

RESISTO:

D3.6_Damage/Vulnerability models for physical and cyber threats of telecom CI



RESISTO

D3.6 – DAMAGE/VULNERABILITY MODELS FOR PHYSICAL AND CYBER THREATS OF TELECOM CI

| | | | |
|--------------------------|------------------------|------------|--------|
| Document Manager: | Mirjam Fehling-Kaschek | Fraunhofer | Editor |
|--------------------------|------------------------|------------|--------|

| | |
|-----------------------------|---|
| Project Title: | RESilience enhancement and risk control platform for communication infraSTructure Operators |
| Project Acronym: | RESISTO |
| Contract Number: | 786409 |
| Project Coordinator: | LEONARDO |
| WP Leader: | Fraunhofer |

| | | | |
|------------------------|------------------------|-----------------|------------|
| Document ID N°: | RESISTO_D3.6_191015_01 | Version: | 1.0 |
| Deliverable: | D3.6 | Date: | 15/10/2019 |
| | | Status: | APPROVED |

| | |
|--------------------------------|---------------|
| Document classification | PUBLIC |
|--------------------------------|---------------|

| Approval Status | |
|---|---|
| Prepared by: | Mirjam Fehling-Kaschek (Fraunhofer) |
| Approved by: (WP Leader) | Mirjam Fehling-Kaschek (Fraunhofer) |
| Approved by: (Coordinator) | Bruno Saccomanno (LDO) |
| Advisory Board Validation (Advisory Board Coordinator) | NA |
| Security Approval (Security Advisory Board Leader) | Bruno Saccomanno (on behalf of Alberto Bianchi) (LDO) |

CONTRIBUTING PARTNERS

| Name | Company / Organization | Role / Title |
|--|------------------------|-----------------------|
| Mirjam Fehling-Kaschek, Gael Haab, Katja Faist, Aishvarya Jain Kumar, Natalie Miller, Jörg Finger | Fraunhofer | Scientific Researcher |
| Cosimo Palazzo Stefano Panzieri | RM3 | Scientific Researcher |
| Alberto Neri | LDO | Contributor |

DISTRIBUTION LIST

| Name | Company / Organization | Role / Title |
|----------------|------------------------|----------------------|
| PMT | RESISTO CONSORTIUM | NA |
| Markus Muller | EC DG REA | EC Programme Officer |
| General Public | NA | NA |

REVISION TABLE

| Version | Date | Modified Pages | Modified Sections | Comments |
|---------|------------|----------------|-------------------|--------------------------|
| 0.1 | 06/08/2019 | All | All | First version |
| 0.2 | 13/08/2019 | | 5.3, 5.1 | WP3 review comments |
| 0.3 | 09/09/2019 | All | 2.3, 3.3, 3.3.5 | Comments by EU reviewers |
| 1.0 | 30/09/2019 | All | All | Final Release |

COPYRIGHT STATEMENT



© 2018-2021 This document and its content are the property of the RESISTO Consortium. All rights relevant to this document are determined by the applicable laws. Access to this document does not grant any right or license on the document or its contents. This document or its contents are not to be used or treated in any manner inconsistent with the rights or interests of the RESISTO Consortium or the Partners detriment and are not to be disclosed externally without prior written consent from the RESISTO Partners. Each RESISTO Partner may use this document in conformity with the RESISTO Consortium Grant Agreement provisions.

The research leading to these results has received funding from the European Union's Horizon 2020 Research and Innovation Program, under the Grant Agreement No 786409.

The views and opinions in this document are solely those of the authors and contributors, not those of the European Commission.

PROJECT CONTACT



LEONARDO
Via Puccini 2 – Genova (GE) – 16154 – Italy
Tel.: +39 348 6505565
E-Mail: bruno.saccomanno@leonardocompany.com

RESISTO PROJECT – PUBLISHABLE EXTENDED ABSTRACT

Communications play a fundamental role in the economic and social well-being of the citizens and on operations of most of the CIs. Thus they are a primary target for criminals having a multiplier effect on the power of attacks and providing enormous resonance and gains. Also extreme weather events and natural disasters represents a challenge due to their increase in frequency and intensity requiring smarter resilience of the Communication CIs, which are extremely vulnerable due to the ever-increasing complexity of the architecture also in light of the evolution towards 5G, the extensive use of programmable platforms and exponential growth of connected devices. The fact that most enterprises still manage physical and cyber security independently represents a further challenge. RESISTO platform is an innovative solution for Communication CIs holistic situation awareness and enhanced resilience (aligned with ECSO objectives). Based on an Integrated Risk and Resilience analysis management and improvement process availing all resilience cycle phases (prepare, prevent, detect, absorb, etc.) and technical resilience capabilities (sense, model, infer, act, adopt), RESISTO implements an innovative Decision Support System to protect communication infrastructures from combined cyber-physical threats exploiting the Software Defined Security model on a suite of state of the art cyber/physical security components (Blockchain, Machine Learning, IoT security, Airborne threat detection, holistic audio-video analytics) and services (Responsible Disclosure Framework) for detection and reaction in presence of attacks or natural disasters. Through RESISTO Communications Operators, will be able to implement a set of mitigation actions and countermeasures that significantly reduce the impact of negative events in terms of performance losses, social consequences, and cascading effects in particular by bouncing efficiently back to original and forward to operational states of operation.

EXECUTIVE SUMMARY

This deliverable summarizes the final status of Task 3.3. The aim of this task is to further develop software tools for the simulation of disruptive events in the telecommunication infrastructures. The assessment and quantification of cyber-physical threats is a necessary step to follow the risk and resilience analysis and management process of the RESISTO project.

This report provides input needed for the simulation software specification:

- a short overview on which kind of threats need to be simulated
- a review of network simulation tools that are either open-source or commercially available
- an evaluation of network models and setups, based on network schemes provided by telecommunication partners within the project

Two simulation tools are introduced, which are available and further developed for the RESISTO project:

- CaESAR (EMI): simulation tool for computing cascading effects within critical infrastructures to be used for a regular weak point and resilience analysis of the network
- CISIApro (RM3): main simulation software for the RESISTO platform used for the direct response simulation. It can also serve for a regular weak point analysis.

This deliverable (D3.6) is based on the intermediate report D3.5. A summary of modifications implemented with respect to the intermediate version is provided at the end of the introduction.

CONTENTS

| | |
|---|-----------|
| 1. INTRODUCTION | 11 |
| 2. REVIEW OF POTENTIAL HAZARDS AND DISRUPTIONS | 13 |
| 2.1. Evaluation of the Excel templates | 13 |
| 2.2. Features needed to simulate potential disruptions in telecommunication infrastructures | 14 |
| 2.3. Analysis of Templates..... | 14 |
| 2.3.1. System analysis..... | 15 |
| 2.3.2. System performance function identification..... | 15 |
| 2.3.3. Disruption Identification | 16 |
| 2.3.4. Selection of options for modifying resilience | 16 |
| 3. REVIEW OF AVAILABLE SIMULATION TOOLS | 17 |
| 3.1. Network simulation tools | 17 |
| 3.1.1. NS2..... | 18 |
| 3.1.2. OMNeT++ | 18 |
| 3.1.3. NS3..... | 18 |
| 3.1.4. OPNET riverbed | 18 |
| 3.1.5. QualNet | 18 |
| 3.1.6. OneSim..... | 19 |
| 3.1.7. PeerSim | 19 |
| 3.2. Packet-based network simulation | 19 |
| 3.3. Specificity of Caesar and CisiaPro in relation to existing tools | 20 |
| 3.3.1. Resilience quantifying for the LTCL | 21 |
| 3.3.2. Expected Availability..... | 22 |
| 3.3.3. Expected Reliability | 22 |
| 3.3.4. Other expected network performance metrics | 22 |
| 3.3.5. Resilience and security..... | 22 |
| 4. EVALUATION OF NETWORK SCHEMES | 24 |
| 4.1. General Network Model | 24 |
| 4.2. Network Schemes Provided in D2.4 | 27 |
| 4.3. Comparison of 4G networks to 5G | 27 |
| 4.4. Testbeds and use cases discussion | 29 |
| 5. DESCRIPTION AND PLANS FOR SIMULATION TOOLS | 34 |
| 5.1. Description of the EMI simulation tool CaESAR | 34 |
| 5.1.1. Implementation plan for CaESAR | 37 |
| 5.2. Description of RM3 tool CISIapro | 39 |
| 5.2.1. Resilience time calculation using CISIapro approach..... | 41 |
| 5.3. Validation of the simulation results | 44 |

| | |
|--------------------------------------|----|
| 5.4. Requirements traceability | 46 |
| 6. SUMMARY | 47 |
| 6.1. Discussion and outlook | 47 |

List of figures:

| | |
|---|----|
| Figure 1: RESISTO logical architecture (see deliverable D2.6 for more information)..... | 11 |
| Figure 2: The RESISTO risk and resilience management process that consists of 9 steps. The inputs and tools/methods relevant for each step are shown. | 15 |
| Figure 3: General idea for package based networking | 20 |
| Figure 4: The resilience cycle as defined by [20]. | 21 |
| Figure 5: Exemplary, time-resolved performance curves for disruptive events. | 21 |
| Figure 6 : The general nodes of a fixed-line network. [1] | 24 |
| Figure 7: A telecommunication network general model. [2] | 25 |
| Figure 8: Older telecommunication network designs ranging from 2G to 4G. [27] | 28 |
| Figure 9: An example model of a 5G network. [27]..... | 29 |
| Figure 10: ORO testbed | 30 |
| Figure 11: The dependency radius describes the probability of connection in-between CIs | 34 |
| Figure 12: Overview over CaESAR simulation tool. Damages are introduced, the vulnerability of the interdependent infrastructure is analysed, critical components are identified and mitigations are applied to them. | 35 |
| Figure 13: The different states of a system influencing the resilience. | 36 |
| Figure 14: Screen of CaESAR computation results: mitigation strategies | 37 |
| Figure 15: Exemplary visualization of a resilience matrix (performance metrics vs threats). The performance functions and threats were taken from real inputs but the curves are not realistic, i.e. not based on real data or a realistic simulation. | 38 |
| Figure 16: CISIAview Orange Romania Case Study | 42 |
| Figure 17: CISIAview Real-Time Resilience graph detail | 43 |
| Figure 18: The performance time curve with different resilience indicators defined. | 44 |
| Figure 19: LTCL and STCL cycle relationship and resilience indicators flow. | 45 |

ABBREVIATIONS

| | |
|-------------------|---|
| 2G, 3G, 4G | Second, third and fourth generation of mobile phone systems |
| CI | Critical Infrastructure |
| EU | European Union |
| GUI | Graphical User Interface |
| IoT | Internet of Things |
| KPI | Key Performance Indicator |
| LTE | Long Term Evolution (= 4G) |
| LTCL | Long term control loop |
| RI | Resilience Indicator |
| STCL | Short term control loop |
| T | Task – referring to tasks within the WPs of the RESISTO project |
| WP | Work Package – referring to other WPs of the RESISTO project |

1. INTRODUCTION

The main objective of the RESISTO project is to improve the security and resilience in communication infrastructures. This is achieved by developing an innovative platform for threat detection, an integrated risk and resilience assessment and optimized decision support. The RESISTO platform interfaces to existing communication infrastructures and modularly integrates tools and methods in the integration platform, which consists of two control loops, the short term control loop (STCL) and the long term control loop (LTCL). A scheme of the architecture of the integration platform is shown in Figure 1.

Aim of work package (WP) 3 “Cyber-physical risk/resilience assessment and improvement process for preparation, prevention and protection” is the definition of the long term control loop of the RESISTO platform, which mainly features the risk and resilience analysis and management process.

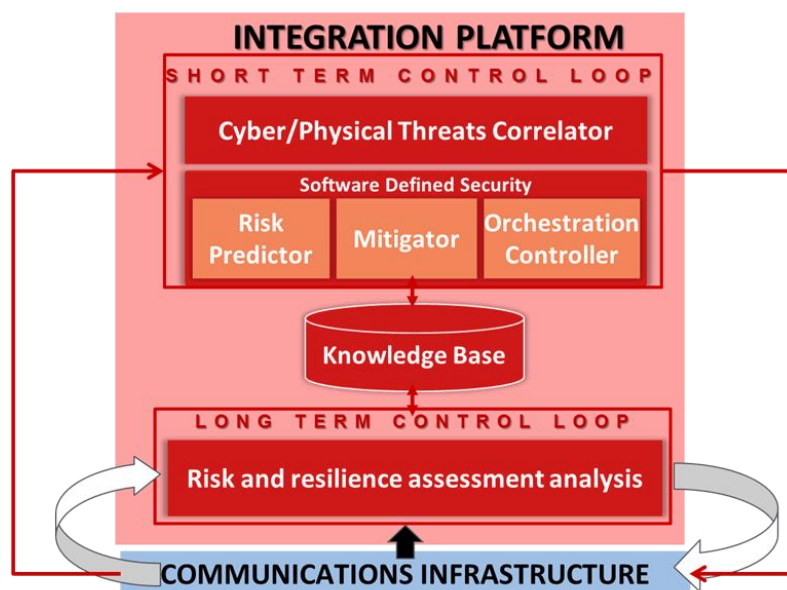


Figure 1: RESISTO logical architecture (see deliverable D2.6 for more information)

The following tasks are included in WP3:

- T3.1 Long term learning cyber-physical risk and resilience management
- T3.2 Methods/Plans for joint cyber-physical security management process
- T3.3 Physical protection and prevention methods: assessment and cyber-physical interaction
- T3.4 Risk and resilience quantities and related KPIs for telecommunications infrastructure
- T3.5 Desk-top application to use case scenarios for second use cases refinement

This report summarizes the final status of T3.3. Main objective of this task is to define and develop software modules for the long term control loop for use in WP4-6. These modules should assess the effect of disruptive events on the telecommunication infrastructure. Therefore, the focus of this report is set on network simulation tools. Other tools and methods relevant for the long term control loop are collected in the first and final reports D3.3 and D3.4 “Methods for cyber-physical security management for telecom CI” of T3.2. The risk and resilience assessment process and the mapping of

the tools to this process is described in D3.1 of T3.1, which will be updated in the final version D3.2 (due to in M18).

