ANASTACIA

Advanced Networked Agents for Security and Trust Assessment in CPS/IoT Architectures

D6.6

Final End-user Validation and Evaluation Report

This deliverable reports the validation and evaluation of the ANASTACIA integrated framework within the use cases implemented. After the validation of the use cases, the end users will evaluate the results and provide their feedback regarding optimizations and improvements.

Distribution level	PU
Contractual date	31.12.2019 [M36]
Delivery date	27.12.2019 [M36]
WP / Task	WP6
WP Leader	OdinS
Authors	Rafael (OdinS), Dan (OdinS), Stefano (SOFT), Diego (MONT), Ivan Vaccari, Enrico Cambiaso, Silvia Scaglione, Elisabetta Punta (CNR), Ruben, Jesus (ATOS), Giannis (UBITECH), Alejandro, Jordi (UMU), Matteo (DG), Federico (MI) Piotr, Satya (UTRC), Miloud, Mohammed and Tarik (AALTO)
EC Project Officer	Carmen Ifrim carmen.ifrim@ec.europa.eu
Project Coordinator	Softeco Sismat SpA Stefano Bianchi Via De Marini 1, 16149 Genova – Italy +39 0106026368 stefano.bianchi@softeco.it
Project website	www.anastacia-h2020.eu

ANASTACIA has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement N° 731558 and from the Swiss State Secretariat for Education, Research and Innovation

Table of contents

PUBLIC SUMMARY 4
1 Introduction 5
1.1 Aims of the document 5
1.2 Applicable and reference documents 5
1.3 Revision History 5
1.4 Acronyms and Definitions 6
2 Contextualization 8
2.1 Scope 8
2.2 End-Users 9
2.2.1 Interviews 10
3 End-User Evaluation of ANASTACIA Framework Validated with Use Cases of D6.5 12
3.1 Use-Case: Smart Building 12
3.1.1 Description of Use Case and Benefits from ANASTACIA Framework 12
3.1.2 Validation based on Test-Cases and Measured KPIs 14
3.2 Use-Case: Mobile Edge Computing in 5G Network 56
3.2.1 Description of Use Case and Benefits from ANASTACIA Framework 56
3.2.2 Validation based on Test-Cases and Measured KPIs 57
3.3 Results of End-User Evaluation 62
3.3.1 Final Evaluation Questionnaire 62
3.3.2 Results of End-Users Questionnaire 63
4 Conclusions 68
5 Annex 1 – Captured Images during Smart Building Demo 69
6 Annex 2 – Captured Images during 5G MEC Demo 88
7 Annex 3 – Interview Questionnaires 93

List of figures

Figure 1: High level ANASTACIA architecture (Detailed in D1.5) 8
Figure 2: ANASTACIA final architecture integrated. (Detailed in D1.5) 9
Figure 3: Targeted end user categories - simplified groups. 10
Figure 4: Use case deployment for Smart Building 12
Figure 5: Use case implementation of Smart Building 13
Figure 6: Diagram of Step 1 of Phase R1 Attack in Smart Building 14
Figure 7: Diagram of Step 2 of Phase R1 Attack in Smart Building 20
Figure 8: Diagram of Step 3 of Phase R1 Attack in Smart Building 26
Figure 9: Diagram of Step 4 of Phase R2 Attack in Smart Building 30
Figure 10: Diagram of Step 5 for the Phase R2 Attack in Smart Building 35
Figure 11: Diagram of Step 6 of Phase R3 Attack in Smart Building 39
Figure 12: Diagram of the Step 7 for the Phase R3 Attack in Smart Building 44
Figure 13: Diagram of the Phase R4 Attack in Smart Building 47
Figure 14: Use case implementation of MEC in 5G scenario 56
Figure 15: Start of the workshop, explaining the demo process 69
Figure 16: Showcasing the different users interfaces of the ANASTACIA framework 69
Figure 17: Audience attending to the ANASTACIA workshop 70
Figure 18: DSPS interface, showing a normalize state – GREEN. 70
Figure 19: Turning IoT devices ON 71
Figure 20: Devices show in the system, but not authenticated and need permission to bootstrapping to access the network 71
Figure 21: Several step process to set up the security policies for the Bootstrapping 72
Figure 22: Definition and adding dependencies in the security policy editor 72
Figure 23: Refining the policies to make them actionable (from HSPL to MSPL) 73
Figure 24: Watching the state of the policies execution. 73
Figure 25: Different policies are enabled successfully 74
Figure 26: Watching the ONOS interface, waiting for IoT devices to show as they bootstrap 74
Figure 27: Devices start showing in ONOS as they are recognized 75
Figure 28: Different intents show in ONOS as the IoT devices perform the bootstrapping 75
Figure 29: A roge IoT device is turned on to emulate an attack sending data to an external endpoint 76
Figure 30: The new device is integrated in the system. 77
Figure 31: The untrusted endpoint is detected by MMT probe and notified to Incident Detector 78
Figure 32: Incident Detector processes the event and generates a “Potential data leakage detected” alert and mitigation strategies such as filtering and forwarding. 78

Figure 33: The ONOS SDN controller received the order from the Security-Orchestrator to create the rule for forwarding the traffic generated by the malicious device. 79
Figure 34: DSPS shows the seal status after the detection and mitigation of the potential data leakage 79
Figure 35: A siren placed in the the premises is prepared to be turned on if any of the temperature readings from the BMS is unusually high 80
Figure 36: After an attacked IoT device publishesh a higher than normal temperature reading, the siren is turned on. 80
Figure 37: The VDSS shows an unusual temperature reading 81
Figure 38: The DSPS shows the alert associated with a MiM attack 81
Figure 39: The system reports the MiM has been mitigated 82
Figure 40: An attack (Slow DOS) is performed by a CNR device in the UMU premises 82
Figure 41: This attack is also detected by the ANASTACIA framework 83
Figure 42: The DSPS shows the Slow DOS activity. 84
Figure 43: The VDSS shows the different alarms triggerd by ANASTACIA 84
Figure 44: DSPS showing GREEN status after the SlowDoS attack is mitigated by ANASTACIA framework. .. 85
Figure 45: The external attacker is shown in the image 86
Figure 46: The potential data leakage is detected 86
Figure 47: Intents are created to address the issue 87
Figure 48: Final view of the ONOS topology after the Smart Building demo 87
Figure 49: Servers running the 5G network core 88
Figure 50: Slicer prototype portal. 88
Figure 51: legitimate IoT camera device with a session attached 89
Figure 52: NUC running the 5G base station 90
Figure 53: Raspberry, runing the legitimata and illegitimate devices. 90
Figure 54: Servers 1-4 running the 5G core 91
Figure 55: View of the ANASTACIA framework showing the 5G monitor activity 91
Figure 56: VDSS showing the anomal activity of the camera 92
Figure 57: DSPS showing the 5G anomaly is mitigated 92

PUBLIC SUMMARY

Regarding the DoA, the aim of D6.6 is to report the validation and end-users evaluation of the ANASTACIA framework final release, following the methodology of use-cases and test-cases designed in D6.5.

The execution of test-cases must be monitored to evaluate the performance and the reliability of the ANASTACIA framework.

After the assessment of the test-cases, the end users will evaluate the ANASTACIA demonstrations in the use cases to provide their feedback regarding any optimizations and further improvements.

In particular, D6.6 provides the ANASTACIA evaluation regarding two ambitious use cases:

Building Management System (BMS) and Mobile Edge Computing (MEC) in 5G Networks.

For each use case, its implementation description and the ANASTACIA benefits are reported. This description provides visual data (i.e. web interfaces and log consoles) recorded in the use case execution with the ANASTACIA framework.

Furthermore, the measurements of Key Performance Indicators (KPI) related to the validated test-cases are reported.

For the end-users evaluation, the document shows the results of the feedback and questionnaires answered by 59 interviewers regarding the demos performed.

To enable the evaluation process, the demonstration of the use cases was recorded as videos. The videos demonstrate the final release of ANASTACIA framework in two mentioned use cases: BMS and 5G MEC.

1 INTRODUCTION

1.1 AIMS OF THE DOCUMENT

This deliverable reports the outcomes of use cases validation and end-users evaluation of the final release of the integrated ANASTACIA framework. The main objectives are:

- to report the validation of the ANASTACIA framework regarding the test cases designed in D6.5;
- to explain the benefits offered by the ANASTACIA framework in the use cases developed;
- to show the performance and usability of the ANASTACIA framework according the KPIs measured;
- to present the results analysis regarding the gathered end-users questionnaires in face-to-face workshops.

1.2 APPLICABLE AND REFERENCE DOCUMENTS

This deliverable refers the following documents:

- Grant Agreement N°731558 and annexes ("Description of Action").
- D1.4 User-centred Requirement Final Analysis.
- D1.5 Final Architecture Design.
- D6.4 Final Technical integration and validation Report.
- D6.5 Final Use cases implementation and tests Report.

1.3 REVISION HISTORY

Version	Date	Author	Description
V0.1	18/11/2019	Rafael Marin (OdinS)	ToC
V0.2	25/11/2019	Rafael Marin (OdinS)	Introduction
V0.3	03/12/2019	Rafael Marin (OdinS)	Contextualization
V0.4	06/12/2019	Rafael Marin (OdinS)	Initial contents of Use Case: Smart Building
V0.5	12/12/2019	Rafael Marin (OdinS)	Initial contents of Use Case: MEC in 5G Network
V0.6	17/12/2019	Rafael Marin (OdinS)	Adding Images from Smart Building Demo
V0.7	18/12/2019	Rafael Marin (OdinS)	Adding Images from MEC Demo in 5G Network
V0.8	19/12/2019	Dan Garcia (OdinS)	Including End-Users feedback
V0.9	20/12/2019	Dan Garcia (OdinS)	Including KPIs measured and descriptions
V0.10	21/12/2019	Dan Garcia (OdinS)	Describing Use Case: Smart Building
V0.11	22/12/2019	Rafael Marin (OdinS)	Describing Use Case: MEC in 5G Network
V0.12	23/12/2019	Rafael Marin (OdinS)	Describing Conclusions and End-Users feedback
V0.13	24/12/2019	Bruno Vidalenc (Thales)	Revision of the whole document
V0.14	27/12/2019	Rafael Marin (OdinS)	Final version

1.4 ACRONYMS AND DEFINITIONS

Acronym	Meaning
API	Application Programming Interface
BMS	Building Management Systems
CRUD	Create/Retrieve/Update/Delete
DDoS	Distributed Denial of Service attack
DoA	Description of Action
DPO	Data Protection Officer
DSPS	Dynamic Security and Privacy Seal
FR	Functional Requirement
GDPR	General Data Protection Regulation
HSPL	High-level Security Policy Language
IoT	Internet of Things
Kafka	Message broker used in ANASTACIA framework to enable distributed communication between components
KPIs	Key Performance Indicators
MAS	Mitigation Action Service
MEC	Mobile Edge Computing / Multi-access Edge Computing
MSPL	Medium-level Security Policy Language
MVP	Minimum Viable Product
NFR	Non Functional Requirement
NFV	Network Function Virtualization
OVS	Open Virtual Switch
PI	Policy Interpreter
PR	Privacy Requirement
REST	Representational State Transfer
SAS	Security Alert Service
SEP	Security Enforcement Plane
SDN	Software Defined Network
SO	Security Orchestrator
TRL	Technology Readiness Level
TC	Test Case
UC	Use Case

Acronym	Meaning
UML	Unified Modeling Language
VDSS	Verdict Decision Support System
VID	Virtualized Infrastructure Domain
VNF	Virtual Network Function
XML	eXtensible Markup Language

2 CONTEXTUALIZATION

2.1 SCOPE

The D6.6 reports the use cases validation and end-users evaluation the ANASTACIA framework final release to enable autonomous decisions using networking technologies (such as Software Defined Networking and Network Function Virtualisation) and monitoring/reaction methodologies/tools as well as intelligent orchestration and dynamic security enforcement. Figure 1 shows the high-level view of the ANASTACIA architecture, which includes the latest changes with respect to the version delivered in D1.5.

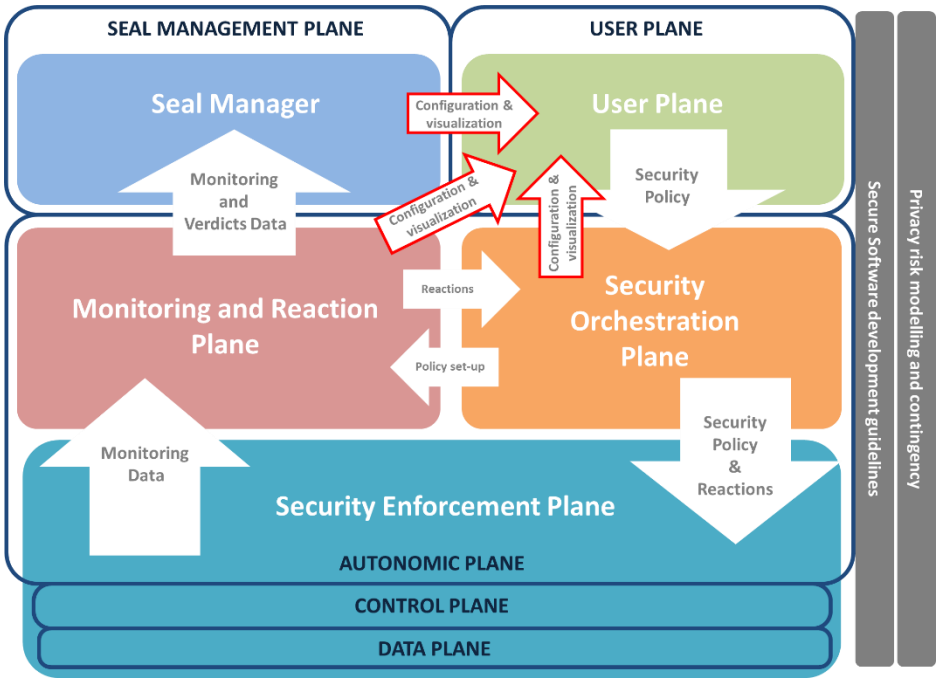


Figure 1: High level ANASTACIA architecture (Detailed in D1.5)

The ANASTACIA architecture integrates the Data Plane with IoT infrastructure where IoT devices, physical and virtual network elements interact. On top of that, the Control Plane manages the computing, storage, and networking resources in the Data Plane by leveraging SDN controllers, NFV orchestration platforms, and IoT controllers. The Autonomic Plane includes the components that provide the ANASTACIA framework with its intelligence and dynamic behaviour.

During the second cycle of the project, the framework of the ANASTACIA components has been improved with respect to the first release evaluated in D6.3. Figure 2 presents the final version of the ANASTACIA framework developed and integrated. The most important aspects that have evolved since then are summarized as follows:

- Monitoring Plane includes new components (**Incident Detector, Resource and QoS monitoring**)
- Reaction plane, where the complete VDSS is defined with more details, including components required by the VDSS to fetch input from different parts of the ANASTACIA framework (**Assets Model, Mitigations Repository**).
- Security Orchestrator, has involved and is now detailed with the components **System Model Service,** the **Security Orchestrator Optimizer,** the **Performance Data Analytics**, and the additional components/**drivers** needed to interact with NFV, SDN and IoT functions.
- Seal Manager, where the components included have been **refined** (DSPS Agent, Seal Manager and DSPS Repository now available).

- User plane extended with more User Interfaces to facilitate the configuration and visualization of the operations of the ANASTACIA framework (**DSPS UI, Alerting and reaction dashboard, Policy Editor UI and Logging Service**).

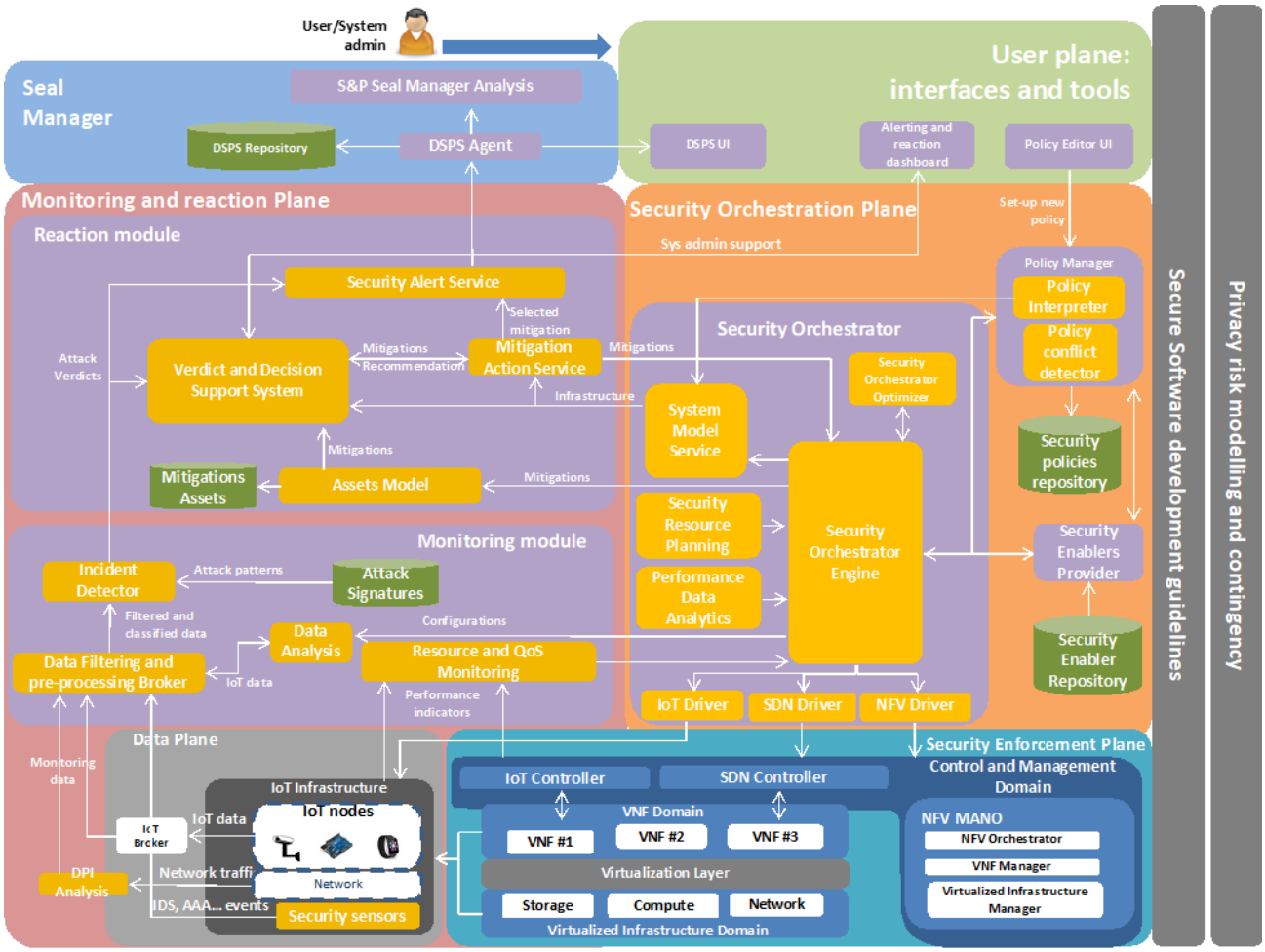


Figure 2: ANASTACIA final architecture integrated. (Detailed in D1.5)

The next subsection shows the categories of end-users and stakeholders considered for performing the ANASTACIA evaluation according to the use cases demonstrated: Building Management System (BMS) and Mobile Edge Computing in 5G Networks (MEC).

2.2 END-USERS

Regarding the DoA, part of the activities of task T6.3 included the evaluation of ANASTACIA framework with the participation of stakeholders and potential end-users. The application domain of the ANASTACIA framework allows potentially multiple user categories:

- Managers of Building Management System (BMS)
- Managers of Mobile Edge Computing (MEC) and 5G Networks
- Privacy and Security Researchers
- Security Managers/Consultants
- Privacy Managers/Consultants
- System / Network Administrators
- CPS/IoT Managers/Researchers
- NFV Developers/Researchers
- SDN Developers/Researchers

- SW Developers/Researchers
- Edge/Cloud Developer/Researchers
- Research Team Leaders
- Solution Integrators
- Lawyers
- Chief Security Officer (CSO)
- Chief Technology Officer (CTO)
- Chief Information Officer (CIO)
- Chief Information and Security Officer (CISO)
- GDPR-associated actors (e.g. Data Protection Officer (DPO))

Following the approach of the D1.4 "Final User Requirements Analysis", this wide range of user categories can be grouped as shown in Figure 3 considering the principal management roles of the use case domains employed for demonstration purposes (i.e. BMS and MEC) the CPS/IoT Managers group includes, for example, Managers of BMS - in charge of overseeing the operational continuity of the building, considered as a "system of systems" that includes ICT (e.g. network components) and IoT (e.g. sensors, gateways etc.) architectures, supported on one side by Security Managers (e.g. CISOs), in charge of the business continuity of large ICT infrastructures, and Privacy Managers (e.g. DPOs) , in charge of compliance with e.g. GDPR.

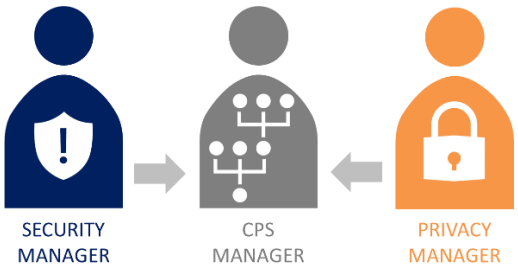


Figure 3: Targeted end user categories - simplified groups.

2.2.1 Interviews

In particular, the evaluation was performed by 59 different users in order to gather meaningful feedback as shown in the following list.

1. Jesús Luna García – Cloud Researcher - Bosch Gmbh
2. Diego López de Ipiña - Research Team Leaders - University of Deusto
3. Stefano Secci – NFV researcher - CNAM
4. Alba Hita Catalá – SDN Developer - University of Murcia
5. Stefano Bianconi – NFV Researcher - Unicam
6. Norberto García Marín – Solution Integrator - University of Murcia
7. Rubén Rodríguez Rey – Network Administrator - University of Murcia
8. Alejandro Javier González – IoT Manager - University of Murcia
9. Georgio Balconi – Edge Developer - University of Murcia
10. Francisco Fernánzez - Chief Security Officer -
11. José Antonio García-Calvo Fernández - Research Team Leaders - College of Computer Engineers / Inforges Seider G
12. Guillermo Fernández Ruiz - SW Developers - University of Murcia
13. Jesús Sánchez Gómez - Privacy and Security Researchers - University of Murcia
14. Rafael Torres Moreno – Cloud Researcher - University of Murcia
15. Ana Hermosilla García – SDN Researcher - University of Murcia
16. Luis Bernal Escobedo - Managers of Mobile Edge Computing (MEC) and 5G Networks - University of Murcia
17. Jorge Gallego Madrid – Edge Researcher - University of Murcia

18. Pedro Javier Fernández Ruiz - Privacy and Security Researchers - University of Murcia
19. Eduardo Fernández Campuzano - SW Developers - Ticarum
20. Alberto Quijada Guillamón – SW Developer - Ticarum
21. Daniel Luján López - Chief Information and Security Officer - Colegio Profesional Ing. Informática
22. Ehsan EBRAHIMI KHALEGHI - Chief Security Officer - THALES
23. Agathe Blaise - Research Team Leaders - THALES
24. Geofroy Cholon - Security Managers - THALES
25. Arianna Fedi - Solution Integrators - CNR
26. Chiara Vitale - IoT Researcher- CNR
27. Davide Chiarella - Security Manager- CNR
28. ADRIEN Wian - NFV Developers - Telecom Paristeah
29. CAVALLI Ana - Privacy Managers – Montimage
30. Vinh Hoa LA – researcher - Montimage
31. Nauman Shahid - Cloud Developer – UTRC
32. Tejasoinee Darure - Researcher- UTRC
33. Ricardo Orizio – Researcher -UTRC
34. Andrea Balogh – Researcher – UTRC
35. Thu Le Pham – Researcher - UTRC
36. Huu Nghia Nguyen - SDN Researcher - Montimage
37. Luong NGUYEN - Managers of Mobile Edge Computing (MEC) and 5G Networks - Montimage
38. Wissam Mallouli - Managers of Building Management System (BMS) - Montimage
39. Rita Pasini – Researcher - SOFTECO
40. Christian Melchiorre – Project Manager - SOFTECO
41. Mauro Vannucci – Researcher - SOFTECO
42. Gianni Viano – CTO - SOFTECO
43. Luca Naso Rappis – Researcher - SOFTECO
44. Simone Pittiglio - Researcher- SOFTECO
45. Marco Boero - Manager of Green Mobility Business Unit - SOFTECO
46. Dimitris Papamartzivanos - Cyber Security Research Associate - UBITECH
47. PANAGIOTIS PARTHENIS – Software Engineer – UBITECH
48. Sofia Anna Menesidou – R&D Engineer, security researcher UBITECH
49. BAGHEHRI SYLLA - Researcher - UDG Alliance
50. Sarah Büchner - Researcher - UDG Alliance
51. Hazan Michael - Researcher - Device Gateway SA
52. Luca Caviglione – Researcher - Institute for Applied Mathematics and Information Technology (IMATI)
53. Othmane HIRECHE, SDN Developer, Aalto University
54. Abderrahmane ABADA, NFV Developer, Aalto University
55. Chafika Benzaid, Software Developer, Aalto University
56. Jonathan Prados, Security Researcher, Aalto University
57. Walid BOUMEZER, Cloud Developer, Aalto University
58. NAIT ABBOU Aiman, Software Developer, Aalto University
59. NADIR ZINELAABIDINE, SDN Researcher, Aalto University

In the following chapters, the detailed description of the end-users evaluation of ANASTACIA framework regarding two use cases (BMS and 5G MEC) is given. Firstly, it describes the implemented use cases and the benefits provided by ANASTACIA framework. Secondly, the measured KPIs and the visual data captured during the demo sessions are presented. Thirdly, the end-users feedbacks collected by the questionnaires and face-to-face workshops are detailed in Section 3.3. Finally, Annexe 3 presents a table with all answers provided by end-users.

3 End-User Evaluation of ANASTACIA Framework Validated with Use Cases of D6.5

In this section, the end-user evaluation of ANASTACIA framework is reported regarding the two ambitious use cases developed: Smart Building and Mobile Edge Computing in 5G networks. Below we describe in details the implementation of the use cases and the benefits provided by the ANASTACIA framework. The description incorporates the information obtained for ANASTACIA components (i.e. end-user interfaces for management and operation) during the demo sessions of the use cases execution. To validate the use cases, multiple test-cases with their key performance indicators (KPIs) have been evaluated and reported according to the methodology defined in the deliverable D6.5. After the validation of use cases execution, the end-users evaluation was performed in face-to-face workshops based on demo sessions, questionnaires and recorded videos. At the end of this document, the results analysis of the end-users feedback is described.

3.1 Use-Case: Smart Building

3.1.1 Description of Use Case and Benefits from ANASTACIA Framework

To evaluate the Smart Building use case, the ANASTACIA framework is deployed and executed in the premises of UMU. In the middle of Figure 4, we can see the deployment of the ANASTACIA core platform consisted of components for monitoring, reaction, orchestration, seal management and users interfaces. In the right side, the deployment of SDN and VNF controllers for enforcing the mitigation actions are deployed and operated by the AALTO OpenStack. In the left side, the attackers and victims involved in this use case are deployed in two rooms of PEANA experimental laboratory in UMU premises. PEANA provides the SDN network infrastructure for the data plane and control plane managed by the ANASTACIA framework.

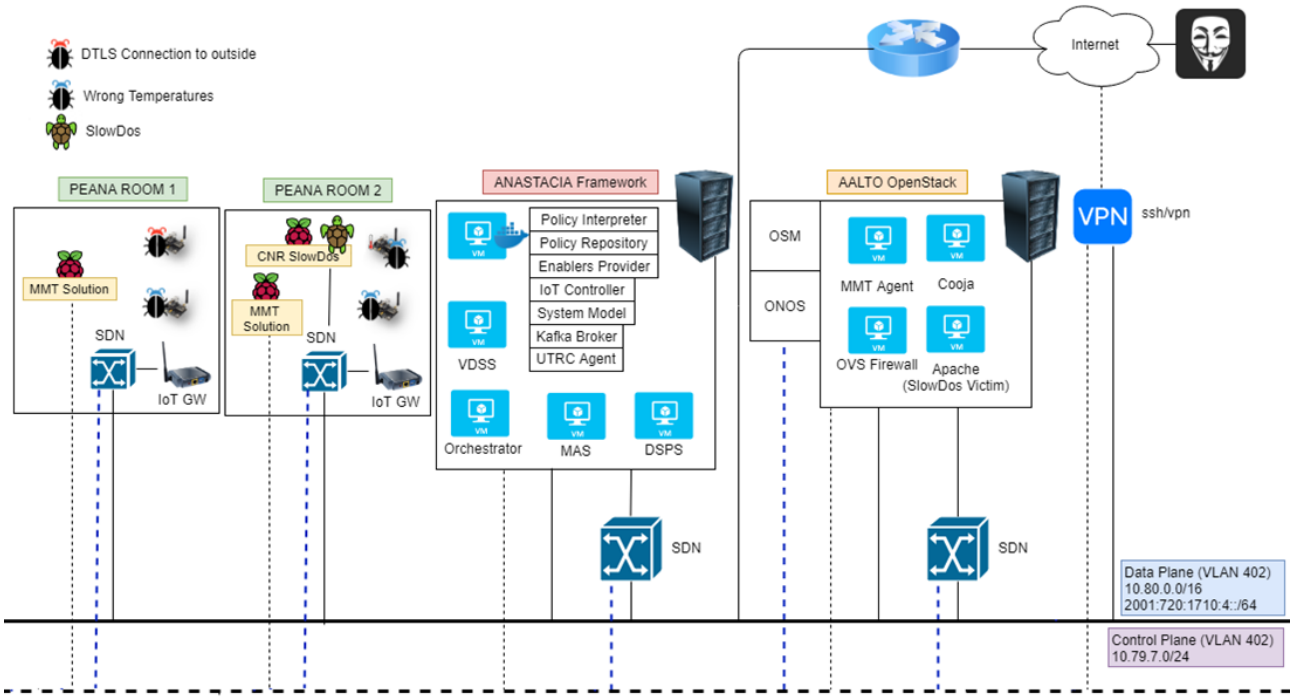


Figure 4: Use case deployment for Smart Building

The implementation of the Smart Building use case consists of 4 attacks simulated in 4 phases with 9 steps as illustrated in Figure 5. The 4 attacks are indicated below:

1. Attack with an infected IoT device sending sensitive data traffic using a DTLS connection to an outside endpoint.
2. Attack with an infected temperature sensor sending high temperature to activate fire alarm in the building.
3. Slow-DoS attack against web service of BMS server.
4. External attacker tries to compromise an IoT device through data request from an outside endpoint.

The figure shows several swimming lanes regarding the main actors involved in the use case such as adversary generating the attacks, UMU premises with the victims (i.e. BMS server, IoT/CPS devices), ANASTACIA framework and DPO/CSO control the security/privacy status via DSPS Seal interface. On top adversary actor is placed that has the role to start the attacks over the smart building premise with the four mentioned strategies in the steps indicated with red circles. In the next lane, the IoT/CPS infrastructure and BMS services of UMU premises are represented in several physical rooms. During the attacks, ANASTACIA framework performs several main actions (i.e. monitoring, detection, mitigation) shown in the diagram. Below, the status of seal is indicated in different colors to report the protection level for security/privacy in the proposed scenario that DPO/CSO might observe. Next DPO/CSO preventive actions are listed. Finally, each phase of the use case have a leader (ANASTACIA project partner) responsible for validate the related steps and test cases.

