailedReadTransactionsCount": 0,
    "FailedTransactionsCount": 0,
    "FollowerInfo": [
      {
        "active": true,
        "id": "member-1-shard-topology-config",
        "matchIndex": -1,
        "nextIndex": 0,
        "timeSinceLastActivity": "00:00:00:335",
        "voting": true
      },
      {
        "active": true,
        "id": "member-3-shard-topology-config",
        "matchIndex": -1,
        "nextIndex": 0,
        "timeSinceLastActivity": "00:00:00:349",
        "voting": true
      }
    ],
    "FollowerInitialSyncStatus": true,
    "InMemoryJournalDataSize": 0,
    "InMemoryJournalLogSize": 0,
    "LastApplied": -1,
    "LastCommittedTransactionTime": "1970-01-01 01:00:00:000",
    "LastIndex": -1,
    "LastLeadershipChangeTime": "2018-04-04 13:53_40:127",
    "LastLogIndex": -1,
    "LastLogTerm": -1,
    "LastTerm": -1,
    "Leader": "member-2-shard-topology-config",
    "LeadershipChangeCount": 3,
    "PeerAddress": "member-1-shard-topology-config: akka.tcp://opendaylight-cluster-
data@192.168.56.101:2550/user/shardmanager-config/member-1-shard-topology-config,
member-3-shard-topology-config: akka.tcp://opendaylight-cluster-data@192.168.56.101:
2550/user/shardmanager-config/member-3-shard-topology-config",
    "PeerVotingStates": "member-1-shard-topology-config: true, member-3-shard-topology-
config: true",
    "PendingTxCommitQueueSize": 0,
    "RaftState": "Leader",
    "ReadOnlyTransactionCount": 0,
    "ReadWriteTransactionCount": 0,
    "ReplicatedToAllIndex": -1,
    "ShardName": "member-2-shard-topology-config",
    "SnapshotCaptureInitiated": false,
    "SnapshotIndex": -1,
    "SnapshotTerm": -1,
    "StatRetrievalError": null,
    "StatRetrievalTime": "14.39 ms",
    "TxCohortCacheSize": 0,
    "VotedFor": "member-2-shard-topology-config",
    "Voting": true,
    "WriteOnlyTransactionCount": 0
  }
}
```

**Figure 3-10: OpenDayLight curl data output**

We execute the following command to create the network topology, using mininet emulator and with the defined topology as shown in Figure 3-11:

