Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

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

D5.1 Trial scenario definitions and evaluation methodology specification

Document Identification			
Status	Final	**Due Date**	31/05/2019
Version	1.0	**Submission Date**	23/06/2019

Related WP	WP5	**Document Reference**	D5.1
Related Deliverable(s)	D5.1	**Dissemination Level (*)**	PU
Lead Organization	GRIDP	**Lead Author**	Andreas Last, GRIDP
Contributors	Andreas Last, GRIDP	**Reviewers**	Kostas Lampropoulos (UoP)
			Manos Athanatos (FORTH)

Keywords:

Trial, Evaluation, Testing, Methodology

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.

(*) Dissemination level.-**PU**: Public, fully open, e.g. web; **CO:** Confidential, restricted under conditions set out in Model Grant Agreement; **CI:** Classified, **Int =** Internal Working Document, information as referred to in Commission Decision 2001/844/EC.

Document Information

List of Contributors	
Name	Partner
Andreas Last	GRIDP
Michal Burdzy	GPS
Adam Nawarycz	GPS
Francisco Hernandez Ramirez	WOS
lmo Rayón Pérez	WOS
Samuel Fricker, Martin Gwerder, Alireza Shojaifar	FHNW
Bilge Yigit Ozkan	UU
Jose Francisco Ruiz Rodriguez	ATOS
Abbas Ahmad	EGM
Noemi Folch	SCYTL
Fady Copty	IBM
Christos Tselios	CITRIX

Document History			
Version	Date	Change editors	Changes
0.1	15/02/2019	Andreas Last, GRIDP	First draft for the general testing strategy
0.2	21/02/2019	Andreas Last, GRIDP	Second draft improved several parts of the document
0.3	29/03/2019	Samuel Fricker, FHNW	CYSEC drafts
0.4	26/04/2019	Michał Burdzy, GPS	Tool owner inputs, approving Samuel changes
0.5	29/04/2019	Francisco Hernandez Ramirez, WOS	General review of the version

0.6		Kostas Lampropoulos	Contribution in section 4 for smart city pilot
0.6.1	23/05/2019	Samuel Fricker, Martin Gwerder, Alireza Shojaifar	Refinement of CYSEC parts.
0.6.1	30/05/2019	Kostas Lampropoulos	First review (QA)
0.7	19/06/2019	Manos Athanatos	Second review
1.0	23/06/2019	ATOS	Quality review + submission to EC

Quality Control		
Role	Who (Partner short name)	Approval Date
Deliverable leader	Michal Burdzy (GPS)	23/06/2019
Technical manager	Christos Tselios (Citrix)	23/06/2019
Quality manager	Rosana Valle (ATOS)	23/06/2019
Project Manager	Jose Francisco Ruiz (ATOS)	23/06/2019

Table of Contents

List of Tables

List of Figures

List of Acronyms

Abbreviation / acronym	Description
GW	Gateway
ROP	Return-oriented programming
SaaS	Software as a Service
SEM	Security Event Management
SIEM	Security Information and Event Manager
SIM	Security Information Management
SME	Small-Medium Enterprise
SW	Software
TaaS	Test as a Service
WP	Work Package
OWASP	Open Web Application Security Project
ITU	International Telecommunications Union
OSI	Open Systems Interconnection model
WHOIS	Query and response protocol that is widely used for querying databases that store the registered users or assignees of an Internet resource, such as a domain name, an IP address block or an autonomous system
ICMP	Internet Control Message Protocol
ARP	Open Web Application Security Project
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
IDLE scan	TCP port scan method that consists of sending spoofed packets to a computer
ARP	Address Resolution Protocol
DNS	Domain Name System
IP	Internet Protocol
SSL	Secure Socket Layer

DHCP	Dynamic Host Configuration Protocol
DOS	Denial of Service
DDOS	Distributed Denial of Service
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
IRS	Intrusion Recognition System
SWG	Secure Web Gateway
LDAP	Lightweight Directory Access Protocol
RADIUS	Remote Authentication Dial In User Service
TACACS+	Terminal Access Controller Access
Negotiate	Microsoft Windows authentication mechanism
FPR / TPR	False Positive Rate / True Positive Rate

Executive Summary

The purpose of the task 5.1 is to define the evaluation methodology (timing, procedures, people, and equipment) that will be used by the four pilot partners in order to test the different modules of the SMESEC framework. Different scenario experiments are defined for each component of the framework, with the goal to cover the widest possible area of the developed SMESEC system modules. A list of usability criteria is also defined, for the evaluation process, with the goal to validate the transition from the lab testing system, to the demonstrator system.

1 Introduction

Testing the different modules of the SMESEC framework, is necessary in order to validate the overall functionality of the final solution. To do so, and in order to verify the effectiveness of the different SMESEC framework assumptions, specific testing strategies must be defined with the goal to cover a maximum of security vulnerabilities.

1.1 Purpose of the document

This document defines different testing strategies for all SMESEC framework components, building the basis for the further tests which will be realized on the system. It describes different scenarios in order to test each component of SMESEC framework as well as related deliverables, testing environment and specific vulnerabilities. It also describes how the specific components used by SMESEC framework are supposed to respond to cybersecurity threats for the four pilots. In the last section of the document, we present a list of possible test cases which aim to validate the totality of the security assumptions of the framework.

Deliverable description

O5.1: Design in field system trials for the SMESEC prototype to evaluate its functionality.

T5.1: Setup and configuration of the SMESEC trials.

Define the **details of the scenarios for the trial** of the SMESEC security framework. It will also provide a **list of usability criteria for the evaluation**, which will be prioritized for testing.

The **evaluation methodology** will detail the procedures that will be followed at various **stages and activities:**

1. **Definition of experiments** that cover a wide area of the developed system modules and aim to evaluate the performance of the individual modules in controlled environments Define a test case for each part of the SMESEC framework.
2. **Definition of the proof of concept** scenarios in the four pilots, aiming to show the performance of the integrated system modules.
3. **Definition of an evaluation methodology** for the technical aspects of individual technologies
4. **Transition from lab testing system to demonstrator system** for the four pilots.
5. **Set trial goals** in light of project objectives and KPIs.
6. **Refine planning for trial** (timing, procedures, people, and equipment).

1.2 Relation to other project work

Testing all the possible vulnerabilities and attack scenarios will be the next step following the integration of the security features in the environment. This will validate the different roles and utilities of the different SMESEC components. Before this, a convincing and complete testing strategy need to be designed.

Related deliverables allowing full understanding of testing strategies are:

- Security characteristics description, security and market analysis report: D2.1.
- SMESEC security products unification report: D2.2.
- Security Awareness Plan Report: D2.3.
- Preliminary Pilots integration reports: D4.1, D4.3, D4.5, D4.7.
- Final Pilots integration reports: D4.2, D4.4, D4.6, D4.8.
- Overall Pilot alignment and integration process report.

The outcomes and evaluation plans of this deliverable will be also used to assist the evaluation of the framework by the means of external SMEs, in Task 5.5 – Open Call.

1.3 Structure of the document

The document is divided in four parts.

Chapter 1 Introduction.

Chapter 2 General testing strategy.

Chapter 3 Security testing for the components used in SMESEC framework.

Chapter 4 Security testing for the different pilots.

2 General testing strategy

This chapter will focus on the relevant IT-security fields, describing the possible attacks that the SMEs using the SMESEC framework are exposed to, a necessary step for defining the evaluation of the framework and the different SMESEC components. The purpose is to ensure that all the vulnerabilities described in D2.1 which are directly related to the SMESEC framework are covered by one or more of the module solutions or the CYSEC framework.

Several parts of this document might not be complete, due to the fact that the SMESEC framework could not have been fully implemented and deployed on the pilots by this time. Additional details will then be added in D5.2.

2.1 IT-Security fields

The testing strategy will cover the three performance layers of the IT-security field: confidentiality, integrity and availability (also known as the CIA triad[1]).

2.1.1 Confidentiality / Secrecy

Secrecy and confidentiality consist in the prevention of unauthorized access to private or protected information. It will be tested for all the different software modules.

2.1.2 Integrity

Integrity[2] consists in the prevention of unauthorized modifications of information. Authentication and reliability are part of integrity. The different tests of the framework will need to validate the following assumptions:

- Remote party is who he claims to be.
- Integrity of the other party needs to be respected.
- Peer entity authentication or identification is needed.
- Data origin is verified.

2.1.3 Availability

Availability is the concept that a system should be available anytime it is needed. More accurately, it will be important for each module to prevent DoS/DDoS type attacks and to have fall-back solutions for any type of issues for critical infrastructure.

[1] https://www.techrepublic.com/blog/it-security/the-cia-triad/
[2] https://en.wikipedia.org/wiki/Data_integrity

2.2 Potential attacks

2.2.1 Definition of the main potential attacks

In relation to the OWASP risks presented in “D2.3 Security Awareness Plan Report”, potential attacks will be directly simulated in pilot environments, to show that the deployed framework and the integrated security components, delivers the expected reaction. For each pilot different scenarios, based on specific vulnerabilities, will be defined in order to test different security requirements. Each possible vulnerability will be covered by at least one of the pilot test scenarios developed in this work package. Table 1 summarizes most of the well-known security attacks that will be covered by the SMESEC platform.

Table 1: Main attacks targeting a network infrastructure.

Attack	Description
Information Gathering	Information Gathering about the target systems. Usually the first step of the attack.
Social Engineering	Psychological manipulation of people that can help an attacker to gather information on how to attack a system (for instance by divulging confidential information).
	Fraudulent attempt in order to obtain sensitive information such as usernames, passwords and credit card details by disguising oneself as a trustworthy entity through an electronic communication.
Scanning	Set of procedures in order to identifying live hosts, ports, and services, discovering operating system and architecture of target system, identifying vulnerabilities and threats in the network.
Sniffing	Process of monitoring and capturing all the packets passing through a given network.
Spoofing	Situation in which a person or program successfully masquerades as another by falsifying data, to gain an illegitimate advantage.
Man-in-the-middle	Attack where the attacker secretly relays and possibly alters the communications between two parties who believe they are directly communicating with each other.

Exploitations: Buffer Overflow, SQL – Injection, XSS	Buffer Overflow is an anomaly where a program, while writing data to a buffer, overruns the buffer's boundary and overwrites adjacent memory locations. SQL injection is a code injection technique, used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution Cross-site scripting (XSS) is a type of computer security vulnerability typically found in web applications. XSS enables attackers to inject client-side scripts into web pages viewed by other users.