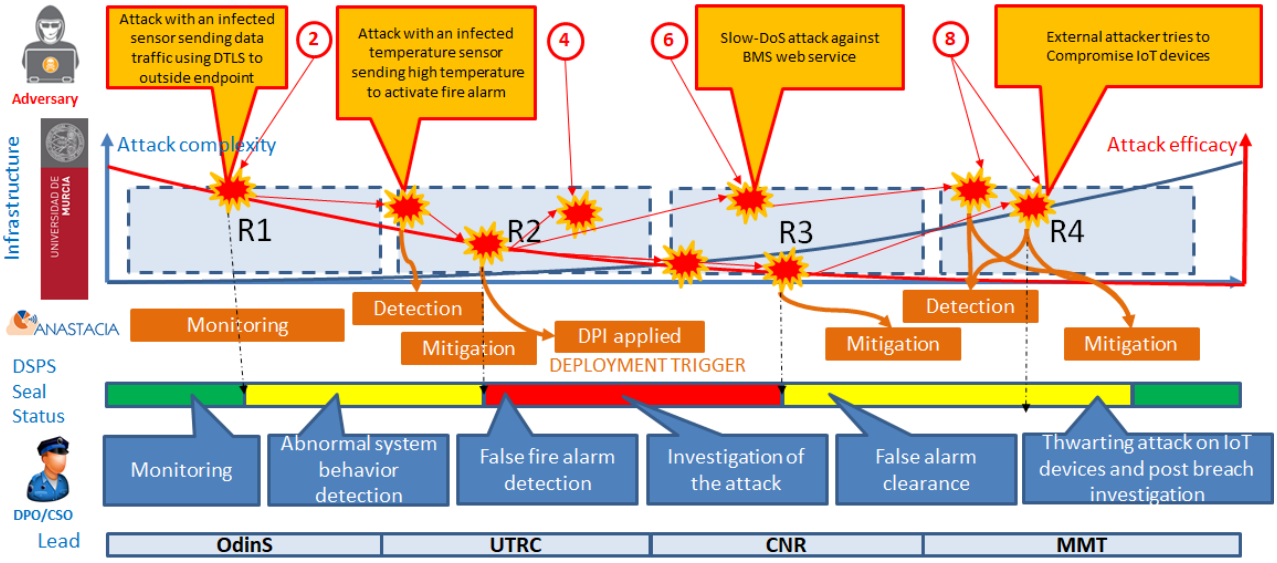


Figure 5: Use case implementation of Smart Building

The main benefit provided by ANASTACIA framework is to address the security management of distributed IoT/CPS scenarios as the use case of Smart Buildings. ANASTACIA provides innovative benefits from holistic policy-based management and efficient orchestration approach, novel monitoring and reaction tools as well as NFV/SDN-based mitigation solutions to tackle well-known cyber-attacks and anomaly threats. In this use case, the key benefits provided by ANASTACIA framework are as follows.

1. Novel security monitoring agents that can be deployed dynamically to detect well-known threats and attacks.
2. AI-based data analysis to detect anomaly behaviors and unknown threats.
3. Automated and cognitive reaction to decide the best mitigation strategies according to the attacks detected.
4. Holistic policy-based security management to establish and interpret the mitigation strategies in specific enforcement actions with conflict detection and policy dependencies.
5. Efficient security orchestration to manage the decided enforcement actions through SDN and NFV controllers.

For the use case of Smart Building, Annexe 1 presents the photos and interfaces captured during the demo sessions performed in the University of Murcia during the days 11-12 December 2019.

3.1.2 Validation based on Test-Cases and Measured KPIs

This section shows the measured KPIs of the validation executions of the use case of Smart Building regarding the test cases defined in the deliverable D6.5. Below, the tables show the results of test-cases and also the KPI measurements in seconds for 5 executions of the performance tests.

3.1.2.1 Phase R1: Attack with an infected sensor sending data traffic to an outside endpoint using DTLS

This section reports the validation of the first phase of the use case of Smart Building in the room1 of PEANA experimental laboratories where an attack is simulated with an infected sensor sending sensitive information to an outside endpoint through a DTLS communication. This phase shows the malicious exploitation of the IoT deployment in the building scenario to generate anomalous traffic from an insider sensor towards an invalid IPv6 address out of the range of BMS services. To deal with this attack, the mitigation action performed is the data filtering of the malicious traffic detected. For this simulated attack, the next subsections provide the results of the validated test-cases and measured KPIs that are grouped in 3 main steps to evaluate the execution of the corresponding components of ANASTACIA framework.

3.1.2.1.1 Step 1 – R1: Results of Test-Cases with KPIs and Captured Data

The following subsection shows the measured KPIs of the validation executions according to the test cases defined in the deliverable D6.5. Below, the tables show the results of test-cases and also the KPI measurements in seconds for 5 executions of the performance tests. In this subsection, we describe the first step of the phase 1 of an infected sensor sending sensitive data to an outside endpoint shown in Figure 6.

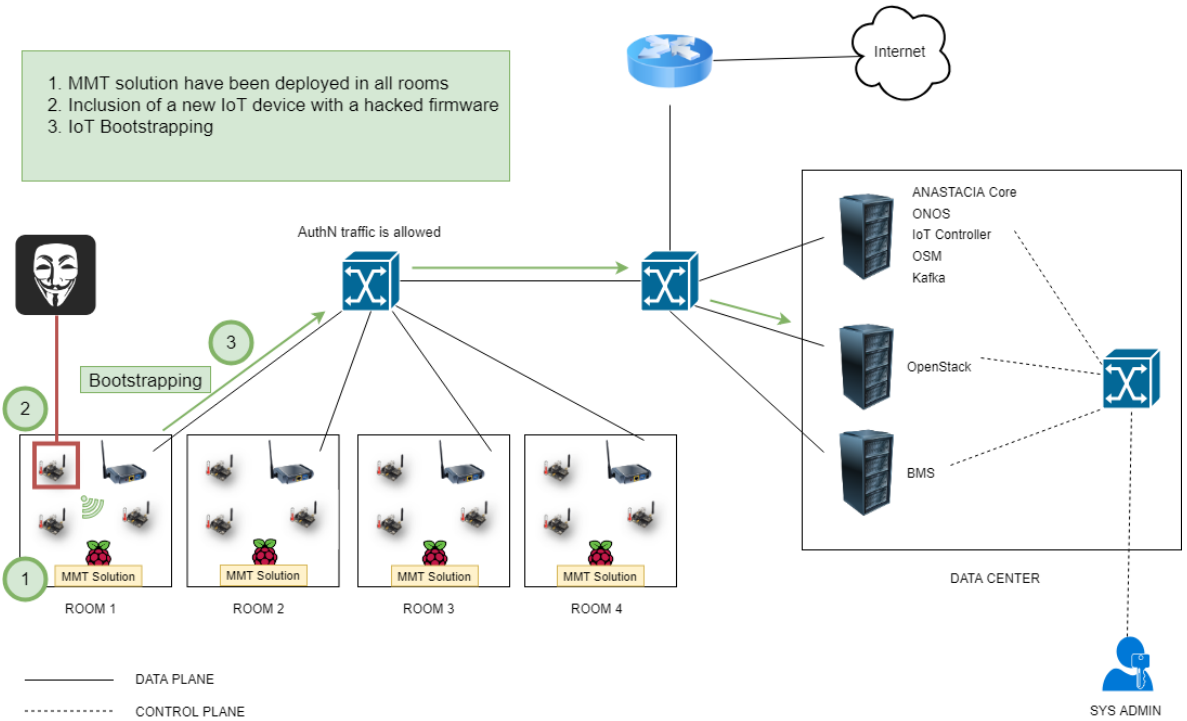


Figure 6: Diagram of Step 1 of Phase R1 Attack in Smart Building.

This step involves the initial operations of ANASTACIA framework for Monitoring and Proactive Security Policies related to the deployment of IoT devices in smart building. For this step, the following list indicates all actions performed by ANASTACIA components and the leader partners:

- TC1.1 MMT Solution is monitoring the 6LowPAN Network. (MONT)
- TC1.2 DSPS is GREEN. (AS/DG/MI)
- TC1.3 Proactive Security Policies definition: (UMU)
    - All traffic is denied by default.
    - Allow AuthN traffic between IoT devices and AuthN agent.
    - Allow DTLS+CoAP traffic from IoT Controller to IoT devices.
    - Authorize PUT/POST DTLS/HTTPS operations from IoT devices to BMS.
    - Allow DTLS/HTTPS traffic from IoT devices to BMS.
- TC1.4 Proactive Security Policies enforcement (AALTO)
- TC1.5 Inclusion of the malicious IoT device (ODINS)
- TC1.6 Bootstrapping process (ODINS/UMU)
- TC1.7 Reactive Security Policies enforcement (AALTO)

3.1.2.1.1.1 TC1.1 MMT Solution is monitoring the 6LowPAN Network. (MONT)

TC1.1 MMT Solution is monitoring the 6LowPAN Network. (MONT)	
Execution	• The MMT Probe software is started on the MMT-Probe machine. • The MMT-IoT Bridge tool is started on the MMT-Probe machine. • Send the activation command to the MMT-IoT Sniffer by using the CLI of MMT-IoT Bridge.
KPI(s)	1. Monitoring is enabled within less than 1 sec

The KPI values measured for the MMT monitoring of 6LowPAN Network is shown in the next table.

- This time indicates the launch of MMT-Probe on the machine took less than 1 second in the VM to load the MMT binaries.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,769	0,813	0,736	0,735	0,732	0,757

3.1.2.1.1.2 TC1.2 DSPS is GREEN. (AS/DG/MI)

TC1.2 DSPS seal has the initial status of GREEN (AS/DG/MI)	
Execution	• DSPS components are started
KPI(s)	• None

This step does not have KPIs to show.

3.1.2.1.1.3 TC1.3 Proactive Security Policies definition: (UMU)

TC1.3 Proactive Security Policies definition: (UMU)	

Execution	• HSPL Orchestration policy is defined by using the Policy Editor Tool in order to:
	  ○ Allow DTLS communication between IoT Controller and IoT devices
	  ○ Allow AuthN traffic between Authentication Agent and IoT devices.
	  ○ Authorise devices to PUT temperature in IoT broker. Include on AuthN success dependency.
	  ○ Allow DTLS traffic between IoT devices and IoT broker. Include on AuthN success dependency.
	• Press "Refinement" button in order to refine the HSPL-OP.
	• Press "Modify" button and add a security policy in order to deny AuthN traffic with the same priority.
	• Press "Refinement" button and verify the enforcement is not allowed.
	• Press "Modify" and remove the conflictive policy.
	• Press "Refinement" in order to refine the HSPL-OP by using the Policy Editor Tool.
	• Press "Enforcement" in order to request the MSPL-OP enforcement.
	• Advanced Proactive enforcement request by using directly an MSPL file in order to allow HTTPS between powerful IoT devices and end-point service.

KPI(s)	1. Policy refinement times. 2. Conflict detection times.

The next table shows the KPI values measured for the Policy refinement time and Conflict detection time.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	1,043	1,079	1,083	1,12	1,10	1,085
2. Time (seconds)	0,056	0,073	0,058	0,057	0,095	0,068

The explanations of the KPI values obtained are:

1. Time takes the policy refiner to refine the proactive High-level security orchestration policy into Medium-level security orchestration policy.
2. Time takes the conflict detector to detect conflict and dependencies during the refinement process.

3.1.2.1.1.4 TC1.4 Proactive Security Policies enforcement (AALTO)

TC1.4 Proactive Security Policies enforcement (AALTO)	

Execution	• Receiving the MSPL file through the REST API of the security orchestrator. The MSPL file consists of:
	  ○ Allow DTLS communication between IoT Controller and IoT devices.
	  ○ Allow AuthN traffic between Authentication Agent and IoT devices.
	  ○ Authorise devices to PUT temperature in IoT broker. Include on AuthN success dependency.
	  ○ Allow DTLS traffic between IoT devices and IoT broker. Include on AuthN success dependency.
	• The Security Orchestrator communicates with Security Enablers Provider.

	• The Security Enablers Provider returns ONOS as enabler. • The Security Orchestrator communicates with the Policy Interpreter to get the low configuration from the MSPL file. • The Security Orchestrator enforces the open flow rules by communicating with the REST API of ONOS controller.
KPI(s)	1. Required time to translate the MSPL files to low-level configuration. 2. Required time to generate the open flow rules and enforcing them.

The next table shows the KPI values measured for the proactive security policies enforcement.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	1,3020615	1,774500132	1,401800156	1,229719877	1,18888998	1,427020416
2. Time (seconds)	1,3603489	1,472899914	1,353439808	1,421270132	1,352210045	1,401989688

The explanations of the KPI values obtained are:

1. Time taken by the security orchestrator to request the translation of the MSPL for enabling the low-level configuration
2. Time taken by the security orchestrator to enforce the open flow rules in the SDN controller

3.1.2.1.1.5 TC1.5 Inclusion of the malicious IoT device (ODINS)

TC1.5 Inclusion of the malicious IoT device (ODINS)	
Execution	• A malicious IoT device is turn on inside the building network. • The IoT device starts its execution for transmitting sensing data to a destination IP address of the outside endpoint.
KPI(s)	1. Obtain the elapsed time for processing the received query and modified the IP destination address in the internal configuration.

The next table shows the KPI values measured for the inclusion of the malicious IoT device.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	2,442	2,422	2,481	2,461	2,444	2,450

The explanation of the KPI values obtained is:

1. The time it takes from a device within the system to establish a DTLS connection and send a command to reconfigure the device

3.1.2.1.1.6 TC1.6 Bootstrapping process (ODINS/UMU)

TC1.6 Bootstrapping process (ODINS/UMU)	
Execution	• IoT device is restarted in order to use the new configuration and start the bootstrapping process • IoT device sends a query for network access to PANA Agent • PANA Agent evaluates if the identity of new device is valid or invalid • If it is valid, PANA Agent sends a publication of new device to IoT Register
KPI(s)	1. Obtain the response time for network authentication between IoT device and PANA Agent 2. Obtain the response time for new device publication between PANA Agent and IoT Register

The next table shows the KPI values measured for the Bootstrapping process.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	2,2330	1,315999985	1,332999945	1,328000069	1,306999922	1,503
2. Time (seconds)	0,395	0,362	0,385	0,364	0,37	0,375

The explanations of the KPI values obtained are:

1. The time it takes to perform the network access authentication using PANA using the EAP-PSK method to authenticate to gain access to the network.
2. The time it takes the PAA to notify the system of a new entity just performed the bootstrapping and can access the network.

3.1.2.1.1.7 TC1.7 Reactive Security Policies enforcement (AALTO)

TC1.7 On AuthN success dependency Security Policies enforcement (AALTO)	

Execution	• IoT Register receives IoT register notification.
	• IoT Register registers the IoT device in the system model.
	• IoT Register notifies the event to the Security Orchestrator.
	• Security Orchestrator verifies policies with pending event dependencies:
	  ○ Authorise PUT temperature in IoT Broker.
	  ○ Allow DTLS traffic between IoT device and IoT broker.
	• Security Orchestrator communicates with the Security Enablers Provider for selecting the best enabler.
	• The Security Enablers Provider selects and returns ONOS as enabler to the Security Orchestrator.
	• Security Orchestrator requests policies translation to policy Interpreter.
	• Policy Interpreter verifies policy conflicts or dependencies by using the conflict detector.
	• Policy Interpreter retrieves enabler plugins from the Security Enabler Provider and translates the MSPL-OP into final configurations.
	• Policy Interpreter returns final configurations and conflicts and dependencies

	detection to Security Orchestrator. • Security Orchestrator generates the open flow rules from the received configuration. • Security enforces the generated open flow rules through the REST API of ONOS. • Security Orchestrator updates the System Model by adding the status about the enforced MSPL file.
KPI(s)	1. Policy translation times. 2. Conflict detection times. 3. Required time to translate the MSPL files to low level configuration. 4. Required time to generate the open flow rules and enforcing them.

The next table shows the KPI values measured for AuthN success dependency and Security Policies enforcement.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,636	0,657	0,681	0,432	0,783	0,6378
2. Time (seconds)	0,06	0,0517	0,0522	0,0513	0,054	0,05384
3. Time (seconds)	1,2134636	1,2234636	1,1954636	1,1834736	1,1605641	1,1952857
4. Time (seconds)	1,3747606	1,0891919	1,4132726	1,3221588	1,36528	1,31293278

The explanations of the KPI values obtained are:

1. Time taken by the policy translator to translate authorisation policies into low-level configurations for PDP and SDN network.
2. Time taken by the conflict detector to detect conflicts and dependencies for the medium-level security policies.
3. Time taken by the security orchestrator to request the translation of MSPL to enable low level configuration
4. Time taken by the security orchestrator to enforce the open flow rules in the SDN controller

3.1.2.1.2 Step 2 – R1: Results of Test-Cases with KPIs and Data Captured

This subsection elaborates the test cases that correspond to the step 3 (see Figure 7). It involves the phase R1, where an infected sensor tries to send data to a server that is outside of the system. Dealing with this kind of behavior shows how the ANASTACIA framework performs detecting attacks, how it reacts and how the seal is managed in regard to an IoT device in the Smart Building.

- TC2.1 IoT device tries to establish a DTLS connection to outside. (ODINS)
- TC2.2 MMT solution detects invalid destination at 6lowPAN level. (MONT)
  - MMT solution notifies the issue.
- TC2.3 Incident detector-VDSS processes the notification and generates alert and mitigations list. (ATOS)
- TC2.4 The SAS forwards alert to DSPS. (CNR)
- TC2.5 DSPS changes to YELLOW. (AS/DG/MI)
- TC2.6 The MAS consume list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS and the VDSS. (MMT)

- TC2.7 The SAS forwards mitigation action (mitigating) to DSPS. (CNR)
- TC2.8 DSPS maintains YELLOW state. (AS/DG/MI)
- TC2.9 SAS keeps polling SO system model. (CNR)

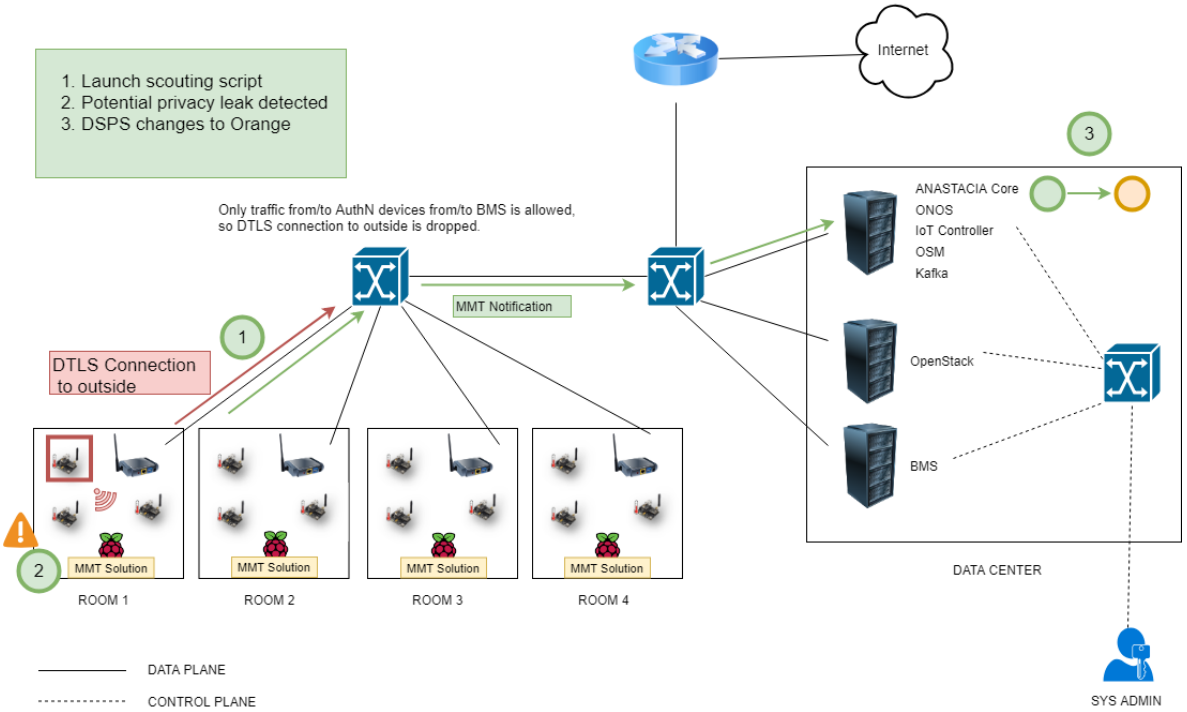


Figure 7: Diagram of Step 2 of Phase R1 Attack in Smart Building.

3.1.2.1.2.1 TC2.1 IoT device tries to establish a DTLS connection to outside (ODINS)

TC2.1 IoT device tries to establish a DTLS connection to outside. (ODINS)	
Execution	• IoT Device starts the DTLS connection to an IP destination address outside the smart building network in order to send the data messages using DTLS+COAP protocols.
KPI(s)	1. Is the rogue device able to establish the DTLS connection to the IP destination address outside the system?

The next table shows the KPI values measured for an IoT device that tries to establish a DTLS connection to outside the system.

Execution Num.	1	2	3	4	5	Mean
1. TRUE or FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE

The explanations of the KPI values obtained are:

1. Determining if it is possible for an outsider to establish a DTLS connection with one of the IoT devices

3.1.2.1.2.2 MT solution detects invalid destination at 6lowPAN level (MONT)

TC2.2 MMT solution detects invalid destination at 6lowPAN level. (MONT)	
Execution	• The script is launched. • The malicious device tries to open a connection to a server outside the trusted IoT network. • MMT-IoT Sniffer extracts the IoT traffic form the network and redirects it to MMT-Probe. • MMT-Probe analyses the traffic according to the loaded security rules.
KPI(s)	1. Check that the attack was correctly triggered and detected by the MMT Probe.

The next table shows the KPI values measured for the MMT solution that detects an invalid destination at 6lowPAN level.

Execution Num.	1	2	3	4	5	Mean
1. TRUE or FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

The explanations of the KPI values obtained are:

1. MMT Probe is able to detect successfully the attack in all occasions.

3.1.2.1.2.3 TC2.3 Incident detector-VDSS process the notification and generates alert and mitigations list (ATOS)

TC2.3 Incident detector – process the notification and generates alert (ATOS)	
Execution	• Raw "Untrusted destination detected" events are submitted by the MMT probe to the Data Filtering and pre-processing broker • The storm topology of the Data Filtering and pre-processing broker process and sends the events through rsyslog to the Incident Detector • The Incident Detector receives the events through its agent, which normalizes them to a common format and submits it to the correlator engine of the Incident Detector, which processes it and generates a "Potential data leakage detected" alert • The alert is pushed to a the RabbitMQ queue, exchange alarms, to be consumed by DSPS and VDSS
KPI(s)	1. Obtain the processing time in the Incident Detector to normalize the event, correlate it, generate the alert and send it to the RabbitMQ exchange

The next table shows the KPI values measured for the Incident detector, which process the notification and generates an alert:

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,1	0,07	0,07	0,07	0,1	0,082

The explanation of the KPI values obtained is:

1. The values include the processing time for the Incident Detector since the notification is received from Kafka Broker until is sent to RabbitMQ. Communication times between Kafka Broker - Incident Detector and Incident Detector - RabbitMQ are also included in the measurements.

3.1.2.1.2.4 TC2.4 The SAS forwards alert to DSPS (CNR)

TC2.4 The SAS forwards alert to DSPS (CNR)	
Execution	• SAS consume the RabbitMQ queue by retrieving the generated alert • The SAS forwards the alert, as is, to another RabbitMQ server shared with the DSPS
KPI(s)	1. The time passing between the reception of the alert from the RabbitMQ queue dedicated to VDSS-to-SAS communication and the sending of the (same) alert to the RabbitMQ queue dedicated to SAS-to-DSPS communication

The next table shows the KPI values measured for the forwarding of the alerts to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,08	0,1	0,05	0,04	0,06	0,066

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, sending the alert to the DSPS, measured by the component itself.

3.1.2.1.2.5 TC2.5 DSPS changes to YELLOW (AS/DG/MI)

TC2.5 DSPS changes to YELLOW (AS/DG/MI)	
Execution	• New alert is received by DSPS (AMQP queue consumption) • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until stored in DSPS secured storage 3. The time measurement since DSPS seal creation receives alert until DSPS GUI is notified of the new seal

The next table shows the KPI values measured for changing the DSPS to YELLOW.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145
3. Time (seconds)	0,017	0,023	0,021	0,032	0,019	0,0224

The explanations of the KPI values obtained are:

1. Time measurement is negligible.
2. Main KPI of the DSPS. Value is under the acceptable range. When the DSPS server needs to write the seal on the on-chain storage (blockchain-based with BFT consensus) it takes in average 1.5 seconds. When it needs to write the seal log on the off-chain storage (based on shamir secret schema) it takes in average 1.0 seconds. These times depend a lot from the number of used nodes and their hardware so potentially it could be improved, however we don't have enough resources to update the infrastructure.
3. Time measurement is negligible.

3.1.2.1.2.6 TC2.6 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS and the VDSS (MMT)

TC2.6 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)	
Execution	• An alert is published by the VDSS in the RabbitMQ channel. • The MAS receives the alert and starts the mitigation strategy processing. • The MAS contacts the auxiliary services to retrieve additional data to compute the MSPL (AM and SMS). • The MAS generates the MSPL and sends it to the SO. • The MAS sends the computed mitigation to the SAS and the VDSS.
KPI (s)	1. Obtain the general time elapsed to compute the mitigation: starting from the reception of the alert until the MSPL is sent to the SO.

The next table shows the KPI values measured for the MAS consuming the list of recommendation and as a consequence, generate the final mitigation in a form of MSPL.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	3,268	2,431	2,329	2,422	2,646	2,6192

The explanations of the KPI values obtained are:

1. The value encompasses the process of the MAS contacting the external services while computing the reaction, thus this measurement also includes the response time of the System Model and Assets Model.

3.1.2.1.2.7 TC2.7 The SAS forwards mitigation action (mitigating) to DSPS (CNR)

TC2.7 The SAS forwards mitigation action (mitigating) to DSPS (CNR)	
Execution	• The MAS sends to the SAS the MSPL file, related to the given alert, previously received by the VDSS • The SAS encapsulates received data inside the previously received alert; in particular, the following attributes are added to the original alert • The SAS forward the enriched alert to the RabbitMQ server, shared with the DSPS
KPI(s)	1. The time passing between the reception of the MSPL data from the MAS and the sending of the enriched alert to the RabbitMQ queue

The next table shows the KPI values measured for SAS forwarding the mitigation action.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,07	0,04	0,03	0,03	0,03	0,04

The explanations of the KPI values obtained are:

1. The value includes the processing time of the SAS component, measured by the component itself.

3.1.2.1.2.8 TC2.8 DSPS maintains YELLOW state (AS/DG/MI)

TC2.8 DSPS maintains YELLOW state (AS/DG/MI)	
Execution	• Alert with mitigation action is received (mitigating) • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until processed

The next table shows the KPI values measured for the DSPS maintaining its YELLOW state.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145

The explanations of the KPI values obtained are the same as T2.5 (see 3.1.2.1.2.5).

3.1.2.1.2.9 TC2.9 SAS keeps polling SO system model (CNR)

TC2.9 SAS keeps polling SO system model (CNR)	
Execution	• The SAS polls the SM for a specific deployment identifier, previously received by the MAS
KPI(s)	1. Measure the time passing between the reception of the message from the MAS and the first polling to the SM

The next table shows the KPI values measured for the SAS polling continuously the SO system model.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,09	0,05	0,04	0,03	0,04	0,05

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.1.3 Step 3 – R1: Results of Test-Cases with KPIs and Data Captured

Continuing the actions in step 2, there is a counter measure that is put in place by the UTRC Data Analysis Agent in this step. The new element is deployed and configured dynamically, so to start generating verdicts for particular set of sensors that are deemed potentially malicious in the previous step. In the current step the ANASTACIA framework will monitor with a second agent to provide a high level of fidelity when an alarm is triggered and sent to the MAS, SAS, VDSS and SO. This test (Figure 8) is build following the steps in the listed next:

- TC3.1 Reactive policy for orchestration enforcement (AALTO)
  - Orchestrator decides to reconfigure the UTRC agent (AALTO/THALES)
- TC3.2 UTRC agent enables behavioral analysis on affected device and sends ACK (UTRC)
- TC3.3 SO updates system model (AALTO)
- TC3.4 SAS polls SO system model and sends to DSPS the alert with mitigated=True
- TC3.5 DSPS changes security status GREEN state (AS/DG/MI)

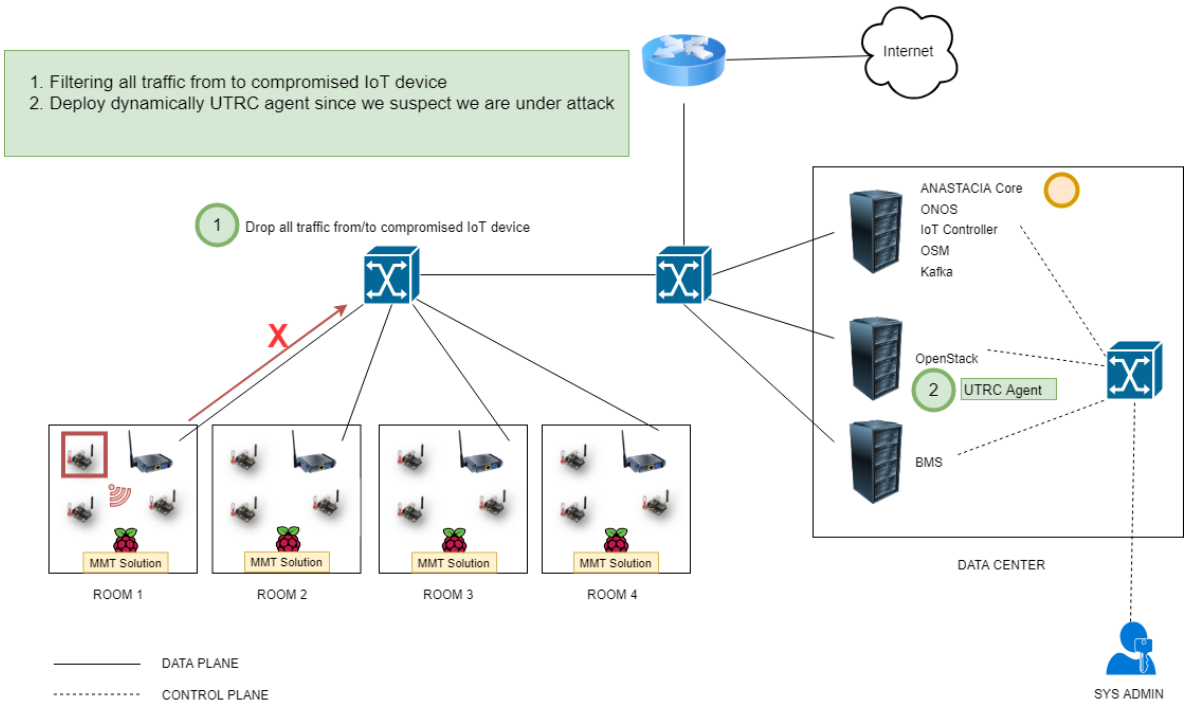


Figure 8: Diagram of Step 3 of Phase R1 Attack in Smart Building.

3.1.2.1.3.1 TC3.1 Reactive policy for orchestration enforcement (UMU/AALTO/THALES)

TC3.1 Reactive policy for orchestration enforcement (UMU/AALTO/THALES)	

Execution	• Security Orchestrator verifies MSPL-OP policy which contains:
	  ○ Filtering
	  ○ Monitoring IoT data.
	• Security Orchestrator communicates with the Security Enablers Provider for selecting the best enabler.
	• Using the Security Enablers Provider, the Security Orchestrator decides to use ONOS as enabler for filtering and UTRC agent for data analysis.
	• Security Orchestrator requests policies translation to policy Interpreter.
	• Policy Interpreter verifies policy conflicts or dependencies by using the conflict detector.
	• Policy interpreter retrieves enabler plugins from the Security Enabler Provider and translates the MSPL-OP into final configurations.
	• Policy Interpreter returns final configurations and conflicts and dependencies detection to Security Orchestrator.
	• Security Orchestrator generates the open flow rules from the received configuration.
	• Security enforces the generated open flow rules through the REST API of ONOS to filter the traffic and enables the communication between the IoT device and the UTRC agent.
	• Security Orchestrator configures the UTRC agent through its offered REST API.

