Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

SMESEC

Protecting Small and Medium-sized Enterprises digital technology through an innovative cyber-SECurity framework

D4.3 Preliminary Integration report on Smart City pilot

Document Identification			
Status	Final	**Due Date**	30/11/2018
Version	1.1	**Submission Date**	30/11/2018

Related WP	WP4	**Document Reference**	D4.3
Related Deliverable(s)	D2.1, D3.1	**Dissemination Level (*)**	PU
Lead Organization	University of Patras (UOP)	**Lead Author**	Kostas Lampropoulos
Contributors	Kostas Lampropoulos (UOP) Apostolos Fournaris (UOP)	**Reviewers**	Pablo Barrientos (Atos) José Fran. Ruíz (Atos)

Keywords:
Smart cities, security, SME, start-ups

This document is issued within the frame and for the purpose of the SMESEC project. This project has received funding from the European Union's Horizon2020 Framework Programme H2020-DS-SC7-2016 under Grant Agreement No. 740787 and supported by Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number 17.00067. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission.

Document Information

List of Contributors	
Name	**Partner**
Kostas Lampropoulos	UOP
Apostolos Fournaris	UOP
Pablo Barrientos	Atos
Manos Athanatos	FORTH
Fady Copty	IBM
Abbas Ahmad	EGM
Jose Fran. Ruíz	Atos

Document History			
Version	**Date**	**Change editors**	**Changes**
0.1	11/10/2018	Jose Fran. Ruiz (Atos)	Table of contents template for all use case partners
0.2	18/11/2018	Kostas Lampropoulos (UOP)	First draft of the document
0.3	26/11/2018	Kostas Lampropoulos (UOP)	Second draft after QA1
0.4	29/11/2018	José Fran. Ruíz (Atos)	Third draft after QA2
1.0	30/11/2018	Kostas Lampropoulos (UOP)	FINAL VERSION TO BE SUBMITTED
1.1	30/11/2018	ATOS	Quality review and Submission

Quality Control		
Role	**Who (Partner short name)**	**Approval Date**
Deliverable leader	Kostas Lampropoulos (UOP)	30/11/2018
Technical manager	Jose Fran. Ruíz (Atos)	30/11/2018
Quality manager	Rosana Valle Soriano (Atos)	30/11/2018
Project Manager	Jose Fran. Ruíz (Atos)	30/11/2018

Table of Contents

List of Tables

List of Figures

List of Acronyms

Abbreviation / acronym	Description
D4.3	Deliverable number 3 belonging to WP 4
EC	European Commission
EU	European Union
GDPR	General Data Protection Regulation
SIEM	Security Information and Event Manager
SME	Small-Medium Enterprise
TaaS	Technology as a Service
VM	Virtual Machine
WP	Work Package

Executive Summary

This deliverable provides an overview of the efforts carried out during the first 18 months of the project to deploy the SMESEC Framework in the second pilot for smart cities. Even though this work is still ongoing and various tools are still being configured and/or integrated, early results depict a clear improvement of the overall security status for the smart city platform of this pilot (sense.city).

With the general security requirements identified in deliverable D2.1, this deliverable initially presents the smart city use case and then provides a detailed description of the sense.city platform and the SMESEC tools which have been selected to protect its components. The document continues with the integration process of the selected tools under a single unified security framework (SMESEC framework) and how this framework is implemented in the smart city pilot. Apart from the technical aspects, deliverable 4.3 also analyses user training and awareness methods to be followed in order to include the human factor in the overall security planning of the sense.city platform. Finally, current and possible future business improvements are also examined to allow a wider evaluation of the impact that the adaptation of SMESEC can have in SMEs working on smart cities' services.

To conclude, this document describes the status at M18 of the work foreseen in the project for pilot II. The final integration of the SMESEC Framework in sence.city platform will be presented in deliverable D4.4 at month 24 and a final and complete evaluation at the end of the project.

1 Introduction

1.1 Purpose of the document

Deliverable 4.3 presents the development of Pilot II (Smart city) at month 18 of the SMESEC project. The document begins with a description of the sense.city service and its platform's architecture and then describes the proposed security solution, the integration process of the SMESEC framework and the security tools selected for this pilot. All selected tools are presented in detail along with their contribution to the pilot, their implementation status and if they are integrated with any other tools. An initial testing planning and some early results are also provided, and the document concludes with the steps to be followed in the upcoming months (up to month 24) to have a final working security framework adjusted to the SMESEC smart city pilot.

1.2 Relation to other project work

Sense.city is one of SMESEC project's pilots selected to evaluate the projects proposed security framework. To successfully complete this task, two milestones have been identified, the first at month 18 and the second 24. For the first milestone the project must have successfully evaluated the pilot's security needs and requirements (WP2), build an initial version of the framework (WP3) and run the initial tests on the pilot's systems (WP4). Then the project must collect the results, analyse any identified issues or problems and make the necessary adjustments and modifications (WP5) in order to reach the second milestone in month 24 where the initial version of the SMESEC Framework for smart city services must be delivered. These two integrations will be reported in D4.3 and D4.4 respectively. At the end of these phases SMESEC must be able to provide a framework capable of supporting the security needs of companies with products focused on smart cities.

1.3 Structure of the document

This document is structured in six major chapters, whose contents are the following:

Chapter 1 introduces the document, presents relation to other documents and describes the structure of its content.

Chapter 2 describes the updates of the architecture for the pilot presented in the deliverable D2.1.

Chapter 3 briefly presents the scope of the SMESEC's second pilot for smart cities, the architecture of the sense.city service and its needs for security enhancements to be supported by the SMESEC framework.

Chapter 4 shows the status of the pilot deployment until M18, trying to describe the main achievements made up to this date.

Chapter 5 identifies pending actions to be completed in order to validate the pilot and the expected functionalities and finally,

Chapter 6 recaps the main contents of the document and concludes the document.

2 Update of requirements and needs

The overall architecture of sense.city platform (Figure 1) is presented in detail in D2.1. All the necessary components are hosted in UOP's private cloud.

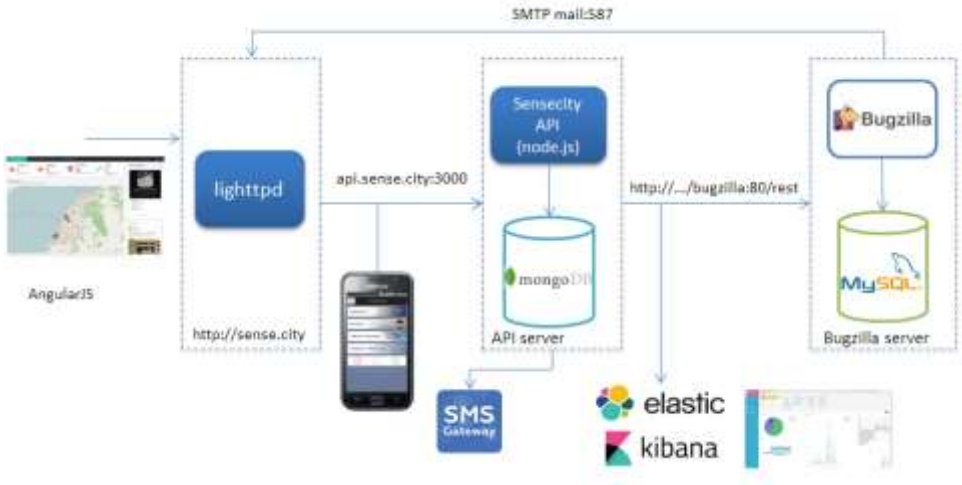


Figure 1. Sense.city architecture.

The integration of the SMESEC framework in sense.city pilot required the addition of extra VMs in UOPs private cloud to host SMESEC's security tools like honeypots, Antimalware server, SIEM-agent etc.

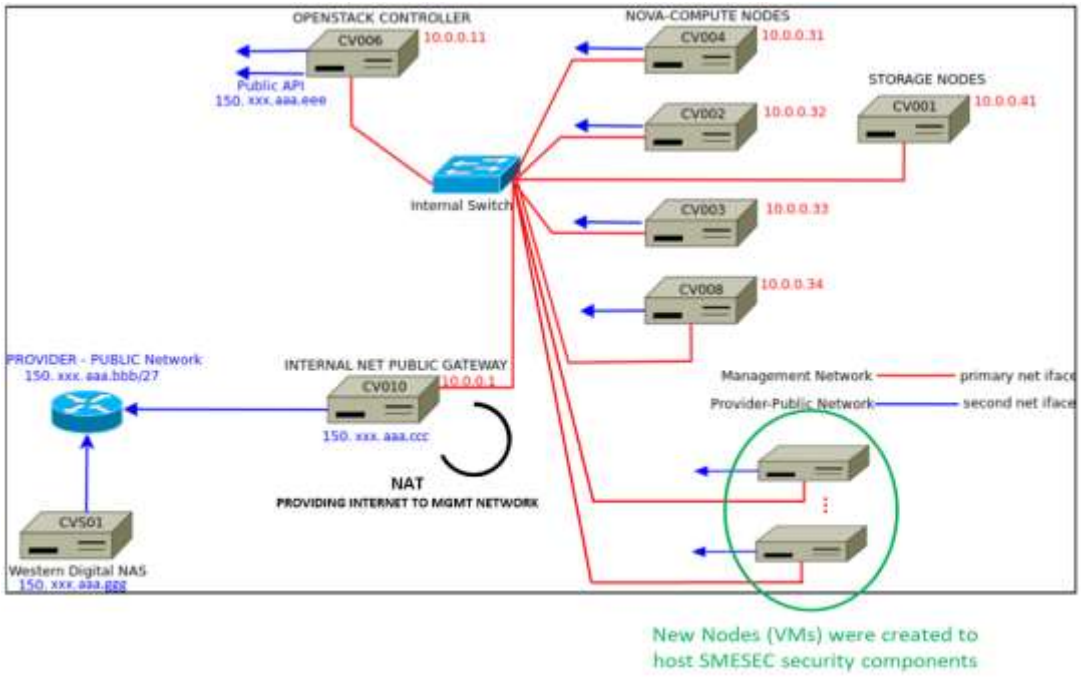


Figure 2. Updated Sense.city platform architecture to support SMESEC framework.