Figure 1 shows the different levels of information possibly looked for by an attacker.

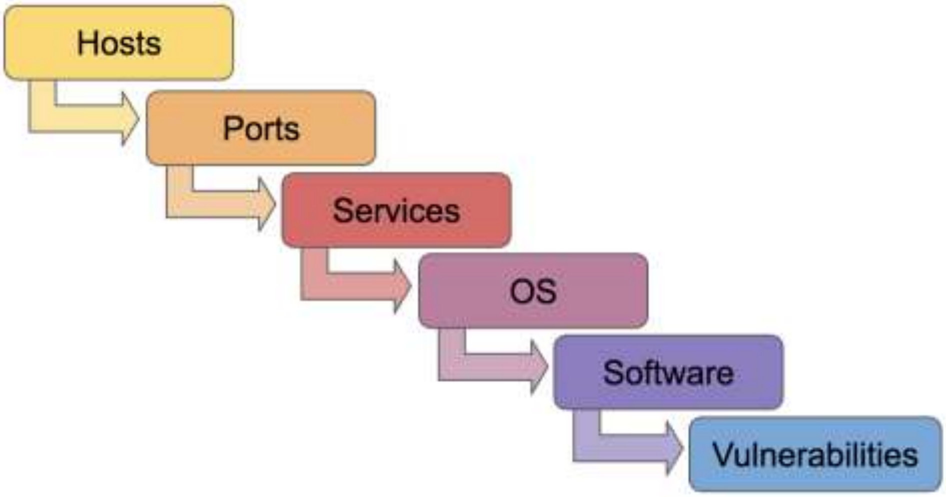


Figure 1: Information an attacker could look for during a network attack

Following, each potential attack is discussed in relation to the SMESEC framework.

2.2.2 Techniques of information gathering and its relation to the SMESEC framework

Information gathering consists in the non-authorized acquisition of private information. It necessitates the use of internet or human social interaction. Most of the information gathering methods are described in Table 2.

Table 2: Possible techniques for information gathering.

Method	Description
Simple search on the internet, website, WHOIS	General information gathering
Administration by other companies	Using exposed private data that is supposed to be administered by third-party company

Google Hacking	Technique that uses Google Search and other Google applications to find security holes in the configuration and computer code that websites use.
Dumpster Diving	Use of various methods to get information about potential victims, including physical methods
Shoulder Surfing	Type of social engineering spying technique used to obtain information such as personal identification numbers (PINs), passwords and other confidential data by looking over the victim's shoulder.

Interaction with the SMESEC framework:

Information Gathering should be covered in the CYSEC framework. This method uses human mistake, so the best defence is training, awareness and taking the view of an attacker to see the risks

2.2.3 Techniques of social Engineering and its relation to the SMESEC framework

Social engineering refers to the psychological manipulation of people into performing actions or divulging confidential information.

Table 3: Possible techniques for social engineering

Method	Description
Computer-based social engineering	Send of fake emails sending warnings about malware, virus and worms causing harm to the computers.
Human-based social engineering	Art of convincing people to reveal corporate secrets and confidential information.
Reverse social engineering	Person-to-person attacks in which an attacker convinces the target that he or she has a problem or might have a certain problem in the future and that he, the attacker, is ready to help solve the problem.
Structured attack with pretexting and elicitation	Pretexting is presenting oneself as someone else in order to obtain private information. Elicitation is the process of extracting information from something or someone.
Usage of Trojan (Baiting)	Trojan is a malware which misleads users of its true intent.

Interaction with the SMESEC framework:

Social engineering should be covered in the CYSEC framework. Antivirus (“Bitdefender”) and firewall (“CITRIX ADC”) should protect against some types of Social Engineering.

2.2.4 Techniques of Phishing and its relation to the SMESEC framework

Phishing is an attack using human factors to get protected or private information. The victim is enticed to divulge data unintentionally. Phishing is often part of social engineering.

Table 4: Possible techniques for Phishing

Method	Description
Faked identity	Ensuring a victim about fake identity of attacker (System Administrator, Government member, etc.).
Email pool	Sending infected files as email attachment.
Faked website	Website that tries to mimic social website, auction site, bank or online payment processors.
Spear-Phishing, Whaling, Pharming	Spear-Phishing is phishing attempt directed at specific individuals or companies. Whaling phishing attack directed specifically at senior executives and other high-profile targets. Pharming is a cyber-attack intended to redirect a website's traffic to another, fake site.

Interaction with the SMESEC framework:

Phishing should be covered in the CYSEC framework. SMESEC components should use only encrypted communication ways such as “https” or “ipsec” to communicate. “TaaS” should provide ways to achieve secure authentication methods. “Bitdefender” should be able to scan email attachments in order to detect malicious code.

2.2.5 Techniques of scanning and its relation to the SMESEC framework

Scanning a network helps to find potential attack areas. This type of attack can happen from outside (scanning of the firewall / router / server) or from inside of a network in a LAN to scan for potential hosts, to get knowledge about the used it-infrastructure / servers / software / ports etc.

Table 5: Possible techniques for scanning

Method	Description
Network recognition	The attacker tries to get as much information as possible about the network and the hosts in the network to prepare a future attack. The attacker checks which hosts are on the network. Next, the attacker detects each hosts’ OS, open ports, services and software used with these services. The attacker can use these methods to extract information about vulnerabilities of the network, then exploit them for an efficient attack. Protocols like ICMP (ping), ARP (arping), TCP (connect,

	half-connect, fin, xmas, null -scan, tcp-fingerprinting), UDP are often targeted. Decoy scan, IDLE scan, OS scan and Banner grabbing can be used by attackers.

Interaction with the SMESEC framework:

Scanning should be detected by Firewalls (Bitdefender and CITRIX ADC) and by the Honeypot, which should be able to detect different types of scans.

2.2.6 Techniques of sniffing, Spoofing and Man-in-the-middle and their relation to the SMESEC framework

Sniffing is the recording of network data (ex: tools like "Wireshark"). Spoofing is the manipulation of unsecure protocols in order to change protocol headers in a way that the data are redirected. Spoofing can also change authorization information or fake a false authenticity. Man-in-the-middle (MITM) is the most popular form of spoofing. It is a method that allows the interception and the manipulation of communication between two partners. Data coming from one victim go through the attacker before to be forwarded to the other communication partner. MITM attack is possible with different protocols such as ARP, DNS, DHCP but also in SSL-Spoofing with half or fully encrypted tunnel to both partners.

Table 6: Possible techniques for Sniffing, Spoofing and Man-in-the-middle

Method	Description
Promiscuous mode exploitation	Listening to the traffic at a network port in a promiscuous mode of the network interface
Physical device exploitation	Using the monitoring port of a switch or other network device
MAC flooding	Attacking old switches using mac flooding to bring them in the "fail over mode" (they are working as a hub so all the data are sent to all device ports)
Various protocols sniffing	Potential spoofing attacks are based on ARP, DNS, IP, Mail-Protocols, SSL, DHCP or WIFI-Protocols. Nearly every protocol can be used if it is not secure enough. Each of them has its own vulnerabilities and potential exploits, and protection from the protocol side is not always possible.

Interaction with the SMESEC framework:

- Sniffing, Spoofing and Man-in-the-middle should be covered by the CYSEC tool. Pilots should use only new and secure network devices in addition of secure protocols if possible.

- Protection against spoofing and man-in-the-middle attack should be covered through secure protocols, detecting and avoiding of attacks through “Bitdefender”, “CITRIX ADC” or “Forth Cloud-IDS”.

2.2.7 Exploit types (ex. Buffer Overflow, SQL-Injection, XSS…) and their relation to the SMESEC framework

Exploits are the systematic use of a vulnerability, security gap or another weakness in the code of a program in order to get higher privilege (privilege escalation), to get access to specific parts of a system or to initiate a DoS or DDoS attack.

Table 7: Exploit types

Method	Description
Local exploit	Ex: Browser exploits for back doors with special privileges
Remote exploit	Manipulated data packages, alternate data streams
Privilege escalation	Bug exploitation, design flaw or configuration oversight in an OS or application
Zero-day exploit	Unknown or not-well-known exploit; no firewall rules / Antivirus signatures can detect it.
XSS	Injection of JavaScript code into webpage to bypass access control of the site.
Code Injection	SQL Injection, HTML Injection
Buffer Overflow	When a program overruns the buffer's boundary and overwrites adjacent memory locations. This can cause unpredictable behaviours of the program.
Command Execution Exploits	Execution of arbitrary commands on the host operating system using a vulnerable application

Interaction with the SMESEC framework:

The different variations of the exploits should be covered in the SMESEC framework. Virus scanner (“Bitdefender”) or the firewalls/IDS (“CITRIX ADC” and “FORTH”) should detect this type of exploits. Pilots should implement common software development good practices such as pair programming, code reviews and others. Source code should not be visible and reachable from outside the network.

2.3 Security features of the SMESEC framework

SMESEC should be able to cover the most critical security attacks described in the section 2.2. These security points can be grouped in four different categories

2.3.1 Authentication protection feature

The attacks during authentication are critical. They can be the consequence of the administrator, software producer but also from the end-user behaviour. It also depends if the authentication can be reached remotely or just locally.

Protection methods:

- Use of a “strong” password.
- Create password rules (Minimum 10 characters, special characters, upper and lower. characters, Maximum number of tries…) and forbid “weak” passwords.
- Captures for login pages reachable from the network / internet.
- Safe storage in the database (Hash + Salt/Pepper), not reachable from the network / internet.

SMESEC framework action:

- Best practice for passwords should be covered by the CYSEC framework and IT-security audits.
- Check if the different software modules have safe login procedures (servers).
- Check if the login pages are reachable from the internet if they shouldn’t.

2.3.2 Network security features

Network security attacks concern security features such as firewalls, IDS, IPS, IRS.

Protection methods:

- Personal Firewall on host (“Bitdefender”) can protect not-allowed communications but cannot detect malware.
- External Firewall WAF (CITRIX ADC) in the router with deep package inspection can inspect the application payload to find suspicious behaviours.
- IDS / IPS / IRS in all networks can help to detect suspicious activities.