The structure of the report is described in the following. It is based on the first version of the report, D3.5, which was thoroughly reviewed and updated. Major modifications and new sections that were added in D3.6 are specifically mentioned in terms of Updates for each section.

Section 2:

First, potential disruptions as collected within T2.2 of WP2 are evaluated in Section 2, in order to get an overview of the events that need to be simulated. This helps to define necessary features for the simulation tools.

- ➔ Updates: New results from T2.2 were considered, in particular a new section 2.3 was added to provide an analysis of the provided information in the templates.

Section 3:

As a next step, available simulation tools for simulating communication infrastructures are evaluated in Section 3. A comparison of these tools allows to get an idea of main features that need to be implemented in order to simulate the networks but also to evaluate which features might be missing in these tools and can be implemented in an own network software implementation.

- ➔ Updates: A new section 3.3 was added to describe the general aim of resilience quantification of the CaESAR and CisiaPro simulators. In particular, also the security vs. resilience aspect is discussed in subsection 3.3.5.

Section 4:

An important input to simulative approaches are schemes of the network structures. A collection of network schemes was provided by telecommunication partners within in the project in D2.4 of T2.3 of WP2. An analysis of the schemes regarding their comparability, usefulness and completeness is provided in Section 4.

- ➔ Updates: The contents of D2.4 were updated for the final version D2.5 of T2.3, focusing more on a first description of the testbeds. For this report, the initial analysis of the contents of D2.4 was kept in section 4.2, but a new content, section 4.4, was added to discuss the testbed and use case specifications, based on the example of the ORO testbed.

Section 5:

A description of network simulation tools, CaESAR and CISIApro, which will be refined and provided for the use within RESISTO is given in Section 5.

- ➔ Updates: For the CaESAR tool, the development plan described in section 5.1.1 was updated. For CISIApro a new view, containing the time-resolved performance curve (resilience curve), was implemented and is described in section 5.2.1. This implementation allows for a validation of the simulation results, which is described in section 5.3. Furthermore, section 5.4 was added, which relates mandatory requirements for RESISTO (defined in D2.1 of T2.1) to the tools and methods described in this report.

Section 6:

Finally, a summary of the report is presented in Section 6.

- ➔ Updates: Since this is the final report of T3.3, section 6.1 “Next Steps” was replaced by a discussion and outlook. It should be noted, that the work on the simulation tools and results will be carried on in other tasks and WPs such as T2.5, T3.5 and WP7-WP9.

2. REVIEW OF POTENTIAL HAZARDS AND DISRUPTIONS

Potential hazards and disruptions for telecommunication infrastructures are investigated in T2.2 of WP2. Aim of T2.2 is to provide a living threat, hazard and disruption list containing cyber, physical and cyber-physical threats.

Input for the list is collected by a tabular Excel template. This template is introduced in the deliverables D2.2/D2.3 of T2.2. An evaluation of information provided by the template is provided in subsection 2.1. The information is used to deduce some general requirements for the simulation in subsection 2.2.

2.1. Evaluation of the Excel templates

The Excel template not only contains the list of threats but also tables containing relevant information about system components and system functions impaired by the threats and possible mitigation options. The information on affected system components can directly support the model specification for the network simulation regarding the level of details to be implemented, i.e. which system components need to be added as nodes or links. The network simulation includes a quantification of resilience quantities, which can be defined based on the system function information. The mitigation options provided are an important input for simulating the effect of possible countermeasures.

The Excel template was sent to all telecommunication operating partners to be filled with relevant threats regarding their network infrastructures. Here, a short list of observations from the available templates is given:

1. As requested, a variety of threats is provided, including cyber, physical and cyber-physical threats with different economic impact and frequency.
2. The threats and system components are associated to different subsystems of the network, e.g. core or radio network. The impact of a threat or a component failure on the telco network have to be explored (in term of availability, throughput, latency, security ...). Telecommunication partners agreed to send further information referring to this for the threats used in the use cases. Literature will be used to complete this knowledge.
3. Most threats affect more than one system component and system function and a significant number of threats affect the majority of system components.
4. The level of complexity, comparing the results from different partners, differs. This is probably caused by different focus and expertise of the persons filling out the templates. This will make more difficult the calculation of the offline resilience of the telecommunication network system (LTCL), approximations will have to be done. To have more precise results, two options have been discussed:
 - 1) The telecommunication partners are going to precise some values for the threats that are used in the use cases.
 - 2) Real data from events which are analyzed in the STCL will be stored in the knowledge base. They can be compared with the settings and results of the LTCL and, if necessary, they will be used in the next LTCL cycle to improve the resilience calculation (see section 5.3).

2.2. Features needed to simulate potential disruptions in telecommunication infrastructures

In general, the software tool must be able to simulate all types of threats relevant for the telecommunication infrastructures. The following conclusions can be deduced from the observations of the previous subsection.

1. The threat list contains examples for all relevant types or classes of events to ensure that all necessary features can be identified.
2. The division in different sub-systems of the network should also be reflected by the simulation. It strongly suggests the implementation of different network classes which are interconnected. The type of interconnection and in particular how the failure in one sub-system affects the other sub-systems are continuously investigated throughout the implementation of the different testbeds and use cases within the project. They will be revised during the implementation of the macro scenarios.
3. A set of typical events on the network can be deduced, e.g. threats that affect all types of components versus threats that affect only specific types of components. The different localization of the effects is an evident feature that the software must be able to cope with: most natural disasters (e.g. earthquakes, floods) will affect a certain region of the infrastructure while man-made attacks can be directed at just a single node of the network or a specific type of system component in the network.
4. It does not work to combine the information of the templates into one global set of tables and a global threat list, due to the diversity of the inputs. This makes the global evaluation of general results harder but still allows to derive main features per operator which can be compared. Moreover, different use case scenarios are foreseen in this project based on different environments of the operators, which will require an adaption of the simulation tools as well (see observation 2.).

2.3. Analysis of Templates

A short introduction to the Excel templates has been provided in subsection 2.1. More detailed information is presented in this section in form of a presentation of the collected data and its usefulness for the RESISTO project. At the same time, it is discussed which information is still needed to improve the resilience computation and the selection of options for modifying resilience via the RESISTO platform.

To present the collected data and clarify their meaning in the RESISTO project, the 9 steps resilience management process is used, see Figure 2. At each step of the process, it is shown which data has been collected. A more detailed description of the process is given in e.g. D3.1 of T3.1.

A discussion about the completeness of the data for the RESISTO project is presented in the following.

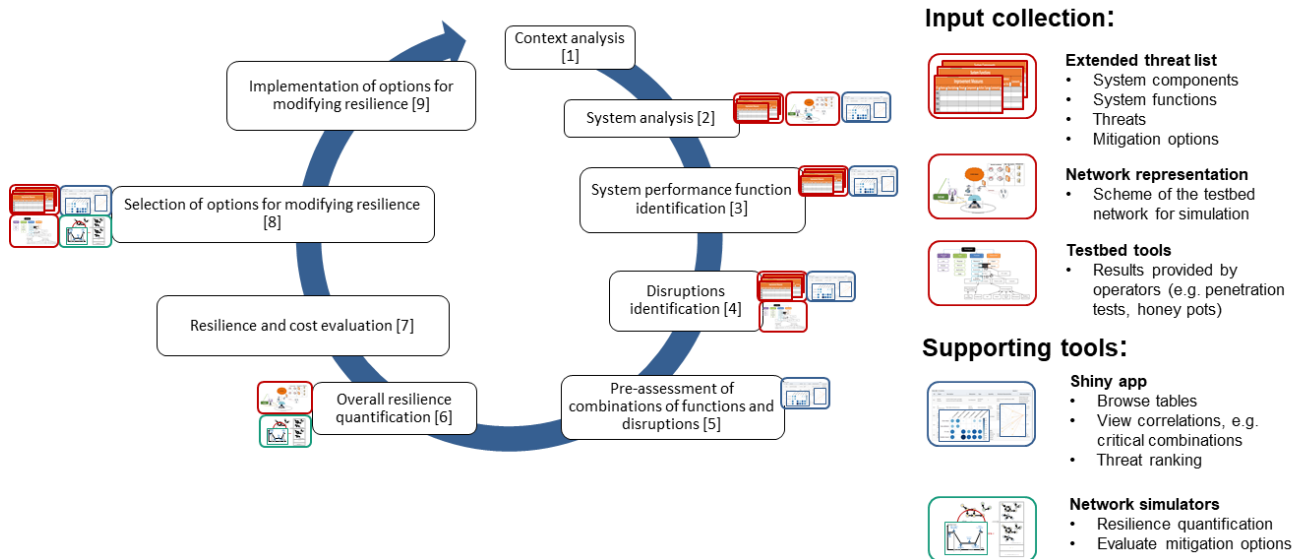


Figure 2: The RESISTO risk and resilience management process that consists of 9 steps. The inputs and tools/methods relevant for each step are shown.

2.3.1. System analysis

The table consists of a collection of relevant system components. As example: Border routers, fibre optical infrastructure, network security equipment, workstation and servers, etc.

Diverse information about this equipment was provided, including a description of the system component, to which subsystem this component belongs, to which other components it is interconnected. Other information is only partially provided. As example, the quantity of the component in the telecommunication network was sometimes provided, sometimes not. It has been decided that for the RESISTO project the simulation and computation will be done for testbeds, which are little networks that are representations of a telecommunication network. For the testbeds, full descriptions including the quantities and connections for all components are provided (see deliverable 2.8 of T2.5). But it is very important to have a complete description of all the components which are used in the various testbeds. Some descriptions are missing and will be complemented in the process of the testbed definition and implementation.

2.3.2. System performance function identification

Here, various system functions were collected, e.g. voice services, mobile data services, L1 connectivity. Some information about the system functions are collected, as a description, to which subsystem they belong, and which system components are needed to deliver this functionality. Missing is what level in terms of network performance (availability, throughput, delay, loss packet, jitter etc...) is needed to deliver these functions in an acceptable way in term of Quality of Service (QoS).

Two ways can be used to fill this gap: the telecommunication operators could define universally acceptable QoS criteria for each application category (voice services, audio/video streaming, mobile data services, etc...) or criteria found in the literature will be used. As example for the voice service, the following network performance would be required:

- One-way delay: < 150ms
- Request/response delay: < 4s
- Loss: <3%
- Jitter < 50ms

A description of acceptable QoS criteria for each system function can be found in [1].

2.3.3. Disruption Identification

Here various threats (physical and cyber) were collected. The frequency and the duration (or recovery time) of each threat were also collected. This is very important for the computation of the expected resilience of the network. In addition, for each threat, it was collected which system component and which system function would be affected by the threat. It allows a good understanding of the impact of a threat on the network and on the network functions. Since it was decided that the RESISTO platform will be tested for the developed use cases and testbeds, it have to be verified that all the threats of the use cases are collected and well documented. This step of the resilience management process will be enlarged and specific security analysis (e.g. penetration tests) will be included to discover if the critical infrastructure is vulnerable to some specific known threats (see D3.4 section 4.-3). This is planned within the scope of T2.5 and WP7-WP9.

2.3.4. Selection of options for modifying resilience

In the excel templates some improvement measures were collected, for example technical improvement as installation of anti-hacking or virus system, installation of battery packs for some system components or improve staff training. Improvement measures were collected for the different resilience phases (prepare, prevent, protect, respond, recover) for all the collected threats. Which system component should be taken into account by each improvement measure was also collected. Some of the measures can be implemented easily in order to estimate the resilience gain, e.g. adding redundancies. For others, however, more detailed feedback would be needed in order to quantify these measures in term of resilience, for example to quantify the effect of an improved staff training.

3. REVIEW OF AVAILABLE SIMULATION TOOLS

In this section, various simulations tools for network computing are presented, then the specificity of CaESAR and CisiaPro in relation to existing tools is discussed.

3.1. Network simulation tools

Table 1 lists several network simulation tools that are out there in the market. The table categorizes the tools based on the programming language used, feasible operating systems, type of the license, and networks that could be implemented using these tools. All the simulators specified in the table use packet-based networking, which is shortly introduced in section 3.2. Further subsections within this section will describe each simulator in more detail.

| S.No. | Tool | Programming language | Environment OS | License type | Network type |
|-------|--------------------------------|-----------------------|--|------------------------------------|--|
| 1 | NS2 | Tcl / Tk / C++ / OTCl | Windows(CYGWIN), Linux MINT/UBUNTU / FEDORA / MINT / etc. | Open Source | Wired / wireless /wireless sensor /ADHOC/ MANET/ Wired cum Wireless / SDN / VANET / Security /Vertical Handover etc. |
| 2 | OMNeT++ | C++ | Windows, Unix-based, Mac OS X 10.6 and 10.7 | Open Source | Wired / wireless / wireless sensor / ADHOC / MANET / Wired cum Wireless / SDN / VANET / WBAN / Under water sensor network / Social sensor network etc. |
| 3 | NS3 | C++, python | Windows (CYGWIN), Linux MINT/UBUNTU /Free BSD X86/ FEDORA / Mac OS | Open Source | Wired / wireless / wireless sensor / ADHOC / MANET / Wired cum Wireless / SDN / VANET / Device to Device Communication etc. |
| 4 | Riverbed Modeler (OPNET) | C/C++ | Hewlett-Packard, Sun- 4 SPARCVarious, Solaris 2.6, 7 8Microsoft Windows NT 4.0/Windows 2000Required System Patches | Commercial network simulator | Wired / wireless / wireless sensor / ADHOC / MANET / Wired cum Wireless / SDN / VANET / Radio Network etc. |
| 5 | QualNet | C++ | Mac OS, Unix, Windows, Linux, Solaris, DOS | Commercial network simulator | Wired / wireless / wireless sensor / ADHOC / MANET / Wired cum Wireless / SDN / VANET etc. |

| | | | | | |
|----------|---------|------|--------------------------|-------------|---|
| 6 | OneSim | Java | Windows/ Linux/ Mac OS X | Open Source | Wired / wireless / wireless sensor / ADHOC / MANET / Wired cum Wireless/SDN/ VANET etc. |
| 7 | PeerSim | Java | Windows/ Linux | Open Source | Parallel Systems / Wired / wireless / wireless sensor / ADHOC / MANET / Wired cum Wireless / SDN / VANET etc. |

Table 1: Comparison table of available network simulation tools (simulators and libraries)

3.1.1. NS2

NS2 (Network simulator 2) is a discrete event simulator which provides support for the simulation of TCP, routing and multicast protocols over wired and wireless networks [2, 3] with an open source license. It is implemented using OTCL scripts and the C++ language and can be run over a windows or a Linux machine (or cluster). Using NS2 disruption events can also be simulated [4].