Apart from the new security requirement for protecting the additional VMs which host the SMESEC tools, the rest of the pilot's requirements are the same as presented in D2.1. The updated list of sense.city's components that must be protected is presented in the following table (Table 1).

Alias	**Description**
OSCONTROLLER (Figure 2, CV006)	Sense.City OpenStack infrastructure is open to the public internet. That is, the OpenStack Controller is accessible since one NIC is connected to the University of Patras core network.
OSNODE (Figure 2, CV002-4, CV008)	Each node is accessible since one NIC is connected to the University of Patras core network.
New Nodes (Figure 2) ***(updated)***	VMs for installing SMESEC tools and components.
NAMHOST (Figure 1)	The Web Server (bare metal) which hosts Sense.City frontend
LIGHTTPD (Figure 1)	The lighttpd server (located at NAMHOST) which hosts other websites along with the Sense.City frontend
SCWEB (Figure 1)	The front-end web application
NAMMS (Figure 1)	The mail server (NAMMS) hosted together with the Sense.City Web Server (NAMHOST)
SCAPIHOST (Figure 1)	The Sense.City API server (Virtual Machine)
SCAPI (Figure 1)	The Sense.City API service
SCAPIMDB (Figure 1)	The API service MongoDB
BUGZILLAHOST (Figure 1)	The server that hosts the Bugzilla service (Virtual Machine)
BUGZILLA (Figure 1)	The bugzilla service hosted on Apache
BUGZILLADB (Figure 1)	The MySQL server hosted at BUGZILLAHOST

3 Description of the use case

The use case selected for smart cities pilot is the sense.city service. Sense.city is a service that enables citizens to inform their fellow citizens and the municipality about problems and incidents that occur in their city. The main features of sense.city are:

- **What is happening in the city:** Using their own communication devices, citizens can inform their fellow citizens about what is happening in the city or report to the municipality problems and incidents they see.
- **Urban participation:** Citizens actively participate in the decision making and solve problems concerning their life in the city. They now can help urban development and build better relationships with the administration services.
- **Co-creativity:** The sense.city platform provides the means to activate citizens and calls for collective thinking and actions for both citizens and administration services.

The service is offered through a web and a mobile application. The mobile app is mainly used to report errors to the public administration, while the web app can be used for reporting problems, for overviewing information over one or more cities, and finally for accessing the administrator's page where public servants can address reported issues for their municipality (issue management backend). Figure 3 depicts the mobile and web applications of sense.city.

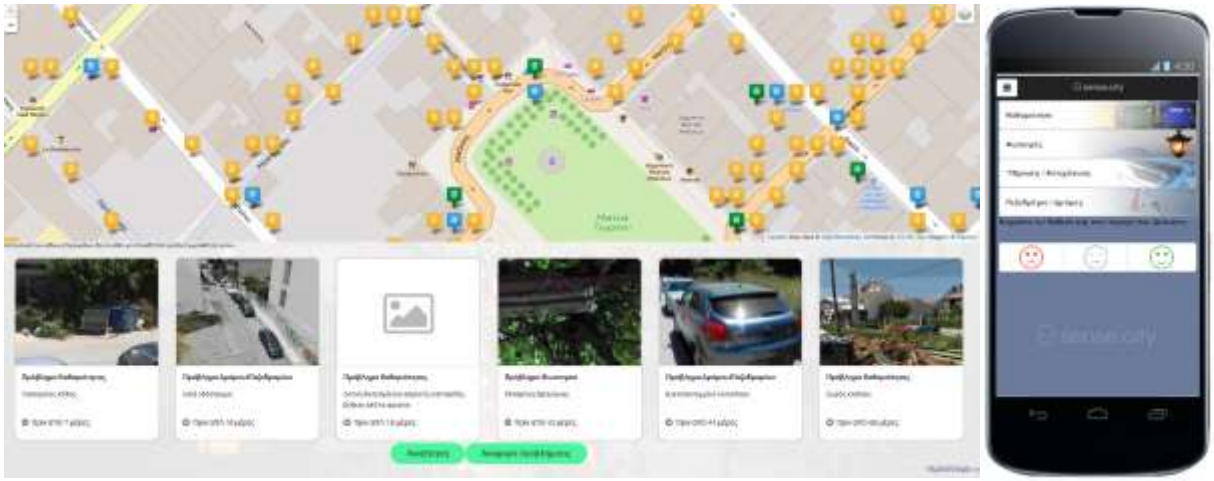
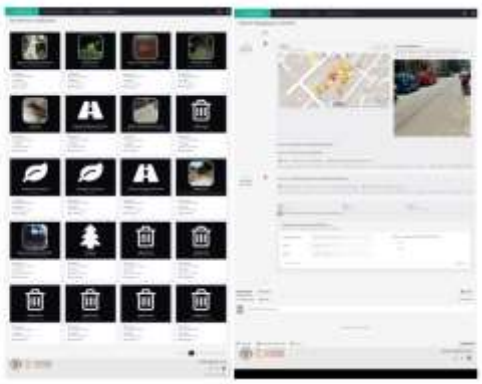


Figure 3. Sense.city web and mobile app.

The sense.city service allows users to report problems to their local administration and at the same time provides to the local administration an efficient way to a) collect problems of the city, b) assign them to the appropriate departments and c) monitor their status. The following figures demonstrate how the sense.city service is used.

1. A concerned citizen wants to report a problem to his municipality. For the sake of this document let's assume that a lamppost is broken outside his house. Until now, the only way to report this, was to go to the local municipality offices, search for the department responsible for the public lighting and fill a form with his problem. With sense.city he doesn't have to do any of the above time-consuming procedures. By simply opening his sense.city app he can directly press a button to report his problem. If he doesn't want to install the mobile app, he/she can do the whole process through his web browser on the https://sense.city website.



2. During the reporting process the citizen is asked to select the category that better matches his problem (e.g. road issues, debris, damaged bins etc.), point the location of the problem on a map and give a small description about the issue. If none of the predefined categories matches his problem, the citizen can name his own category. Furthermore the user has the ability to upload a picture and also give his mobile number to receive notifications about the progress of the solution to his problem. Of course a problem can also be submitted anonymously without the use of a mobile number.

3. After submission, the problem is recorded to the account of the local administration (municipality) and is assigned to the appropriate department. Employees of the municipality have their own credentials and can login to the management console to see and solve any reported issues.

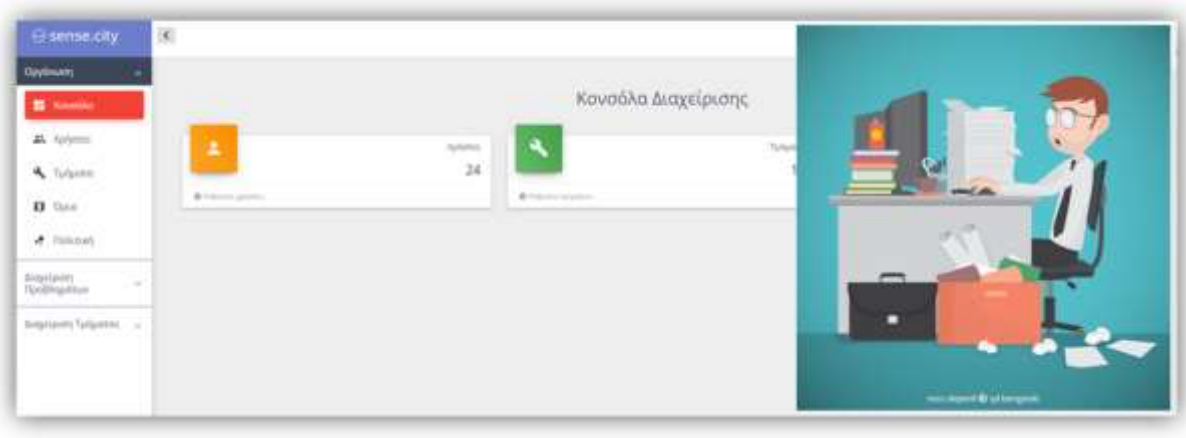


Figure 4. Sense.city service.

3.1 Architecture and design

The system to be protected as well as its security requirements are presented in detail in D2.1. The sense.city platform is composed by the web frontend (lighttpd), its mobile app and its backend which communicates with the frontend and the mobile through the api.sense.city interface. The backend includes the webserver, the DBs and a set of tools for management and monitoring. Figure 5 depicts the overall architecture and the various users of sense.city.