KPI(s)	1. Policy translation times. 2. Conflict detection times. 3. Required time to translate the MSPL files to low level configuration. 4. Required time to generate the open flow rules and enforcing them. 5. Requires time to configure the UTRC agent.

The next table shows the KPI values measured for the reactive policy for orchestration and enforcement.

Execution Num.	1	2	3	4	5	Mean

1. Time (seconds)	0,372	0,216	0,258	0,354	0,21	0,282
2. Time (seconds)	0,0181	0,0187	0,0179	0,0181	0,021	0,01876
3. Time (seconds)	0,5999162	0,5313601494	0,5503499508	0,5445599556	0,6967298985	0,5565465639
4. Time (seconds)	0,4398163	0,5570700169	0,5300400257	0,5523300171	0,6027600765	0,5198140899
5. Time (seconds)	0,3303483	0,2065799236	0,2076301575	0,22570014	0,2444500923	0,2425646303

The explanations of the KPI values obtained are:

1. Time taken by the policy translator to translate reactive filtering and monitoring policies into low-level configurations for SDN and UTRC agent.
2. Time taken by the conflict detector to detect conflicts and dependencies between the reactive medium-level security policies.
3. Time taken by the security orchestrator to request the translation of MSPL for enabling low level configuration
4. Time taken by the security orchestrator to enforce the open flow rules on the SDN controller
5. Time taken by the security orchestrator to configure the UTRC agent

3.1.2.1.3.2 TC3.2 UTRC agent enables behavioural analysis on affected device and sends ACK (UTRC)

TC3.2 UTRC agent enables behavioural analysis on affected device and sends ACK (UTRC)	
Execution	• Agent continuously receives data from Kafka broker on topic "IoTBrokerTopic" • Policy Interpreter sends information about new MSPL to SO • SO sends new MSPL to Policy Interpreter • Policy Interpreter sends new configuration to UTRC Data Analysis Agent
KPI(s)	1. Verdict generation per single data point is less than 300 msec

The next table shows the KPI values measured for enabling behavioural analysis on affected devices through the UTRC agent.

Execution Num.	**1**	**2**	**3**	**4**	**5**	**Mean**
1. Time (seconds)	0,285	0,228	0,269	0,227	0,24	0,2498

The explanations of the KPI values obtained are:

1. The time is measured between message reception from Kafka broker for parsing and processing in the UTRC Data Analysis to generate a verdict.

3.1.2.1.3.3 TC3.3 SO updates system model (AALTO)

TC3.3 SO updates system model (AALTO)	
Execution	• Security Orchestrator sends post request to update the enforcement API status
KPI(s)	1. Measure the needed time for retrieving and updating the information.

The next table shows the KPI values measured for updating the system model.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,1190596	0,1163699627	0,1015000343	0,1048099995	0,1017799377	0,1104348991

The explanations of the KPI values obtained are:

1. The time is measured in the security orchestrator to update the information about the network in the system model.

3.1.2.1.3.4 TC3.4 SAS polls SO system model and send to DSPS the alert with enabled mitigation (CNR)

TC3.4 SAS polls SO system model and send to DSPS the alert with enabled mitigation	
Execution	• The SAS polls the SM for a specific deployment identifier, previously received by the MAS • The SM answers with deployment data • The SAS extracts relevant fields from the data received by the SM • The SAS enriches (again) the alert to include relevant information received from the SM • The SAS forwards the enriched alert to the DSPS, through the dedicated RabbitMQ queue
KPI(s)	1. The time spent between the reception of the (latest/valid) message from the SM and the sending of the enriched alert to the DSPS

The next table shows the KPI values measured for polling the SO system model and updating the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,05	0,03	0,06	0,04	0,02	0,004

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.1.3.5 TC3.5 DSPS changed security status GREEN state (AS/DG/MI)

TC3.5 DSPS changed security status GREEN state (AS/DG/MI)	
Execution	• Alert with mitigated flag/information is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until stored in DSPS secured storage 3. The time measurement since DSPS seal creation receives alert until DSPS GUI is notified of

The next table shows the KPI values measured for changing the security status to GREEN.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145
3. Time (seconds)	0,017	0,023	0,021	0,032	0,019	0,0224

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.1.2.2 Phase R2: Attack with an infected temperature sensor sending high temperature to activate fire alarm

In this section, the validation of the second phase of the use case of Smart Building in the room2 of PEANA experimental laboratories is described. This phase simulates an attacker that manipulates a temperature sensor in order to send high values to activate the fire alarm and the evacuation protocol.

In this phase, it exploits the data measured about the environment to produce anomalous information for creating an emergency alert in the building. Suddenly, the temperature value is too high and an anomalous behavior is detected by AI-based data analysis. To tackle this attack, the mitigation action performed is the data filtering of the compromised sensor device. To evaluate this simulated attack and the performance of ANASTACIA framework, the next subsections provide the results of the validated test-cases and measured KPIs that are grouped in 2 main steps.

3.1.2.2.1 Step 4 – R2: Results of Test-Cases with KPIs and Data Captured

The next phase (step 4) in this subsection describes the following test-cases regarding an infected temperature sensor that sends malicious data in the form of erroneous high temperature readings, to the Building Management System (BMS). This will show the ANASTACIA framework Data Monitoring, Alert Detection, Reaction and Seal Management in regard to high temperature values coming from an IoT device in the Smart Building (see Figure 9: Diagram of Step 4 of Phase R2 Attack in Smart Building.Figure 9).

- TC4.1 Misbehavior IoT device: (ODINS)
  - IoT device starts sending wrong temperature values.
  - Fire alarm is turned on
- TC4.2 UTRC agent detects the issue and sends alarm to Incident Detector. (UTRC)
- TC4.3 Incident detector – process the notification and generates alert and mitigations list (ATOS)
- TC4.4 The SAS forwards alert to DSPS. (Alert level between 8 and 10) (CNR)
- TC4.5 DSPS changes to RED (AS/DG/MI)
- TC4.6 VDSS generates a list of recommended mitigations (ATOS)
- TC4.7 The MAS consume list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)
- TC4.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)

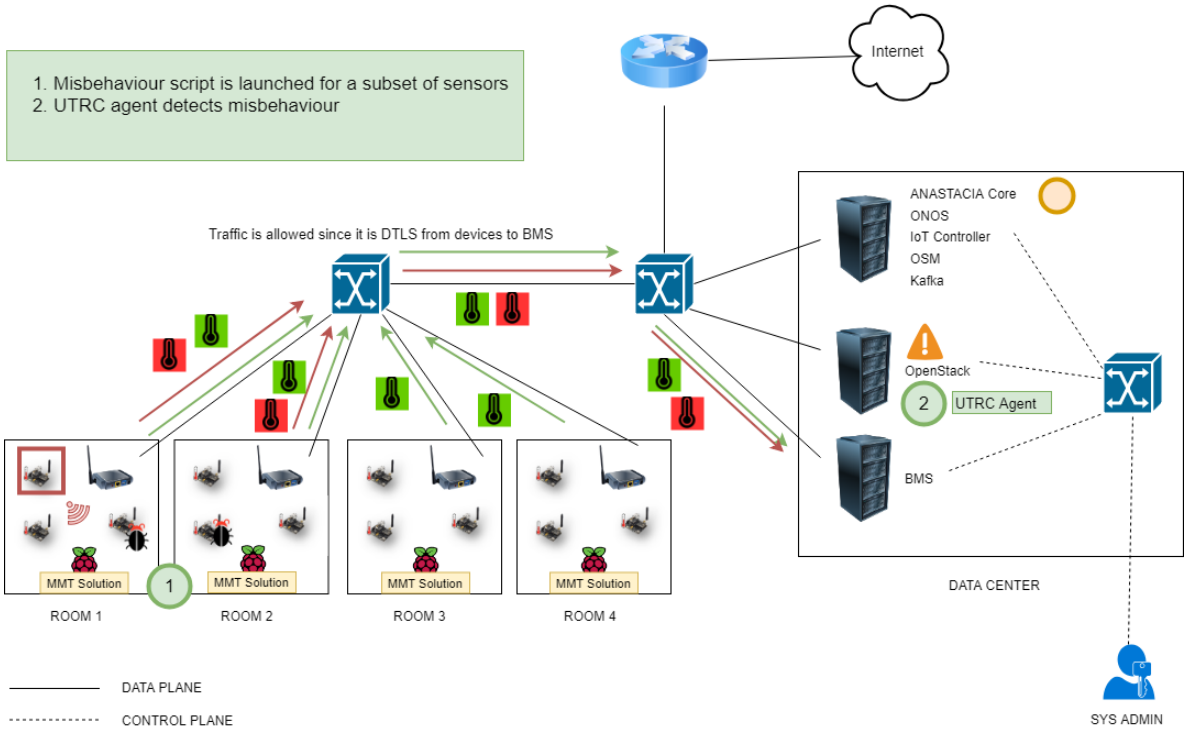


Figure 9: Diagram of Step 4 of Phase R2 Attack in Smart Building.

3.1.2.2.1.1 TC4.1 Misbehavior script/IoT (ODINS)

TC4.1 Misbehaviour script/IoT (ODINS)	
Execution	• IoT device receives a DTLS+COAP query to modify the temperature value generated to be sent towards the IoT broker. • IoT broker receives, stores and forwards temperature values in JSON format to inform the ANASTACIA framework. • Receiving JSON payload of temperature messages coming from OdinS IoT-broker on Kafka Broker being logged on Data Analysis.
KPI(s)	1. Obtain the elapsed time for transmitting the temperature data from IoT devices to be received by the Data Analysis component through the BMS IoT broker and the ANASTACIA Kafka broker.

The next table shows the KPI values measured for a misbehavior generated by an IoT device.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	1,039000034	0,2069997787	0,2109999657	0,2039999962	0,2150001526	0,375

The explanations of the KPI values obtained are:

1. The time it takes from a device that is able to publish a temperature reading to when the BMS sends the message to all the subscribed entities to the BMS and they receive it

3.1.2.2.1.2 TC4.2 UTRC agent detects the issue and sends alarm to Incident Detector (UTRC)

TC4.2 UTRC agent detects the issue and sends alarm to Incident Detector. (UTRC)	
Execution	• Agent receives new IoT temperature sensor information (attacked values) • Agent detects anomalous behaviour on the sensor • Agent generates new verdict about the incident corresponding to sensor
KPI(s)	1. New verdict is generated with less than 300ms 2. Successful detection ratio is higher than 80%

The next table shows the KPI values measured for the detection of anomalies and send the alarms to the Incident Detector.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,235	0,284	0,235	0,232	0,241	0,2454
2. Detection Ratio (%)	100%	100%	100%	100%	100%	100%

The explanations of the KPI values obtained are:

1. The time is measured between message reception from Kafka broker and its parsing and processing in the UTRC Data Analysis for generating a verdict.
2. This metric represents the percentages of correct anomaly detections made by UTRC Data Analysis agent model.

3.1.2.2.1.3 TC4.3 Incident detector-VDSS processes the notification and generates alert and mitigations list (ATOS)

TC4.3 Incident detector – processes the notification and generates alert (ATOS)	
Execution	• Raw “Data analysis report: Anomaly detected” events are submitted by the MMT probe to the Data Filtering and pre-processing broker • The storm topology of the Data Filtering and pre-processing broker processes and sends the events through rsyslog to the Incident Detector • The Incident Detector receives the events through its agent, which normalizes them to a common format and submits it to the correlator engine of the Incident Detector, which processes it and generates a “Man in the Middle on IoT data” alert • The alert is pushed to a the RabbitMQ queue, exchange.alarms, to be consumed by DSPS and VDSS
KPI(s)	1. Obtain the processing time in the Incident Detector for the received event and send the generated alert to the RabbitMQ exchange

The next table shows the KPI values measured for the process of notifications and generation of alerts.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,21	0,07	0,08	0,15	0,08	0,118

The explanations of the KPI values obtained are:

1. Those values includes the processing time for the Incident Detector since the notification is received from Kafka Broker until is sent to RabbitMQ. Communication times between Kafka Broker - Incident Detector and Incident Detector - RabbitMQ are also included in the measurements.

3.1.2.2.1.4 TC4.4 The SAS forwards alert to DSPS. (Alert level between 8 and 10) (CNR)

TC4.4 The SAS forwards alert to DSPS. (Alert level between 8 and 10) (CNR)	
Execution	• SAS consume the RabbitMQ queue by retrieving the generated alert • The SAS forwards the alert, as it is, to another RabbitMQ server shared with the DSPS
KPI(s)	1. The time passing between the reception of the alert from the RabbitMQ queue dedicated to VDSS-to-SAS communication and the sending of the (same) alert to the RabbitMQ queue dedicated to SAS-to-DSPS communication

The next table shows the KPI values measured for forwarding alerts to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,04	0,05	0,06	0,05	0,06	0,052

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.2.1.5 TC4.5 DSPS changes to RED (AS/DG/MI)

TC4.5 DSPS changes to RED (AS/DG/MI)	
Execution	• Security alert with high alert level is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until stored in DSPS secured storage 3. The time measurement since DSPS seal creation receives alert until DSPS GUI is notified of the new seal

The next table shows the KPI values measured for DSPS changing to RED state.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145
3. Time (seconds)	0,017	0,023	0,021	0,032	0,019	0,0224

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.1.2.2.1.6 TC4.6 VDSS generates a list of recommended mitigations (ATOS)

TC4.6 VDSS generates a list of recommended mitigations (ATOS)	
Execution	• The VDSS received an alert from the Incident Detector • The VDSS retrieves from the Assets Model the list of available mitigation strategies capable of mitigating the detected alert • The VDSS retrieves from the System Model Service information about the devices affected by the incident • The VDSS calculates suitability scores for every mitigation strategy and pushes it to an exchange.recommendations queue, to be consumed by the MAS and the SAS.
KPI(s)	1. Obtain the elapsed time since the VDSS receives the alert until it sends the recommendations to RabbitMQ

The next table shows the KPI values measured for the generation by the VDSS of a list of recommended mitigations.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,38	0,32	0,33	0,39	0,32	0,348

The explanations of the KPI values obtained are:

1. Those values includes the time to receive from RabbitMQ the alert generated by the Incident Detector, processing it by getting additional information about the affected asset from the Assets Model and the System Model APIs, to calculate the ranking for each of the mitigation strategies and sending the recommendations for the MAS to the RabbitMQ server.

3.1.2.2.1.7 TC4.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)

TC4.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)	
Execution	• An alert is published by the VDSS in the RabbitMQ channel. • The MAS receives the alert and starts the mitigation strategy processing. • The MAS contacts the auxiliary services to retrieve additional data to compute the MSPL (AM and SMS). • The MAS generates the MSPL and sends it to the SO. • The MAS sends the computed mitigation to the SAS and the VDSS.
KPI(s)	1. Obtain the general time elapsed to compute the mitigation: starting from the reception

	of the alert until the MSPL is sent to the SO.

The next table shows the KPI values measured for the consumption of recommendations by the MAS and generate the mitigations in MSPL format.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	4,838	5,01	4,924	5,02	4,882	4,9348

The explanations of the KPI values obtained are:

1. The MAS contacts external services while computing the reaction, thus this measurement also includes the response time of the System Model and Assets Model.

3.1.2.2.1.8 TC4.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)

TC4.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)	
Execution	• The MAS sends to the SAS the MSPL file, related to the given alert, previously received by the VDSS • The SAS encapsulates received data inside the previously received alert; in particular, the following attributes are added to the original alert • The SAS forward the enriched alert to the RabbitMQ server, shared with the DSPS
KPI(s)	1. The time passing between the reception of the MSPL data from the MAS and the sending of the enriched alert to the RabbitMQ queue

The next table shows the KPI values measured for forwarding the mitigations by the SAS to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,03	0,04	0,04	0,12	0,04	0,054

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.2 Step 5 – R2: Results of Test-Cases with KPIs and Data Captured

This subsection elaborates step 5, which deals with the integration amongst the reaction module components of ANASTACIA. Particularly, the DSPS maintains a state of alert and the other component prepare the mitigations to the threat that was introduced in the previous step (step 4). The main objective of this test is to ready the security policies that are to be used later in the steps that follow to mitigate the attack (Policy Interpreter) and test the SO and SAS communication regarding the system model. This composed by the following sequence (Figure 10):

- TC5.1 Reactive policy for orchestration enforcement (UMU/AALTO)
  - Turn off device which could reach other Rooms (IoT control).
  - Reboot compromised device (IoT control).
  - Drop traffic for compromised device (Filtering).
  - Monitor the BMS network.
  - Mirroring traffic in order to allow traffic inspection (Traffic Forward (Mirroring)).
- TC5.2 SO updates system model. (AALTO)
- TC5.3 SAS polls SO system model and sends to DSPS the alerts.
- TC5.4 DSPS remains RED. (AS/DG/MI) – SEE STEP3 as for “mitigating”

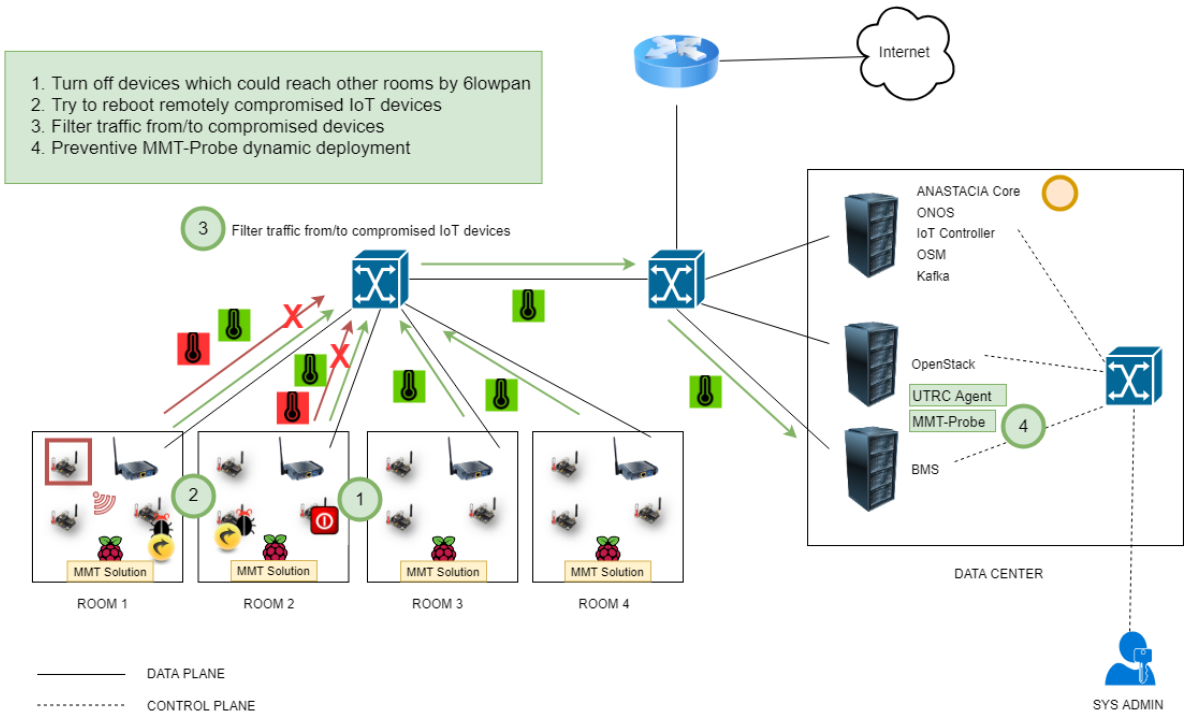


Figure 10: Diagram of Step 5 for the Phase R2 Attack in Smart Building

3.1.2.2.1 TC5.1 Reactive policy for orchestration enforcement (UMU/AALTO)

TC5.1 Reactive policy for orchestration enforcement (AALTO)	

Execution	• Security Orchestrator verifies MSPL-OP policy which contains:
	  ○ IoT control (Turn off).
	  ○ IoT control (Reboot)
	  ○ Filtering.
	  ○ BMS Network Monitoring.
	  ○ Traffic Forward (Mirroring)
	• Security Orchestrator communicates with the Security Enablers Provider for selecting the best enabler.
	• Security Orchestrator decides to use IoT Controller enabler for IoT control, ONOS for filtering and MMT Probe for network monitoring.
	• Security Enablers Provider informs the Security Orchestrator to use IoT controller enabler and ONOS enabler for mirroring and filtering the traffic.
	• Security Orchestrator requests policies translation from the Policy Interpreter.
	• Policy Interpreter verifies policy conflicts or dependencies by using the conflict detector.
	• Policy interpreter retrieves enabler plugins from the Security Enabler Provider and translates the MSPL-OP into final configurations.
	• Policy Interpreter returns final configurations and conflicts and dependencies detection to Security Orchestrator.
	• Security Orchestrator communicates with IoT controller to turn off and reboot the IoT devices.
	• Security Orchestrator communicates with NFV orchestrator to instantiate the network service instance (NSI) using the network service descriptor (NSD) of the MMT Probe.
	• NFV orchestrator communicates with OpenStack VIM to instantiate and configure the MMT Probe.
	• Security Orchestrator generates the open flow rules from the received configuration.
	• Security enforces the generated open flow rules through the REST API of ONOS to filter the traffic and mirror the generated traffic from the IoT devices to the MMT Probe for

	network monitoring. • Security Orchestrator configures the MMT Probe.
KPI(s)	1. Policy translation times. 2. Conflict detection times. 3. IoT Controller enforcement times. 4. MMT Probe instantiation times. 5. MMT Probe configuration times. 6. Traffic mirroring and filtering times.

The next table shows the KPI values measured for the enforcement of the reactive policy for orchestration.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,708	0,679	0,678	0,703	0,673	0,6882
2. Time (seconds)	0,713	0,716	0,72	0,746	0,716	0,7222
3. Time (seconds)	0,886	0,863	0,896	0,891	0,778	0,8628
4. Time (seconds)	39,19197989	34,14253998	28,29281998	34,06030011	34,04206991	33,92190999
5. Time (seconds)	11,23938012	11,23917007	11,55706	11,19225001	11,22160006	11,30696505
6. Time (seconds)	0,9312798977	0,9493200779	0,9873099327	0,8775398731	0,959800005	0,9363624454

The explanations of the KPI values obtained are:

1. Time taken by the policy translator to translate reactive IoT control, network and monitoring policies into low-level configurations for IoT Controller, MMT-Probe and SDN network.
2. Time taken by the conflict detector to detect conflicts and dependencies between the reactive medium-level security policies.

3. Time taken by the IoT controller to request the IoT operations to ioT devices
4. Time taken by the security orchestrator to instantiate MMT Probe VNF and configure the control plane
5. Time taken by the security orchestrator to configure and start MMT Probe using SSH
6. Time taken by the security orchestrator to enforce the mirroring and filtering intents on the SDN controller

3.1.2.2.2.2 TC5.2 SO updates system model (AALTO)

TC5.2 SO updates system model (AALTO)	
Execution	• Security Orchestrator sends post request to update the enforcement API status
KPI(s)	1. Measure the request time

The next table shows the KPI values measured for updating the system model.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,1098999977	0,1198699474	0,1057600975	0,1063101292	0,1078300476	0,110460043

The explanations of the KPI values obtained are:

1. Time the security orchestrator takes to update the enforcement API status from pending to success or fail.

3.1.2.2.3 TC5.3 SAS polls SO system model and sends to DSPS the alert (CNR)

TC5.3 SAS polls SO system model and sends to DSPS the alert	
Execution	• The SAS polls the SM for a specific deployment identifier, previously received by the MAS • The SM answers with deployment data • The SAS extracts relevant fields from the data received by the SM • The SAS enriches (again) the alert to include relevant information received from the SM • The SAS forwards the enriched alert to the DSPS, through the dedicated RabbitMQ queue
KPI(s)	1. The time spent between the reception of the (latest/valid) message from the SM and the sending of the enriched alert to the DSPS

The next table shows the KPI values measured for polling the system model by the SAS and send alerts to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,050	0,040	0,060	0,030	0,040	0,044

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.2.4 TC5.5 DSPS remains RED (AS/DG/MI)

TC5.5 DSPS remains RED (AS/DG/MI) – Same as TC2.8 but with RED status.	
Execution	• Alert with mitigation action is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until processed

The next table shows the KPI values measured for the DSPS to remain in RED state.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.1.2.3 Phase R3: Slow-DoS attack against BMS web service

This section elaborates the validation of the phase R3 regarding the Slow-DoS attack against the BMS web service. The evaluation of this phase is done in the next two subsections. The first subsection will deal with the tests regarding the launch of the Slow-DoS attack against the BMS web service, the detection of the attack and the notification to the different security components that update the system status and generate the actions to manage the attack. The next step deals with the isolation of malicious or infected devices.

For this attack, the next two subsections provide the results of the validated test-cases and measured KPIs to evaluate the execution of the corresponding components of ANASTACIA framework.

3.1.2.3.1 Step 6 – R3: Results of Test-Cases with KPIs and Data Captured

This step showcases the start of the Slow-DoS attack (Figure 11) and the process of detection, notification and reaction by the different components of the ANASTACIA framework, readying the system for the mitigation. Next we can see the sequence that encompasses this step.

- TC6.1 SlowDos attack is launched: (CNR)
    - IoT device starts HTTPS SlowDos against BMS web service.
- TC6.2 MMT probe detects the issue and notifies it to Incident Detector. (MONT)
- TC6.3 Incident detector-VDSS processes the notification and generates alert and mitigations list. (ATOS)
    - VDSS receives the alert from Incident Detector and generates mitigations list.
- TC6.4 The SAS forwards alert to DSPS. (Alert level between 6 and 8) (CNR)
- TC6.5 DSPS changes to YELLOW or RED according to severity & criticality. (AS/DG/MI)
- TC6.6 VDSS generates a list of recommended mitigations. (ATOS)
- TC6.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS. (MMT)
- TC6.8 The SAS forwards mitigation action (mitigating) to DSPS. (CNR)

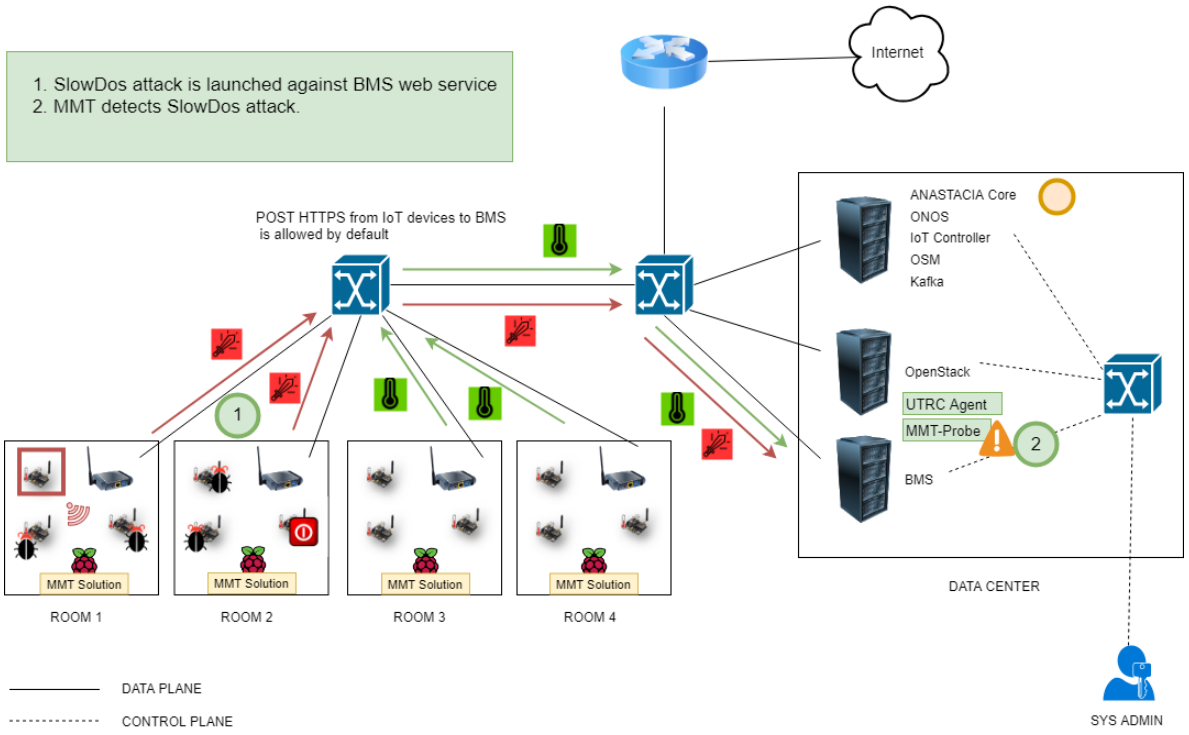


Figure 11: Diagram of Step 6 of Phase R3 Attack in Smart Building

3.1.2.3.1.1 TC6.1 SlowDos attack is launched (CNR)

TC6.1 SlowDos attack is launched: (CNR)	
Execution	• The attacker runs the Slow DoS Attack tool targeting the victim
KPI(s)	• [No KPI for this test case]

There is not KPI in this test-case.

3.1.2.3.1.2 TC6.2 MMT probe detects the issue and notifies it to Incident Detector (MONT)

TC6.2 MMT probe detects the issue and notifies it to Incident Detector. (MONT)	
Execution	• The SlowDoS attack is launched as described in TC6.1.
KPI(s)	1. Validate if the detection is successful.

The next table shows the KPI values measured for the detection of issues by the MMT.

Execution Num.	1	2	3	4	5	Mean
1. True of False	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

The explanations of the KPI values obtained are:

1. MMT probe is able to detect successfully the SlowDoS attack in all occasions.

3.1.2.3.1.3 TC6.3 Incident detector-VDSS processes the notification and generates alert (ATOS)

TC6.3 Incident detector – processes the notification and generates alert (ATOS)	
Execution	• Raw "SlowComm attack detected" events are submitted by the MMT probe to the Data Filtering and pre-processing broker • The storm topology of the Data Filtering and pre-processing broker processes and sends the events through rsyslog to the Incident Detector • The Incident Detector receives the events through its agent, which normalizes them to a common format and submits it to the correlator engine of the Incident Detector, which processes it and generates a "Slow DDoS Attack Detected" alert • The alert is pushed to a the RabbitMQ queue, exchange alarms, to be consumed by SAS and VDSS
KPI(s)	1. Obtain the processing time in the Incident Detector to normalize the event, correlate it, generate the alert and send it to the RabbitMQ exchange

The next table shows the KPI values measured for detecting incidents, process notifications and generating alerts.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,08	0,36	0,09	0,1	0,09	0,144

The explanations of the KPI values obtained are:

1. Those values includes the processing time for the Incident Detector since the notification is received from Kafka Broker until is sent to RabbitMQ. Communication times between Kafka Broker - Incident Detector and Incident Detector - RabbitMQ are also included in the measurements.

3.1.2.3.1.4 TC6.4 The SAS forwards alert to DSPS (CNR)

TC6.4 The SAS forwards alert to DSPS. (Alert level between 6 and 8) (CNR)	
Execution	• SAS consumes the RabbitMQ queue by retrieving the generated alert • The SAS forwards the alert, as it is, to another RabbitMQ server shared with the DSPS
KPI(s)	1. The time passing between the reception of the alert from the RabbitMQ queue dedicated to VDSS-to-SAS communication and the sending of the (same) alert to the RabbitMQ queue dedicated to SAS-to-DSPS communication

The next table shows the KPI values measured for the SAS forwarding alert to DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,07	0,06	0,04	0,06	0,08	0,062

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.3.1.5 TC6.5 DSPS changes to YELLOW or RED according to severity & criticality (AS/DG/MI)

TC6.5 DSPS changes to YELLOW or RED according to security risk (AS/DG/MI)	
Execution	• Alert is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until processed

The next table shows the KPI values measured for the DSPS changing to state YELLOW or RED.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.1.2.3.1.6 TC6.6 VDSS generates a list of recommended mitigations (ATOS)

TC6.6 VDSS generates a list of recommended mitigations (ATOS)	
Execution	• The VDSS receives an alert from the Incident Detector • The VDSS retrieves from the Assets Model the list of available mitigation strategies capable of mitigating the detected alert • The VDSS retrieves from the System Model Service information about the devices affected by the incident • The VDSS calculates suitability scores for every mitigation strategy and pushes it to an exchange recommendations queue, to be consumed by the MAS and the SAS.
KPI(s)	1. Obtain the elapsed time since the VDSS receives the alert until it sends the recommendations to RabbitMQ

The next table shows the KPI values measured for the VDSS to generate a list of recommended mitigations.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,33	0,31	0,36	0,3	0,32	0,324

The explanations of the KPI values obtained are:

1. Those values includes the time of receiving from RabbitMQ the alert generated by the Incident Detector, processing it by getting additional information about the affected asset from the Assets Model and the System Model APIs, calculate the ranking for each of the mitigation strategies and sending the recommendations for the MAS to the RabbitMQ server.