```
sudo mn --topo linear,3 --mac --controller=remote,ip=controller-
ip,port=6633 -switch ovs,protocols=OpenFlow13
```



*Figure 3-11: OpenDayLight web GUI*

The monitoring can be done with the cluster monitor tool. This tool provides real-time visualisation of the cluster members' roles for all shards in the configuration data store. This is useful in understanding the clustering behaviour of controllers when they are in different roles and states such as leader/follower, isolated (i.e., the controller has no followers), shutdown or rebooted. The tool assumes that all cluster members have the same shards. The file is located in `test/tools/clustering/cluster-monitor/`. We needed to modify the file `monitor.py` by the IP of each controller.

In the first part of Figure 3-12 (see next page), all controllers are in follower mode after cluster initialisation in member 1 state. In member 2 state of Figure 3-12, all controllers are eligible to be elected as a leader of the cluster. In member 3, state a controller with topology information is elected as the leader of the cluster after a voting process.

Though ODL supports clustering using Raft and AKKA framework, it requires pre-configuration of cluster members and shards. This requires pre-determination of the number of controllers and the distribution of data store. Any modification to the cluster configuration requires re-boot of the entire cluster members along with modification of number of configuration files. Such approach limits the flexibility in scaling up and down the number of controller nodes and shard members in real-time according to the underlying network conditions.

### 3.2.1.2 ONOS

One of distributed controllers beside ODL is open network operating system (ONOS) [ONOS18]. ONOS is a multi-module project whose modules are managed as open services gateway initiative (OSGi) bundles with two main protocols: Raft and anti-entropy protocols. ONOS is a distributed controller, implemented using an in-memory based key-value data storage system which is adopted to improve the data read and write efficiency. Different data models use different data consistency methods for actual needs. These consistency methods include eventual consistency and strong consistency.

Though there is a distributed and clustering feature is provided in ONOS 1.13, it lacks in terms of dynamicity (e.g. scale up and down of controllers' instances with network load awareness)

	people	default	car	toaster	car-people	inventory
<3						
member-1	Follower	Follower	Follower	Follower	Follower	Follower
member-2	down	down	down	down	down	down
member-3	down	down	down	down	down	down
<3						
member-1	Candidate	Candidate	Candidate	Candidate	Candidate	Candidate
member-2	down	down	down	down	down	down
member-3	down	down	down	down	down	down
<3						
member-1	Candidate	Candidate	Candidate	Candidate	Candidate	Candidate
member-2	Follower	Follower	Follower	Follower	Follower	Follower
member-3	down	down	down	down	down	down
<3						
member-1	Leader	Leader	Leader	Leader	Leader	Leader
member-2	Follower	Follower	Follower	Follower	Follower	Follower
member-3	down	down	down	down	down	down
<3						
member-1	Leader	Leader	Leader	Leader	Leader	Leader
member-2	Follower	Follower	Follower	Follower	Follower	Follower
member-3	Follower	Follower	Follower	Follower	Follower	Follower

Figure 3-12: OpenDayLight cluster monitor tool - election procedure of shared leaders

### Consistency models: eventual consistency

There are two consistency models, namely the eventual consistency and strong consistency models. The former model provides a weak form of consistency, in the sense that a data update on a certain controller will be eventually updated on all the nodes. This implies that for some time, some nodes may read different values from the actual updated ones; however, after some time, all the nodes will have the updated values, given that they are able to communicate. This model is employed in systems which require high availability. The anti-entropy protocol, implemented in ONOS, supports this consistency model (a network topology store to describe the network topology in terms of links, hosts, and switches). Anti-entropy protocol is a type of gossip-based protocols that realises synchronisation between multiple instances of the controller. It is used to manage the network topology store. It is based on a simple gossip algorithm in which each controller chooses at random another controller in the cluster every 5 seconds and then sends a message to compare the actual content of its store with the one available at the other controller. This synchronisation message includes the information about the elements (switches, links and hosts) present in the topology, as well as the removed elements, i.e., weak synchronisation. It does not guarantee consistency.

### Consistency models: strong consistency

This model ensures that each controller always reads the latest version of data. If certain data have not been updated yet at all (or most of) the controllers, then they are denied the permission to be read, thereby giving availability less priority in favour of consistency. The Raft consensus protocol implemented in ONOS supports this consistency model (a mastership store to keep mapping between switches and their master(s)).

### Raft protocol

It is a recently proposed scheme [SDN-COMPARE] which provides strong consistency for the mastership store in ONOS. A Raft implementation requires a cluster of nodes, each having a database termed as the “log”, which is replicated in all the nodes. Each update is appended to this shared data structure. The consistency is coordinated by a leader node in the cluster, which is responsible for receiving update requests from all the other nodes and later relaying log updates to the other nodes. Once most of the followers have acknowledged the updates, this is committed to the log [ODL] except for the eventual consistency map, which is handled by anti-entropy protocol. All other primitives are operated

by Raft-based log system. The Raft protocol is able to achieve strong consistency, whereas the anti-entropy protocol, aka the gossip protocol, is suited for ultimate consistency.

### ***Distributed data and cluster configuration***

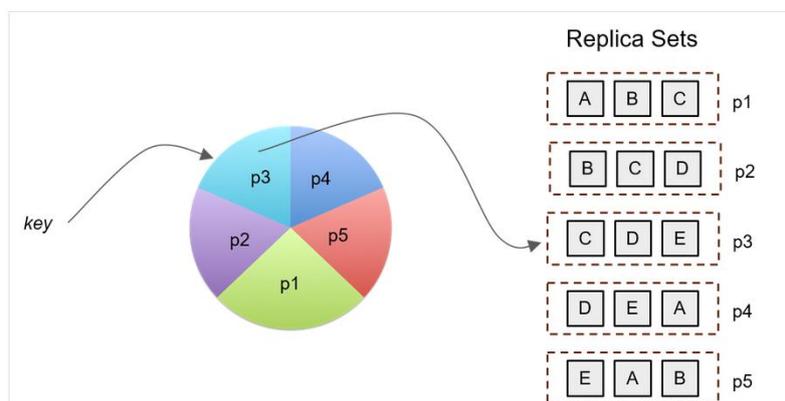
In ONOS, for scalability issues, multiple instances of the Raft algorithm run simultaneously, each of which is responsible for the synchronisation of a data subset. This is due to the anti-entropy protocol; as each controller randomly selects another controller to synchronise the network topology periodically. The controllers are always logically connected in a full mesh. ONOS uses port 9876 for the inter-controller communications. For clustering, when a cluster is formed, a metadata file is created that tells each ONOS node where other nodes in the cluster are.

ONOS uses a partitioned Raft cluster with ‘n’ partitions, with each partition being a separate 3-node instance of the Raft protocol by default, to decrease the amount of overhead for scaling the cluster size. This means that by default, a cluster of any size can technically only tolerate the failure of one node before losing partial availability. In some scenarios, a cluster of ten nodes can experience some unavailability issues within some partitions even when only two out of the ten nodes fail. This can occur if these unavailable nodes are members of a partition that has only three-member nodes and that can cause loss of some partition information. This is due to the fact that the loss of one partition can affect all partitions for certain types of operations. In any deployment, it is necessary to place these 3 nodes in different failure zones or use a different partitioning scheme to have high availability.

So, the fault tolerance of the controller cluster effectively depends on the number of nodes in each partition rather than the number of nodes in the cluster (at least two nodes in each partition to conform to raft election). Partitions are designed to scale performance, not fault tolerance. But for large clusters, there is the possibility to conceivably scale the partition size to five nodes to tolerate the failure of two nodes in any partition. This will obviously hurt performance, but only slightly. Quorum based consensus algorithms scale rather well for up to seven nodes, since they tend to take advantage of the fastest majority of nodes in the cluster. In the above scheme, each node is a member of a similar number of partitions. An administrator decides whether a different partitioning strategy is better, e.g., a more powerful node, i.e., one that has more resources compared to others, can accommodate more partitions, due to its resource capacity.

### ***Raft for data partition***

In order to improve data access efficiency, ONOS data uses partitions’ storage which is managed by the raft algorithm. Access to the Raft-based storage is done using a client-server model. Each partition on the storage has some of the ONOS instances that are acting as the server for that partition, but a given instance will not necessarily be a server for every partition in the system. Each instance is usually a client of each partition because it may need to access data from partitions that it is not hosting.



***Figure 3-13: Data partitions and replication set***

Once the leader of one partition is down, a new leader will be elected, but ONOS keeps the same partition. When the previous leader is up again, the current leader continues to lead the partition until its failure. In terms of log replication, after replicating a new entry to all servers along with the reception of an acknowledgement, the leader commits the entry and sends the response to the client by sending

the remote procedure call (RPC) and conveys that the new entry has already been committed so other servers can commit to their state machine. With two nodes only, it is possible to elect a leader but not to commit a new entry. Since there is no majority of votes, the log stays uncommitted. If a new leader is found, both nodes will become followers and synchronise the logs with the new leader. Uncommitted entries will be deleted. Figure 3-13 explains the partition set of possible members in a cluster.

### Scalability Analysis

In this section, we describe the scalability analysis in order to know how ONOS survives a network failure. We begin with five partitions and five controller nodes (five containers) as members of each partition, as shown in Figure 3-14. Each partition stores different data of the network. Data in each partition is replicated to every member. If one partition is down, network will fail. If a controller is down, the network may or may not fail; depending on the number of available controllers to handle a partition. In Figure 3-15, one controller is down. In this case, the network will not be affected as there is backup from other controllers. As shown in Figure 3-16, controllers are maintained stopped.

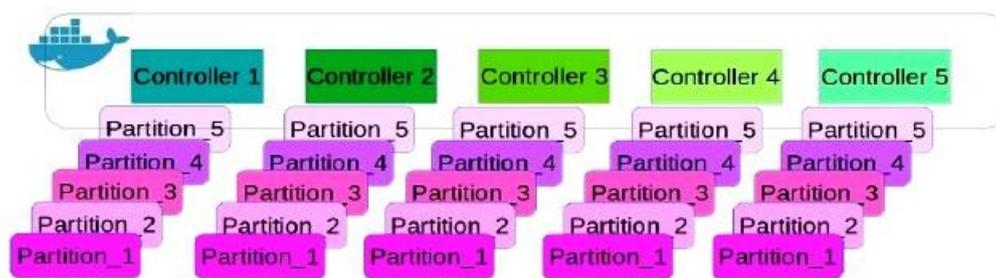


Figure 3-14: Cluster with 5 nodes

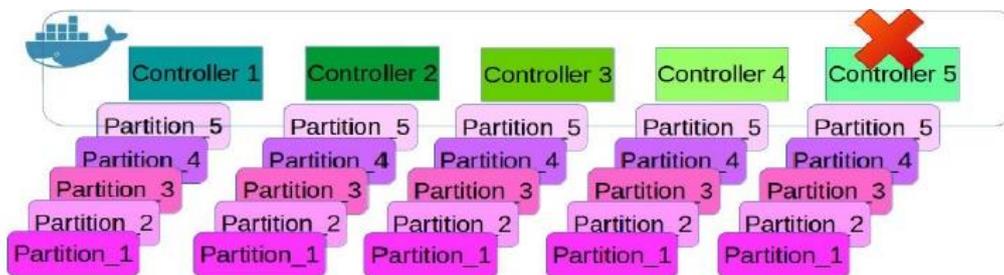


Figure 3-15: Cluster with 1 node failure

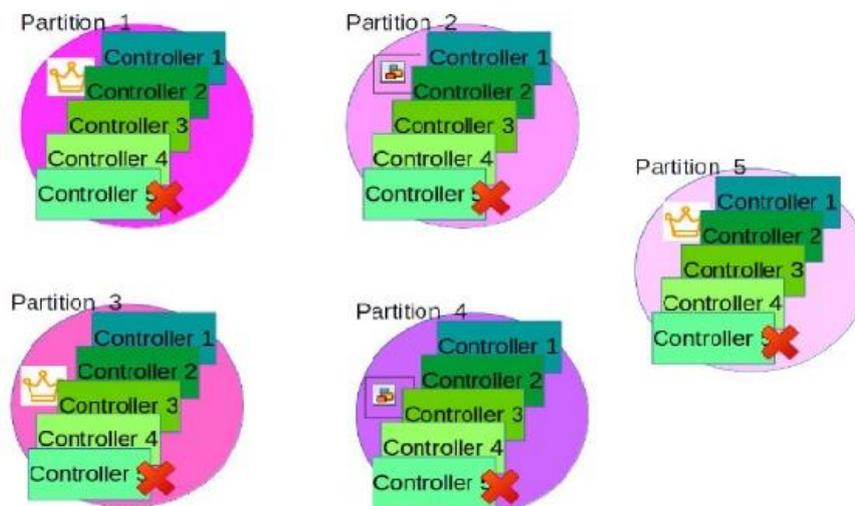


Figure 3-16: One node is down from partition perspective

The behaviour of the cluster during node failures is related to raft consensus, as there will be no majority when there is one controller. The controller keeps waiting for votes until a time-out occurs and the system goes down. Concerns come up when a user wants to change the cluster configuration. ONOS needs to stop the running system (existing cluster) and restart with new cluster metadata. In a typical low latency network, such as mobile networks, it is unacceptable to bring down the entire controller cluster during network operation (cf. Figure 3-17). The second issue found is that there should be at least two controllers to participate in one partition since ONOS depends on the Raft algorithm, where the master-slave paradigm applies. This will impact resilience when there are only two controllers left.

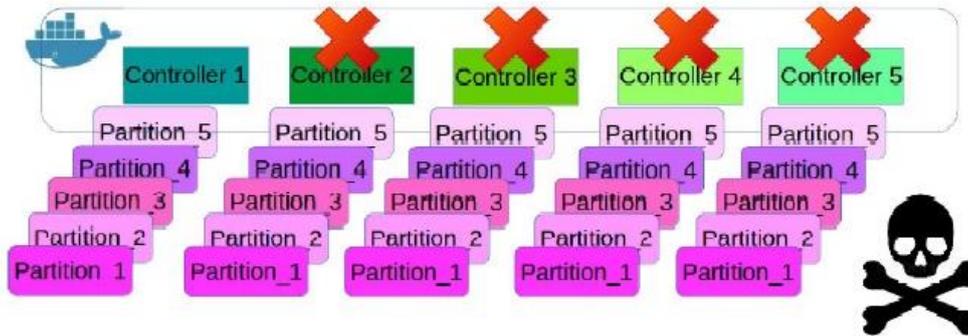


Figure 3-17: Four nodes are down

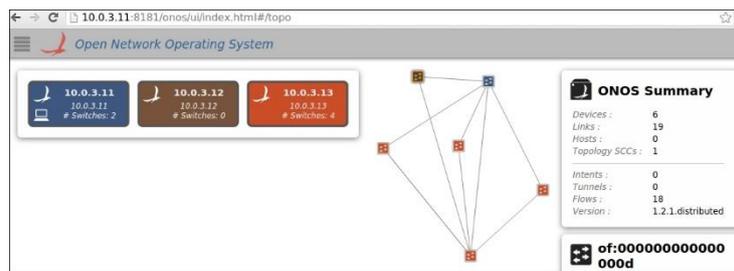


Figure 3-18: ONOS clustering

### 3.2.2 Scalable controller framework

#### Architecture

In order to make the controller cluster scalable and able to dynamically add or remove number of nodes in the cluster, we extended the distributed core to be dynamic and able to accept any new nodes joining the cluster and synchronise with the running nodes as shown Figure 3-19. Data can be synchronised to a new controller as a new member of the existing partitions. Also, a new leader election begins as a new member joins the existing cluster. Such framework is scalable and the number of nodes in the cluster can be adjusted according to the ongoing traffic load in the network.

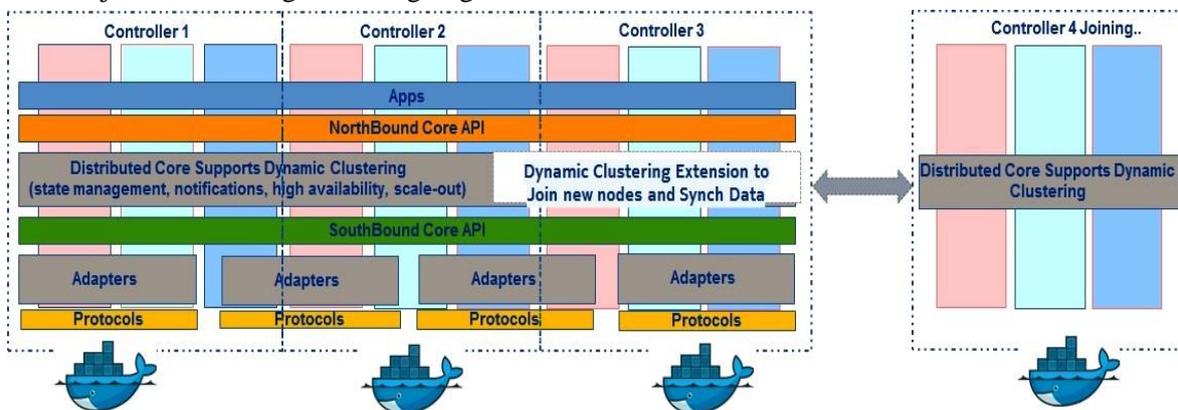
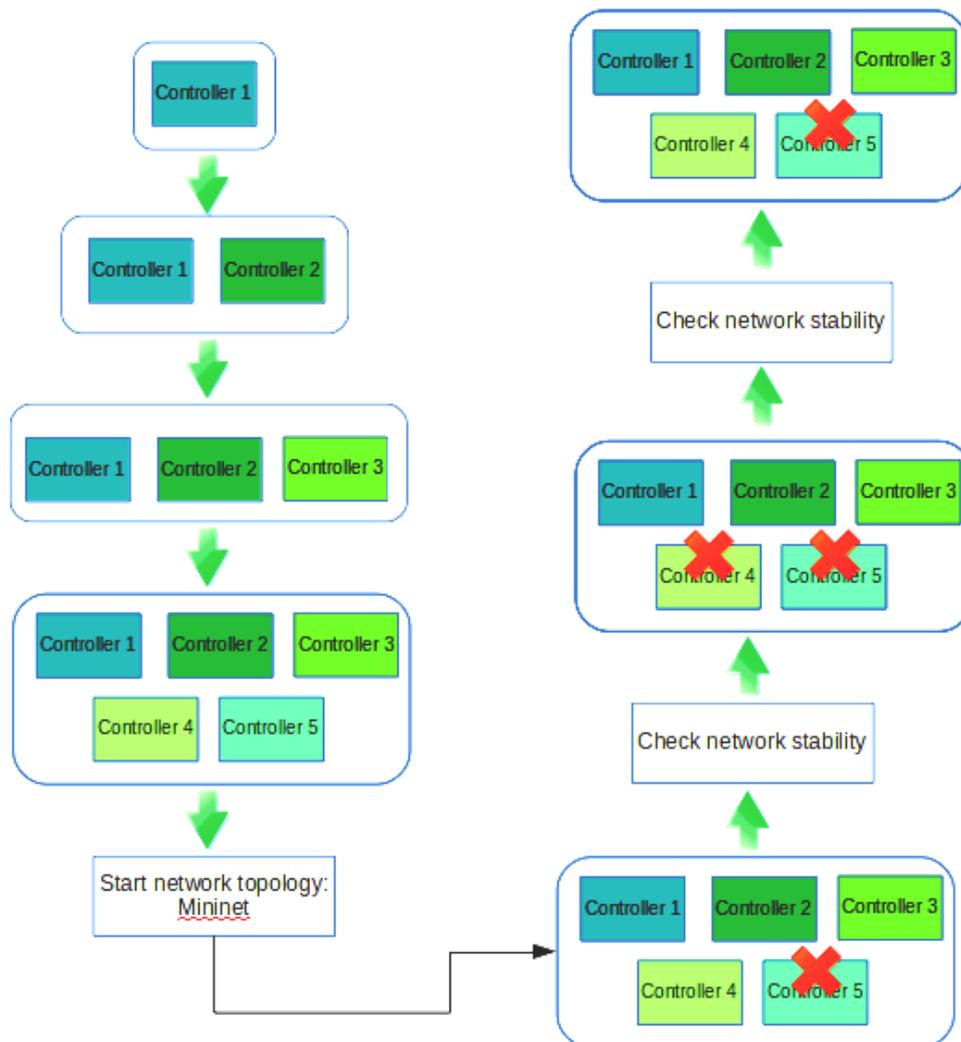


Figure 3-19: Scalable controller framework

### Evaluation

In order to verify the scalability of such frameworks, we have started a single controller and later increased the number of controllers in the cluster. We also emulated an Open Virtual Switch (OVS) network with mininet emulator to verify the stability of the platform as shown in Figure 3-20. We used the ping test between two hosts at the edge of the network to verify the connectivity. The test is continued by stopping and re-starting a number of controllers in the network. The outcome is that a ping continues to be successful during the failure of nodes as well as during the modification of the cluster.



**Figure 3-20: Scalable controller framework – evaluation scenario**

In order to analyse the performance of the developed scalable controller framework, computing resource usage is measured by increasing the number of controller nodes in the test bed. The test bed consists of a mesh network with 40 OVSs and 40 virtual hosts emulated using mininet emulator on a PC equipped with 16 GB RAM and 4 CPUs. The scalable controller framework is connected to the network via OpenFlow protocol. The default packet forwarding application on the controller is activated to support Internet Control Message Protocol (ICMP) packet transmission between hosts. As it can be derived from Figure 3-21 (see next page), if the number of controllers in the cluster is increased, the CPU load of each controller can be reduced and hence the controller performance improves.

The figure consists of three terminal screenshots showing the performance of a scalable controller framework across multiple containers. Each screenshot displays a table with the following columns: CONTAINER ID, NAME, CPU %, MEM USAGE / LIMIT, and MEM %.

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %
a62c1f97043c	demo-node1	33.65%	1.337GiB / 15.57GiB	8.59%
905362e876aa	demo-node2	30.63%	1.718GiB / 15.57GiB	11.04%
1d8d6b4c013a	demo-node3	47.43%	1.609GiB / 15.57GiB	10.34%
4d69c770e36f	demo-node4	56.98%	1.433GiB / 15.57GiB	9.20%
35637d27a4ce	demo-node5	56.48%	249MiB / 15.57GiB	1.56%

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %
a62c1f97043c	demo-node1	44.59%	1.3GiB / 15.57GiB	8.35%
905362e876aa	demo-node2	39.49%	1.72GiB / 15.57GiB	11.05%
1d8d6b4c013a	demo-node3	33.50%	1.623GiB / 15.57GiB	10.43%
4d69c770e36f	demo-node4	34.32%	1.477GiB / 15.57GiB	9.48%
35637d27a4ce	demo-node5	59.63%	497.6MiB / 15.57GiB	3.12%
6ec7396b621e	demo-node6	50.15%	118.8MiB / 15.57GiB	0.75%

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %
a62c1f97043c	demo-node1	16.68%	1.271GiB / 15.57GiB	8.16%
905362e876aa	demo-node2	18.09%	1.739GiB / 15.57GiB	11.17%
1d8d6b4c013a	demo-node3	11.36%	1.661GiB / 15.57GiB	10.67%
4d69c770e36f	demo-node4	19.00%	1.568GiB / 15.57GiB	10.07%
35637d27a4ce	demo-node5	27.60%	817MiB / 15.57GiB	5.12%
6ec7396b621e	demo-node6	33.43%	501.9MiB / 15.57GiB	3.15%
614b5648a1a8	demo-node7	24.94%	313.4MiB / 15.57GiB	1.97%

Figure 3-21: Scalable controller framework – performance measurement

### 3.3 5G Islands: evaluating migration cost and outage loss for context-aware NF migration

While redundancy can be exploited to improve the telco cloud resilience, it has to be predictively prepared and periodically updated, which expenses an extra operational expenditure (OPEX). Moreover, in some scenarios, frequent routine preparation and update of redundancy can generate a significant backhaul traffic, which depraves the telco cloud resilience itself. As a solution, this section describes a novel approach of selective VNF migration by means to balance the resilience and OPEX. Due to the autonomous nature of such VNF migration, in the telco cloud resilience framework of WP3 of 5G-MoNArch this approach is dubbed “5G Islands” and represents a separate architecture enabler which resides in the *Management and Orchestration layer* of the 5G-MoNArch architecture (c.f. Figure 1-3).

#### 3.3.1 Central-to-edge VNF migration

Telecommunication networks can suffer from malfunctions or service degradations, and therefore fail to deliver the QoS promised in SLA to the customers. In virtualised networks, such phenomenon can occur with every VNF, which we refer to as VNF outage. To improve the network resilience and enable autonomous failsafe operations against such VNF outages, different technologies are available, including state management, VNF migration and rollback recovery. All these methods require some sort of VNF redundancy that is periodically updated.

In the perspective of network resilience, VNF outages can occur either at the NF server or at the network infrastructure. Specifically, for central cloud VNFs, these refer to the central cloud server and the backhaul connection, respectively. In the case of backhaul connection outage, neither state management nor rollback recovery, but only a VNF migration from the central cloud to the local edge cloud, can effectively recover the service from the VNF outage.

Meanwhile, there is another reason in the perspective of autonomous failsafe operation to prefer the central-to-edge (C2E) VNF migration to the other solutions. Generally, sources that can trigger central cloud VNF outages consist of planned disconnections, cyber-attacks and unintentional errors. Among them, cyber-attacks are usually followed by attempts of illegal access to confidential information; unintentional critical network infrastructure malfunctions can be usually caused by dangerous disasters and emergencies such as fire, explosion, earthquake and massive blackout. In both cases, autonomous

failsafe operations shall be executed for both the telecommunication network itself and the relevant vertical systems, in order to minimise the loss. A temporary solution of the local edge cloud services is thereby required to support such operations; such solution shall be decoupled from the environments – including the central cloud - to the most possible extent.

### 3.3.2 Migration cost versus outage loss

The solution of C2E VNF migration requires a preventive and periodical update of the central cloud VNF redundancies in edge clouds, because the migration process: 1) may fail when the central cloud VNF outage already occurs; and 2) generates extra backhaul traffic that may deprave the conditions. However, the update process generates extra cost in two ways: 1) an extra data traffic over the backhaul network will be generated to transmit the necessary data; 2) an additional OPEX is needed to maintain the redundancy on the local edge cloud server(s). If the network continuously updates all redundancies of every VNF in every edge cloud, it will lead to a huge sum of the migration cost which is economically unaffordable or at least inefficient.

Instead, we apply a selective redundancy update solution, the *5G Island (5GI)*, in which the update process is triggered, only when a significant outage risk of certain VNF in certain edge cloud is detected. This leads to a trade-off between the network resilience and the OPEX when deploying the C2E VNF migration solution.

Given a certain central cloud VNF and a certain local edge cloud, the expected outage loss of not updating this VNF in this edge cloud is

$$c_o = \sum_{u \in U} E\{t_{o,u}\}l$$

where  $U$  is the set of all UEs in local and neighbour edge clouds,  $t_{o,u}$  is the time of  $u$  suffering from this VNF outage in this local edge cloud, and  $l$  is the loss caused by this VNF outage per unit time per UE.

A rational decision of redundancy updating can be made by comparing  $c_o$  with the migration cost  $c_m$  of this VNF: the VNF should be updated if and only if  $c_m \leq c_o$ .

### 3.3.3 Estimating outage loss

While the migration cost can be estimated straightforwardly by the MNO according to the amount of data as well as unit cost of backhaul traffic and local storage required, it is challenging to estimate the expected outage loss. Depending on the nature of VNF, two different approaches can be applied.

#### 3.3.3.1 Stateful VNF

A VNF is called stateful if it depends on the profiles of served subscribers. In this case, the edge cloud (EC) must know the exact identification of every  $u \in U$  to estimate  $c_o$ :

$$E\{t_{o,u}\} = p_o \eta_u \int_{t=0}^T f_{arr,u}(t) \int_{\tau=t}^T f_{stay,u}(\tau) \tau d\tau dt$$

where  $T$  is the update period,  $p_o$  is the central cloud VNF outage probability in the next period,  $\eta_u$  is the VNF duty rate for  $u$ ,  $f_{arr,u}(t)$  and  $f_{stay,u}(\tau)$  are the PDFs of  $u$ 's arrival in  $t$  and  $u$ 's stay time in next  $\tau$ , respectively. For every UE  $u \in U$ ,  $\eta_u$ ,  $f_{arr,u}(t)$  and  $f_{stay,u}(\tau)$  are individually computed with supports of service log, handover prediction, etc. For every VNF-EC pair,  $p_o$  is individually estimated through service monitoring.

#### 3.3.3.2 Stateless VNF

A VNF is called stateless if it only contains functionalities that are independent of the subscriber's identity. In this case, the EC only needs to know the statistical knowledge to estimate  $c_o$ :

$$c_o = \bar{N} \bar{\eta} \bar{\tau}_{stay} p_o$$

where  $\bar{N}$ ,  $\bar{\eta}$  and  $\bar{\tau}_{stay}$  are the average number of served UEs, the average VNF duty rate and the average stay time in every  $T$  per UE, respectively.  $\bar{N}$  and  $\bar{\eta}$  can be obtained from the AMF while  $\bar{\tau}_{stay}$  can be obtained from the VNFM.

### 3.3.4 Simulation analysis

Consider two stateless central cloud VNFs F1 and F2, for which  $c_m$  is  $20Tl$  and  $100Tl$ , respectively.

The mobility is simulated by considering random walking UEs in an EC covering a  $4\pi\text{km}^2$  urban area surrounded by four suburban and rural areas. UEs are uniformly distributed in four mobility classes (high, medium, low and still, cf. Table 3-1), and each area has an individual vector describing different mobility penalties to the UEs of different mobility classes, cf. Table 3-2. This simulation returns in equilibrium a UE density of  $187.23/\text{km}^2$  in the urban area, as illustrated in Figure 3-22.

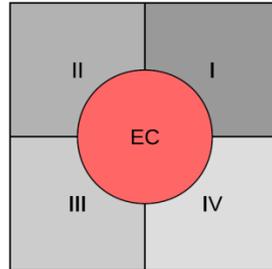


Figure 3-22: Map of mobility simulation

Table 3-1: Basic motion speed of different mobility classes

Mobility Class	STILL	LOW	MEDIUM	HIGH
Basic Speed (m/s)	0	$\sim N(1.5, 0.5)$	$\sim N(10, 2)$	$\sim N(40, 5)$

Table 3-2: Mobility penalty factors in different areas

	EC	I	II	III	IV
LOW	1	1	1	1	1
MEDIUM	0.7	1	0.9	0.8	0.9
HIGH	0.2	1	0.9	0.85	0.8

The status of VNF availability is simulated as a Markov process, which takes account of the effort of recovery, as shown in Figure 3-23.

We compare the performance of 5GI and two naïve benchmark update policies: “never” where no VNF migration is available and “always” where the VNF redundancy is updated every period independent of the context information. The update period  $T$  is set to 30 minutes, and in total 30 days of operation is simulated. Results are shown Figure 3-24. It can be observed that 5GI provides a high resilience level with a low migration cost. Note that the sum of migration cost and outage loss of 5GI exceeds the outage loss of “never” policy in the simulation results. This phenomenon is caused by the variance of estimators, so that it will vanish in long term, and can be reduced by deploying more advanced estimators.

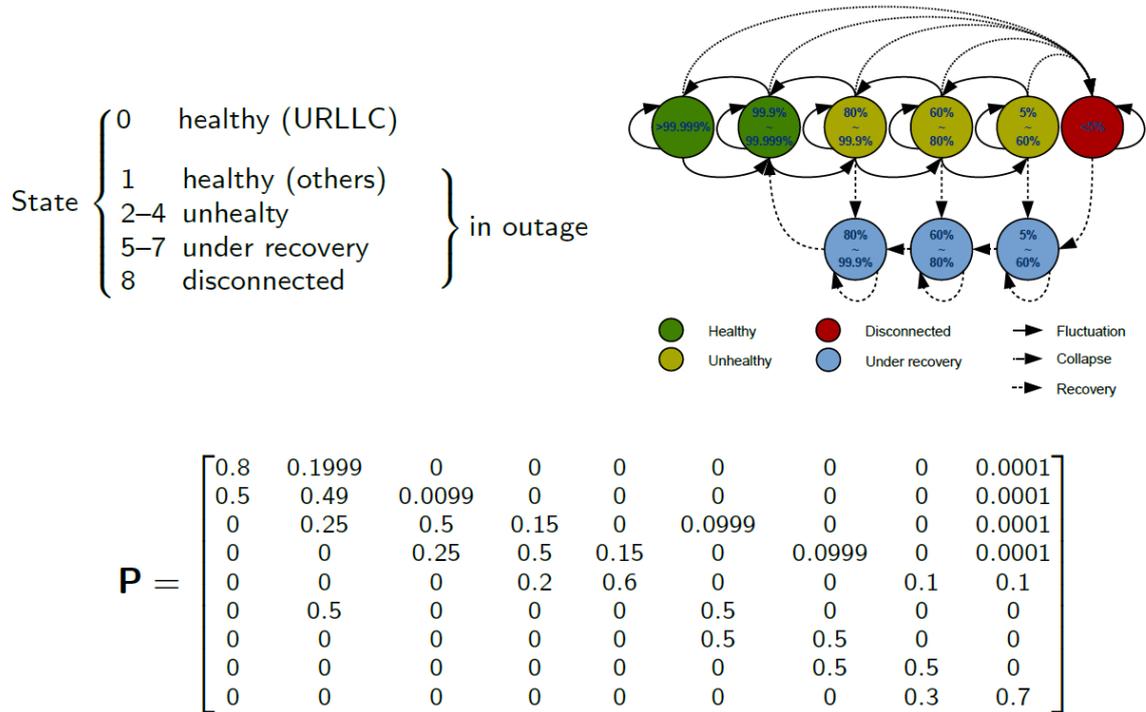


Figure 3-23: Markov chain to simulate the central cloud VNF outage

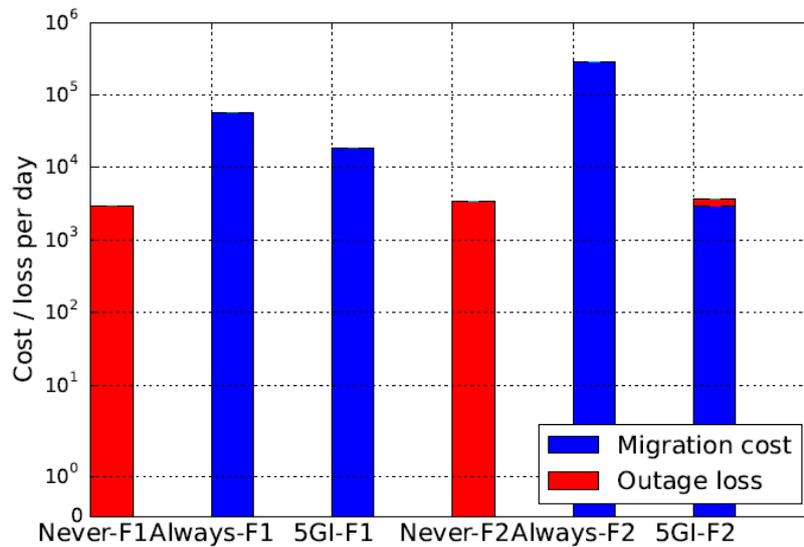


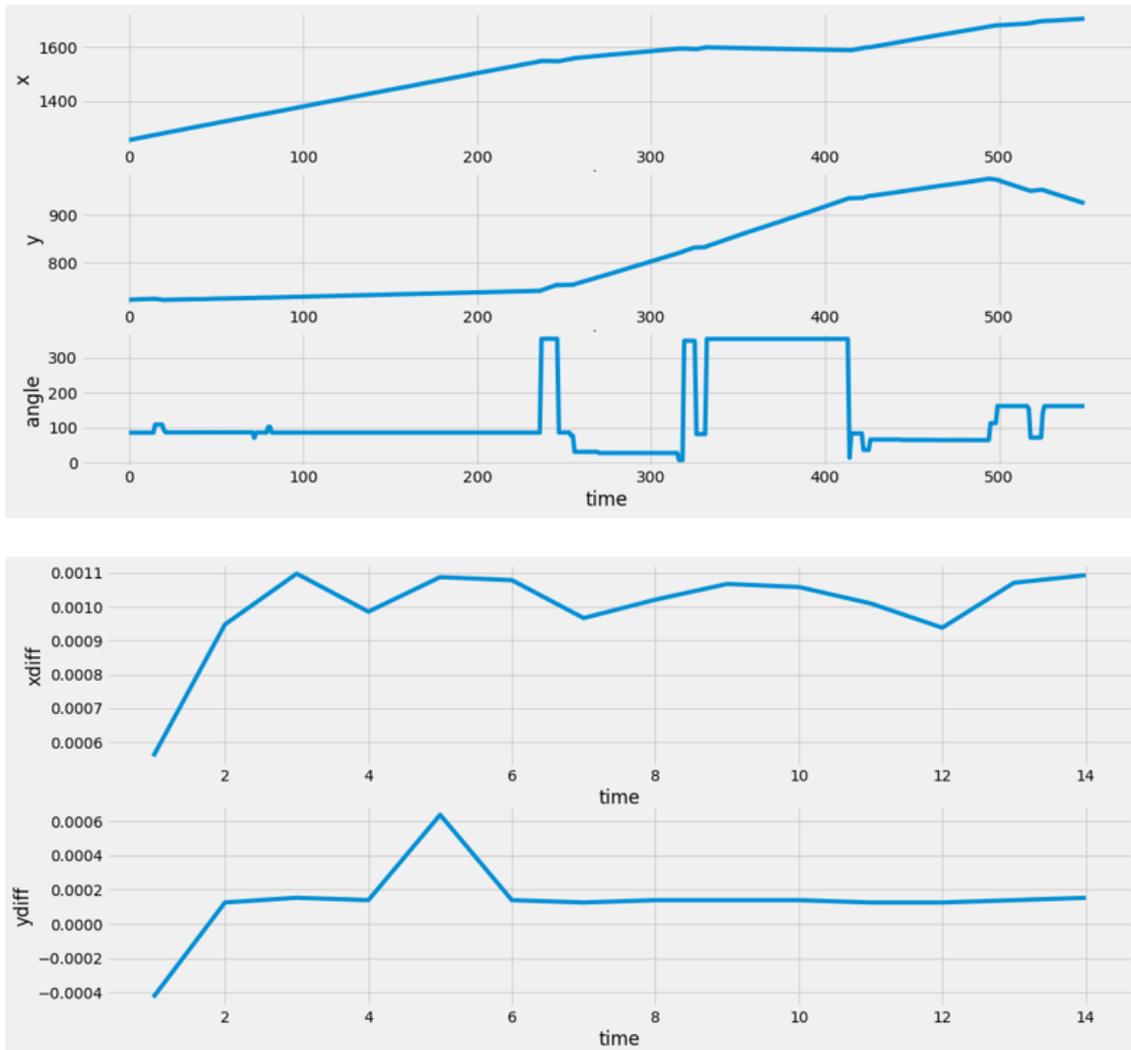
Figure 3-24: Cost/loss per day: simulation results

### 3.3.5 Neural network assisted 5G Island for stateful VNFs

So far, the effectiveness of 5G Island on stateless VNFs has been demonstrated. For stateful VNFs, as discussed in 3.3.3.1, a precise mobility prediction of every individual UE will be needed. Although this function has not been fully implemented in scope of 5G-MoNArch, some initial attempts have been made.

The well-known technique of Long Short-Term Memory (LSTM) network can be deployed to execute online prediction of UE traces. First, the trace information of every UE, containing its current position and angle of motion, shall be provided to the system. Every trace is then divided into short-term segments according to turning angle, and every segment then differentiated by the order of 1, so that the stationarity of data sequence can be guaranteed. Every differentiated segment is then independently

normalised in magnitude and value-padded to the length of the longest segment. The output is then provided to a 2-layer LSTM network. The first layer, which is the masking layer, masks the padded values in the input sequences; while the second layer executes a one-step-forward online prediction with a Dropout regularisation to reduce overfitting effect. An example prediction of a simulated pedestrian user in the urban area of Munich city centre is illustrated in Figure 3-25.



**Figure 3-25: An example output of UE trace online prediction with LSTM network**

To efficiently train the neural network, dataset of UE traces in various scenarios can be offline generated with off-the-shelf simulators. One example is the Simulation of Urban MObility (SUMO), which is a Python-based, open-source and free-to-use simulator that supports flexible environment specification of heterogeneous traffics (vehicles, public transport, bicycles and pedestrians). It also supports importing real maps from third-party map service providers to simplify the generation of simulation map.

As a remaining issue for future study, the LSTM neural network shall be extended to output the spatial PDF of the UE arrival on the next step(s), instead of the most likely destination. One possible technique to achieve this aim is the mixture density network [B94].

## 4 Security on 5G networks

The reliability and resilience mechanisms presented, respectively, in Chapter 2 and Chapter 3, need to be accompanied with a proper management of security. Security threats put at risk 5G infrastructures, which are exposed to a myriad of different security incidents and to malicious attackers. This is especially critical when 5G infrastructures are providing with critical services with strict reliability and resilience requirements. Therefore, guaranteeing the prevention, detection and reaction to security incidents are paramount to guarantee the integrity of the infrastructure protected and of the services provided over it.

This section documents the progress on the security aspects which were reported in [5GM-D3.1], and complements the initial conceptual analysis with a simulation campaign. With reference to Figure 1-3, where the enablers of the WP3 framework of 5G-MoNArch are depicted on the basis of the overall 5G-MoNArch architecture, this Section elaborates on the functionalities of the security-related functional modules, residing in the controller and network layer of Figure 1-3.

In this respect, a comprehensive security threat analysis is done in Section 4.1, with special emphasis on the implications in the Hamburg Sea Port testbed. Section 4.2 evaluates in-depth the suitability of the STZ-based approach for 5G networks, along with the main mechanisms to manage them. Then, Section 4.3 contains simulation-based analyses which were used in the context of WP3 of 5G-MoNArch for evaluating the security-related concepts proposed in [5GM-D3.1]. In particular, the testbed built to validate the Security Trust Zone (STZ) approach introduced in [5GM-D3.1] is presented in Section 4.3.1. The developed tool can serve as potential security probes to include in a STZ. In Section 4.3.2, details are added pertaining to the developed network behavioural analysis, elaborating on a graph-based anomaly detection method along with its extension that is based upon a machine learning approach.

### 4.1 Threat analysis on main 5G components

Capitalising on a holistic security approach described in Chapter 1, in this section various security considerations for the main as well as peripheral 5G components of the Hamburg Sea Port use case are put forward. The scope of this section is to integrate –to the best possible extent– the outcome of this analysis with the concept of STZs or implement new mechanisms as standalone security solutions.

The main security areas that are involved in the Hamburg Sea Port use-case can be extracted from Figure 4-1. In this respect, devices refer obviously to the end-devices and sensors deployed throughout the port, namely the ship sensors for pollution measurements, the smart traffic lights for enhanced port operations, and the end-user goggles and tablets for assisted engineering capabilities of the port's personnel. The second part (referred to as 5G network in Figure 4-1) refers to the main 5G network elements of the Hamburg Sea Port use case, and network slicing refers to all software and hardware components that mechanise the required network slices in the use-case. We examine these areas separately in the next paragraphs.

#### 4.1.1 Device security

The Hamburg Sea Port use-case encompasses all the characteristics of a typical industrial Internet of Things (IoT) application. This is dictated not only by the significant number of sensors, but also by the different requirements posed by each sensor-family. Those requirements fall within the classes of URLLC, or massive Machine Type Communications (mMTC) or eMBB. As such, we need to take a closer look to device security.

The first consideration relates to the support of universal subscriber identity module (USIM)/ universal integrated circuit card (UICC) cards. Since the end-devices need to authenticate themselves as typical mobile devices in the 5G network, it is expected that they support at least a secure computing environment for storing critical keying material and performing sensitive cryptographic operations. Since many of the sensors are physically exposed, this turns into a major security concern. Integrated modules like Hardware Secure Modules (HSM) may mitigate the risk, since there exist already lightweight HSM-smart card solutions even for IoT devices [GEM18].

Another issue relates to the support of cryptographic operations, in parallel to addressing the stringent requirements of some use-cases, e.g., the URLLC application of smart traffic light systems. In this case,

a crypto-algorithm that would not hinder the end-to-end latency is preferred. There exist several tailored crypto-solutions for such use-cases, although their standardisation is still a work in progress [NIST18]. Nevertheless, the well-established AES algorithm has exhibited remarkable security over the years and its lightweight versions might be able to address the performance requirements of most use-cases in the port. Careful examination of performance figures can reveal whether standardised AES-based libraries are sufficient for each case.

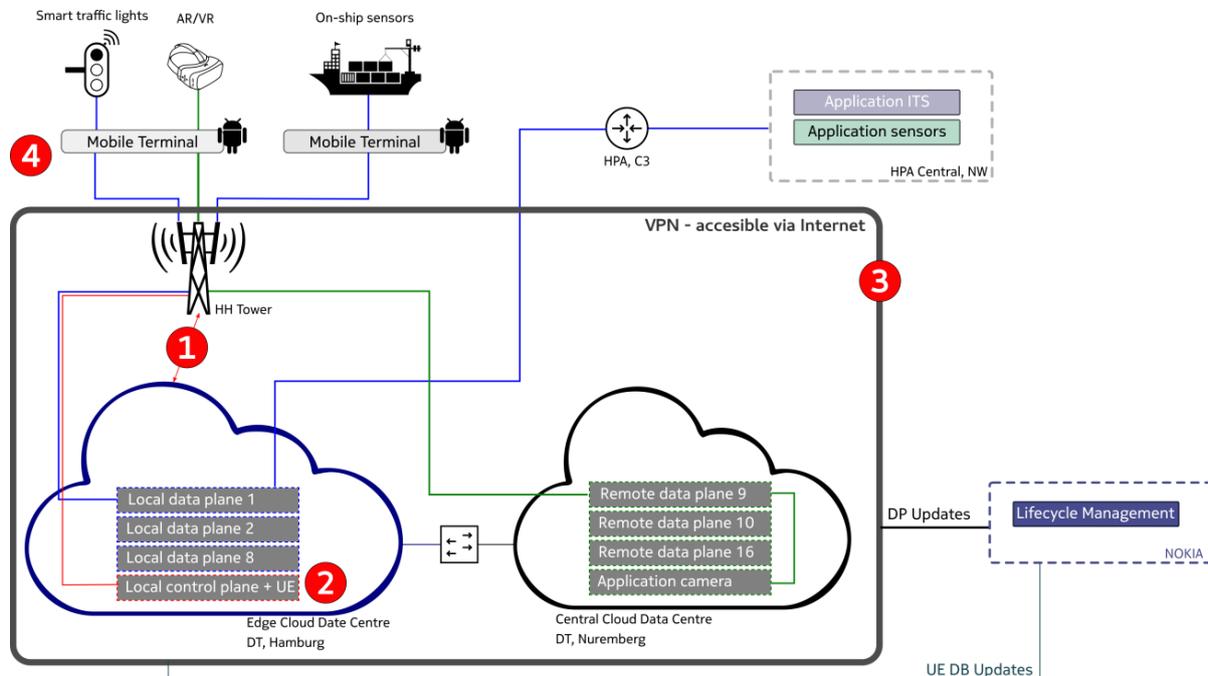


Figure 4-1: General architecture of the Hamburg Sea Port use-case

Next, a key security challenge, that is closely related to the physical access to the devices, is physical attacks. Physical attacks may include intrusive and non-intrusive cases, with the latter being of great importance, assuming that dedicated security personnel could hinder intrusive attacks. Non-intrusive attacks may refer, for example, to electromagnetic or power analysis attacks, during which the attacker may be able to expose keying material or other sensitive information only by analysing the electromagnetic emissions of the sensor or by examining its power traces during its operation. In these cases, the devices should adhere to at least all mandatory packaging regulations or incorporate countermeasures (could be in hardware) to mitigate the issue, e.g., trivial modifications in the computations data-path may produce normalised power dissipation traces or electromagnetic emissions without a significant hardware footprint. Closely related are jamming attacks, enabled by the close proximity of human presence to the devices. The mechanisms of STZs could play an important role here, since they can be utilised to detect jamming attempts and at least inform on an ongoing attack within a reasonable timeframe.

Finally, as a general remark, we should not underestimate the design flow of secure device manufacturing. Security should be enabled by design and not as an on-top feature. This means that proper assurance of safe programming as well as secure hardware design should be provided by the device manufacturers. This can be achieved by providing certifications of safe programming and hardware designing techniques. As a last word, the patching process of end devices and sensors is also considered as a risky operation, which means that any firmware or software updates should be executed in a secure environment and via well-defined procedures.

#### 4.1.2 Security in 5G networks

When it comes to 5G security, 3GPP standardisation work may serve as the main anchor point for our baseline security [3GPP 33.501]. Indeed, the latest version addresses most of the security procedures

required in a 5G network environment namely from gNB configuration to core network and UE-5G network connectivity security. In the Hamburg Sea Port these requirements are considered as a prerequisite and in the next paragraphs we build further security considerations on top of 3GPP security work.

In Figure 4-1 the general architecture of the Hamburg Sea Port use case is illustrated. The numbered circles depict parts of the network where extra attention is required to safeguard a secure environment.

In part 1, the connection between the HH gNB Tower and the edge cloud infrastructure in DT Hamburg is considered. The main requirement here is a secure IPSEC connectivity between these two entities.

In part 2, the security of the unified data management (UDM) and authentication server function (AUSF) is examined (they provide partly the functionality of the home subscriber server (HSS) in the EPC domain). The main role of AUSF is to handle authentication requests both for 3GPP and non-3GPP access and inform the UDM upon successful or unsuccessful authentication of a subscriber. In the port scenario, these functions are part of the software stack of the local edge cloud, hence it is of paramount importance to assure a robust software implementation, especially against attacks which target subscribers' data, denial of service (DoS), etc.

Part 3 refers to the perimeter security provided by the virtual private network (VPN) infrastructure. Although this might seem as a trivial remark, there are several examples of badly configured VPN connections with detrimental results. For example, in [RZ11] a typical username/password enumeration vulnerability is described. According to this attack, a VPN server may reply with NULL PSK or not reply at all to an incorrect username. This enables the attacker to infer whether a username exists or not and create usernames with the same pattern. After discovery of a valid username, an attacker can receive the hash of the username's password from the server, by using PSK in aggressive mode. An offline crack of the hash that retrieves the password completes the attack (note that this is totally feasible because the probabilistic model of the VPN password hashing is not hidden) [RZ11]. Recently, another attack based on a 20-year old protocol (internet key exchange - IKEv1) was unveiled. The researchers in [FGS+18] proved that by using a Bleichenbacher oracle in IKEv1 they could break RSA encryption and RSA signature-based authentication, both in IKEv1 and IKEv2. Other risks involve insecure password storage on the server side, re-use of cryptographic parameters, etc. As such, a VPN infrastructure needs to be meticulously planned to avoid implementation flaws. At the same time, high awareness of the latest advances in cryptanalysis and insecure software/hardware implementations is required, in order to apply the respective patches as quickly as possible.