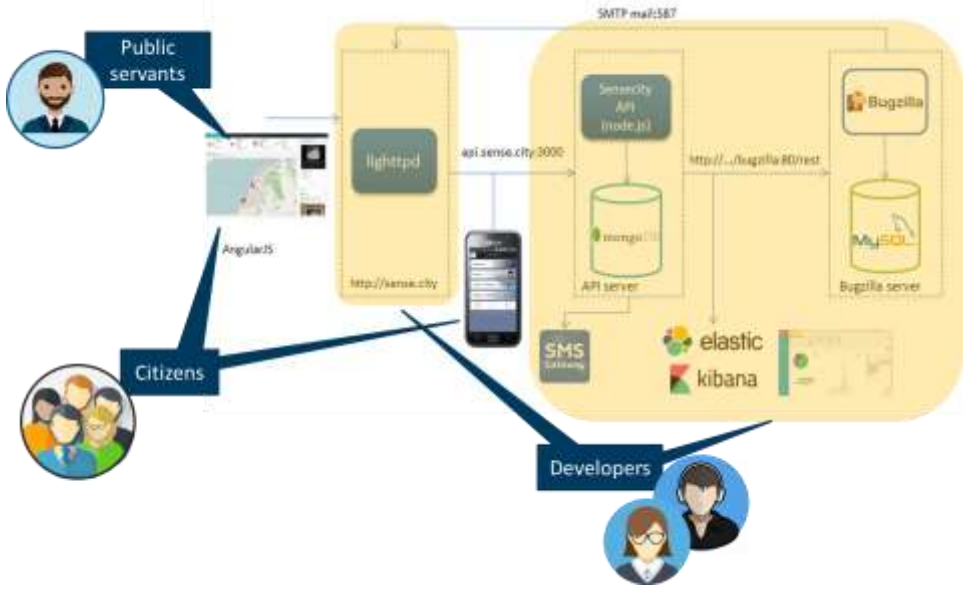


Figure 5. Sense.city architecture, components and users

From the set of proposed security tools inside the context of SMESEC, the UOP team selected to adopt the Antimalware-Antivirus from Bitdefender, the XL-SIEM solution from ATOS, the solution for intrusion detection from FORTH, the code security tool from IBM, the test as a Service from EGM and finally the cloud security solution from FORTH. The platform already had installed firewalls in specific nodes that handle incoming traffic.

Contrary to the usual practice that is followed in similar research/EU projects, for this pilot UOP decided to not only evaluate the SMESEC framework on a testing site but benefit from the security solutions offered inside the project's context and implement some of the proposed tools in its operational nodes. Since sense.city is a live product used by thousands of citizens in Greece, this decision is implemented very carefully and after following a specific set of steps. These steps ensure that the sense.city service is protected from any kind of anomalies or problems caused due to failed installations or misconfigurations. These steps are the following.

a) Additional VMs are created in UOP's private cloud to host newly added security tools.
b) All security tools will be tested in VMs that do not host any kind of sense.city's components.
c) Any deployment of security tools to sense.city's operational VMs will only be made after they have been successfully tested in step b).

3.2 Scenarios of application

The decision to use SMESEC framework (or part of it) in the sense.city platform is the result of a general plan to make use of the high-quality tools provided within SMESEC context and protect the overall UOP cloud infrastructure. This approach affects all the UOP lab's services, tools and applications hosted in its private cloud. Such services include network management tools, various websites, portals for services orchestrations etc. all designed and implemented by the NAM group of UOP. Even though it is not possible to make an extensive analysis of the requirements of all these applications and design a better security architecture for the overall infrastructure, some of SMESEC components like intrusion detection, antimalware etc. can make a significant impact in the general protection of the lab's assets.

At the same time the nature of the sense.city service requires public servants to log in the system and manage reported issues. This means that the platform is exposed to connections from public PCs (used by public servants). The SMESEC framework provides valuable support in this matter by fortifying the UOP cloud infrastructure with various security tools e.g. antimalware. Furthermore, it allows the UOP team to continue designing services that can be operated by external users using their own devices. This capability directly supports our goal for developing open services and tools.

Finally, with the final SMESEC framework UOP can take advantage of its training courses to promote training and the use of best practices not only for its personnel but also for the public servants working for the municipality of Patras. UOP plans to design a scenario where an information day will be organized for the public servants of the municipality of Patras, to discuss on general cybersecurity matters, demonstrate the capabilities of SMESEC framework and also showcase how the training courses of the project can help people increase their cybersecurity skills.

3.3 Cybersecurity threats and impact

Being a typical online service developed with well know tools and frameworks (javascript, node.js mongoDB etc) sense.city is facing a large set of cybersecurity threats. Furthermore, the fact that the service is currently hosted at the private cloud of a University makes the attack surface wider, due to the fact that university networks are not designed to have strict security policies like companies. University networks generally permit open access to promote research and innovation.

Within the context of SMESEC the sense.city team and platform went through various vulnerability assessments and tests to evaluate not only the cyber-security status of the system but also the awareness and preparation of the team to handle an incident. Some of the results from these assessments are described below.

1. **CYSFAM model.**
   The UOP team was asked to complete the CYSFAM cyber maturity model created by the UU. Sense.city's score with this model was the lowest among all four pilots. The results are depicted below.
2. **Independent security analysis from FHNW.**

   Apart from SMESEC's planned activities for evaluating the security status of pilots, FHNW cybersecurity experts also performed an independent security check for sense.city platform and provided to its team a 162 page report (Figure 7).

Cybersecurity Maturity Model	A	B	C	D	E
Organisational and Technical					
Server Protection	0/4	1/4	0/4	0/4	
End-user Control	0/4	1/4	0/4	0/3	
Social Engineering Controls	0/2	0/2	0/2	0/1	
Network Security	0/3	1/3	0/3	0/3	
Web Application Security	1/2	0/2	0/2	0/2	
Organisational					
Cybersecurity Incident Management	0/4	0/4	0/4	0/4	
Cybersecuriy Awareness	0/4	0/4	0/4	0/3	0/1
Cryptography	0/4	0/4	0/4	0/4	
Cybersecurity Governance	0/4	0/4	0/4	0/2	
Mobile Security	0/4	0/4	0/4	0/4	
Vulnerability Management	0/2	0/2	0/2	0/2	

Figure 6. Sense.city CYSFAM model score

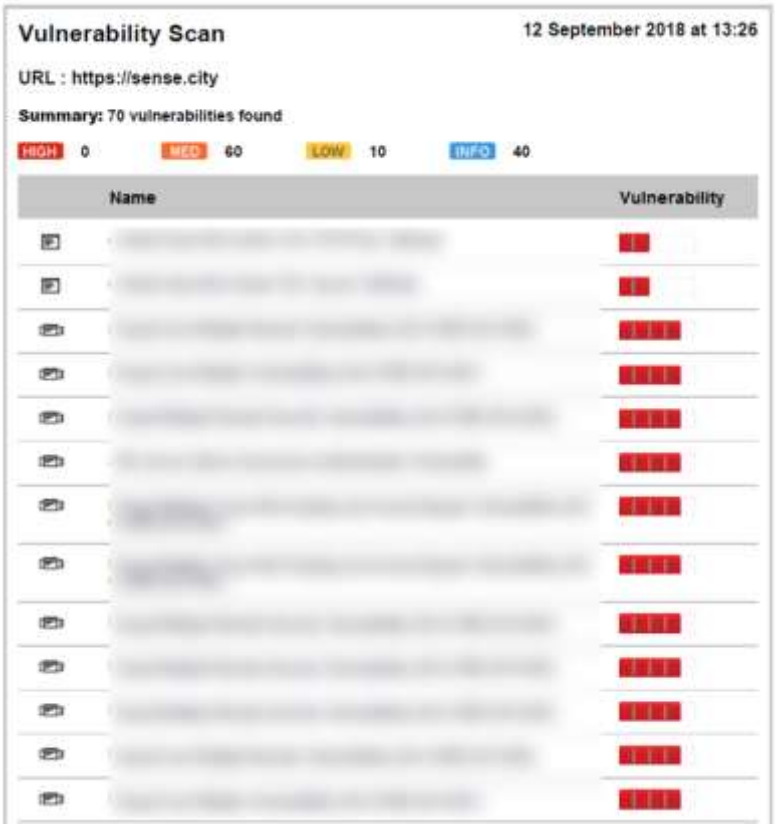


Figure 7. FHNW report on Sense.city security status

The above figures clearly depict that sense.city joined the SMESEC project with a clear weakness in the area of cybersecurity. This weakness was not originating only from the unprotected components of the system but also from the lack of security training of the personnel, and the lack of internal security procedures and planning to handle and mitigate attacks. The impact of a successful "targeted" or "untargeted" attack could have severe impact in the functionality of the system and associated services.

3.4 Cybersecurity training and awareness status

One of the most critical issues that were identified within the context of SMESEC for the sense.city pilot was the low levels of cybersecurity training and awareness status. The CYSFAM model proved that the team did not have any established procedures to train their developers and people involved in cybersecurity. CERT teams and security experts were not part of the team and the approach followed was "we all do a bit of some security".

The lab in UOP that is developing sense.city is doing research and has a good experience in other aspects of cyber security, namely hardware security and identity management. This expertise however is not sufficient to secure a market product (like sense.city) and the infrastructure behind it.

3.5 Business opportunity

The generally low scores in cybersecurity protection for sense.city originated from the fact that the service was until recently in beta phase and development was focused on resolving functionality issues. The services were provided in a "free - as it is" agreement with basic security settings.

The low security status however, has an immediate impact to business growth and market. To be able to give sense.city as a market product it is essential for UOP to pay special attention to security as well. SMESEC project is an excellent opportunity to guide the UOP team in the security assessment, planning and implementation that will allow sense.city to reach a more mature business level.

On the other hand, sense.city market is not only limited to municipalities. Various business opportunities have been identified since the launch of the service. In particular, the deployment of IoT devices to be managed and monitored over a city-wide area is one of the next features that this platform will implement. This kind of use case cannot be supported though, unless the security of the system and the integrity of the data are ensured. SMESEC framework can provide solutions for IoT systems and is an excellent opportunity for UOP to increase their portfolio on smart cities' solutions.