3.1.2.3.1.7 TC6.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)

TC6.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)	
Execution	• An alert is published by the VDSS in the RabbitMQ channel. • The MAS receives the alert and starts the mitigation strategy processing. • The MAS contacts the auxiliary services to retrieve additional data to compute the MSPL (AM and SMS). • The MAS generates the MSPL and sends it to the SO. • The MAS sends the computed mitigation to the SAS and the VDSS.
KPI(s)	1. Obtain the general time elapsed to compute the mitigation: starting from the reception of the alert until the MSPL is sent to the SO.

The next table shows the KPI values measured for the MAS to consume the list of recommendations and generate a final mitigation in MSPL format.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	1,385	1,436	1,409	1,402	1,443	1,415

The explanations of the KPI values obtained are:

1. The MAS contacts external services while computing the reaction, thus this measurement also includes the response time of the System Model and Assets Model.

3.1.2.3.1.8 TC6.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)

TC6.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)	
Execution	• The MAS sends to the SAS the MSPL file, related to the given alert, previously received by the VDSS • The SAS encapsulates received data inside the previously received alert; in particular, the following attributes are added to the original alert • The SAS forwards the enriched alert to the RabbitMQ server, shared with the DSPS
KPI(s)	1. The time passing between the reception of the MSPL data from the MAS and the sending of the enriched alert to the RabbitMQ queue

The next table shows the KPI values measured for the SAS to forward the mitigation actions to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,04	0,04	0,05	0,06	0,03	0,044

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.3.2 Step 7 – R3: Results of Test-Cases with KPIs and Data Captured

The next stage (step 7) shown in Figure 12, validates the reaction module that enforces the isolation of malicious or infected devices. This step validates capacity of the ANASTACIA framework as well, to indicate a change in system state through the DSPS UI and other reaction module components. The sequence involves the following steps:

- TC7.1 Reactive policy for orchestration enforcement (UMU/AALTO)
  - Isolate compromised Rooms (Filtering)
  - Deploy and configure IoT Honeynet
  - Traffic Divert (Mirroring) of external traffic against MMT Probe for traffic inspection
- TC7.2 SO updates system model (AALTO)
- TC7.3 SAS polls SO system model and send to DSPS the alert
- TC7.4 DSPS depends from previous status

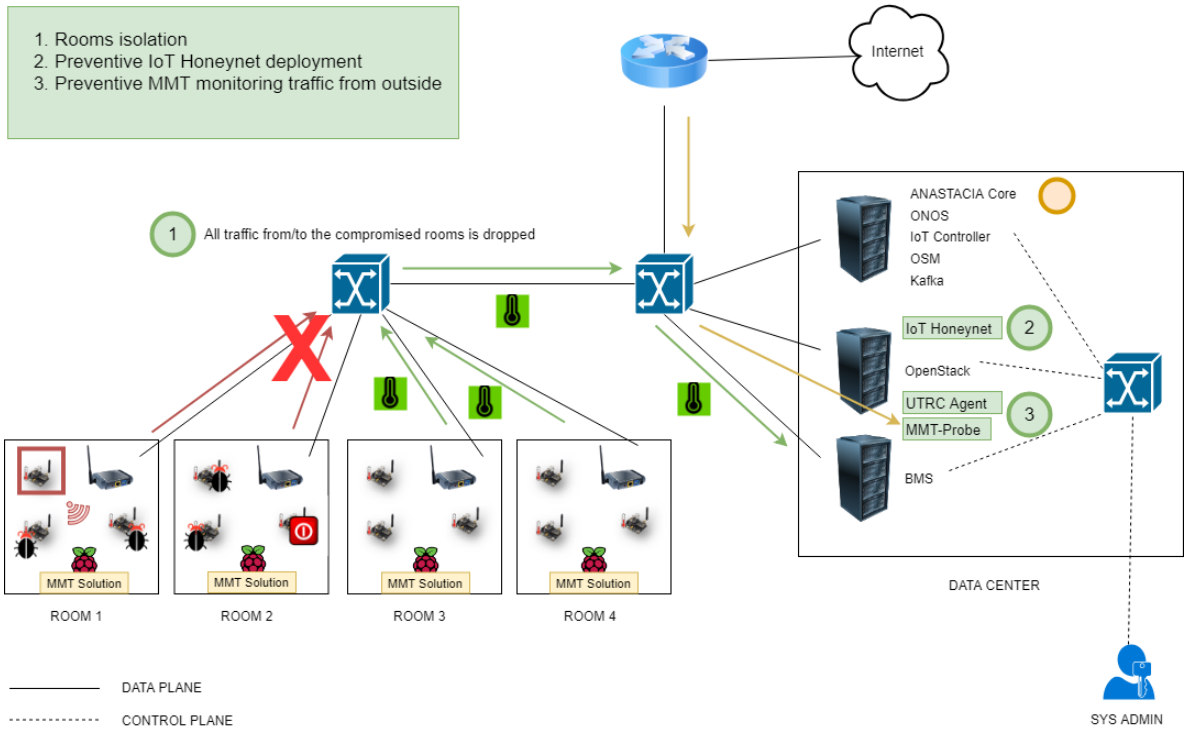


Figure 12: Diagram of the Step 7 for the Phase R3 Attack in Smart Building

3.1.2.3.2.1 TC7.1 Reactive policy for orchestration enforcement (UMU/AALTO)

TC7.1 Reactive policy for orchestration enforcement (UMU/AALTO)	

Execution	• Security Orchestrator verifies MSPL-OP policy which contains:
	  ○ Filtering MSPL
	  ○ IoT Honeynet MSPL
	  ○ Traffic divert (Mirroring)
	• Security Orchestrator communicates with the Security Enablers Provider for selecting the best enabler.
	• Security Orchestrator decides to use: i) OVS-FW enabler for filtering the traffic; ii) IoT Honeynet; iii) ONOS enabler for traffic network monitoring.
	• Security Orchestrator requests policies translation from the Policy Interpreter.
	• Policy Interpreter verifies policy conflicts or dependencies by using the conflict detector.
	• Policy Interpreter retrieves enabler plugins from the Security Enabler Provider and translates the MSPL-OP into final configurations.
	• Policy Interpreter returns final configurations and conflicts and dependencies detection to Security Orchestrator.
	• Security Orchestrator communicates with NFV orchestrator to instantiate the network service instance (NSI) using the NFV-OVS network service descriptor (NSD).
	• NFV orchestrator communicates with OpenStack VIM to instantiate and configure the OVS-FW.
	• Security Orchestrator generates the open flow rules from the received configuration to redirect the traffic to OVS-FW.
	• Security enforces the generated open flow rules through the REST API of ONOS to redirect the traffic to pass through the new instantiated OVS-FW.
	• Security Orchestrator communicates with NFV orchestrator to instantiate the network service instance (NSI) using the IoT Honeynet network service descriptor (NSD).
	• NFV orchestrator communicates with OpenStack VIM to instantiate and configure the IoT Honeynet.

	• Security Orchestrator generates the open flow rules from the received configuration to redirect the traffic to the IoT Honeynet. • Security enforces the generated open flow rules through the REST API of ONOS to redirect the traffic to the IoT Honeynet. • Security Orchestrator deploys a Cooja agent and enforces IoT Honeynet configurations by using the Cooja agent driver.
KPI(s)	1. Policy translation times. 2. Conflict detection times. 3. Cooja IoT Honeynet instantiation times. 4. Cooja IoT Honeynet configuration times. 5. Traffic mirroring and filtering times.

The next table shows the KPI values measured for the enforcement of the reactive policy for orchestration.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,656	0,667	0,6212	0,648	0,617	0,64184
2. Time (seconds)	0,4558	0,4551	0,4585	0,453	0,463	0,45708
3. Time (seconds)	42,26443005	47,30840993	39,26019001	34,0927999	51,44159007	40,73145747
4. Time (seconds)	146,43625	142,0580502	148,9751101	154,29316	142,9788499	147,9406425
5. Time (seconds)	0,5533599854	0,5704700947	0,573319912	0,6091399193	0,5618200302	0,5765724778

The explanations of the KPI values obtained are:

1. Time takes the policy translator to translate reactive filtering, mirroring and IoT-honeynet policies into low-level configurations for IoT Honeynet agent and SDN network.
2. Time takes the conflict detector to detect conflicts and dependencies between the reactive medium-level security policies.
3. Time the security orchestrator takes to instantiate Cooja IoT Honeynet VNF and configure the control plane
4. Time the security orchestrator takes to configure Cooja IoT Honeynet
5. Time the security orchestrator takes to enforce the mirroring and filtering intents on the SDN controller

3.1.2.3.2.2 TC7.2 SO updates system model (AALTO)

TC7.2 SO updates system model (AALTO)	
Execution	• Security Orchestrator sends post request to update the enforcement API status
KPI(s)	1. Measure the request time

The next table shows the KPI values measured for updating the system model.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,1048898697	0,1130301952	0,1039700508	0,111700058	0,1211698055	0,1083975434

The explanations of the KPI values obtained are:

1. Time the security orchestrator takes to update the enforcement API status from pending to success or fail on the System model

3.1.2.3.2.3 TC7.3 SAS polls SO system model and send to DSPS the alert (CNR)

TC7.3 SAS polls SO system model and sends to DSPS the alert	
Execution	• The SAS polls the SM for a specific deployment identifier, previously received by the MAS • The SM answers with deployment data • The SAS extracts relevant fields from the data received by the SM • The SAS enriches (again) the alert to include relevant information received from the SM • The SAS forwards the enriched alert to the DSPS, through the dedicated RabbitMQ queue
KPI(s)	1. The time spent between the reception of the (latest/valid) message from the SM and the sending of the enriched alert to the DSPS

The next table shows the KPI values measured for polling the system model and send it to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,03	0,05	0,02	0,03	0,04	0,034

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.3.2.4 TC7.4 DSPS changes status (AS/DG/MI)

TC7.4 DSPS changes status	
Execution	• Alert with mitigated flag/information is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until processed

The next table shows the KPI values measured for changing the DSPS status.

Execution Num.	1	2	3	4	5	Mean

1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.1.2.4 Phase R4: External attacker tries to compromise IoT devices

The objective of the last phase of the Smart Building use case is to prevent a potential overtake of the IoT network by an external attacker. Here, we will evaluate the capacity of the ANASTACIA framework to protect the IoT network from an external attacker that attempts to gain control over a set of already-compromised IoT devices. As shown in Figure 13, the process unfolds as follows: an external attacker tries to connect to a set of IoT nodes compromised in previous phases, with the objective to escalate the attack and takeover the network. Here we present two sub-steps that the R4 of the Smart Building testbed consists, each in a subsection, which demonstrate the detection, mitigation and enforcement to manage the attack.

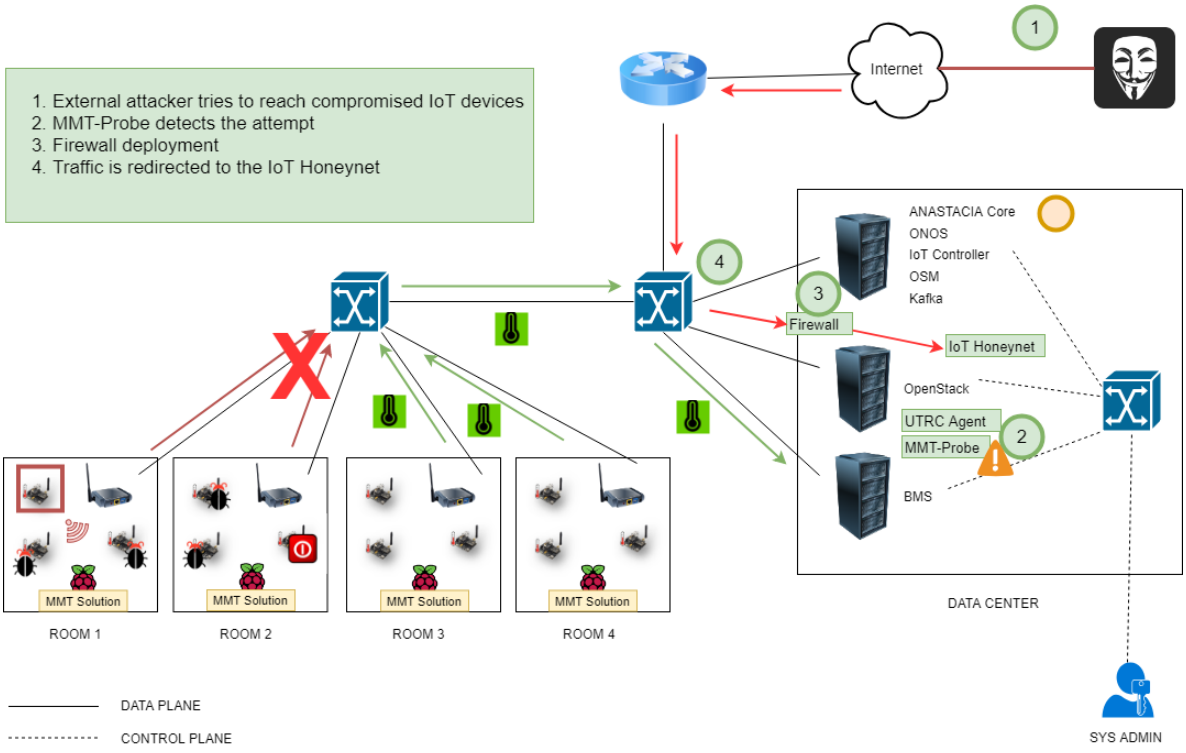


Figure 13: Diagram of the Phase R4 Attack in Smart Building

3.1.2.4.1 Step 8 – R4: Results of Test-Cases with KPIs and Data Captured

Here we describe the test cases for the first of the two tests involving R2. The objectives of this test are: 1) to simulate a connection from an external user (an attacker trying to gain access to take over the compromised IoT devices); 2) detect the attempt using the security components available on the network; 3) generate the associated security alert; 4) create the mitigation plan for the attack and 5) send the mitigation to the Security Alert Service to update the DSPS.

- TC8.1 External attacker tries to reach IoT domain: (ODINS)
  - CoAP requests coming from external IP addresses
- TC8.2 MMT solution detects attacker from external IP address. (MONT)

- TC8.3 Incident detector-VDSS processes the notification and generates alert and mitigations list. (ATOS)
- TC8.4 The SAS forwards alert to DSPS. (CNR)
- TC8.5 DSPS changes to RED. (AS/DG/MI)
- TC8.6 VDSS generates a list of recommended mitigations. (ATOS)
- TC8.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS. (MMT)
- TC8.8 The SAS forwards mitigation action (mitigating) to DSPS. (CNR)
- TC8.9 DSPS maintains RED state. (AS/DG/MI)
- TC8.10 SAS keeps polling SO system model. (CNR)

3.1.2.4.1.1 TC8.1 External attacker tries to reach IoT domain: (ODINS)

TC8.1 External attacker tries to reach IoT domain: (ODINS)	
Execution	• Malicious Script is launched. • Data traffic from an external IP address is trying to reach a IoT device.
KPI(s)	1. Obtain the elapsed time for generating the external data traffic of the received query and modified the IP destination address in the internal configuration.

The next table shows the KPI values measured for an external attacker trying to reach IoT devices.

Execution Num.	1	2	3	4	5	Mean
1. True or False	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE

The explanations of the KPI values obtained are:

1. Determine if an external attacker is able to reach an IoT device using different protocols (ICMP, etc)

3.1.2.4.1.2 TC8.2 MMT solution detects attacker from external IP address. (MONT)

TC8.2 MMT solution detects attacker from external IP address. (MONT)	
Execution	• The external connection is launched as described in T8.1.
KPI(s)	1. The external connection has been correctly detected by the MMT-Probe

The next table shows the KPI values measured for detecting the attempt to connect form unrecognized IP address.

Execution Num.	1	2	3	4	5	Mean
1. True of False	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

The explanations of the KPI values obtained are:

1. MMT probe is able to detect successfully the external IP address of the attacker in all occasions.

3.1.2.4.1.3 TC8.3 Incident detector-VDSS processes the notification and generates alert and mitigations list (ATOS)

TC8.3 Incident detector – processes the notification and generates alert (ATOS)	
Execution	• Raw "SQL injection" events are submitted by the MMT probe to the Data Filtering and pre-processing broker • The storm topology of the Data Filtering and pre-processing broker processes and sends the events through rsyslog to the Incident Detector • The Incident Detector receives the events through its agent, which normalizes them to a common format and submits it to the correlator engine of the Incident Detector, which processes it and generates a "SQL Injection Detected" alert • The alert is pushed to a the RabbitMQ queue, exchange alarms, to be consumed by DSPS and VDSS
KPI(s)	1. Obtain the processing time in the Incident Detector to normalize the event, correlate it, generate the alert and send it to the RabbitMQ exchange

The next table shows the KPI values measured for the processing, notification generation of the alert.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,07	0,1	0,21	0,07	0,08	0,106

The explanations of the KPI values obtained are:

1. Those values includes the processing time for the Incident Detector since the notification is received from Kafka Broker until it is sent to RabbitMQ. Communication times between Kafka Broker - Incident Detector and Incident Detector - RabbitMQ are also included in the measurements.

3.1.2.4.1.4 TC8.4 The SAS forwards alert to DSPS (CNR)

TC8.4 The SAS forwards alert to DSPS (CNR)	
Execution	• SAS consumes the RabbitMQ queue by retrieving the generated alert • The SAS forwards the alert, as it is, to another RabbitMQ server shared with the DSPS
KPI(s)	1. The time passing between the reception of the alert from the RabbitMQ queue dedicated to VDSS-to-SAS communication and the sending of the (same) alert to the RabbitMQ queue dedicated to SAS-to-DSPS communication

The next table shows the KPI values measured for the SAS to forward the alert to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,03	0,02	0,03	0,03	0,04	0,03

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.4.1.5 TC8.5 DSPS changes to RED (AS/DG/MI)

TC8.5 DSPS changes to RED (AS/DG/MI)	
Execution	• Security alert is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until stored in DSPS secured storage 3. The time measurement since DSPS seal creation receives alert until DSPS GUI is notified of the new seal

The next table shows the KPI values measured for the DSPS changing the state to RED.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145
3. Time (seconds)	0,017	0,023	0,021	0,032	0,019	0,0224

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.1.2.4.1.6 TC8.6 VDSS generates a list of recommended mitigations (ATOS)

TC8.6 VDSS generates a list of recommended mitigations (ATOS)	
Execution	• The VDSS receives an alert from the Incident Detector • The VDSS retrieves from the Assets Model the list of available mitigation strategies capable of mitigating the detected alert • The VDSS retrieves from the System Model Service information about the devices affected by the incident • The VDSS calculates suitability scores for every mitigation strategy and pushes it to an exchange recommendations queue, to be consumed by the MAS and the SAS.
KPI(s)	1. Obtain the elapsed time since the VDSS receives the alert until it sends the recommendations to RabbitMQ

The next table shows the KPI values measured for the VDSS generating a list of recommended mitigations.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,34	0,29	0,35	0,27	0,38	0,326

The explanations of the KPI values obtained are:

1. Those values includes the time of receiving from RabbitMQ the alert generated by the Incident Detector, processing it by getting additional information about the affected asset from the Assets Model and the System Model APIs, calculate the ranking for each of the mitigation strategies and sending the recommendations for the MAS to the RabbitMQ server.

3.1.2.4.1.7 TC8.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)

TC8.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)	
Execution	• An alert is published by the VDSS in the RabbitMQ channel. • The MAS receives the alert and starts the mitigation strategy processing. • The MAS contacts the auxiliary services to retrieve additional data to compute the MSPL (AM and SMS). • The MAS generates the MSPL and sends it to the SO. • The MAS sends the computed mitigation to the SAS and the VDSS.
KPI(s)	1. Obtain the general time elapsed to compute the mitigation: starting from the reception of the alert until the MSPL is sent to the SO.

The next table shows the KPI values measured for the MAS consuming the list of recommendations and generating the mitigation in MSPL form.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	1,185	0,655	0,664	0,665	0,674	0,769

The explanations of the KPI values obtained are:

1. The MAS contacts external services while computing the reaction, thus this measurement also includes the response time of the System Model and Assets Model.

3.1.2.4.1.8 TC8.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)

TC8.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)	
Execution	• The MAS sends to the SAS the MSPL file, related to the given alert, previously received by the VDSS • The SAS encapsulates received data inside the previously received alert; in particular, the following attributes are added to the original alert • The SAS forwards the enriched alert to the RabbitMQ server, shared with the DSPS
KPI(s)	1. The time passing between the reception of the MSPL data from the MAS and the

	sending of the enriched alert to the RabbitMQ queue

The next table shows the KPI values measured for forwarding the mitigation actions to the DSPS.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,04	0,05	0,06	0,03	0,04	0,044

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.4.1.9 TC8.9 DSPS maintains RED state (AS/DG/MI)

TC8.9 DSPS status remains RED (AS/DG/MI)	
Execution	• Alert with mitigation action is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until processed

The next table shows the KPI values measured for the DPSP status to remain RED.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.1.2.4.1.10 TC8.10 SAS keeps polling SO system model (CNR)

TC8.10 SAS keeps polling SO system model(CNR)	
Execution	• The SAS polls the SM for a specific deployment identifier, previously received by the MAS
KPI(s)	1. Measure the time passing between the reception of the message from the MAS and the first polling to the SM

The next table shows the KPI values measured for the SAS to poll the system model.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,03	0,02	0,03	0,03	0,04	0,03

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.1.2.4.2 Step 9 – R4: Results of Test-Cases with KPIs and Data Captured

This is the last test case of the current phase. Here we demonstrate how the ANASTACIA framework deploys a reactive policy through the SO, redirecting the traffic to the Honeynet, changing the state of the security dynamically to green status. This is a consequence of the successful deployment of the ANASTACIA security policies into SEP. The following sequence describes the testbed:

- TC9.1 Reactive policy for orchestration enforcement (AALTO)
  - Redirect traffic to the honey network.
- TC9.2 DSPS will change to GREEN after the security and privacy analysis (AS/DG/MAND)

3.1.2.4.2.1 TC9.1 Reactive policy for orchestration enforcement (UMU/AALTO)

TC9.1 Reactive policy for orchestration enforcement (AALTO)	
Execution	• Security Orchestrator verifies MSPL-OP policy which contains: ○ Traffic Divert (Forward) MSPL • Security Orchestrator communicates with the Security Enablers Provider for selecting the best enabler. • Security Orchestrator decides to use OVS-FW enabler for filtering the traffic and ONOS enabler for redirecting the traffic. • Security Orchestrator requests policies translation from the Policy Interpreter. • Policy Interpreter verifies policy conflicts or dependencies by using the conflict detector. • Policy Interpreter retrieves enabler plugins from the Security Enabler Provider and translates the MSPL-OP into final configurations. • Policy Interpreter returns final configurations and conflicts and dependencies detection to Security Orchestrator. • Security Orchestrator communicates with NFV orchestrator to instantiate the network service instance (NSI) using the NFV-OVS network service descriptor (NSD). • NFV orchestrator communicates with OpenStack VIM to instantiate and configure the OVS-FW. • Security Orchestrator generates the open flow rules from the received configuration to redirect the traffic to OVS-FW. • Security enforces the generated open flow rules through the REST API of ONOS to redirect the traffic to pass through the new instantiated OVS-FW.

KPI(s)	1. Policy translation times. 2. Conflict detection times. 3. OVS-FW instantiation times. 4. OVS-FW configuration times. 5. Traffic mirroring and filtering times.

The next table shows the KPI values measured for the enforcement of the reactive policy orchestration.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,217	0,178	0,186	0,165	0,172	0,1836
2. Time (seconds)	0,228	0,226	0,232	0,225	0,222	0,2266
3. Time (seconds)	22,44802999	28,50819993	28,43665004	33,62998009	33,48498988	28,25571501
4. Time (seconds)	294,46437	294,56407	290,39695	290,36498	289,8972499	292,4475925
5. Time (seconds)	0,5533599854	0,5704700947	0,593269825	0,5991399288	0,562019825	0,5790599585

The explanations of the KPI values obtained are:

1. Time taken by the policy translator to translate reactive forwarding and monitoring policies into low-level configurations for SDN network.
2. Time taken by the conflict detector to detect conflicts and dependencies between the reactive medium-level security policies.
3. Time taken by the security orchestrator to instantiate OVS-FW VNF and configure the control plane.
4. Time taken by the security orchestrator to configure OVS-FW rules using SSH
5. Time taken by the security orchestrator to enforce the mirroring and filtering intents on the SDN controller

3.1.2.4.2.2 TC9.2 DSPS will change to GREEN after the security and privacy analysis (AS/DG/MI)

TC9.2 DSPS will change to GREEN after the security and privacy analysis	
Execution	• Alert with mitigated flag/information is received • The message format of the alert is converted using STIX • Alert is processed by DSPS Creation Service • Privacy risks are mapped from security alert • New seal is created with updated security and privacy risk levels • New seal is stored • GUI is notified about new seal • CISO/DPO restores seal for security and privacy
KPI(s)	1. The time measurement since AMQP message is consumed from queue until posted to DSPS seal creation service 2. The time measurement since DSPS seal creation receives alert until stored in DSPS secured storage 3. The time measurement since DSPS seal creation receives alert until DSPS GUI is notified of the new seal

The next table shows the KPI values measured for changing the state to GREEN by the DPSP after the analysis.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	<5ms	<5ms	<5ms	<5ms	<5ms	<5ms
2. Time (seconds)	4,134	4,356	4,216	4,012	4,007	4,145
3. Time (seconds)	0,017	0,023	0,021	0,032	0,019	0,0224

The explanations of the KPI values obtained are the same as TC2.5 (see 3.1.2.1.2.5).

3.2 USE-CASE: MOBILE EDGE COMPUTING IN 5G NETWORK

3.2.1 Description of Use Case and Benefits from ANASTACIA Framework

This use case of Mobile Edge Computing (MEC) was implemented to showcase the application of ANASTACIA framework with new mobile network technologies like 5G. An innovative characteristic of 5G technology is the incorporation of network slicing. Network slicing enables to operate virtual mobile networks over the physical infrastructure. Network slicing is implemented by means of network function virtualization (NFV) and software defined network (SDN). For each slide, SDN allows the configuration of a particular traffic type and VNF enables the customized functions for a specific use case. Network slices are isolated in terms of traffic and resources (CPU, bandwidth, storage, etc.). Therefore, the reduction of resources in one slice does not affect other slices. Regarding the cybersecurity, this is an essential feature. So, a security threat in one slice is avoided from extending to other slices. The resources isolation is a key property for avoiding particular attacks (i.e. DDoS) between slices. In the evaluation of ANASTACIA project, specific slices are proposed to particular applications, regions, device types and even to individual devices. Concretely, we propose the use of quarantine slices reserved for devices with reduced trust, e.g. as a result of suspicious behavior observed by a monitoring system.

To evaluate the MEC use case in 5G network, we leverage the same deployment of the ANASTACIA framework in the UMU premises described in section 3.1.1. Additionally, Ericsson provided the attacker and victim involved in the use case of 5G network as well as the 5G core infrastructure. The implementation of the MEC use case consists of the simulation of a denial-of-service (DoS) attacker over a 5G video producer. In particular, Figure 14 shows the MEC use case implemented with a 5G network slicing demonstration managed by ANASTACIA framework. The use case is divided into the ANASTACIA components deployed in the UMU premises on the bottom and the 5G core network part located in the Ericsson's premises on the right top. For the evaluation, we deployed locally in Murcia a 5G Base Station Emulator with two client devices connected: one legitimate video producer and a DoS attacker. The 5G Base State Emulator is a software implementation of the central base station executing in an Intel NUC with the radio network developed in Wi-Fi. The client devices are both developed with RaspberryPis and connect to the Base Station Emulator via Wi-Fi. One of the client devices is infected with malware script causing the massive traffic transmission toward the video producer. To monitor the traffic and detect the attack, an ANASTACIA component called MMT Probe was installed in the 5G Base Station Emulator.

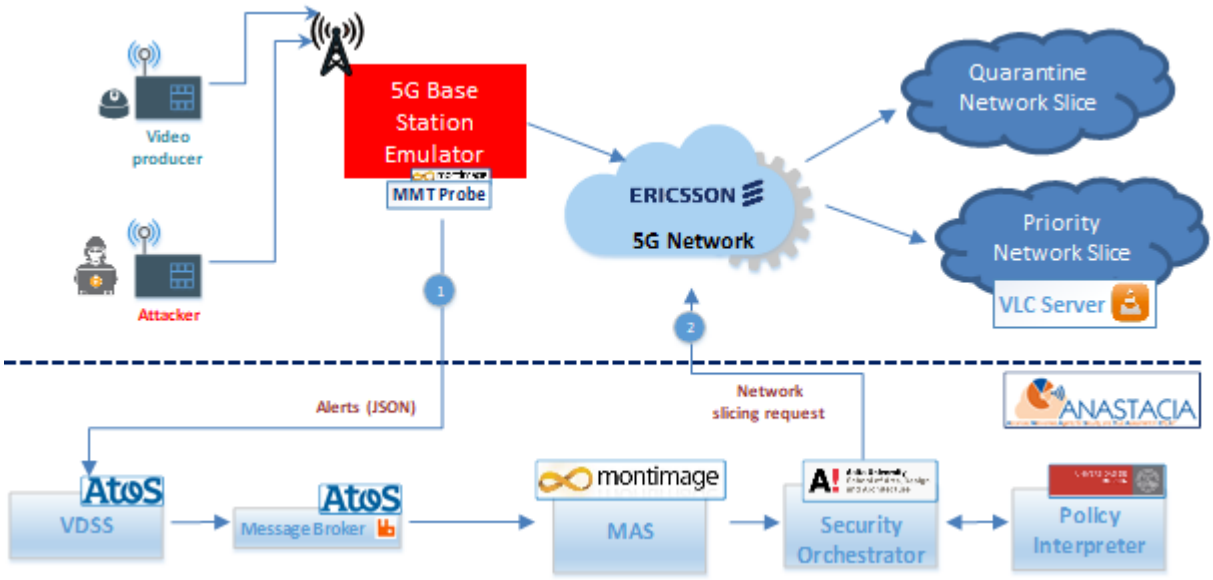


Figure 14: Use case implementation of MEC in 5G scenario

The 5G infrastructure with the core network is running in a remote data center provided by Ericsson. The 5G network supports multiple network slices orchestrated via an orchestration platform including a Multi-domain Slice Orchestrator, an NFV Orchestrator (NFVO), a Network Slice Selection Function (NSSF), an Overlay Network Controller and a SDN controller. These components were developed outside the ANASTACIA project but have been modified with functionality to support security operations. The new features include the possibility to deny access to a given slice and to transfer client devices between network slices.

In a normal behaviour, the client devices are connected to a network slices dedicated to camera streaming traffic. This slice contains a streaming server implemented using the VLC software that allows video receivers connect with video transmitters and performs adaptation of the video format. There is a pre-existing definition of a quarantine slice with strict limitations in the available bandwidth. Once the exceptional behavior in the attacking device has been detected, the device is blocked from the normal slice. Using ANASTACIA framework, we might prefer to isolate these devices through quarantine slices rather than completely blocking or disconnecting device especially in the cases where the malicious behavior is unconfirmed or as a first reaction. This prevents critical services and breaking potential service agreements.