Finally, part 4 refers to the mobile terminals that serve as the connecting point between the UEs and the gNB tower. The mobile terminals are basically Android devices and, as is of course the case with all OS, they might exhibit risks and flaws especially when it comes to user handling and zero-day attacks. For example, researchers in Nightwatch Cybersecurity published recently a vulnerability that purportedly exposes information about a user's device to all applications running on the device. Similar to the VPN case, an up-to-date OS and proper awareness/education of the device administrators are required to ensure an error-free operation. As a last word of notice, mobile terminals and UEs allow for close proximity in their surrounding environment, mainly due to their physical footprint and location. As a result, jamming attacks in their radio interfaces cannot be excluded. In this case, the concept of STZs might prove useful to detect a jamming attack and apply mitigation actions, as will be explained in Section 4.1.3.

### 4.1.3 Network slicing security

Network slicing security is a rather generic term, as it relates, in the general sense, to software or hardware strategies and best practices that achieve the desired levels of security and slice isolation. To illustrate the variety of the topics related to network slicing security we offer a comprehensive enlistment in Table 4-1.

We analyse their status in 5G-MoNArch below:

- 1) The first risk concerns the common interfaces between the Hamburg port authority (HPA) and the MNO. A crucial point is the connection between the mobile terminals and the UEs. Fortunately, this connection is encrypted and, as a result, resource mixing is prohibited. Hence, a satisfactory level of security is achieved. The second risk relates to DoS attacks targeting

specific slices. In this case, the concept of STZs along with the possibility of deploying tailored HIDS systems and overload mechanisms is important and mitigates the risks.

- 2) The 3rd risk refers to attacks in inter-slice interfaces or in other words how is network slicing achieved. Again, the project offers several mechanisms to protect against such threats. Firstly, different slices forward packets to different PGWs, thus separation is achieved by design. On top, we have the option to apply encryption, thus strengthening isolation. Secondly, the connection between the PGW SGi interface and the tenant's data centre is a P2P static connection and as a result injection of traffic from one slice to another can be prohibited.

**Table 4-1: Network slicing security risks in Hamburg Sea Port use case**

Security risks		Countermeasures	Status
1	Attacks on common i/f (HPA - MNO)	Mobile Terminal ↔ UE is encrypted. Resource mixing is prohibited	✓
2	DoS	HIDS systems deployment, STZ, overload mechanisms (can be included)	✓
3	Attacks on inter-slice i/f	1) Different slices forward packets to different PGWs. → Separation is achieved 2) Encryption can be applied on top 3) Static p2p connection between PGW SGi i/f ↔ tenant's data centre. Injection is prohibited	✓
4	Procedural attacks: slice authentication, authorisation, Mgmt Insider attacks: make use of another slice for "cheaper" performance	STZ, access attempt monitoring, brute-force attacks, policies for level-access per user	✓
5	Malicious message routing among slices	IDS, traffic analysis, behavioural analysis, anomaly detection, STZ	✓
6	Attacks on management i/f	Secure MANO is provided - VPN	✓
7	SDN/NFV security	Robust software implementations, secure coding, overload control, cryptographic protection, integrity assurance of VNFs, logical separation of VNFs	?

- 3) The 4th risk refers to procedural attacks, namely slice authentication, authorisation and management. The concept of STZs offer access attempt monitoring and resistance against brute force attacks. In the extreme case of a malicious insider attack that may by-pass or misuse a slice to achieve for example better rates "for free", there is the possibility to implement level-access per user. In any case, those considerations are with the operator to implement and assess.
- 4) The 5<sup>th</sup> risk asserts the scenario of malicious message routing among slices (e.g., malware infected packets that may spread in the network). This topic is closely related to the slice isolation problem, but especially for malware we could leverage on behavioural analysis, traffic analysis, and anomaly detection mechanisms – all well compatible with the concept of STZs.
- 5) The 6<sup>th</sup> risk relates to the secure management of network slices. This topic poses no extreme risks, since a secure MANO interface has been meticulously implemented throughout the project as explained in previous deliverables. In the same direction, a proper VPN connection serves as an extra security anchor for secure MANO operations.
- 6) Finally, in the 7<sup>th</sup> risk we include the concept of NFV/SDN security, since they are closely related to the slicing framework. There is minimal standardisation work in this area, so our security assurance is associated with robust software implementations, secure coding, overload control, cryptographic protection, and the integrity assurance of VNFs. All these techniques, although generic, apply quite logically to the software stack of VNF/SDN. The point is that, as

SW systems, VNF/SDN should adhere to the best possible secure coding techniques, undergo thorough testing, and support a safe procedure for SW updates in case of security failures. Thus, the question mark relates to providing those assurances and compliance to safe coding techniques at least. As an extra measure – associated with additional hardware cost- is the logical separation of VNFs through tailored hardware.

#### **4.1.4 General remarks**

To summarise, 3GPP offers a solid security baseline that should be taken into consideration throughout the project [3GPP 33.501]. Especially for STZ, they can cover most of the detection capabilities for crucial parts of the network as we have shown for network slicing and inter/intra-interface connections between the various stakeholders. STZs can also be enhanced to detect physical compromise of the UEs or jamming attacks, provided that the manufacturers allow for the respective detection sensors.

A semi-open point remains the assurance of robust SDN/NFV implementations. This can be provided by means of secure coding techniques or logical separation. In any case, the stakeholder that implements the respective software stacks may need to provide at least a certification of secure coding methodology throughout the implementation phase.

When it comes to cryptographic protection, the employed mechanisms and crypto-libraries should adhere to the latest standards (avoiding weak crypto, misconfigurations, bad handling of libraries, etc). In this case, special care for secure upgrading and patching of all affected systems needs to be considered.

Finally, an important open point remains the option for over-the-top security for the sensors deployed in the port. The idea is to switch off user-plane security in favour of performance and implement a direct encrypted link between the devices and the respective application servers. This option requires a careful examination of the performance gains – if any. In case such a setup is favourable the device manufacturer should equip the devices with the necessary crypto hardware/software without compromising performance.

## **4.2 On the suitability of security trust zones**

Network slicing is one of the most relevant characteristics in 5G networks. Supported by the virtualisation of resources, virtual resources can be dynamically allocated to optimise the service provided within a network slice. For example, a 5G infrastructure deployed to support seaport activities can set up different networks slices for different operations with different requirements. A network slice can support the traffic control operations, which have high availability and resilience requirements, while another network slice can support virtual reality for training operations, which have high bandwidth and low latency requirements. Different requirements mean different resources to deploy in different parts of the 5G infrastructure. Virtualising them allows to tailor which components to deploy depending on the service provided by the network slice: not all components are required in a network slice and a different resource capacity is required for every component.

All this results in many possible and different network slices, which from a security perspective, entails challenging situations as different parts of the same infrastructure can be exposed to different types of security threats. Additionally, every network slice can have different security requirements, which might depend on the criticality of the devices operating within the network slice and on the type of service offered by it. In the security domain, the component and resources devoted to security protection are different depending on the type of network slice. For instance, continuing with the Smart Seaport example, a network slice offering measurements on the quality of the air using sensors deployed all around the seaport need to be protected against certain critical threats within that domain (i.e., unauthorised tampering or data manipulation), while for network slices that offer virtual reality operations the critical threats are different (i.e., DoS attacks).

To solve the special characteristics of security requirements for network slices, the 5G-MoNArch project has developed the concept of security trust zones (STZ)s. The STZ concept was described in detail in the deliverable D3.1 of 5G-MoNArch [5GM-D3.1] and is revisited in this document with focus on its suitability in 5G networks. As described in [5GM-D3.1], an STZ is a subset of elements (devices, networks, etc.) with common security requirements, where specific components are deployed for their

security protection (detectors, security probes, etc.). This approach brings clear advantages derived from the inherent characteristics of network slices in 5G infrastructures:

- Adaptation of security resources to the characteristics of the infrastructure to protect: in this case, adaptation to the characteristics of the network slice, and more specifically to the subset of elements that share the same security requirements within a network slice. This entails the deployment of detection, prevention or reaction components and different security probes capable of retrieving monitoring information associated with the protection against the required security threats.
- Dynamic allocation of resources devoted to security protection. The different criticality of certain STZs with respect to others entails changes on the computational resources required for the instances deployed for protecting security. These instances are independent for every STZ, where the computational resources of every instance can be tailored and optimised dynamically at any time.
- Efficient management. Same as in 5G infrastructure, using virtualised resources for deploying security protection components allows to dynamically fine-tune their computational resources (CPU, memory, network configurations), which makes configuration and management easier than in physical deployments, where physical machines are deployed when new resources are needed.
- In general, addressing problems in small and homogeneous sets is easier than addressing them in big and heterogeneous ones. This is also the case in STZs. STZs are chosen by grouping elements of a network slice with common requirements and characteristics, e.g., temperature sensors of a network slice in charge of producing climate measurements. The complexity of the security protection infrastructure of such STZ is simpler than the one required to protect a complete 5G infrastructure as a whole. This would allow to deploy just the components required for such STZ, i.e., deploying only detection capabilities but not prevention capabilities in case it is not required for such STZs.
- Preventing propagation of incidents. Early detection of incidents within an STZ allows to prevent its propagation to other STZs or network slices. Threat intelligence exchange mechanisms allow to exchange information about detected threats in different STZs, which can be used to deploy prevention activities.
- The usage of STZs allows to simplify even more the security protection process by using STZs templates. STZs templates are pre-configured containers with different instances of security capabilities (detectors, security probes, etc). The number and type depend on the STZ template. Different “flavours” can be available in advanced with minor configuration required when using it. The selection of one flavour over another depends on the security requirements of the infrastructure to protect.

### 4.2.1 Suitability analysis

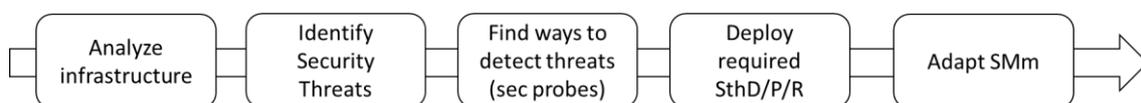
This section summarises the suitability of the STZ approach, comparing it with the most paramount features of 5G infrastructures. The analysis considers the main features of 5G infrastructures, evaluating their main security implications and how the STZs approach helps to address its protection from a security perspective. The results of this analysis are summarised in Table 4-2, which compares three of the main features of 5G infrastructures: resource virtualisation, adaptation to infrastructure requirements and diversity of devices. Although it is evident that these features bring noticeable benefits, they also include additional security implications, such as: security risks associated to hypervisors when using virtualised resources, changing security threats in different parts of the infrastructure due to very different services and devices sharing the same 5G network. STZs contribute to deal with these emerging security concerns by providing tailored security protection mechanisms or prevention of incidents propagation among other benefits.

**Table 4-2: Analysis of STZs vs 5G infrastructures**

5G feature	Security implication	STZ approach solution
Virtualisation of resources	Hypervisor risks (Infected hypervisor affects virtual resources, incorrect hypervisor configuration, side channel vulnerabilities)	STZs can deploy specific security probes to retrieve security information directly from hypervisors, which allows to monitor at with different levels of granularity
	Different trust level in different virtual resources	The level of security can be easily adapted in every STZ by using specific configurations (for detectors, security probes) in order to protect every different type of virtual resource
Adaption of NS to requirements of the infrastructure (availability, resilience, etc.)	Unforeseen security threats	STZs can be dynamically adapted by deploying new security capabilities to deal with new threats. In case of using STZ templates a new template can be used to increase the level protection by replacing the previous one with one with better protection capabilities
	New unknown vulnerabilities	
	Changes in the NS (devices, network layout, performance, resilience, etc.)	
Myriad of new and different devices and services over the same 5G infrastructure	Different threats for different devices, networks and services	STZs can be dynamically adapted by deploying new security capabilities to deal with new threats. In case of using STZ templates a new template can be used to increase the level protection by replacing the previous one with one with better protection capabilities
	Risk of propagation of incidents between different parts of the infrastructure	Early detection of incidents within a STZ and notification of the detected incidents to other STZs allows to control propagation of incidents

#### 4.2.2 Process for defining STZs within a 5G infrastructure

This section defines the process to follow when choosing and setting up STZs within a 5G infrastructure. The process is depicted in Figure 4-2. It is structured in five steps and it is based on the STZ components described in D3.1.

**Figure 4-2: Process for defining STZs within a 5G infrastructure**

##### **Analyse the infrastructure to protect**

This step constitutes the identification of infrastructures *niches*, which are elements with similar requirements and logical proximity (i.e., same subsystem, same subnetwork) within the same network slice. Every identified niche becomes an STZ. For example, on the sea port infrastructure considered in WP5 there are three different networks slices:

- A network slice for managing the signalling system.
- A network slice for virtual reality services.

- A network slice managing environmental sensors installed on ships

The network slices for the signalling system and the virtual reality service can be considered as two different STZs. Both network slices provide different services, with different requirements and are exposed to different security threats. The network slice, for managing environmental sensors installed on ships, consists of three barges with sensors installed on board. In this case, considering that the sensors installed on every barge are physically separate, we can consider three different STZs, one per barge.

### ***Identify security threats per STZ***

Once we know the assets to protect within each of the identified STZs, the next step is to evaluate the main security threats that they are exposed to. To this end, the characteristics of the STZ determine the main threats to consider when choosing the security components required. It should be considered that a complete protection is impossible. This analysis provides the minimum level of protection based on the identified security threats. Several factors are considered in this analysis; these include:

- Evaluation of the exposition to threats per STZ: assets exposed to the public internet are easily exposed to DoS attacks or those exposed to the public spectrum are easily exposed to sniffing of jamming attacks
- Criticality of the STZ: the importance of the service provided in the STZ depends on the criticality of the operations supported by the assets included in the STZ.
- STZ complexity: the higher the number of different devices or the higher the size of the STZ, the more difficult it is to manage security, which impact on the security capabilities deployed and on their configuration.

The usage of threat models can help to identify these threats, which would also allow to establish a threat priority, with additional information such as likelihood of receiving attacks associated to those events.

### ***Find ways to detect threats***

Among the identified critical threats, a set of security probes are required to monitor the elements of an STZ. For example, protection against DoS attacks requires the deployment of NIDS probes such as Suricata<sup>6</sup>, or to be protected against jamming attacks requires dedicated anti-jamming detectors. Sometimes HIDS probes are required to monitor machines from inside (for example, OSSEC<sup>7</sup> to report about CPU load, number of active connections or file integrity). Depending on the case, custom security probes might be required in order to report specific events. This case would require special adaptation of detectors to interpret the information included in those specific events.

### ***Deploy required security capabilities***

This activity is based on the deployment of the security capabilities required in the STZs identified. As defined in D3.1, STZs can provide detection, reaction and prevention capabilities through Security Threat Detector (SthD), Reaction (SthR) and Prevention (SthP). Not all STZs need all possible capabilities. Again, the deployment or not of those capabilities depends on the type of STZ. However, the deployment of SthD is mandatory for all STZs, as the incident monitoring is an essential activity to be aware of what is happening in the infrastructure and which support the other two activities:

- Detection: supports events normalisation and correlation for detecting incidents.
- Reaction: uses incidents detected to propose actions to mitigate them.
- Prevention: uses incidents detected in other STZs and mitigations enforced there to carry out actions to prevent those incidents.

It is also worth noticing that, depending on the security probes deployed in the STZ, it will also be necessary to properly configure the SthD to process the events sent from those probes. This includes knowing the information included in those events, extracting the relevant ones and normalising them in a common format understandable by the Security Monitoring manager (SMm).

---

<sup>6</sup> <https://suricata-ids.org/>

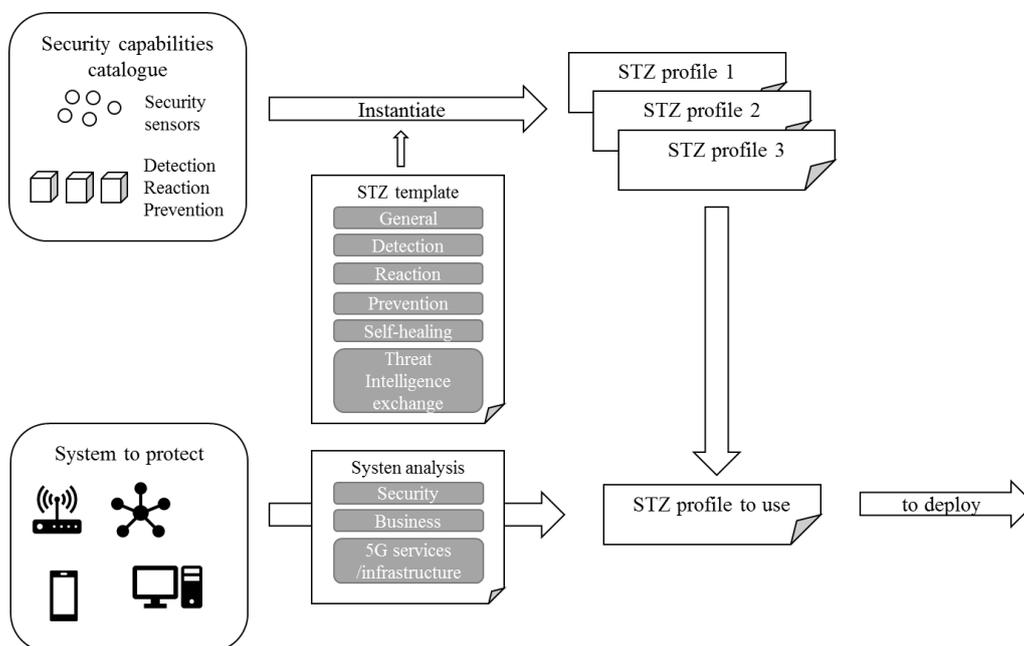
<sup>7</sup> <https://www.ossec.net/>

### Adapt Security Monitoring Manager

As defined in [5GM-D3.1], SMm manages the security capabilities deployed in different STZs within the same network slice. The final step in the process of setting up STZs is to properly configure the SMm to process information received from the STZs it manages. The main functions of the SMm are to correlate events received from the SthDs, generate security alerts when certain patterns are identified in the received events. Therefore, SMm needs to activate/create specific rules to correlate the type of events received from the SthDs and generate the corresponding security alerts. The number and type of rules would depend on the security probes deployed in the different STZs, which results in custom configuration of SMms for each Network Slice.

#### 4.2.3 Templates based deployment of STZs

During the process of designing the security capabilities required for an STZ it is possible either to manually personalise them or to use STZs templates. STZ templates describe a set of predefined configurations for an STZ, which include a set of security capabilities and security probes. STZ templates are instantiated in any of the available STZ profiles, which contain components and probes for the configurations described in the corresponding templates. When using STZs templates an additional evaluation of the STZ is required, as it is required to choose the most convenient STZ profile from the ones available. Figure 4-3 represents the process. A catalogue of security capabilities includes a set of available security probes and capabilities (detection, reaction and prevention). A subset of these security probes and capabilities are used to define different flavours of STZs. There can be configurations with protection against simple threats, while more sophisticated configurations can include prevention capabilities. The analysis of the STZ, in terms of security required or criticality of the services provided, will determine the level of protection required and the STZ template to choose. The chosen template is used to select the STZ profile that better adapts to the requirements of the devices to protect, which is deployed in the corresponding network slice.



**Figure 4-3: Process for selecting STZs based on templates**

While this approach provides with less flexibility than the direct adaptation of the security capabilities to the STZ, it simplifies management activities, as minor configuration in the template-based approach. A hybrid approach is also possible, which is based on the usage of templates as the initial deployment of security components, while further fine-tuning can be done by including new security probes or more complex capabilities.

#### 4.2.4 Changing security requirements of a STZ

In the security domain there is no complete protection. Changing context (e.g., due to additional devices deployed or changes in the layout of the network) and emerging new threats requires dynamic adaptation of security protection mechanisms. The STZ approach in 5G infrastructures needs also to face with this problem as it is possible that they are exposed to new threats not considered before. To address this issue there are two possibilities:

- Select a new template from the templates catalogue with a higher security protection level.
- Fine-tune the current security components of an STZ to deal with new threats by identifying probes to detect such threats, updating also the SMm of the network slice with new rules capable of correlating the new events to generate alerts associated to them.

In general, the major impact when updating the security protection level of an STZ occurs at the SthD. SthDs are mostly in charge of extracting the information from the myriad of heterogeneous event formats received from many different security probes, normalising them to a common format for their processing at the SMm. When new security probes are added, it is required either to update the SthD or to replace it with a new instance capable of processing them. Depending on the concrete deployment approach, there are several options to update it:

- Virtualisation of SthD, which would allow a seamless withdrawal and deployment of different instances of SthDs.
- Usage of a plugin-based SthD, allowing for adaptation to events from new security probes by simply creating new plugins capable of processing them.

In general, the major impact when updating the security protection level of an STZ occurs at the SthD. SthD are mostly in charge of extracting the information from the myriad of heterogeneous events format received from many different security probes, normalising them to a common format for its processing at the SMm. When new security probes are added it is required either to update the SthD or to replace it with a new instance capable of processing them. Depending on the concrete deployment approach, there are several options to update it:

- Virtualisation of SthD, which would allow a seamless withdrawal and deployment of different instances of SthDs
- Usage of plugin based SthD, allowing for adaptation to events from new security probes by simply creating new plugins capable of processing them.

### 4.3 Simulated threats and corresponding detectors

The STZ-based approach described in [5GM-D3.1] allows to design protection mechanisms for the prevention, detection and mitigation of the security threats introduced in Section 4.1 and analysed and extended in Section 4.2. STZ are introduced in [5GM-D3.1] as a solution to manage the protection of groups of assets within a Network Slice that share common security requirements, functionalities or which are physically or logically close. In STZs a set of security probes are deployed to monitor the infrastructure, gathering events and logs from the network or from the devices. Several capabilities can be deployed to detect, react or prevent security incidents. In this section we firstly introduce a testbed that has been deployed to validate the STZ approach, deploying several STZs, detectors and probes to simulate attacks and to check the monitoring and detection capabilities. In Section 4.3.2 details are given about one of the possible probes to be included in a STZ, focused on the detection of network behavioural anomalies.

#### 4.3.1 Security simulation campaign for monitoring 5G network slices

Network slicing entails several challenges when preserving security and protecting 5G networks from security incidents. To this end, it is paramount to consider the resources required to protect network slices with very specific requirements, either in terms of security or resilience. The STZ approach described in [5GM-D3.1] allows to dynamically adapt security protection capabilities to the special characteristics of a network slice. Figure 4-4 represents the main elements involved in this approach. It is worth noticing that more than one STZ might be part of one Network Slice. For instance, using the

Hamburg Sea Port example, an STZ can be set-up to manage the security protection of the ship sensors deployed to measure pollution while another one can be used for the protection of smart traffic lights. Different STZs might protect different parts of the network with different security requirements, and therefore, different security protection capabilities might be deployed in each STZ. The security testbed deployed in 5G-MoNArch illustrates the flexibility of the solution envisioned to protect 5G infrastructure against security threats, allowing for the adaptation of the resources devoted to the security protection.

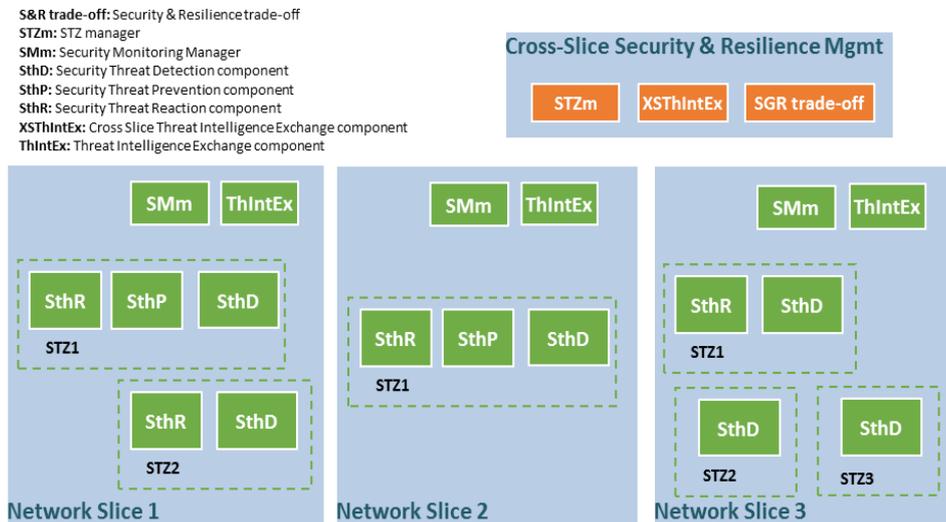


Figure 4-4: Security Trust Zone approach for protecting 5G network slices

The security testbed built in 5G-MoNArch consists of one STZ manager (STZm), that manages the security requirements of every network slice, and one SMm, that handles the security requirements of every STZ. Details about these elements can be found in [5GM-D3.1]. Figure 4-5 depicts the detailed structure of an STZ. For every STZ, a set of resources is considered to be protected. These resources can be: i) sensors deployed in the pollution measurements infrastructure, ii) traffic lights, iii) the network that interconnects them, iv) user agents, v) network components such as: servers, firewalls or routers. Several security probes are deployed around these assets to protect resources. These probes monitor any activity in or around these assets and report detected anomalies.

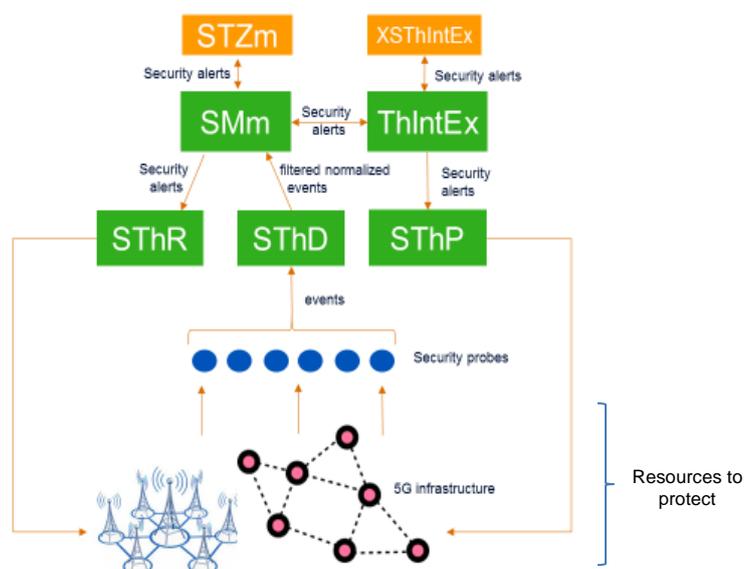


Figure 4-5: Detailed STZ and data flows

Security probes within the STZ report the monitored events to the security threat detector of the STZ, which filters and normalises them in order to be interpreted, processed and correlated by the SMm. The SMm, based on several security rules, reports detected anomalies both to the STZm (i.e., for its presentation to the system admin) and to the Threat Intelligence Exchange (ThIntEx). The reported anomalies are then exported to the rest of the 5G infrastructure (i.e., in order to prevent the propagation of detected incidents).

The simulation testbed created in 5G-MoNArch considers 6 different security probes, which cover the security threats identified in Section 4.1. Table 4-3 summarises the security probes, the assets to protect and the specific threat detected. Details about the user and entity behaviour analytics (UEBA) probe are given in Section 4.3.2.

**Table 4-3: Security Probes integrated in the 5G-MoNArch security testbed**

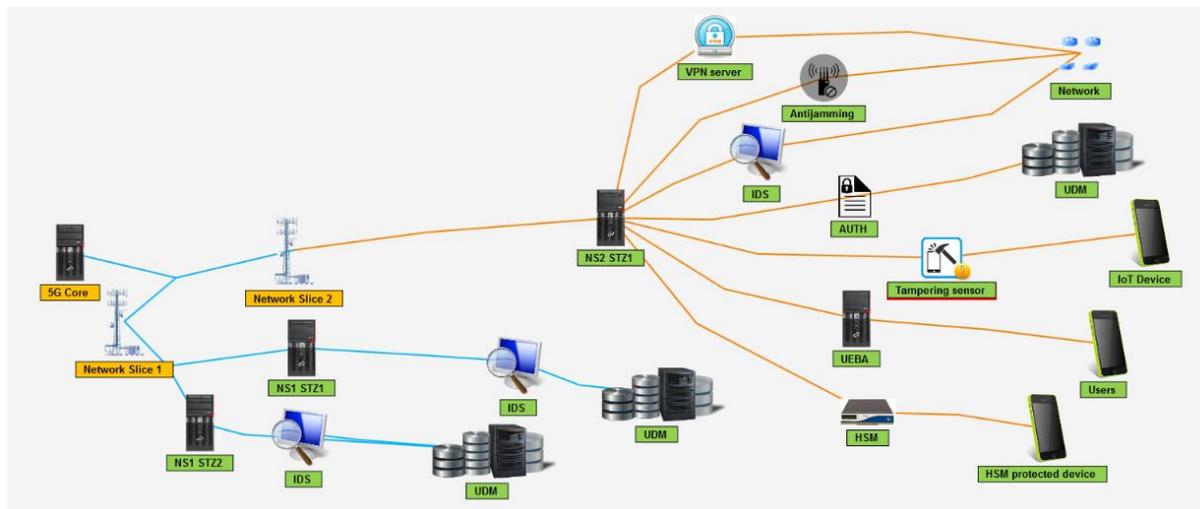
Security Probe	Asset monitored	Possible attack/detections
HSM	UE with HSM	Brute-force attack against HSM protected device Man in the middle attack: modification of message integrity Detection of unsecure connections through HSM
Antijamming	Wireless network	Pulsed Based jamming attack Wide Band jamming attack Wave Form jamming attack LFM Chirp jamming attack
VPN Server Logging	VPN server and connections	Detection of weak encryption in VPN connections DoS attack against VPN server: connection requests flooding Brute force attack against VPN server Settings manipulation attack: authorised change of configuration for VPN connections
Tampering Sensor	Physical devices	Detection of unauthorised physical manipulation of devices
User and Entity Behaviour Analytics/Network Behavioural Analysis (UEBA/NBA)	UE	Detection of anomalies in the behaviour of UE when using several services: SMS, Voice Calls, VR, etc.
IDS	Network UDM	Denial of Service attack against assets (i.e., UDM) Brute-force attack against assets (i.e., UDM, SthD, etc.) Malicious scanning of services (i.e., Port scanning)

Different security probes can be deployed in every STZ, depending on its security requirements. For instance, a traffic-lights network slice would require anti-tampering sensors in order to warn against physical manipulation of the device, though having HSM might be less relevant in this context. The adaptation of the security capabilities available in an STZ, with respect to the characteristics of the infrastructure to protect, is twofold:

- Detection, prevention and reaction capabilities. While the detection capability (carried out by the SthD) is mandatory for all STZs, the prevention and reaction capabilities are optional and depend on the capabilities required for an STZ.
- Available security probes. Depending on the domain where to instantiate the STZ, the security probes to deploy would be different and the SthD would need to be adapted, such that they are capable of normalising the information received from these security probes. The testbed allows to easily configure the available security probes, as well as activating or deactivating them, depending on the ones required.

Figure 4-6 represents the complete testbed deployed for the simulation of the security infrastructure designed in 5G-MoNArch. This includes two network slices and three STZs:

- Network Slice 1. It includes two STZs:
  - NS1-STZ1. It contains an IDS probe that protects the UDM instance that manages User Agents operating in STZ1.
  - NS1-STZ2. It contains an IDS probe that protects the UDM instance that is operating in STZ2.
- Network Slice 2. It includes one STZ:
  - NS2-STZ1. It contains all the possible security probes as shown in Table 4-3, in order to protect the network, UDM or UEs from potential threats.



*Figure 4-6: Complete 5G-MoNArch security simulation testbed*

#### 4.3.1.1 Simulation of attacks against an STZ

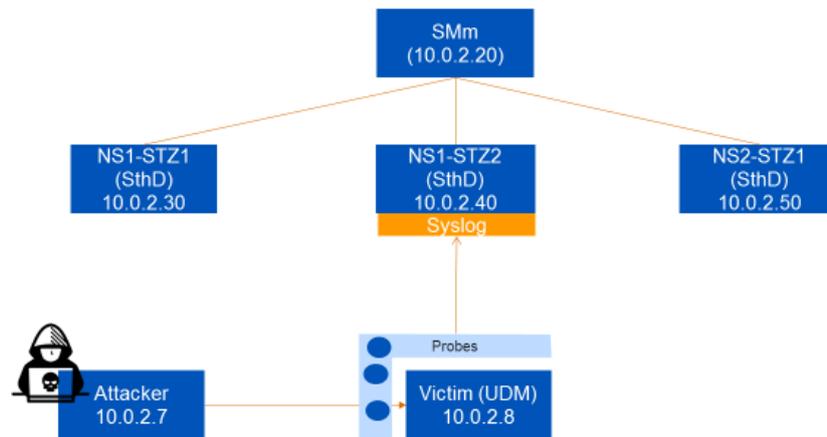
In order to clearly show the capabilities of the security infrastructure, a set of simulated attacks have been created. Figure 4-7 represents the deployment done for the simulation of attacks. A set of different virtual machines have been deployed over the same subnet. In this deployment the SMm is running with the IP 10.0.2.20, while the SthD are deployed using the IPs 10.0.2.30, 10.0.2.40 and 10.0.2.50. Another virtual machine is representing the UDM running on 10.0.2.8 while a Kali linux is acting as attacker using the IP 10.0.2.7.

In the framework of the conducted simulation, the attacker will launch or simulate attacks. These attacks, simulated or not, will allow to check the suitability of the STZ approach and the customisation of the security capabilities per STZ. Considering the list of probes described in Table 4-3 an IDS (based on a Suricata<sup>8</sup>) and the UEBA (provided by CERTH) have been deployed in the testbed. Additionally, the rest of sensors described in Table 4-3 has been simulated. In all cases, the events created by the security probes are reported to the SthDs using syslog<sup>9</sup>. The SthD filters and normalise events coming from different sensors and reports them to the SMm to be correlated.

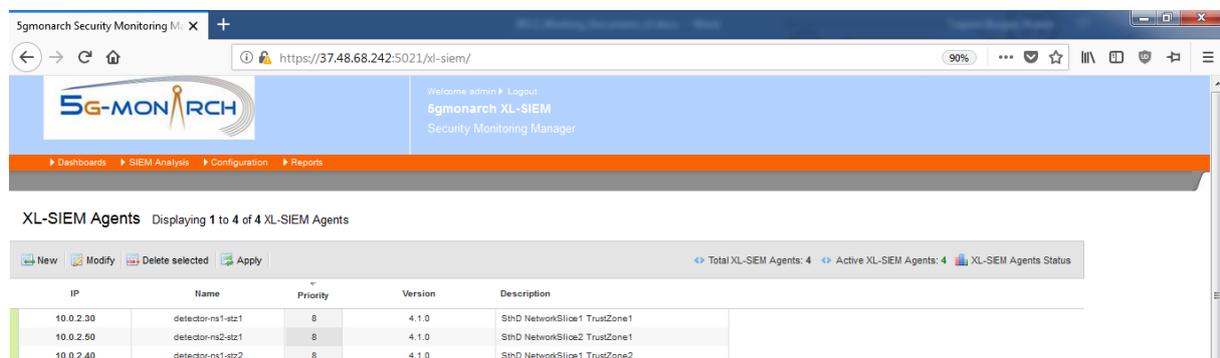
The testbed has been created based on the Atos XL-SIEM. The XL-SIEM is an incident detection tool that correlates security events and detects anomalies, infers incidents and triggers alerts. The XL-SIEM has been adapted to work as an SMm and STZm, modified to support the STZ approach. Figure 4-8 shows a screenshot of the three SthDs configured at the STZm in the deployment shown in Figure 4-7.

<sup>8</sup> <https://suricata-ids.org/>

<sup>9</sup> <https://tools.ietf.org/html/rfc5424>



**Figure 4-7: Testbed deployment for simulating attacks**



**Figure 4-8: SthD configured at the STZm (using the Atos XL-SIEM GUI)**

Considering the security probes described in Table 4-3, a set of different attacks has been simulated. A Kali Linux distribution has been used to simulate DoS, brute-force and network scanning attacks. These attacks are detected by the IDS probe deployed in the infrastructure:

- DoS attacks. The utility hping3 allows to flood a target with ICMP packets, provoking a DoS attack. Figure 4-9 shows the result of executing the DoS attack using hping3. The screenshot represents the flow of packages sent to the targeted host.
- Network Scanning. The utility nmap is used to scan the network or a specific target to discover open ports and running services in open ports. Figure 4-10 depicts a screenshot from the result of the network scanning attack. The result shows the services discovered: SSH server and a Web server running in port 80 and 443.
- Brute-force attack. The utility ncrack allows perform consecutive attempts to establish an SSH connection with a specified target by using passwords from a list or file. Figure 4-11 shows the result of the brute force attack and the command used to trigger such simulation.

In addition to the Kali Linux tool that allows to simulate real attacks, a set of attacks are simulated by generating events that are reported to the syslog server of the SthD. To this end, a script has been created in order to trigger the desired attack. Figure 4-12 represents the possible attacks that can be simulated with such a script.