3.1.2. OMNeT++

OMNET++ (Objective Modular Network Testbed in C++) in itself is not a network simulator but a component based C++ simulation library to build scalable network simulators. OMNET++ runs on Windows, Linux, Mac OS and other Unix systems. IDE support is only available on Windows, Linux and Mac OS [5]. It has support for sensor network, wireless ad-hoc network, internet protocols, performance modelling and is highly customizable. Being a simulation library it also provides the option to trigger the disruption events at will.

3.1.3. NS3

NS3 (Network simulator 3) is also a discrete event simulator intended to eventually replace the NS2 simulator. It is designed to improve scalability and coding style. Its core is written in C++ with optional Python scripting interface [6, 7]. NS3 is supported over Linux, Mac OS and can be used on a windows machine using virtualization [8]. Being an event simulator, it supports the introduction of events like loss of communication packet or node malfunction.

3.1.4. OPNET riverbed

OPNET (Optimized Network Engineering Tools) is currently the most widely used network simulator available free for academic research and education, apart from that, it is a proprietary software [9]. It is a discrete event simulator developed by Massachusetts Institute of Technology in 1987 using C++. It can model all types of networks (wired and wireless) and technologies (including VoIP, TCP, OSPFv3, MPLS, and IPv6). OPNET is supported over Windows, Linux and Solaris platforms [10].

3.1.5. QualNet

Qualnet is a proprietary network simulator designed to mimic real communication networks. It is built in C++ and is supported over Windows and Linux systems [11]. It can be used to simulate both wired

and wireless networks and support major communication protocols. For large networks it is highly scalable.

3.1.6. OneSim

OneSim is an agent-based discrete event based simulator and is written in JAVA [12]. Onesim is capable to simulate the node movement and can use various DTN routing algorithm for different sender and receiver types. It is maintained by Aalto University and Technische Universität München [13]. It is supported over Windows, Linux, Unix and Mac OS X with open source license.

3.1.7. PeerSim

PeerSim is focused towards the simulation of very large (of the order of millions of nodes) peer-to-peer systems [14]. It is started under the EU projects BISON and DELIS. The simulator is written in JAVA and has an open source availability. PeerSim provides the opportunity to write personalized communication protocols [15].

3.2. Packet-based network simulation

In packet switched networks, data move in separate, small blocks (termed as “Packets”), based on the destination address in each packet. The received packets are then reassembled in a proper sequence by the receiver to make up the original message. Packets are made of a header and a payload. Based on the information stored in the header the networking hardware directs the packets to its destination where the payload is extracted and used by the application software [16]. A scheme for the packet-based networking is shown in Figure 3.

Simulators described in the previous section 3.1 rely on packet based networking. This simulation approach can be computationally expensive [17]. Therefore, a tool operating at flow level is efficient to save computation time [18, 19].

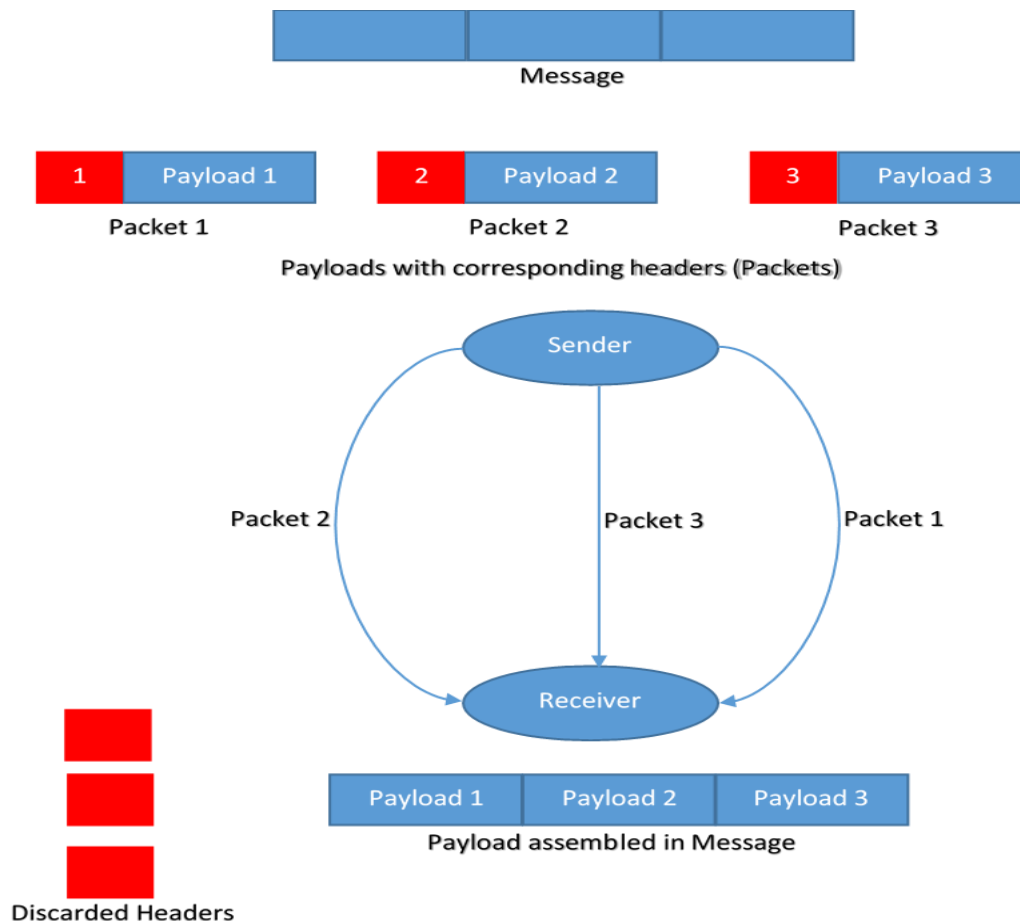


Figure 3: General idea for package based networking

3.3. Specificity of Caesar and CisiaPro in relation to existing tools

Caesar and CisiaPro are specially developed to quantify the resilience and its improvement for the telecommunication infrastructure. The previous presented simulators (section 3.1) are network simulators, while CisiaPro is an infrastructure simulators, i.e. it take in account a whole communication infrastructure (e.g. power supplies, conditioning, building, tower...). The main objective of resilience analysis RESISTO is to protect a complete communication infrastructure and not only the communication network. In RESISTO 5 phases has been identified to face threats: identification, protection, detection reaction mitigation. In order to achieve this objective, two tools are developed. The Caesar tool as part of the long term control loop has the scope to increase the resilience of the network by identifying potential weaknesses of the critical infrastructure and proposing measures to improve the resilience of the telecommunication infrastructure (Identification and protection phase from resilience cycle.). Differently from this, the CisiaPro tool is developed for the short term control loop, which is an online analysis of the network. The scope of the short term control loop is to detect the occurrence of a threatening event, analyze the effect of it (in term of network performance) and choose a response to this. The main goal of the response is to maintain critical functionality and provide relief. We can see the short term control loop as part of detection, reaction and mitigation phases from resilience cycle.. The aim of the combination of the LTCL and the

STCL is then to take the total resilience cycle in account. To enable a better comprehension of the links between the LTCL and the STCL a brief description of resilience quantification for telecommunication infrastructure is given in the next section.



Figure 4: The resilience cycle as defined by [20].

3.3.1. Resilience quantifying for the LTCL

The resilience is defined as how a network reacts to a disruption. We can see an illustration of this in the Figure 5, following guidelines presented e.g. in [21]. The impact of an event (Earthquake or Storm) succeeding at a time-point of 1-hour is represented by the evolution of a performance function of the network over time.

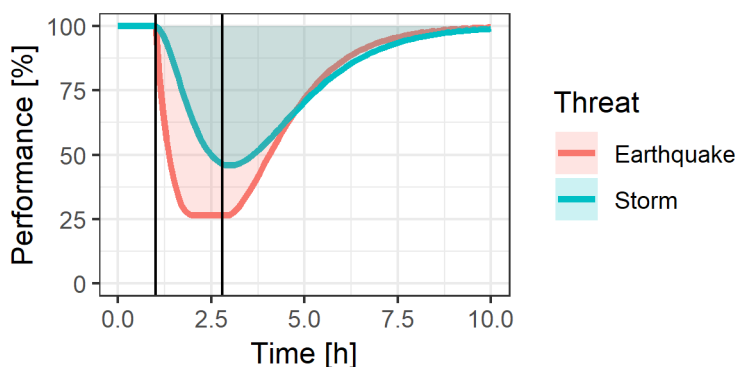


Figure 5: Exemplary, time-resolved performance curves for disruptive events.

It is important to understand that various performance metrics are given for a telecommunication network, such as availability, security, throughput, jitter. This results in a resilience matrix (performance metrics vs threats) for a network. Indeed, a resilience curve has to be computed for each combination of disruptive event and performance metric. This is illustrated for a better comprehension in section 5.1 (see Figure 15). Important to understand is that the resilience in the LTCL is calculated via an offline simulation in the preparedness phase with the aim to measure how well the system is prepared to faults and challenges. It is therefore design-based resilience, while the resilience calculated in the short term loop (corresponding to the detection, reaction and mitigation phases from the resilience cycle) are true metrics, which are to be measured time-resolved during the

operation of service (see [22] p. 91). To explicit this difference in the following, the term expected resilience will be used for the LTCL resilience computation and the term operational resilience will be used for the STCL computed resilience. In other words, “a clear distinction should be made between expected [...]resilience] and the operational [...]resilience]. The difference between these metrics is the source data that is used: while the operational [...] [resilience] uses historical data, the expected [...]resilience] uses vendor-provided [or others] statistics of the equipment to calculate a certain network [expected performance]”(see [22] p.92). Please note, that in case of RESISTO a data exchange between the LTCL and the STCL is planned for better modelling of the functionality of the network, see section 5.3.

In the next sections, 3.3.2 - 3.3.4, we will present a short introduction to network metrics, and discuss the relevance of these for the LTCL (for more details see [22]).

3.3.2. Expected Availability

The expected availability is defined as the percentage of time the system is available to the end user. Availability is without doubt one of the most important network performance metrics for the telecommunication operator, since users need to have access to their services (internet, voice, etc.) seamlessly all the time.

It must be noted that “the expected [availability] of an ICT system is calculated during the design phase and should be recalculated when components are added or removed or changes in topology occur. Other than that, the expected [resilience] values are static and do not require periodic reporting.” ([22], p. 93).

This comment is valid also for all other resilience computations.

3.3.3. Expected Reliability

The expected reliability is the ability of a system to perform its required function for a specific period of time (see [22] p. 96).

3.3.4. Other expected network performance metrics

As discussed in section 2.3.2, to know if a system function of a telecommunication network (as voice services for example) is available with the needed QoS, the analysis of various network performance measures is needed. In order to calculate the expected resilience for those functions, other expected network performance metrics such as throughput, delay, packet loss, jitter will be used.

3.3.5. Resilience and security

One very important aspect to take into account when speaking of resilience of a telecommunication network is the network security. Indeed, each end user expects his telecommunication network to be secure to transmit sensitive data (as buying on the net, sending sensitive e-mails, etc...). Some cyber-attacks may not have any consequence on the technical network performance functions described before, for example a “Man in the middle” attack would not affect metrics like the throughput or loss of packets. However, the network is not secure anymore, potentially leading to drastic consequences on the telecommunication network. As example, if through the attack, a password from the domain administrator is obtained.

As explained in section 3.3.1 and 5.3 resilience is a matrix concept in which for each couple event/function a different resilience curve can be quantified.(see Table 4)

The scope of the RESISTO project is to analyze the resilience of a telecommunication system for natural, cyber and combined cyber and natural threats. The analysis concerning the security of the telecommunication infrastructure is therefore a part of the resilience analysis. This correspond in the matrix to the elements in which cyber-security threats are taken as event. Possible security threats would be identified during step 4 (e.g. using pentest) and analyzed in the following steps. In order to take in account security aspects, resilience indicators for security can be added e.g. in terms of definitions of reverse safety performance functions: $\text{performance} = (100\% - \text{security loss})$. Basically, the simulation would be used to quantify the security loss for a specific attack.

The ENISA report [22] makes some suggestions for different indicators for security.

For the LTCL (identification and protection phase) following indicators are suggested: (see [22] p. 56-60)

- The risk assessment coverage, which indicates the percentage of systems that have been subject to a risk assessment at any time.
- The risk treatment coverage, which reports the percentage of systems for which risk treatment plans have been documented.
- The security testing coverage, which indicates the percentage of the organization system that has been tested for security risks.

But, “because of the lack of experiential data from the field, no consensus on the range of acceptable goal values for security [...] [resilience indicators] exists.” ([22] p. 59). We still have to explore further how the RESISTO platform could exploit these indicators.

In addition, how other tools, which allow an improvement of the security resilience from the telecommunications network, could be integrated in the RESISTO platform, are being considerate. For example, the use of attack trees, inclusion of honeypots in the telecommunication network, use from MITRE ATT&CK™ knowledge. These methods and tools are discussed in more detail in deliverable D3.4.

For the STCL (detection, reaction, mitigation phase from resilience cycle) following indicators are suggested and could be measured and stored in the knowledge base (real data) (see [22] p. 72-77).

