

RESISTO:

D3.5_Damage/Vulnerability models for physical and cyber threats of telecom CI



RESISTO

D3.5 – 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_D3.5_190516_01	Version:	1.0
Deliverable:	D3.5	Date:	16/05/2019
		Status:	APPROVED

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

Approval Status	
Prepared by:	Mirjam Fehling-Kaschek (Fraunhofer)
Approved by: (WP Leader)	Mirjam Fehling-Kaschek (Fraunhofer)
Approved by: (Coordinator)	Federico FROSALI (LDO)
Advisory Board Validation (Advisory Board Coordinator)	NA
Security Approval (Security Advisory Board Leader)	Alberto BIANCHI (LDO)

CONTRIBUTING PARTNERS

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

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
0.9	20.12.2018	All	All	Release for SAB review
1.0	16.05.2019	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 delle Officine Galileo 1 – Campi Bisenzio (FI) – 50013 – Italy

Tel.: +39 055 5369640, Fax: +39 055 5369640

E-Mail: frederico.frosali@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 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.

CONTENTS

1. INTRODUCTION	10
2. REVIEW OF POTENTIAL HAZARDS AND DISRUPTIONS	12
2.1. Evaluation of the Excel templates	12
2.2. Features needed to simulate potential disruptions in telecommunication infrastructures	12
3. REVIEW OF AVAILABLE SIMULATION TOOLS	14
3.1. Network simulation tools	14
3.1.1. NS2.....	15
3.1.2. OMNeT++.....	15
3.1.3. NS3.....	15
3.1.4. OPNET riverbed	15
3.1.5. QualNet	15
3.1.6. OneSim.....	16
3.1.7. PeerSim	16
3.2. Packet-based network simulation	16
4. EVALUATION OF NETWORK SCHEMES	18
4.1. General Network Model	18
4.2. Network Schemes Provided in D2.4	21
4.3. Comparison of 4G networks to 5G	21
5. DESCRIPTION AND PLANS FOR SIMULATION TOOLS	24
5.1. Description of the EMI simulation tool CaESAR	24
5.1.1. Development plan for extension of CaESAR	27
5.2. Description of RM3 tool CISIApro	28
6. SUMMARY	30
6.1. Next steps	30

List of figures:

Figure 1: RESISTO logical architecture (see deliverable D2.6 for more information).....	10
Figure 2: General idea for package based networking	17
Figure 3: The general nodes of a fixed-line network. [1].....	18
Figure 4: A telecommunication network general model. [2]	19
Figure 5: Older telecommunication network designs ranging from 2G to 4G. [23]	22
Figure 6: An example model of a 5G network. [23].....	23
Figure 7: The dependency radius describes the probability of connection in-between CIs	24
Figure 8: 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.	25
Figure 9: The different states of a system influencing the resilience.....	26
Figure 10: Screen of CaESAR computation results: mitigation strategies	27

ABBREVIATIONS

2G, 3G, 4G	Second, third and fourth generation of mobile phone systems
CI	Critical Infrastructure
EU	European Union
GUI	Graphical User Interface
LTE	Long Term Evolution (= 4G)
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 and the long term control loop. A scheme of the architecture of the integration platform is shown in Figure 1.

Aim of WP3 is the definition of the long term control loop of the RESISTO platform, which mainly features the risk and resilience analysis and management process.