Finally, another business aspect that UOP is also examining is to sell/rent the service to a bigger company which will integrate it to its systems, thus expanding its portfolio in social networks, public administrations, smart cities etc. The security is a crucial matter to be addressed before moving forward with such a business deal. It must be ensured that both systems (UOP – big company) are protected against each other, do not introduce security vulnerabilities and their personnel is trained to address the security challenges of the integration. The SMESEC project can provide to UOP high quality tools and solutions but also proper user training in order to support such kinds of business opportunities.

4 Integration of the SMESEC Framework

As mentioned above, the sense.city platform aims to benefit from the SMESEC framework and its security tools to the maximum extent possible. For all proposed solutions in the project, UOP is performing an evaluation testing and at a later phase will select those that will be adopted to protect the actual production system.

4.1 SMESEC-enhanced business pilot

At M18 an initial architecture to protect the smart city pilot has already been defined. For this architecture UOP provided details on sense.city's architecture, components and requirements, and the SMESEC's security providers proposed how their solutions can be adopted to support the specific use case. Among all the security tools offered within the SMESEC context, UOP has selected the following.

Pilot II: adopted tools		
Tools	Provider / partner	Purpose
Security tests	EGM	Integrity checking of the infrastructure and service
Code analysis	IBM	Code analysis for javascript
EWIS - honeypot	Forth	Intrusion detection for the sense.city system
XL-SIEM	ATOS	Events log management and anomaly detection
GravityZone	Bitdefender	Antivirus-Antimalware solution
Cloud Security	FORTH	Security between VMs on the same cloud

Table 1. Adopted tools in Pilot II (Smart Cities)

An abstract architecture on how SMESEC's security tools are inserted on sense.city's architecture is presented in the following figure.

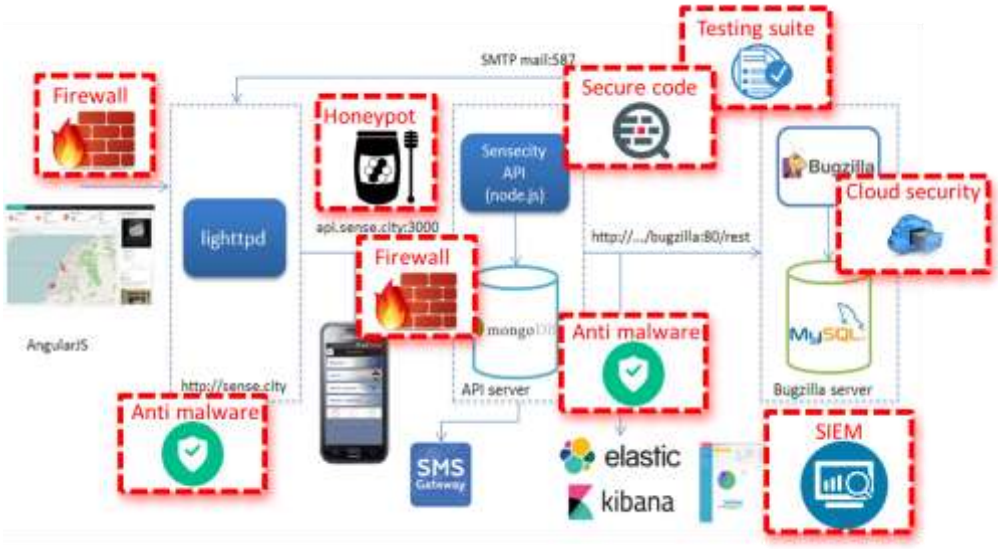


Figure 8. SMESEC security tools adopted in sense.city architecture

At M18 the antimalware, EWIS and SIEM solutions have been completely deployed, integrated and are fully operational, while the solutions for testing suite, code analysis and cloud security are still under deployment. The status of the three solutions which are still under development is the following.

EGM Testing suite: Due to the fact that the sense.city system was undergoing various changes (migration to a new cloud, redesign of services and components) and the testing suite needed a stable system architecture to define its models, the implementation of this solution was delayed. An initial plan for this solution is described below in section 4.2.

FORTH cloud security: Cloud security is a key aspect for the sense.city system, since it is hosted in a private cloud which runs various services for the needs of the UOP team (research apps, websites etc.). Currently the solution for cloud security created by FORTH is not compatible with the cloud infrastructure in UOP and cannot be adopted as it is. A plan to evaluate if and how FORTH's security solution may be installed in UOP's private cloud is also presented below in section 4.2.

IBM code analysis: Initially, IBM's code analysis toolkit did not include solutions for javascript, and UOP could not benefit from it. However, at M17, IBM introduced a new tool (javascript fuzzing) that was under development and could possibly give some early results for sense.city. UOP has shared sense.city's code with IBM and the results will be produced in the next months.

Finally, considering the firewall solutions for the sense.city platform, UOP was already using the Openstack (Linux) firewall before joining the SMESEC project. Since these firewalls are part of the production part of the sense.city platform, UOP has decided to avoid making any kind of modification or integration with the rest of the SMESEC framework. Once the SMESEC tools are tested, the integration of sense.city's firewall with the overall SMESEC framework will be examined.

4.2 Process and tools integrated

The following sections describe the SMESEC tools that have been selected for pilot II, how these tools can support the sense.city platform and how they have been integrated so far inside UOP's cloud infrastructure.

4.2.1 ATOS – XL SIEM

Atos XL-SIEM tool provides the capabilities of a Security Information and Event Management (SIEM) solution with the advantage of being able of handling large volumes of data and raise security alerts. The tool offers a) Real-time collection and analysis of security events b) Prioritization, filtering and normalization of the data gathered from different sources and c) Consolidation and correlation of the security events to carry out a risk assessment and generation of alarms and reports. Furthermore, it provides a multilevel web-based visualization framework for security monitoring and incident response.

Within the smart city pilot (Pilot II), XL-SIEM collects logs from other SMESEC tools and presents cybersecurity incidents that took place in near real-time. The XL-SIEM can be installed as a standalone service inside a company, however for the sense.city use case, UOP team decided to directly use the XL-SIEM in ATOS' cloud. With this decision, information regarding security events are sent from the sense.city platform to an external party (ATOS) but it was necessary to avoid the installation and management of a highly demanding system in terms of resources and personnel. To be able to connect with ATOS' cloud (and the XL-SIEM), the UOP team created a new VM in its cloud

and installed a "cyber-agent" that collects logs from the rest of the pilot's installed security components. This agent forwards all collected logs to the XL-SIEM hosted in ATOS' cloud. On the other hand, ATOS' team created the necessary accounts and credentials in its XL-SIEM to provide to sense.city its own dashboard. Apart from the VM for the XL-SIEM, the UOP team also created additional VMs to host other security components which are described in the following sections. For the clarity of the architecture though we present here all the newly added VMs and the communication between them (Figure 9).

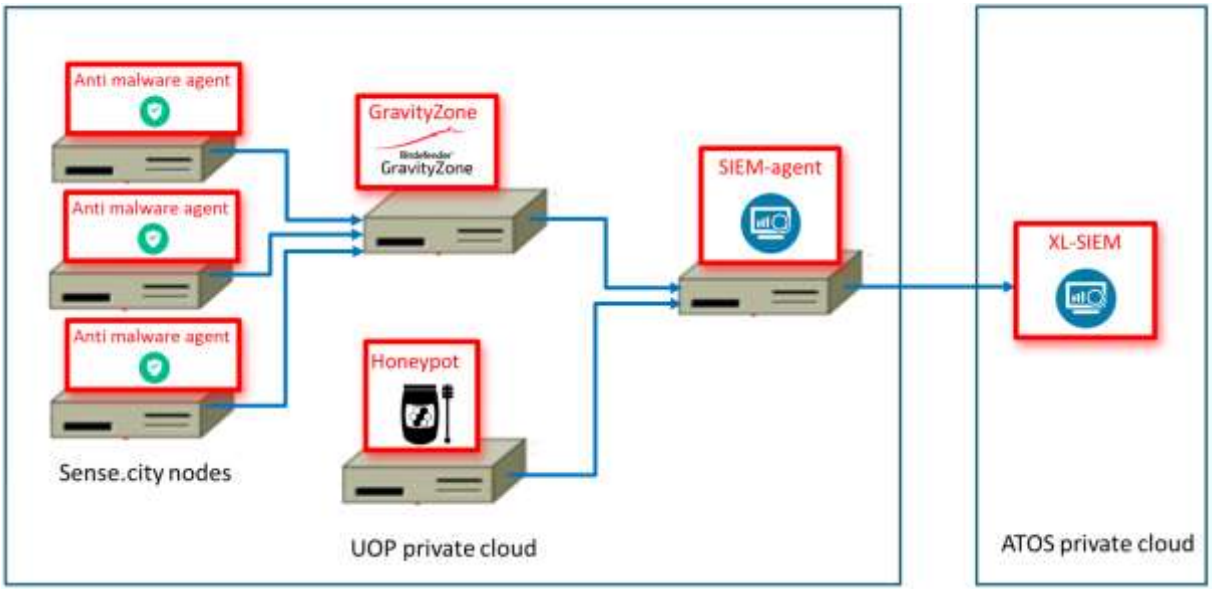


Figure 9. New VMs created in UOP cloud and XL-SIEM communication with other components.