In the MEC use case of 5G network, the main benefit provided by ANASTACIA framework is to detect the non-legitimate traffic from malicious devices and isolate them using the 5G network slicing. To achieve that, ANASTACIA includes innovative components such as holistic policy-based interpreter and efficient security orchestrator, novel monitoring probes and reaction tools (i.e. VDSS). In this use case, the key benefits provided by ANASTACIA framework are as follows.

1. Novel monitoring agents that can be deployed dynamically to detect non-legitimate 5G traffic.
2. Automated reaction to decide the best mitigation strategies according to the attacks detected.
3. Holistic policy-based security management to establish and interpret the mitigation strategies in specific enforcement actions with conflict detection and policy dependencies.
4. Efficient security orchestration to generate the enforcement actions in a 5G infrastructure with network slicing.

For the use case of MEC in 5G networks, Annexe 2 provides the photos and interfaces captured during the demo sessions performed in the University of Murcia during the days 11-12 December 2019.

3.2.2 Validation based on Test-Cases and Measured KPIs

This section reports the measured KPIs of the validation executions of the 5G use case of Mobile Edge Computing according to the test cases defined in the deliverable D6.5. Below, the tables show the results of test-cases and also the KPI measurements in seconds for 5 executions of the performance tests.

3.2.2.1 DoS attack to 5G camera

This specific use case is selected to demonstrate how ANASTACIA framework is able to mitigate cyber-attacks in a 5G infrastructure using the network slicing as isolation countermeasure. This use case shows the malicious exploitation of a 5G connected device to generate a DoS attack sending traffic to overload a 5G video producer. The aim of this use case is to detect the abnormal traffic from the malicious device and to isolate it using network slicing. For this simulated attack, below we provide the results of the validated test-cases and measured KPIs to evaluate the 5G scenario and the operations of ANASTACIA components.

3.2.2.1.1 Results of Test-Cases with KPIs and Data Captured

This subsection shows the measured KPIs of the validation executions of the 5G scenario according to the test cases defined in the deliverable D6.5. Below, the tables show the results of test-cases and also the KPI measurements in seconds for 5 executions of the performance tests. These test-cases involve the operations of ANASTACIA framework for monitoring, reaction, seal management, policy interpretation and orchestration. The following list indicates all actions performed by ANASTACIA components and the partner leaders:

- TC10.1 A malicious device starts sending excessive traffic to an 5G camera target. (ER)
- TC10.2 MMT probe detects excessive traffic in the 5G network from the attacker. (MONT)
- TC10.3 Incident detector – processes the notification and generates alert. (ATOS)
- TC10.4 The SAS forwards alert to DSPS. (CNR)
- TC10.5 DSPS changes to RED. (AS/DG/MI)
- TC10.6 VDSS generates a list of recommended mitigations. (ATOS)
- TC10.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS. (MMT)
- TC10.8 The SAS forwards mitigation action (mitigating) to DSPS. (CNR)
- TC10.9 DSPS maintains RED state. (AS/DG/MI)
- TC10.10 SAS keeps polling SO system model. (CNR)
- TC10.11 Reactive policy for orchestration enforcement. (UMU/AALTO)
- TC10.12 DSPS will change to GREEN after the security and privacy analysis. (AS/DG/MI)

3.2.2.1.1.1 TC10.1 IoT device starts sending excessive traffic to an outside attack target (ER)

TC10.1 IoT device starts sending excessive traffic to an outside attack target. (ER)	
Execution	• IoT Device starts a new HTTP connection to an IP destination address external to the network in order to send a bandwidth intensive stream of data
KPI(s)	None

There is KPIs in this use case to launch the attack.

3.2.2.1.1.2 TC10.2 MMT solution detects excessive traffic to unknown destinations (MONT)

TC10.2 TC10.2 MMT solution detects excessive traffic to unknown destinations. (MONT)	
Execution	• The traffic to the outside target is started as described in TC10.1
KPI(s)	1. Validate if the attack is correctly detected

The next table shows the KPI values measured for X.

Execution Num.	**1**	**2**	**3**	**4**	**5**	**Mean**
1. Detected	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

The explanations of the KPI values obtained are:

1. MMT probe is able to detect successfully the massive traffic from the attacker to the target in all occasions.

3.2.2.1.1.3 TC10.3 Incident detector-VDSS processes the notification and generates alert and mitigations list (ATOS)

Please refer to same test case as in TC8.3.

3.2.2.1.1.4 TC10.4 The SAS forwards alert to DSPS (CNR)

Please refer to same test case as in TC8.4.

3.2.2.1.1.5 TC10.5 DSPS changes to RED (AS/DG/MI)

Please refer to same test case as in TC8.5.

3.2.2.1.1.6 TC10.6 VDSS generates a list of recommended mitigations (ATOS)

Please refer to same test case as in TC8.6.

3.2.2.1.1.7 TC10.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)

TC10.7 The MAS consumes list of recommendations and generates final mitigation in MSPL format and sends to SO, SAS (MMT)	
Execution	• An alert is published by the VDSS in the RabbitMQ channel. • The MAS receives the alert and starts the mitigation strategy processing. • The MAS contacts the auxiliary services to retrieve additional data to compute the MSPL (AM and SMS). • The MAS generates the MSPL and sends it to the SO. • The MAS sends the computed mitigation to the SAS and the VDSS.
KPI(s)	1. Obtain the general time elapsed to compute the mitigation: starting from the reception of the alert until the MSPL is sent to the SO.

The next table shows the KPI values measured for X.

Execution Num.	**1**	**2**	**3**	**4**	**5**	**Mean**
1. Time (seconds)	0,397	0,433	0,398	0,392	0,395	0,403

The explanations of the KPI values obtained are:

1. The MAS contacts external services while computing the reaction, thus this measurwement also includes the response time of the System Model and Assets Model.

3.2.2.1.1.8 TC10.8 The SAS forwards mitigation action (mitigating) to DSPS (CNR)

Please refer to same test case as in TC8.8.

3.2.2.1.1.9 TC10.9 DSPS maintains RED state (AS/DG/MI)

Please refer to same test case as in TC8.9.

3.2.2.1.1.10 TC10.10 SAS keeps polling SO system model (CNR)

TC10.10 SAS keeps polling SO system model(CNR)	
Execution	• The SAS polls the SM for a specific deployment identifier, previously received by the MAS
KPI(s)	1. Measure the time passing between the reception of the message from the MAS and the first polling to the SM

The next table shows the KPI values measured for X.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,06	0,07	0,03	0,05	0,04	0,05

The explanations of the KPI values obtained are:

1. The values include the processing time of the SAS component, measured by the component itself.

3.2.2.1.1.11 TC10.11 Reactive policy for orchestration enforcement (UMU/AALTO/ER)

TC10.11 Reactive policy for orchestration enforcement (AALTO)	
Execution	• Security Orchestrator verifies MSPL-OP policy which contains: ○ Network Slicing Control MSPL • Security Orchestrator requests policies translation to policy Interpreter • Policy Interpreter verifies policy conflicts or dependencies by using the conflict detector. • Policy Interpreter retrieves enabler plugins from the Security Enabler Provider and translates the MSPL-OP into final configurations. • Policy Interpreter returns final configurations and conflicts and dependencies detection to Security Orchestrator. • Security Orchestrator sends request to 5G Security Enabler to deploy and enforce mitigation action recommended by policy orchestrator • 5G Security Enabler receives request from Security Orchestrator to restrict slice access • 5G Security Enabler communicates with AMF in 5G network which enforces the restriction
KPI(s)	1. Policy translation times. 2. Conflict detection times.

3. Time to restrict device from current slice

The next table shows the KPI values measured for X.

Execution Num.	1	2	3	4	5	Mean
1. Time (seconds)	0,149	0,123	0,14	0,113	0,116	0,1282
2. Time (seconds)	0,0179	0,0321	0,0185	0,0255	0,018	0,0224
3. Time (seconds)	10,05721998	10,02243996	10,02435994	10,02598	10,04304004	10,03249997

The explanations of the KPI values obtained are:

1. Time takes the policy translator to translate reactive network slicing policies into low-level configurations for SDN network.
2. Time takes the conflict detector to detect conflicts and dependencies between the reactive medium-level security policies.
3. Time it takes to perform the necessary operations in the 5G core to create the restrictions for the device to the current slice.

3.2.2.1.1.12 TC10.12 DSPS will change to GREEN after the security and privacy analysis (AS/DG/MI)

Please refer to same test case as in TC9.2.

3.3 RESULTS OF END-USER EVALUATION

This chapter shows the results achieved with the end-users feedback during the ANASTACIA evaluation regarding the developed use cases.

3.3.1 Final Evaluation Questionnaire

This section provides generic user questionnaire that will be used to evaluate the feedback provided by end-users. This end-user questionnaire allows evaluating the second release of the integrated ANASTACIA framework regarding the validated use cases of **Building Management System** (BMS) and **Mobile Edge Computing in 5G Network** (MEC). The second release integrates monitoring, detection, reaction, orchestration and enforcement. After the validation of use cases, the end-users evaluate the results and provide their feedback. This document provides generic user questionnaire for each use case to validate the outcomes.

This questionnaire contains general questions that have been defined in D6.5 in accordance to the end-user requirements indicated in D1.2. Table 1 represents generic questions to be answered in the range from 1 fully disagree to 5 fully agree.

Table 1. End-user questionnaire

Number	Question	Assessment level (1-5)
1	Is the ANASTACIA framework **useful to monitor and detect cyber-attacks?**	
2	Does the ANASTACIA framework provide **valuable automatic reactions to mitigate cyber-attacks**?	
3	Does the ANASTACIA framework have **intuitive user interfaces for managing security policies and operations**?	
4	Does the ANASTACIA framework provide **clear seal and report about the cybersecurity status?**	
5	Would you like to have ANASTACIA framework **to protect your cybersecurity and privacy in your workspace or home**?	

Moreover, two open questions are available to collect further information from ANASTACIA users. Questions are listed in Table 2 below.

Table 2. End-user questionnaire – open questions

Number	Open question
1	What do you like the most about the ANASTACIA framework?
2	What do you dislike the most about the ANASTACIA framework?

3.3.2 Results of End-Users Questionnaire

Interviewees were asked to rate relevant features of ANASTACIA framework related to the end-users requirements specified in D1.4. End-users rated each question from 1 to 5 in accordance to a Liker scale:

1. = **Very Low** – **fully disagree**,
2. = **Low** – **partially disagree**,
3. = **Medium** – **neutral**,
4. = **High** – **partially agree**,
5. = **Very High** – **fully agree**.

The next table shows the results of general questions regarding the end-users feedback that are quite satisfied about all main features of ANASTACIA framework. In particular, the intuitive user interfaces for managing security policies and operations were the most appreciated by the end-users.

Results of General Questions about ANASTACIA framework	MEAN	DEVIATION
1. useful to monitor and detect cyber-attacks	4,196550699	0,637334
2. valuable automatic reactions to mitigate cyber-attacks	4,339280404	0,733677
3. intuitive user interfaces for managing security policies and operations	4,589057389	0,53186
4. clear seal and report about the cybersecurity status	4,267320844	0,750609
5. protecting your cybersecurity and privacy in your workspace or home	4,106749926	0,952096

3.3.2.1 Main highlights

Main valuable highlights from the collected questionnaires are reported here according to the most appreciated aspects and the most unsatisfied ones.

3.3.2.1.1 Positive comments and observations (PCO)

	What do you like the most about the ANASTACIA framework?
PCO-1	Fast mitigation of attacks
PCO-2	The different types to specify the security policies
PCO-3	Cyber-attacks mitigation in real-time
PCO-4	I like his huge potential for controlling, monitoring and mitigating cybersecurity attacks
PCO-5	At first sight all appears well
PCO-6	The automatic process of the security
PCO-7	It provides the means to automatize security in IoT networks, reducing the risk of human error that is usually the main cause of security problems.

	What do you like the most about the ANASTACIA framework?
PCO-8	The thing I like the most is the reaction time for detect the attacker and the quick mitigation
PCO-9	The dynamic detection of threats in IoT systems and the policy based security model.
PCO-10	SEAL Management
PCO-11	Flexibility to setup rules and behaviour patterns. Necessary framework to face an iot Project
PCO-12	Automatic reactions to menaces and security risks
PCO-13	Report and Audit system to "vaccinate" the system against attacks carried out in other deployments.
PCO-14	High responsive with effective results. Low latency.
PCO-15	The ability to react in a dynamic way against a lot of attacks.
PCO-16	Simplicity and extensibility
PCO-17	The interfaces are intuitive, simplicity, reactivity
PCO-18	Good and interesting ideas.
PCO-19	The effort to bring together a set of bundled capabilities to ensure security on highly populated IoT environment
PCO-20	The ONOS orchestrator
PCO-21	The capability of learning new security attacks
PCO-22	Continuous monitoring while reacting on alerts and events
PCO-23	De dynamic generation of mitigation activities
PCO-24	Orchestration was working perfectly. Integration tasks have been completed perfectly. The coordination between different WPs. The main idea of ANASTACIA project.
PCO-25	Full process to detect attacks, including the detection and the mitigation. Detailed report about the identified alerts. I like the fact that the process behind is quite sophisticated, with several components communication between them, but the interface stays clear for the user.
PCO-26	Having the ability to automatically mitigate attacks in behalf of the operator. Spotting the rogue devices easily. Having some clues about privacy leaks and what kind of data has been stolen on a specific time frame.
PCO-27	ANASTACIA is able to take autonomous decisions through the use of new networking technologies such as Software Defined Networking (SDN) and Network Function Virtualisation (NFV).
PCO-28	It combines security and privacy standards and real time monitoring and online testing (to provide quantitative and qualitative run-time evaluation of privacy risks and security levels).
PCO-29	The GUI interface which seems very intuitive.
PCO-30	Its holistic approach to security management and cyber-attack mitigation.
PCO-31	This framework succeeds to orchestrate network devices in order to respond to security threat. The technology used (Openstack, ONOS, OSM…) are widely deployed by industrials.
PCO-32	The framework is able to detect advanced attacks like SlowDoS which is very relevant in IoT environments. The framework deals with privacy issues and is able to detect and react against data exfiltration

	What do you like the most about the ANASTACIA framework?
PCO-33	It is useful to monitor and detect cyber-attacks
PCO-34	Monitoring & detecting of attacks
PCO-35	The monitoring and detecting of cyber-attacks
PCO-36	The framework deploys both proactive and reactive security controls which are very interesting to harden protection of IoT environments. The framework deals both with real and virtualized worlds
PCO-37	Information on privacy and security in the plan
PCO-38	The seal management
PCO-39	Automated reaction
PCO-40	ANASTACIA focuses not only to the detection of an offensive event, but also triggers mitigation actions. Detection of SlowDoS is hard to achieve mainly due to the small fingerprint of the attack. This is a valuable contribution of ANASTACIA project. All the events were illustrated properly and in an informative manner through the user interfaces.
PCO-41	Proactive/Reactive security policies
PCO-42	Presentation of the Information
PCO-43	It is simple and easy to understand
PCO-44	The ability to track the privacy and security in near real time
PCO-45	Behaviour detection engine with machine learning
PCO-46	Attack Identification
PCO-47	The modern approach used for detection of anomalies
PCO-48	DSPS Interface Design
PCO-49	The framework is able to detect attacks and mitigate them in IoT environments. The framework is easy to use and allows hardening security in complex environments.
PCO-50	The ability of the framework of detecting a wide variety of attacks, for instance those exploiting “slow” mechanisms. In fact, IoT ecosystems are often characterized by resource scarcity, thus attacks are expected to have a limited fingerprint (e.g., in terms of network traffic). Besides, ANASTACIA also offers a method to collect/report forensics information and this is a plus to manage and protect real IoT/smart environments.

3.3.2.1.2 Negative comments and observations (NCO)

	What do you dislike the most about the ANASTACIA framework?
NCO-1	Maybe the User Interface
NCO-2	The sound of the fire alarm
NCO-3	Nothing
NCO-4	No one

	What do you dislike the most about the ANASTACIA framework?
NCO-5	The complexity of the policies need to react
NCO-6	There are very few things to dislike but I think it would be the amount of technologies involved.
NCO-7	Some interface windows were too simple
NCO-8	Reaction time
NCO-9	Protection of isolated devices outside of a LAN
NCO-10	Needs descriptions on policy lists
NCO-11	Not in production
NCO-12	The probability of not continuing the project
NCO-13	The complexity of the security reports
NCO-14	New IoT controller
NCO-15	Detection algorithms are not sufficiently put forward for zero-day attacks
NCO-16	It requires some extra devices to be deployed. Maybe it does not make sense in small installations.
NCO-17	I'm missing to see more the holistic approach, because the demo is too focused on the IoT-sided, and not on the cloud-backend.
NCO-18	Using the term "5G network slice" in the interface which can raise further ambiguities about the final architecture of ANASTACIA and the main concept of network slices for 5G.
NCO-19	Intuitive interface for the framework, but may need some time to fully understand its features. Maybe a tutorial on the platform would be nice. The framework could include a "protection" part before the detection, for example proposing some main security updates to avoid some attacks to happen.
NCO-20	Too many graphical interfaces. It makes the monitoring and tracking of attacks too fuzzy. The framework could perhaps be more strict / more reactive by being able to block attacks as soon as they start, without triggering any alarms. Missing the opportunity to trap rogue devices into isolated honeypots.
NCO-21	The complexity of ANASTACIA architecture.
NCO-22	The fact that no demo is available for remote execution.
NCO-23	Rebooting a compromised IoT device might not be enough if the malicious code is embedded deep in the system. There should be a way to remotely flash the ROM to a known good state to ensure full protection.
NCO-24	I wonder how this framework could react to a control plane attack. What is the cost of enforcing security rules in OpenFlow switches? TCAM memory is limited and applying fine-grained security rules might affect switches performance.
NCO-25	Mitigation against attacks can extended to achieve an IoT environment that is attack tolerant/ resilient to attacks. Solutions like diversification and reflection strategies should be studied.
NCO-26	Privacy aspects of ANASTACIA
NCO-27	The configuration of the framework is not an easy task. A clear documentation is necessary to

	What do you dislike the most about the ANASTACIA framework?
	exploit it. - The user interface can be improved by providing a map-like dashboard showing the monitored environment
NCO-28	A suggestion to consider more attacks in the context of project.
NCO-29	Alert/notification mechanism
NCO-30	Some functions are difficult to use if we don`t have good knowledge in computing
NCO-31	The effect of the security measure tested against production environments
NCO-32	It should improve privacy aspects
NCO-33	It is not clear what kind of attacks can be detected and what protocols are supported
NCO-34	The framework appears to have an overhead in terms of management for extending its capability of detecting new attacks, e.g., new probes and metrics are needed for each type of threat.

4 CONCLUSIONS

In conclusion, this report provides the results for the validation and end-users evaluation of ANASTACIA framework according to the implemented use cases (Smart Building and MEC in 5G networks).

For each use case, the document describes the implemented testbed and the benefits achieved by ANASTACIA framework. The use case validation is reported using visual information (i.e. web user interfaces) captured during the demo sessions as well as the results of developed test-cases and measured KPIs following the methodology defined in D6.5.

The KPIs measurements show the good performance of the ANASTACIA components from the attack detection to the mitigation actions along the modular architecture for monitoring, reaction, orchestration, seal management and enforcement. In most of test-cases, the response time is less than 3 seconds.

For the end-users evaluation, we performed face-to-face workshops to show the ANASTACIA framework execution within the developed use cases in real-time demo sessions or using recorded demo videos. This deliverable includes the questionnaire results and the feedback provided by 59 end-users who participated in the ANASTACIA evaluation.

The evaluation results of general questions show that end-users are quite satisfied with a mean more than 4 over 5 about the main features of ANASTACIA framework such as useful to monitor and detect cyber-attacks; valuable automatic reactions to mitigate cyber-attacks; intuitive user interfaces for managing security policies and operations; and clear seal and report about the cybersecurity status. Moreover, almost all end-users would like to have ANASTACIA framework to protect their cybersecurity and privacy in their workspace or at home.

5 ANNEX 1 – CAPTURED IMAGES DURING SMART BUILDING DEMO

Below it shows the photos and captured images taken during the Smart Building demo in the face-to-face workshop done in Murcia to showcase the ANASTACIA framework.



Figure 15: Start of the workshop, explaining the demo process

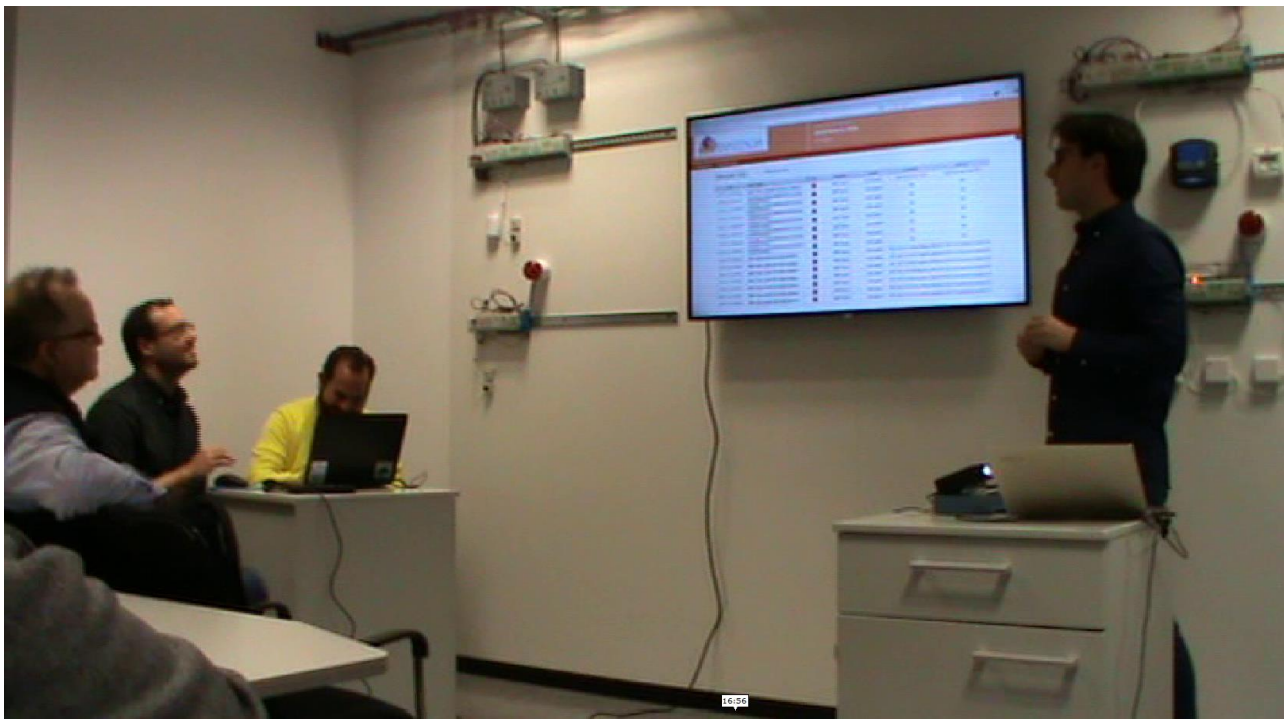


Figure 16: Showcasing the different users interfaces of the ANASTACIA framework



Figure 17: Audience attending to the ANASTACIA workshop

The system is in a steady state, and we proceed to add IoT devices into the system.

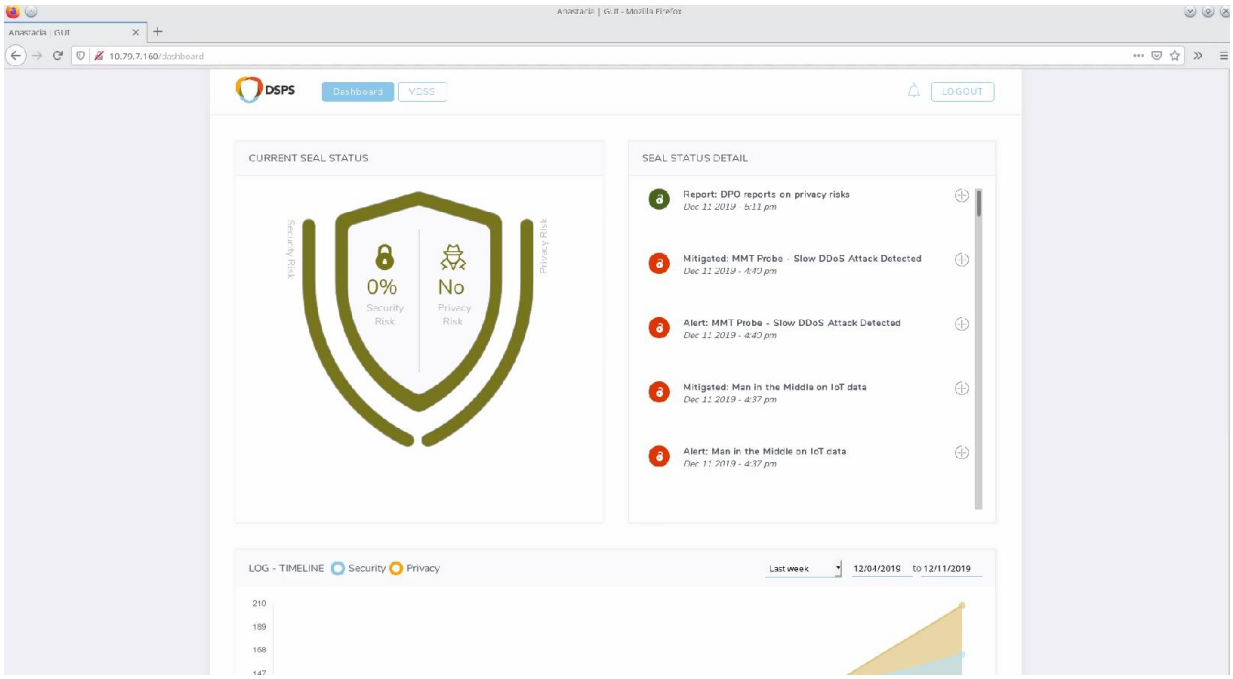


Figure 18: DSPS interface, showing a normalize state – GREEN.



Figure 19: Turning IoT devices ON

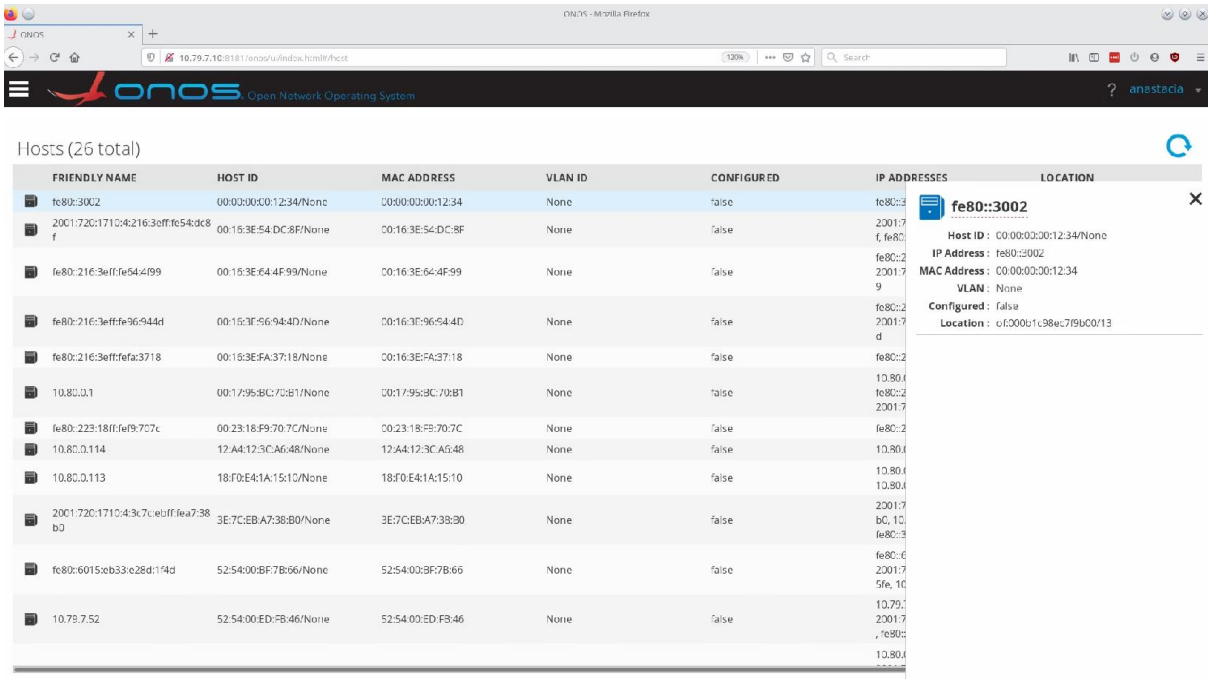


Figure 20: Devices show in the system, but not authenticated and need permission to bootstrapping to access the network

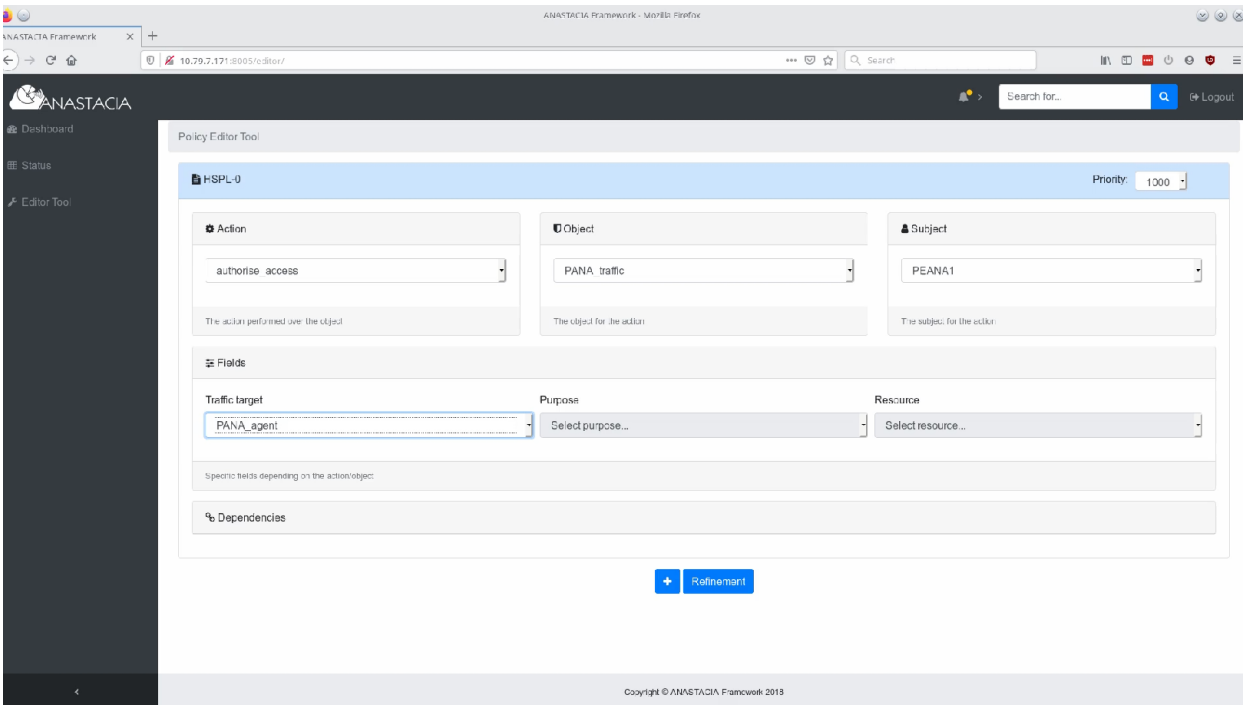


Figure 21: Several step process to set up the security policies for the Bootstrapping