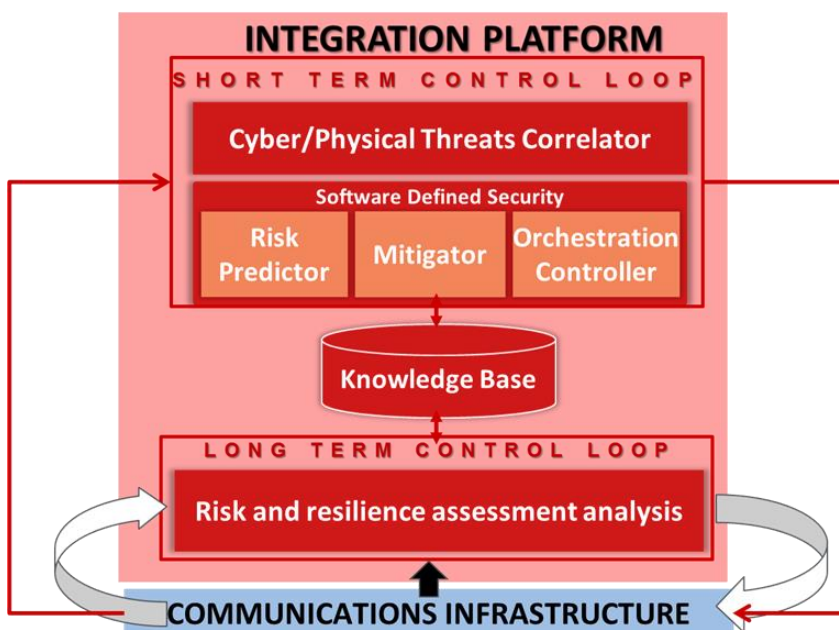


Figure 1: RESISTO logical architecture (see deliverable D2.6 for more information)

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 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 report D3.3 “Methods for cyber-physical security management for telecom CI” of T3.2. The risk and resilience assessment process and the mapping of all tools to this process will be described in D3.1 of T3.1.

This report is structured as follows:

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.

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.

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.

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

Finally, a summary of the report is presented in Section 6. This section also includes an outlook to the next steps to be followed in T3.3.

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 deliverable D2.2 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. At this point three partners returned the templates, making a thorough evaluation difficult. The remaining templates are expected to be returned within the next couple of weeks. Therefore, the complete evaluation will follow once all templates are provided. 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.
3. Most threats affect more than one system component and system function and a significant number of threats affects 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.

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. This needs to be checked once all templates are returned.
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 needs to be investigated.

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 might 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 will make the evaluation harder but still allow 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.

3. REVIEW OF AVAILABLE SIMULATION TOOLS

3.1. Network simulation tools

Table 1 lists several network simulation tools that are out there in the market. The table categorises 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	Open Source	Wired / wireless /

			OS X		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 [1, 2] 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 [3].

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 [4]. It has support for sensor network, wireless ad-hoc network, internet protocols, performance modelling and is highly customisable. 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 [5, 6]. NS3 is supported over Linux, Mac OS and can be used on a windows machine using virtualization [7]. 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 [8]. 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 [9].

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 [10]. 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 [11]. 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 [12]. 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 [13]. 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 [14].

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 [15]. A scheme for the packet-based networking is shown in Figure 2.

Simulators described in the previous section 3.1 rely on packet based networking. This simulation approach is can be computationally expensive [16]. Therefore, a tool operating at flow level is efficient to save computation time [17, 18].