- The incident rate metric, which measures the number of security incidents that occur in a given time period from selected incident categories.
- The illegitimate traffic rate metric, which indicates the resistance against unauthorized traffic that tries to enter on the network
- The percent of system without known severe vulnerabilities, which measures the organization's relative exposure to known severe vulnerabilities. (With the help from CVSS base score).

To conclude, (cyber-)security events and security performance functions have to be included in the values of the resilience matrix computed in the LTCL and in the STCL. In this way, the RESISTO platform help operators to study in the same process security typical issues as well as other possible (i.e. physical and combined cyber and physical) threats. How compute the (cyber-) security performance is a big challenge, which still needs some work. Since a general implementation of all kinds of security performance functions to cover all kinds of cyber-attacks does not seem feasible within the scope of this project, we plan to set a focus on a few relevant security examples defined in the use case scenarios or identified during step 4 (i.e. using pentest).

4. EVALUATION OF NETWORK SCHEMES

4.1. General Network Model

In general, fixed-line networks are divisible into three main sections: access, aggregation and core [23], each with specific node types. Additionally, the access section of the network includes the network termination nodes. A simplified model of a fixed-line network with the different node types and sections clearly defined is provided in Figure 6, adapted from [23].

The access section, also sometimes called the “last mile” of the network, is what initially connects the users to the network. The network termination nodes, or points, are where the end users are located. These termination nodes can be thought of as the devices that want to connect to the network. In a fixed access network, the devices would use wired terminations like modems or landline telephones. The mobile access network devices would be phones or tablets.

The aggregation or distribution section of the network collects all of the data from the access section and connects it to the core network. This section reduces the number of nodes present by collecting the user data from multiple access nodes. The amount of aggregation nodes depends on the population density and how many users want to access the network [23].

While each node of the aggregation section will cover a specific region, the core section, or backbone, will be nationwide [23]. The core is where the services are provided and distributed (Deliverable 2.4 section 4.3.1).

The sections become more meshed and more redundant, the farther into the network they are. The access networks are not meshed at all, the aggregation networks are somewhat meshed while the core is fully meshed. Meshing allows for a more reliable and redundant system.

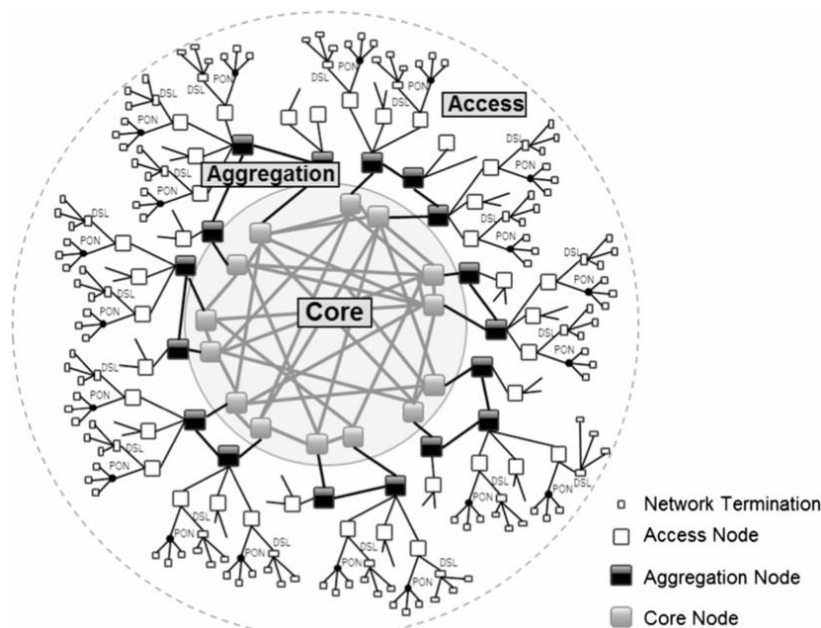


Figure 6 : The general nodes of a fixed-line network. [1]

A structure, from [24], that includes both fixed and mobile services can be seen in Figure 7 with the equipment abbreviations expanded in Table 2.

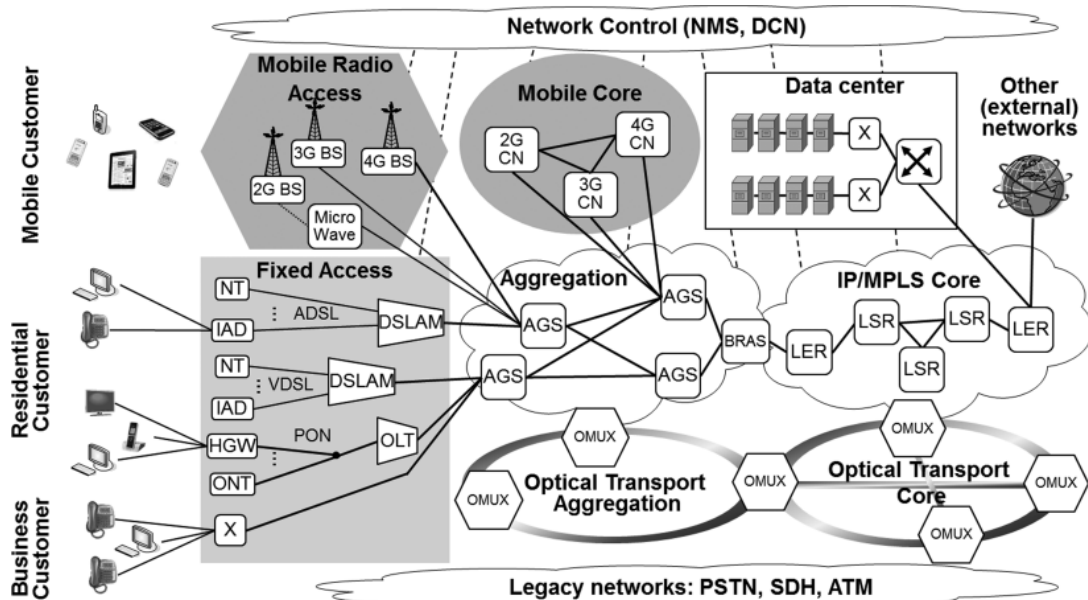


Figure 7: A telecommunication network general model. [2]

The structure is an expanded and more specific version on Figure 6, but follows the same hierarchy. Network terminals (NT) are located in the fixed access section, which is not meshed. The aggregation and core networks are both meshed.

| Abbreviation | Name |
|------------------------|---|
| Fixed Access | |
| NT | Network Termination |
| IAD | Integrated Access Device |
| HGW | Home Gateway |
| ADSL | Asymmetric Digital Subscriber Line |
| VDSL | Very High Speed Digital Subscriber Line |
| DSLAM | Digital Subscriber Line Access Multiplexors |
| ONT | Optical Network Terminations |
| PON | Passive Optical Network |
| OLT | Optical Line Terminations |
| Mobile Access and Core | |
| BS | Base Station |

| | |
|--|-----------------------------------|
| CN | Core Network |
| Aggregation | |
| AGS | Aggregation Switches |
| BRAS | Broadband Remote Access Server |
| IP/MPLS Core | |
| IP | Internet Protocol |
| MPLS | Multi-Protocol Label Switching |
| LER | Label-Edge Routers |
| LSR | Label-Switch Routers |
| Optical Transport Aggregation and Core | |
| OMUX | Optical Multiplexing Systems |
| Legacy Networks | |
| PSTN | Public Switched Telephone Network |
| SDH | Synchronous Digital Hierarchy |
| ATM | Asynchronous Transfer Mode |

Table 2: Abbreviations from Figure 7 explained. [2]

Both the fixed network and the mobile network follow the same general hierarchy as mentioned in Figure 6, starting with the access then aggregation and finally the core. However, because the fixed and mobile network use different devices, their access sections are separate as well as their core section of the network.

Within the fixed access section, the architecture used varies depending on the device and the use. For example, there can be different speeds, which would use either an ADSL or VDSL connection from a device to the DSLAMs. Other differences between ADSL and VDSL include the construction. VDSLs are in street cabinets while ADSLs are in central offices [24].

Within the aggregation section there are multiple AGSs meshed together. Within this section of the network, there is a split between the Ethernet and the optical transport [24]. BRASs are used to connect the aggregation section to the IP/MPLS core section. BRASs help with the authentication, authorization and accounting [24].

The IP/MPLS core consists of LSRs and LERs and connects to the optical transport core. The last LER of this core connects to the data centers. These data centers are used to support the actual operation of the network like billing as well as service provision for the customers like storage options [24].

The mobile radio network starts at the access section. The BSs that are used is depending on the type of connection possible (2G, 3G or 4G). The signal will be aggregated and then sent to the different mobile core networks, which again, varies on the connection type.

4.2. Network Schemes Provided in D2.4

Each network representation provided in Deliverable 2.4 has the three distinct node types: access, aggregation and core, however, the characteristics of each varies. The access node varies depending on the type of service the user wants to access. It varies between fixed or mobile services and within mobile services it will vary between 2G, 3G, 4G/LTE and 5G access. Each node is connected with fiber optic wires or copper cables.

Orange Romania's service provider network follows this hierarchy of networks from access, to the aggregation or distribution as they call it, and finally to core. However, in their general model the access grid is connected directly to the core using their IP backbone network and fiber optic interconnections. Their IP/MPLS (Internet Protocol/Multi-Protocol Label Switching) Network is the stage in between the access and the service provider core (Figure 9 from Deliverable 2.4). Orange Romania Backbone, or core, design has many LSR routers and is similar to Figure 7 with redundancies present.

OTE follows a similar hierarchy, going from access to aggregation to core. They state that meshing is used throughout the system as well, increasing the reliability of the systems. Similar to Figure 7, an IP/MPLS core is used; in OTE's case, it contains seven core nodes. OTE also uses BRAS (Broadband Remote Access Server) in their aggregation node.

BTC uses many access nodes to aggregate data and send it to the main backhaul. This is done with both copper cables and fiber optics wires. Within BTC's access node, DSLAM (Digital Subscriber Line Access Multiplexors) are used. BTC also uses BRAS to connect the aggregation to the core node. BTC also contains an MPLS Core.

There were discussions within the RESISTO consortium about how much details and specifications of the network schemes would be needed by the partners working on the simulation tools. In general, the communication infrastructure partners cannot share very detailed schemes of their networks for security reasons. However, testbed environments will be used for the use case scenarios in WP7-9. It was agreed, that more information, and in particular network schemes can be shared for the testbeds. These will be collected for the final report of T2.3, D2.5.

4.3. Comparison of 4G networks to 5G

Compared to 4G, the 5G network has different engineering requirements [25]. By the time that the 5G networks are rolled out, much more traffic will be online, maybe even reaching fifty billion devices, including machines that will connect to the internet [26]. According to [25], the 5G networks will need to be able to handle an increased data rate, latency restraints, as well energy and cost requirements. For 5G networks, the data rate will increase while the latency will decrease (the network will be faster), and ideally, the energy and costs will also decrease. The 5G networks will also have to have better coverage and higher adaptability [26].

Not only will the new 5G networks have to deal with the increase in human communication, but also machine communication. As society begins to have more devices that can connect to the internet, 5G networks need to be able to deal with this increase in use and diversity [25], [27]. This concept can be described as the Internet of Things (IoT), where typical internet use is combined with new uses like autonomous car-to-car communication, device-to-device communication, and more [27]. The IoT means that more nodes will be required in the 5G network when compared to the 4G [25].

Historical networks (2G – 4G) general models can be seen in Figure 8 [27]. The 4G network has certain limitations, including a lack of flexibility and scalability, high complexity, and U/C planes that are centralized in the network [27]. Much research is occurring to determine the best ways to reach

the 5G requirements. As 5G will be used to improve many different aspects of the electricity grid, a few specific solutions are mentioned in [27]. These include network splitting or slicing, introducing cloud capabilities, and spectrum sharing [27], [28]. The transition towards 5G will require, according to [25], shifting to mm-Wave spectrum, offloading and densification, and a spectral efficiency increase. This will involve an increase in nodes, base stations and antennas [25]. One example of a potential model for a 5G network is seen in Figure 9 [27]. This model has an emphasis on the virtualization and softwarization of the 5G network as methods to meet the 5G network requirements [27]. The differences between the historical networks and the potential 5G network include a bigger use of cloud technology.