```

root@kali:~/home/user# hping3 -S -p 23 -V 10.0.2.8
using eth0, addr: 10.0.2.7, MTU: 1500
PING 10.0.2.8 (eth0 10.0.2.8): S set, 40 headers + 0 data bytes
len=46 ip=10.0.2.8 ttl=64 DF id=11570 tos=0 iplen=40
sport=23 flags=RA seq=0 win=0 rtt=7.8 ms
seq=0 ack=1666035615 sum=807f urp=0

len=46 ip=10.0.2.8 ttl=64 DF id=11623 tos=0 iplen=40
sport=23 flags=RA seq=1 win=0 rtt=5.0 ms
seq=0 ack=809945302 sum=3790 urp=0

len=46 ip=10.0.2.8 ttl=64 DF id=11728 tos=0 iplen=40
sport=23 flags=RA seq=2 win=0 rtt=9.6 ms
seq=0 ack=81955085 sum=7c04 urp=0

len=46 ip=10.0.2.8 ttl=64 DF id=11928 tos=0 iplen=40
sport=23 flags=RA seq=3 win=0 rtt=1.6 ms
seq=0 ack=181275101 sum=b77d urp=0

len=46 ip=10.0.2.8 ttl=64 DF id=12057 tos=0 iplen=40
sport=23 flags=RA seq=4 win=0 rtt=1.4 ms
seq=0 ack=1510009674 sum=1e4d urp=0

len=46 ip=10.0.2.8 ttl=64 DF id=12201 tos=0 iplen=40
sport=23 flags=RA seq=5 win=0 rtt=10.0 ms
seq=0 ack=1337874765 sum=5ce9 urp=0

len=46 ip=10.0.2.8 ttl=64 DF id=12273 tos=0 iplen=40
sport=23 flags=RA seq=6 win=0 rtt=5.0 ms
seq=0 ack=24143783 sum=5029 urp=0

len=46 ip=10.0.2.8 ttl=64 DF id=12461 tos=0 iplen=40
sport=23 flags=RA seq=7 win=0 rtt=10.0 ms
seq=0 ack=692879153 sum=e6e9 urp=0

```

Figure 4-9: DoS attack simulated with hping3 in Kali Linux

```

root@kali:~/home/user# nmap -sS -sV -PO 10.0.2.20
Starting Nmap 7.70 ( https://nmap.org ) at 2018-10-01 11:45 EDT
Nmap scan report for 10.0.2.20
Host is up (0.00014s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh      OpenSSH 7.4p1 Debian 10+deb9u3 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.25 ((Debian))
443/tcp   open  ssl/http Apache httpd 2.4.25 ((Debian))
MAC Address: 08:00:27:B2:7D:D0 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:Linux:Linux kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.04 seconds

```

Figure 4-10: Network scanning attack simulated using Nmap in Kali Linux

```

root@kali:~/home/user# ncrack -p 22 --user root -P 500-worst-passwords.txt 10.0.2.50
Starting Ncrack 0.6 ( http://ncrack.org ) at 2018-10-01 11:47 EDT
Stats: 0:00:06 elapsed; 0 services completed (1 total)
Rate: 0.00; Found: 0; About 0.40% done
Stats: 0:00:07 elapsed; 0 services completed (1 total)
Rate: 0.00; Found: 0; About 0.60% done
Stats: 0:00:10 elapsed; 0 services completed (1 total)
Rate: 0.00; Found: 0; About 0.80% done
Stats: 0:00:12 elapsed; 0 services completed (1 total)
Rate: 0.00; Found: 0; About 1.00% done; ETC: 12:07 (0:19:48 remaining)
Stats: 0:00:19 elapsed; 0 services completed (1 total)
Rate: 0.07; Found: 0; About 9.00% done; ETC: 11:51 (0:03:12 remaining)
Stats: 0:00:20 elapsed; 0 services completed (1 total)
Rate: 0.07; Found: 0; About 9.00% done; ETC: 11:51 (0:03:22 remaining)

```

Figure 4-11: Brute-force attack simulated using ncrack in Kali Linux

```

user@kali:~$ java -jar attacker_5G.jar
Please, specify of attacks as arguments from the following list:

USER AND ENTITY BEHAVIOR ANALYTICS ATTACKS

ueba_sms      - Trigger anomalous behavior of devices when using SMSs
ueba_call    - Trigger anomalous behavior of devices when using Voice Calls
ueba_vr      - Trigger anomalous behavior of devices when using Virtual Reality services
ueba_service - Trigger anomalous behavior of devices when using other services

HSM BASED ATTACKS

hsm_bruteforce - Trigger bruteforce attacks against HSM device
hsm_mim       - Trigger man in the middle attack modifying message integrity with HSM device
hsm_unsecure  - Trigger establishment of an unsecure connection with HSM

JAMMING ATTACKS

jamming_pulse - Trigger Pulsed jamming attack at a pre-defined frequency
jamming_wide  - Trigger Wide Band jamming attack at a pre-defined frequency
jamming_wave  - Trigger wave form jamming attack at a pre-defined frequency
jamming_lfm   - Trigger LFM chirp jamming attack at a pre-defined frequency

VPN ATTACKS

vpn_encryption - Trigger a VPN connection with a weak encryption
vpn_dos       - Trigger a DOS attack against the VPN server by establishing too many connections
vpn_bruteforce - Trigger a Bruteforce attack against the VPN server
vpn_manipulation - Trigger a settings manipulation attack with an incorrect configuration for the VPN connection