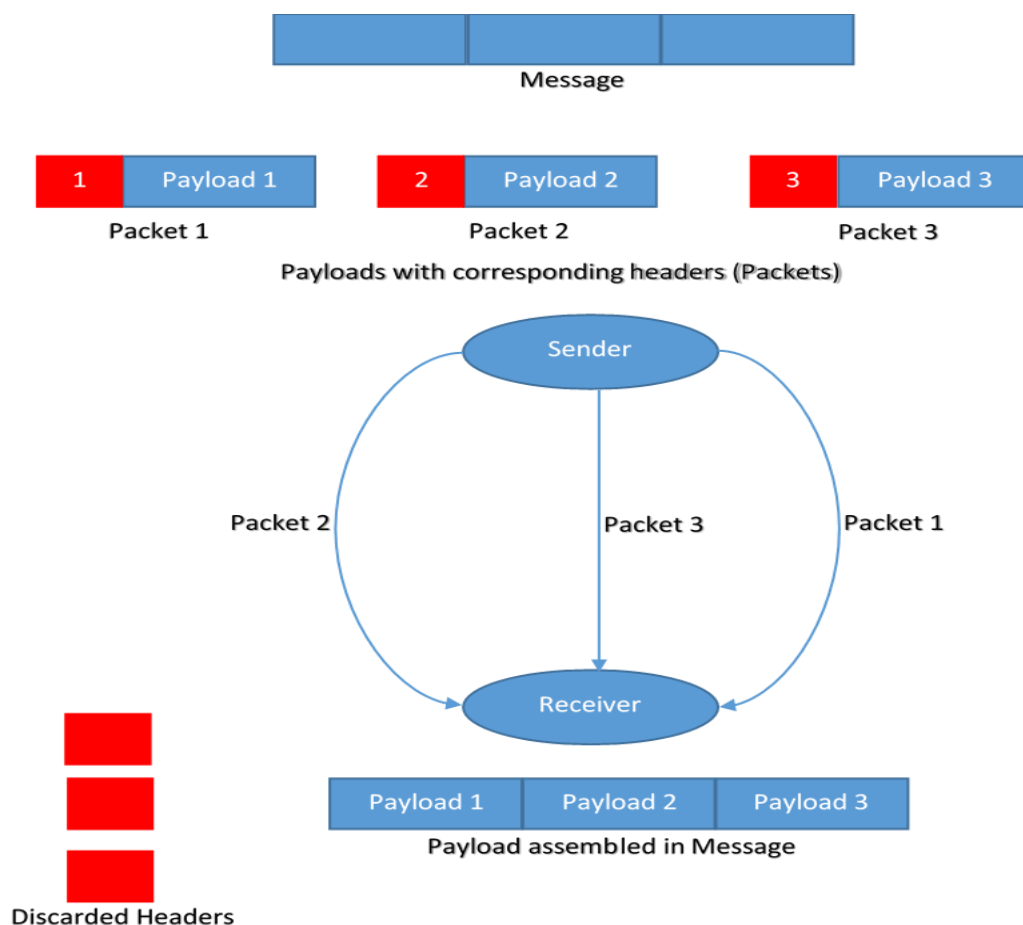


Figure 2: General idea for package based networking

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 [19], 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 3, adapted from [19].

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 be technology that are wired like computers 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 [19].

While each node of the aggregation section will cover a specific region, the core section, or backbone, will be nationwide [19]. 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.

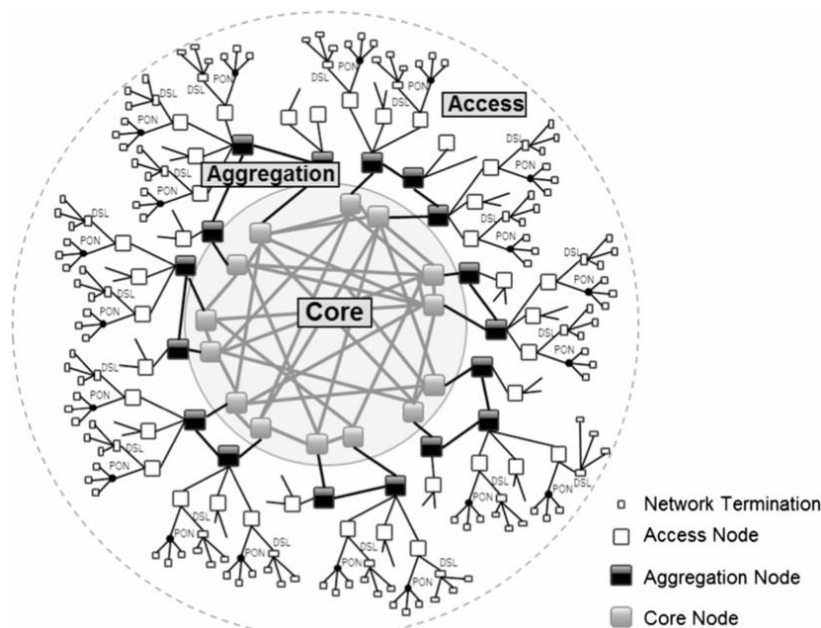


Figure 3: The general nodes of a fixed-line network. [1]

A structure, from [20], that includes both fixed and mobile services can be seen in Figure 4 with the equipment abbreviations expanded in Table 2.