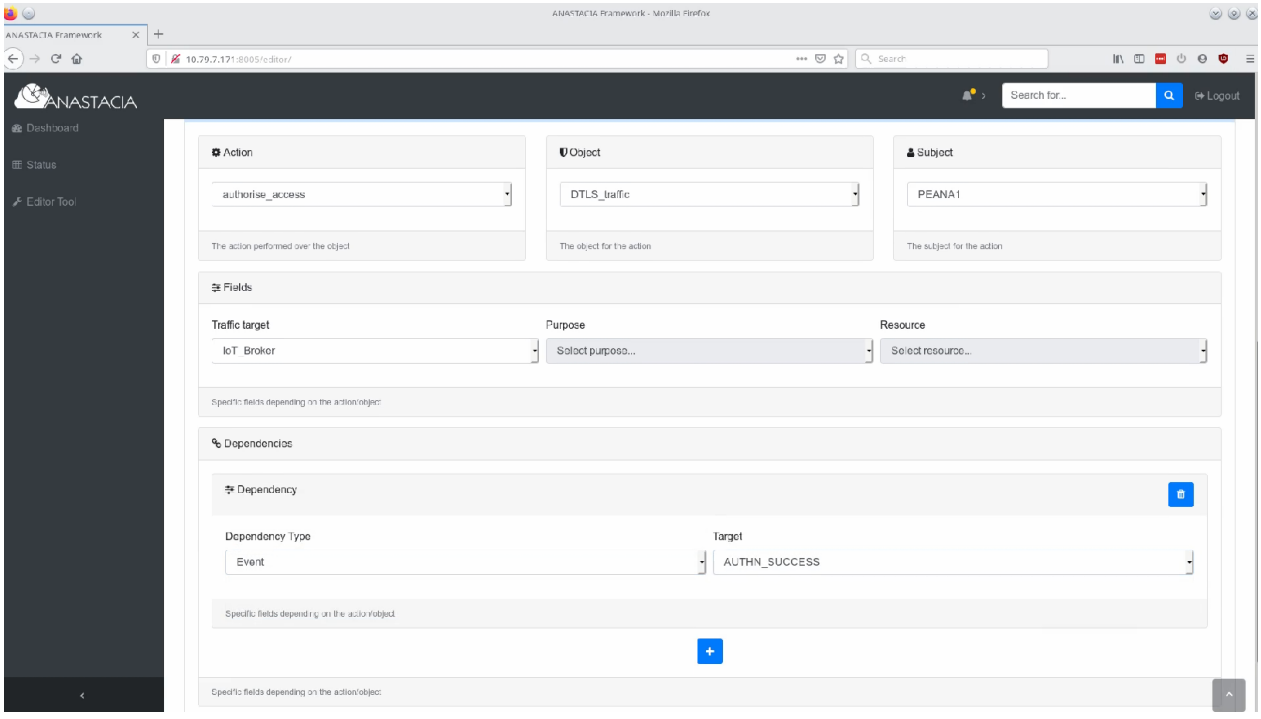


Figure 22: Definition and adding dependencies in the security policy editor

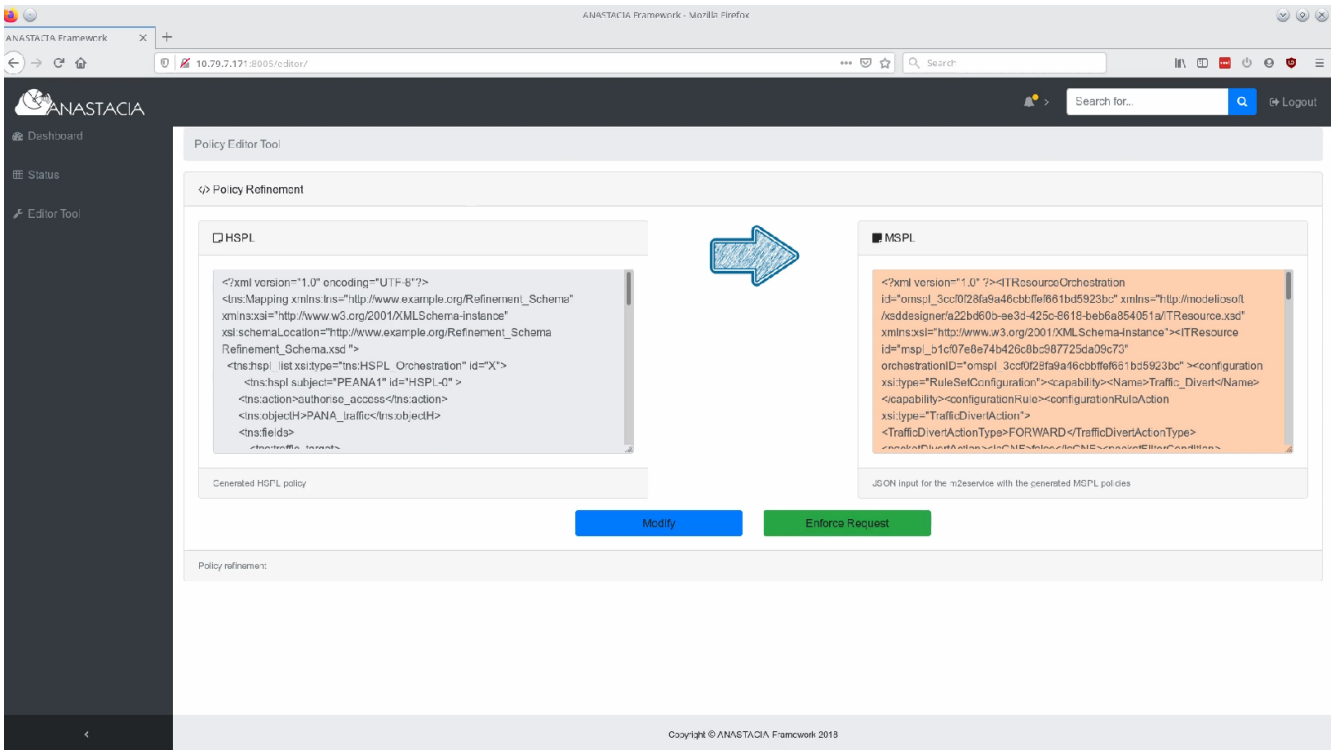


Figure 23: Refining the policies to make them actionable (from HSPL to MSPL).

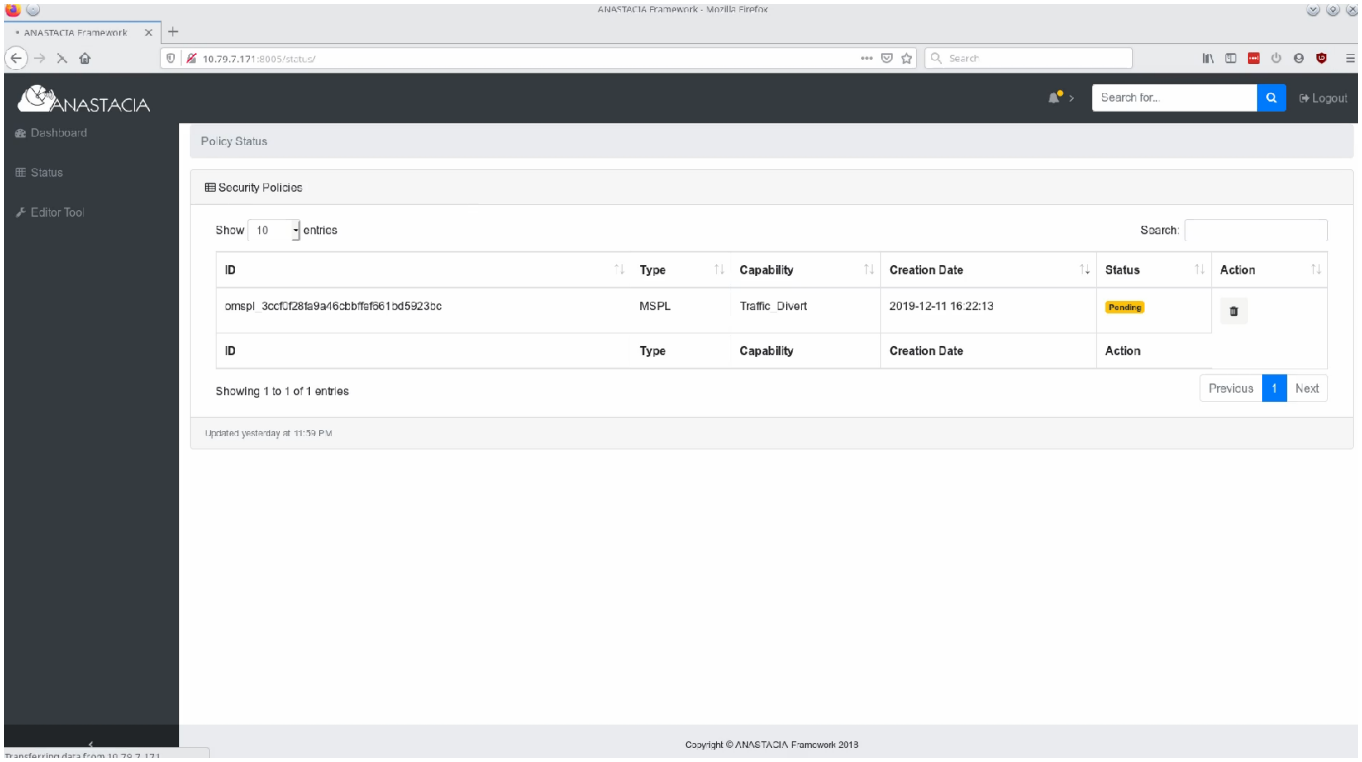


Figure 24: Watching the state of the policies execution.

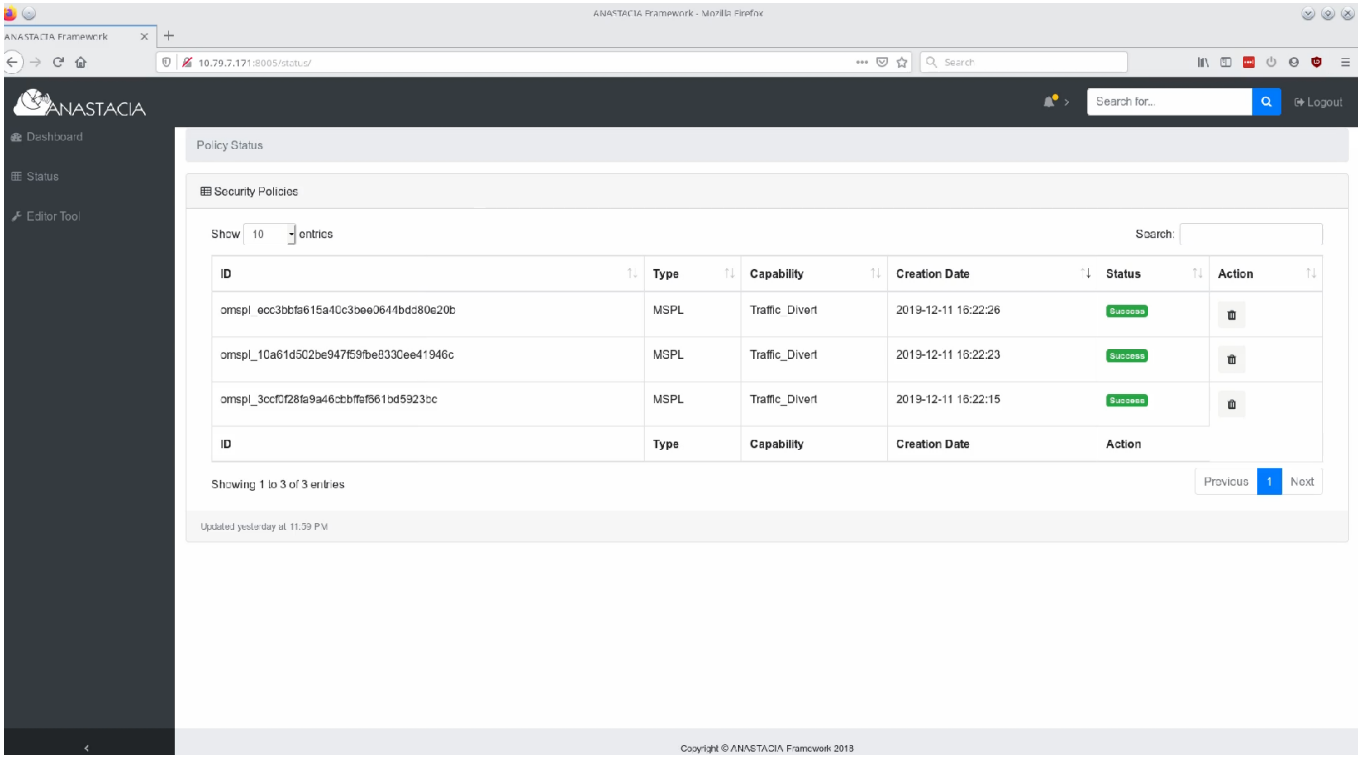


Figure 25: Different policies are enabled successfully

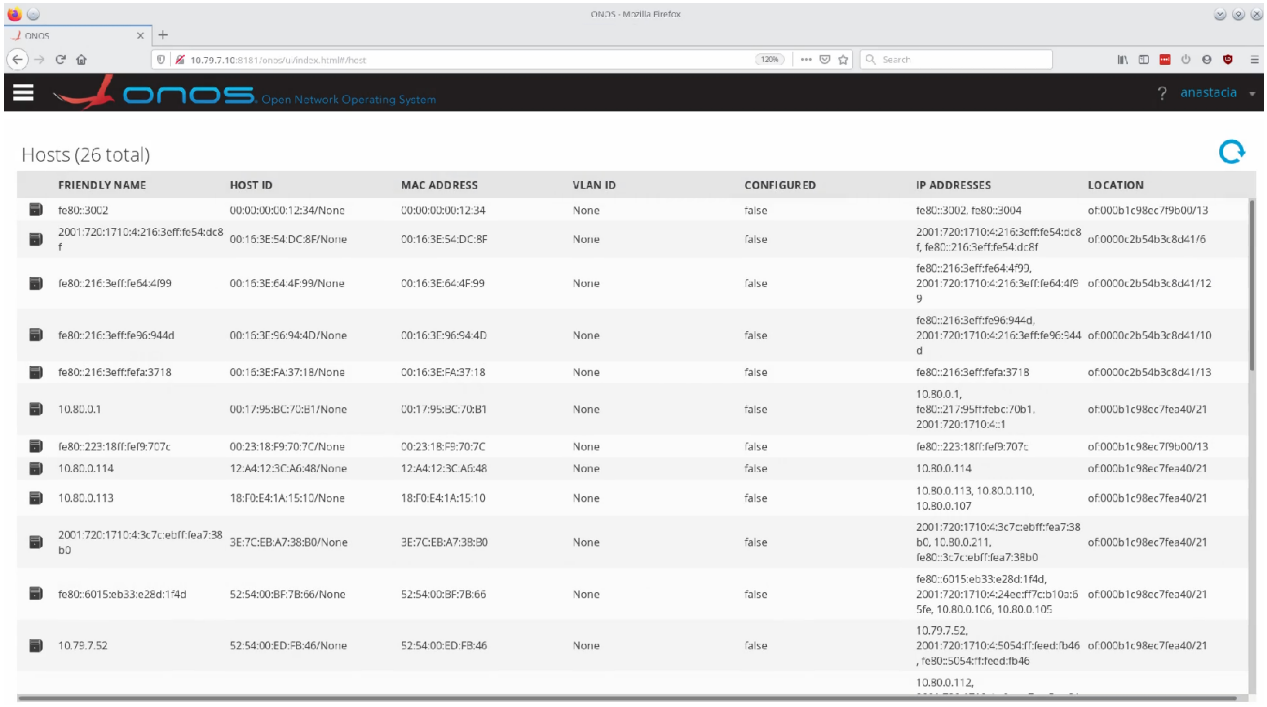


Figure 26: Watching the ONOS interface, waiting for IoT devices to show as they bootstrap

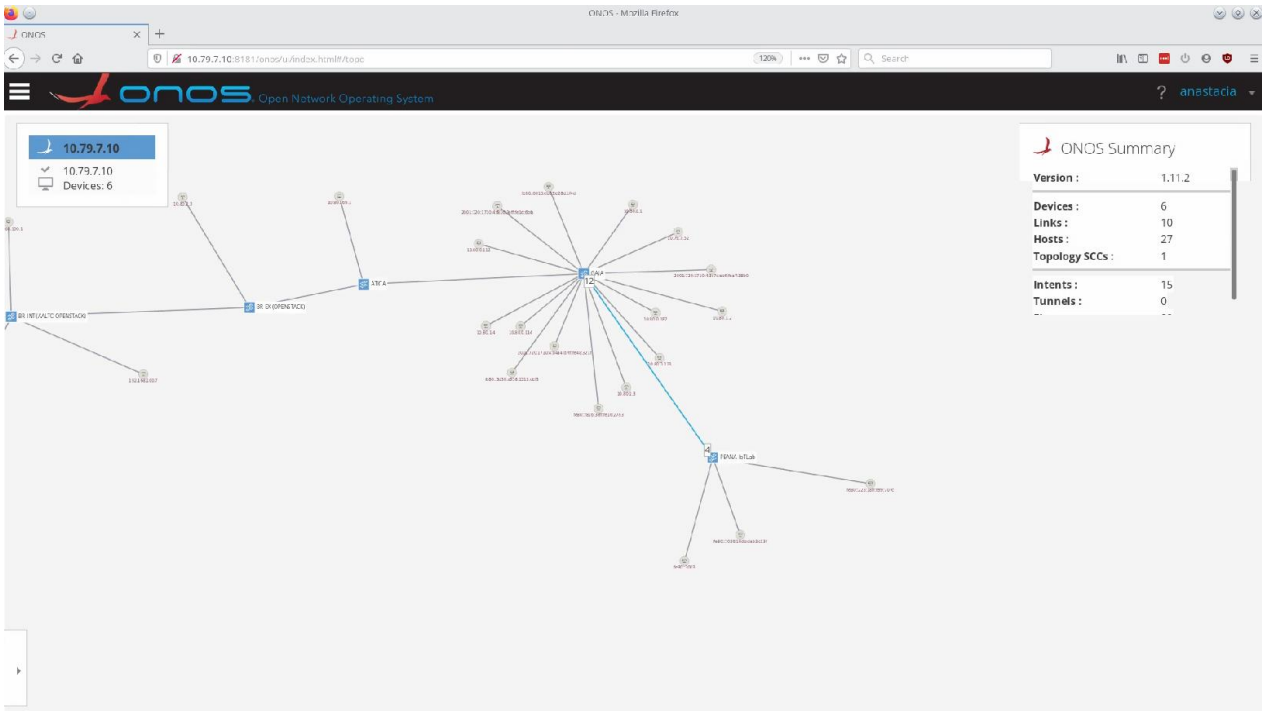
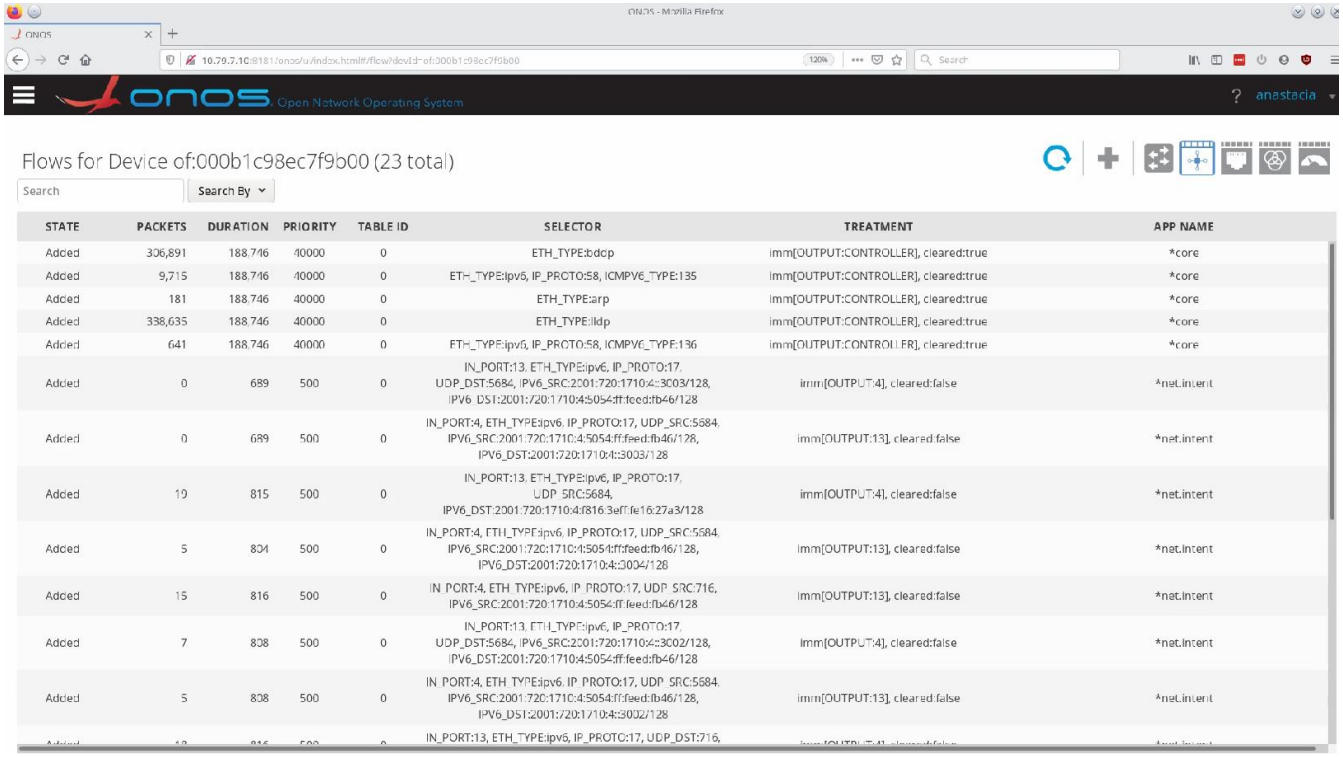


Figure 27: Devices start showing in ONOS as they are recognized



STATE	PACKETS	DURATION	PRIORITY	TABLE ID	SELECTOR	TREATMENT	APP NAME
Added	306,891	188.746	40000	0	ETH_TYPE:bdcp	imm[OUTPUT:CONTROLLER], cleared:true	*core
Added	9,715	188.746	40000	0	ETH_TYPE:ipv6, IP_PROTO:58, ICMPV6_TYPE:135	imm[OUTPUT:CONTROLLER], cleared:true	*core
Added	181	188.746	40000	0	ETH_TYPE:arp	imm[OUTPUT:CONTROLLER], cleared:true	*core
Added	338,635	188.746	40000	0	ETH_TYPE:lldp	imm[OUTPUT:CONTROLLER], cleared:true	*core
Added	641	188.746	40000	0	ETH_TYPE:ipv6, IP_PROTO:58, ICMPV6_TYPE:136	imm[OUTPUT:CONTROLLER], cleared:true	*core
Added	0	689	500	0	IN_PORT:13, ETH_TYPE:ipv6, IP_PROTO:17, UDP_DST:5684, IPV6_SRC:2001:720:1710:4::3003/128, IPV6_DST:2001:720:1710:4:5054:ff:feed:fb46/128	imm[OUTPUT:4], cleared:false	*net.intent
Added	0	689	500	0	IN_PORT:4, ETH_TYPE:ipv6, IP_PROTO:17, UDP_SRC:5684, IPV6_SRC:2001:720:1710:4:5054:ff:feed:fb46/128, IPV6_DST:2001:720:1710:4::3003/128	imm[OUTPUT:13], cleared:false	*net.intent
Added	19	815	500	0	IN_PORT:13, ETH_TYPE:ipv6, IP_PROTO:17, UDP_SRC:5684, IPV6_DST:2001:720:1710:4:f816:3eff:fe16:27a3/128	imm[OUTPUT:4], cleared:false	*net.intent
Added	5	804	500	0	IN_PORT:4, ETH_TYPE:ipv6, IP_PROTO:17, UDP_SRC:5684, IPV6_SRC:2001:720:1710:4:5054:ff:feed:fb46/128, IPV6_DST:2001:720:1710:4::3004/128	imm[OUTPUT:13], cleared:false	*net.intent
Added	15	816	500	0	IN_PORT:4, ETH_TYPE:ipv6, IP_PROTO:17, UDP_SRC:716, IPV6_SRC:2001:720:1710:4:5054:ff:feed:fb46/128	imm[OUTPUT:13], cleared:false	*net.intent
Added	7	808	500	0	IN_PORT:13, ETH_TYPE:ipv6, IP_PROTO:17, UDP_DST:5684, IPV6_SRC:2001:720:1710:4::3002/128, IPV6_DST:2001:720:1710:4:5054:ff:feed:fb46/128	imm[OUTPUT:4], cleared:false	*net.intent
Added	5	808	500	0	IN_PORT:4, ETH_TYPE:ipv6, IP_PROTO:17, UDP_SRC:5684, IPV6_SRC:2001:720:1710:4:5054:ff:feed:fb46/128, IPV6_DST:2001:720:1710:4::3002/128	imm[OUTPUT:13], cleared:false	*net.intent
Added	13	816	500	0	IN_PORT:13, ETH_TYPE:ipv6, IP_PROTO:17, UDP_DST:716,	imm[OUTPUT:4], cleared:false	*net.intent

Figure 28: Different intents show in ONOS as the IoT devices perform the bootstrapping

- **Phase R1.** Attack with an infected IoT device sending sensitive data traffic using a DTLS connection to an outside endpoint.



Figure 29: A roge IoT device is turned on to emulate an attack sending data to an external endpoint

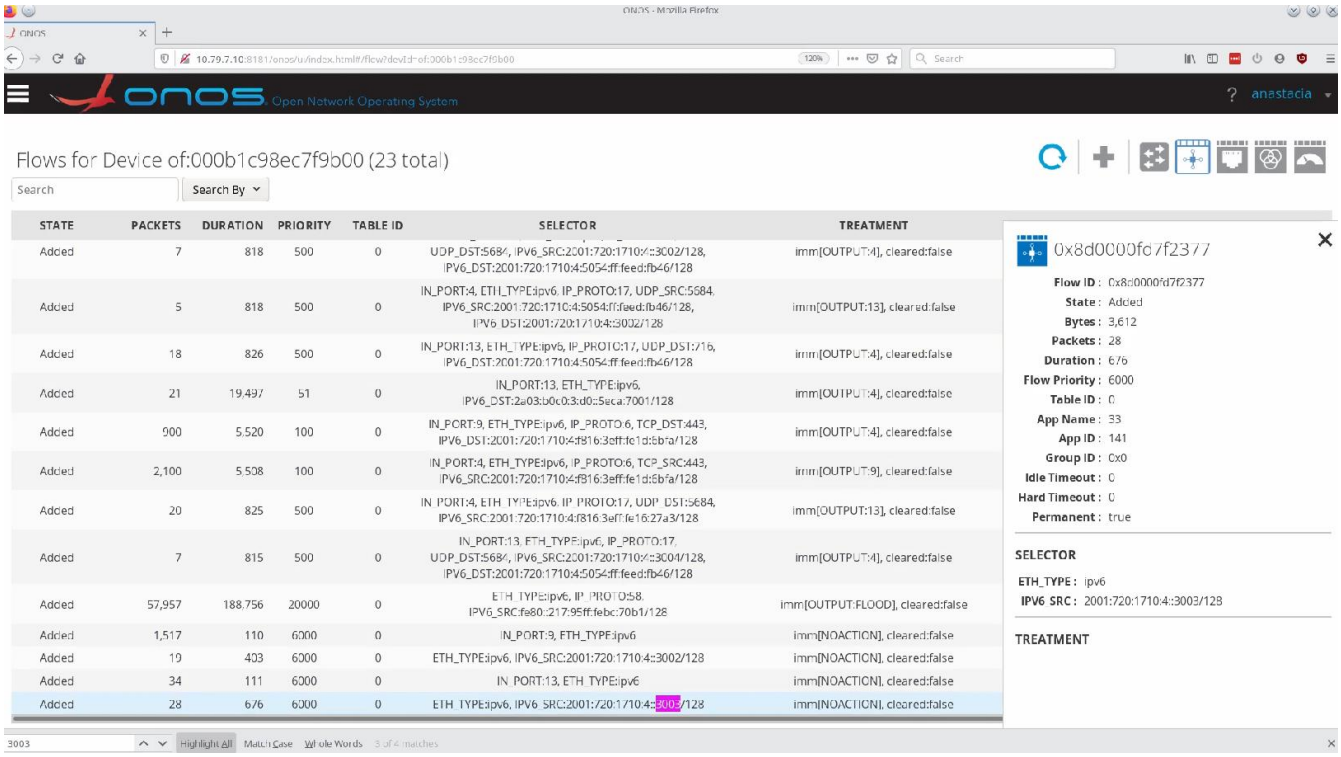


Figure 30: The new device is integrated in the system.

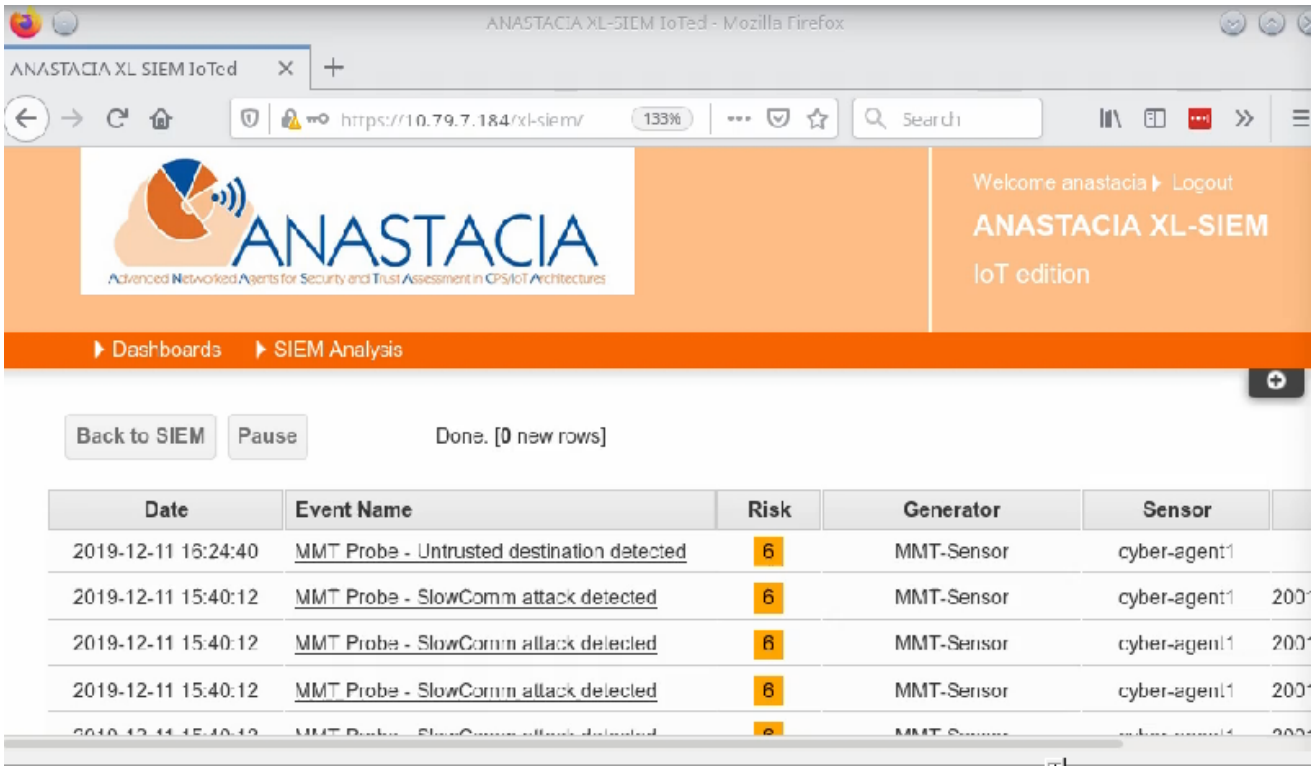


Figure 31: The untrusted endpoint is detected by MMT probe and notified to Incident Detector.