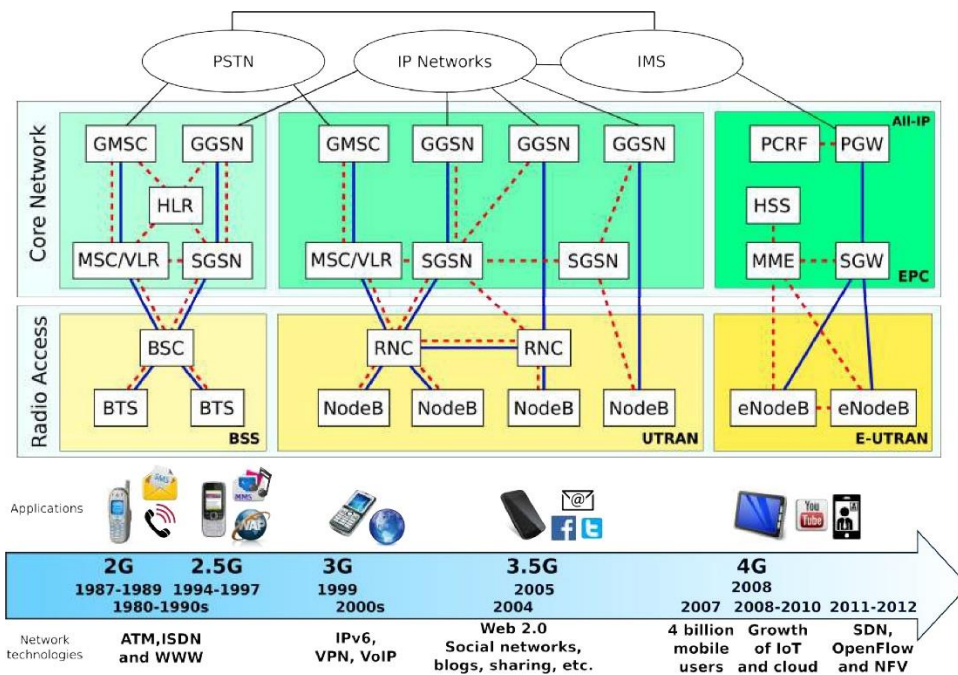


Figure 8: Older telecommunication network designs ranging from 2G to 4G. [27]

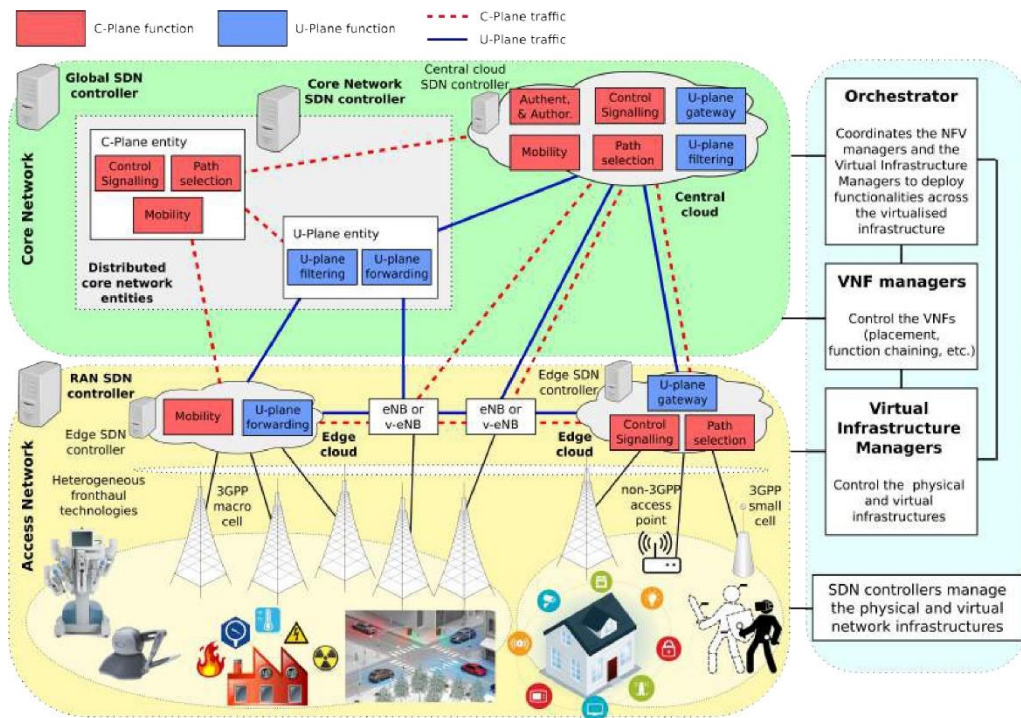


Figure 9: An example model of a 5G network. [27]

4.4. Testbeds and use cases discussion

It was decided in the RESISTO project to test the developed tools for the RESISTO platform within specific use cases and testbeds provided by the telecommunication partners. Presently we do not yet have the complete description of all the use cases (e.g. which threats are going to be tested) and the final complete description of all the testbeds (with the precise description of all the system components). This information is currently collected in the deliverable D2.8.

In this section, a description of one testbed (from ORO) is provided. Through this example it will be shown which information has been collected for this testbed and corresponding use cases, and how this information can be used in the project. It will also be discussed which information is still needed for a better modelling of the network performances from testbeds, and should be completed in the deliverable D2.8 and/or the following work packages dealing with the implementation of the test beds and use cases (WP7-WP9).

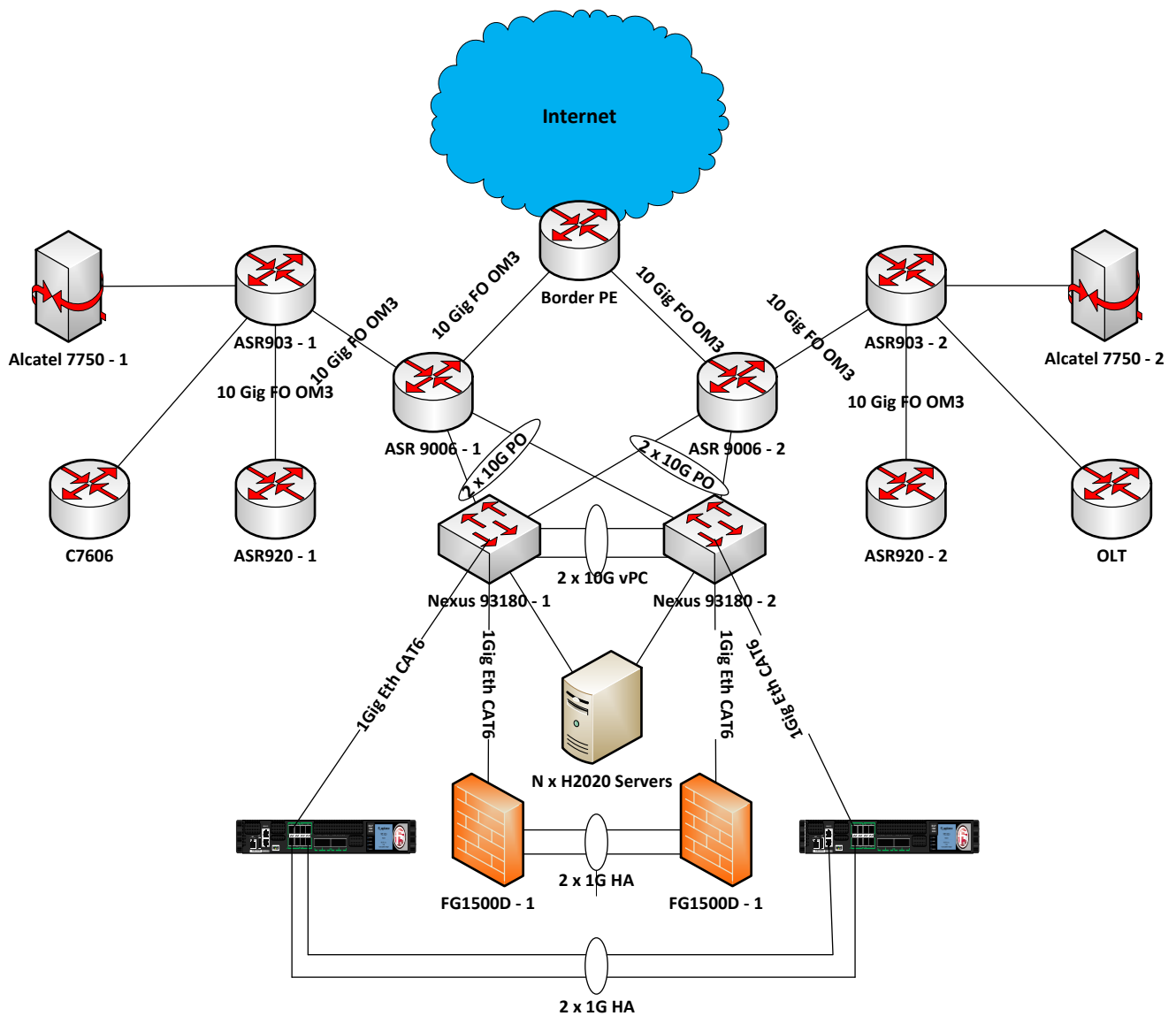


Figure 10: ORO testbed

A scheme of the testbed is shown in Figure 10. ORO use-case will be tested on the testbed deployed to cover the testing needs for the ORO fixed-mobile live network. Also 5G is being implemented and tested on the respective testbed.

The following mapping can be used in order to link the described elements to the elements presented in Figure 10. In brackets will be indicated to which system components collected in the excel templates this is corresponding (see section 2.3.1).

- The two Nexus switches are only used for physical – Layer 1 – connectivity. From a service provider perspective you can view them as cables interconnecting the various equipment. (SC3)
- The following elements (routers) will form a fully functional MPLS Network :
 - ASR9006 (SC1)
 - ASR920 (SC4)
 - ASR903 (SC10)
 - Alcatel 7750(SC1)
 - Cisco 7606 (SC1)
- From an MPLS perspective the following mapping will be used :
 - ASR 9006 – Datacentre Gateway Routers / PE Routers + Route Reflectors
 - ASR 920 – Mobile Network Access Router / PE Router
 - ASR 903 – Mobile Network Distribution Router / PE Router
 - Cisco 7606 – Fixed B2B client distribution router / PE Router
 - Alcatel 7750 – Mainly P Backbone Routers
 - OLT – Non-MPLS Broadband client aggregation
 - F5BIGIPs – Non-MPLS application delivery controllers in the Datacentre
 - FG1500 Firewalls – Non-MPLS next generation firewalls in the Datacentre (SC5)
- From a Datacentre Perspective all the servers can run either VMWare vSphere or Openstack virtualization technologies.
- Services such as MPLS Layer 3 or Layer 2 VPNs can be delivered between/from any PE router described above.

To allow a good modelling from the testbed network, and use collected information in the excel templates, the given components for the testbeds have to match with given system components in the excel templates. These matchings are sometimes missing and need to be provided by the telecommunication operators.

In the case from ORO, The components declared in the excel template cover the real network, on top of which ORO provides the whole mobile and fixed services from its portfolio. Nevertheless most of components exists or could be simulated on the testbed infrastructure, even if it is a small scale replication.

Errore. L'origine riferimento non è stata trovata. present the correspondence between excel components and testbed components from ORO.

| Name (Real Network) | Testbed |
|--|--|
| Border Routers | Cisco ASR Series Routers |
| FO Infrastructure | FO OM3 connections |
| Mobile Switching Centers (MSC) | vEPC running on OpenStack |
| Radio Infrastructure (BTS, BSC, RNC, NodeB) | eNodeB using OpenAir Interface USRP X310 RRU |
| Network Security Equipment (IPs, FWs) | Fortinet FG-1500D |
| Workstations and Servers | Virtual machines on HP ProLiant DL300 Series (configured as Workstations and/or Servers) |
| Microsoft Security Domain | Simulated MSD on Virtual Machines |
| Business Applications | Simulated Business Applications installed on Virtual Machines |
| Equipment Shelters | Testbed room (laboratory) inside offices building with badge based access (situated underground) |
| Mobile Core Network | vEPC running on OpenStack |

Table 3 Correspondence between excel components and testbed components from ORO

As explained in section 2.3.3 and section 3.3 to understand the consequences of a threat on the network, and compute this in terms of a resilience matrix, information about the various performance functions of the network have to be calculated using theoretical knowledge. In the case of the testbeds, in order to calculate the impact of a threat, it would be useful to have the further information on the following items:

- The capacity of the fibre optic capacity.
- The mean demand and the peak demand on each link of the network.
- Which service function from the network use which system component from the testbed.
- Which location (geo coordinate) and population size are represented with which component of the testbed (as example: this component represent a core router of a large city).

In ORO's example:

- The capacity of the connections between components is as follows:
 - 1Gbps/10Gbps between all components
 - 1Gbps uplink
- The peak traffic could be considered as follows:
 - 75% from maximum capacity for 1Gbps links
 - 30% from maximum capacity for 10Gbps links
- The mean traffic could be considered as 50% from maximum traffic in both cases
- The capacity of the routers corresponding to the population size is as detailed below
 - Alcatel 7750 Series routers – high capacity Core router (aggregates the traffic from all access routers)
 - CISCO 76xx Series router - high capacity (for large cities > 150 000 inhabitants)
 - CISCO ASR 9xxx Series routers- medium capacity (for medium cities, number of inhabitants between 10 000 and 150 000)?
 - CISCO ASR 9xx Series routers- small capacity (small cities < 10 000 inhabitants)?

The same job has to be done with the threats presented in the use cases. As example for the ORO testbed the following threats are going to be simulated. The brackets indicate to which threat collected in the excel templates this is corresponding (see section 2.3.3).

- DoS Attack from inside the security perimeter (T1)
- DDoS attack on border (T1)
- DDoS attack on peering point (T1)
- Routing Table Poisoning on core network (T?)
- Routing Table Poisoning on access-side network (T?)
- BGP Hijacking (T2)
- Detection of botnet infection of end-points inside the core network (T?)
- External Network Scanning of internet-facing assets and lateral movement attempts (T?)
- Power Outage in MSC Site (T4)
- Link disruption (cable cut) between RAN components (T3)
- Rouge access to MSC Site (break-in) where a human actor physically disables communications links on the access side of OROs network
- Rouge access to ORO's datacentre(s) where a human actor physically disables cyber-detection and protection equipment such as UTMs/Anti-DDoS/Proxies etc.

Here too, further specific information for the cyber-threats and in particular their effects on the testbed components would be helpful (see also section 3.3.5).

In the next section, a description and plans for the simulation tools from the LTCL and the STCL will be given, in order to explain how testbeds and use cases will be implemented by these tools.

5. DESCRIPTION AND PLANS FOR SIMULATION TOOLS

5.1. Description of the EMI simulation tool CaESAR



CaESAR (**C**ascading **E**ffects **S**imulation in urban **A**reas to assess and increase **R**esilience) is a simulation tool for computing cascading effects within critical infrastructure and especially across infrastructure borders, i.e. in interdependent infrastructures [29]. The overall target of the CaESAR tool is to find weaknesses in the interdependent infrastructures, to find optimized strategies to overcome the weaknesses and to increase the resilience of those interdependencies. Up to now, CaESAR includes the power grid, the water grid and a part of the mobile phone grid.

The computation in CaESAR consists of two loops:

1. setting connections between different types of critical infrastructures (interdependencies)
2. simulating damages, the consequences, resilience computation and mitigation strategies

For setting the interdependencies, CaESAR takes the single infrastructures and computes the possible interconnections in-between the different infrastructure types, e.g. between water and power grid. The computation depends on two conditions. First, the type of the components, e.g. not every component in the water grid needs electricity. Second, the distance between the infrastructure components, because interconnections between components with a big distance are very unlikely. Within a defined radius a connection of two components in different CIs is more likely. This radius is called dependency radius as shown in Figure 11. This means that for every possible connection in-between infrastructures a probability is allocated. This probability is very high for connections within the dependency radius and small for connection outside the dependency radius.