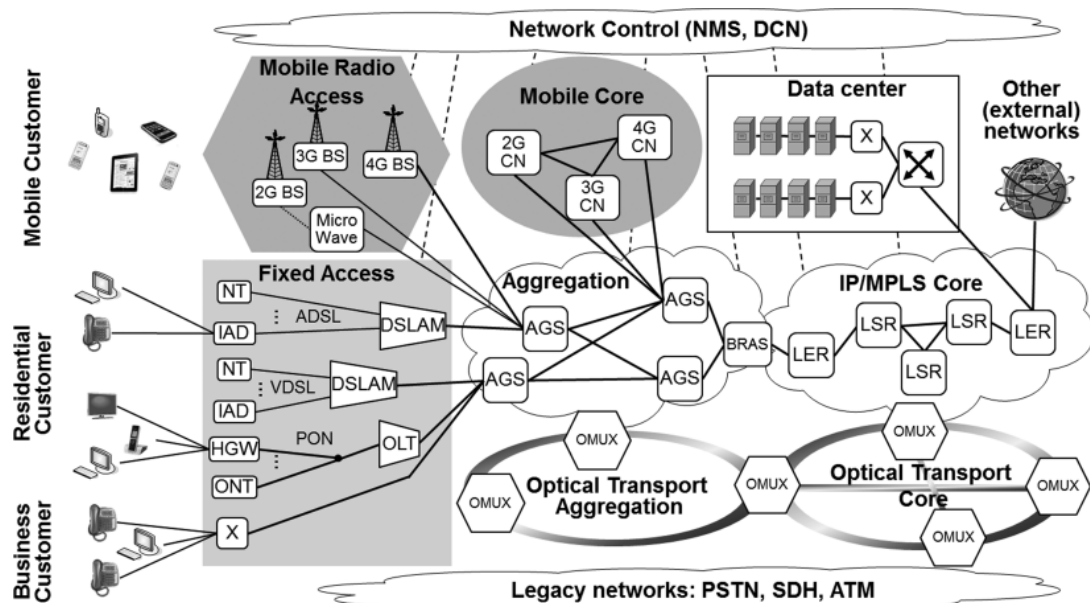


Figure 4: A telecommunication network general model. [2]

The structure is an expanded and more specific version of Figure 3, 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 2 explained. [2]

Both the fixed network and the mobile network follow the same general hierarchy as mentioned in Figure 3, 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 [20].

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 [20]. BRASs are used to connect the aggregation section to the IP/MPLS core section. BRASs help with the authentication, authorization and accounting [20].

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 centres. These data centres are used to support the actual operation of the network like billing as well as service provision for the customers like storage options [20].

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 fibre 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 fibre 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 2 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 2, 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 fibre 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 [21]. 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 [22]. According to [21], 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 [22].

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 [21], [23]. 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 [23]. The IoT means that more nodes will be required in the 5G network when compared to the 4G [21].

Historical networks (2G – 4G) general models can be seen in Figure 5 [23]. 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 [23]. 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 [23]. These include network splitting or slicing, introducing cloud capabilities, and spectrum sharing [23], [24]. The transition towards 5G will require, according to [21], shifting to mm-Wave spectrum, offloading and densification, and a spectral efficiency increase. This will involve an increase in nodes, base stations and antennas [21]. One example of a potential model for a 5G network is seen in Figure 6 [23]. This model has an emphasis on the virtualization and softwarization of the 5G network as methods to meet the 5G network requirements [23]. The differences between the historical networks and the potential 5G network include a bigger use of cloud technology.