TAMPERING ATTACKS

device_tampering - Trigger the unauthorized physical manipulation of devices

OTHER
list             - Shows these options
ALL DONE

```

Figure 4-12: Script to simulate several attacks

### 4.3.1.2 Detection of attacks at SMm

The following section shows the reception of security events from the security probes deployed at the testbed and the generation of alerts based on their correlation and on the anomalies detected.

#### Denial of Service Attack

A Denial of Service (DoS) attack was triggered by using the hping3 tool. The Intrusion Detection System (IDS) security probe detected anomalous flooding packages. The IDS detects fake flooding packages coming from fake source IP addresses. The SthD labels that traffic as Spamhaus traffic, in order to highlight the fake source IP. The GUI of the SMm allows to visualise the events directly received from the SthD (Figure 4-13). The SMm correlates the received Spamhaus events and generates the corresponding alerts warning about an ongoing DoS attack (Figure 4-16).

Date	Event Name	Risk	Generator	Sensor	Source IP	Dest IP
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	27.146.48.187:50029	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	155.71.220.126:58783	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	147.119.4.87:58938	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	27.146.176.88:58950	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	155.11.155.251:59065	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	153.53.250.19:61433	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	147.16.87.231:62050	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	128.188.106.27:62120	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	152.109.62.235:63672	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	170.67.115.84:64440	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	207.110.120.91:441	UDM.23
2018-08-30 11:04:53	snort-ET DROP Spamhaus DROP Listed Traffic Inbound	0	snort	detector-ns1-stz1	196.193.170.84:1791	UDM.23

Figure 4-13: Denial of Service events received by the SMm

### Network scanning attack

A network scanning attack was triggered by using the nmap tool. The IDS security probe detected port scanning activities, followed by sending them to SthD and labelling the traffic as Nmap user agent. The GUI of the SMm allows to visualise the events directly received from the SthD (Figure 4-14). Same as in DoS attacks, these events are correlated by the SMm, thereby generating the alerts represented in Figure 4-16.

Date	Event Name	Risk	Generator	Sensor	Source IP	Dest IP
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37406	Host-10-0-2-2-8080
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37416	Host-10-0-2-2-8080
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:60784	10.0.2.20.80
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:60796	10.0.2.20.80
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37956	Host-10-0-2-2-9000
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37934	Host-10-0-2-2-9000
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37440	Host-10-0-2-2-8080
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:60806	10.0.2.20.80
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:60816	10.0.2.20.80
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37462	Host-10-0-2-2-8080
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37974	Host-10-0-2-2-9000
2018-08-30 13:56:56	ET_SCAN Possible Nmap User-Agent Observed	4	snort	detector-ns1-stz1	10.0.2.7:37968	Host-10-0-2-2-9000
2018-08-30 13:54:38	snort-SURicata HTTP missing Host header	0	snort	detector-ns1-stz1	10.0.2.7:49704	Host-10-0-2-2-12345
2018-08-30 13:54:24	SSHd_Did not receive identification string	1	sshd	detector-ns1-stz1	Host-10-0-2-2	10.0.2.30.22
2018-08-30 13:54:24	SSHd_Did not receive identification string	1	sshd	detector-ns1-stz1	10.0.2.7	10.0.2.30.22

Figure 4-14: Network scan events received by the SMm

### Brute-force attacks

A brute-force attack was triggered by using the ncrack tool. The IDS security probe detected port scanning activities, sending them to SthD and reporting failed passwords events. The GUI of the SMm allows to visualise the events directly received from the SthD (Figure 4-15). Similar to the previous attacks, these events are correlated by the SMm, generating the alerts represented in Figure 4-16.

Date	Event Name	Risk	Generator	Sensor	Source IP	Dest IP
2018-08-30 14:12:28	SShd_Failed_password	1	sshd	detector-n5-stz1	10.0.2.7:33012	10.0.2.30:22
2018-08-30 14:12:28	SShd_Connection_closed	0	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:28	SShd_PAM_X_more_authentication_failures	1	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:28	SShd_Maximum_authentication_attempts_exceeded	2	sshd	detector-n5-stz1	N/A	10.0.2.30:22
2018-08-30 14:12:28	SShd_Failed_password	1	sshd	detector-n5-stz1	10.0.2.7:33030	10.0.2.30:22
2018-08-30 14:12:28	SShd_Connection_closed	0	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:25	SShd_Failed_password	1	sshd	detector-n5-stz1	10.0.2.7:33014	10.0.2.30:22
2018-08-30 14:12:25	SShd_Connection_closed	0	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:25	SShd_PAM_X_more_authentication_failures	1	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:25	SShd_Failed_password	1	sshd	detector-n5-stz1	10.0.2.7:33028	10.0.2.30:22
2018-08-30 14:12:25	SShd_Connection_closed	0	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:25	SShd_PAM_X_more_authentication_failures	1	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:25	SShd_Failed_password	1	sshd	detector-n5-stz1	10.0.2.7:33002	10.0.2.30:22
2018-08-30 14:12:25	SShd_Connection_closed	0	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22
2018-08-30 14:12:25	SShd_PAM_X_more_authentication_failures	1	sshd	detector-n5-stz1	10.0.2.7	10.0.2.30:22

Figure 4-15: Brute-force attack events received by the SMm

Signature	Events	Risk	Duration	Source	Destination	Status
Attacks, Bruteforce attempt against host	6	0	8 secs	10.0.2.7:33000	10.0.2.30:ssh	open
Attacks, DoS attempt from Spamhaus traffic	3	8	0 secs	206.143.216.98:world-im	10.0.2.20:telnet	open
Attacks, DoS attempt from Spamhaus traffic	3	8	0 secs	155.66.251.213:3195	UDM:telnet	open
Network scan, Nmap scan against 10.0.2.20	2	4	0 secs	10.0.2.7:80784	10.0.2.20:http	open
Network scan, Nmap scan against 10.0.2.2	3	4	0 secs	10.0.2.7:37416	Host-10-0-2-2:http-proxy	open
Network scan, Nmap scan against 10.0.2.2	2	4	0 secs	10.0.2.7:37408	Host-10-0-2-2:http-proxy	open
Network scan, Nmap scan against 10.0.2.20	3	4	0 secs	10.0.2.7:80158	10.0.2.20:http	open
Network scan, Nmap scan against 10.0.2.20	2	4	0 secs	10.0.2.7:80154	10.0.2.20:http	open

Figure 4-16: Alerts for attacks created with Kali Linux tools (DoS, Network Scanning and Brute-force)

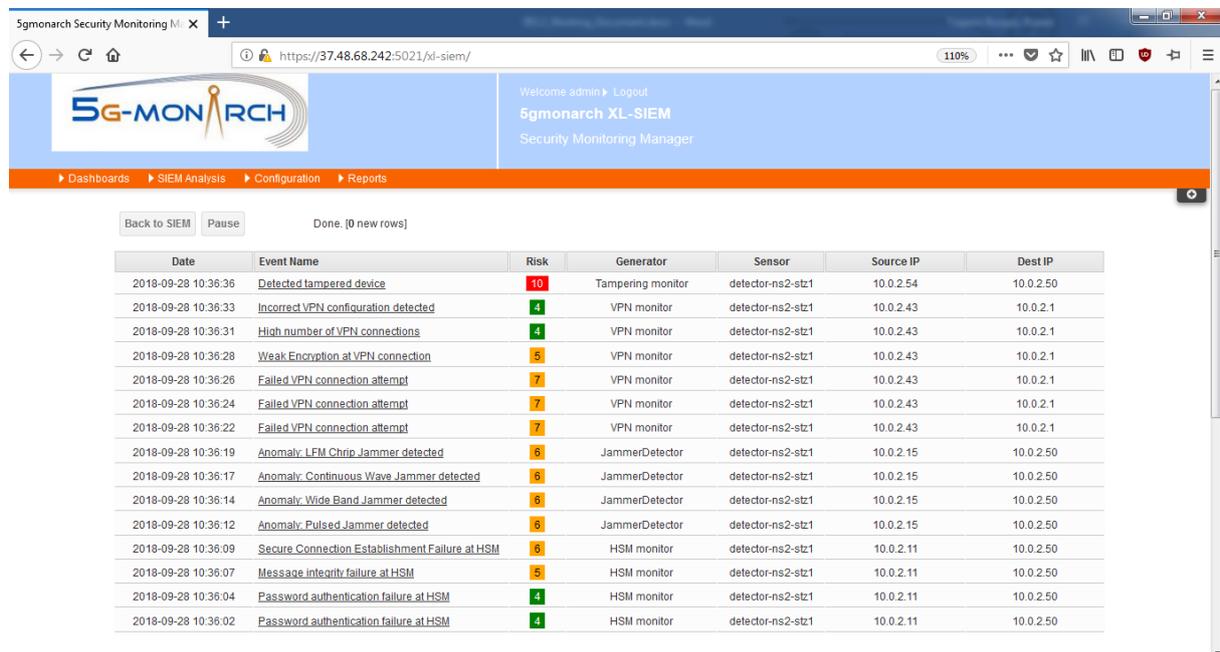
**Incidents detected by simulated sensors**

Several incidents were simulated using the script shown in Figure 4-12. This script sends events that represents incidents. These events are received by the SthD, normalised and processed by the SMm. Figure 4-17 represents a sample of every event received by the SMm. Figure 4-18 represents a list of alerts generated after correlating events received from the simulated sensors described in Figure 4-12.

As can be seen, this STZ based approach provides a flexible way to protect partial subsets of elements in a network slice of a 5G network. The security capabilities of an STZ can be easily adapted by using STZ templates, which stem from several STZ profiles. The simulated testbed has proven that STZ profiling is easy to manage, as it is just required to deploy a SthD, which is a quite light component from a computational point of view.

Depending on the STZ profile needed, several security probes are required, which are easily managed thanks to the plugin-based architecture of the SthD. The deployed SthD allows to easily activate the

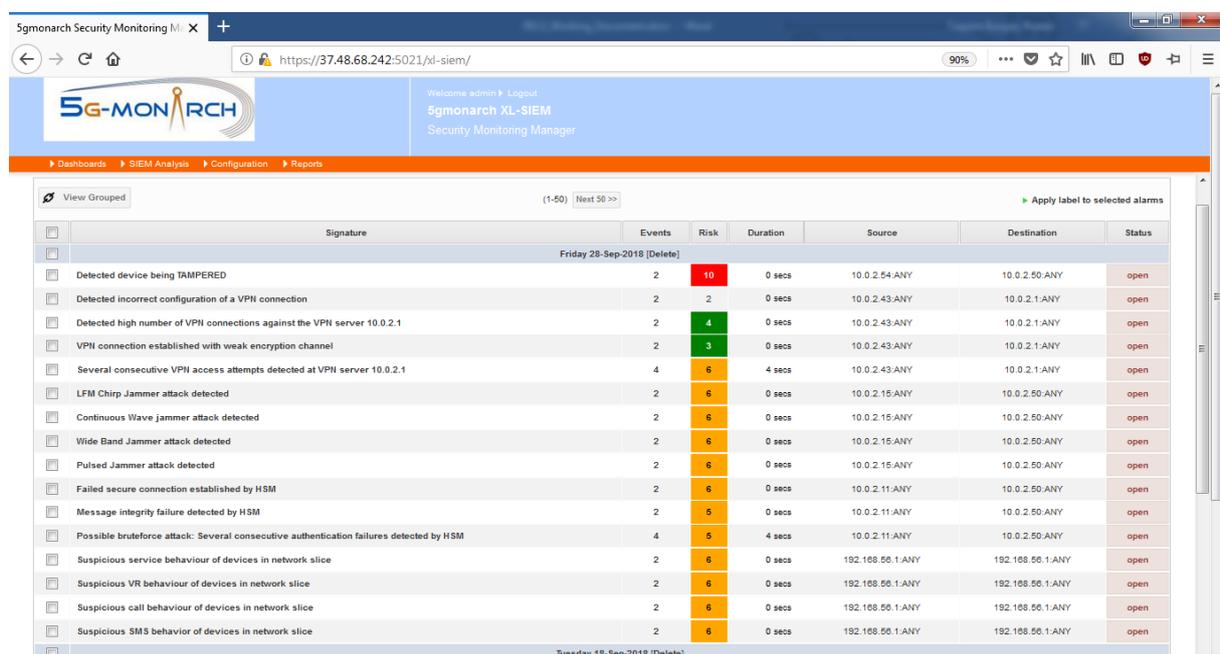
plugins that are needed to understand the format of the events received from the security probes. Additionally, the SthD offers an easy way to deploy new plugins, as long as new security probes are required.



The screenshot shows the 5gmonarch Security Monitoring Manager interface. The main content area displays a table of events received from the SthD to the SMm. The table has columns for Date, Event Name, Risk, Generator, Sensor, Source IP, and Dest IP. The events are listed with their respective risk levels and details.

Date	Event Name	Risk	Generator	Sensor	Source IP	Dest IP
2018-09-28 10:36:36	<a href="#">Detected tampered device</a>	10	Tampering monitor	detector-ns2-stz1	10.0.2.54	10.0.2.50
2018-09-28 10:36:33	<a href="#">Incorrect VPN configuration detected</a>	4	VPN monitor	detector-ns2-stz1	10.0.2.43	10.0.2.1
2018-09-28 10:36:31	<a href="#">High number of VPN connections</a>	4	VPN monitor	detector-ns2-stz1	10.0.2.43	10.0.2.1
2018-09-28 10:36:28	<a href="#">Weak Encryption at VPN connection</a>	5	VPN monitor	detector-ns2-stz1	10.0.2.43	10.0.2.1
2018-09-28 10:36:26	<a href="#">Failed VPN connection attempt</a>	7	VPN monitor	detector-ns2-stz1	10.0.2.43	10.0.2.1
2018-09-28 10:36:24	<a href="#">Failed VPN connection attempt</a>	7	VPN monitor	detector-ns2-stz1	10.0.2.43	10.0.2.1
2018-09-28 10:36:22	<a href="#">Failed VPN connection attempt</a>	7	VPN monitor	detector-ns2-stz1	10.0.2.43	10.0.2.1
2018-09-28 10:36:19	<a href="#">Anomaly: LFM Chrip Jammer detected</a>	6	JammerDetector	detector-ns2-stz1	10.0.2.15	10.0.2.50
2018-09-28 10:36:17	<a href="#">Anomaly: Continuous Wave Jammer detected</a>	6	JammerDetector	detector-ns2-stz1	10.0.2.15	10.0.2.50
2018-09-28 10:36:14	<a href="#">Anomaly: Wide Band Jammer detected</a>	6	JammerDetector	detector-ns2-stz1	10.0.2.15	10.0.2.50
2018-09-28 10:36:12	<a href="#">Anomaly: Pulsed Jammer detected</a>	6	JammerDetector	detector-ns2-stz1	10.0.2.15	10.0.2.50
2018-09-28 10:36:09	<a href="#">Secure Connection Establishment Failure at HSM</a>	6	HSM monitor	detector-ns2-stz1	10.0.2.11	10.0.2.50
2018-09-28 10:36:07	<a href="#">Message integrity failure at HSM</a>	5	HSM monitor	detector-ns2-stz1	10.0.2.11	10.0.2.50
2018-09-28 10:36:04	<a href="#">Password authentication failure at HSM</a>	4	HSM monitor	detector-ns2-stz1	10.0.2.11	10.0.2.50
2018-09-28 10:36:02	<a href="#">Password authentication failure at HSM</a>	4	HSM monitor	detector-ns2-stz1	10.0.2.11	10.0.2.50

Figure 4-17: Events received from the SthD to the SMm sent by different simulated sensors



The screenshot shows the 5gmonarch Security Monitoring Manager interface displaying a list of alerts generated by the SMm after correlating events from simulated sensors. The table has columns for Signature, Events, Risk, Duration, Source, Destination, and Status. The alerts are listed with their respective risk levels and details.

Signature	Events	Risk	Duration	Source	Destination	Status
Friday 28-Sep-2018 [Delete]						
Detected device being TAMPERED	2	10	0 secs	10.0.2.54:ANY	10.0.2.50:ANY	open
Detected incorrect configuration of a VPN connection	2	2	0 secs	10.0.2.43:ANY	10.0.2.1:ANY	open
Detected high number of VPN connections against the VPN server 10.0.2.1	2	4	0 secs	10.0.2.43:ANY	10.0.2.1:ANY	open
VPN connection established with weak encryption channel	2	3	0 secs	10.0.2.43:ANY	10.0.2.1:ANY	open
Several consecutive VPN access attempts detected at VPN server 10.0.2.1	4	6	4 secs	10.0.2.43:ANY	10.0.2.1:ANY	open
LFM Chrip Jammer attack detected	2	6	0 secs	10.0.2.15:ANY	10.0.2.50:ANY	open
Continuous Wave jammer attack detected	2	6	0 secs	10.0.2.15:ANY	10.0.2.50:ANY	open
Wide Band Jammer attack detected	2	6	0 secs	10.0.2.15:ANY	10.0.2.50:ANY	open
Pulsed Jammer attack detected	2	6	0 secs	10.0.2.15:ANY	10.0.2.50:ANY	open
Failed secure connection established by HSM	2	6	0 secs	10.0.2.11:ANY	10.0.2.50:ANY	open
Message integrity failure detected by HSM	2	5	0 secs	10.0.2.11:ANY	10.0.2.50:ANY	open
Possible bruteforce attack: Several consecutive authentication failures detected by HSM	4	5	4 secs	10.0.2.11:ANY	10.0.2.50:ANY	open
Suspicious service behaviour of devices in network slice	2	6	0 secs	192.168.56.1:ANY	192.168.56.1:ANY	open
Suspicious VR behaviour of devices in network slice	2	6	0 secs	192.168.56.1:ANY	192.168.56.1:ANY	open
Suspicious call behaviour of devices in network slice	2	6	0 secs	192.168.56.1:ANY	192.168.56.1:ANY	open
Suspicious SMS behavior of devices in network slice	2	6	0 secs	192.168.56.1:ANY	192.168.56.1:ANY	open
Tuesday 19-Sep-2018 [Delete]						

Figure 4-18: Alerts generated by the SMm after correlating events from simulated sensors

A new plugin was developed to process and normalise such events, proving that incorporating new sources of information from new security probes is easy and efficient, leveraging the easy adaptation of the security infrastructure to the security requirements of an STZ. The simulated testbed has also integrated an SMm, which in this case also acts as a STZm for simulation purposes. This SMm allows to separate the information received from different STZs in different databases, providing with different correlation policies in order not to mix events from different STZs. The SMm deployed in the simulated

testbed is based on the Atos XL-SIEM, an incident correlation engine that allows for the detection of security incidents based on information received from security probes. The Atos XL-SIEM was modified in order to logically separate information received from different SthD, which allowed to simulate the concept of STZs. In this case, although the SMm is able to logically separate information from different STZs, e.g., to apply different correlation rules to events received from different STZs and generate separated security alerts for the different STZs available, the information received from the different SthDs are stored in the same database.

However, it is noted that in a real production environment, different databases can also be deployed, separating the information from different STZs, also physically, if needed. As a result, the STZ approach represents a flexible and convenient way to protect different parts of a network slice, by grouping assets in STZs, customising the security capabilities available in such groups of assets (which are indeed the STZs), tailoring the resources devoted for the security protection of STZs, and adapting them to the security requirements of the STZ.

### 4.3.2 Network behaviour analysis

In addition to the simulated security testbed, six security probes were integrated in the 5G-MoNArch and described in Table 4-3 of the Section 4.3.1. In this regard, methods that involve a behavioural analysis of the network in the context of the specific network slice they are deployed with can be also utilised. This section describes the development of anomaly detection that concern the security Probe named “User and Entity Behaviour Analytics and Network Behavioural Analysis”. This probe can be deployed in every STZ depending on its security requirements as describes in the Figure 4-6 of the Section 4.3.1.

Specifically, two methods are discussed in this section, related to specific security probes integrated in the WP3 framework of 5G-MoNArch. In the first part of the section, a method is developed that applies the use of graphs features to identify groups of users with similar behaviour in mobile networks with great efficiency. The second part of the section is based on the usage of artificial neural network (ANN) models for anomaly detections of network threats. An ANN binary classification model used in a first layer to filter attacks from normal traffic and in a second layer nine ANN models used to categorise the threats into different type of attacks. The main contribution of this method is that it can detect all the type of attacks in comparison to other methods cited in the literature that identify only the attacks that appear more frequently.

The network behaviour analysis (NBA) is the procedure to enhance the security of a network by monitoring traffic and noting unusual actions from the normal operation. A Network behavioural analyser can help a network administrator to minimise the time and effort involved in locating and resolving problems. In a similar context, the Intrusion detection system (IDS) is a device or software application that monitors a network or systems for malicious activity or policy violations and consists a part of an NBA. The IDS is a monitoring infrastructure or application that examines all events or communication traffic taking place in a computing system or over networks and generates reports to the management system by differentiating intrusions, suspicious activities, and other malicious behaviour. Moreover, it is a dynamic discipline that has been associated with diverse techniques and an efficient approach for protecting wireless communications in 5G networks. Network-based IDS grouped into five basic categories the signature-based detection (SBD), the anomaly-based detection (ABD), the specification-based detection (SPBD), the stateful protocol analysis detection (SPAD), and the hybrid intrusion detection. [GQTZ16]

The ABD system refers to an approach of identifying possible inconsistencies between the target events and predefined normal transmissions. The comparison can determine whether there is a partition between normal and unusual behaviours, and the unusual behaviour considered as an active or potential attack, depending on the level of differences. Three common techniques are used for this comparison the statistical-based, the knowledge-based, and the machine learning-based technique.

In the remainder of this section, a graph-based method is first presented, which is used as an ABD. It is described for network mining and visualisation of user activities in a mobile network. The aim of this statistical-based method is the identification of clusters with distinct behaviours in 5G networks. In the second part of this section, an extension of the above method based on deep machine learning techniques

is put forward, aiming to identify different types of network attacks. Finally, the effect of undetected security threats to the network performance is examined in the third part of this section. In this part, the behaviour of attacked users is investigated, followed by a simulation-based analysis which is used to highlight the anticipated throughput reduction caused by abnormal user behaviours leading to network congestions.

#### 4.3.2.1 A graph-based anomaly detection method

This section provides the details about the UEBA probe method that is a part of the simulation testbed created in 5G-MoNArch. The main idea of this method is to evaluate the use of graphs directly as features and to apply graph matching techniques for the analytical task of detecting groups of users with similar behaviour in mobile networks. [PDK+18].

Inspired by the multi-objective approaches that focus on clustering of entities in an environment where entities are mobile devices [KDT15] the graph-based anomaly detection method uses an entity-based analysis scheme in order to analyse any type of record data. A collection of attributes or features defines each entity. Mobile devices, database records, user profiles, articles, and duration are similar examples of the included entities. Multiple multidimensional graph-based features are extracted for each entity, in order to capture its behavioural characteristics motivated by the efficiency of graphs for feature extractions and object recognition [ATK15], [MDA+08].

Let us suppose that the input dataset comprised of a set  $A = \{a_1, \dots, a_{|A|}\}$  of multidimensional attributes, where each attribute  $a_l = \{u_1, \dots, u_{|a_l|}\}$  consists of a set of possible values of the corresponding attribute,  $R = \{r_1, \dots, r_{|R|}\}$  denotes the number of records and each record is a set of attribute values  $r_j = \{u_1, \dots, u_{|r_j|}\}$ . All the attributes are considered to be discrete or transformed into discrete attributes using binning. The dataset entities are defined based on the values of a specific attribute. More specifically, the set of entities  $a_{ent}$  is defined as the set of different values of a specific attribute  $a_l \in A$ , where  $a_{ent} \equiv a_l = \{v_1, \dots, v_{|a_l|}\}$ . In case that the dataset arises from a mobile network, and the task of the identification of anomalous mobile devices, the entities are the mobile devices, as defined from the set of different mobile devices found in the ‘‘source of the call’’ attribute of the communication records.

The  $a_{ent}$  is used to separate the set of records  $R$  into  $|a_{ent}|$  disjoint sets  $R_k$ , such that  $R = \bigcup_{k \in [1, |a_{ent}|]} R_k$  and  $R_i \cap R_j = \emptyset$  for  $i \neq j$ . Each subset of records  $R_k$  is constructed from the records that contain the specific entity  $v_k \in a_{ent}$ :  $R_k = \{r_j | \forall v_k \in r_j, v_k \in a_{ent}\}$ .

The behavioural characteristics of each entity are captured using graph-based features. Each graph-based feature of an entity  $u_k \in a_{ent}$  is an undirected weighted graph,  $G_k^i (V_k^i, E_k^i, f_k^i)$ , where  $V_k^i$  denotes the set of vertices,  $E_k^i \subseteq V_k^i \times V_k^i$  the set of edges and  $f_k^i: E_k^i \rightarrow R^+$  is the function that maps the edges to their respective positive weights and  $i \in [1, n]$  is the index of the  $i^{\text{th}}$  feature out of a total of  $n$  features, and  $k$  is the index of the  $k^{\text{th}}$  entity  $v_k \in a_{ent}$ .

For the creation of the graph feature,  $G_k^i$  a set of dataset attributes is selected  $F_i \subseteq A$ . The set of vertices  $V_k^i$  and the set of edges  $E_k^i$  are defined as follow:

$$V_k^i = \bigcup_{a_l \in F_i} a_l$$

$$E_k^i = \{(v_p, v_j) : \forall v_p \in a_l, v_j \in a_k, \text{ and } a_l, a_k \in F_i, \text{ and } l \neq k, \text{ and } v_p, v_j \in r_q, \text{ where } r_q \in R_k\}$$

The weight of each edge is defined as the number of records that contain the corresponding two vertices used for the creation of the edge as:

$$f_k^i(e_q^i) = |R_k^{q,i}|.$$

Where  $R_k^{q,i} \subseteq R_k$  and  $R_k^{q,i} = \{r_t | \forall v_i, v_j \in e_q^i \text{ and } u_i, u_j \in r_t\}$ . It is assumed that the dataset is a set of Call Detail Records (CDRs) representing the origin, the destination of the communication calls, and the network slice used for the call. The entity attribute  $a_{ent}$  is set to be the origin of the call, and the set of attributes of the graph-based feature  $F_i$  are comprised of the destination of the call and the slice. The weight of the edges corresponds to the co-occurrences of the corresponding vertex-pair in the CDRs.

After the calculation of the graph-based features for each entity, graph matching techniques are employed. A dissimilarity measure of the distance between the different entities for each feature

measures the dissimilarity between the respective graphs. More specifically, the distance between two entities  $u_k$  and  $u_l$  with respect to feature  $F^i$  is defined as follows:

$$D(G_k^i, G_l^i) = D_{eig} + D_{adj}$$

where  $D_{eig}$  is the eigenvalue graph matching method [KPR+11] and  $D_{adj}$  is the absolute difference between the weighted adjacency matrices of  $G_k^i$  and  $G_l^i$ , which considers the content of the graph. Given  $M_k^i$  as the weighted adjacency matrix of  $G_k^i$ :  $D_{adj}(G_k^i, G_l^i) = |M_k^i - M_l^i|$ . The computed distances are used to construct minimum spanning trees  $H_i$  for each graph-based feature, where the vertices are the entities and the edges have weights equal to the corresponding entity distances. The multiple graphs are used as the input to the multi-objective problem [LR13], [GXT10] and the solution is a set of Pareto-optimal solutions, namely the Pareto front, representing multiple trade-offs among the various behavioural characteristics. The proposed graph-based features are able to efficiently encode behaviours related to different communication patterns, such as the destination, the time of the communications events, or different network slice activities. The proposed features constitute an extension of the methods that already exist and based to one dimensional histogram and multidimensional histograms features, [KDT15] to the graph features since that way they are more efficient to capture more complex behaviours. A multi-objective optimisation problem is an optimisation problem that involves multiple objective functions, and can be formulated for  $k$  objectives and the feasible set  $X$  of decision vectors as:

$$\min(f_1(x), f_2(x), \dots, f_k(x)) \\ x \in X$$

where  $f_1(x), f_2(x), \dots, f_k(x)$  are the different objective functions that obtain from the different graph features which constitute an extension of the one-dimensional histogram and multidimensional histograms features that already proposed by [KDT15]. The graph features describe the same information with the histogram features for a specific attribute.

Based on the proposed method two applications of the proposed multi-objective visualisation approach for network mining on multiple datasets in cellular mobile networks were developed [PDKT18]. The first application presents an approach for detecting different user behavioural groups of the CDRs in a mobile cellular network and the second application represents an approach demonstrated on the task of identifying users with anomalous behaviour, which are involved in an SMS flood attack against the core network.

Specifically, the dataset consists of Call Detail Record (CDR) data generated by 1,000 mobile devices, performing calls and SMSs for the duration of one day. Four different groups of 250 user each were simulated:

- Group-1 consists of 250 users with normal SMS and normal call behaviour.
- Group-2 consists of 250 users with high SMS and normal call behaviour.
- Group-3 consists of 250 users with normal SMS and high call behaviour.
- Group-4 consists of 250 users with high SMS and high call behaviour.

The CDRs are comprised of the following fields:

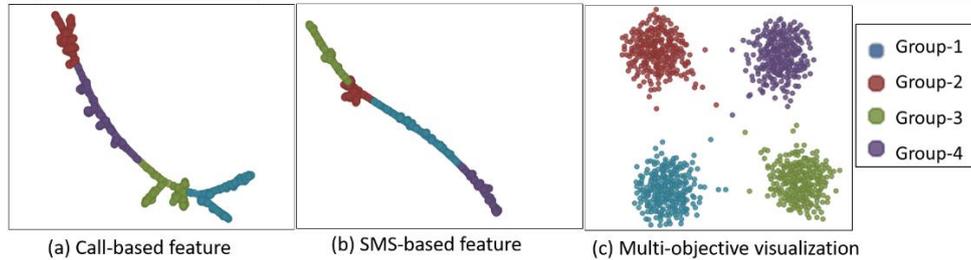
- Origin that is the identifier of the origin of the communication event.
- Destination that is the identifier of the destination of the communication event.
- Time that is the timestamp of the communication event.
- Communication type that is the Call (in the first slice) or SMS (in the second slice).

For the analysis, the Origin attribute is used for the creation of the entities. Additionally, two graph-based features are created: 1) Using the destination and time (with 24 quantisation levels) attributes for the SMS communications, and 2) Using the destination and time (with 24 quantisation levels) attributes for the Call communications.

The results of the first application are summarised in Figure 4-19 and Figure 4-20.

Each point in Figure 4-19 represents an origin of the communication events, while colours are used to illustrate one of the four different behavioural groups. Figure 4-19 (a) and (b) show the single-feature representations as minimum spanning tree, created using the destination and time attributes for the

Call/SMS communications respectively. The different classes are well separated in each feature. Figure 4-19 (c) shows the multi-objective visualisation using the two features in (a) and (b) with equal importance, i.e., weights 0.5 and 0.5 respectively. The different clusters are well separated and easily identified, while they also correspond to the different behavioural groups. The Dunn Index<sup>10</sup> of the clusters in Figure 4-19 (c) is equal to 3.91.



**Figure 4-19: First application – results of the proposed approach for the identification of different user behavioural groups in a cellular mobile network**

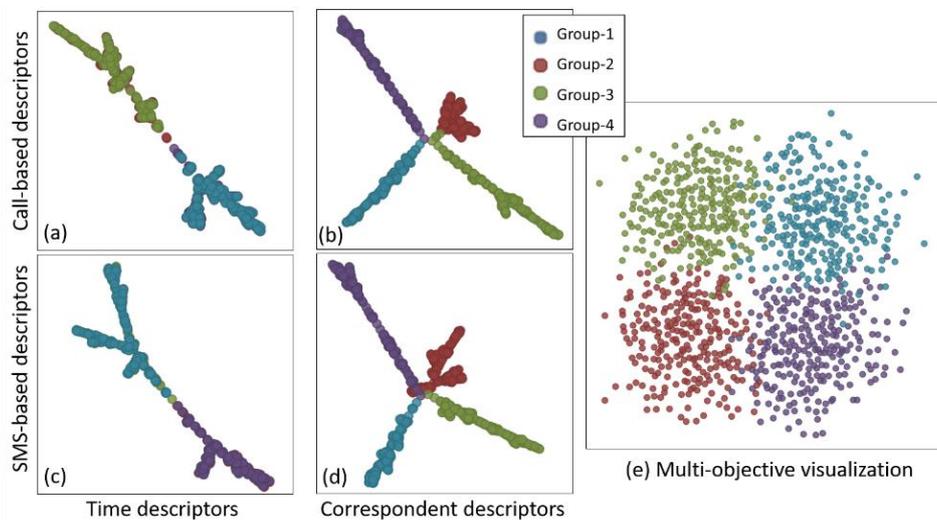
The results obtained from the proposed method are compared with the results that obtain from the multi-objective clustering approach proposed by [KDT15]. The multi-objective clustering approach based on the extraction of histogram features. Let us suppose that the raw data collected of the mobile network traffic are a set of records  $\mathbf{R} \in R$ . Each record  $\mathbf{R}$  is a set of attributes  $r_k$   $\mathbf{R} = \{r_k, k \in A\}$ , where  $A$  is the set of all attribute types. The attribute types is a specific piece of information such as phone call, ID of the caller etc. The histogram  $h_k$  that corresponds to an attribute, is considered as a h-vector, The h-vector defined as  $h = (h_1, h_2, \dots, h_D)$ ,  $h \in R^D$ , where  $D$  are the  $D$  equal-sized bins that the histogram is spited. The value of the  $i$ -th bin  $h_i$  is defined as  $h_i = |\{\mathbf{R} \in R \cap C \mid r_k \in \text{bin}_i\}|$ , where  $|\cdot|$  denotes the cardinality of a set,  $C$  is a set of records satisfying specific constraints for the construction of the histogram, such as keeping only those SMS messages that are sent towards premium numbers,  $k$  is the associated attribute type and  $\text{bin}_i$  denotes the set of values in the range of the  $r_k$  attribute that constitute the  $i$ -th bin.

The four histogram features are histograms of the frequency of the communication events within a day, with bin sizes equal to one hour. More specific the Time Histogram Descriptor (THD) that is histogram where of the hours of the day at which a user sends SMS messages defined from the value  $D = 24$ , the  $\text{bin}_i = \{\mathbf{R} \in R \mid r_{\text{hour}} = i\}$  and the  $C = \{\mathbf{R} \in R \mid r_{\text{type}} = \text{“SMS”} \text{ AND } r_{\text{from}} = u\}$ , where  $r_{\text{from}}$  is the attribute regarding the ID of the user from whom the event originated In a similar way it is obtained the THD for type calls events within a day with bin sizes equal to one hour. The Recipient Histogram Descriptor (RHD) is a histogram of the recipients to whom a user sends SMS messages defined from the value  $D$  that is equal to the number of contacts of each user, the  $\text{bin}_i = \{\mathbf{R} \in R \mid r_{\text{to}} = c_i\}$ , where  $c_i$  is the ID of the  $i$ -th contact of the user and the  $C = \{\mathbf{R} \in R \mid r_{\text{type}} = \text{“SMS”} \text{ AND } r_{\text{from}} = u\}$ , where  $r_{\text{to}}$  is the attribute regarding the ID of the user to which an event is directed. In a similar way it is obtained the RHD for type calls events within a day with bin sizes equal to one hour. The distance metric used for the histogram-based features is L1 norm, [KDT15].