As shown in Figure 12, the connections between different CIs are set n times. For each connection configuration, the inner simulation loop shown in Figure 12 takes part. After n repetitions of the dependency loop, the average system answer is computed for the interdependent CIs and a result is given to the user.

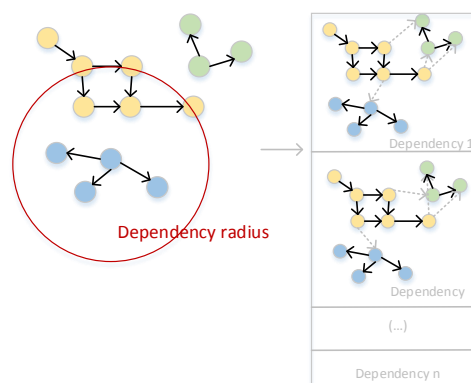


Figure 11: The dependency radius describes the probability of connection in-between CIs

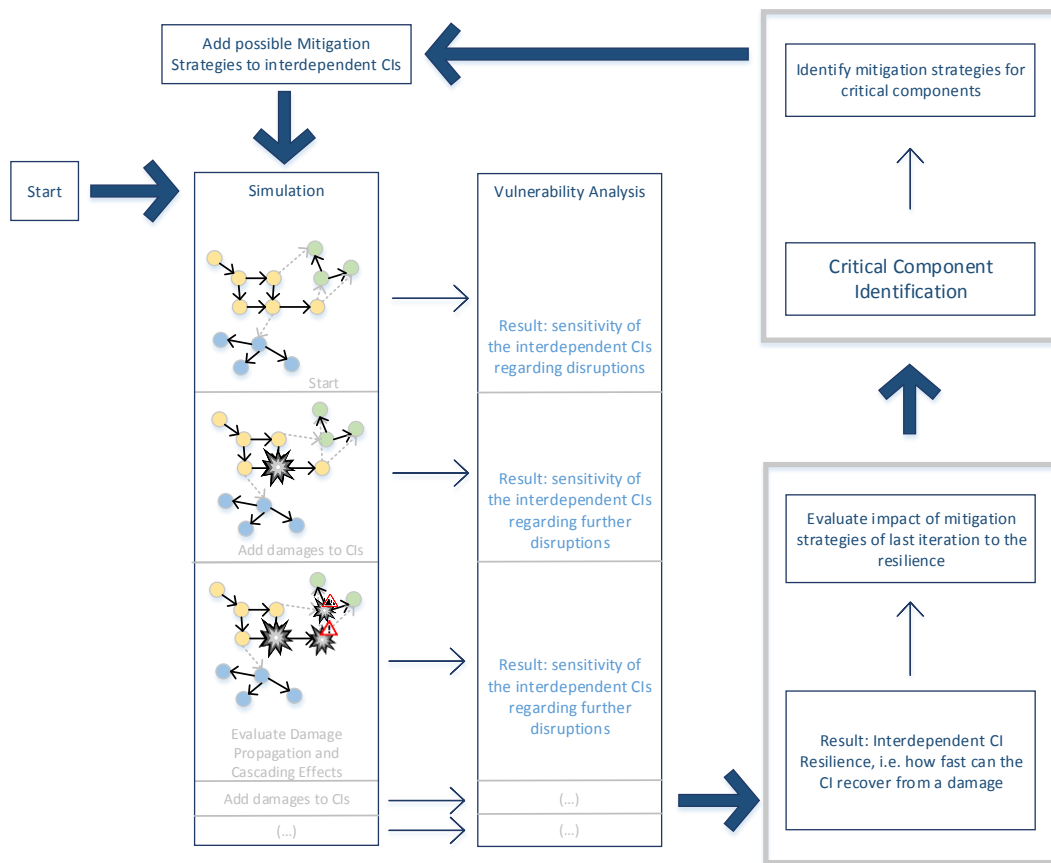


Figure 12: Overview over CaESAR simulation tool. Damages are introduced, the vulnerability of the interdependent infrastructure is analysed, critical components are identified and mitigations are applied to them.

Figure 12 gives an overview over the inner simulation loop. For each repetition of the inner simulation loop, the simulation is executed with the target of identifying weaknesses in the interdependent infrastructure and of finding mitigations to overcome them.

For achieving this, CaESAR implements two types of damages: First, a threat-based damage resulting from modelled events, e.g. flood or earthquake. With the threat-based damage model, it is possible to introduce damages in specific areas. The second type is a generic damage model, where single or multiple components fail according to a defined attack strategy describing the order of component removal in the infrastructure, e.g. remove well connected components or remove components in random order. Based on the generic damage model, general weaknesses in the interdependent infrastructure can be identified. The generic damage model includes also a time-dependent component removal, i.e. a single component or a set of components is attacked in one time step and the next set is removed in a further time step. After each component removal, generic or threat-based, the consequences on the interdependent infrastructure are evaluated with a time-dependent model.

This evaluation builds the base for the identification of components which strongly contribute to the infrastructure functionality and where a failure leads to severe consequence, i.e. to a significant

reduction of resilience. These critical components are used for applying mitigation strategies for increasing the infrastructure resilience.

The resilience is computed time-dependent as shown in Figure 13. After the crisis event impinges on the interdependent infrastructure system, it may provoke effects on the infrastructure. After some cascading effects within the system, an impact on the infrastructure functionality is given and the system loses functionality. It reaches the state of lowest functionality in respect to the given damage model. After the first recovery actions, the system could still stay in a state of low functionality for some time. Subsequently, the recovery actions have an impact on the infrastructure functionality and it increases. The actions and the increase of functionality take part until the infrastructure is fully functional again. Based on this curve, CaESAR evaluates the vulnerability of the system in each time step and builds a resilience value. The resilience value serves then as base for further simulation loops.

After the computation of resilience, the inner simulation loop is repeated with the applied mitigation strategies. In the repetition, the impact of those strategies on the resilience are evaluated and the next set of mitigation strategies is applied to the interdependent infrastructure.

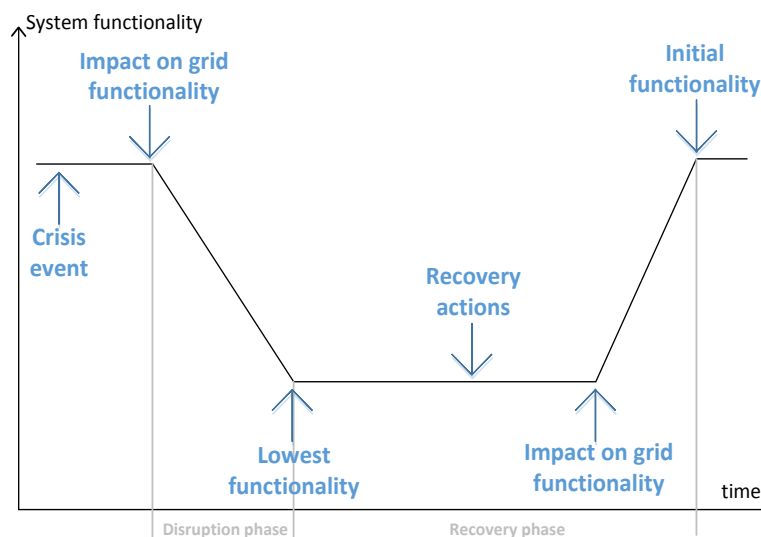


Figure 13: The different states of a system influencing the resilience.

Figure 14 shows an example for results of CaESAR. The result shows weakest points, where mitigation strategies have the best effect and suggests a good strategy.

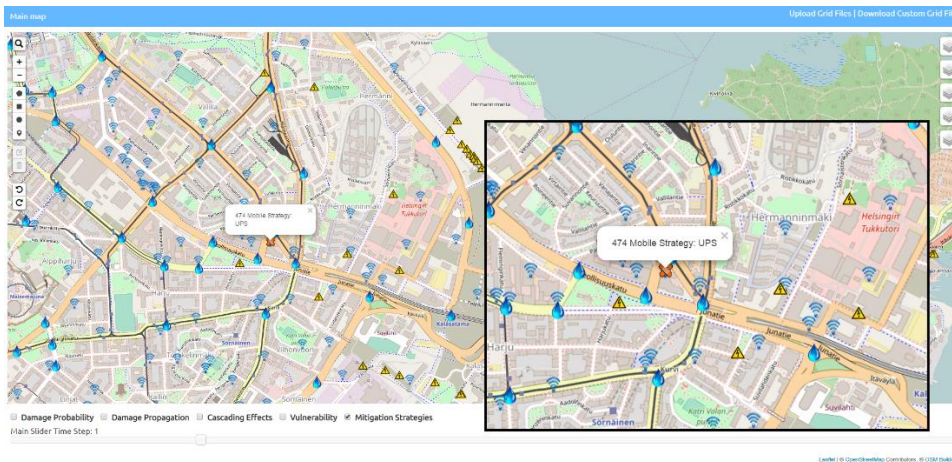


Figure 14: Screen of CaESAR computation results: mitigation strategies

5.1.1. Implementation plan for CaESAR

The CaESAR tool was developed to simulate interconnected power, water and communication networks. For each of the three networks, a simplified model was used to set up the networks and their interconnections. The general idea for the RESISTO project is to use this implementation as a starting point for a more precise modelling and simulation of the telecommunication infrastructures. Therefore, further relevant parts need to be modelled and integrated, e.g. protocols, 3G, 4G and 5G specifications.

The software design of CaESAR is modular, allowing for adaptations and extensions of the simulated networks. The main challenge for the modification of the communication network is to obtain necessary information about the structure and level of details to be implemented. Input for this challenge is discussed in sections 2, 3 and 4 in this report, and the following adaptations are being implemented:

- Setting parameters to characterize the nodes as system components, and describe this function in the network (routers, switches, servers, firewalls...).
- Setting parameters needed for the computation of network performances (throughput, delay, packet loss).
- Setting links between network performances and system performance function (as Voice Services, Mobile Data Services...) in term of acceptable Quality of Services
- Threats are being implemented. For cyber-threats, implementations are in development phase, in order to have realistic simulations for cyber-attacks consequences.
- Resilience computation for various network performances and system performance functions are being developed. In Figure 15, an exemplary illustration of the matrix of different resilience curves is presented. This is an exemplary illustration and not based on real data.

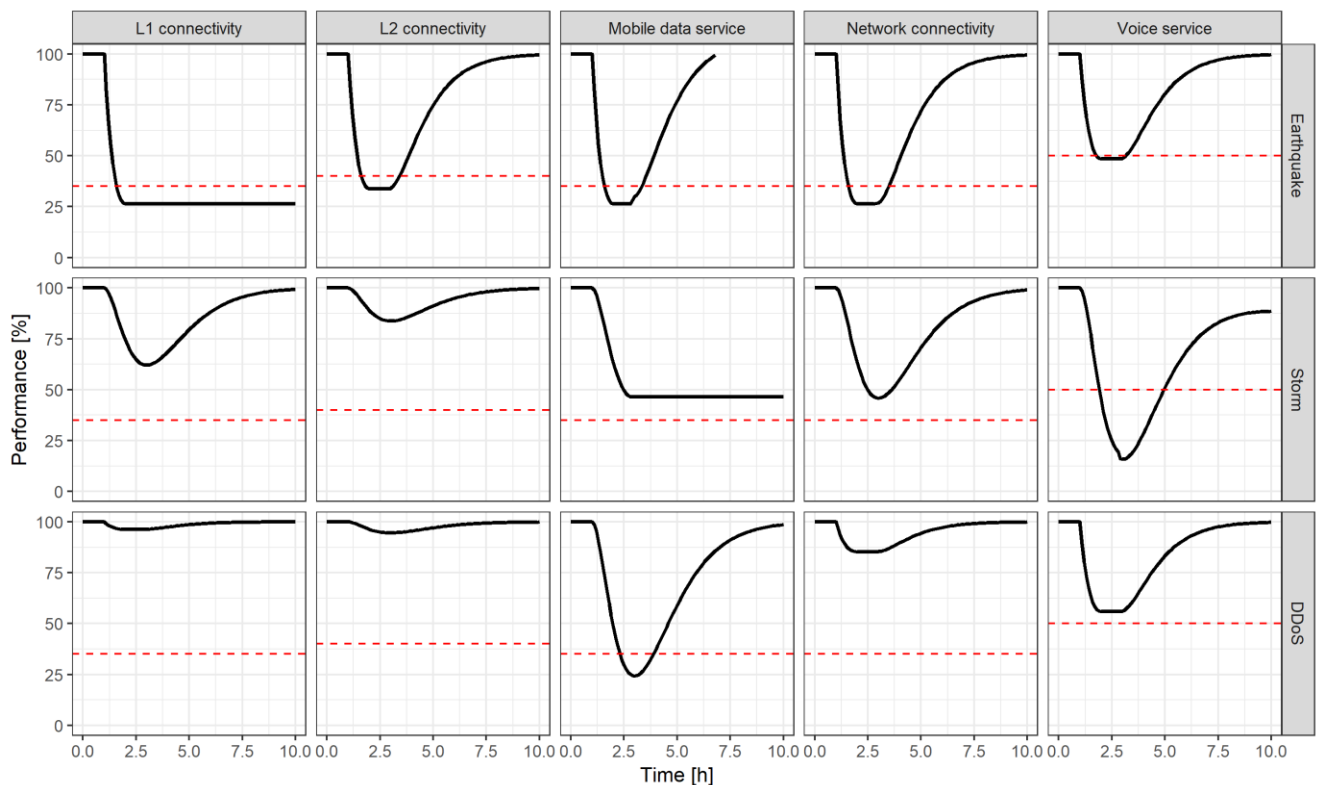


Figure 15: Exemplary visualization of a resilience matrix (performance metrics vs threats). The performance functions and threats were taken from real inputs but the curves are not realistic, i.e. not based on real data or a realistic simulation.