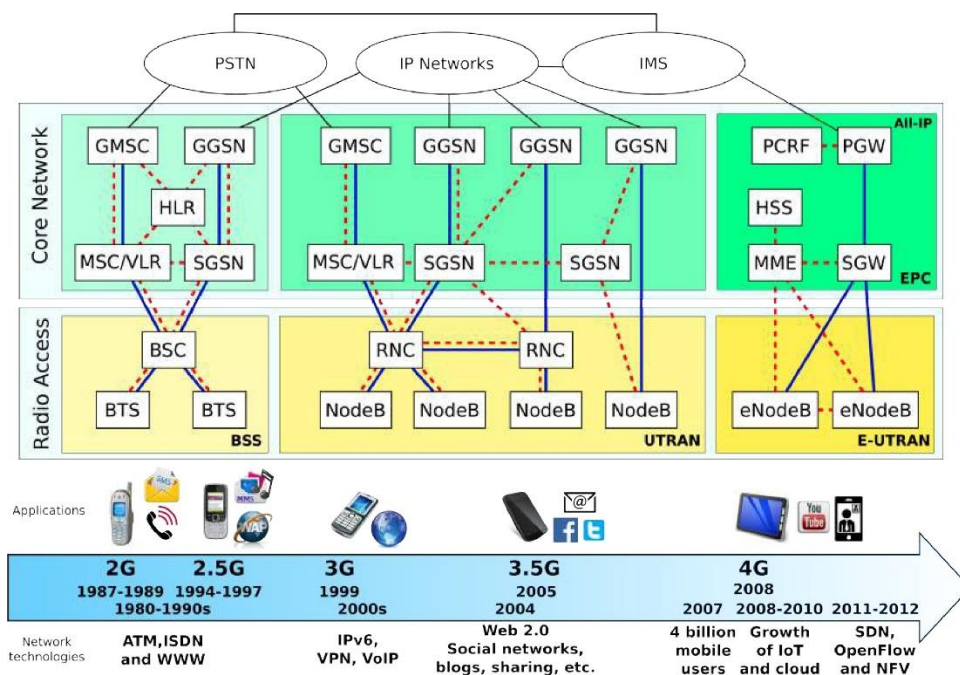


Figure 5: Older telecommunication network designs ranging from 2G to 4G. [23]

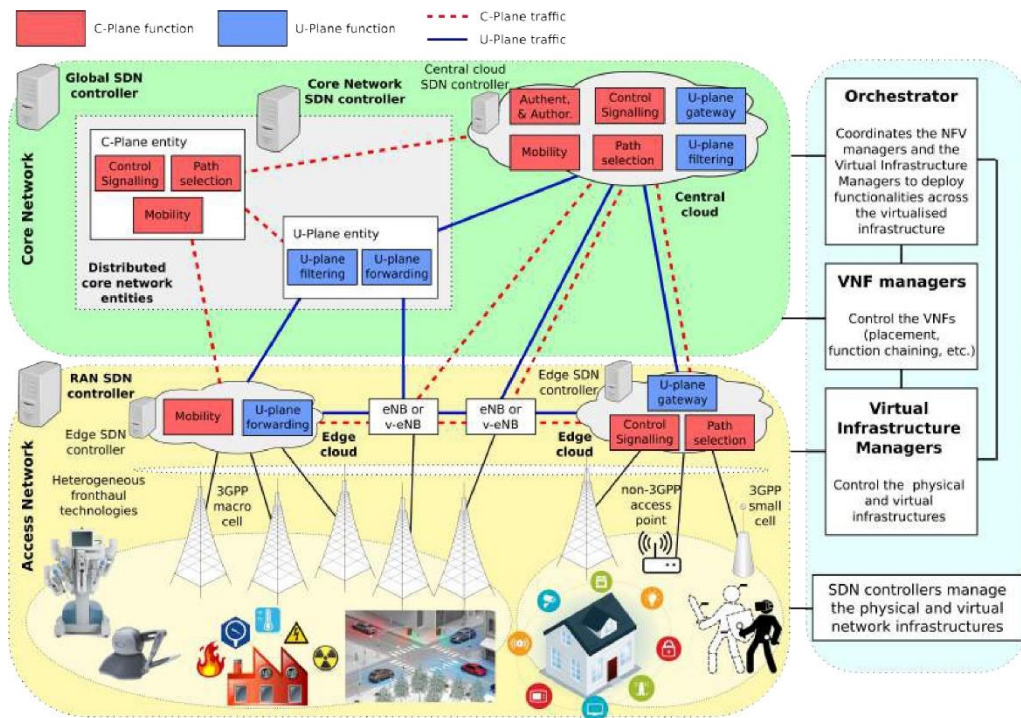


Figure 6: An example model of a 5G network. [23]

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 [25]. 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 7. 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 7, the connections between different CIs are set n times. For each connection configuration, the inner simulation loop shown in Figure 8 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.