Figure 4-20 shows the multi-objective visualisation using the four features proposed in [KDT15]: (a) and (c) show the THD with respect to the Call and SMS activities respectively. Figure 4-20 (b) and (d) show the RHD with respect to the Call and SMS activities respectively. The RHD features, in (b) and (d), are able to efficiently identify the different behavioural groups, since they have different number of destinations. On the other hand, the THD features, in (a) and (c) are not able to completely separate the different behavioural groups since the groups 1 and 3 have normal SMS behaviour and groups 1 and 2 have normal call behaviour. The multi-objective visualisation in (e) shows the four features in (a), (b), (c), and (d) with equal importance, i.e., weights equal to 0.25, 0.25, 0.25, and 0.25 respectively. As it

<sup>10</sup> The Dunn Index [DUN73] is a metric that can be used for evaluating clustering algorithms in order to identify sets of clusters that are compact, well separated, and with a small variance between members of the cluster. The means of different clusters shall be sufficiently far apart compared to the variance within each cluster. The higher the Dunn Index is, the better is the clustering for a given assignment of clusters.

can be seen in (e), the different behavioural groups are not separated well. This happened due to the inclusion of the RHD features, in (b) and (d), which are not able to completely separate the different behavioural groups. The Dunn Index is equal to 1.82.



**Figure 4-20: First application – results based on four features**

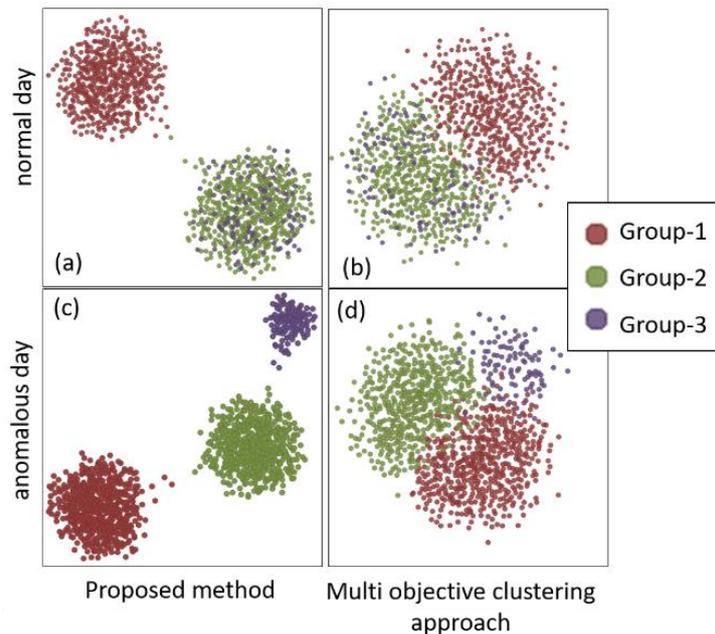
The second application represents an approach demonstrated on the task of identifying users with anomalous behaviour, which are involved in an SMS flood attack against the core network. Three different behaviours are included in the dataset:

- Group-1 consists of 500 users with normal SMS and normal call behaviour.
- Group-2 consists of 500 users with high SMS and normal call behaviour.
- Group-3 consists of 100 users with anomalous SMS behaviour and normal call behaviour.

Figure 4-21 (see next page) shows the results of the SMS flood attack dataset. Each point in the visualisation represents a different origin of the communication events, while colours are used here to represent the three different behavioural groups. Specifically, red colour represents the low SMS users, green the high SMS users, and purple the anomalous users (which are active only in the last day). Figure 4-21 (a) and (b) illustrate a normal day using the proposed approach (a) and a multi-objective clustering approach (b). The proposed approach is able to more efficiently discriminate between the two different normal SMS behaviours. The Dunn Indices are 3.78 in (a) and 1.61 in (b), respectively. Figure 4-21 (c) and (d) show the anomalous day using the proposed approach and the multi-objective clustering approach. The proposed approach is able to efficiently separate the anomalous cluster from the two normal ones. The reason for this is that the graph-based features and the graph matching techniques are able to more efficiently characterise the user activities of the users than the simple histogram features. The Dunn Indexes are 3.2 and 1.69 respectively.

Comparing the proposed method with the method that already exist, the Dunn index is higher in the proposed method which means that it derives to a better clustering for the given groups in each application, [PDK+18]. The proposed approach is able to identify Pareto-optimal visualisations, which correspond to different trade-offs between the available features. Selecting a solution in the middle of the Pareto front results in visualisation that combine the characteristics of all the available features, which can uncover useful data relationships and provide evidence with respect to the efficiency in visualising the behavioural similarities of users and in separating different behavioural patterns. The results obtain for the experimental results prove that the proposed graph-based features are able to encode the behaviours more efficiently in comparison with the results that obtain from the four histogram features in the applications that concern different behavioural groups that in a cellular mobile network. The data of the applications can contain billing information about the calls and SMSs

performed by the mobile users, including time of communication, its duration, the IDs of the communication origin and the recipient or the slice for each event.



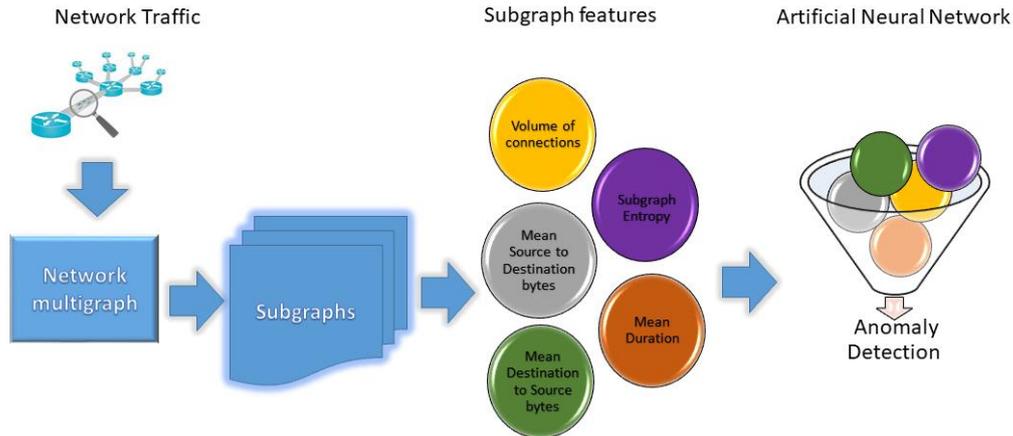
*Figure 4-21: Second application – results for different behavioural groups*

#### 4.3.2.2 An extension of the anomaly detection method based on machine learning

The graph-based method is an identical approach for network mining on multiple datasets especially in cellular mobile networks. Following the methodology presented so far, the next goal is to extend both the types of attacks beyond Call Detailed Records and the methodology that is used to categorise them as well.

Regarding the methodology, the next goal is to combine the features that arise from a graph-based method with ANN models in order to develop tools for identifying anomaly-based detection in 5G Networks. The large volume of data that is expected in the 5G networks and the network architecture necessitate the usage of neural networks, especially on issues related to network security. In the first part of this section inspired by the features obtained from the graph-based descriptor and by the flexibility and adaptability to environmental changes provided by an ANN, we propose a methodology that combines the above assets to identify anomaly detection in 5G Networks. More specifically we assume graph features as inputs of an ANN model for anomaly detection that will identify threats from normal traffic. In the second part of this section we develop a methodology for anomaly detection that lead to threat identification per attack category, this methodology based only on the features that arises from the dataset because the properties that arise from graph features are not appropriate for the anomaly detection per type of attack.

The communication activities within a network can be linked with a weight graph network representation, where the vertices denote the IP addresses and the edges denote the communication among the addresses regarding specific characteristics. In the current method, we assume that the network traffic is represented by a directed weighted multigraph, the specified subgraph features that are obtained from the individual subgraphs constitute the inputs to an ANN model. The output of the ANN model leads to a binary classification and distinguishes whether the communication that has the above features is either normal or abnormal. The proposed methodology uses a smaller number of features in comparison with methods that already exist [MS16]. The usage of many features requires cost and time to be available in a real network. Moreover, the subgraph-based architecture reduces dramatically the number of the required inputs. Figure 4-22 describes the general architecture of the proposed methodology.



**Figure 4-22: Architecture of the proposed methodology for anomaly detection based on graph features and ANN models**

Let  $G(V, E, f_k)$  be a directed weighted multigraph, where  $V$  denotes the set of vertices  $E \subseteq V \times V$ , the set of edges and  $f^i: E \rightarrow R^+$ , the function that maps their respective positive weights. Denoted  $G_k$  the  $k$ - subgraph of the graph  $G$ , where  $V_k = \{v_{1k}, v_{2k}, \dots, v_{lk}\}$  represents the set of vertices,  $E_k \subseteq V_k \times V_k$  the set of edges and  $f_k^i: E_k \rightarrow R^+$ , the function of the weights for each  $k$ - subgraph. If  $v_{xk}$  is adjacent to the  $v_{ky}$  vertex, then we say that  $v_{xk}$  and  $v_{ky}$  are neighbours. In the current method, we consider the one edge subgraphs to be derived from the initial multigraph.

**Table 4-4: Description of sub graph features that constitute the inputs of the ANN model for anomaly detection based on graph features**

Basic features	Description
Mean source to destination (MStD) bytes.	Measures the average number of bytes transferred from the Source IP to the Destination IP.
Mean destination to source (MDtS) bytes.	Measures the average number of bytes transferred from the Destination IP to the Source IP.
Mean duration (MD)	Measures the average time of connection given a certain period
Subgraph Features	Description
$f_{vol}$ : Volume of contacts	Measures the number of times that two addresses were contacted given a certain period (Expression 4-1).
$I$ : Weighted entropy	Measures the information rate achievable by communicating two addresses (Expression 4-2).

Table 4-4 describes the basic and the subgraph features that set the inputs to the ANN model and provides a brief description for each of those features. More specifically, the volume of a weighted graph [PDT16] that captures the size of the graph regarding the number of connections is calculated using the following expression

$$f_{vol} = \sum_{e_i \in E_k^i} g(f_k^i(e_j)) \quad (\text{Expression 4-1})$$

$$\text{where } g(f_k^i(e_j)) = \begin{cases} f_k^i(e_j), & \text{for } f_k^i(e_j) \neq 0 \\ 0, & \text{for } f_k^i(e_j) = 0 \end{cases}$$

The entropy for the edge weighted graph  $G_k^i$  [K16] is defined as follow,

$$I(G_k^i, f_k^i) = - \sum_{e_i, e_j \in E_k^i} p_{e_i e_j} \log(p_{e_i e_j}) \quad (\text{Expression 4-2})$$

$$\text{where } p_{e_i e_j} = \frac{f_k^i(e_j)}{\sum (f_k^i(e_j))}$$

A part of the UNSW-NB 15 dataset is used, which contains nine types of attacks [MS2015], [MS2016] to justify the proposed method. The raw network packets of the UNSW-NB 15 data set were created by the IXIA Perfect Storm tool in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) for generating a hybrid of real modern normal activities and synthetic contemporary attack behaviours. This dataset has been chosen because it describes with accuracy current network traffic accurately. The attack types were classified into the following nine groups:

- **Fuzzers:** an attack in which the attacker attempts to discover security loopholes in a program, operating system, or network by feeding it with a massive inputting of random data to make it crash.
- **Analysis:** a variety of intrusions that penetrate the web applications via ports (e.g., port scans), emails (e.g., spam), and web scripts (e.g., HTML files).
- **Backdoor:** a technique of bypassing a stealthy normal authentication, securing unauthorised remote access to a device, and locating the entrance to plain text as it is struggling to continue unobserved.
- **DoS:** an intrusion, which disrupts the computer memory resources, to be extremely busy, in order to prevent the authorised requests from accessing a device.
- **Exploit:** a sequence of instructions that takes advantage of a glitch, bug, or vulnerability to be caused by an unintentional or unsuspected behaviour on a host or network.
- **Generic:** a technique that establishes against every block-cipher, using a hash function to collide without respect to the configuration of the block-cipher.
- **Reconnaissance:** can be defined as a probe. It is an attack that gathers information about a computer network to evade its security controls.
- **Shellcode:** an attack in which the attacker penetrates a slight piece of code, starting from a shell, to control the compromised machine.
- **Worm:** an attack whereby the attacker replicates itself in order to spread on other computers. Often, it uses a computer network to spread itself, depending on the security failures on the target computer to access it.

Figure 4-23 describes the distribution of the attacks across the connections and their corresponding frequencies. The total number of records is 2,540,044. Normal records are the majority (87,375%) while attacks records represent the 12,69% as in a real network. Generic attacks are the most common type of attacks and the Worms seem to be the last encountered among attacks.

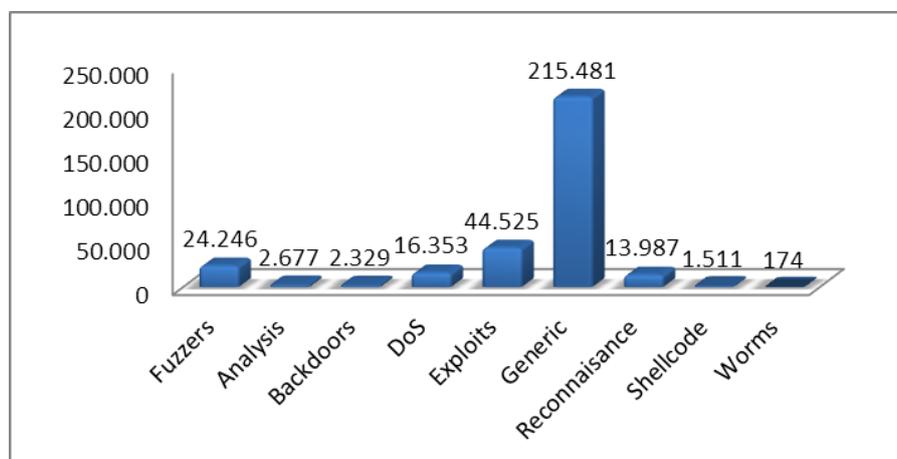


Figure 4-23: Frequency of each type of attack in the UNSW-NB dataset

Based on the described distribution of the UNSW-NB15 dataset we considered a part of 700000 records with normal and abnormal activity. A sampled training set and a sampled test set, that contains created form this part considering the one-ended subgraphs of the initial multigraph. For each subgraph are calculated the volume of the contacts, the mean source to destination bytes, the mean destination to source bytes, the mean duration and the weighted graph entropy are calculated. The output of the ANN model leads to a binary classification regarding the existence of normal or abnormal behaviour.

Different combinations of hidden layers have been checked, including neurons, activation functions and optimisers until the proposed model was concluded. The proposed ANN model consists of four hidden layers with 12, 8, 4, 4 neurons for each layer, respectively. The first three layers based on the Relu activation function and the last layer uses the Sigmoid activation function.

The accuracy of the proposed model is 97,47%. Comparing the accuracy with the most recent techniques that used to identify the anomaly detection in the same dataset [MS15], we conclude that the proposed method improves the accuracy. Table 4-5 shows the comparison between state-of-the-art results in literature and the proposed anomaly detection method based on graph features.

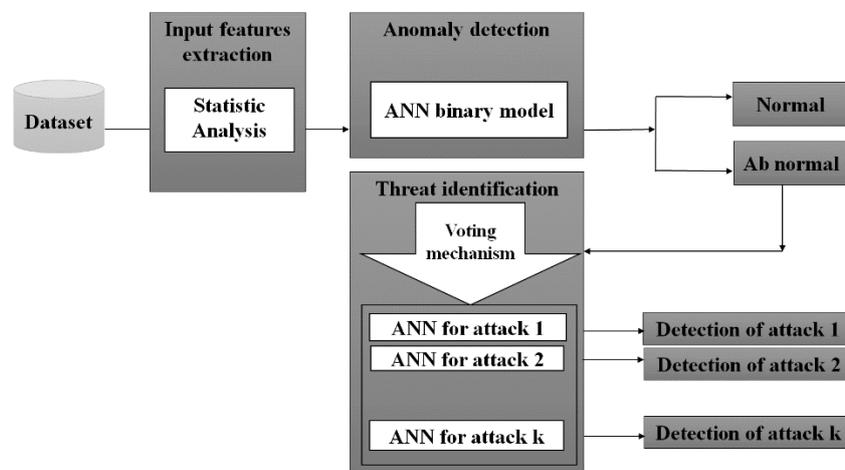
**Table 4-5: Comparison between state-of-the-art and the proposed method for anomaly detection**

Techniques	(References)	Accuracy	FAR
Decision tree	[MS15]	85.56%	15.78%
ANN models	[WM05], [MS15]	81.34%	21.73%
Expectation maximisation clustering	[SB12], [MS15]	78.34%	23.79%
Proposed method and related ANN model		97,47%	3.5 %

The method already described combines the features that arise from graph properties and the artificial neural networks to identify anomalies with high accuracy. In order to detect threats per attack category, the usage of features is based on graph properties is not helpful and therefore it is suggested to follow a different procedure.

The usage of the ANN multiclass classification model sets the basis to develop a methodology procedure that will lead to threat identification per attack category. Since the frequency of threats varies among categories and there are threats such as Worms that have a very low frequency (0.1% among attacks and 0.007 % among normal and attacks), the detection procedure evolves into a very complex issue. Recent studies that develop ANN models fail to predict effectively all type of attacks [C18].

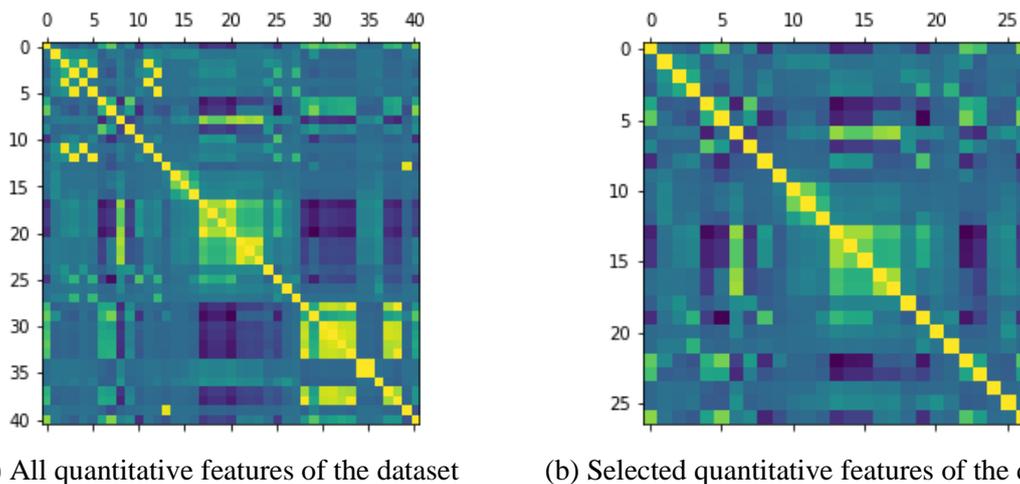
The proposed methodology aims to increase the predicted number of attacks and to take the stage for future research on this topic. Since the attacks represent 13-15% of network traffic, we propose a two-layer methodology inspired by Catak, [C18]. Figure 4-24 describes the architecture of the methodology procedure that consists of three main parts:



**Figure 4-24: The Architecture of the proposed anomaly detection methodology procedure**

- The feature extraction. In this part statistical analysis measures set the threshold for the reduction of the demanded features of the corresponded dataset. The feature reduction is very important since fewer features save time and speed for the detection procedure.
- The first level of anomaly detection is the stage that based on ANN binary classification and filters abnormal traffic from normal traffic.
- The second level of threat detection, based on the development of different ANN models, that include only the ab-normal outcomes and categorises them into different types of attack.

The UNSW-NB15 dataset consists of 45 features (c.f. Appendix A) and is used to validate the proposed methodology. The 45 features of the UNSW-NB15 get either numerical or categorical values. We assume the absolute value of the coefficient correlation to be the criterion in order to select the most suitable features that get numerical values from the dataset. Two or more variables are correlated if the coefficient of correlation is very close to 1, that means that the existence of one variable can predict the value of the other variable very well. We assume that if two variables have a coefficient correlation close to 1, then one of them can replace the other since no extra benefit arises from the existence of both of variables. Figure 4-25 (a) describes the coefficient correlation among the quantitative features of the dataset. The yellow areas denote very strong correlation among features while the green and blue areas low correlation level.



**Figure 4-25: Heat maps of coefficient correlation prove the existence of less strong correlations (yellow area) in case (b) where the features excluded**

A detailed correlation matrix among all the features of the dataset is provided in Appendix A. Based on the correlation matrix Table 4-6 gathers the excluded and the corresponded selected variables regarding their value of the coefficient of correlation. For example, the variable “spkts” that denotes the count of source to destination packets is strongly correlated with the variable “Sbytes” which denotes the source to destination transaction bytes ( $r=0.9631$ ) and with the variable “Dbytes” ( $r=0.971069$ ) denotes the destination to source transaction bytes. Following the same procedure for the remaining variables, we lead to the extraction of 13 features of the overall dataset. Figure 4-25 (b) shows the heatmap of coefficient correlation among the selected features.

**Table 4-6: Coefficient of correlation among features with value close to 1 leads to the extraction of 13 features of the overall dataset**

Selected features (cf. Appendix A)	Excluded features	Coefficient correlation (r)
spkts	Sbytes	0.963791
	Sloss	0.971069
dpkts	Dbytes	0.971907
	Dloss	0.978636
sinpkt	Is_sm_ips_ports	0.941319

swin	Dwin	0.99014
synack	Tcprrt	0.949468
Ct_src_dport_ltm	Ct_srv_src	0.86601
	Ct_dst_ltm	0.96025
	Ct_dst_sport	0.906793
	Ct_dst_src_ltm	0.869941
	Ct_src_ltm	0.89743
	Ct_srv_dst	0.8685

The second step of the procedure concerns the first layer of anomaly detection. In this layer an ANN neural network model for binary classification is used to filter the normal from abnormal incidents. Since among the features there exist categorical values (e.g. transaction protocol, state of the dependent protocol and service) that are not considered in the coefficient of correlation threshold, during feature selection, we will compare whether their existence of them affect the accuracy of the ANN network model. An ANN network with seven hidden layers calculates 75,5 % accuracy. Comparing the ANN model with the model without the qualitative variables we conclude that the coding of qualitative variables do not improve the accuracy of the existent model (72.4%) that means that we can excluded the qualitative variables from the selected features as well.

The third step of the procedure concerns the second level of the threat detection. Recent surveys based in the same intrusion detection dataset have proved that it is very difficult to develop a neural network that will detect with high precision the categories of attack [C18]. The main reason for this obstacle is the big range among the size of the classes and the lack of enough information for the classes that they rarely appear.

The development of multi-classification models due to the existence of imbalanced data leads to the lack of detecting attacks that rarely appear. In order to overcome the previous issues we assume an approach that based on the usage of different ANN models which correspond to different type of attacks. More specific for the UNSW-NB15 we developed nine ANN binary models, where each model aims to identify specific type of attack. An overview about the different models is provided in Table 4-7.

- The ANN model that detects Analysis attacks, consists of six layers. The input layer has 25 nodes, the four hidden layers consist of 300, 100, 50, 25 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 81 epochs. The accuracy and the precision for this model are 63.12 % and 58.33% respectively.
- The ANN model that detects the Backdoors attacks, consists of eight layers. The input layer has 25 nodes, the six hidden layers consist of 75, 50, 25, 15, 10, 5 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 500 epochs. The accuracy and the precision for this model are 64.57% and 60.88% respectively.
- The ANN model that detects the DoS attacks, consists of three layers. The input layer has 25 nodes, one hidden layer consists of 1000 nodes and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 1000 epochs. The accuracy and the precision for this model are 90.90% and 100.0% respectively.
- The ANN model that detects the Exploits attacks, consists of four layers. The input layer has 25 nodes, the two hidden layers consist of 600, 300 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 30 epochs. The accuracy and the precision for this model are 71.79% and 50.04% respectively.
- The ANN model that detects the Fuzzers attacks, consists of nine layers. The input layer has 25 nodes, the seven hidden layers consist of 175, 125, 100, 50, 20, 10, 5 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation

function has been used across layers and the model has been trained for 26 epochs. The accuracy and the precision for this model are 81.37% and 89.16% respectively.

- The ANN model that detects the Generic attacks, consists of seven layers. The input layer has 25 nodes, the five hidden layers consist of 50, 25, 20, 10, 2 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 150 epochs. The accuracy and the precision for this model are 73.11% and 61.14% respectively.
- The ANN model that detects the Reconnaissance attacks, consists of five layers. The input layer has 25 nodes, the three hidden layers consist of 1000, 550, 550 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 125 epochs. The accuracy and the precision for this model are 59.27% and 41.03 % respectively.
- The ANN model that detects the Shellcode attacks, consists of six layers. The input layer has 25 nodes, the four hidden layers consist of 500, 500, 500, 400 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 250 epochs. The accuracy and the precision for this model are 51.25% and 51.20% respectively.
- The ANN model that detects the Worms attack, consists of five layers. The input layer has 25 nodes, the three hidden layers consist of 1000, 550, 550 nodes respectively and the output layer that leads to the binary classifications consists of one node. The sigmoid activation function has been used across layers and the model has been trained for 200 epochs. The accuracy and the precision for this model are 87.83% and 25 % respectively.

**Table 4-7: Overview of ANN model architecture, accuracy and precision per type of attack obtained from the proposed method**

Type of attack	Model architecture	Accuracy	Precision
Analysis	6 layers (25-300-100-50-25-1)	63.12%	58.33%
Backdoors	8 layers (25-75-50-25-15-10-5-1)	64.57%	60.88%
DoS	3 layers (25-1000-1)	90.90%	100.0%
Exploits	5 layers (25-600-300-1)	71.79%	50.04%
Fuzzers	9 layers (25-175-125-100-50-20-10-5-1)	81.37%	89.16%
Generic	7 layers (25-50-25-20-10-2-1)	73.11%	61.14%
Reconnaissance	4 layers (25-1000-550-550-1)	59.27%	41.03%
Shellcode	6 layers (25-500-500-500-400-1)	51.25%	51.20%
Worms	6 layers (25-1000-500-500-1)	87.37%	25.00%

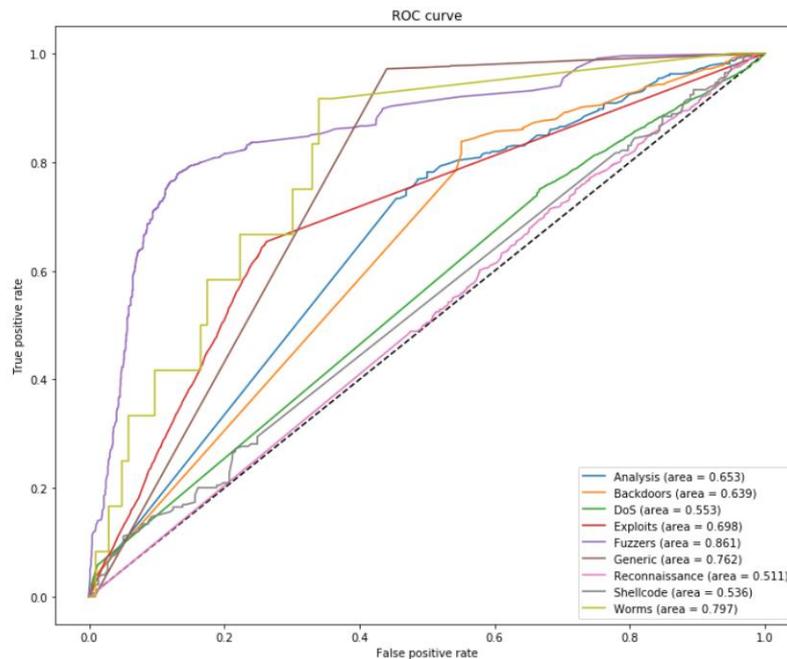
Comparing the experimental results of the proposed method in terms of precision % (and recall %), see Table 4-8, with the experimental results that use the same dataset in the state of the art (Decision tree (DT), Random forest (RF) and Adaboost (AB)) and different artificial network (ANN) model classifiers, the proposed method is superior in terms of classification per type of attack.

More specifically, the ANN model proposed by [C18] cannot detect the Analysis, the Backdoors, the Reconnaissance, the Shellcode and the Worms type of attacks. The ANN model based on methodology that proposed by [TE18] cannot detect the Backdoors type of attack and the ANN model based on methodology that proposed by [BAE17] cannot detect the Analysis, the Backdoor, the Fuzzers, the Shellcode and the Worms type of attack. The DT and the RF classifiers are not detecting the Analysis, the Backdoors, the Shellcodes and the Worms type of attacks. The AB classifier cannot detect the Analysis, the Backdoors and the Worms type of attack. Finally, in terms of binary classification, the RF classifier detects with higher precision and recall the abnormal incidents from the normal. The advantage of the RF classifier is not exceeded in case of the classification per attack since it cannot detect all type of threats in case of abnormal incidents.

**Table 4-8: Comparison of the experimental results in terms of precision % (and recall %)**

	Proposed method	ANN [C18]	ANN [TE18]	ANN [BAE17]	DT [C18]	RF [C18]	AB [C18]
<b>Classification per attack</b>							
Analysis	<b>58.33</b> (75.47)	NA (NA)	0.21 (0.14)	0.0* (0.0)*	NA (NA)	NA (NA)	NA (NA)
Backdoors	<b>60.88</b> (83.76)	0.0* (0.0)*	0.0* (0.0)*	0.0* (0.0)*	0.0* (0.0)*	0.0* (0.0)*	0.0* (0.0)*
DoS	<b>100.0</b> (0.15)	27 (6.0)	30 (49)	6.25 (0.02)	<b>100</b> (0.0)*	<b>100</b> (0.0)*	32 (49)
Exploits	50.04 (61.40)	62 (84)	65 (69)	31.58 (56.93)	54 (94)	54 (94)	59 (62)
Fuzzers	<b>89.16</b> (76.04)	5.0 (63)	33 (75)	0.0* (0.0)*	77 (79)	77 (79)	<b>78</b> (52)
Generic	61.14 (97.15)	49 (39)	99 (96)	90.81 (97.81)	<b>100</b> (97)	<b>100</b> (97)	98 (97)
Reconnaissance	41.03 (97.42)	7.0 (0.0)*	20 (0.028)	40.45 (33.74)	<b>91</b> (60)	<b>91</b> (60)	65 (74)
Shellcode	<b>51.20</b> (85.27)	0.0* (0.0)*	5 (16)	0.0* (0.0)*	0.0* (0.0)*	0.0* (0.0)*	26 (5.0)
Worms	25 (8.33)	0.0* (0.0)*	<b>50</b> (0.02)	0.0* (0.0)*	0.0* (0.0)*	0.0* (0.0)*	0.0* (0.0)*
<b>Classification binary</b>	83.9 (90.78)	75 (2.0)	NA (NA)	86.74 (93.3)	92 (99)	<b>98</b> (98)	95 (96)

\*Based on the state of the art [C18], [TE18], [BAE17]

**Figure 4-26: The ROC curve per each type of attack**

Although the main attribute of the proposed method it is not the excellent accuracy for all the type of attacks we consider that it constitutes an important step toward the anomaly detection of all the type of attacks, since the methods that already exists can identify only certain type of attacks that appear more

often in comparison with other attacks [C18]. The ability of the detection for each model illustrated from the ROC curve, as shown in Figure 4-26. The ROC curve per each type of attack illustrates the ability of the detection for each ANN model. The area under each curve denotes the capability of each model to distinguish the incidents between the classes of the model.

### 4.3.2.3 Behaviour of attacked users and effect on the throughput performance

In this section the performance of the network traffic in case that an attack occurs is examined. More specifically, this section contains the development of a DoS attack scenario and describes how the throughput performance is affected when a security attack occurs.

The DoS attacks are the most common cited attack scenario in the literature, [TYZ+11] [GB18] it is noticed that DoS traffic is similar to the flash crowd attack traffic. The flash crowd attack is an effect of a high volume of illegitimate packets from attack sources, it occurs similar to the flash crowd traffic that generated by real users. A flash crowd traffic can obtain for example when an unpopular site becomes popular after being mentioned in a popular newsfeed. The attack sources follow the programmer's instructions and they seem to have high degree of automation. When the attack sources perform a DDoS attack on the victim, their transmission rate appears to be predictable. On the other hand, human users unpredictably create request packets at any period [TYZ+11].

Following the same procedure proposed by [GB18] we assume that the traffic which is related to the normal traffic can be generated by using an HTTP generator and the attack traffic can be generated by a constant bitrate generator. The HTTP generator demands information regarding the distribution of the main objects, the distribution of the embedded objects, the distribution of the number of the embedded objects and the distributions of the reading and parsing time respectively.

The main object follows the truncated lognormal distribution. The probability density function (PDF) of the truncated lognormal distribution  $f_x$  obtains from expression (3),

$$f_x = \frac{1}{\sqrt{2\pi} \sigma_1 x} \exp \left[ \frac{-(\ln x - \mu_1)^2}{2\sigma_1^2} \right], \quad x \geq 0 \quad (3)$$

Let us suppose that  $x$  denotes the size of the main object in bytes, then  $\bar{x}$  denotes the mean value of  $x$ ,  $X_{\max}$  and  $X_{\min}$  the maximum value and the minimum value of  $x$ . The estimators of the parameters  $\mu_1$  and  $\sigma_1$  of the PDF denoted  $\hat{\mu}_1$  and  $\hat{\sigma}_1$  are calculated from the expressions (4a) and (4b),

$$\hat{\mu}_1 = \log \bar{x} - 0,5 \cdot \log \left( \frac{1 + \text{var}(x)}{\bar{x}^2} \right) \quad (4a), \quad \hat{\sigma}_1 = \sqrt{\log \left( \frac{1 + \text{var}(x)}{\bar{x}^2} \right)} \quad (4b).$$

The embedded objects follow the truncated lognormal distribution as well. The PDF of the embedded objects  $f_y$  obtains from expression (5) where  $y$  and denotes the size of the embedded objects in bytes,

$$f_y = \frac{1}{\sqrt{2\pi} \sigma_2 y} \exp \left[ \frac{-(\ln y - \mu_2)^2}{2\sigma_2^2} \right], \quad y \geq 0 \quad (5)$$

The mean value of  $y$  denoted  $\bar{y}$  and the  $Y_{\max}$  and  $Y_{\min}$  are the maximum and the minimum value of  $y$ . The estimators of the parameters  $\mu_2$  and  $\sigma_2$  denoted  $\hat{\mu}_2$  and  $\hat{\sigma}_2$  obtain from the expressions (6a) and (6b),

$$\hat{\mu}_2 = \log \bar{y} - 0,5 \cdot \log \left( \frac{1 + \text{var}(y)}{\bar{y}^2} \right) \quad (6a), \quad \hat{\sigma}_2 = \sqrt{\log \left( \frac{1 + \text{var}(y)}{\bar{y}^2} \right)} \quad (6b).$$

The number of embedded objects follow the truncated pareto distribution. The PDF of the number of embedded object  $f_z$  obtains from expression (7), where  $z$  denotes the number of the embedded objects

$$f_z = \frac{a \cdot k^a}{z^{a+1}} \quad \text{if } k \leq z < m, \quad f_z = \left( \frac{k}{m} \right)^a \quad \text{if } z = m \quad (7)$$

The mean value of  $z$  is  $\bar{z}$  and the  $Z_{\max}$ ,  $Z_{\min}$  are the maximum and the minimum value of  $z$ . The estimators of the parameters  $a$ ,  $k$ ,  $m$  denoted as  $\hat{a}$ ,  $\hat{k}$ ,  $\hat{m}$  respectively and calculated from expressions (8a)-(8c):