SMESEC framework action:

- Network-based attacks should be covered in the CYSEC framework and IT-security audits (use of firewalls and IDS).
- Protection against “Metasploit” attacks as browser or session hijacking.
- Easy and understandable configuration surfaces for the firewalls to avoid mistakes (“Bitdefender”, “CITRIX ADC”).
- Use of “DMZ” for critical services with two different back to back firewalls (front and back ones) (if one is broken, we can use the second one for security protection).
- Cloud-IDS (“Forth”) in all networks and maybe on hosts like webserver/mail server/databases or other servers, if possible, with IPS functionalities □ preferred using hybrid IDS (host and network).
- Use of dynamic packet filtering.
- Presence of webservers and other services for DMZ or cloud access (no local).

2.3.3 Malware protection feature

A Malware (Malicious Software) is developed by attackers in order to infiltrate a network or a host, to manipulate data structure, to spy or to destroy the data.

Some examples of common malware:

- Virus.
- Worm.
- Trojan.
- Ransomware.
- Adware.
- Scareware.
- Drive by Exploits.
- Spyware.
- Keylogger.

SMESEC framework action:

- Malware detection by the WAF (“CITRIX ADC”) or in the local virus scanner (“Bitdefender”).
- Malware detection through Forth “Cloud-IDS” or “IPS/IRS”.
- Use the on-demand and on-access opportunities from the virus scanners.
- Check Bitdefender status.
- Check CITRIX ADC status.
- Check Forth Cloud-IDS status.
- Check that the firewall protects unwished data streams.
- Control administration rights of the applications.

2.3.4 Communication protocol features

Protocols are necessary for the communication in a network. Their level of security can vary and security recommendations are necessary to avoid vulnerabilities in the local network.

Protection methods:

- Use of encrypted communication to transfer data (https, ssl, ipsec…)
- Use of secure protocols for safe communication (DNSSEC, https, wpa2…)
- Use of a secure wpa2 (AES) connection for WIFI and use of software to secure authentication of the users (without using WPS functions)

SMESEC framework action:

- Communication guide / encryption methods should be covered in the CYSEC framework, IT-security audits.
- Use tools to check the detection of unsecure protocols use.

Use tools to check the detection of unsecure authentication methods.

3 Evaluation of the SMESEC framework

To validate the security features of the different tools used in the SMESEC framework, each module functionality needs to be verified.

3.1 Definition of the different components of the SMESEC framework

The following table describes the different components of the SMESEC framework.

Table 8: Overview components in the SMESEC framework

Tool	Partner	Purpose
Monitor & Protect		
XL-SIEM and XL-SIEM agent	ATOS	security information and event management are event loggers that can be analyse malicious behaviours in a network. They can be used to detect and solve APT (Advanced Persistent Threats) attacks.
Bitdefender Endpoint Security and GravityZone server	Bitdefender	Endpoint Security (antimalware, firewall…) for enterprises in combination with a central management platform. The GravityZone has an integrated Advanced Threat Control (ATC) to monitor running processes and to detect malicious behaviour.
CITRIX ADC - Web App Firewall (WAF) - Gateway - Secure Web Gateway (SWG)	CITRIX	ADC is an application delivery controller, which performs application-specific traffic analysis to intelligently distribute, optimize and secure Layer 4 -7 network traffic for web applications. Citrix ADC supports highly sophisticated load balancing functionality, while its feature set includes switching, security, protection and server-farm optimization features. Citrix ADC offers a virtualized flavour, which can be deployed in AWS, Google Cloud or bare-metal infrastructure. Upon obtaining specific licensing, Citrix ADC can be internally configured to efficiently support: ∉ Web application firewall (WAF), to protect web applications and sites from application layer and zero-day threats. ∉ Gateway to provide secure remote access to services and applications ∉ Secure Web Gateway to eliminate blind spots introduced by encrypted traffic and apply corporate

		policies to block malicious content ensuring user protection.
EWIS Honeypot	FORTH	Early Warning Intrusion Detection System has distributed low interaction honeypots (sensors) as a “Playground” for attackers to detect malicious activities. EWIS includes various flavours of honeypots covering well-known services and protocols (HTTP, FTP, SMB, SQL and more).
Cloud-IDS	FORTH	Cloud-based Intrusion detection system that monitors all inter- and intra- traffic of the VMs running on the hypervisor. In the original version, XEN hypervisor is supported.
Vulnerability Discovery & Patch		
TaaS	EGM	Test as a service software to proof a SUT (System Under Test) configuration and their functionalities offline or online.
Virtual Patching (AngelEye)	IBM	Tool producing a predictive model for a security solution to show if an input will exploit the vulnerability of an application.
Testing Platform (ExpliSat)	IBM	Testing Platform acting as a fuzzing engine. It receives source code and a test as inputs and produces code tests to execute run-time paths adjunct to the run time of the given test. (Hybrid Testing Platform)
Moving Target		
Moving Target (AntiROP) - Code analysis	IBM	Analysis of the JavaScript code in order to find potential vulnerabilities, malicious code or exploits before running it on a user device.
User Training		
CYSEC Cybersecurity Coach	FHNW	CYSEC provides SMEs with the ability to assess, plan, and track improvements in cybersecurity in a simple, do-it-yourself fashion. For a SME that is aware of cyber risks, CYSEC offers easily understandable cybersecurity advice and offers a personalized, self-adaptive journey of building cybersecurity capabilities to protect the SME

3.2 Definition of an evaluation methodology

The evaluation of the SMESEC framework is based on practical trial scenarios, based on the Architecture of the SMESEC framework and also covering the most important features of the different integrated security solutions. These trial scenarios aim to validate if main attack types can be eliminated or minimalized by using SMESEC framework.

3.2.1 Monitor & Protect

3.2.1.1 XL-SIEM

The XL-SIEM agent server is provided by Atos with a test library, which can be run to ensure that the events are being sent to the XL-SIEM server, proving that the configuration was done in the right way. If XL-SIEM server works correctly, all events logs should be visible in XL-SIEM panel.

Library provided by Atos tests the events coming from FORTH EWIS and BitDefender GravityZone. The XL-SIEM can also be tested with a shell script that will send some predefined events to the XL-SIEM agent. Each event can reflect a specific test case. To run the script, a documentation is designed to explain how to use it.

3.2.1.2 Bitdefender Endpoint protection and GravityZone

Bitdefender is the anti-malware component of SMESEC and has two different components:

- GravityZone server and
- Endpoint Security

The Endpoint Security component is deployed on every machine and offers real-time protection. The GravityZone can be deployed locally and must be able to connect to the endpoints. The Endpoint security component should communicate with GravityZone, which in turn must send logs to XL-SIEM.

Bitdefender uses a dummy malware file to test the security of local hosts protected by the GravityZone. This file is not an actual malware but has been created to simulate a malware for testing purposes.

The Bitdefender Endpoint protection has to detect malicious files as malicious and must inform the GravityZone. An alert has to be generated and information must be sent to the administrator and to the XL-SIEM. The same reaction should be tested for the different hosts, protected by Bitdefender, used in the pilot.

3.2.1.3 CITRIX ADC (Firewall / Gateway / Security Web Gateway)

Testing the overall Citrix ADC functionality deployed under the auspices of SMESEC can be done through malicious traffic forwarding to systems protected by the specific solution. A virtualized Citrix ADC node is collocated with the server it protects, intercepts all ingress traffic streams and categorizes requests based on pre-defined policy rules. All incoming requests that are not aligned with the aforementioned policy rules are considered as malicious, are effectively blocked and finally discarded by the system thus pose no threat to the backend infrastructure. Citrix ADC is able to cooperate with auxiliary nodes of the SMESEC framework to support additional functionality once an attack is identified and tackled, as well as advanced monitoring of any incident that may considered being suspicious.

Table 9 provides an extended overview of the various security features/protocols of each Citrix solution, which can be deployed under the auspices of SMESEC Project, if considered appropriate for a specific use case, pilot or SME.

Table 9: Security features Citrix ADC/Netscaler

Tool	Security features
CITRIX ADC	Auth to the appliance (mgmt): Local, LDAP, RADIUS, TACACS+, SSH key-based
CITRIX ADC AppFirewall	Auth to vservers/applications: Local, LDAP, RADIUS, TACACS+, Client Certificate, Kerberos/NTLM, Negotiate, SAML SP/IdP, Web, OAuth, OpenID Connect, Multi-Factor
CITRIX ADC Gateway	Auth to VPN GW: Local, LDAP, RADIUS, TACACS+, Client Certificate, SAML, OTP, Multi-Factor
CITRIX ADC Secure Web Gateway	Auth to SWG: LDAP, RADIUS, TACACS+, Negotiate (explicit proxy mode), LDAP (transparent mode)

3.2.1.4 EWIS Honeypot

EWIS Honeypot, focuses on attacks targeting database services. Honeypot exists for local and on-cloud deployment and is able to identify SQL version scan attacks and SQL unauthorised login attempts. Honeypot can also detect DDOS and brute force attacks.

3.2.1.5 Cloud-IDS

The Cloud-IDS aim to detect the presence of malicious traffic inside the cloud. To test the installation, malicious traffic can be generated from inside the private network. IDS must be able to detect potential attack.

The tests must focus on an attack on the network / system / cloud where the IDS is running. For example, all attack scenarios including malicious file downloads, buffer overflow attacks etc, should be detectable by the Cloud-IDS and reported back to the EWIS backend as well as alert the XL-SIEM.

3.2.1.6 Trial goals

Trials are designed to evaluate the security capabilities of SMESEC providing added-value to overall security of each pilot. They focus on several gdefined in the related D2.1 document: Usability, Privacy, Cost, Alerting, Scalability, System Integrity, Confidentiality, Non-repudiation, Authentication. They also allow to test specific components of the architecture of each Pilot, such as Database servers, Network traffic, Web servers, Email servers etc. A security level will be defined for each pilot after the testing process.

The consortium has designed tests to both validate the individual security solutions integrated into the framework and the new functionalities developed on top. The fields described above are the base for testing the implementation in the different pilots and demonstration systems. This chapter describes the basic idea of the different tests which will be run in the different test environments. Chapter 5.1 “Definition of experiments” describes the details of tests for each component of SMESEC framework in order to validate it. The framework components are shown in graphic below.