- Mitigation strategies and their quantification in terms of benefit for the resilience of the network are explored. Mitigation strategies are developed for all the phases from a resilience cycle. For example:
 - For the defend phase: Installation from Anti-DDoS appliance, Information Security Training and awareness sessions for new employees and periodically after, installing video surveillance camera at sensible zones. The purpose of these measures is the reduction of influence by such threats. If the threats are fend off with no impact on the system, the resilience increases. In other word, the probability that a threat impacts the system will be lower.
 - For the remediate phase: There are measures like having more personnel to repair infrastructure damaged by storm, or to have more system components in reserve by the sites, to replace eventual defect system components. The purpose of these measures is to decrease the mean time to repair so that the networks can recover their usual capacities as soon as possible.
 - For the recovery phase: There are mitigation strategies as adding redundancy on the network or increasing the capacity from some links. The aim of these strategies is to reduce the consequences of a threat on the network performances.

In addition to the network adaption, three major development-points for CaESAR were identified in the intermediated version of the report (D3.5). The current status of them is described in the following:

- There are issues with the current implementation of the GUI for presenting the simulation results, as shown in Figure 14. A new implementation of the GUI or desktop application was considered as best solution. Currently results returned by the simulation can be visualized via a leaflet map in R. This can be further implemented in the Shiny app of the LTCL as described in D3.1. If this implementation is sufficient needs to be discussed in the implementation phase.
- The simulation of large networks with a high multiplicity of connections is computationally intensive. Therefore, the integration of a cluster or cloud service for parallel computing was investigated. Since we are working on very small testbeds for the development phase, this is not a priority any more. In a next step, if the RESISTO platform is used for analyzing complete telecommunications networks, with thousands of nodes and arcs, this point must be reflected.
- In respect of the RESISTO objectives, the demand of simulations and the representation of results needs to be adjusted to follow the developed risk and resilience approach. This allows to integrate the tool into the long term control loop. As discussed above, the representation of results can be easily added to the Shiny app currently available for the LTCL.

5.2. Description of RM3 tool CISIApro

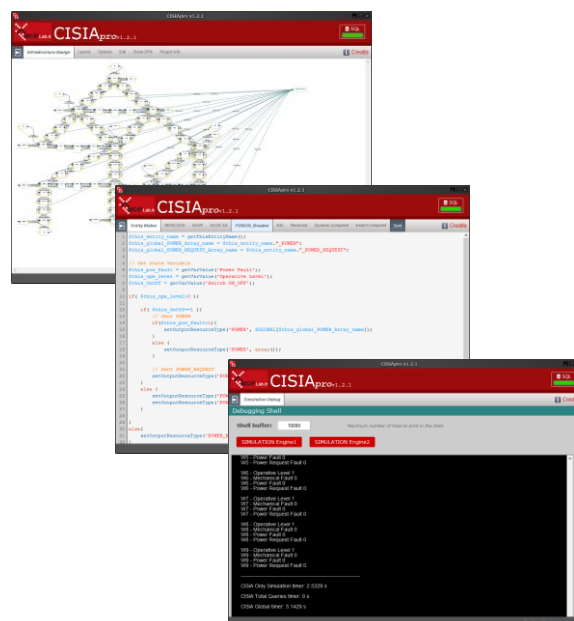


CISIApro simulator (Critical Infrastructure Simulation by Interdependent Agents) has mainly been designed for analyzing the short-term effects of failure both in terms of faults propagation and with respect to performance degradation.

However, modelling complex systems interaction in CISIApro means deeply analyze possible threats and identify vulnerabilities in the preliminary risk analysis and

implementation. For this reason, CISIApro is a useful tool also for analyzing long-term risk effects which may help domain operators to improve resilience and redundancy in CIP contexts.

Typically, Risk Management deals with the use of mathematical techniques not always able to handle the dynamic associated with the risk evolution. The main objective of the proposed framework, is to provide a flexible methodology and software able to exceed the limits of other existing approaches, achieving a proper level of complexity. From this perspective, CISIApro represents a good solution to assess risk due to resources/failures/capabilities propagation also considering cascading effects.



It should be noted that CISIApro has introduced efficient ways to model, execute and debug simulations and cascading effects. In particular, an intuitive Graphical User Interface is provided to create entities and connect them in an easy way. After the creation of the entities with their interconnections (i.e., interdependencies) and its exchanged resources, the users need to implement the behavior of each entity. In CISIApro, the adjacency matrices, representing the interdependencies among entities, are generating during the design phase. During the simulation, the matrices are represented as queue data structures for fast computing.

CISIApro is designed using particular programming techniques which allow use of common programming languages like C/C++ and languages that are used to create web/cloud platforms. Although it might seem “a controversial choice”, it support a high productivity, usability and scalability along with the capability to integrate third parties software in the same architecture.

For the implementation of the CISIApro simulation engine, a combination of PHP language (server-side programming language) with C++ compiling techniques was adopted. This is possible because PHP libraries was created through a C/C++ implementation. The main difference between C/C++ and PHP lies in the fact that C/C++ is a compiled language while PHP is an interpreted language. Thanks to PHP interpreter, inside CISIApro, it is possible to implement all the behaviors and mechanisms of a modelled entity.

Below, some typical advantages of implementing entities using the PHP programming language are summarized:

- to instantiate a variable in PHP it is sufficient to assign a value;
- the declaration of the variable type is implied when assigning a value;
- a variable can also be removed in the course of the script;
- a variable "type" can be changed during the script execution;

- it is possible to use object-oriented programming;
- PHP implements more than 90% of C/C++ functions without mentioning the countless available classes developed by its community.

The actual state of the agent is summarized through the operational level concept: the operational level is the ability of the agent to perform its required job; it is an internal measure of the potential production/service, if the operative level is 100% it does not mean that it is providing the maximum value but that it could, if necessary.

Agent inputs and outputs are necessary in order to perform interactions among agents. There are three kinds of inputs and, similarly, three kinds of outputs:

1. **Induced/propagated faults:** faults propagated to the considered agent from its neighborhoods and from the considered agent to its neighborhood.
2. **Input/output resources:** amount of resources requested by/to other objects.
3. **Induced/propagated capabilities:** reaction or mitigation strategies are propagated as positive propagation effects to the considered agent from its neighborhoods and from the considered agent to its neighborhood.

In CISIApro, the agent dynamic is described as an input/output model among the previously listed quantities. This description of agent's behavior is highly abstracted but it is rich enough to leave the experts to model the model dynamics in the most appropriate way.

The relations among agents are based on their interdependencies, and they are described by incidence matrices. In fact, each matrix is able to spread a different type of interdependency, following the classical methodology among physical, geographical logic, and cyber connection.

5.2.1. Resilience time calculation using CISIApro approach

Resilience concept feeds a lot of discussion boards increasing confusion due to the complexity in estimating, in advance, the real capacity of a complex system to restore normal functionalities immediately after specific critical situation. Such a parameter gets even more essential in a TELCO Critical Infrastructure where provide a high quality of service became a key factor from the Business Continuity and Risk Assessment.

Among the main objectives of RESISTO there are concepts not only related to prevention but also focused in handling critical situations in a real-time context. From this point of view a correct resilience estimation must be compared with a real-time resilience performance monitoring. This dual perspective reflects the dual objective of:

- Compare a priori resilience estimation (performed into the LTCL) with a global real-time resilience indicator (performed by CISIApro engine into the STCL);
- Give the capability to continuously improve preparedness in case of critical situations.

Although it is a forced comparison, based on engines which use different mathematical framework and different source data (see section 3.3.1), the final estimation allows a RESISTO operator to produce a useful evaluation with respect to recovery procedures suggested by RESISTO platform.

With this aim, in CISIApro view (Figure 16) it was introduced, on the top right of the synoptic dashboard, a real-time graph which take into account some important Key Performance Indexes (KPI) calculated through using the CISIApro engine.

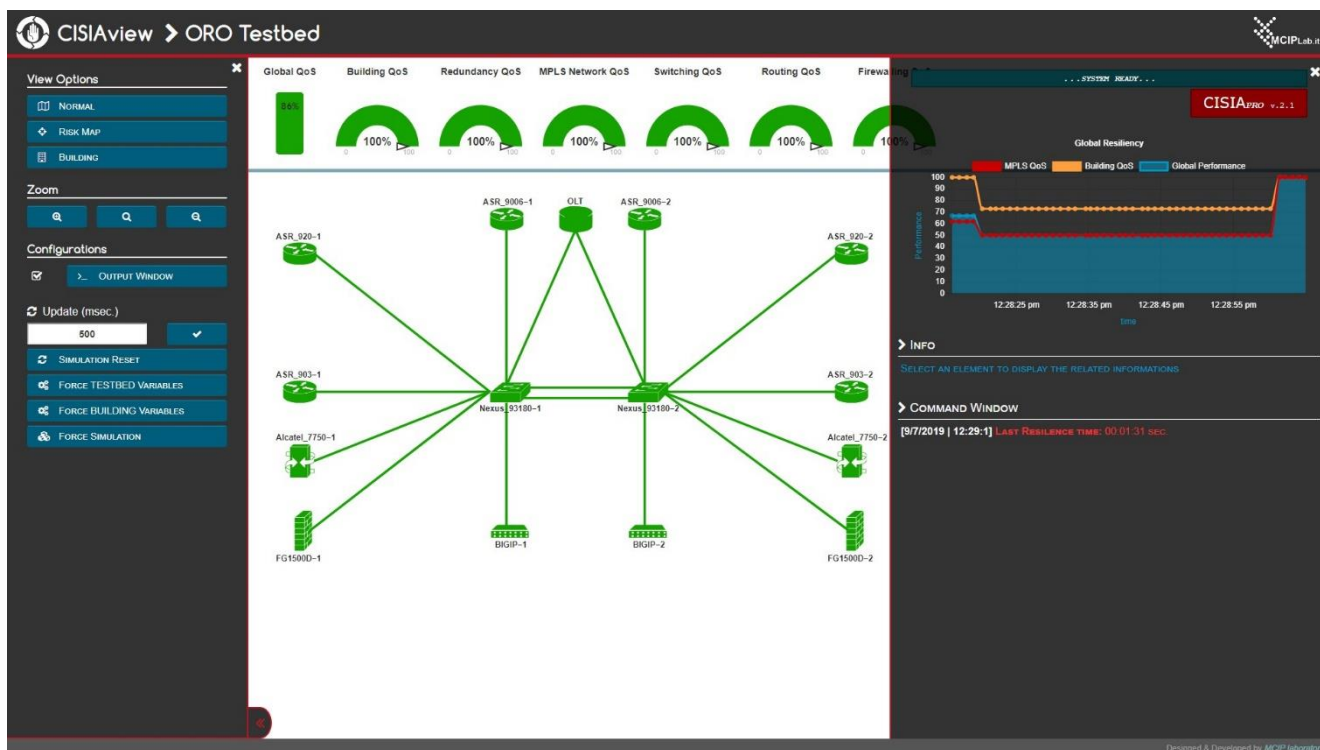


Figure 16: CISIAview Orange Romania Case Study



Figure 17: CISIAview Real-Time Resilience graph detail

In this particular case study, as shown in Figure 17, using the WHAT-IF Analysis feature, provided by CISIApro simulation, it was possible to force some critical scenarios. Figure 17 underlines a detected reduction of global performance in the ORO Testbed [Global QoS KPI blue line] due to a critical situation in which:

1. a Cyber Attack was performed [MPLS QoS KPI red line];
2. a physical damage occurs into the building which housing TELCO equipment [Building QoS KPI yellow line];
3. system was completely restored.

We notice that, at the end of this process, a resilience time was calculated and shown to the TELCO operator. Thanks to this kind of parameter it was possible to compare the resilience time estimated by LTCL with the resilience time provided by the STCL (see section 5.3). Of course, this final resilience estimation it will be strictly correlated to the real data that are injected in the RESISTO Platform.

5.3. Validation of the simulation results

The RESISTO platform operates with two different control loop, the Long Term Control Loop (LTCL) and the Short Term Control Loop (STCL). The LTCL contains the resilience analysis and simulations on the system after an event occurs. The loop occurs offline, investigating the vulnerabilities of the infrastructure and defining methods to improve the resilience. The cycle is not continuously running, occurring only periodically as needed. On the other hand, the STCL works in real time monitoring the system for irregularities. The STCL also investigates how much of an impact an event had, and once response measures are selected, can create action workflows to help mitigation. The results of the LTCL simulations and real events from the STCL can be related through the definition of resilience indicators (RI) which are extracted from the resilience curve (Figure 18).

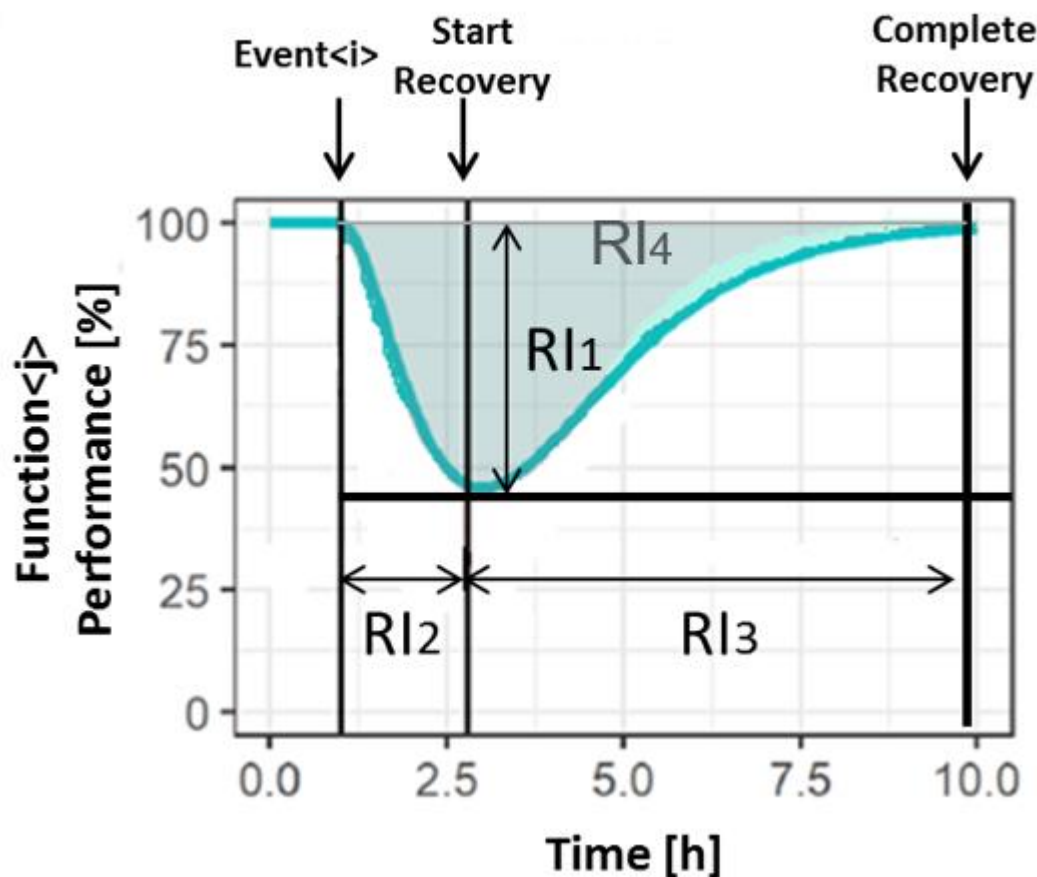


Figure 18: The performance time curve with different resilience indicators defined.