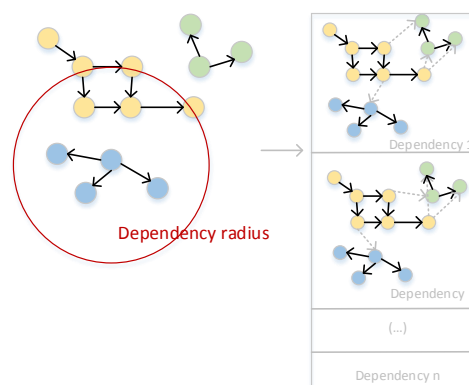


Figure 7: The dependency radius describes the probability of connection in-between CIs

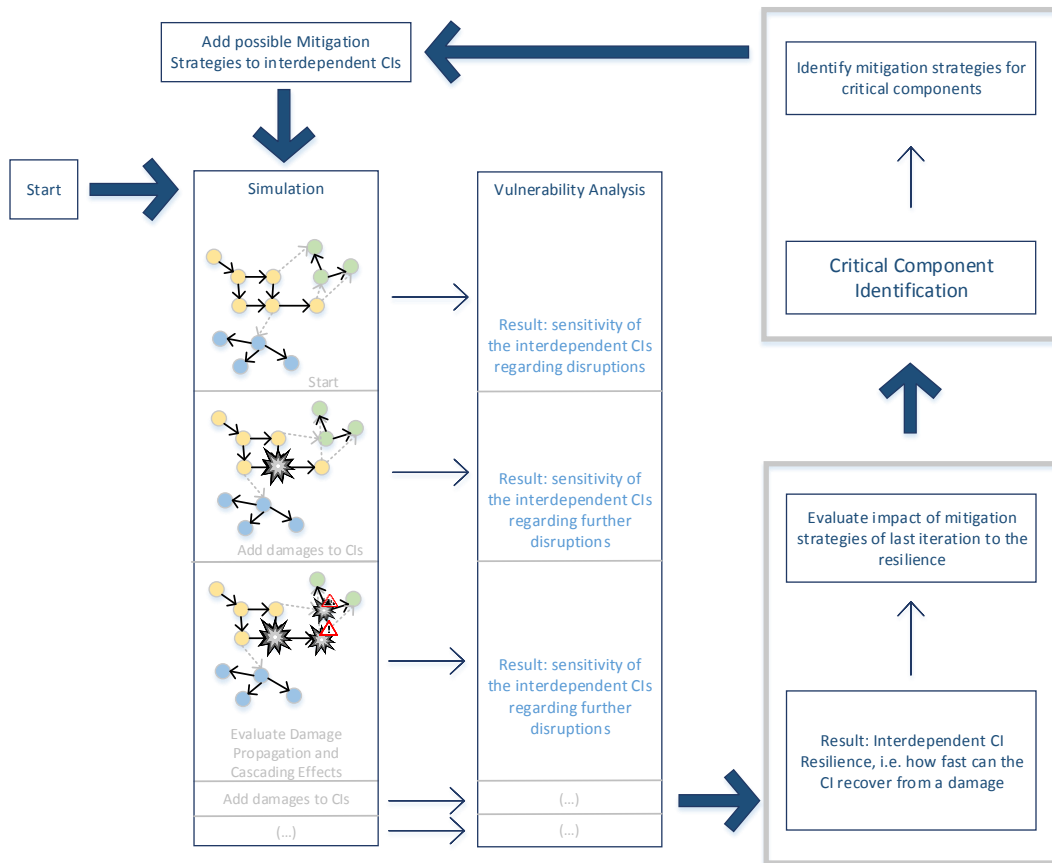


Figure 8: Overview over CaESAR simulation tool. Damages are introduced, the vulnerability of the interdependent infrastructure is analysed, critical components are identified and mitigations are applied to them.

Figure 8 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 9. 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.