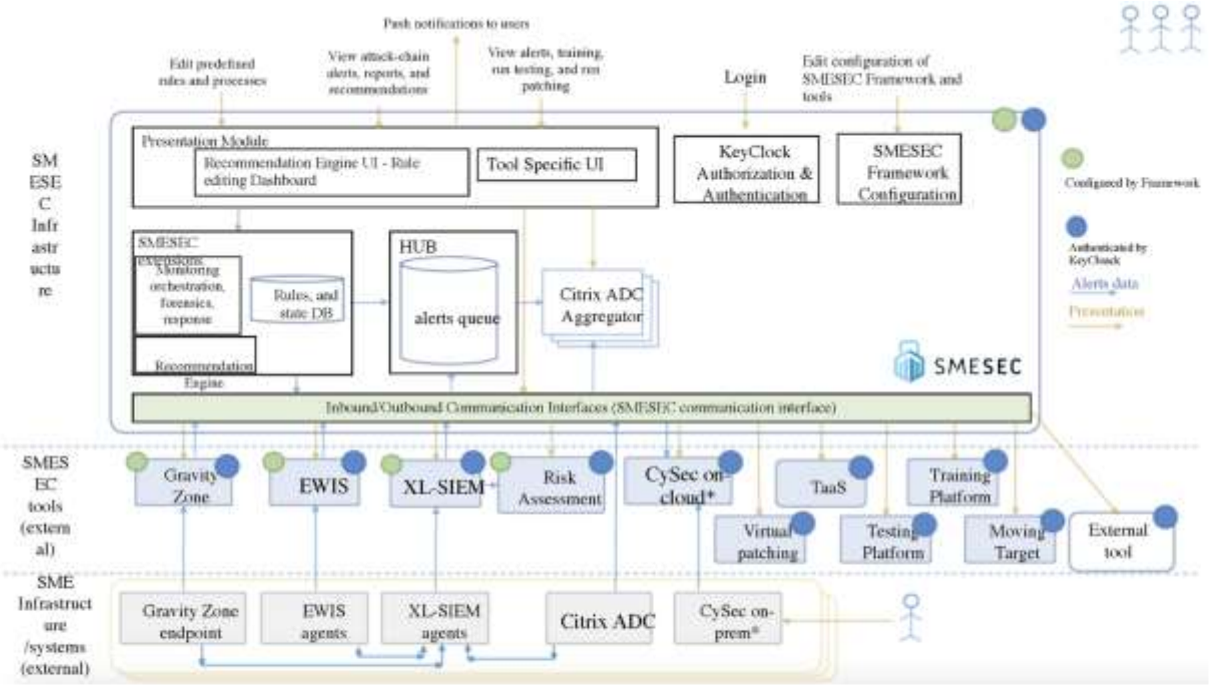

Push notifications to users
Edit predefined rules and processes
View attack-chain alerts, reports, and recommendations
View alerts, training, run testing, and run patching
Login
Edit configuration of SMESEC Framework and tools
SMESEC Infrastructure
Presentation Module
Recommendation Engine UI - Rule editing Dashboard
Tool Specific UI
KeyClock Authorization & Authentication
SMESEC Framework Configuration
SMESEC extensions
Monitoring orchestration, forensics, response
Rules, and state DB
Recommendation Engine
HUB
alerts queue
Citrix ADC Aggregator
SMESEC
Inbound/Outbound Communication Interfaces (SMESEC communication interface)
Configured by Framework
Authenticated by KeyClock
Alerts data
Presentation
SMESEC tools (external)
Gravity Zone
EWIS
XL-SIEM
Risk Assessment
CySec on-cloud*
TaaS
Training Platform
Virtual patching
Testing Platform
Moving Target
External tool
SME Infrastructure /systems (external)
Gravity Zone endpoint
EWIS agents
XL-SIEM agents
Citrix ADC
CySec on-prem*

Figure 2: Final components architecture

3.2.2 Vulnerability Discovery & Patch

3.2.2.1 Taas

EGM Test-as-a-Service (TaaS) is an online and offline testing solution where users can setup their System Under Test (SUT) configuration and launch test execution without any manual installation on the infrastructure . End-users can first define a configuration through a web application, then select which test cases should run. TaaS will produce readable reports in the web interface containing statistics, reports about test failures, etc.

Taas can create tests for threads as[1]:

- Man-in-the-middle (MitM) attack.
- Phishing and spear phishing attacks.
- Drive-by attack.
- Password attack.
- SQL injection attack.
- Cross-site scripting (XSS) attack.

There are 4 essential usages to the tool:

- Test campaign configuration.
- Test cases selection.
- Test cases execution.
- Result analysis.

Main components provided by TaaS:

- Platform for Test Execution with a Web Interface.
- Management of System Under Test.
- Management of Test Suites.
- Management of Test Reports.

3.2.2.2 Virtual Patching (Angel Eye)

The purpose of Virtual Patching is to produce a predictive model for a security solution (firewall/IDS). This solution will predict if an input can introduce a vulnerability exploit in this application.

Virtual Patching is a tool for the formal verification of C/C++ software. Based on a program in C/C++, Virtual Patching can verify that the program satisfies its properties - C assertions embedded within the program itself. Satisfying a property means no feasible execution path can lead to a violation of a corresponding C assertion. ExpliSAT is able either to prove that the property is indeed satisfied, or to falsify it by providing a counter example - C test case leading to an assertion violation.

Virtual Patching verifies a program using symbolic interpretation - combining explicit exploration of every feasible execution path of the program with symbolic representation of input variables. Symbolic representation of an input variable has a semantics of "a variable can take any value in its domain". Thus, by evaluating a single execution path, Virtual Patching verifies absence of failures for every possible input of a program that leads it through this execution path.

The testing goal is to validate that the predictive model provides reasonable FPR/TPR on input-samples and that the Integration into custom log file analysis produces the same results as in first goal.

3.2.2.3 Testing platform

The purpose of the Testing platform is to analyse a JavaScript code in order to find vulnerabilities and exposures that can appear during its execution.

It can use diverse testing technologies to ensure highest coverage possible. It also leverages the power of parallel testing to its highest levels.

3.2.3 Moving Target

3.2.3.1 AntiROP – Code analysis with ExpliSat

AntiROP is a solution that provides protection by creating unique libraries and devices. The system provides with prevention and detection techniques deployed on the endpoint software application against ROP and memory corruptions attacks.

The test purpose of antiROP source is to generate multiple unique copies of an executable to defend against ROP attacks. The testing goal is to validate that antiROP unique copies do not change executable functionality and defend against ROP attack.

3.2.4 User Training

3.2.4.1 CYSEC – Cybersecurity Coach

The CYSEC Cybersecurity Coach offers self-adaptive recommendations for do-it-yourself assessment and improvement of cybersecurity practices of SMEs. In comparison to the outsourcing of assessment and improvement advice, CYSEC aims at cost reduction for establishing cybersecurity controls and practices in the SME while achieving a quality level comparable to the advice that can be achieved with the outsourcing approach. The CYSEC validation targets scenarios encountered by the majority of SMEs where SMESEC will have a maximal impact: those SMEs that did acquire in-depth cybersecurity knowledge but have understood the importance of the topic thanks to the SMESEC and other cybersecurity awareness campaigns. The controls and practices of the highest priority for these SMEs correspond to the fast ramp-up areas described in the deliverable D2.3.

We will use qualitative research as the primary paradigm for evaluating CYSEC, augmented by the collection of quantitative data that is being discussed from the perspective of the participating SMEs. The use of CYSEC is a setting that involves the members of the SME as human actors who are active decision-maker for what is being done in the SME and how the SME employees behave in the technical domain of cybersecurity. In such a setting of socio-technical research, case studies are a common method (Runeson 2009). They allow a rich description of the setting and use diverse data collection methods, such as observation, artefact, and interviews, to answer what, how, and why questions asked in the evaluation. The answers to these questions allow testing hypotheses deductively or developing new hypotheses inductively when theory is not available to explain the phenomenon under investigation, e.g. for why SMEs adopt cybersecurity practices or abandon them.

In the evaluation of CYSEC, we will adopt a phenomenological approach to studying SMEs and analyse the data that is being collected. A phenomenological study describes the meaning of a concept or phenomenon for several individuals that experience the concept or phenomenon (Creswell 2017). For each SME, we will use a workshop to let the SME experience the use of CYSEC and collect data from them about their experience of using CYSEC, their stance towards the experience, and what the implications of CYSEC are for their SME. Also, for the study subjects who have experienced outsourced cybersecurity consultancy, we will elicit experiences of using that approach to eventually compare the CYSEC do-it-yourself approach with this benchmark approach. The collected data of the lived experiences will be aggregated into a composite description of the essence of the experience for all of the study subjects.

To interpret the case study results and insights, we will perform focus group workshops with cybersecurity experts and an SME association. Focus groups are interviews that let a group opinion of a phenomenon, the study's focus, emerge. In particular, each participant will bring his knowledge and background and allow the discussion to interpret the trial SMEs' experiences from a diversity of angles. The result will confirm some of the hypotheses underlying CYSEC, respectively stimulate new ideas and innovative concepts for overcoming barriers that were observed, e.g. for how to deploy and integrate CYSEC into an SME and how to structure the knowledge transfer from experts to the SMEs that the CYSEC tool automates for achieving cost-efficient capability improvements.

The CYSEC validation will be guided by the KPI stated in the SMESEC Description of Action and described in the following table. Of key interest is the cost reduction of the definition of cyber-secure digital technology for protecting the SME, which we will evaluate as a comparison between the use of CYSEC-enabled do-it-yourself and outsourced cybersecurity assessment and capability improvement.

Table 10: CYSEC evaluation KPI and approach.

KPI	Scope	Participants	Evaluation Approach
Costs reduction on the definition of cyber-secure digital technology, comparing CYSEC-enabled do-it-yourself and outsourced capability assessment and improvement.	Fast ramp-up cybersecurity capabilities defined in D2.3 for SMEs with little to no cybersecurity expertise.	Use case and real-life demonstration involving the 4 SMESEC use case SMEs and at least 2 open call SMEs[3].	Phenomenological research performed in an embedded multi-case study with CYSEC as the primary phenomenon experienced by the employees of the SME under investigation.
	Do-it-yourself experiences of SMEs.	Cybersecurity experts from the SMESEC consortium partners and 1 open call SME association.	Focus group interview for interpreting the results obtained in the multi-case study from the cybersecurity and SME experts' perspectives.