4.2.2 Bitdefender – Gravity Zone

Bitdefender GravityZone is a next-generation endpoint security suite that meets the requirements of a layered, adaptive endpoint protection platform by balancing security efficacy, manageability, and performance. It reduces the attack surface with a set of controls that protects against malicious websites and hardens the configuration of endpoints, offering web filtering, firewall with intrusion detection and prevention, device control and USB scanning, application whitelisting and data security.

The architecture of sense.city platform (private cloud, multiple VMs, third party software components/libraries) and the fact that some of its users (municipality employees) use public PC's to log in their accounts makes the need for a top-tier antimalware solution one of the highest priorities. To deploy the GravityZone in this pilot, the UOP team created a new VM which only hosts the GravityZone Control Center. With the Control Center in place, the next step was to deploy the "endpoint agents". An endpoint agent is software which protects the VM/PC/node that is installed and reports any kind of activity to the Control Center (Figure 10). At this point it must be noted that the VMs that the endpoints were installed, were not part of the sense.city production infrastructure to ensure that the system would not be affected from any kind of problem during the installation or testing of this tool.

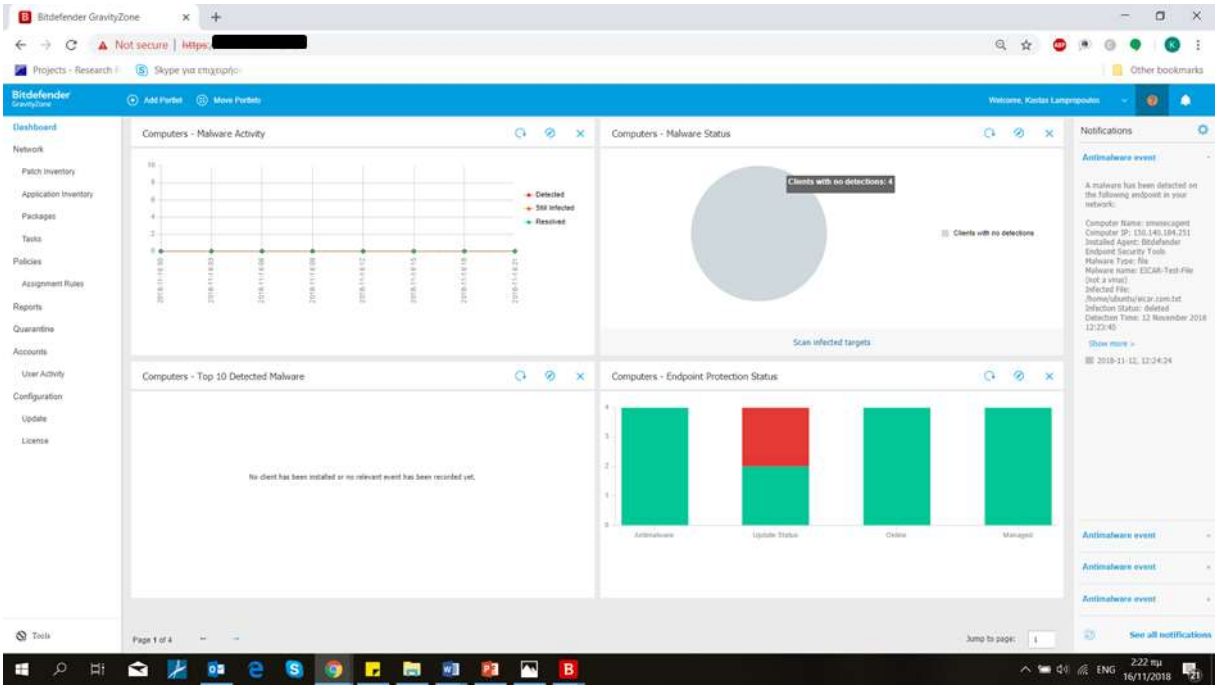


Figure 10. GravityZone Control Center for Pilot II (Smart cities)

4.2.3 FORTH – Early Warning Intrusion Detection System

In the area of intrusion protection, FORTH has developed, deployed, and maintains an Early Warning Intrusion Detection System (EWIS) based on honeypots. Honeypot sensors are installed in points inside a corporate network and are capable of collecting security information related to possible cyber-attacks in real time. The system is able to provide distributed network monitoring capabilities, to detect and centrally log all the "malicious" activity, and to provide real-time alerts. The results are provided through a web interface although a raw data access can be easily provided.

For the sense.city platform, FORTH installed a honeypot sensor inside the UOP private cloud. This component was installed on a separate VM which does not host anything else. The honeypot is configured through a web interface and is programmed to report information from attacks and intrusion attempts to a) FORTH's control center, located at FORTH's premises and b) the SIEM-agent installed in UOP's private cloud (as described in section 4.2.1). UOP can see all identified attacks identified by the honeypot from their dashboard in XL-SIEM. The honeypot interface in FORTH's premises is depicted in the following figure (Figure 11).

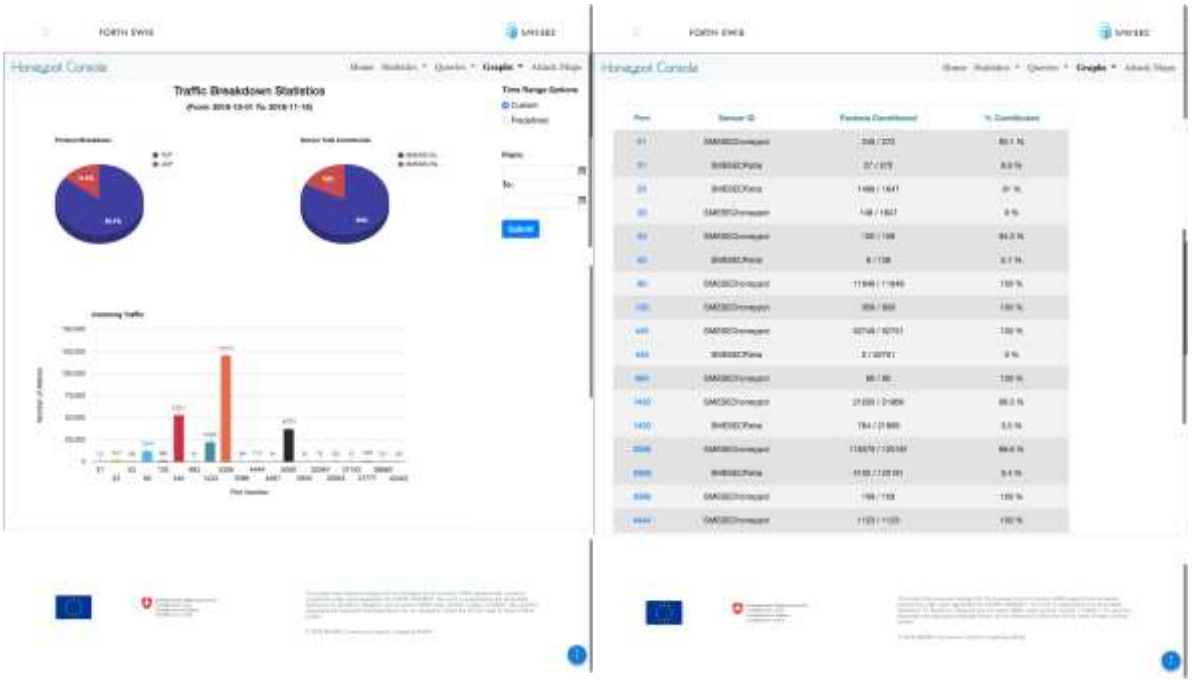


Figure 11 EWIS – honeypot interface for Pilot II (Smart Cities)

4.2.4 EGM – Test as a Service

The EGM TaaS is a web service that allow users to execute tests concerning security issues. Test suites need a prior ad-hoc accommodation to the use cases and after the test execution they pinpoint failing security requirements that can be fixed afterwards. The TaaS platform is able to integrate modular end-to-end test cases in order to provide automation to the testing phase. The test can be of various objectives: DDoS, security compliance, specification compliance and threat analysis.

As described above, due to various modifications in sense.city platform's infrastructure, the coordination with EGM to design the appropriate tests for this pilot was delayed. Currently the design for these tests is undergoing and are focusing on DDoS attacks and threat analysis. More specifically:

- A Distributed Denial of Service (DDoS) attack is an attempt to make an online service unavailable by overwhelming it with traffic from multiple sources. The TAAS infrastructure will deploy many back-end components (TaaS Back End, Figure 12) in order to overwhelm the sense.city infrastructure.
- Threat analysis assessment is the practice of determining the credibility and seriousness of a potential threat, as well as the probability that the threat will become a reality. On the sense.city platform, the TaaS performs simulated attacks related to well defined threads such as:
  - Man-in-the-middle (MitM) attack
  - Phishing and spear phishing attacks
  - Drive-by attack
  - Password attack
  - SQL injection attack
  - Cross-site scripting (XSS) attack
  - etc…

The TaaS integration goal is to execute automated test campaigns and provide feedback on the security aspect of end-to-end usage of the sense.city platform. Many access points are defined through REST APIs' and can be accessed by the TaaS platform as it supports virtually all testing languages.