$$\hat{a} = \frac{2\text{Var}(z)}{\text{Var}(z) - \bar{z}} \quad (8a), \quad \hat{k} = Z_{\min} \quad (8b), \quad \hat{m} = Z_{\max} \quad (8c).$$

The reading time follows exponential distribution. The PDF of the reading time  $f_t$  with parameter  $\lambda_1$  obtains from the expression (9) where  $t$  denotes the reading time (in sec),

$$f_t = \lambda_1 e^{-\lambda_1 t}, \quad t \geq 0. \quad (9)$$

The unbiased estimator of the  $\lambda_1$  parameter  $\hat{\lambda}_1$  calculated as follow,  $\hat{\lambda}_1 = \bar{t}$ .

The parsing time follows the exponential distribution as well. The PDF of the parsing time  $f_t$  with parameter  $\lambda_2$  obtains from expression (10), where  $t$  denotes the parsing time (in sec),

$$f_t = \lambda_2 e^{-\lambda_2 t}, t \geq 0. \quad (10)$$

The unbiased estimator of the  $\lambda_2$  parameter is  $\hat{\lambda}_2$  and calculated as,  $\hat{\lambda}_2 = \bar{t}$ .

The UNSW-NB 15 dataset is used to validate the proposed methodology, based on this dataset we focus only on the DoS attacks and we calculate the estimated parameters from the corresponded features of the dataset, for each distribution that is described from expression (3) – (10).

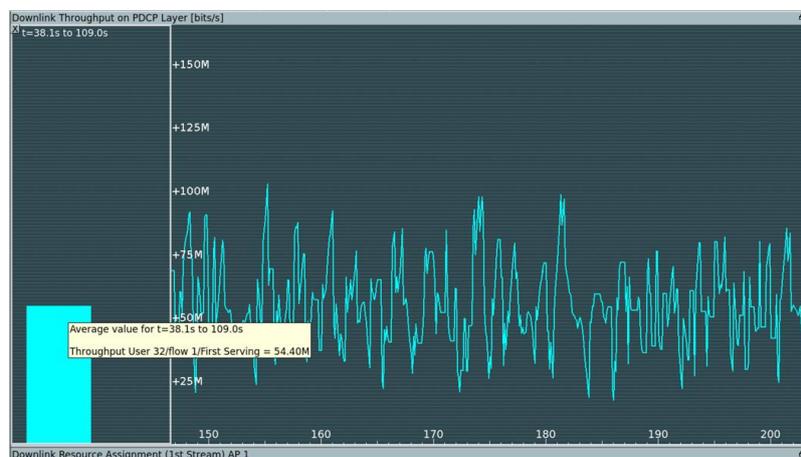
Table 4-9 describes in details the values of the estimated parameters that concern the distribution of each attribute for the HTTP traffic generator which related to the normal traffic and for the Constant bitrate generator which related to the constant bitrate traffic. Under such traffic assumptions, two simulation experiments developed.

**Table 4-9: Corresponded attributes and estimated parameters**

Generator	Attribute	Distribution	Estimated parameters
HTTP traffic generator	main object	truncated lognormal	$\mu_1=4.422, \sigma_1=0.9080$
	embedded objects	truncated lognormal	$\mu_2=4.3697, \sigma_2=0.9199$
	number of embedded object	truncated lognormal	$a = 2.0074, \kappa=1, m = 364$
	reading time	exponential	$\lambda_1= 1.743$
	parsing time	exponential	$\lambda_2=1.056$
constant bitrate generator	rate packet count packet size		*rate= 72727.272 *packet count= 2 *packet size=100

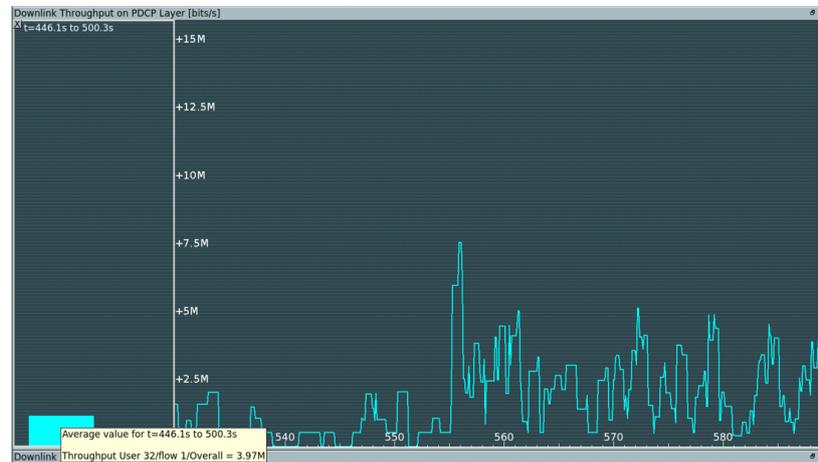
\*this is an example of the multiple cases that obtain of the dataset.

The first simulation experiment was based on 32 users and describes the situation of normal traffic. That is, one user is having HD video traffic and 31 users have traffic associated with the http protocol. This scenario is used here as the reference scenario, illustrating the throughput of the user with HD video traffic in normal situations. Figure 4-27 depicts the throughput value on PDCP level of the HD video traffic user within a dedicated time window. The average throughput value is also recorded and shown to be approximately equal to 54Mbps.



**Figure 4-27: The throughput value on PDCP level for the first simulation experiment that concerns the situation of normal traffic**

The second simulation experiment was based again on 32 users, but it describes the situation of abnormal traffic. That is, one of the users is again having HD video traffic while the remaining 31 users are assumed under attack. The traffic of the attacked users corresponds to a constant bitrate generator with a traffic of 200Kbps. This is caused by an assumed attack that such users are facing, thereby significantly increasing their generated traffic. Figure 4-28 depicts the corresponding throughput results for this “under attack” scenario, showing that the average throughput equals approximately 4Mbps.



**Figure 4-28: The throughput value on PDCP level for the second simulation experiment that concerns the situation of abnormal traffic**

As a result, the main message extracted from a direct comparison of Figure 4-27 and Figure 4-28 is that, in case of an attack scenario, the throughput value on PDCP level of the non-attacked users decreases. This is due to the additional traffic congestion that comes as an outcome of the considered security attack. For the particular example considered in this section, the observed reduction was large, reaching a level of over 90% of the initial throughput value. It should be noted, however, that the level of throughput reduction depends in principle on a variety of factors, ranging from the assumed traffic of the non-attacked user, the number of attacked users, the severity of attack, etc.

The aim of this analysis is to provide an indication of the impact of security attacks to the throughput of neighbouring, non-attacked users, and thus such factors were not treated in this analysis due to lack of sufficient project time. It is noted that such investigation on the throughput effect of security attacks was not listed in the initial planning of the WP3 activities, but rather came as an extension after the initial and planned network behaviour analysis was conducted. Further investigations on this topic are planned as part of future work activities.

## 5 Resilience and security on common infrastructure: synergies and resource allocation issues

Fault Management enables the resilience to network faults by monitoring the network performance and acting to identified network degradation. Thus, the fault management (FM) strongly relates to network resilience. Security Management is responsible for detection, reaction, and prevention from security attacks originating from malicious users of the network. Such attacks might target fetching a sensitive or confidential information to which the malicious user does not have the access. On the other hand, the security attacks may aim at creating the special conditions in the network that may lead to network failure and degradation or failure of communication service. Although both domains, security and fault management, can be seen as enablers for achieving higher network availability, the two domains are usually handled separately. In this respect, the 5G-MoNArch studies on fault and security management described beforehand in Section 3.1 and Chapter 4 have followed this approach.

Nevertheless, with the emergence of 5G networks, which broke up the monolithic implementation of network elements and introduced virtualisation concepts in the implementation of network elements, more network design flexibility became possible. Specifically, the virtualisation concept enabled also easier and more cost-effective mitigation from both network faults and security attacks. As virtualised resources are used by both domains, security and fault management, there is a need to consider them jointly. The target of such joint consideration discussed within 5G-MoNArch was previously highlighted in Section 1.2.1 and in Figure 1-3 and Figure 1-4, where the interaction between the respective architecture modules located at the Management and Orchestration layer (namely “5G Security Management” and “5G Fault Management” modules) is underlined.

The first part of this Section describes such interaction in detail, which is targeted to enable more robust network operation with minimal (resource) cost implications while fulfilling the required quality of service. In the second part, namely Section 5.2, the outcome of such joint consideration of fault and security management is put forward, with emphasis on its application to the Hamburg Smart Sea Port use case scenario.

### 5.1 Interaction between fault management and security

In mobile networks the security and fault management entities are focusing on different kinds of network problems thus have been studied independently in Section 3.1 and Chapter 4. However, the resulting effects of such problems on the network functionality might be common, e.g. unavailability of a certain network entities or even an entire service. Furthermore, as mentioned above the means for mitigation within security and fault management considering the virtualised resources may be common. In following, such and further commonalities between the fault and security management analysis are highlighted, on the basis of their deployment in 5G networks.

#### 5.1.1 Commonalities of fault and security management in 5G networks

##### *Common root causes and mitigation actions*

In many cases, a security threat might lead into problems in network functionality that will be detected by the fault management in addition to the security management. That is, a single threat (root cause) affects both domains of security and fault management. One example of such security threat is a denial of service attack, which might result in unusual KPI patterns that will be detected at fault management as a network anomaly.

Further commonalities between security and fault management can be seen in the anomaly detection procedures which in both cases generally rely on monitoring of current network performance and comparing such inputs with pre-defined normal states, i.e. profiles of the network in order to identify any anomaly in network operation.

Finally, common approaches can be applied for mitigation for identified network problems, e.g. network function re-configuration, and relying on existence of virtual replicas of affected network functions which is of particular interest in our joint security and fault management study on resource optimisation. Such virtual replicas are used for temporarily or permanently transferring the functionality of the

affected network function to another network function. In order to enable such a mitigation approach, some level of network overprovisioning/redundancy needs to be supported by network planning. Overprovisioned network functions can take over the functionality of network functions affected by either security attacks or network faults in the case of unexpected events. However, applying the overprovisioning is associated with increased costs in the network deployment as well as operational complexity.

### ***Network slicing and network function virtualisation aspects***

Virtualisation of network functions enables more efficient realisation of overprovisioning as network functions might be deployed on less expensive underlying infrastructure and more easily be replicated or migrated. However, even with applying the virtualisation a certain cost needs to be accounted when deploying redundant network functions. Thus, utilising virtualisation in order to enable redundancy in the network needs to be carefully implemented in order to enable efficient utilisation of underlying resources and minimise resource costs. In other words, the overprovisioned resources need to be shared and re-used as much as possible. Thus, the commonalities between security and fault management can be exploited for optimisation of resource usage.

Furthermore, the concept of network slicing envisions existence of multiple logical networks that are sharing a common infrastructure, thus the resources that can be used to handle security and network fault issues need to be shared among network slices. This fact emphasises further the need for joint security and fault management considerations for efficient resource utilisation and fulfilment of reliability requirements.

Different network slices might have completely different reliability requirements that may lead to different levels of security and fault resilience. In addition, the actual usage of available shared resources for security or fault management purposes needs to be in line with according security/resilience slice requirements of a slice, as well as with the overall SLAs agreed with the tenants.

## **5.1.2 Security & Resilience (S&R) cross domain / cross slice management entities**

As described in [5GM-D3.1], [5GM-D2.2], 5G-MoNArch aims at utilising the aforementioned commonalities between security and fault management in resource optimisation. In this regard, the 5G-MoNArch x-domain and x-slice S&R (Security & Resilience) Management entities perform the joint security and fault management considerations and derive the decisions on suitable resource allocation and re-allocation during the slice runtime.

In some cases, the same common resources can be used for handling events from security and fault management even across different slices, e.g., when the root cause of the event is the same. On the other hand, in order to guarantee slice-specific required level of robustness against security threats certain amount of available overprovisioned resources might need to be “foreseen/anticipated” for handling the security threats. The same applies for network fault problems and their recovery. However, the amount of overprovisioned resource needs to be minimised, and carefully provisioned based on a specific use case, e.g., slice requirements, SLAs, amount of available resources, network state and likelihood for network problems, etc. The x-domain and x-slice S&R (Security & Resilience) Management entities consider the aforementioned constraints and anticipate the amount of overprovisioned resources (within a single slice and across different slices) to be used for recovery from security threats and network faults.

## **5.1.3 Joint security and fault management considerations for resource optimisation**

The x-domain and x-slice S&R Management entities perform different actions in order to derive suitable scheme for resource allocation across domains i.e. within a single network slice as well as across different slices. Furthermore, based on the current network state the resource re-allocation can be performed once the trigger conditions are satisfied.

In order to perform the optimised resource management, the x-domain and x-slice S&R (Security & Resilience) Management need to determine the need for overprovisioned virtual resources to be used for mitigation purposes, considering the expected network fault and security issues, as well as their inter-

dependencies. As described in [5GM-D3.1] different security attacks can have different impact on the fault management, e.g. certain attacks such as denial of service can also be identified by a fault management. However, from the resource optimisation point of view the most relevant aspect is if the common virtual resources can be used for mitigation of occurred faults and security problems, such that joint resource allocation for mitigation purposes can take place.

### ***Security and fault management use case examples***

In view of the above, the x-domain and x-slice S&R Management analyses the potential fault and security issues along with the mitigation approaches to overcome them, especially focusing on the cases where additional (overprovisioned) virtual resources will be needed/used in order to perform the mitigation. With this respect the x-domain and x-slice S&R Management identifies following use cases, which are illustrated in Figure 5-1 and are mapped into specific resource requirements and corresponding allocations:

- **Use case 1:** No over-provisioned virtual resources are needed for network fault and security mitigation, i.e. when the mitigation is done by re-configuration of available resources/NFs. For example, this use case comprises the mitigation of security threats such as unusual activity by blocking the requests from certain sources after a given number of n attempts. In the case of network faults identified by fault management e.g. cell outage, the compensation may be done by reconfiguration of neighbouring cells. *It is emphasised that this use case is not relevant for joint security and FM (fault management) study on resource optimisation as it does not have impact on virtual resources.*
- **Use case 2:** Certain additional and/or overprovisioned virtual resources are needed for security mitigation of a NF, which do not correspond to having an exact/full replica of that NF. For example, for security attacks such as denial of service, a certain amount of additional resources is needed for mitigation. Nevertheless, this amount of resources does not correspond to having a full copy of the active NF instance, but it is usually used to create a “fake” copy of the active NF in order to divert the traffic towards it. *This amount of resources needs to be taken into account for estimating an overall amount of required overprovisioned resources, thus it is relevant for joint security and FM resource optimisation.*

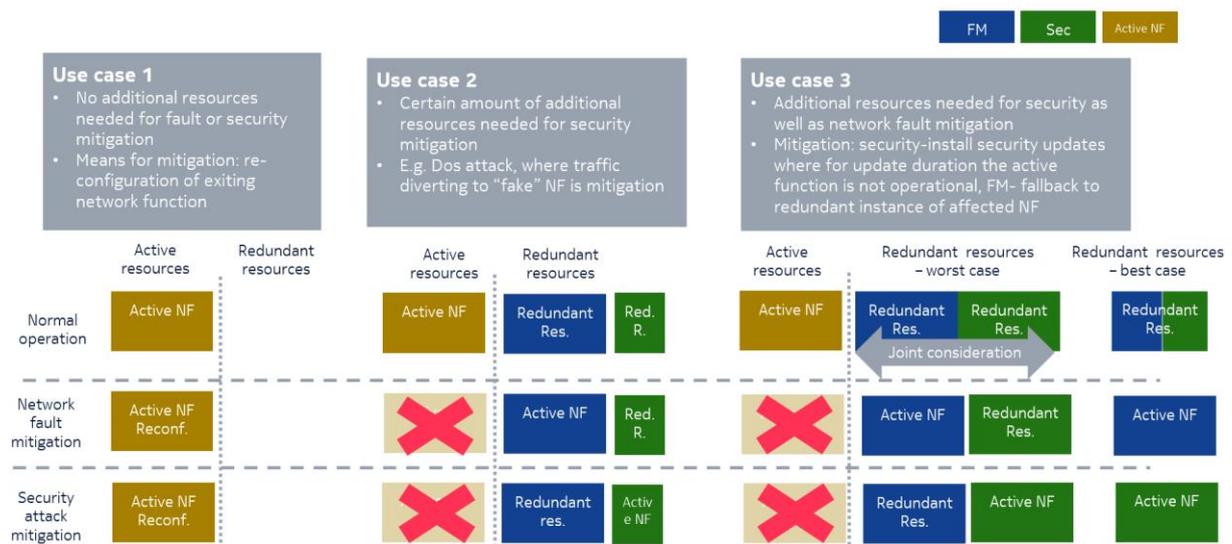
It is noted that, for handling the potentially concurrent network faults (identified by fault management), the amount of resources needed for its mitigation depends on the actual network fault, e.g. the network fault can be mitigated by re-configuration without resource requirements (e.g. as in use case 1 above) or failover using the NF replica. Figure 5-1 shows the latter case. For simplicity reasons, in the following we assume that for mitigation from network faults (identified by fault management) the mitigation always requires additional virtual resources as this is more relevant for our joint security and fault management study.

- **Use case 3:** Exact/full virtual replica of NF is needed in order to overcome the network fault or security issues. For instance, faults in VM running the NF may require standby VM that can take over the functionality of failing NF. In another example, in the case of performing required security patches, the active NF may need to be taken out of operation during certain period of time. During that time, the redundant NF needs to take over the operation.

It follows from the above analysis that in this use case both domains (security and fault management) require extensive amount of additional/overprovisioned virtual resources in order to perform the mitigation operation. Hence, *this use case is of particular interest for joint security and fault management study due to the larger amount of resources required for mitigation, thus the need and potential for resource optimisation.* In this use case the x-domain and x-slice S&R Management need to determine the trade-offs in actual resource overprovisioning and the level of resilience to network faults and security problems that can be achieved. That is, while minimising the actual amount of overprovisioned resources, the x-domain and x-slice S&R Management needs to assure the fulfilment of slice requirements with respect to resilience.

Use case 3 shows how different amount or overprovisioned resources may be allocated. In this use case resource pre-provisioning can be either 100% or 200% compared to active (currently used resources)

for the purpose of mitigation from FM and security problems. As provisioning of 200% more resources is very expensive, 100% seems as more suitable approach, especially if the temporal unavailability of redundant resources is acceptable from FM or security point of view for a given network slice and network context. Furthermore, as some of security attacks, such as malware or data and device tampering may have high impact to resilience and will be concurrently identified by fault management, using a common resources for mitigation (accounting for 100% over-provisioning) may be more suitable approach (see Table 5-2). The exact amount of overprovisioned resources is computed by the x-domain and x-slice S&R Management based on slice requirements, network context, likelihood of problem appearance, inter-dependencies between security and fault management from resource and impact point of view (see Table 5-2) as well as the tolerance to fault and security problems which are defined through resilience requirements. This amount and allocation of resources can be changed during network slice runtime.



**Figure 5-1: use cases derived and considered by x-domain and x-slice S&R management in virtual resource allocation and optimisation**

Based on such information, the x-domain and x-slice S&R Management can allocate certain amount of resources [5GM-D2.2], [5GM-D3.1]. This can include, for instance, 100% overprovisioning for a certain NF for use case 3, which are dedicated for either security or FM issues. Consider, for example, that during the run-time of a slice the considered NF experiences substantially higher number and severity of security attacks, such that very frequent security patches need to be installed. Then, using the redundant resources mainly for mitigating security problems and occasionally lacking the resources to address the network faults, the x-domain S&R Management will opt to:

- 1) re-allocate a portion of redundant resources from other subnets of the same network slice, where the redundant resources were underutilised, i.e. the initial resource allocation was higher than the actual current need.

If the corrective action from 1) is not possible due to the current network state where the number and severity of network and security problems correspond to the actual resource allocation/over-provisioning, the x-domain S&R Management will opt to:

- 2) request from x-slice S&R Management additional resources that are currently over-provisioned for other slices but may be unutilised. Based on the network states and agreed SLAs across slices the x-slice S&R Management may temporarily or permanently grant a request from x-domain S&R Management.

The resource re-allocation procedure described above including the cause and the outcome will be recorded and used by x-domain and x-slice S&R Management for future resource allocations/over-provisioning within and across slices.



- within a single network slice
- or across different network slices.

### Learning phase:

- The system is able to learn from resource allocation and prioritisation during the operation. Optionally, such information is provided to other network slice management functions for resource provisioning optimisation.

### Message exchanging sequence

The aforementioned resource allocation process involves a specific signalling exchange between the involved network entities. This signalling exchange process is illustrated in Figure 5-3, highlighting the architecture elements depicted in Figure 1-3 that are involved in the joint fault management and security management study.

The main steps of the message exchanging process are summarised as follows.

- 1) The slice requirements are derived at the CSMF. This involves acquiring the respective resilience and security requirements from the slice tenant. Such requirements are typically used to derive respective SLAs, which are established between the tenant and the infrastructure or service provider.
- 2) NSMF analyses the probability of occurrence of network fault issues, as well as security issues. Moreover, NSMF derives the amount of virtual resources needed for fulfilling the given slice requirements.
- 3) NSMF, NFV MANO and VIM network entities exchange information which include orchestration request in order to deploy required network functions on required resource as well as respective feedback if such deployment is feasible. The actual enforcement of such orchestration requests and actual network function deployments are enforced via the interface X (Itf-X), specified in the WP2 framework of 5G-MoNArch and reported in [5GM-D2.3].
- 4) The initial resource allocation process is iterated, leading to respective resource re-allocation steps, both at an intra-slice and inter-slice level, leading to corresponding re-orchestration requests and feedback.

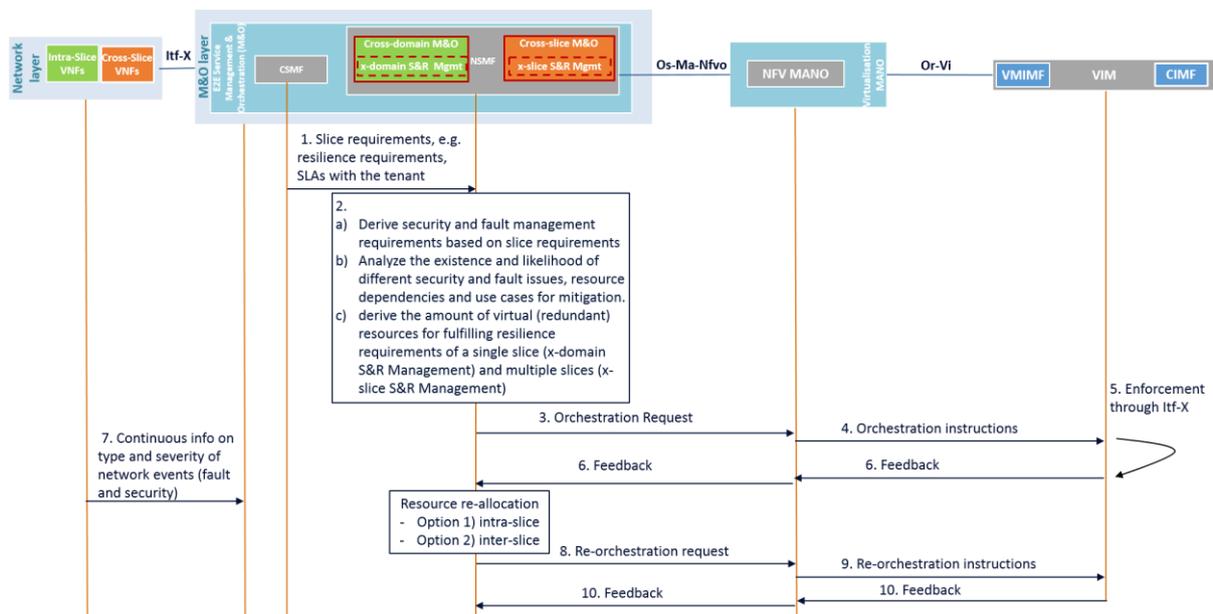


Figure 5-3: Message sequence chart for joint fault management and security management

## 5.2 *The Hamburg Smart Sea Port use case scenario: the effect of security threats to resources*

This section complements the initial threat analysis reported in [5GM-D3.1] with a deeper evaluation of the role of the main components of an STZ in terms of means of detection, potential reactions and impact on resilience. This analysis adds an evaluation of some of the most representative threats in the context of the Smart Sea Port testbed in Hamburg, comprising threats from different categories, summarised in Table 5-1. While it can not be considered a comprehensive analysis of all the potential incidents threatening the Smart Sea Port infrastructure, it provides with a good approximation about how to handle such incidents and what is the likelihood of dealing with them. In fact, this analysis meets with one of the main recommendations published by the European Commission regarding cybersecurity aspects in 5G Networks, which stresses the importance of defining mitigations and impact of security threats over an 5G infrastructure [ECSEC19].

**Table 5-1: Incidents considered in the study**

Attack	Internal/External	Type
Unusual activity	Internal	Application Level
Denial of Service	External	Network level
Slow DDoS	External	Network level
DoS in wireless spectrum	External	Network level
Privilege escalation	Internal	Application level
Botnets	Internal	Application level
Service Discovery	External	Network level
Data and device tampering	External	Physical integrity
SQL injection	External	Application level
Malware	External	Application level

Table 5-2 details the ten attacks included in Table 5-1 in the context of a 5G network. For every attack it is described the ways to detect it (such as the security probes required to detect it), the possible mitigation actions to react to such attack and the type of resources needed to enforce the mitigation. It is also evaluated the impact when mitigating, in terms of effect on the infrastructure (for instance, the time required to enforce it, the computation resources required, etc.).

A similar exercise is done with the impact on operation of a network (or network functionality) identified by the network fault management, referred also as network resilience in this deliverable. As described in [5GM-D3.1] and in Section 5.1 of this document, there is a close relationship between security management operation and resilience requirements. The impact in network resilience is not derived just from the incident detected, but also derived from the potential mitigation actions. Such evaluation is also included in the table below for every security attack assessed.

Table 5-2 provides concrete examples on different security attacks with different requirements on the (virtual) resources for mitigation purposes. As highlighted in Section 5.1 such security attacks are of special interest for joint resource optimisation at security and fault management. E.g., DoS attack complies with the use case 2 described in Section 5.1, whereas data tampering and insertion of USB devices comply with use case 3. Such classification of security attacks is performed by x-domain and x-slice S&R Management as part of input analysis. Furthermore, the likelihood of their appearance in certain context, such as in sea port, as well as their impact to the functionality of a certain network function (as described in Table 5-1) are other inputs needed by the x-domain and x-slice S&R Management in order to derive the resource requirements for mitigation purpose. E.g. for the use case 3 if there is high impact of security attack to network resilience/fault management it is more likely that only one virtual replica of the network function can be used to jointly mitigate security and network fault problem.

**Table 5-2: Example of attacks analysis, mitigations, impact on resilience and on the seaport infrastructure**

Attack	Description	Detection	Mitigation	Resources required to mitigate	Impact when Mitigating	Impact on Resilience/Network Fault Management	Likelihood in Sea Port Testbed
Unusual activity	Generic category including anomalous activities such as many login attempts	In Linux based machines Linux Pluggable Authentication Modules (PAM) can report authentication attempts. HSM based devices can also report about unauthorised activities.	Block requests from certain sources after n unsuccessful attempts	Capability to remotely perform actions against devices (i.e., SDN/NFV capabilities such as OpenContrail) using protocols such as Netconf or openflow)	Low: Easy to deploy rules for blocking requests, with no real impact on the infrastructure. False positives might be considered when blocking requests	High: might cause a change in performance of affected network function, e.g. due to overload	High: being exposed to public networks, many devices are exposed to this type of attacks in the Seaport (traffic lights, VR-devices, environmental sensors)
Denial of Service	Flood devices with packages, exhausting them and affecting their normal operation, resulting in lower performance or decreased availability	NIDS sensors can detect flood attacks by analysing network traffic	Different possibilities: (1) Create firewall rules to redirect malicious traffic. (2) Instantiate virtual replica of attacked infrastructure to redirect and isolate attack.	Capability to remotely modify firewall rules or to instantiate virtual devices through NFV	Depending on the mitigation: (1) Firewall rules: Low. Simple, not affecting to the current infrastructure. (2) Virtual replica: Medium. Time to deploy virtual replica might take time. However, it allows for the isolation of the attack and further study and forensics	High: might cause a change in performance or even a failure of a certain network function or multiple network functions running on affected machine	High: DoS are common attacks in all domains, and very likely affecting 5G infrastructure as well

Slow DDoS	Send small packages spaced in time, occupying connections slots and not releasing them, and blocking the attacked device that is not able to open additional connections	Logging requests and checking request headers tags and timeout values	If vulnerable to these attacks, a simple modification to the server configuration helps to mitigate this attack	Capability to remotely change the configuration of the server	Low: A simple modification and restart of the server is required	High: might cause a change in performance of affected network function	Medium: Although, in principle, SlowDoS attacks have the same effect than DoS attacks, this type of attacks are less probable in infrastructure with high capacity in terms of resources. However, still mobile phone and personal devices with limited resources (e.g., IoT devices) can be exposed to this attack.