The validation will answer the following validation questions VQ1, VQ2, and VQ3:

- VQ1: *In comparison to outsourced cybersecurity consultancy, how does the use of CYSEC affect the cost of defining the cybersecure digital technology in the SME?* With the answer to the question, we evaluated the performance of CYSEC in the use case environments. Since the effectiveness of CYSEC depends much on social factors, it is critical that the research goes beyond the controlled environment and is performed in the real-world contexts of the SMEs. Hence, for CYSEC, we opt at immediate transit from lab testing to working with the full-scale pilts.
- VQ2: *How do the SMEs perform cybersecurity improvement when assisted with the CYSEC digital cybersecurity coach?* This question will be answered in a bottom-up fashion by observing the use case and open call SMEs over a prolonged period of CYSEC use. The outcome will be a description of the natural behaviour and needs of SMEs towards a digital coach such as CYSEC for process improvement. We expect to observe well-documented barriers and resistances for cybersecurity technology adoption and adherence by the SMEs and their employees. Among other causes, these barriers and resistances may be due to usability problems of the CYSEC tool or psychological and social factors preventing the adoption and adherence of the CYSEC cybersecurity capability improvement method. We also expect to collect lessons-learned and obtain the

[3] The 6 use cases and real-life demonstrations also contribute to the corresponding KPI of 6 use case demonstrations stated in the DoA.

feedback from the perspective of the SMEs in the form of recommendations for reducing these barriers and resistances and maximising the acceptance and impact of CYSEC.

- VQ3: *How should the CYSEC method be adapted to minimise cost and maximise impact in SMEs?* This question will be answered in the group interviews focusing on the experiences documented as answers to VQ1 and VQ2 and involving cybersecurity and SME experts. The answer to the question will inform the evolution of the CYSEC method and tool to increase the technology readiness towards TRL8/9.

Each case study will be based on the protocol specified in the table below. The protocol describes the people involved (study participants and researcher), the procedure to be applied, the timing of the interaction between the researcher and the case participants, and the equipment to be used. The work has started with the pilot workshops performed with the four use case SMEs during M23 as part of WP4, will started with the open call SMEs in M28 and continues until the month M32 where the system prototype demonstration and evaluation ends.

Table 11: CYSEC use protocol (relative timings used, starting with the first month of the study M1 expected to match M28).

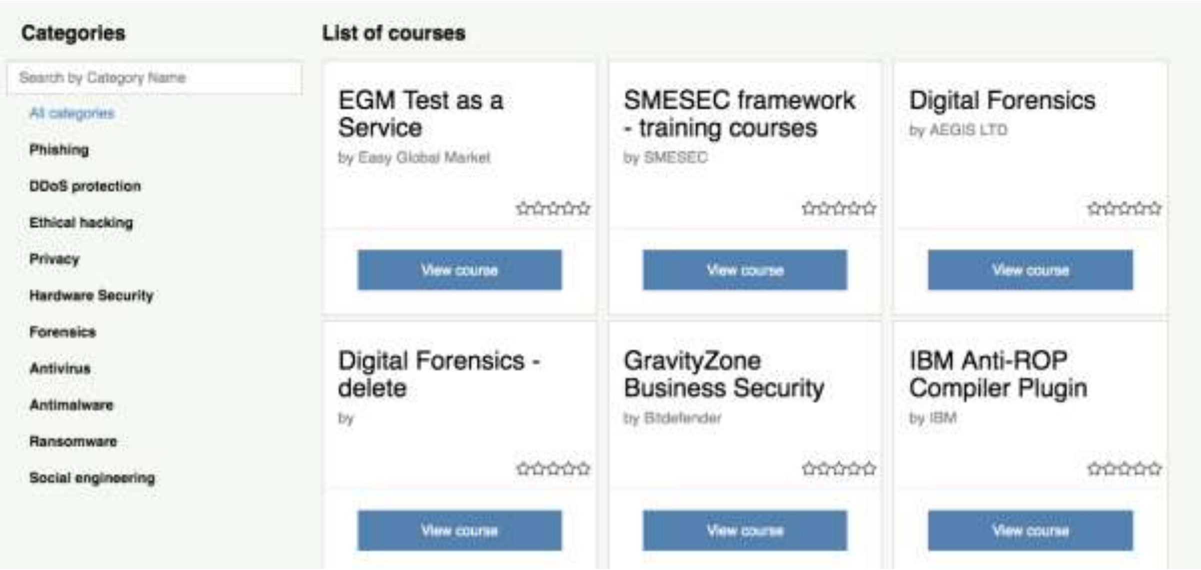
Timing	People and Procedure	Equipment
M1	1. **Physical Kick-off Workshop** 1.1 The participants fill out a questionnaire a) on organisational characteristics such as size, structure, industry type, and geographical distribution and b) on personal characteristics such as education, expertise in cybersecurity, and role in the SME, and c) the importance and priority of cybersecurity controls in the SME covering formal (policies), informal (culture, training), and technical aspects. 1.2 The participants install the CYSEC tool on the premise (optional) and create accounts for accessing its functionality. 1.3 The researcher introduces the CYSEC aims, tool, features, coaches, and content to the participants, offers guidelines for implementing the CYSEC method, and clarifies any questions about the use of CYSEC in the specific SME. 1.4 The participants fill out a questionnaire about the perception of CYSEC and the expected use of it in their SME.	- CYSEC tool - Interview questionnaire
M1-M5	Iterations over the following steps, first weekly, then progressively longer timespans: 2. **Prolonged use of CYSEC**: the participants use the CYSEC tool according to the recommended guideline, adapting what they believe to be best practice in their company. 3. **Online Status Meeting** 3.1 The participants share the data captured in the CYSEC tool with the researcher: the SME profile, the achieved maturity, and the log of interacting with the tool	- CYSEC tool updates - Interview questionnaire - Online meeting facilities

Timing	People and Procedure	Equipment
	3.2 The participants report their experience and reflect on lessons-learned, barriers, and potential resistance to the use of the CYSEC tool and to adopt or adhere to the recommended cybersecurity controls and practices. 3.3 The researcher informs about updates to the CYSEC tool and the coaches/contents contained in the tool and introduces protocol changes if any would be needed.	
M2 and M5	To be performed twice during the study: 4. **Physical Expert Focus Group Workshop** 4.1 The participants fill out a questionnaire on personal characteristics such as education, expertise in cybersecurity, and their role in the domain of cybersecurity for SMEs. 4.2 The researcher introduces the CYSEC aims, tool, features, coaches, and content to the participants, describes the guidelines for implementing the CYSEC method, and clarifies any questions about the use of CYSEC in SMEs. 4.3 The researcher introduces the case study results to the participants and discusses with them the meaning of these results, including their implications on the European community of SMEs, cybersecurity technology, corporate process/capability improvement, and digital coaching. 4.4 The researcher and participants conclude the focus group with recommendations for improving the CYSEC tool and method.	- CYSEC tool - Case study reports - Interview questionnaire - Physical meeting facilities
M1-M8	Iteratively when new results are available: 5. **Incremental Analysis and Reporting**: 5.1 The researcher analyses the findings of the case studies and focus groups to answer the validation questions. 5.2 The researcher communicates to the CYSEC development team requirements for the evolution and maintenance of the CYSEC tool.	- Study results
M8	6. **De-briefing** of the study participants with member checking of the documented study results.	- Study report - Online meeting facilities

The validation qualitatively and quantitatively assess the performance gains introduced by the CYSEC tool in each SME pilot environment. It will finalise the techno-economic analysis for CYSEC based on the awareness plan developed in WP2 and implemented in WP3. The validation contributes with a total cost of ownership (TCO) analysis, allowing future SMEs and stakeholders to evaluate the expected savings of bringing CYSEC into use in an SME or a community of SMEs. The validation also contributes guidelines for such CYSEC deployment to minimise cost and maximise value grounded on real-world empirical evidence. As a result of the real-world evaluation, import groundwork will have been laid towards exploiting CYSEC as a product.

3.2.5 UOP Training Platform

The UOP Training Platform for human end-user user training should be tested in its functionality on different devices and in the usability and customer experience point of view. It ensures that all important topics relevant to IT-security and especially in relation to the SMESEC framework are covered and understandable. To test this training platform, neutral (not involved in the project) test users will be found to evaluate and optimize the training.



Categories

Search by Category Name

- All categories
- Phishing
- DDoS protection
- Ethical hacking
- Privacy
- Hardware Security
- Forensics
- Antivirus
- Antimalware
- Ransomware
- Social engineering

List of courses

EGM Test as a Service by Easy Global Market	SMESEC framework - training courses by SMESEC	Digital Forensics by AEGIS LTD
Digital Forensics - delete by	GravityZone Business Security by Bitdefender	IBM Anti-ROP Compiler Plugin by IBM

Figure 3: Training platform dashboard

3.3 Definition of the requirements from the SME / pilots

In the beginning of the project, the pilots and the partners defined in document 2.1 different requirements and capabilities for the SMESEC framework, to create the secure environment for the SMEs. The requirements had been divided in business and platform requirements and protection capabilities. The evaluation is based on trials for most of the technical requirements and an evaluation schema to be filled out by the pilots after a test period. Table 12 and Table 13present the evaluation schema with the requirements in the first column ("Business-and Platform requirements"). Each pilot had to define their own requirements for their environment, which are depicted in the second column ("Required"). The evaluation of the different requirements can be found in the rows after. Thereby, requirements which cannot be covered by a tool are marked with "/". The pilots evaluate all the tools used by them, and indicate fulfilment rate between 0 and 100%, together with the numbers of the trial that test each requirement. All trials and their definitions are described in chapter 5.1 "Definition of experiments".