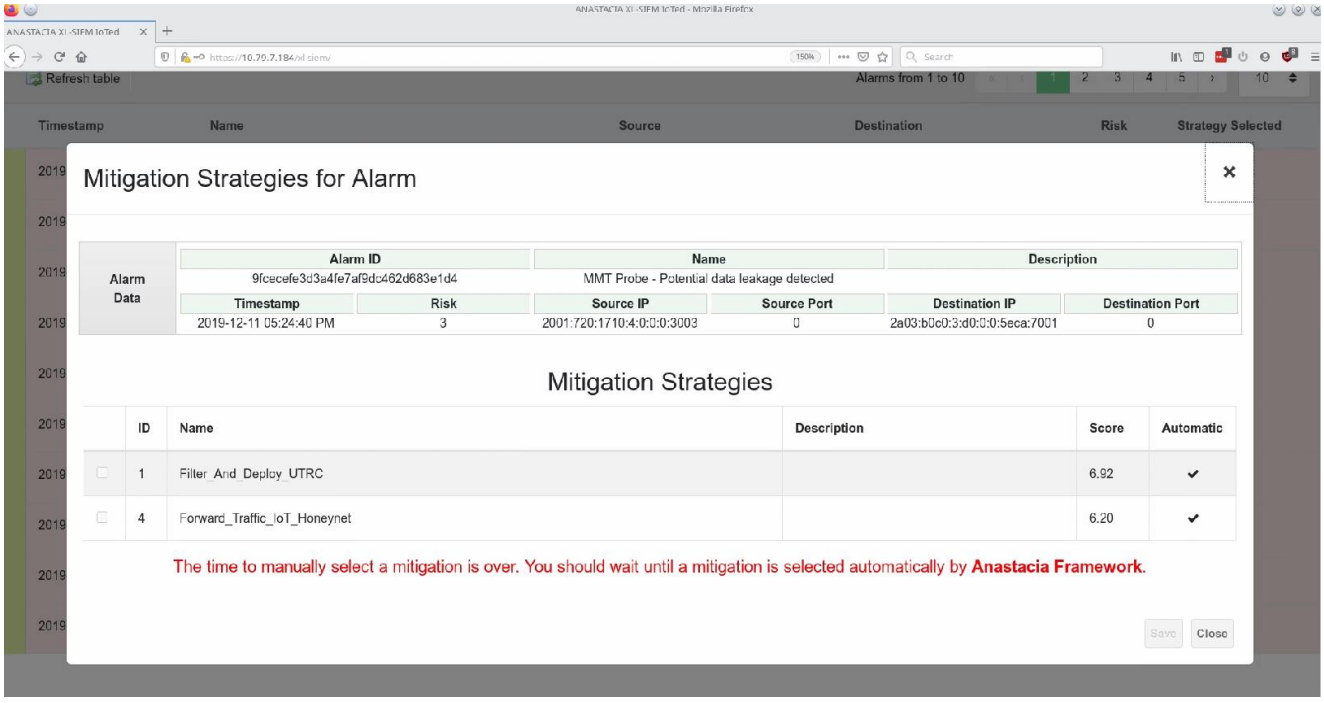


Figure 32: Incident Detector processes the event and generates a "Potential data leakage detected" alert and mitigation strategies such as filtering and forwarding.

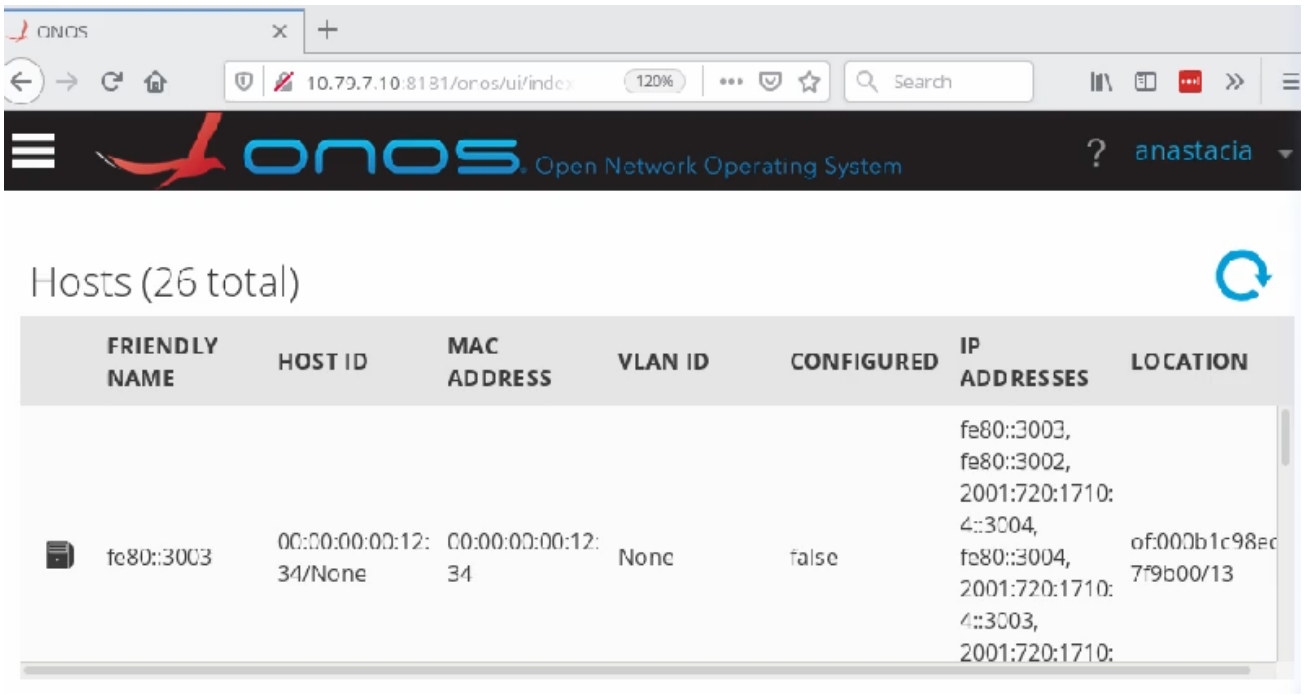


Figure 33: The ONOS SDN controller received the order from the Security-Orchestrator to create the rule for forwarding the traffic generated by the malicious device.

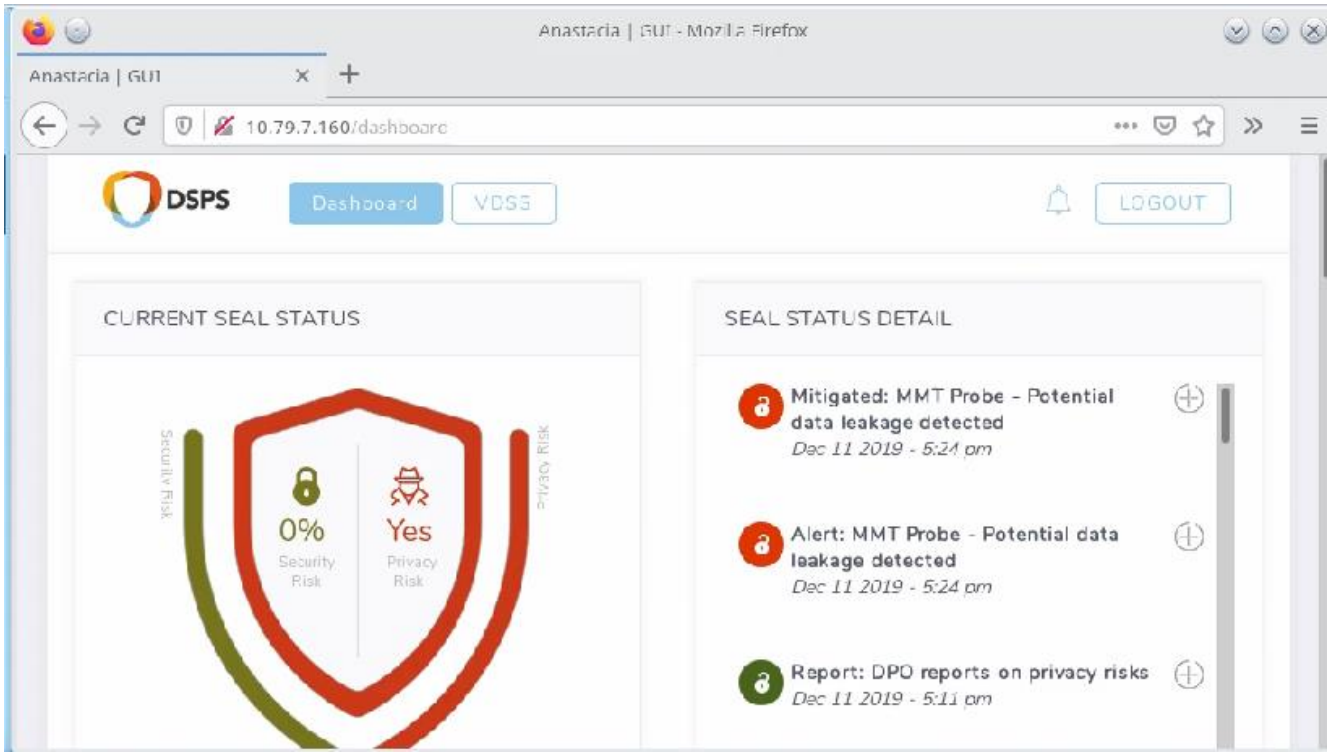


Figure 34: DSPS shows the seal status after the detection and mitigation of the potential data leakage

- **Phase R2.** Attack with an infected temperature sensor sending high temperature to activate fire alarm in the building.

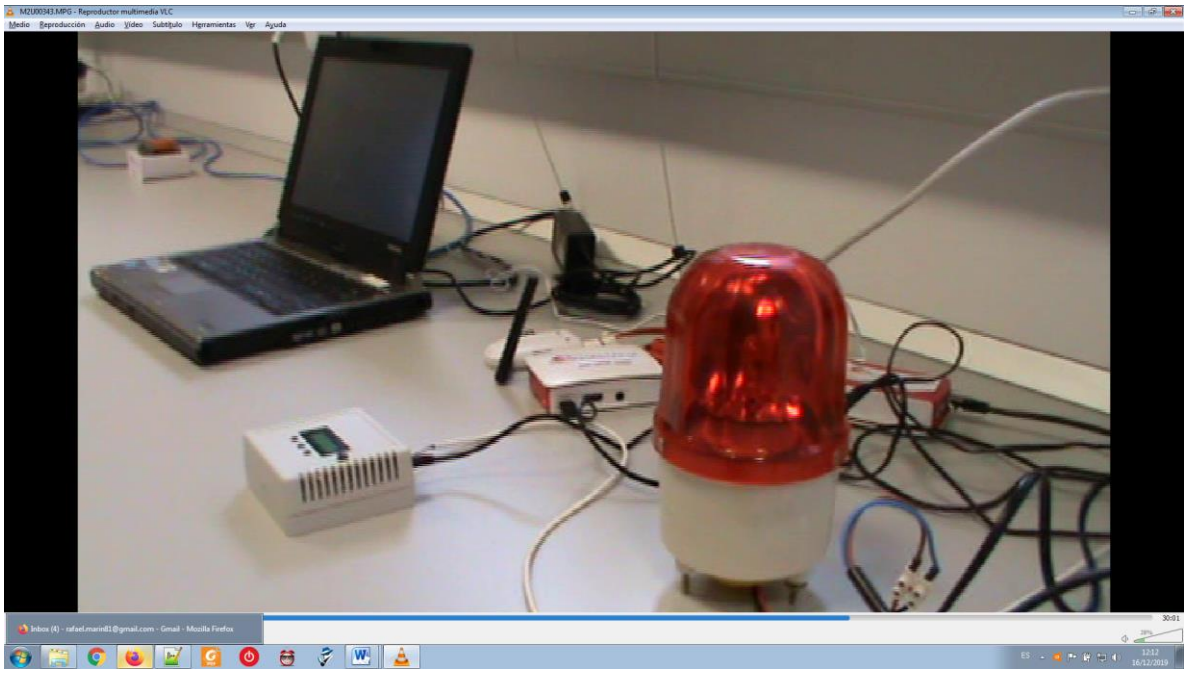


Figure 35: A siren placed in the the premises is prepared to be turned on if any of the temperature readings from the BMS is unusually high

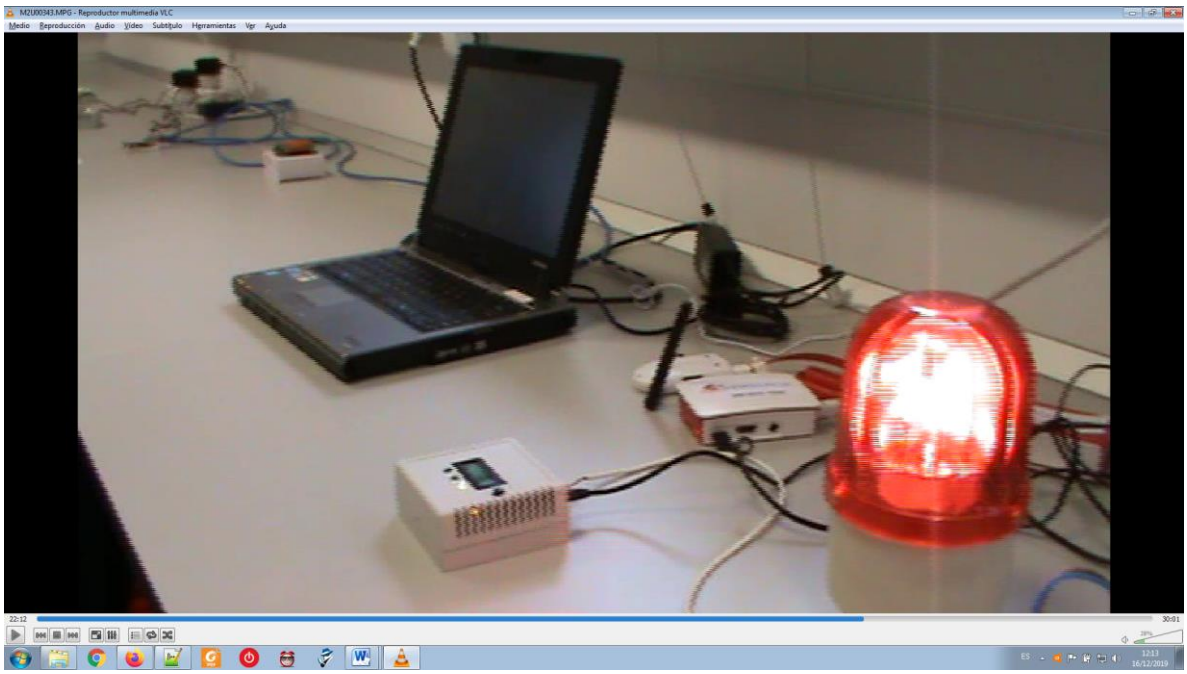


Figure 36: After an attacked IoT device publishesh a higher than normal temperature reading, the siren is turned on.

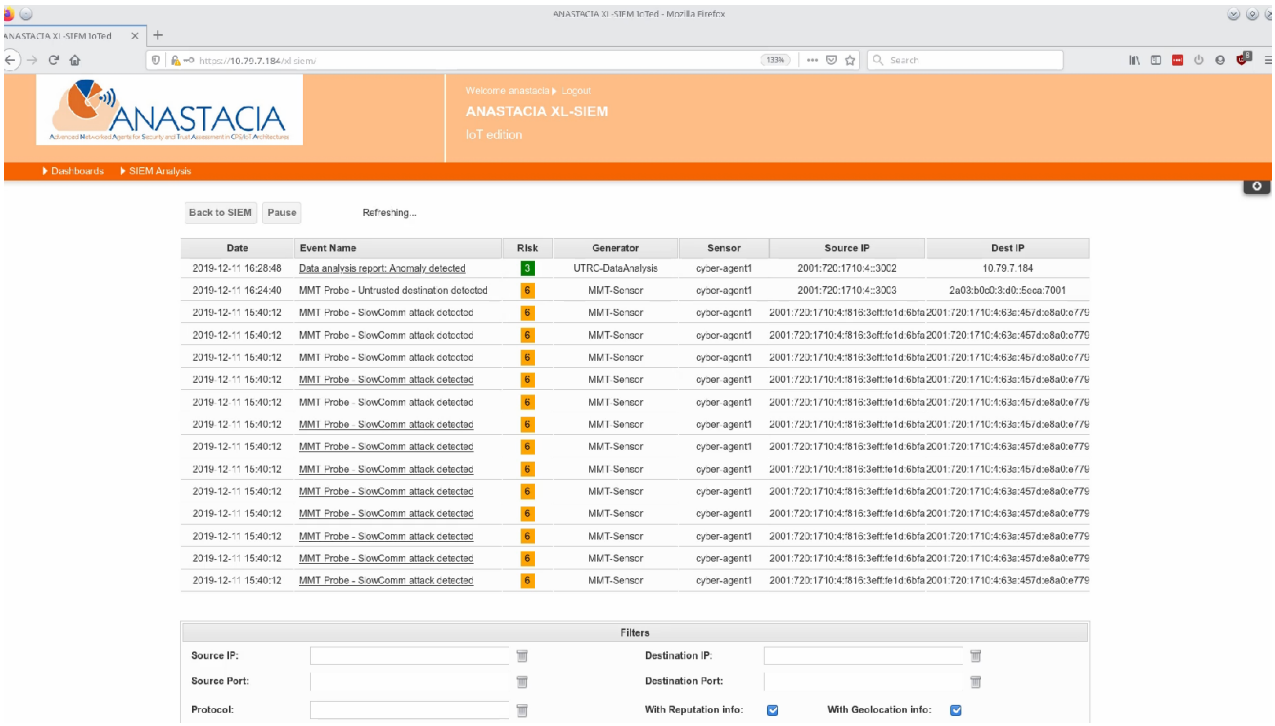


Figure 37: The VDSS shows an unusual temperature reading

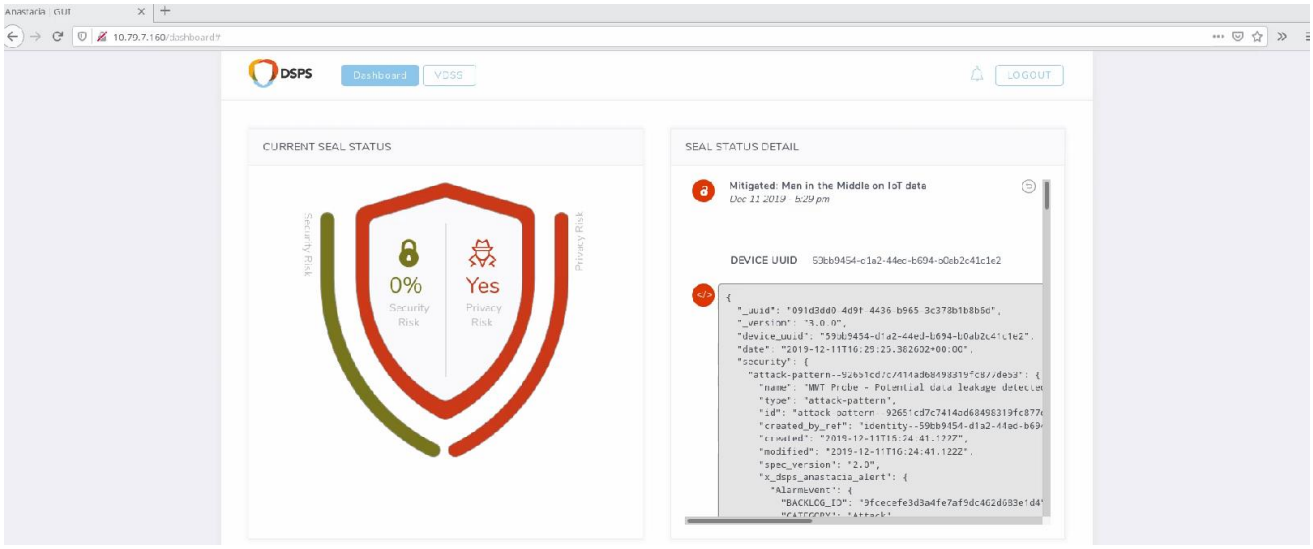


Figure 38: The DSPS shows the alert associated with a MiM attack

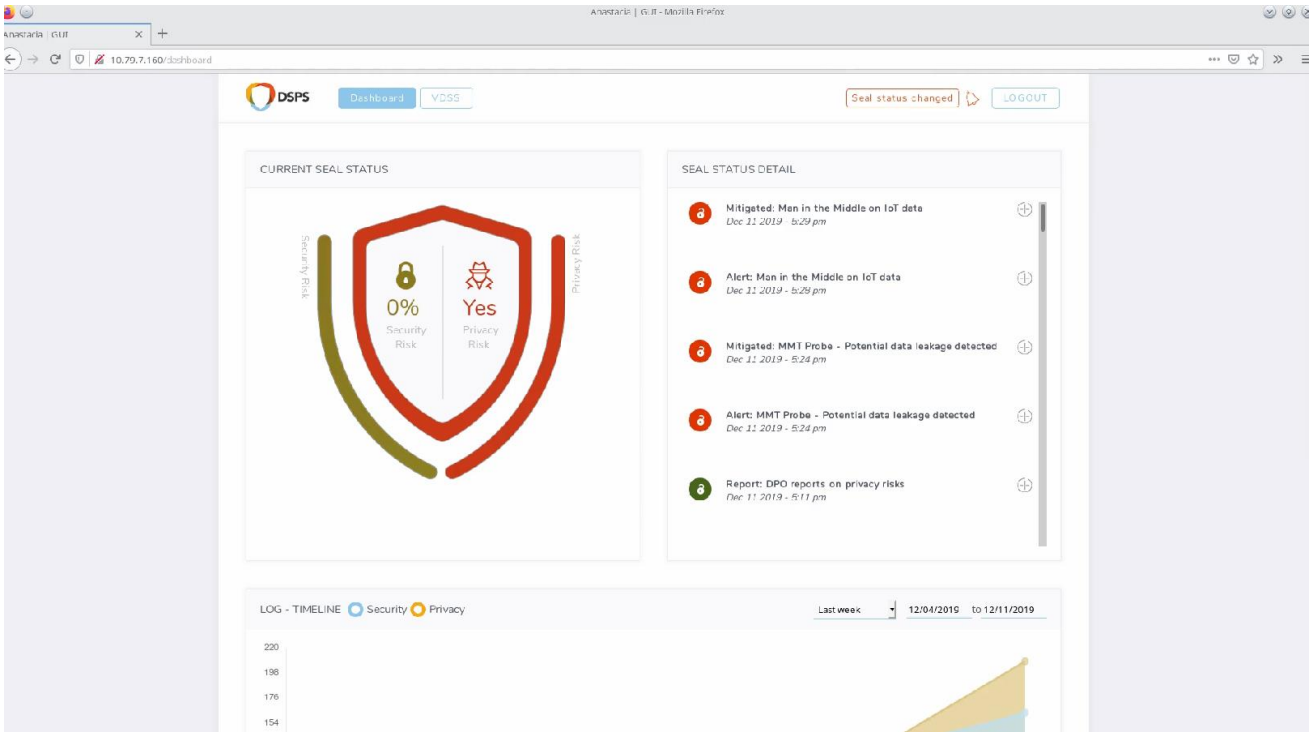


Figure 39: The system reports the MiM has been mitigated

- **Phase R3:** Slow-DoS attack against web service of BMS server.



Figure 40: An attack (Slow DOS) is performed by a CNR device in the UMU premises

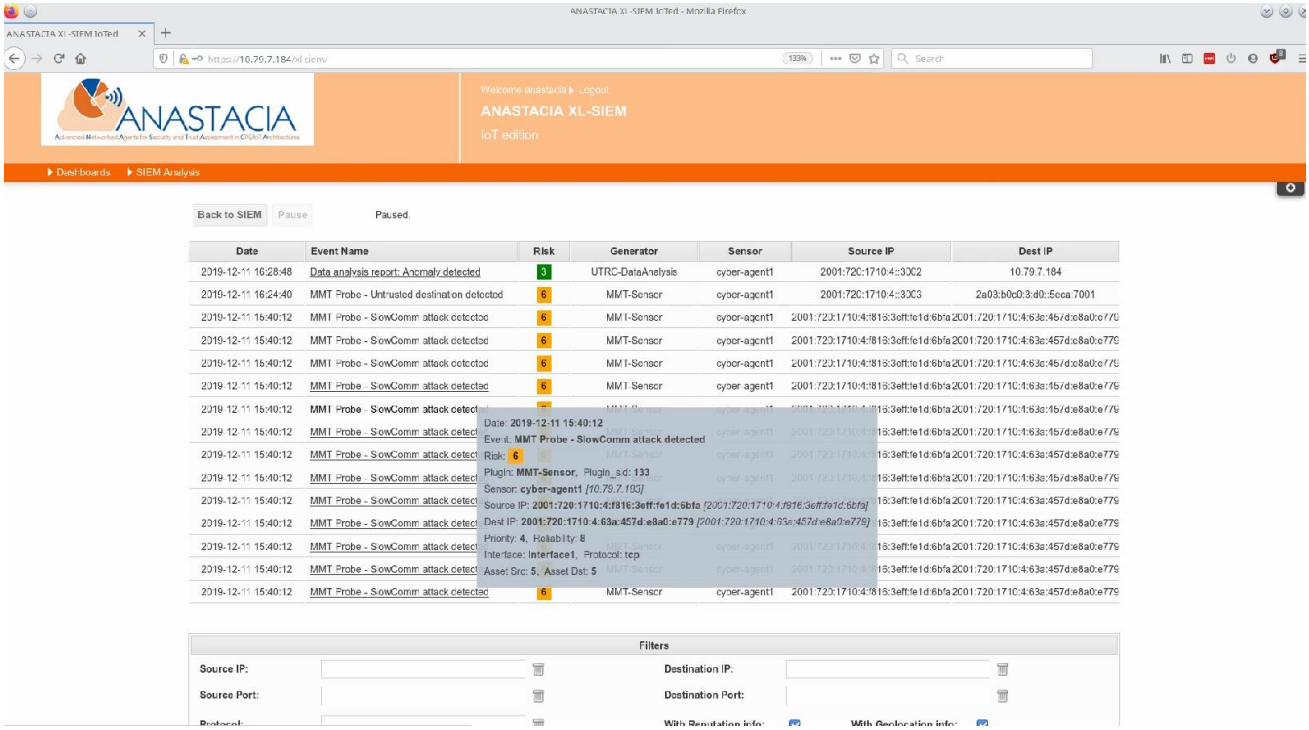


Figure 41: This attack is also detected by the ANASTACIA framework

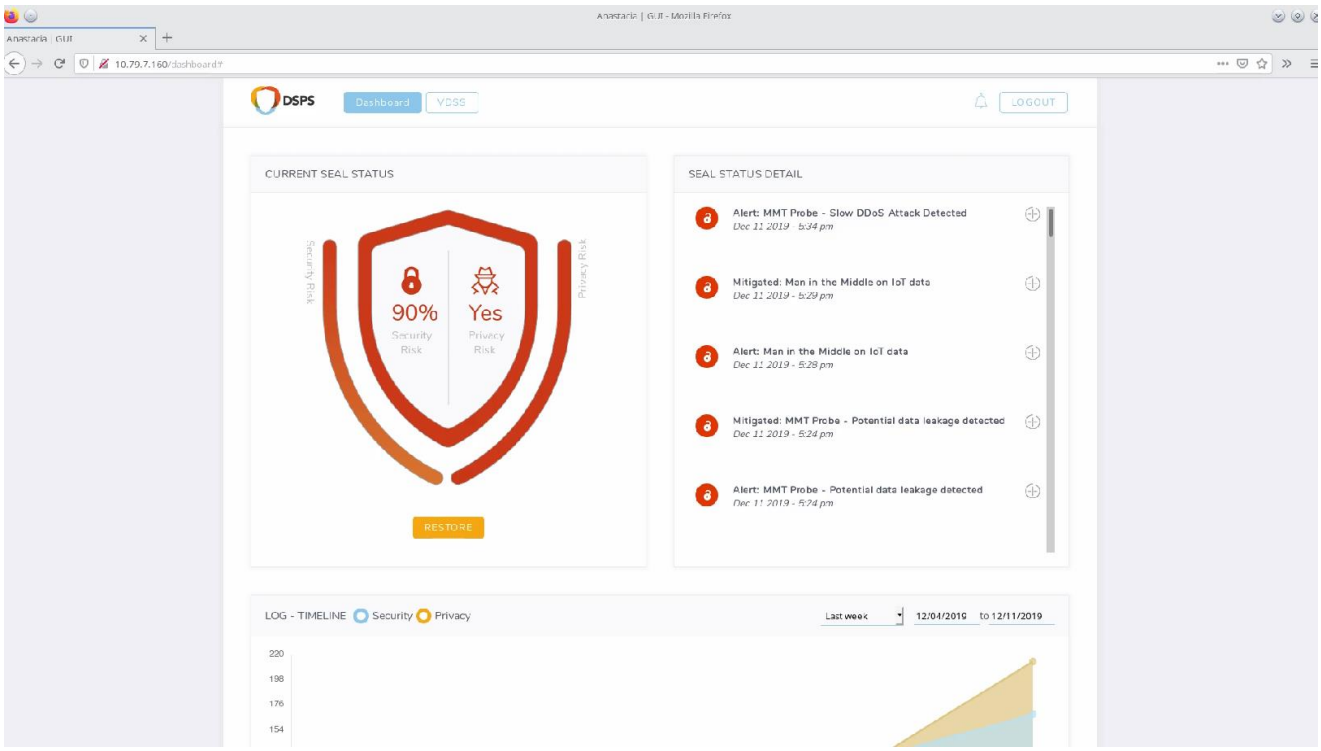
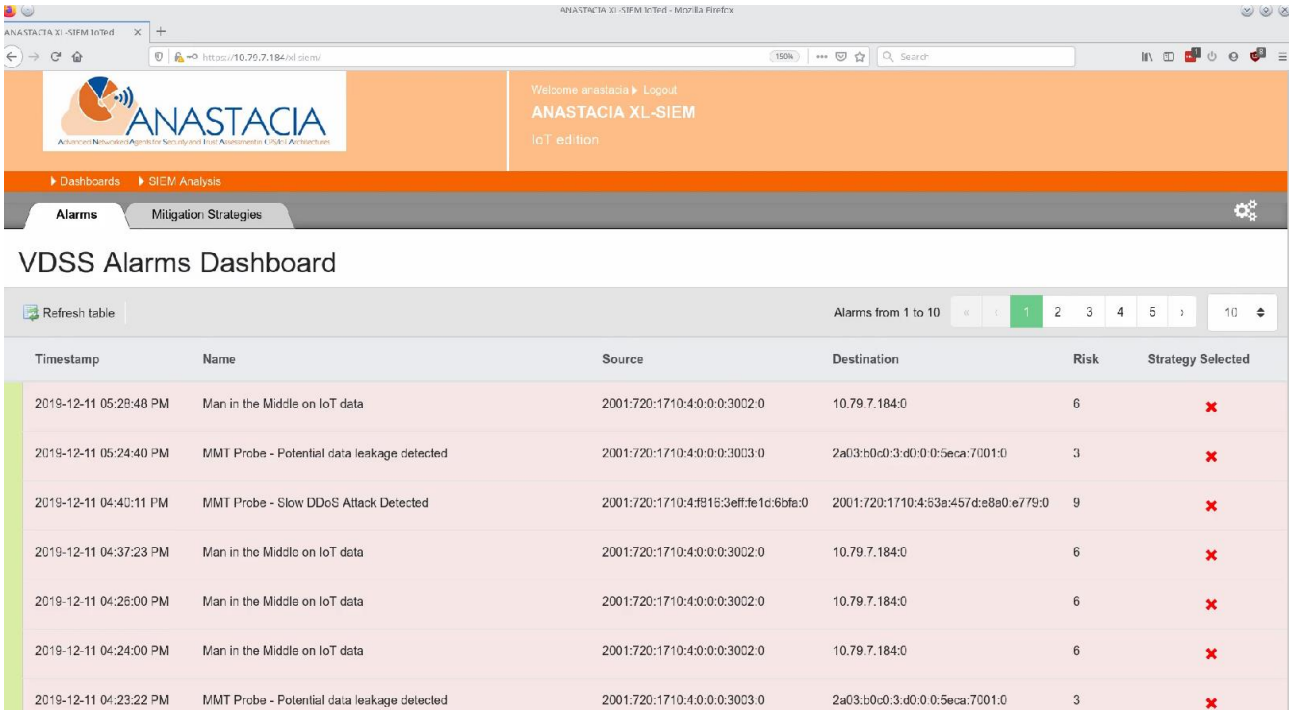