DoS in wireless spectrum	Alter wireless spectrum provoking interference in certain frequencies (e.g., jamming attacks)	Specific antijamming hardware devices are required to detect these incidents	Change devices to connect through different frequencies. Difficult to react as long as the physical source of the attack is not located. Current research tries to react to these attacks by reconstructing the jammer signal to mitigate the interference	Capability to remotely change the frequency that wireless devices uses to operate. However, considered that the spectrum might be not available this is quite difficult. In case of signal based mitigation it is required the capability of deploy and start the device that builds and emits the signal	High: It might be required to modify the configuration of many devices, many of them might not be accessible or easily reconfigurable (for example very resource constrained ones).	High: might cause a change in performance of affected network function	High: Similar to DoS, these types of attacks are targeting wireless devices. Considering that most of the devices deployed at a 5G infrastructure are using the wireless spectrum (such as hand-held devices) the exposition to this type of attacks is high

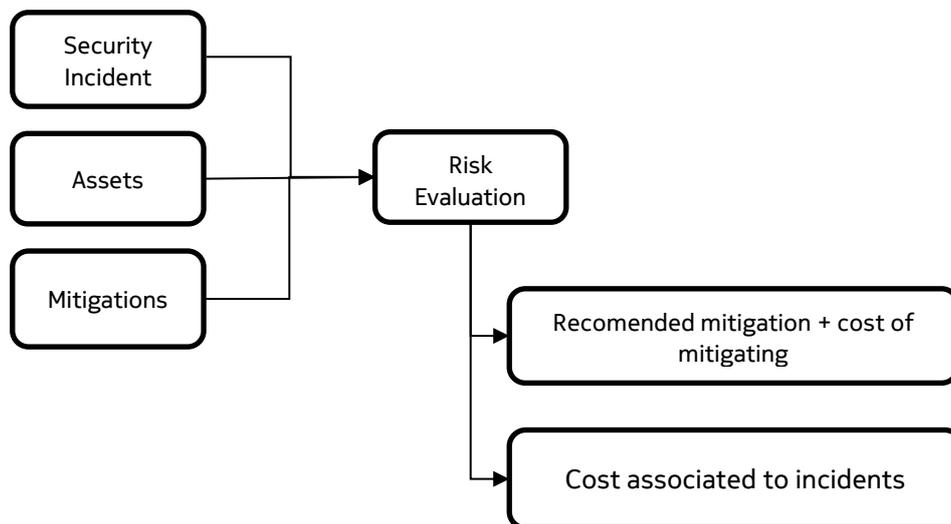
<p>Privilege escalation</p>	<p>Gain access to privileges to which the user is not entitled. This is done possibly by performing kernel level operation. Two types: Vertical: lower privilege user or application accesses user or content reserved for higher users or applications (e.g. become the administrator). Horizontal: normal user accesses functions or contents reserved for other normal users (e.g. access information from another user of the same category)</p>	<p>FIM – File Integrity Monitoring can be used to detect system changes.  HIDS - Host-based Intrusion Detection Systems (e.g. OSSEC) can monitor users’ activities within a host</p>	<p>Change file permissions and user privileges</p>	<p>Capability to remotely perform actions against devices (i.e., SDN/NFV capabilities such as OpenContrail)</p>	<p>Low: Simple modification of some system permissions would be required.</p>	<p>Low: privilege escalation per se should not impact the resilience as long as the malicious user (using the new privileges) does not deliberately cause the failure of network functions or hosts</p>	<p>Low: It is expected for a 5G infrastructures to have a robust configuration of permissions and privileges. It is unlikely that outsiders are capable of exploiting this threat. Insider attacks with privileges would be capable of exploiting it, although, in general, insider attacks have a low probability to happen</p>
<p>Botnets</p>	<p>Infected machines that perform attacks under the control of a master</p>	<p>Usage of IDS to Identify machines belonging to client’s infrastructure belonging to a botnet</p>	<p>Block unsolicited inbound traffic at firewalls</p>	<p>Capability to remotely modify firewall rules</p>	<p>Low: A simple modification of a firewall is required</p>	<p>High: might impact the functionality of switch/router.</p>	<p>Low: Similar to privilege escalation, this threat is more probable by insiders rather than outsiders. Therefore, it remains with low probability to happen</p>

Service Discovery	Attempt to discover running services using port scanning and ARP requests	NIDS sensors detect network scanning by analysing network traffic	In principle it is not possible to mitigate these incidents. However, there are two possibilities: (1) Scanning from outside. Scans can be mitigated by closing access to any port (2) Scanning from inside. Not possible to be detected. However, MAC filtering can be used to allow access to the network just to authorised MAC addresses	Capability to remotely modify firewall rules	Medium: Although this type of incidents are in principle harmless (as this attack is just detecting services and not attacking them), it can be considered as Medium impact in case critical services are detected, discovering potential vulnerabilities and exploiting it	Medium: hacker might use learned vulnerabilities to deliberately cause failure of network function or host/infrastructure	High: Service discovery by exploiting port scanning is very common in all domains, being very often the first action that an attacker performs prior to a more sophisticated attack.
Data and device tampering	Device manipulation to modify, either physically (e.g., device destruction) or logically (e.g., upload and deploy unauthorised applications)	Once gained access to the system the detection of data manipulation is difficult. Data tampering can be detected using forensic analysis methodologies, which can provide with an estimation of an attack with a certain probability (e.g., detecting installation of new applications out of scheduled	Once gained access to upload and deploy applications to the server, the mitigation is difficult. Rather than mitigation, data tampering can be prevented by updating and patching potential vulnerabilities, checking and limit privileges for executing applications or	Capability to remotely perform actions against devices (i.e., SDN/NFV capabilities to turn off devices or uninstall packages)	High: Given the difficulty to mitigate the impact is high as it might entail to roll back to the situation before the modification of the data (restoring backups), checking permissions and patching software or restoring	High: deployment of new application may cause failure of network function or host/infrastructure	High: Considering that many devices deployed in the seaport infrastructure are in public areas, the probability of manipulation is quite high.

		<p>maintenance time frames). To this end, NIDS detectors can alert about installation of software in certain machines, such as Linux based machines that use synaptic repositories.</p>	<p>installing new ones. Device tampering can just be mitigated with physical sensors (e.g., turning off device when the manipulation is detected)</p>		<p>physically the device</p>		
<p>SQL injection</p>	<p>Inclusion of SQL malicious code in an entry field when communicating with an application</p>	<p>NIDS sensors can alert about SQLi attempts</p>	<p>SQL injection success in case of bad designed servers and databases structures. SQL injection can be mitigated by detecting the flaw exploited and patching the server attacked with more secure ways to access to the database (for example, accepting just parameterised queries)</p>	<p>Capability to remotely change the configuration of the server</p>	<p>M: It requires patching a server, which might entail its reboot when done, impacting on the service availability</p>	<p>M: unless the inclusion may cause failure of network function or host/infrastructure the impact of this attack should be low</p>	<p>Medium: SQL injection attacks are quite common in all domains. In the case of the seaport, it would depend on the correct isolation of the databases from external connections. Although in general it should be low, it is possible that attackers are capable to inject SQL attacks accessing from an authorised device or intercepting requests combining a man in the middle attack.</p>

Malware	Infect devices with malware, e.g., insertion of infected USB devices as a source of attacks, enabling exploit of SCADA systems and other malicious activities against assets, such as stuxnet, a worm designed to attack industrial networks	Antivirus and antimalware detectors	Detected malware should be automatically detected and removed by antivirus and antimalware tools. Devices that have been successfully infected must be isolated or even turned off till the threat has been controlled to prevent propagation	Capability to remotely perform actions against devices to install/configure/update anti malware protection	Low, Medium: Depends on the detection. If the threat has not been detected by an antimalware the impact increases as it requires to isolate (disconnecting from network or turning off) affected devices	High: might impact the functionality of network functions and infrastructure	High: In all 5G infrastructures there are elements that contains physical interfaces (such as USB). This include computers in control rooms, personal devices (computers, tablets, mobile phones), etc. Therefore, tt is very likely that, either deliberately or not, these devices are exposed to these malicious events
---------	--	-------------------------------------	---	--	--	--	--

It is also worth noticing the potential economic impact of mitigations of security incidents. In general terms, the enforcement of mitigation actions entails certain cost. No matter if it is a cost related to computational or human resources or time. There is always an economic impact associated to them. To this end, the economic impact is two-fold. On the one side, there is a cost associated to the additional resources required to mitigate an incident (for example, to deploy a new database that replaces a compromised one). On the other side there is a cost associated to the impact on the infrastructure. Incidents compromising any of the assets of the infrastructure derive in costs associated to the time that the assets are not working (for instance, number of transactions lost). Security threat reaction components need to reason about what is the best possible mitigation for a given incident. The aforementioned costs associated to the impact on the assets and the cost of enforcing a mitigation are required for such evaluation. There are risk assessment engines [WS-D5.2] capable of providing such analysis, by using as input the incident detected by the security threat detectors (for example, the severity of the incident), information about the assets affected (including their criticality within the concerned infrastructure), and the list of possible mitigations (including the cost associated to it). A risk evaluation engine, typically consisting of statistical models and decision support evaluators [WS-D5.2], uses such information as input to estimate the cost associated to the incident, for example to indicate how much money would be lost if the incident detected over an affected asset is not mitigated, c.f. Figure 5-4. Additionally, the risk evaluation engines also provide with the most convenient mitigation with considers the list of available mitigations (and its cost) and the cost associated to the incident.



**Figure 5-4: Process for evaluating incidents and estimate the most convenient reaction**

In general terms, incident reaction mechanisms very deeply depend on the type of infrastructure and on the ways to interact with it. An example of a process for mitigating incidents is the one followed by the ANASTACIA project (<http://www.anastacia-h2020.eu/>), which is focused on the detection and mitigation of incidents in IoT infrastructures leveraged by the use of SDN/NFV interfaces.

The simplified process is depicted in Figure 5-5. The underlying infrastructure exposes SDN/NFV interfaces that allows to interact with it. Among the activities that these interfaces can carry out there are deployment of security enablers such as detectors, and security capabilities such as firewalls or honeypots. The infrastructure exposes the available mitigations which depends on the security enablers and capabilities deployed, which also determines the security policy that the infrastructure is enforcing. Whenever a mitigation is triggered, the security policy allows to identify the security capabilities to use, which are invoked through the SDN/NFV interfaces exposed by the infrastructure. A similar process can be used in the Security Trust Zone approach described in 5G-MoNArch. To this end, the available resources and interfaces to interact with when enforcing reactions to incidents are known by the MANO layer. A security policy can be defined for every STZ, which includes the list of available actions to be performed by the MANO in the infrastructure using the available NFV interfaces.

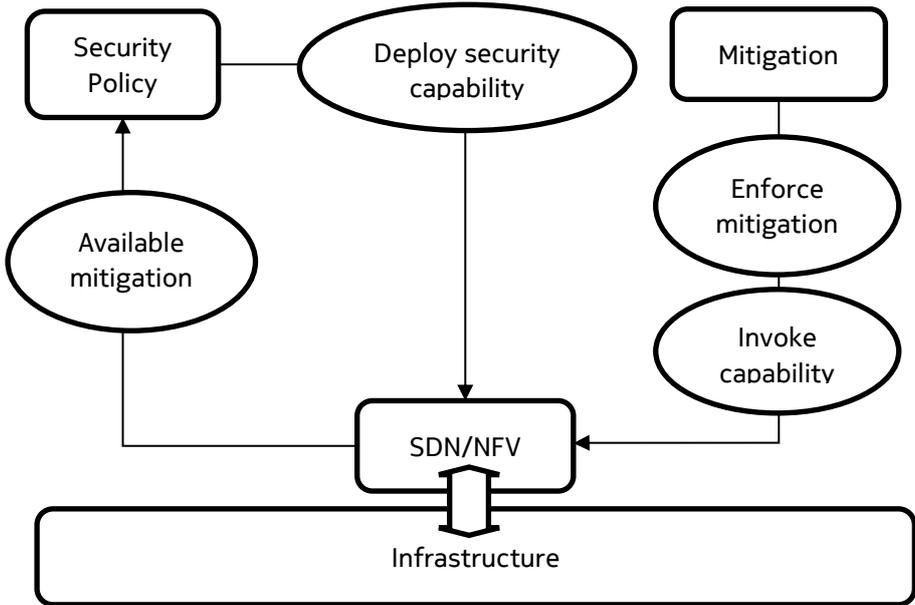


Figure 5-5: General process for mitigating security incidents

## 6 Summary

This deliverable provides a consolidated view on the developments on resilience and security carried out in the months M12-M21 of the 5G-MoNArch project. Such developments have mainly concentrated on conducting an evaluation assessment of the resilience and security concepts developed in the project's first year and reported in the public deliverable associated with the WP3 framework [5GM-D3.1]. In addition to such evaluation analysis, new concepts have been proposed, complementing the enablers proposed in [5GM-D3.1], and followed by a brief evaluation analysis.

In a close-up view, this deliverable contains developments in the three major topics of the WP3 framework, namely the RAN reliability; telco cloud resilience; and security. Chapter 2 provides an evaluation campaign of the former, including simulation results on data duplication approaches spanning across different configuration assumptions and focusing on major KPIs associated with reliability in mission-critical applications. In addition to data duplication, the considered network coding approaches have been studied by means of a simulation framework, addressing specific integration issues and followed by numerical results. Moreover, the analysis has been complemented by a hybrid approach, which combines the benefits of data duplication and network coding into a common scheme which is able to switch between the two schemes aiming at higher performance with given requirements on reliability and latency. It is noted that the concepts of Chapter 2 have led to idea protection by means of patents, scientific publications in major international conferences (such as the EU-sponsored flagship conference on communications and networking – EUCNC, and IEEE ICC), as well as respective technical contribution to standardisation fora such as 3GPP.

Topics related to telco cloud resilience are treated in Chapter 3. This included an extensive root-cause identification of network failures by means of event correlation techniques in slicing environments, followed by an analysis of redundancy methods and their potential in increasing the service availability. Chapter 3 also provides an elaborated view of controller scalability approaches, shedding light onto the performance of scalability solutions by studying them as part of a general scalable controller framework. Furthermore, the concept of 5G Islands, originally proposed in [5GM-D3.1], has been elaborated and evaluated in terms of its ability to mitigate the VNF migration cost while taking into account the corresponding loss with respect to outage probability. The results pertaining to all such analysis have been disseminated in international conferences such as EUCNC by means of scientific articles, as well as respective demonstration campaigns.

Along with the aforementioned evaluation framework for the WP3 enablers on RAN reliability and telco cloud resilience, this deliverable also presents an analysis of the effect of security threats on the main 5G network components in Chapter 4. Focusing on the 5G-specific aspects of the considered network deployment and with reference to the Smart Sea Port testbed, Chapter 4 provides a security analysis that spans all main components of 5G networks, including devices, network elements, and network slicing-specific elements. Together with the 5G threat analysis, the concept of security trust zones, proposed in [5GM-D3.1], has been further elaborated and assessed by means of a simulation campaign. The simulation campaign imitates attacks against security trust zones, thereby assessing the ability of trust zones to detect such attacks and protect the critical network components. Furthermore, in the framework of the security simulation, a graph-based anomaly detection method is presented in Chapter 4, followed by an extension that is based upon machine learning approaches. Chapter 4 furthermore provides evaluation details pertaining to both such scenarios. The above analyses have led to scientific results which have been published in major international conferences such as IEEE ICC and IEEE WCNC.

Besides the individual evaluation activities with respect to corresponding research domains of the WP3 framework, provided in Chapters 2-4, this deliverable comprises a joint study that has identified synergies and common virtual resource allocation issues between resilience and security. Specifically, the interaction between fault and security management has been investigated in a telco cloud environment involving common virtual resources and deployed in a redundancy form. This investigation has jointly considered the optimised handling of available resources as well as the relevant resource provisioning. In addition, the joint resilience and security study has considered the impact of security threats to resources, as well as its impact on the network functionality reflected through the network fault management. In this respect, the effect of ineffective security mitigation to the resource availability for network resilience purposes has been discussed.

Finally, besides the WP-specific evaluation mechanisms presented in Chapters 2-5, this deliverable addresses project-wide aspects of the considered resilience and security enablers. This is treated in Chapter 1, where certain architectural issues related with the implementation of the developed enablers are discussed. In addition to architectural aspects, aspects that are related to a project-wide evaluation of the proposed WP3 enablers are also put forward, highlighting their performance in large-scale evaluation scenarios. Both these aspects, which are related to a project-wide architecture implementation and evaluation, underline the relation of WP3 with WP2 (Overall Architecture) and WP6 (Verification and Validation) within the 5G-MoNArch framework, sketching thus the respective interworking between such work packages.

## 7 References

- [3GPP 22.261] 3GPP TS 22.261, “Technical Specification Group Services and System Aspects; Service requirements for the 5G system; Stage 1; (Release 16),” Dec 2018
- [3GPP 33.501] 3GPP TS 33.501, “Technical Specification Group Services and System Aspects; Security architecture and procedures for 5G system (Release 15),” v15.2.0, Dec 2018
- [3GPP 36.890] 3GPP TR 36.90, “Study on Single-cell Point-to-multipoint transmission for E\_UTRA (Release 13),” July 2015.
- [3GPP 38.211] 3GPP TS 38.211, “Technical Specification Group Radio Access Network; NR; Physical channels and modulation (Release 15),” v15.4.0, Dec 2018
- [3GPP 38.300] 3GPP TS 38.300, “Technical Specification Group Radio Access Network; NR; NR and NG-RAN Overall Description; Stage 2 (Release 15),” v15.4.0, Dec 2018
- [3GPP 38.470] 3GPP TS 38.470, “Technical Specification Group Radio Access Network; NG-RAN; F1 general aspects and principles (Release 15),” v15.4.0, Dec 2018
- [3GPP 38.801] 3GPP TR 38.801, “Study on new radio access technology: Radio access architecture and interfaces (Release 14),” March 2017
- [3GPP 38.901] 3GPP TR 38.901, “Study on channel model for frequencies from 0.5 to 100 GHz (Release 15),” June 2018
- [3GPP 38.913] 3GPP TR 38.913, “Study on Scenarios and Requirements for Next Generation Access Technologies; (Release 15),” June 2018
- [5GM-D2.2] 5G-MoNArch project, deliverable D2.2, “Initial overall architecture and concepts for enabling innovations,” June 2018
- [5GM-D2.3] 5G-MoNArch project, deliverable D2.3, “Final overall architecture,” April 2019
- [5GM-D3.1] 5G-MoNArch project, deliverable D3.1, “Initial resilience and security analysis,” June 2018
- [A18] A. Aijaz, “Packet Duplication in Dual Connectivity Enabled 5G Wireless Networks: Overview and Challenges,” Arxiv document, [Online] available: <https://arxiv.org/pdf/1804.01058.pdf>
- [AD13] T. Alexandrov, and A. Dimov, “Software availability in the cloud,” in proc. 14th ACM International Conference on Computer Systems and Technologies, 2013pp 193–200
- [AMF16] Availability Management Framework, [Online] available: <http://devel.opensaf.org/SAI-AISAMF-B.04.01.AL.pdf>
- [AVA18] The Availability Digest Article Archive, [Online] available: <http://www.availabilitydigest.com/articles.htm>
- [AVA19] The Availability Digest, “Mission-Critical Network Planning,” September 2009, [Online] available: [http://www.availabilitydigest.com/public\\_articles/0409/mission\\_critical\\_network\\_planning.pdf](http://www.availabilitydigest.com/public_articles/0409/mission_critical_network_planning.pdf)
- [ATK15] L. Akoglu, H. Tong, and D. Koutra, “Graph based anomaly detection and description: a survey,” Data mining and knowledge discovery, 29(3), 626-688, 2013
- [B94] C. M. Bishop, “Mixture density networks,” Technical report NCRG/4288, Aston University, Birmingham, UK, 1994
- [BAE17] M. M. Baig, M. M. Awais, and E. S. M. El-Alfy, “A multiclass cascade of artificial neural network for network intrusion detection,” Journal of Intelligent & Fuzzy Systems, 32(4), 2875-2883, 2017
- [C06] Jung-Fu Cheng, “Coding performance of hybrid ARQ schemes,” IEEE Transactions on Communications vol. 54, no. 6, pp 1017-1029, June 2006
- [C18] F. Catak, “Two-layer malicious network flow detection system with sparse linear model-based feature selection,” Journal of the National Science Foundation of Sri Lanka, 46(4), 2018

- [CEL10] CELTIC WINNER+ project deliverable D5.3, “WINNER+ Final Channel Models,” CELTIC/CP5-026 D5.3 V1.0, June 2010.
- [DUN73] Dunn, J. C., “A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters,” *Journal of Cybernetics*. 3 (3): 32–57, September 1973, [doi:10.1080/01969727308546046](https://doi.org/10.1080/01969727308546046)
- [ECSEC19] European Commission, “Cybersecurity of 5G networks,” Recommendation, Version: C, 2335, March 2019. [Online] available: [https://ec.europa.eu/newsroom/dae/document.cfm?doc\\_id=58154](https://ec.europa.eu/newsroom/dae/document.cfm?doc_id=58154).
- [FGS+18] D. Felsch, M. Grothe, J. Schwenk, A. Czubak, and M. Szymanek, “The Dangers of Key Reuse: Practical Attacks on IPsec IKE,” 27<sup>th</sup> USENIX Security Symposium, Baltimore, MD, USA, Aug. 2018
- [GB18] J. Gera, and B. P. Battula, “Detection of spoofed and non-spoofed DDoS attacks and discriminating them from flash crowds,” *EURASIP Journal on Information Security*, July 2018.
- [GEM18] Gemalto, “IoT Secure Manufacturing,” [Online] available: <https://safenet.gemalto.com/data-protection/iot-secure-manufacturing/>
- [GQT+16] K. Gai, M. Qiu, L. Tao, and Y. Zhu, “Intrusion detection techniques for mobile cloud computing in heterogeneous 5G,” *Security and Communication Networks*, pp. 3049-3058, Sep. 2016
- [GT09] L. Georgiadis, and L. Tassiulas. “Broadcast erasure channel with feedback-capacity and algorithms,” *IEEE Workshop on Network Coding, Theory, and Applications, NetCod*, 2009.
- [GXT10] X. Gao, B. Xiao, D. Tao, and X. Li, “A survey of graph edit distance,” *Pattern Analysis and applications*, vol. 13, no. 1, pp.113-129, Feb. 2010
- [HJB16] A. Hilt, G. Jaro, and I. Bakos, “Availability Prediction of Telecommunication Application Servers Deployed on Cloud,” *Periodica Polytechnica, Electrical Engineering and Computer Sciences*, vol. 60, no. 1, 72-81, March 2016.
- [K16] R. Kazemi, “Entropy of weighted graphs with the degree-based topological indices as weights,” *MATCH Communications in Mathematical and in Computer Chemistry*, no. 76, 69-80, 2016
- [KDT15] I. Kalamaras, A. Drosou, and D. Tzovaras. “A multi-objective clustering approach for the detection of abnormal behaviors in mobile networks,” in *proc. IEEE International Conference on Communication Workshop (ICCW)*, pp. 1491-1496, 2015.
- [KPR+11] D. Koutra, A. Parikh, A. Ramdas, and J. Xiang, “Algorithms for graph similarity and subgraph matching,” *Ecological Inference Conference*, 2011
- [LR13] L. Livi and A. Rizzi, “The graph matching problem,” *Springer Pattern Analysis and Applications*, vol. 16, no. 3, pp. 253-283, Aug. 2013
- [MDA+08] A. Mademlis, P. Daras, A. Axenopoulos, D. Tzovaras, and M. G. Strintzis, “Combining topological and geometrical features for global and partial 3-D shape retrieval,” *IEEE Transactions on Multimedia*, vol. 10, no. 5, pp. 819-831, Aug. 2008
- [MGC+18] L. F. Maimó, Á. L. P. Gómez, , F. J. G. Clemente, M. G. Pérez, and G. M. Pérez, “A self-adaptive deep learning-based system for anomaly detection in 5G networks,” *IEEE Access*, no. 6, pp. 7700-7712, Feb. 2018
- [MS15] N. Moustafa, and J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” in *proc. IEEE Military Communications and Information Systems Conference (MilCIS)*, pp. 1-6, Nov. 2015
- [MS16] N. Moustafa, and J. Slay, “The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set,” *Information Security Journal: A Global Perspective*, 25(1-3), 18-31, Jan. 2016

- [NGMN18] Next Generation Mobile Networks Alliance, “5G Extreme Requirements: Radio Access Network Solutions,” 2018, [Online] available <https://www.ngmn.org/publications/technical-deliverables.html>
- [NIST18] NIST Issues, “First Call for ‘Lightweight Cryptography’ to Protect Small Electronics,” [Online] available: <https://www.nist.gov/news-events/news/2018/04/nist-issues-first-call-lightweight-cryptography-protect-small-electronics>
- [NWCS18] Sensitive data exposure via WiFi broadcasts in Android OS [CVE-2018-9489] [Online]: <https://www.nightwatchcybersecurity.com/2018/08/29/sensitive-data-exposure-via-wifi-broadcasts-in-android-os-cve-2018-9489/>
- [PDK+18] S. Papadopoulos, A. Drosou, I. Kalamaras, and D. Tzouvaras, “Behavioural Network Traffic Analytics for Securing 5G Networks,” IEEE International Conference on Communications Workshops (ICC Workshops), May 2018
- [PDT16] S. Papadopoulos, A. Drosou, and D. Tzouvaras, “A novel graph-based descriptor for the detection of billing-related anomalies in cellular mobile networks,” IEEE Transactions on Mobile Computing, vol. 15, no. 11, pp. 2655-2668, Jan. 2016
- [PPM18] G. Pocovi, K. I. Pedersen, and P. Mogensen, “Joint Link Adaptation and Scheduling for 5G Ultra-Reliable Low-Latency Communications,” IEEE Access, no. 6, pp. 28912–28922, May 2018
- [Raft18] Raft Github. [Online] available: <https://raft.github.io/>
- [RV18] J. Rao and S. Vrzic, “Packet duplication for URLLC in 5G dual connectivity architecture,” IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, 2018
- [RZ11] S. Rahimi, and M. Zargham, “Analysis of the security of VPN configurations in industrial control environments,” ELSEVIER International Journal of Critical Infrastructure Protection, vol. 5, no. 1, March 2012
- [TB11] O. Trullols-Cruces, J. M. Barcelo-Ordinas, and M. Fiore, “Exact decoding probability under random linear network coding,” IEEE Communications Letters, vol. 15, no. 1, pp. 67–69, Jan. 2011
- [TE18] T. A. Tchakoucht, and M. Ezziyyani, “Multilayered Echo-State Machine: A Novel architecture for efficient intrusion detection,” IEEE Access, no. 6, pp. 72458-72468, Nov. 2018
- [TYZ+11] T. Thapngam, S. Yu, W. Zhou, and G. Beliakov, “Discriminating DDoS attack traffic from flash crowd through packet arrival patterns,” IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2011
- [SB12] M. Salem and U. Buehler, “Mining techniques in network security to enhance intrusion detection systems,” International Journal of Network Security and operations, Dec. 2012
- [SI12] X. Song, and O. İşcan. “Network coding for the broadcast Rayleigh fading channel with feedback,” IEEE International Symposium on Information Theory Proceedings (ISIT), 2012
- [SWD+18] J. Sachs, G. Wikstrom, T. Dudda, R. Baldemair, and K. Kittichokechai, “5G Radio Network Design for Ultra-Reliable Low-Latency Communication,” IEEE Network, vol. 32, no. 2, pp. 24–31, March 2018
- [WM05] I. H. Witten and E. F. D. Mining, “Practical machine learning tools and techniques,” The Morgan Kaufmann Series in Data Management Systems, San Francisco, 2015
- [WMW05] A. Willig, K. Matheus, and A. Wolisz, “Wireless Technology in Industrial Networks,” Proceedings of the IEEE, 93(6), 1130–1151, June 2005
- [WS-D5.2] H2020 WISER project, Deliverable 5.2, “Wiser Real-time assessment infrastructure,” July 2016

## Appendix A

### *UNSW-NB15 features and description*

	Feature Name	Description
	Proto	Transaction protocol
	state	Indicates to the state and its dependent protocol, e.g. ACC, CLO, CON, ECO, ECR, FIN, INT, MAS, PAR, REQ, RST, TST, TXD, URH, URN, and (-) (if not used state)
	dur	Record total duration
	sbytes	Source to destination transaction bytes
	dbytes	Destination to source transaction bytes
	sttl	Source to destination time to live value
	dttl	Destination to source time to live value
	sloss	Source packets retransmitted or dropped
	dloss	Destination packets retransmitted or dropped
	service	http, ftp, smtp, ssh, dns, ftp-data ,irc and (-) if not much used service
	Sload	Source bits per second
	Dload	Destination bits per second
	Spkts	Source to destination packet count
	Dpkts	Destination to source packet count
	swin	Source TCP window advertisement value
	dwin	Destination TCP window advertisement value
	stcpb	Source TCP base sequence number
	dtcpb	Destination TCP base sequence number
	smeansz	Mean of the row packet size transmitted by the src
	dmeansz	Mean of the row packet size transmitted by the dst
	trans_depth	Represents the pipelined depth into the connection of http request/response transaction
	res_bdy_len	Actual uncompressed content size of the data transferred from the server's http service.
	Sjit	Source jitter (mSec)
	Djit	Destination jitter (mSec)
	Stime	record start time
	Ltime	record last time
	Sintpkt	Source interpacket arrival time (mSec)
	Dintpkt	Destination interpacket arrival time (mSec)
	tcprrt	TCP connection setup round-trip time, the sum of 'synack' and 'ackdat'.
	synack	TCP connection setup time, the time between the SYN and the SYN_ACK packets.
	ackdat	TCP connection setup time, the time between the SYN_ACK and the ACK packets.
	is_sm_ips_ports	If source and destination IP addresses equal and port numbers equal then, this variable takes value 1 else 0
	ct_state_ttl	No. for each state according to specific range of values for source/destination time to live.
	ct_flw_http_mthd	No. of flows that has methods such as Get and Post in http service.
	is_ftp_login	If the ftp session is accessed by user and password then 1 else 0.

	ct_ftp_cmd	No of flows that has a command in ftp session.
	ct_srv_src	No. of connections that contain the same service and source address*.
	ct_srv_dst	No. of connections that contain the same service and destination address*.
	ct_dst_ltm	No. of connections of the same destination address*.
	ct_src_ltm	No. of connections of the same source address*.
	ct_src_dport_ltm	No of connections of the same source address and the destination port*.
	ct_dst_sport_ltm	No of connections of the same destination address and the source port*.
	ct_dst_src_ltm	No of connections of the same source and the destination address*.
	attack_cat	The name of each attack category. In this data set , nine categories e.g. Fuzzers, Analysis, Backdoors, DoS Exploits, Generic, Reconnaissance, Shellcode and Worms
	Label	0 for normal and 1 for attack records

\* in 100 connections according to the last time.