Table 12: Evaluation schema - business and platform requirements

Business-and Platform requirements	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	GravityZone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Availability	✓				/		/	/	/	/	
Usability	✓		/								/
Privacy	✓	/			/		/				
Cost	✓										
Alerting	✓						/				
Scalability	✓		/		/	/	/	/	/	/	/
System integrity	✓			/		/		/	/	/	/
Confiden-tiality	✓	/	/		/		/				/
Non-repudiation	✓	/		/	/	/	/				
Authen-tication	✓	/			/		/	/	/	/	/

Table 13: Evaluation schema - detection capabilities

Protection capabilities	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	Gravity Zone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Web application servers	✓	/			/	/	/	/	/	/	
Database servers	✓	/			/	/	/	/	/	/	
Network traffic	✓		/		/	/	/	/	/	/	
Web servers	✓	/			/	/	/	/	/	/	
Email servers	✓	/			/	/	/	/	/	/	
DDoS	✓		/		/	/	/	/	/	/	
Access abuse	✓		/			/					
Software misuse	✓	/		/		/					
Zero-day	✓			/	/	/	/				/

Protection capabilities	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	Gravity Zone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
attacks											
Code injection	✓	/	/	/	/	/	/				/
Man-in-the-Middle attacks	✓	/	/	/	/	/	/				/

4 Testing for the different pilots / demonstration systems

In order to prepare the tests from chapter 3 and use them in the different pilots, a general description of the test systems and the implementation of the used security features has been represented. The testing procedures for the pilots and demonstration systems has been created in connection with the system owners.

4.1 E-Voting – Scytl

The voting system has been deployed in the Amazon Web Services (AWS) system. It is composed of three main components: the web server (Apache), the application server (Tomcat) and the database (DB). The web server is deployed in a DMZ network, which is accessible through Internet. The application server and the database are deployed in a Secure Zone network, which is not directly accessible through Internet. In addition, when the voters connect to the system, a Javascript Voting Client is locally executed in their computers.

4.1.1 Pilot Architecture

In the pilot architecture we added the SMESEC Framework components. The integrated components are NetScaler, Angel-Eye, XL-SIEM and the different instances of the EWIS HoneyPot. NetScaler is used as an application firewall; thus, it is configured to be the first element that process the incoming connections that arrive from the Javascript Voting Clients in Internet to the web server. The EWIS HoneyPot that, for the pilot, is externally deployed, is used as a system to receive redirected connections rejected by NetScaler, i.e. connections that NetScaler have determined that are not compliant with the voting REST API. Angel-Eye is used as a periodically analyser of the HTTP requests received in order to detect attacks. The EWIS HoneyPot that is installed in the Secure Zone is a regular honeypot system used to attract attackers that are trespassing into this private network. And, finally, the XL-SIEM agent, deployed in a dedicated subnet, listens for Syslog connections from the other components deployed, e.g. web server, web application server, etc. The syslog of these components is forwarded to this agent that, in turn, forwards it to the external XL-SIEM server. The TaaS, that is not shown in the picture, is used as tool to test the voting system software before deployment.

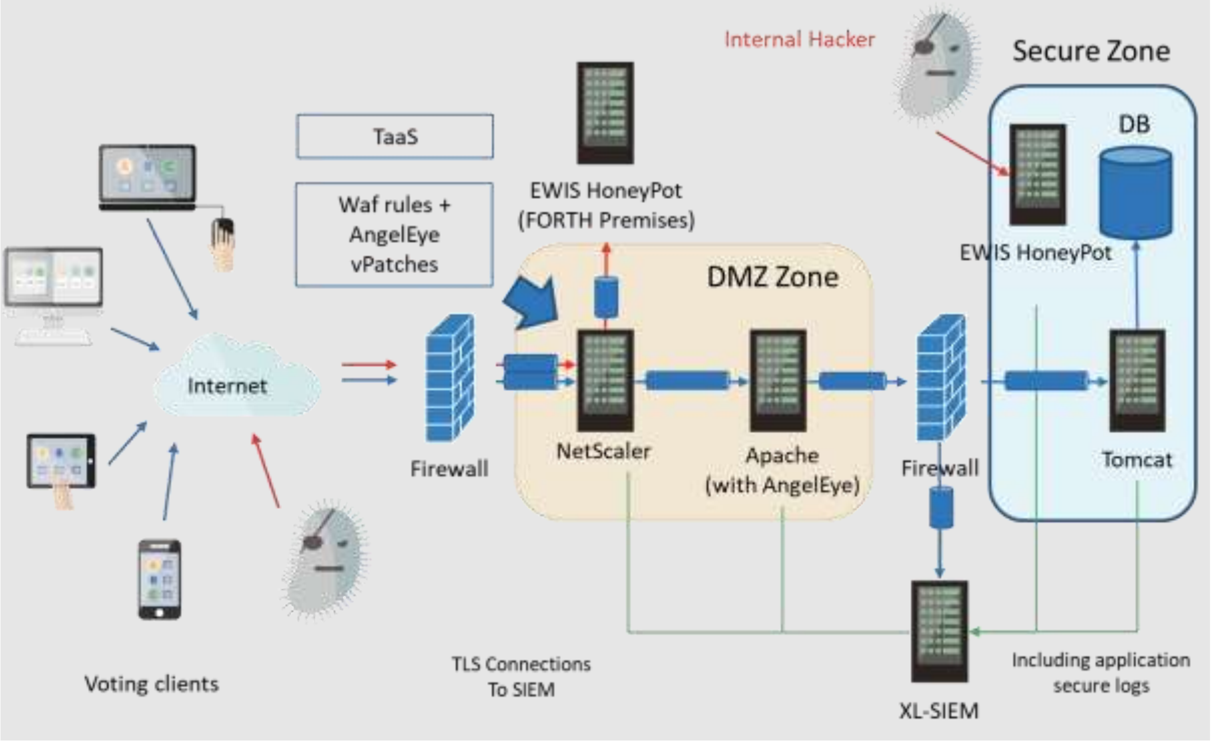


Figure 4: Pilot Architecture - E-Voting

4.1.2 Transition from lab testing system to demonstrator system

The demonstrator system is deployed on top of the Amazon Web Services, which is the same infrastructure used during the lab testing. Thus, there is no infrastructure transition in practise. Since the infrastructure is elastic, in case the demonstrator requires more capacity it can be provided by just updating the virtual hosts that support all the components (more processors, memory and network bandwidth can be added).

The main changes performed during this transition are fine tuning of the different elements that compose the system, for example the firewall rules that are included in the ACL lists of AWS, the ports open in each of the machines, the adjustment of the Netscaler rules, etc. Another change is the deployment of Angel-Eye as a periodic analysis tool instead of a real-time analysis tool.

4.1.3 Security tools used

Table 14: Security tools – E-Voting - Scytl

Tool	Partner	Status
XL-SIEM	ATOS	Integrated and working
CITRIX ADC	CITRIX	Integrated and working
EWIS Honeypot	FORTH	Honeypot in Secure Zone is integrated and working Honeypot in DMZ is externally deployed and configured by

Tool	Partner	Status
		CITRIX and FORTH
TaaS	EGM	Not integrated / not working right now
AngelEye	IBM	Integrated and working
CYSEC	FHNW	Plan to use

4.1.4 Requirements evaluation

Table 15: Requirements evaluation 1 - E-Voting - Scytl

Business-and Platform requirements	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	GravityZone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Availability	✓				/		/	/	/	/	
Usability			/								/
Privacy	✓	/			/		/				
Cost	✓										
Alerting						/	/				
Scalability			/		/	/	/	/	/	/	/
System integrity	✓			/		/		/	/	/	/
Confiden-tiality	✓	/	/		/		/				/
Non-repudiation		/		/	/	/	/				
Authen-tication	✓	/			/		/	/	/	/	/

Table 16: Requirement evaluation 2 - E-Voting Scytl

Protection capabilities	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	Gravity Zone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Web application servers	1	/			/	/	/	/	/	/	
Database servers	2	/			/	/	/	/	/	/	
Network traffic	3		/		/	/	/	/	/	/	
Web	4	/			/	/	/	/	/	/	

servers											
Email servers		/			/	/	/	/	/	/	
DDoS			/		/	/	/	/	/	/	
Access abuse			/			/					
Software misuse		/		/		/					
Zero-day attacks				/	/	/	/				/
Code injection		/	/	/	/	/	/				/
Man-in-the-Middle attacks		/	/	/	/	/	/				/

4.1.5 Testing procedure for technical tools

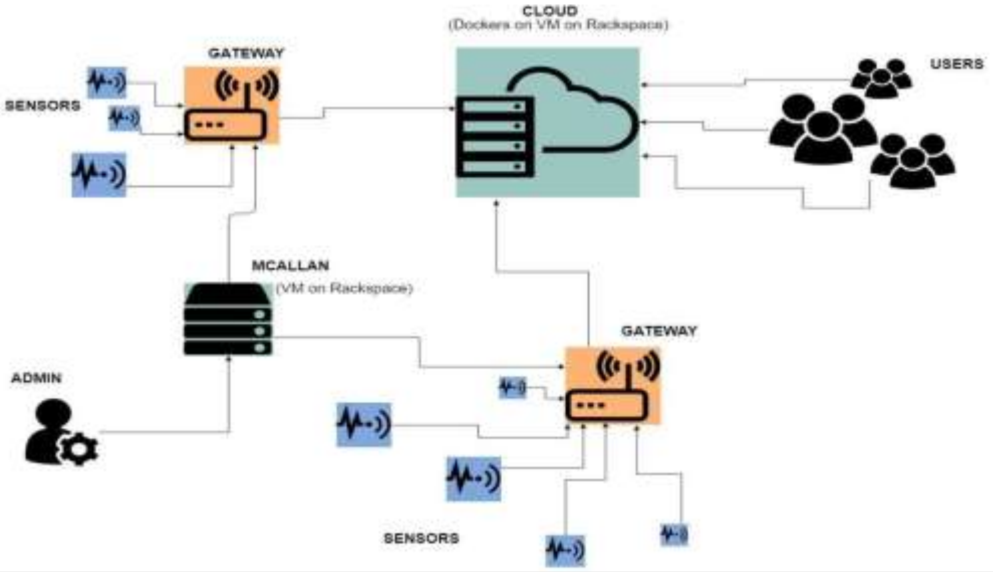
To test the SMESEC tools used in this E-Voting pilot the tests described in chapter 3 can be used. For this pilot, it is important to focus on the security in the DMZ and to ensure that the local network behind with the database cannot be reached in any ways from the DMZ using malicious connections. Since there is a webserver running, the credentials, authentication and sanitizing functionalities have to be tested and network attacks filtered or detected by the security features included in the pilot (see the definition of the different formal tests in section 5).