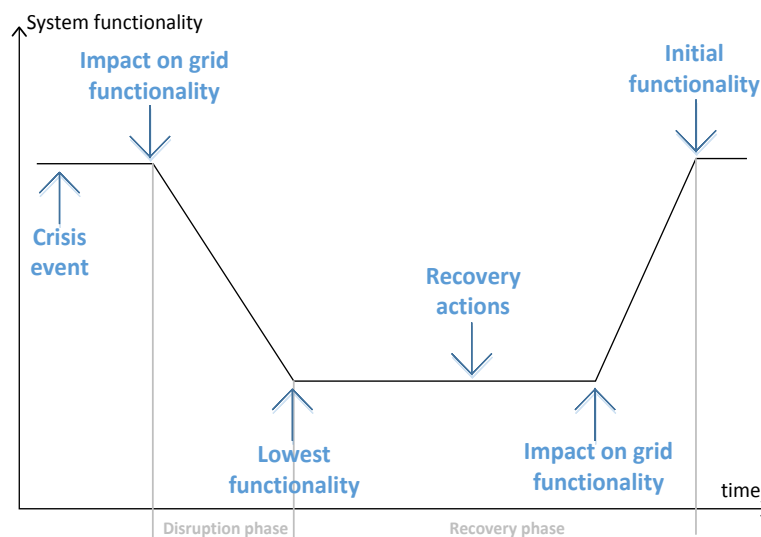


Figure 9: The different states of a system influencing the resilience.

In Figure 10 shows an example for results of CaESAR. The result shows weakest points, where mitigation strategies have the best effect and suggests a good strategy.

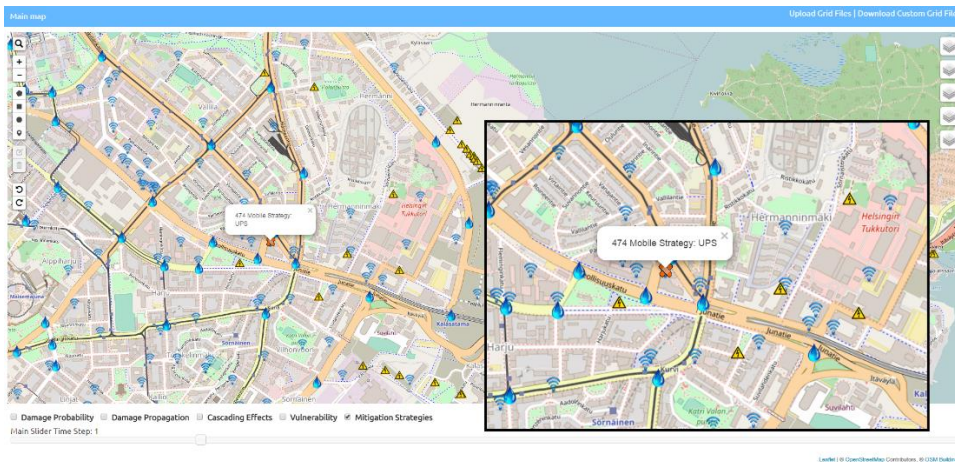


Figure 10: Screen of CaESAR computation results: mitigation strategies

5.1.1. Development plan for extension of CaESAR

The CaESAR tool was developed to simulate interconnected power, water and communication grids. For each of the three grids, 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 collected in Chapters 2, 3 and 4 in this report.

In addition to the network adaption, the following points need to be addressed:

- There are issues with the current implementation of the GUI for presenting the simulation results, as shown in Figure 10. A new implementation of the GUI or desktop application is considered as best solution.
- 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 is investigated.
- 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.

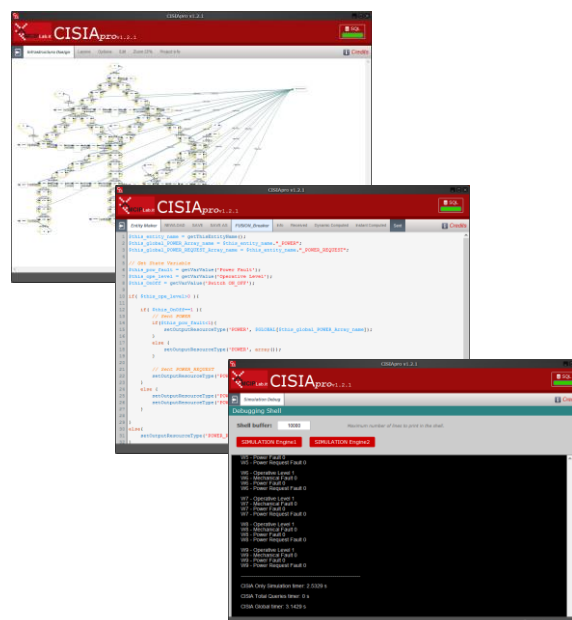
5.2. Description of RM3 tool CISIApro



CISIApro simulator (Critical Infrastructure Simulation by Interdependent Agents) has mainly been designed for analysing 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 analyse possible threats and identify vulnerabilities in the preliminary risk analysis and implementation. For this reason, CISIApro is a useful tool also for analysing 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 behaviour 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 behaviours 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 summarised 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 kind of inputs and, similarly, three kind of outputs:

1. **Induced/propagated faults:** faults propagated to the considered agent from its neighbourhoods and from the considered agent to its neighbourhood.
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 neighbourhoods and from the considered agent to its neighbourhood.