Figure 42: The DSPS shows the Slow DOS activity



Timestamp	Name	Source	Destination	Risk	Strategy Selected
2019-12-11 05:28:48 PM	Man in the Middle on IoT data	2001:720:1710:4:0:0:0:3002:0	10.79.7.184:0	6	✖
2019-12-11 05:24:40 PM	MMT Probe - Potential data leakage detected	2001:720:1710:4:0:0:0:3003:0	2a03:b0c0:3:d0:0:0:5eca:7001:0	3	✖
2019-12-11 04:40:11 PM	MMT Probe - Slow DDoS Attack Detected	2001:720:1710:4:f816:3eff:fe1d:6bfa:0	2001:720:1710:4:63a:457d:e8a0:e779:0	9	✖
2019-12-11 04:37:23 PM	Man in the Middle on IoT data	2001:720:1710:4:0:0:0:3002:0	10.79.7.184:0	6	✖
2019-12-11 04:26:00 PM	Man in the Middle on IoT data	2001:720:1710:4:0:0:0:3002:0	10.79.7.184:0	6	✖
2019-12-11 04:24:00 PM	Man in the Middle on IoT data	2001:720:1710:4:0:0:0:3002:0	10.79.7.184:0	6	✖
2019-12-11 04:23:22 PM	MMT Probe - Potential data leakage detected	2001:720:1710:4:0:0:0:3003:0	2a03:b0c0:3:d0:0:0:5eca:7001:0	3	✖

Figure 43: The VDSS shows the different alarms triggerd by ANASTACIA

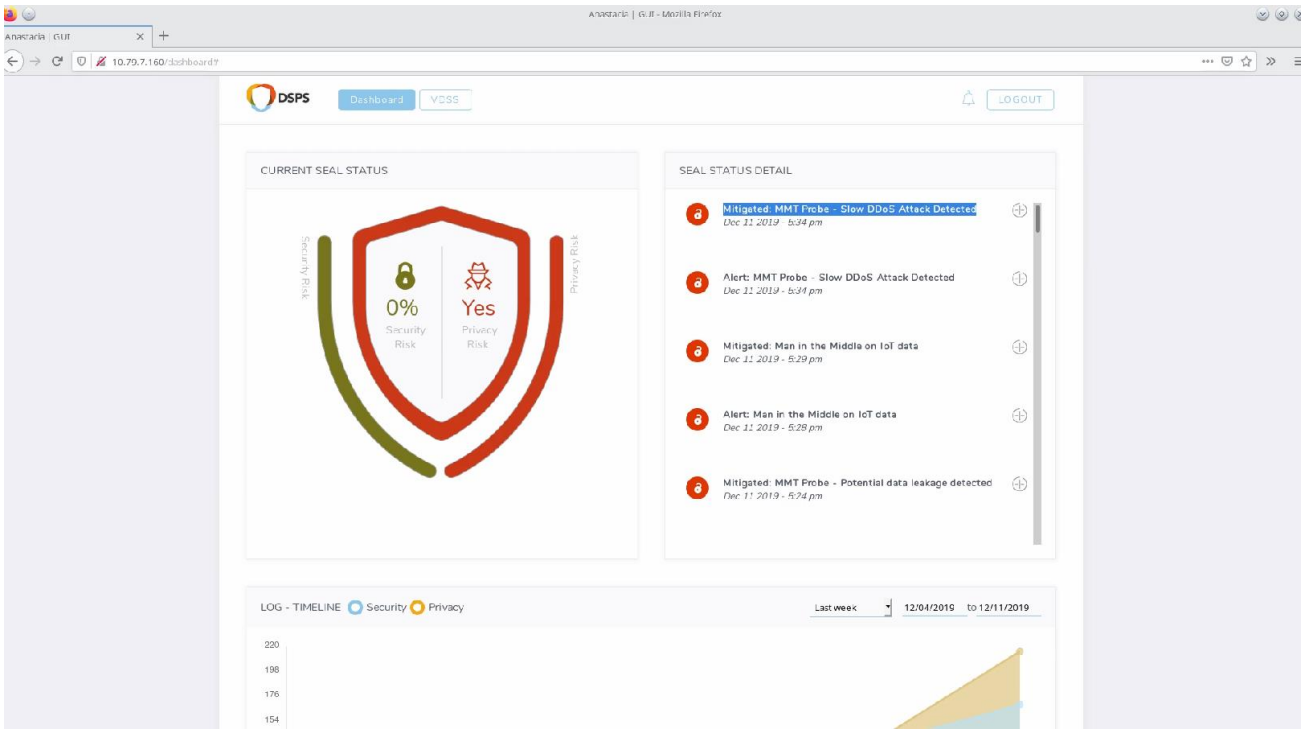


Figure 44: DSPS showing GREEN status after the SlowDoS attack is mitigated by ANASTACIA framework.

- **Phase R4:** External attacker tries to compromise an IoT device through data request from an outside endpoint.



Figure 45: The external attacker is shown in the image

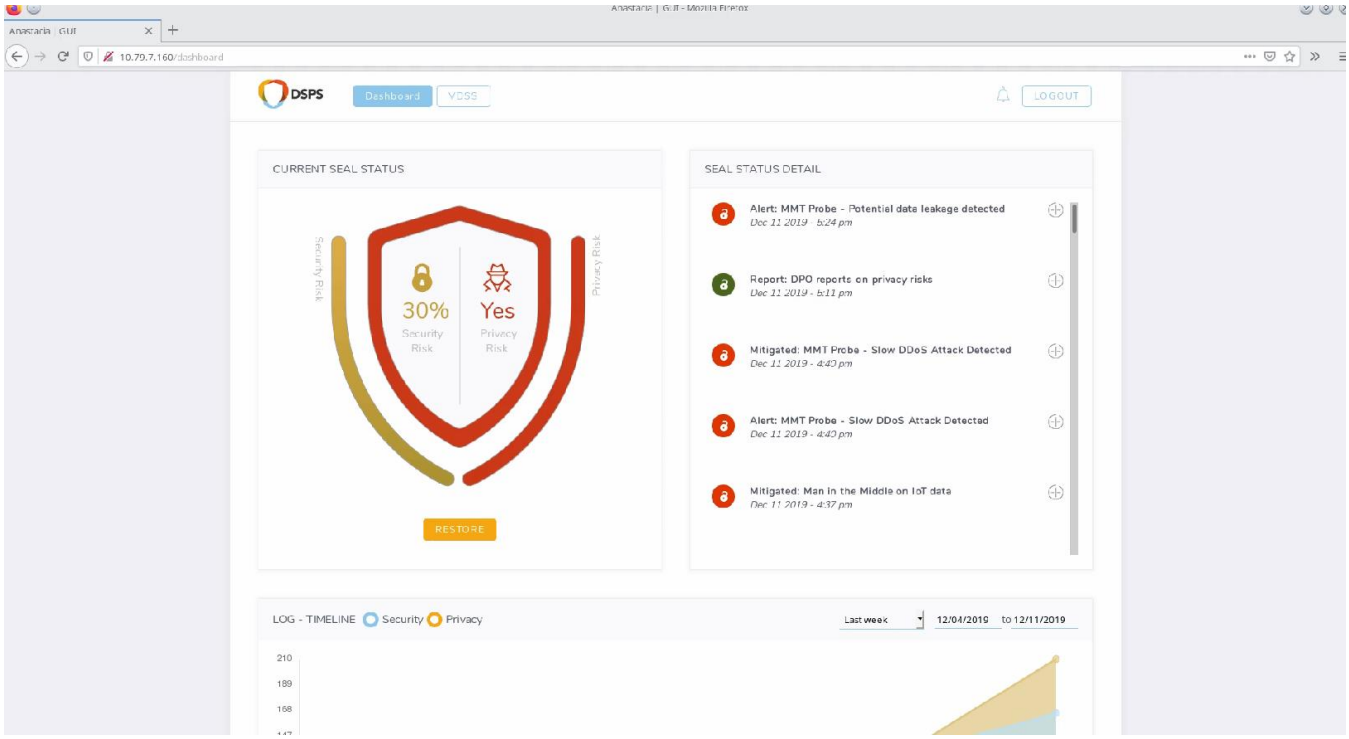


Figure 46: The potential data leakage is detected

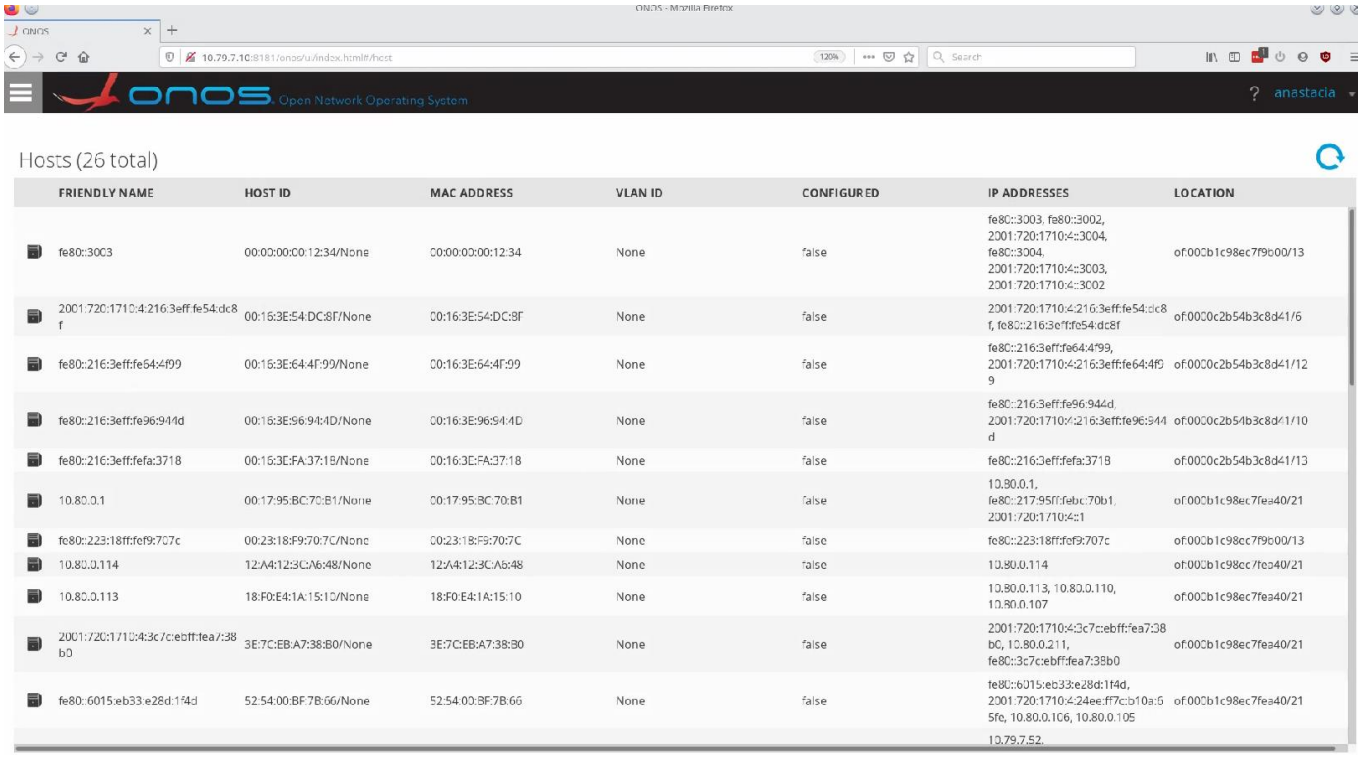


Figure 47: Intents are created to address the issue

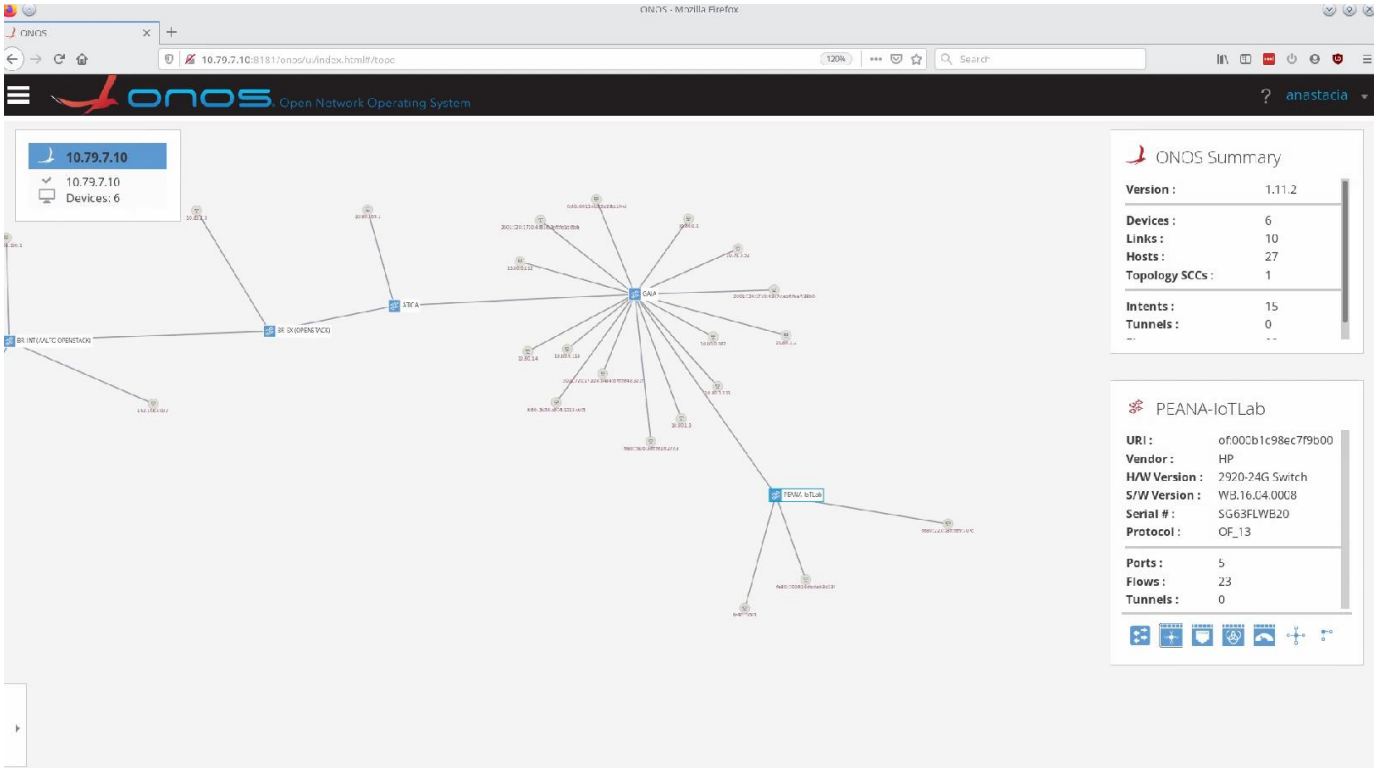


Figure 48: Final view of the ONOS topology after the Smart Building demo

6 ANNEX 2 – CAPTURED IMAGES DURING 5G MEC DEMO

This annex includes the photos and visual interfaces captured in demo session of MEC in 5G scenario performed in Murcia to showcase the benefits of ANASTACIA framework. Below, some photos of the equipment involved in the 5G scenario are showed. The 5G core is running in generic rack machines as VNFs so there are no fancy special hardware. Due to company policies the figures have been modified so that identification tags are removed from machines. There are also some screenshots from the execution of ANASTACIA framework and the 5G slice orchestrator to avoid an attack over a 5G camera.



Figure 49: Servers running the 5G network core

The following screenshots show the available network slices. The priority slice provides unrestricted access with high bandwidth to legitimate devices. The filestreamer slice and the limited filestreamer slice are slices with more restrictions, additional security and lower available bandwidth. Misbehaving devices are restricted from accessing the priority slice and can only access the filestreamer slices.

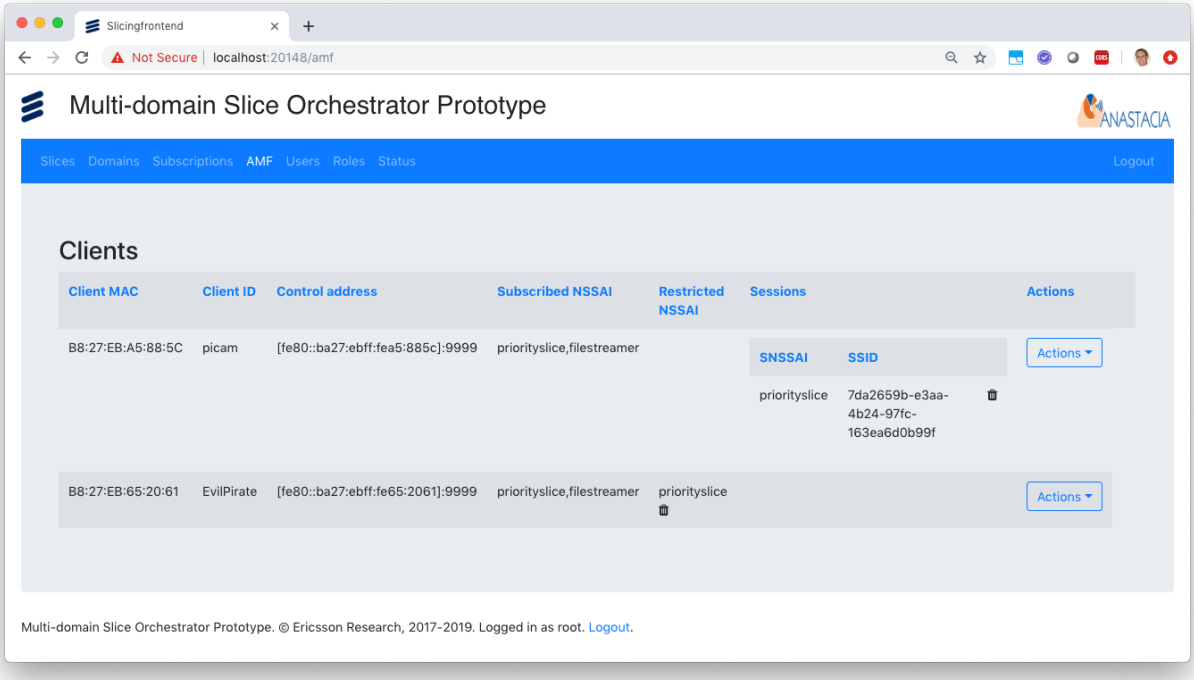


Figure 50: Slicer prototype portal

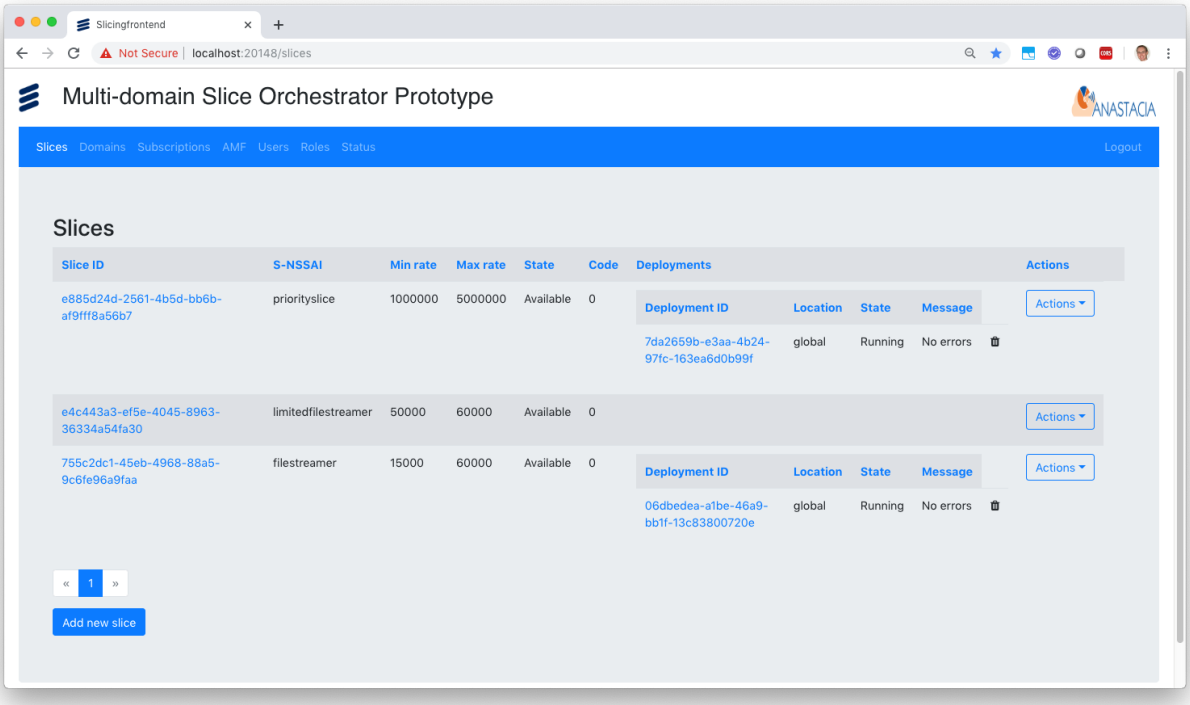


Figure 51: legitimate IoT camera device with a session attached

Figure 51 shows a legitimate IoT camera device with a session attached to the priority slice. It also shows a misbehaving IoT device that due to the detection and mitigation taking place in the Anastacia framework has been restricted from accessing the priority slice.

Figure 53 show a raspberry that acts as the legitimate and the misbehaving devices. Figure 52 show a NUC, which that implements the device running emulated 5G base station. Figure 54 shows servers 1-4 that are running the 5G network core.



Figure 52: NUC running the 5G base station



Figure 53: Raspberry, runing the legitimata and illegitimate devices.



Figure 54: Servers 1-4 running the 5G core

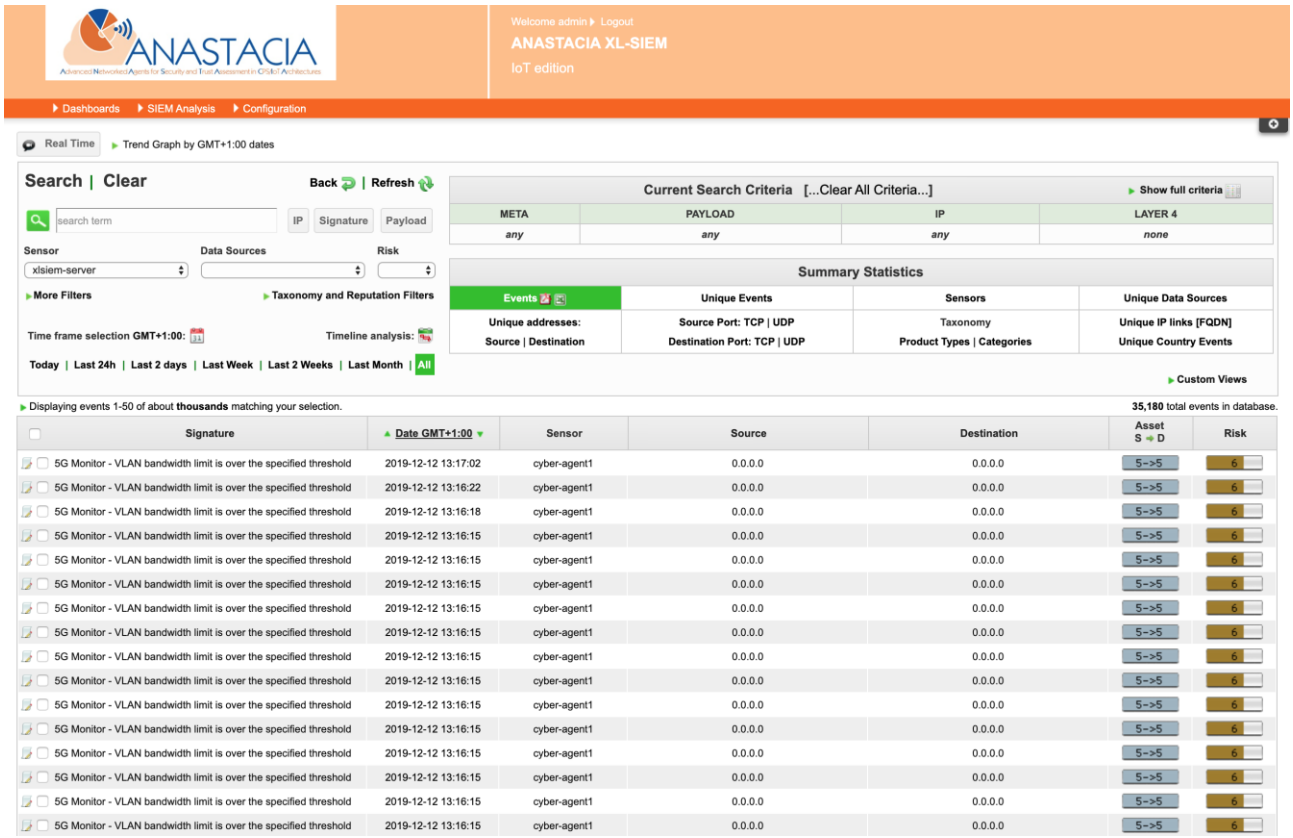


Figure 55: View of the ANASTACIA framework showing the 5G monitor activity

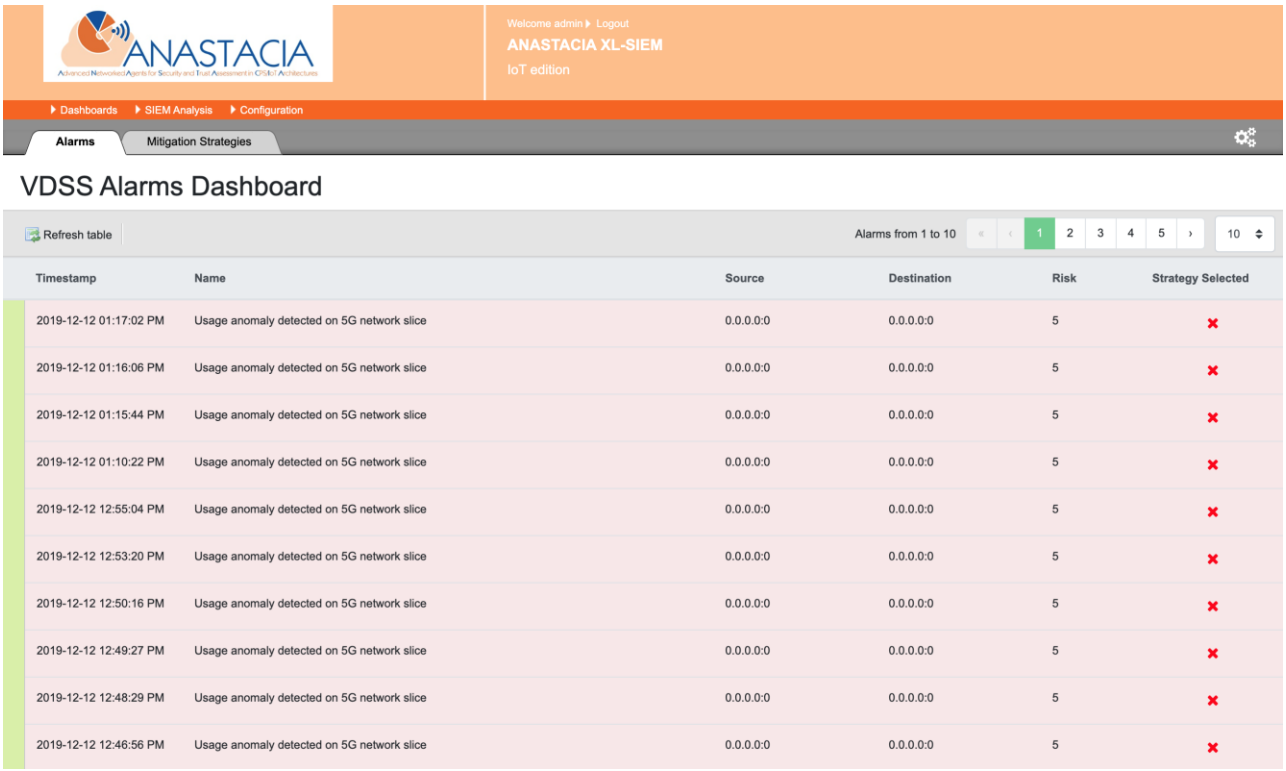


Figure 56: VDSS showing the anomal activity of the camera

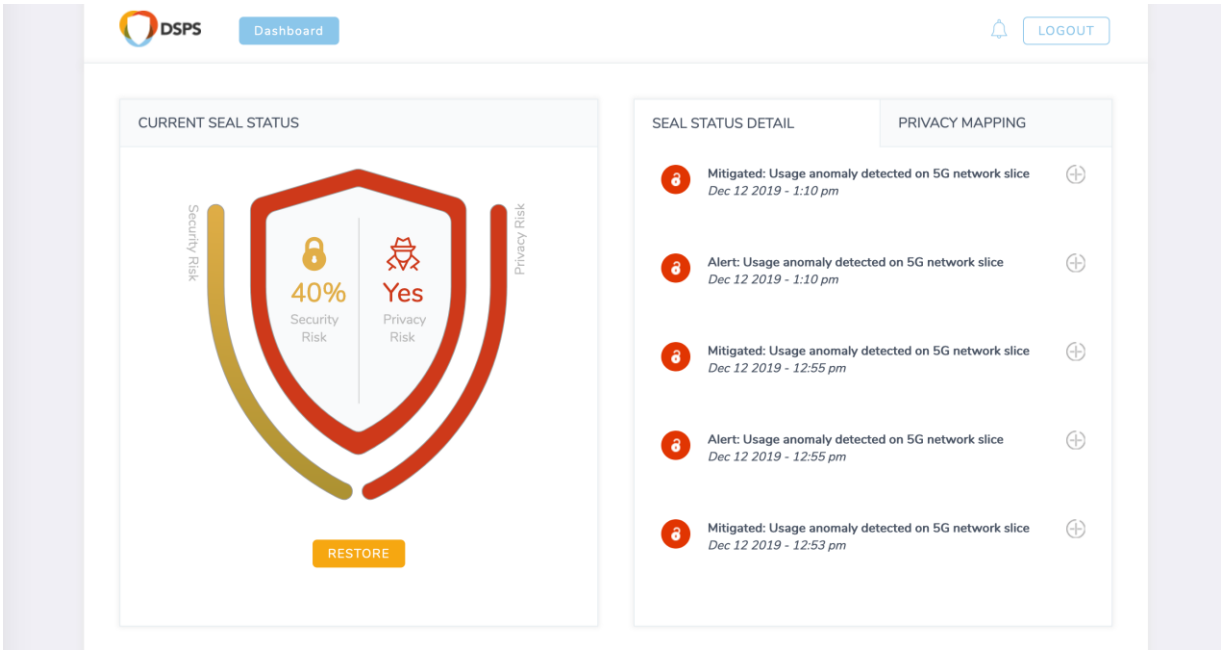


Figure 57: DSPS showing the 5G anomaly is mitigated

7 ANNEX 3 – INTERVIEW QUESTIONNAIRES

The following tables present the results of 59 end-users answers to the evaluation questionnaire.

Number of Questions	Results of General Questions about ANASTACIA framework	MEAN	DEVIATION
1	1. Does ANASTACIA framework have an intuitive user interface for managing security policies and operations?	4,196550699	0,6373345
2	2. Does the ANASTACIA framework provide valuable automatic reactions to mitigate cyber-attacks?	4,339280404	0,7336768
3	3. Is the nastacia framework useful to monitor and detect cyber-attacks?	4,589057389	0,5318604
4	4. Does the ANASTACIA framework provide clear seal and report about the cybersecurity status?	4,267320844	0,7506086
5	5. Would you like to have ANASTACIA framework to protect your cybersecurity in your workspace or home?	4,106749926	0,9520963

| Number of Questions | GENERAL INTERVIEW | | |