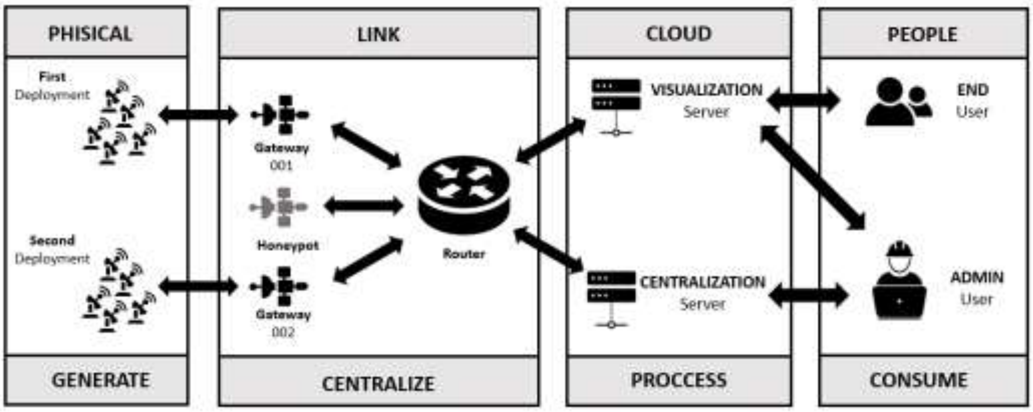
4.1.6 Validation protocol for social tools

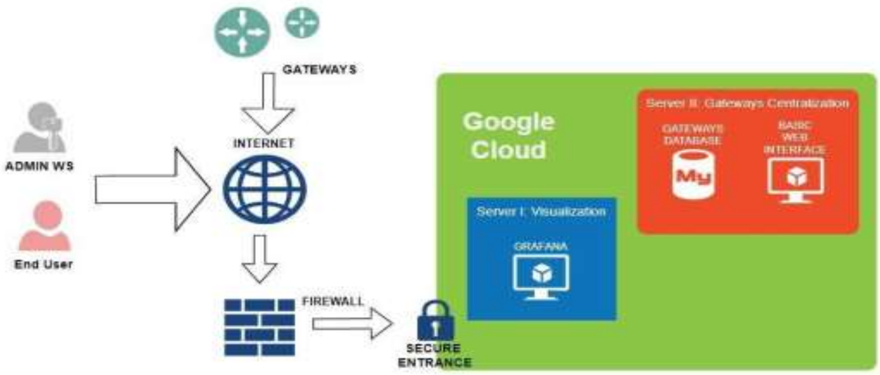
To test the social tool CYSEC, the project-wide protocol described in Table 11 will be applied.

4.2 Industrial Services – WoS

4.2.1 Pilot Architecture







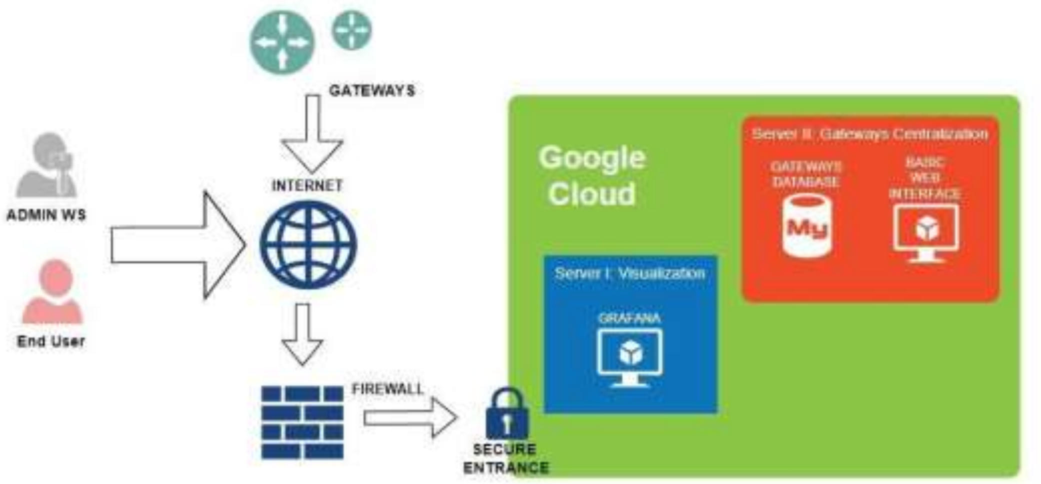


Figure 5: Pilot architecture - Industrial Services

4.2.2 Transition from lab testing system to demonstrator system

After finalizing the pilot, the architecture has been updated to the demonstration system as followed:

The whole infrastructure is deployed from the very beginning on the premises of the client, so all the hardware elements were all the time working in production environments.

In regard to the servers, as well as the hardware, were the same systems that were deployed in the first place. Yet, some testing and continuous monitoring is deployed, to be sure about the correct functioning of the whole system.

4.2.3 Security tools used

Table 17 : Security tools - Industrial services - WoS

Tool	Partner	Status
XL-SIEM	ATOS	Integrated and working
Bitdefender / GravityZone	Bitdefender	Integrated and working
CITRIX ADC	CITRIX	Provisionally stand by.
TaaS	EGM	Ongoing integrations.
AntiROP - Code analysis	IBM	Integrated and working
CYSEC	FHNW	Plan to use

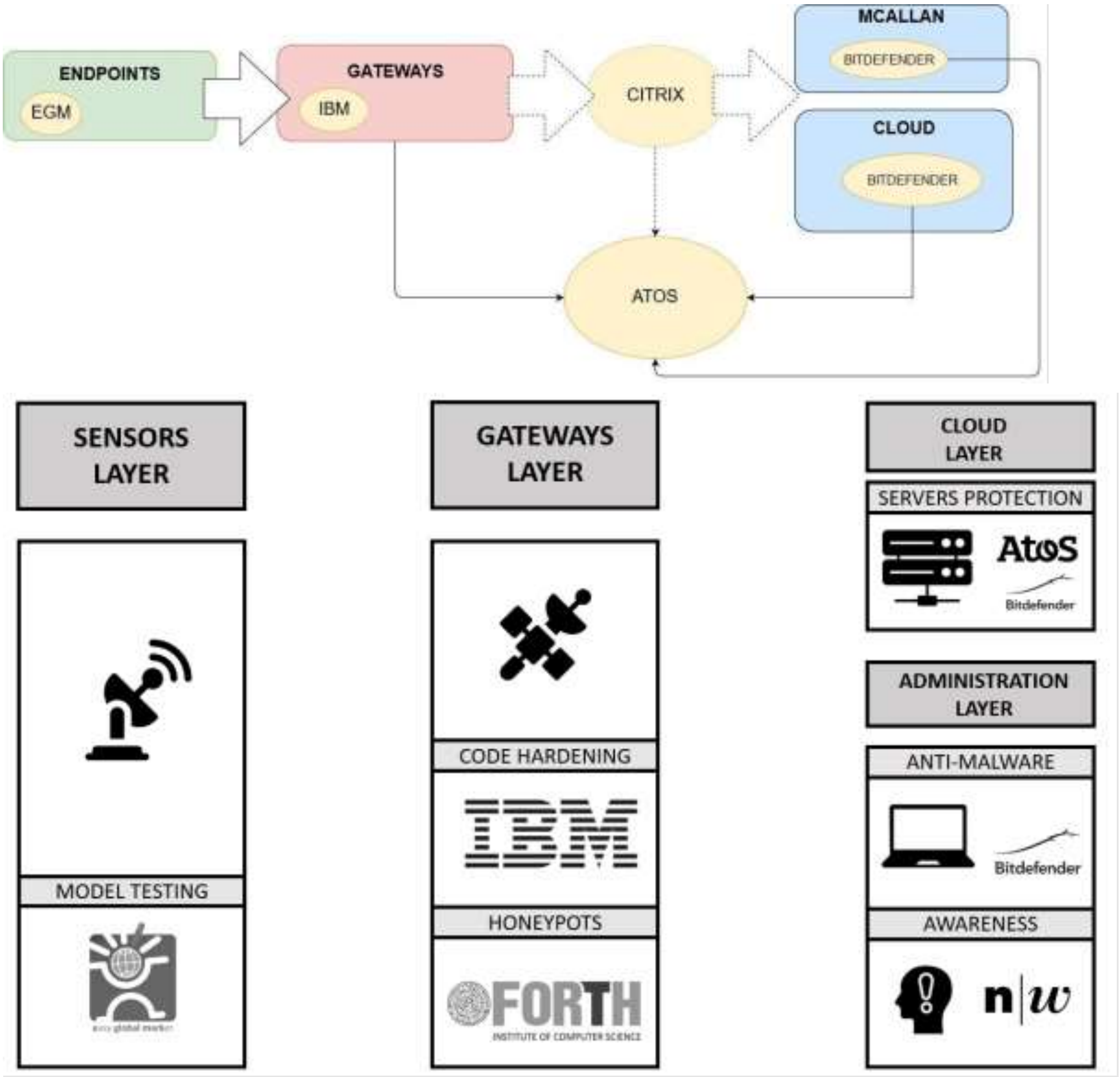


Figure 6: Schematic of the security tools in the Pilot III (Industrial Services) and interdependencies

4.2.4 Requirements evaluation

Table 18: Requirements evaluation 1 - Industrial services - WoS

Business-and Platform requirements	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	GravityZone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Availability	✓				/		/	/	/	/	
Usability	✓		/								/

Privacy	✓	/			/		/				
Cost	✓										
Alerting	✓					/	/				
Scalability	✓		/		/	/	/	/	/	/	/
System integrity	✓			/		/		/	/	/	/
Confiden-tiality		/	/		/		/				/
Non-repudiation	✓	/		/	/	/	/				
Authen-tication	✓	/			/		/	/	/	/	/

Table 19: Requirements evaluation 2 - Industrial Services

Protection capabilities	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	Gravity Zone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Web application servers	4	/			/	/	/	/	/	/	
Database servers		/			/	/	/	/	/	/	
Network traffic			/		/	/	/	/	/	/	
Web servers		/			/	/	/	/	/	/	
Email servers		/			/	/	/	/	/	/	
DDoS	5		/		/	/	/	/	/	/	
Access abuse	2		/			/					
Software misuse	1	/		/		/					
Zero-day attacks	6			/	/	/	/				/
Code injection	8	/	/	/	/	/	/				/
Man-in-the-Middle attacks	3	/	/	/	/	/	/				/

4.2.5 Testing strategy for technical tools

WoS aims to guarantee the functionality of the SMESEC framework through a set of specific tests. The objective is to check the protection against targeted attacks provided by the orchestrated operation of the different tools, as well as the usability and resilience of the system.