In CISIApro, the agent dynamic is described as an input/output model among the previously listed quantities. This description of agent's behaviour 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.

6. SUMMARY

This deliverable reports the status of T3.3 of WP3. The task is ongoing and an outlook to the next steps is given in the following subsection.

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 chapter 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 (chapter 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 analysed.
- Network simulators (chapter 3): provide information how the telecommunication network is simulated by other companies and institutions.
- Network schemes (chapter 4): provide direct input on the network implementation, i.e. nodes and links of all relevant sub-networks.

At this point all three sources are still evaluated. The drawing of conclusions from these inputs will be addressed in the ongoing work of T3.3 (see below).

6.1. Next steps

The input sources will be further evaluated:

- Excel templates (chapter 2): thorough analysis of all templates once all are returned by the partners. From this general features example setups for the event simulation will be derived.
- Network simulators (chapter 3): all available simulators are implemented packet-based. The (dis-)advantaged of using agent-based and flow-based approaches needs to be investigated.
- Network schemes (chapter 4): realistic schemes for the testbed environments for the use case scenarios should be provided by the operators. The usability of these needs to be checked.

Two more input sources for collecting information for the specifications of the network tools will be evaluated:

- In the scope of T2.1 of WP2 interviews with telecommunication operator experts were conducted. These included general questions about the modelling and simulation of their infrastructures. The interviews are currently collected and will be jointly evaluated. The results are considered another possible input for the simulation tools that will be investigated.
- A particular issue is to retrieve realistic network schemes including localization information, i.e. how the nodes are distributed. This could be addressed by evaluating databases provided by open-source projects collecting the positions of mobile phone cell towers.¹

For CaESAR a few features need to be implemented, as described in section 5.1.1.

¹ OpenCellid: <https://opencellid.org>

References

- [1] NS2 <https://www.isi.edu/nsnam/ns/index.html>
- [2] 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
- [3] Ibrahim F H *Network Simulator 2: a Simulation Tool for Linux*
<https://www.linuxjournal.com/article/5929>
- [4] OMNET++ <https://omnetpp.org/intro/>
- [5] 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
- [6] 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>
- [7] NS3 Development team *NS3 Wiki* https://www.nsnam.org/wiki/Main_Page
- [8] *Riverbed Modeler* (Riverbed (OPNET))
- [9] K'oksal M M *A Survey of Network Simulators Supporting Wireless Networks*
- [10] *QualNet Network Simulator Software* (scalable-network technologies)
- [11] *Network Simulation Tools: OneSim Simulator* <http://networksimulationtools.com/onesim-simulator/>
- [12] *The ONE: The Opportunistic Network Environment simulator* <https://akeranen.github.io/the-one/>
- [13] *PEERSIM* <http://networksimulationtools.com/peersim/>
- [14] Jelasity M, Montresor A, Paolo Jesi G and Voulgaris S *PeerSim: A Peer-to-Peer Simulator*
- [15] Baran P 1964 *On Distributed Communications: I. Introduction to Distributed Communications Networks* (RAND Corporation)
- [16] 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
- [17] 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
- [18] Anggono G and Moors T 2017 *IEEE Commun. Lett.* **21** 496–9
- [19] 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
- [20] 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
- [21] 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
- [22] 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
- [23] Condoluci M and Mahmoodi T 2018 *Computer Networks* **146** 65–84
- [24] Morgado A, Huq K M S, Mumtaz S and Rodriguez J 2018 *Digital Communications and Networks* **4** 87–97
- [25] Hiermaier, Stefan, Sandra Hasenstein, and Katja Faist *7th REA Symposium 2017 University of Liège, Belgium*