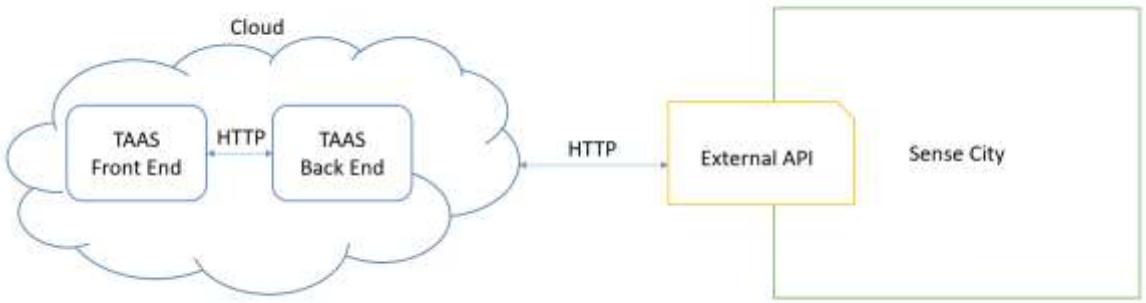


Figure 12 TAAS Integration with pilot II (Smart Cities)

4.2.5 FORTH – Cloud security

The cloud-based security tool proposed by FORTH runs in hypervisor level. It is an IDS tool running on top of Xen hypervisor that is able to monitor all inter- and intra- hypervisor traffic. This tool currently runs only over the XEN hypervisor but FORTH is studying how to port it to other hypervisor technologies as well. Since the current cloud components of Smart Cities pilot run over OpenStack, we cannot apply that specific tool in this use case. Finally, the popular Snort tool, as well as other free components are currently used to display real-time results via a web interface. FORTH plans to extend the proposed tool by a) adding GPU acceleration b) incorporating the results to the SMESEC framework c) Support for other hypervisors.

4.2.6 IBM – Code analysis: Javascript fuzzing tool

JavaScript is one of the most popular programming languages in the world. Only in GitHub there are more than 2.3 million unique projects written in JavaScript. Amongst other applications, JavaScript is used for server-side development using the NodeJS infrastructure. Nowadays, there are some static analysis tools that alert developers of bad, or potentially vulnerable places in their code. These tools mainly focus on better coding style, and trivial best practices for secure development. Currently, there is no solution for more complex vulnerability analysis of server-side JavaScript code. This is an important issue to address.

JSFuzz is a pluggable fuzzing infrastructure prototype that uplifts a state-of-the-art fuzzing paradigm (AFL) into a high-level, interpreted, dynamic, and weakly typed language (JS), with focus on server-side JavaScript code (NodeJS) to detect vulnerabilities, find attack vectors and exploits, and discover various bugs, ranging from simple OS command injections to more complex logical errors that can lead to dangerous exploits. In a collaboration between Patras University and IBM, both the JavaScript fuzzer, and Sense.City will be evaluated with joined effort to make Sense.City's code more secure and resistant to adversarial attacks in various aspects. In order to run the fuzzer, the files we would like to fuzz on and the types of vulnerabilities we were looking for have already been specified.

4.2.7 SMESEC security tools integration status for Pilot II

SMESEC framework is not just a collection of security tools but a unified solution where multiple cybersecurity products work together to a) offer a holistic view of the security status of the protected system and b) efficiently identify incidents and deal with them before they become critical. To this end, in the smart city pilot, there is an undergoing process of integrating/connecting the installed security SMESEC tools. Currently among all tools selected for the smart city use case, only the XL-SIEM, the EWIS and the GravityZone have been integrated. More specifically the ATOS' XL-SIEM is receiving logs from FORTH's EWIS and Bitdefender's GravityZone, presenting all cyber events in its dashboard. In the next months, SMESEC will examine the rest of the tools to identify additional integrations/connections between them that can further advance the overall protection of the sense.city pilot. We need to clarify at this point that all of SMESEC solutions will be integrated under the SMESEC unified framework. Finally, as described above, for the firewall solutions, UOP team is currently using the Openstack (Linux) firewall. These firewalls have not yet been integrated to the overall sense.city pilot for the following reasons.

a) Firewalls are part of the production infrastructure. Currently UOP team does not want to mix any parts of the production service with components created and installed for the testing of the SMESEC framework. This process will carefully take place once the SMESEC framework is fully tested.
b) Firewalls are not part of the security tools provided by the SMESEC partners. At this phase of the project, the work mainly focuses on integrating tools by SMESEC partners. External solutions and their addition to the framework will be investigated at a later stage.

4.3 Testing

The deployment of the selected security components for the sense.city platform is at a different stage for each tool. Some tools are fully functional (complete installation and configuration), and others are not yet fully adopted, or still being evaluated (e.g. cloud security). At M18 the tools that have been fully integrated with the pilot's infrastructure and been tested are the following.

4.3.1 FORTH – EWIS (Early Warning Intrusion Detection System)

For this tool, the test focused on an attack on UOP's cloud. FORTH team deployed this attack in order to check whether the installed honeypot will be able to identify it. The attack was an SQL version scan and a follow up SQL unauthorised login attempt. The attack was successfully identified by the honeypot and reported as a cybersecurity event. Figure 13 depicts a screenshot from FORTH's team performing the attack and Figure 14 depicts the XL-SIEM dashboard where the event was presented after it was reported by the Honeypot installed in UOP's cloud infrastructure.

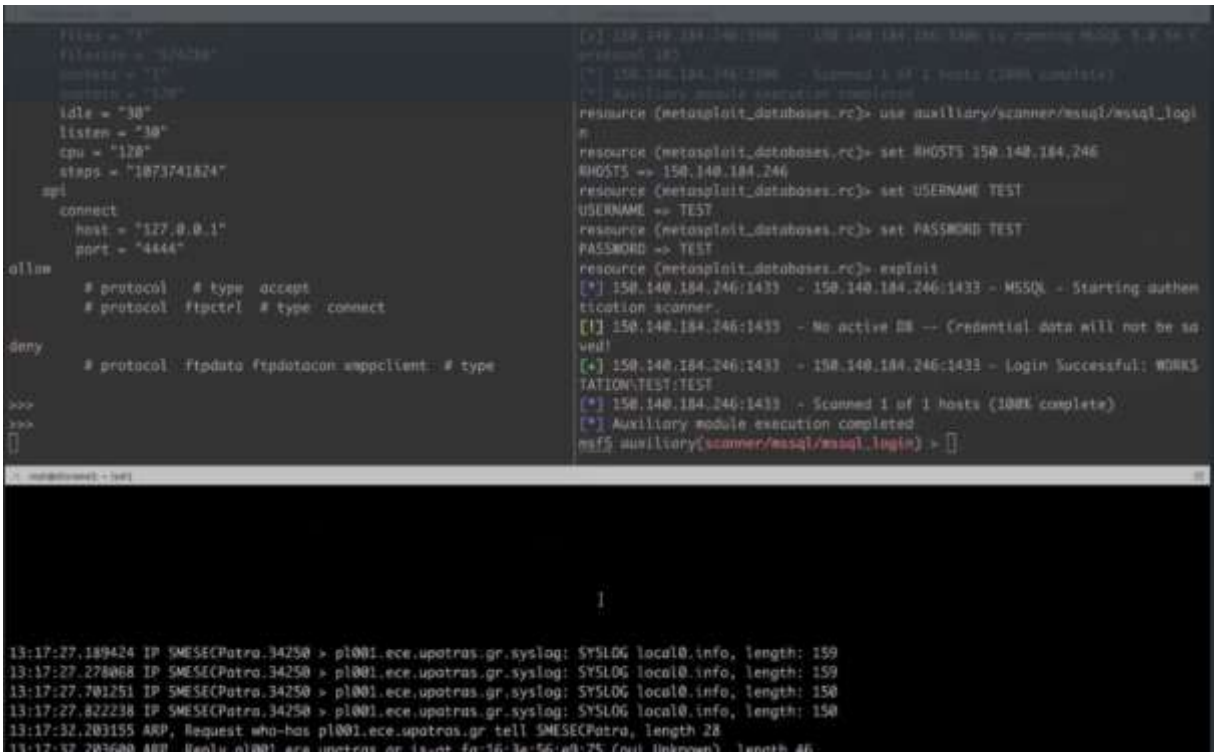


Figure 13 SQL attack on UOP cloud

Figure 14 Honeypot successfully identified and reported the SQL attack

4.3.2 Bitdefender – GravityZone

Bitdefender's antimalware solution was tested with the following scenario. UOP team was instructed by Bitdefender's experts to download a dummy malware file to one of the VMs that are protected by the GravityZone. This file is not an actual malware but has been created to simulate a malware for testing purposes.

Bitdefender's security solution successfully identified and blocked the malware file. A log event was created and presented to GravityZone dashboard as depicted below (Figure 15).