Throughout the performance time curve of the system different RI can be defined (Figure 18). For example, resilience indicators are defined for the maximal performance loss (RI1), the time it takes for the recovery process to begin after an event occurs (RI2), the time it takes for the system to reach full performance again after the recovery actions have started (RI3) and the total (integrated) performance loss from the event starting to the complete recovery (RI4). However, it is possible to have different definitions for the indicators, leading to more freedom in the analyses being conducted on the system of interest.

Figure 19 depicts the RI flow within the RESISTO platform. The RI are what links the two control loops together. The LTCL estimates the RI while the STCL measures the actual values. The resilience indicators from each control loop are then compared and utilized in the next iteration of the LTCL cycle. In this sense, the STCL monitoring results can validate the LTCL simulation results and the STCL measures can be used to improve the LTCL resilience computation. The storage of RIs in the knowledge base would follow a matrix presentation as discussed in Section 5.1 and is schematically presented in Table 4.

It is important to note that the higher level RI loop not only allows improvement of the RI estimation but, mainly, it allows a continuous CI resilience monitoring and improvement.

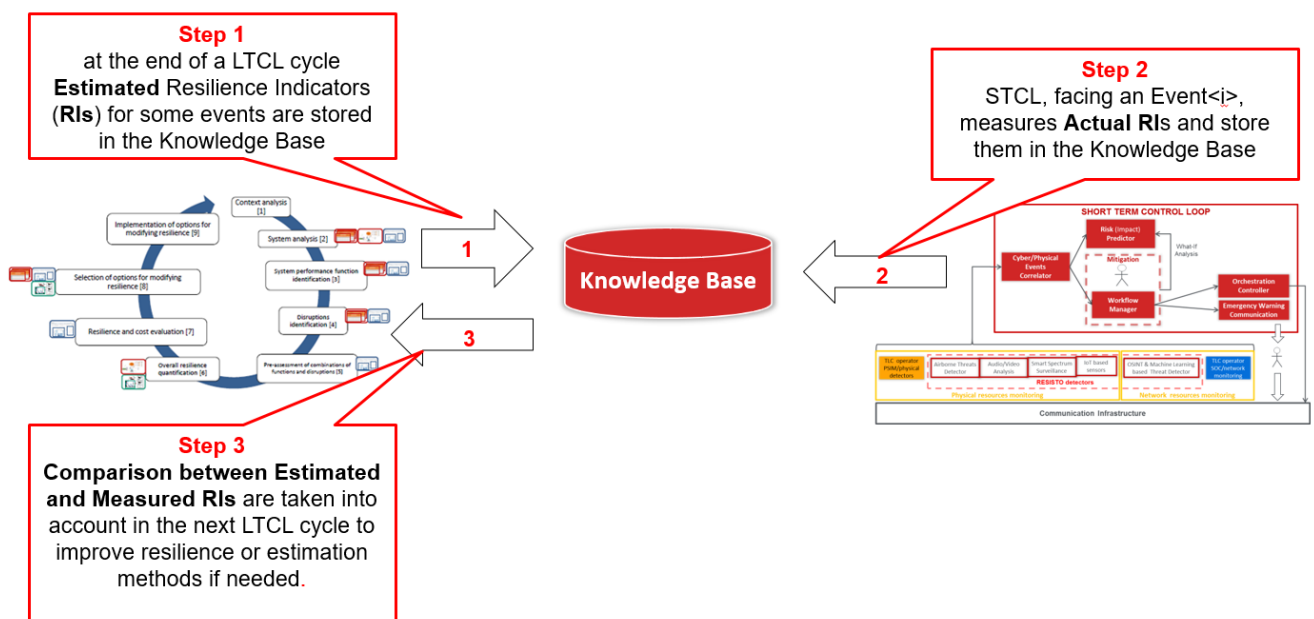


Figure 19: LTCL and STCL cycle relationship and resilience indicators flow.

Table 4: Storage format of RIs in the Knowledge Base of the RESISTO platform, based on the matrix structure of potential events and performance functions.

| | Event<1> | ... | Event<i> | ... | Event<M> |
|----------|----------|-----|--|-----|----------|
| Funct<1> | | | | | |
| ... | | | | | |
| Funct<j> | | | Estimated RI(1;i;j) RI(2;i;j) RI(3;i;j) | | |
| ... | | | | | |
| Funct<N> | | | | | |

5.4. Requirements traceability

In task 2.1 (D2.1), the operators and technical experts of the consortium collected a large number of requirements that the RESISTO platform shall, should or could comply with – the modal verb stating the level of technical readiness of the platform. 38 of them have been labelled “SHALL”, which means the implementation of those requirements is mandatory for a TRL7 prototype. “SHOULD” requirements are mandatory for TRL 8/9 and “COULD” requirements are not mandatory at any level but could improve performance and functions of the platform.

During the design and development phase of the RESISTO platform, all “SHALL” requirements have been and will be traced and monitored. If a requirement cannot be covered, the reason for that will be analysed and appropriate recovery actions will be identified.

The goal is a traceability matrix at the end of the project, where the completeness of the implementation of the requirements can be demonstrated together with the information about why, where and how they were implemented.

To ease the complexity of that task, the tracing of requirements will be carried out in each relevant deliverable to come, where the “SHALL”-requirements will be matched with the methods, tools, cases etc. described in that document.

In this deliverable, a contribution to the Mandatory Requirements listed in Table 5 has been identified.

| Mandatory Requirements (D2.1, section 6.3) | | |
|--|--|------------------------|
| Requirement Identity Code | Requirement Description | Related to method/tool |
| RES_FUN_0005 | Resisto shall exploit the outcomes of the cyber security and the physical systems of the TLC infrastructures (if existing). | LTCL/CaESAR |
| RES_FUN_0070 | Resisto shall suggest to the operator the necessary steps to mitigate effect of a cyber/physical attack. | LTCL/CaESAR |
| RES_FUN_0570 | The risk and resilience assessment analysis shall also take into consideration network single point of failures nodes, using networks metrics. | LTCL/CaESAR |

Table 5: Mandatory requirements related to the tools/methods described in this report (D3.6).

6. SUMMARY

This deliverable reports the final status of T3.3 of WP3 at the end of its runtime. It is based on the intermediate report D3.5, which was extended and improved for various aspects (see Introduction).

Aim of the task is to develop simulation tools to assess the effect of disruptions in telecommunication CIs. To this end, two tools are provided and further developed for the RESISTO project: CaESAR from EMI and CISIApro from RM3 (see Section 5).

The main challenge for the use of these tools is the collection of necessary network model specifications. Various sources for information are addressed within this report:

- Excel templates (Section 2): provides information on which types of threats need to be considered, which system components and functions are affected and which possible mitigation options can be analyzed. An analysis from all returned templates has been done and setups for the event simulation are derived.
- Network simulators (Section 3): provides information how the telecommunication network is simulated by other companies and institutions. The specificity from the tools developed for the RESISTO project are also presented.
- Network schemes (Section 4): provide direct input on the network implementation, i.e. nodes and links of all relevant sub-networks. A discussion of testbeds representation and use for the simulations tools has been done.
- Description and plans for simulation tools (Section 5): provide direct input on the two simulations tools (for LTCL and STCL). The link between the two simulations tools was clarified.

6.1. Discussion and outlook

As discussed throughout this report, further information and interdependencies from collected information need to be explored in order to produce realistic simulation results.

- The impact of the threats on the network performance and mitigation options on the resilience gain have to be explored. In particular for the cyber-threats.
- The impact, from the network performance indicators (throughput, delay, loss of packets) decrease, on the network functions in term of quality of services have to be further investigated.

We plan to refine specific setups for each use case and testbed in strong collaboration with the corresponding operator during the implementation phase (WP7-WP9). This means that the list of threats and performance functions of interest might be updated to a smaller but more precise list with complementary information. In case the information is not sufficient to make realistic simulations of rather complex performance functions (such as Voice Service or security functions), the analysis will be restricted to simple performance measures such as network availability.

The simulation tools themselves are constantly improved throughout the implementation of the RESISTO platform and features added when needed. The representation of simulation results of the LTCL will be further discussed in T3.1 (D3.2) and T3.5 (D3.9).

References

- [1] 2006 *Measuring impact of DoS attacks Proceedings of the DETER community workshop on cyber security experimentation* (Citeseer)
- [2] NS2 <https://www.isi.edu/nsnam/ns/index.html>
- [3] Issariyakul T and Hossain E 2012 Introduction to Network Simulator 2 (NS2) *Introduction to network simulator NS2* ed T Issariyakul and E Hossain 2nd edn (New York: Springer) pp 21–40
- [4] Ibrahim F H *Network Simulator 2: a Simulation Tool for Linux* <https://www.linuxjournal.com/article/5929>
- [5] OMNET++ <https://omnetpp.org/intro/>
- [6] Riley G F and Henderson T R 2010 The ns-3 Network Simulator *Modeling and Tools for Network Simulation* ed K Wehrle et al (Heidelberg: Springer) pp 15–34
- [7] Henderson T R, Lacage M and Riley G F Network Simulations with the ns-3 Simulator <http://conferences.sigcomm.org/sigcomm/2008/papers/p527-hendersonA.pdf>
- [8] NS3 Development team *NS3 Wiki* https://www.nsnam.org/wiki/Main_Page
- [9] *Riverbed Modeler* (Riverbed (OPNET))
- [10] K'oksal M M *A Survey of Network Simulators Supporting Wireless Networks*
- [11] *QualNet Network Simulator Software* (scalable-network technologies)
- [12] *Network Simulation Tools: OneSim Simulator* <http://networksimulationtools.com/onesim-simulator/>
- [13] *The ONE: The Opportunistic Network Environment simulator* <https://akeranen.github.io/the-one/>
- [14] *PEERSIM* <http://networksimulationtools.com/peersim/>
- [15] Jelasity M, Montresor A, Paolo Jesi G and Voulgaris S *PeerSim: A Peer-to-Peer Simulator*
- [16] Baran P 1964 *On Distributed Communications: I. Introduction to Distributed Communications Networks* (RAND Corporation)
- [17] Drzewiecki L and Antoniuk-Lewandowska M 2008 Flow Simulator - a flow-based network simulator *Eurocon 2007 - the international conference on "computer as a tool." EUROCON 2007 - The International Conference on "Computer as a Tool" (Warsaw, Poland, 9/9/2007 - 9/12/2007)* ed I O E A E Engineers ([Place of publication not identified]: John Wiley) pp 2132–6
- [18] Anggono G and Moors T 2015 FLEO: A flow-level network simulator for traffic engineering analysis *25th International Telecommunication Networks and Applications Conference (ITNAC) 2015 International Telecommunication Networks and Applications Conference (ITNAC) (Sydney, Australia, 11/18/2015 - 11/20/2015) (Piscataway, NJ: IEEE)* pp 131–6
- [19] Anggono G and Moors T 2017 *IEEE Commun. Lett.* **21** 496–9
- [20] Mehrdad S, Mousavian S, Madraki G and Dvorkin Y 2018 *Current Sustainable/Renewable Energy Reports* **5** 14–22
- [21] Carlson J L, Haffenden R A, Bassett G W, Buehring W A, Collins III M J, Folga S M, Petit F D, Phillips J A, Verner and Whitfield R G 2012 *Resilience: Theory and Application* (Argonne National Lab.(ANL), Argonne, IL (United States))
- [22] Trimintzios P 2011 *Measurement frameworks and metrics for resilient networks and services: Technical report, European Network and Information Security Agency (ENISA)* (Tech. Rep., February)
- [23] Andreas Betker, Inken Gamrath, Dirk Kosiankowski, Christoph Lange, Heiko Lehmann, Frank Pfeuffer, Felix Simon and Axel Werner 2014 *J. Opt. Commun. Netw.* **6** 1038–47
- [24] Lange C, Kosiankowski D, Betker A, Simon H, Bayer N, Hugo D von, Lehmann H and Gladisch A 2014 *Journal of Lightwave Technology* **32** 571–90
- [25] Andrews J G, Buzzi S, Choi W, Hanly S V, Lozano A, Soong A C K and Zhang J C 2014 *IEEE Journal on selected areas in communications* **32** 1065–82
- [26] Chávez-Santiago R, Szydelko M, Kliks A, Foukalas F, Haddad Y, Nolan K E, Kelly M Y, Masonta M T and Balasingham I 2015 *Wireless Personal Communications* **83** 1617–42
- [27] Condoluci M and Mahmoodi T 2018 *Computer Networks* **146** 65–84

- [28] Morgado A, Huq K M S, Mumtaz S and Rodriguez J 2018 *Digital Communications and Networks* 4 87–97
- [29] Hiermaier, Stefan, Sandra Hasenstein, and Katja Faist 7th REA Symposium 2017
University of Liège, Belgium