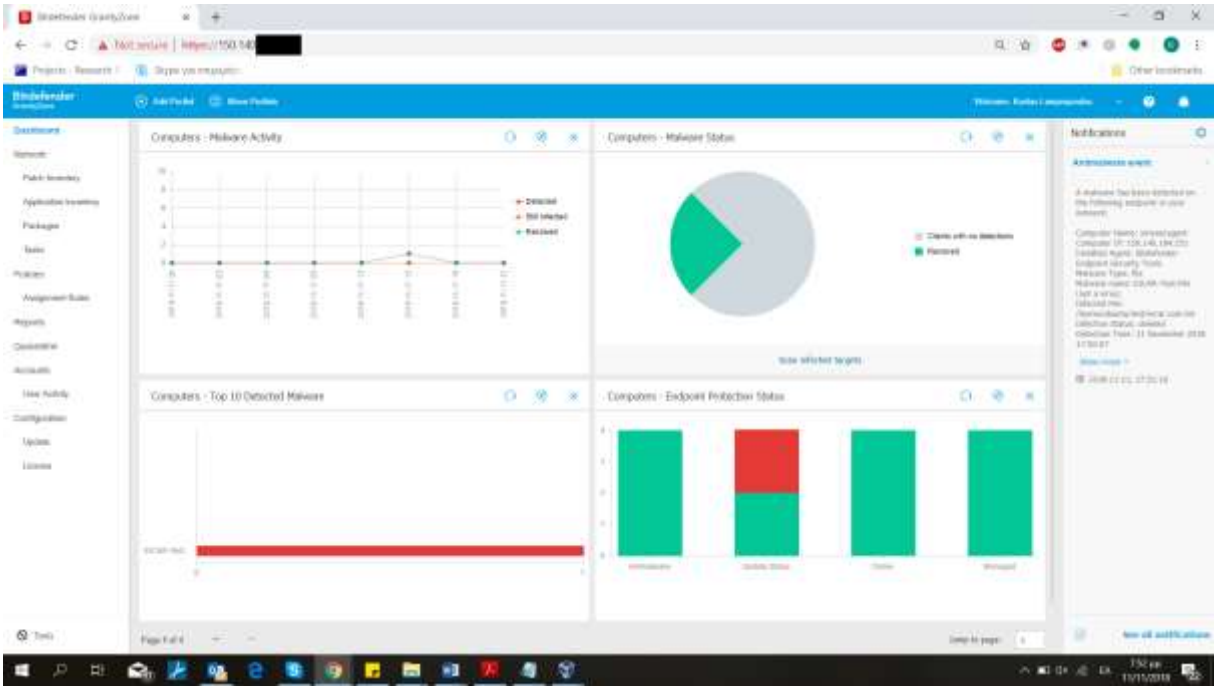


Figure 15 GravityZone identified and blocked a malware in UOP cloud.

4.3.3 ATOS – XL SIEM

For ATOS – XL SIEM testing, efforts were concentrated on checking the connection with the rest of security components installed in UOP's infrastructure. As mentioned above until M18, FORTH EWIS and Bitdefender's GravityZone are successfully communicating with the SIEM-agent installed in UOP's private cloud. Also, the SIEM-agent is also communicating successfully with the XL-SIEM installed in ATOS' cloud.

4.3.4 Rest of tools

In sections 4.1 and 4.2 we presented the integration status of the remaining three security tools, namely the EGM TaaS, FORTH's cloud security and IBMs code analysis. Since these tools are not yet fully adopted in the sense.city pilot, their testing phase has not yet started. Finally, the firewall presented in the architecture is the Openstack linux firewall activated in specific VMs. This firewall has been successfully used for a long time now and additional testing for these components is not needed.

4.4 Initial feedback

At M18 the sense.city pilot has three SMESEC components fully installed and integrated and three more security solutions that are still under the deployment phase. Even though we cannot fully measure the exact impact of the project to the sense.city platform, some early opinions and results can be provided.

4.4.1 FORTH – EWIS

The process to deploy the EWIS solution in UOP cloud was not too difficult, however it requires the provision of at least one VM to act as a honeypot and be used only for this purpose (cannot host any other services). FORTH suggested to deploy more than one honeypot, but currently in the UOP cloud there is a limitation of resources. The configuration and management of this honeypot was done remotely from the FORTH's team. Despite the fact that this action speed up the process of deployment, in the upcoming months it is required to train the UOP team to manage this tool by themselves.

4.4.2 Bitdefender – Gravity Zone

The deployment of the Gravity Zone Control Center was also not difficult. What was confusing is how to create the agents that must be deployed inside the UOP cloud to protect selected VMs. Once Bitdefender experts explained the process and provided with the proper documentation the UOP team was able to successfully deploy the agents in specific VMs in its cloud. The interface of GravityZone is user friendly and easy to understand and the connection with the local SIEM-agent was also very easy (just enabled syslog and indicated the IP and port of the receiving end). Tests to this tool were successful and UOP is looking to extend the use of this product to its production VMs.

4.4.3 ATOS – XL SIEM

The ability to use the XL-SIEM without having to install it in UOP's cloud was a major plus for adopting this tool. Some configuration was needed from the UOP team to be able to activate the local agent (installed inside UOP's cloud) and also connect it with the rest of SMESEC tools. We feel that this step is not too difficult for developers/IT personnel. The ability to see all attacks targeting your system in one dashboard is a very good start to plan your defenses properly. The following things will be further investigated in the upcoming months.

a) If and how can a company integrate/connect external tools to their XL-SIEM account on its own.
b) Can XL-SIEM be programmed to react on specific events. e.g. if this event shows up, activate this service or deploy this rule.

4.4.4 EGM – TaaS

Since this solution is still under deployment, it is still early to provide valuable feedback. However, based on the initial discussions and planning, UOP has identified that this tool can offer great

flexibility and test a large variety of architectures and attacks. In the upcoming months, the two partners will work together to make the best out of this collaboration for their companies and services.

4.4.5 FORTH – cloud security

As described above, currently this tool is not supported for UOP's cloud infrastructure. The solution however is of great interest for UOP and in the next months we will see if some updates may be compatible with sense.city.

4.4.6 IBM – code analysis

The javascript fuzzing tool is a new technology that has only been brought to SMESEC project in M17. Results from this tool are still pending, thus no accurate feedback can be provided at this point.

5 Next steps

The integration of SMESEC framework in pilot II at M18 has some tools completely installed and functional and some other solutions in the deployment state. The overall process has been progressed smoothly and remaining steps do not indicate that there will be any big risks or anomalies towards the integration of the full SMESEC framework in sense.city platform.

5.1 Integration of business in the SMESEC Framework

As also mentioned in section 3 the business aspects for implementing SMESEC framework in this pilot mainly focus on the transition of the proposed service from a "free as-it-is" product to a market product that can protect its infrastructure, functionality and data from cyber-attacks. Already with the initial integration of high-quality security products, UOP team has a better view of the kind of attacks that target their systems and is protected from a large number of them (e.g. malware).

Another aspect of the business impact that SMESEC framework has in this pilot is that the UOP is looking to deploy SMESEC's solutions to the UOP private cloud which hosts various services and research applications. With this decision the UOP team's goal is to gain better protection to more than one services, have an overall view of its security status and be able to create more reliable research solutions that can promote the lab and also its research and possible business activities.

5.2 Training and awareness plan

Pilot II is unique compared to the rest of the pilots in the sense that the service has been created and is hosted by a University lab team. For this reason, some internal procedures do not directly match those of a company. For example, security CERT teams do not exist inside a university lab but are generally located at a highest level e.g. the network management center of the whole campus. At the same time, the security levels issued by the network management center do not follow strict policies (e.g. open ports, http access etc.) since each department may have its own unique requirements for network access. The definition of a training and awareness plan for University of Patras is not yet finalized but will generally focus around three groups of people.

a) Developers: The developers of sense.city platform have a good knowledge on security, but it is not among the highest priority tasks to perform security checks on their code. A strategy to promote secure code writing must be defined in SMESEC project. Emphasis should also be given to security and privacy by design techniques, as well as frameworks and solutions for GDPR compliance.
b) Infrastructure and service management: This team also has good knowledge in multiple security topics however since they do not operate a market product, or a company network their focus is more on service maintenance and delivery and less on security. For this team, we plan to present courses designed by the SMESEC partners that explain in detail various security issues and provide solutions through specific products. These courses will also provide installation and configuration information for SMESEC tools.

c) Public servants: The people working in public administration have a different level of access to the system than ordinary citizens. Even though the platform is designed to be protected against involuntary actions from public servants, still the need to increase their security awareness is crucial to the protection of the whole system. For this group of people, we plan to promote a series of courses for basic cybersecurity that focus on good practices, phishing attacks' protection, antivirus protection etc.

Finally, UOP is currently building a training website focused on cybersecurity called https://www.securityaware.me. For this website, cybersecurity experts, groups and organizations are invited to create courses of various security topics and level of complexity. The sense.city team is also looking to benefit from the content of this website to increase their security expertise (Figure 16).

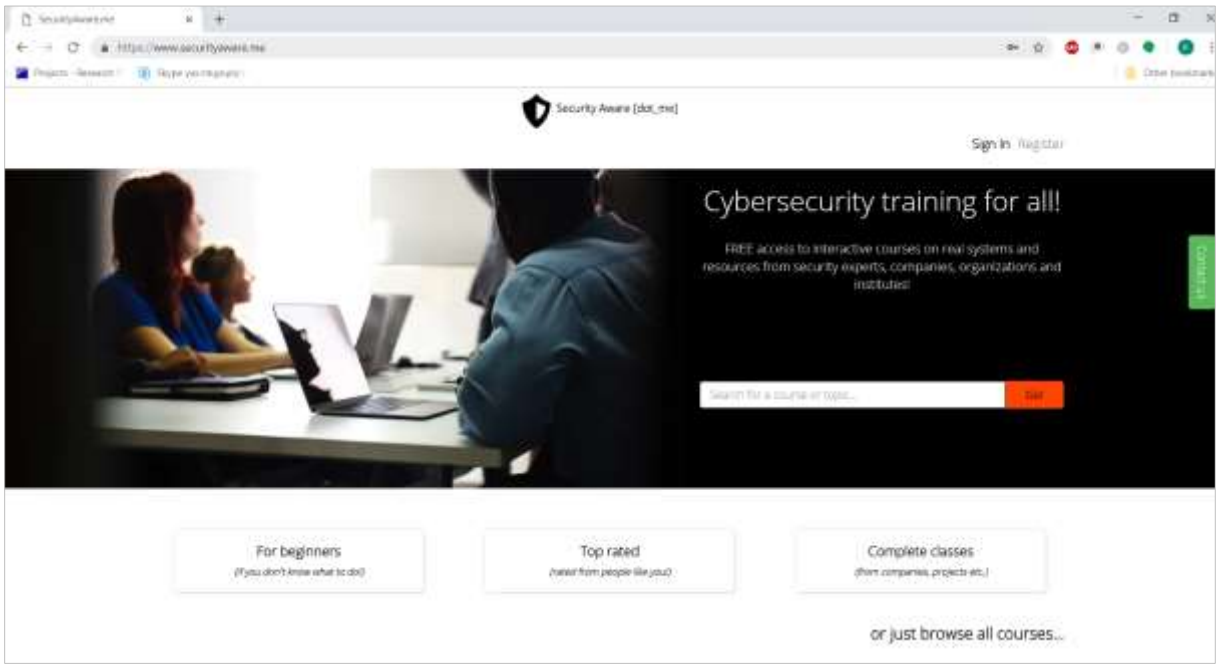


Figure 16. Securityaware.me training platform

5.3 Initial testing and validation plan

The Sense.city Platform consists of several components that need to be protected from possible cyberattacks. For this reason, we have identified and installed several products of the SMESEC toolset in the Sense.city backend and frontend to enhance the platform security protection. More specifically, ATOS XL-SIEM agent, Bitdfender's GravityZone and FORTH's Honeypot were installed in the sense.city backend in independent Virtual Machines. Also, the developed code for sense.city services are passed through the IBM Code assessment tool to assess possible vulnerabilities and the overall sense.city functionality is modelled using the EGM modelling toolset. For the above SMESEC products/solutions we have designed an initial test plan to verify their ability to identify and mitigate cybersecurity attacks.

The following table provides an overview of the initially identified tests to be launched on the SMESEC framework components that are currently been integrated in the sense.city system. Some tests are associated with individual components/tools while there are also tests that are validating the functionality of more than one SMESEC tools (denoted as jointed tests).

Validation campaign: planned tests			
Code	**I/ J**	**Tool**	**Description**
T_01_ATOS	Individual	XL-SIEM	Create relevant alerts from suspicious behaviours
T_02_BD	Individual	Gravity Zone	Malware deployment and detection by the BD antivirus
T_03_FORTH	Individual	FORTH honeypot	Detect and get information of an Intrusion attack
T_04_CA_IBM	Individual	IBM Code Assessment	Detect possible Code vulnerabilities in sense.city developer program code.
T_05_EGM	Individual	Security Testing	Model Sense.city platform and identify possible security vulnerabilities of this model
JT_01_ATOS_BD	Joint	XL-SIEM & Gravity Zone	Malware detection and reporting on the XL-SIEM system
JT_02_ATOS_FORTH	Joint	XL-SIEM & FORTH honeypot	Possible attacks on the honeypot reported on the XL-SIEM system

Table 2. SMESEC framework validation: list of planned tests

Below the individual and joint tests are briefly described, and the success criteria identified.

T_01_ATOS: SIEM anomaly behaviour alerts	
Objective:	Generate a series of relevant alerts by introducing abnormal behaviour in the Sense.city system
Test definition:	The XL-SIEM must be able to collect logs from its agents and generate alerts based on possible abnormal internal behaviour.
Fail criteria:	The XL-SIEM system does not collect the expected number of logs. The XL-SIEM does not generate alerts although it has collected logs of abnormal behavior (does not characterize the behavior as abnormal, false negative). The XL-SIEM characterizes as abnormal, traffic that is not malicious (false positive).
Success criteria:	The XL-SIEM is able to correctly detect abnormal behaviour. The XL-SIEM collects all relevant logs. Acceptable true positive and true negative percentage of 80%.

T_02_BD: Private Cloud protection	
Objective:	Detect the presence of malware within the servers of the Sense.city private cloud. Provide appropriate reaction to the attack
Test definition:	A malware is introduced in the Sense.city private cloud servers using some known exploitation. The BD gravity Zone is meant to detect the attack and perform appropriate actions, log entries etc.
Fail criteria:	No detection of the malware neither proper reaction: deletion of the file, quarantine or similar.
Success criteria:	Detection of the malware and proper reaction.

T_03_FORTH: Intrusion detection	
Objective:	Detect an external to the Sense.city network Intrusion attack
Test definition:	Perform an Intrusion attack designed and executed by an external entity to the FORTH honeypot and Sense.city system.

Fail criteria:	The attack is not detected or there is no information about it.
Success criteria:	The attack is detected and there is available information regarding the attacker in the form of log entries and alerts in the FORTH's Honeypot database.

T_04_CA_IBM: IBM Code Assessment	
Objective:	Detect possible Sense.city developed code vulnerabilities
Test definition:	Program Code that is to be deployed in the Sense.city platform needs to be passed through the IBM code assessment tool to find possible vulnerabilities. Introduce a series of known code vulnerabilities in a Sense.city test service and use the IBM tool to detect them.
Fail criteria:	The vulnerabilities are not detected or are partially detected (ie. There are more than 30% undetected vulnerabilities or more than 10% falsely detected vulnerabilities)
Success criteria:	The vulnerabilities are detected. We have a true positive of at least 70% and a true negative of 90%.

T_05_EGM: Security Vulnerabilities through EGM Modeling of the Sense.city platform	
Objective:	Detect possible Sense.city security vulnerabilities by modelling the platform behavior
Test definition:	Model and Simulate the Sense.city operational behaviour to detect security flaws
Fail criteria:	The Model is not accurate or is unable to detect known Sense.city vulnerabilities
Success criteria:	The model is complete and detects known Sense.city vulnerabilities

JT_01_ATOS_BD: Malware detection and XL-SIEM alert generation	
Objective:	The BD antimalware detects a malware attack and forwards log entries to the ATOS XL-SIEM tool to generate alerts and notifications
Test definition:	Similar to T_02_BD test we introduce a Malware in the Sense.city platform and generate log enties. Such entries are captured by the XL-SIEM tool and produce alert messages that are visually evident to the security administrators
Fail criteria:	No detection of the attack or no reporting to the administrator
Success criteria:	Simultaneous detection and alarm triggering

JT_02_ATOS_FORTH: Network attack FORTH honeypot detection and XLI-SIEM alert generation	
Objective:	The FORTH Honeypot detects a network attack (eg. A DDoS or intrusion attack) and forwards log entries to the ATOS XL-SIEM tool to generate alerts and notifications
Test definition:	Similar to T_03_BD test an external entity performs a network security attack (eg. A DDoS or intrusion attack) on the Sense.city platform servers. The FORTH honeypot should be able to detect the attack and the attackers and generate log entries. Such entries are captured by the XL-SIEM tool and produce alert messages that are visually evident to the security administrators
Fail criteria:	No detection of the attack or no reporting to the administrator
Success criteria:	Simultaneous detection and alarm triggering

6 Conclusions

The integration of SMESEC framework so far has been a smooth process. The consortium did not face major problems and the remaining steps do not seem to pose any significant risks for the successful completion of this task. Considering the benefits of this framework to pilot II, UOP admits that their participation in this project was the main reason to focus on securing their services and infrastructures. Only after the initial evaluations of the security status of sense.city and their alarming low scores, the UOP team decided to dedicate more time for protecting their assets. Currently the security status is at a far better state and until the end of the project UOP is looking to have robust protection at least against untargeted attacks and a set of internal security polices/plans for the team to follow.

6.1 Experience of the initial integration

As mentioned above the initial phase of the integration did not meet any major obstacles and at M18 some tools are fully installed and functional, while the rest of the selected ones for pilot II are already under deployment. All partners have done their best to help the UOP team properly install and configure the selected tools in sense.city service. It must be noted that some of them have done more that was required from them e.g. FHNW providing an extensive vulnerability analysis for the UOP infrastructure, IBM and FORTH bringing the javascript fuzzing and cloud security tools respectively, solutions that were not part of their initial toolkit.

UOP is already adding some of the successfully tested tools to the sense.city platform (antimalware, honeypots) and is looking forward to use the complete SMESEC framework or part of it on their production infrastructure.

6.2 Fulfilling of objectives

Considering UOP's main objectives and how they are addressed inside the context of SMESEC project the status is as follows.

a) Protection against untargeted attacks: New vulnerability analysis after the installation of some SMESEC security tools and updates of the existing components indicate that the security levels of UOP infrastructure is now higher. There are still actions that need to be implemented in order to reach an adequate level of protection. Among the most important things that UOP is looking for is to develop a proper security plan will be able to follow after the end of the project.

b) Increase awareness: Within SMESEC project UOP identified the low cybersecurity levels of its infrastructure and how this situation may severely impact its services and business/research goals. The UOP team put cybersecurity as one of their highest priorities and has started working to apply the necessary security solutions and also train/educate all people involved in sense.city service (developers, IT, public servants etc.).

c) Create a market product: To be able to make sense.city a product that can be sold, UOP team must make sure that it is, among others, properly protected from cyber threats. SMESEC is a

great opportunity for the UOP team to work with security professionals, increase sense.city's security levels and create high quality product that can have immediate market impact.

6.3 Use in SME environment

Until M18 the UOP team worked together with SMESEC security professionals to install the selected tools and make proper configurations. Our experience so far is that all installations were user friendly and easy to configure. The only exception is the CYSFAM model we were asked to complete. For this model our opinion is that it was very long, complicated and occasionally difficult to understand. Finally, another issue we presented to the consortium is the fact that it is unclear how difficult is to integrate external (our own) security tools with some of its security components (e.g. XL-SIEM) or with the overall SMESEC framework.

6.4 Improvements for the scenario

At this point the UOP team does not have any suggestions to improve the proposed scenario. This is due to the fact that the implementation of the initial planning is still undergoing. Once more security components are added to the pilot and the SMESEC framework we shall evaluate again the overall configuration for the sense.city platform and present our proposed adjustments if any.