4.2.6 Validation protocol for social tools

Regarding the feedback obtained from the client, Worldsensing is already working on the implementation of measures to be able to comply with their requests. The two main aspects to cover were:

- Enlarge the coverage offered by the current monitoring system. Scope of the system was to monitor just half of the infrastructure and the administration of the place was asking to cover it all.
- Enhance the end user experience regarding the data displayed and the way it is displayed, especially after the enlargement with the new amount of data.
- To test the social tool CYSEC, the project-wide protocol described in Table 11 will be applied.

4.3 Sense.City – University of Patras

University of Patras has selected the following SMESEC security tools and solutions to protect its private cloud and sense.city platform. GravityZone (Antimalware-Antivirus) from Bitdefender, XL-SIEM from ATOS, EWIS (intrusion detection) from FORTH, the Code analysis tool (for javascript) from IBM, the TaaS tool from EGM and finally the Cloud Security solution from FORTH. From these tools/solutions, until M24, GravityZone, XL-SIEM, EWIS and cloud security are fully integrated, and can offer a general overview of the security status of the UOP infrastructure.

4.3.1 Architecture

The figure below describes Sense City architecture

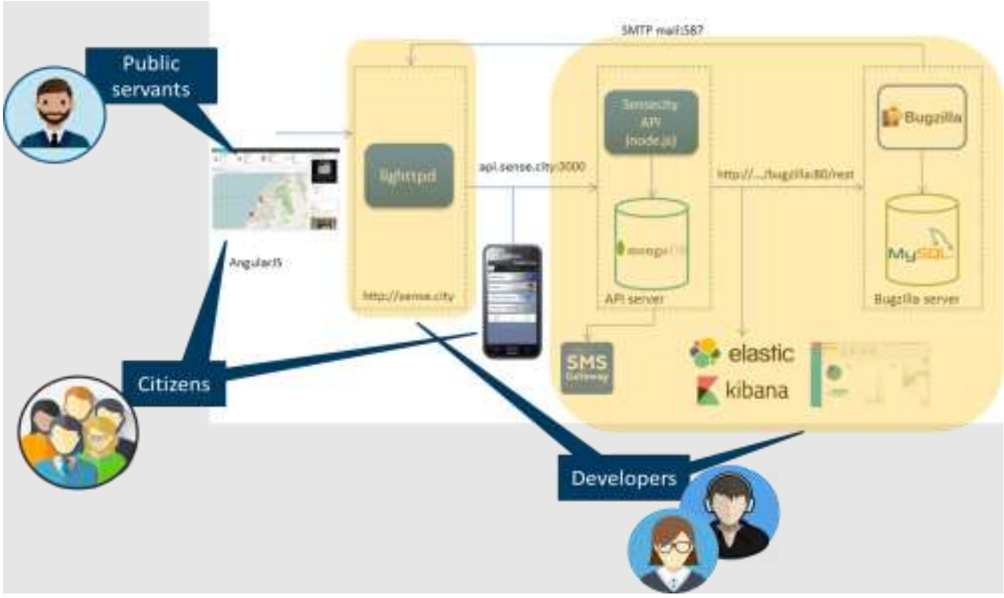


Figure 7 : Pilot architecture - Sense.city

4.3.2 Transition from lab testing system to demonstrator system

UOP has created a number of new Virtual Machines inside its private cloud in order to deploy the SMESEC security solutions and tools. It must be noted that, apart from its testing nodes, UOP has decided to adopt selected SMESEC's solutions in some of its operational nodes.

All evaluation tests will take place on the testing VMs and not on the sense.city's operational VMs so as to not jeopardize the normal operation of the system. We need to note that both testing and operational VMs exist inside the same cloud (UOP private cloud). For the cloud security solution, after discussions between UOP and FORTH, it was decided to not directly deploy it in the cloud since the whole process required the installation of components on operational nodes. To address this issue, UOP setup a new physical machine with the same hypervisor as its private cloud (clone). In this physical machine FORTH has successfully deployed its solution. The figure below demonstrates UOP's complete demonstrator system.

4.3.3 Security tools used

Table 20: Security tools - Sense.City - UoP

Tool	Partner	Status
XL-SIEM	ATOS	Integrated and working
Bitdefender / GravityZone	Bitdefender	Integrated and working
EWIS Honeypot	FORTH	Integrated and working
TaaS	EGM	Not integrated
AntiROP - Code	IBM	Not fully integrated, initial results.

Tool	Partner	Status
analysis		
Cloud - IDS	FORTH	Integrated and working
CYSEC	FHNW	Plan to use

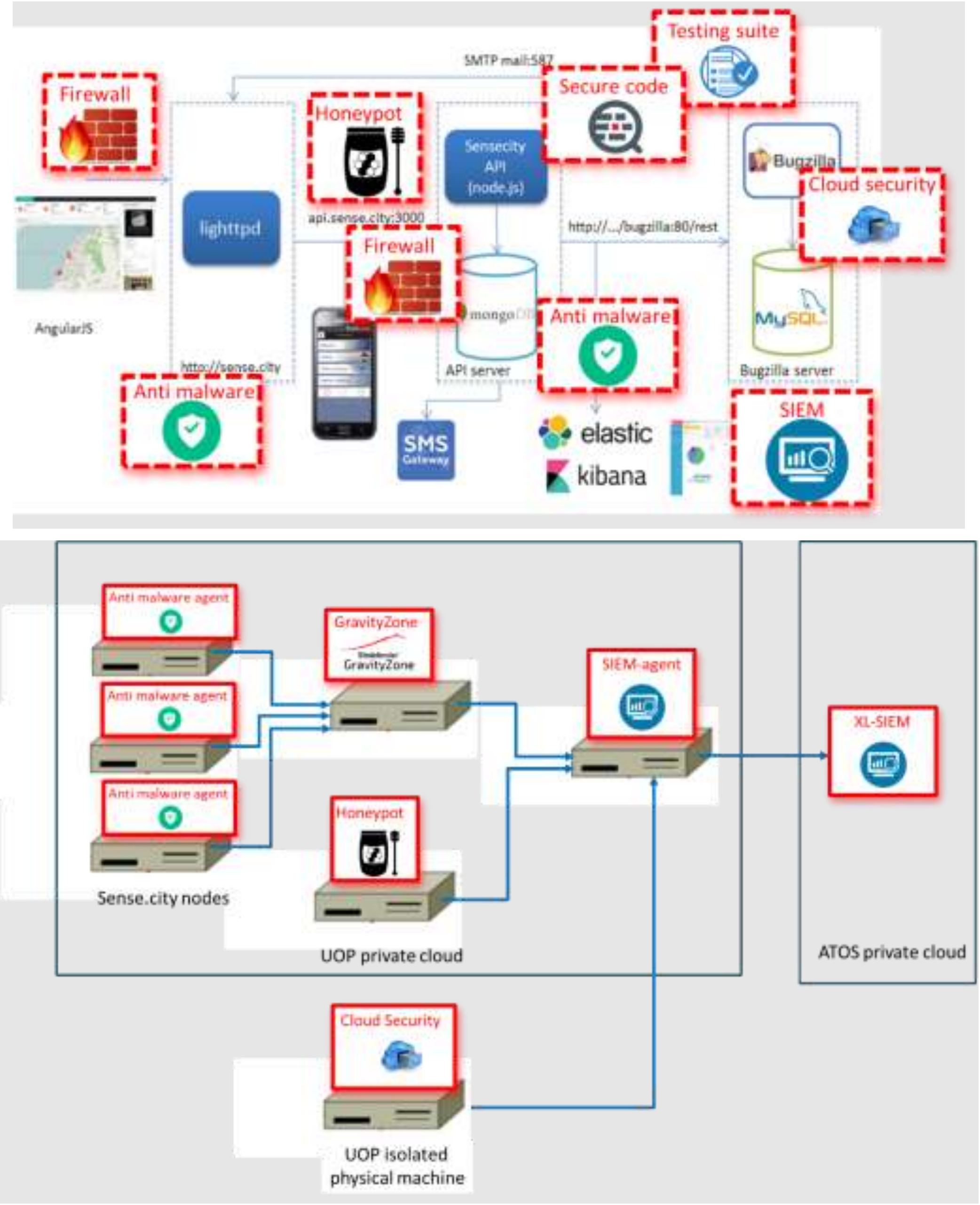


Figure 8 : SMESEC security tools adopted in Sense.city architecture

4.3.4 Requirements evaluation

Table 21: Requirements evaluation 1 - Sense.City - UoP

Business-and Platform requirements	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	GravityZone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Availability	✓				/		/	/	/	/	
Usability			/								/
Privacy		/			/		/				
Cost	✓										
Alerting						/	/				
Scalability			/		/	/	/	/	/	/	/
System integrity	✓			/		/		/	/	/	/
Confiden-tiality		/	/		/		/				/
Non-repudiation	✓	/		/	/	/	/				
Authen-tication	✓	/			/		/	/	/	/	/

Table 22: Requirements evaluation 2 - Sense.City - UoP

Protection capabilities	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	Gravity Zone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Web application servers	1	/			/	/	/	/	/	/	
Database servers		/			/	/	/	/	/	/	
Network traffic	5		/		/	/	/	/	/	/	
Web servers		/			/	/	/	/	/	/	
Email servers	3	/			/	/	/	/	/	/	
DDoS	1		/		/	/	/	/	/	/	
Access abuse			/			/					
Software		/		/		/					

misuse											
Zero-day attacks				/	/	/	/				/
Code injection		/	/	/	/	/	/				/
Man-in-the-Middle attacks		/	/	/	/	/	/				/

4.3.5 Testing Strategy

UOP testing strategy is divided in two phases. The first phase is testing SMESEC individual tools, while the second one involves the testing of the integration between the installed solutions. A set of preliminary tests have already taken place for both phases during installation to identify whether the deployment of SMESEC tools was successful. The main testing plan that has already been constructed and presented in D4.3 will be followed in WP5 includes concrete tests for both individual and integrated tools (see the description of the tests in section **¡Error! No se encuentra el origen de la referencia.**).

4.3.6 Validation protocol for social tools

To test the social tool CYSEC, the project-wide protocol described in Table 11 will be applied.

4.4 SmartGrid – GridPocket

4.4.1 Architecture

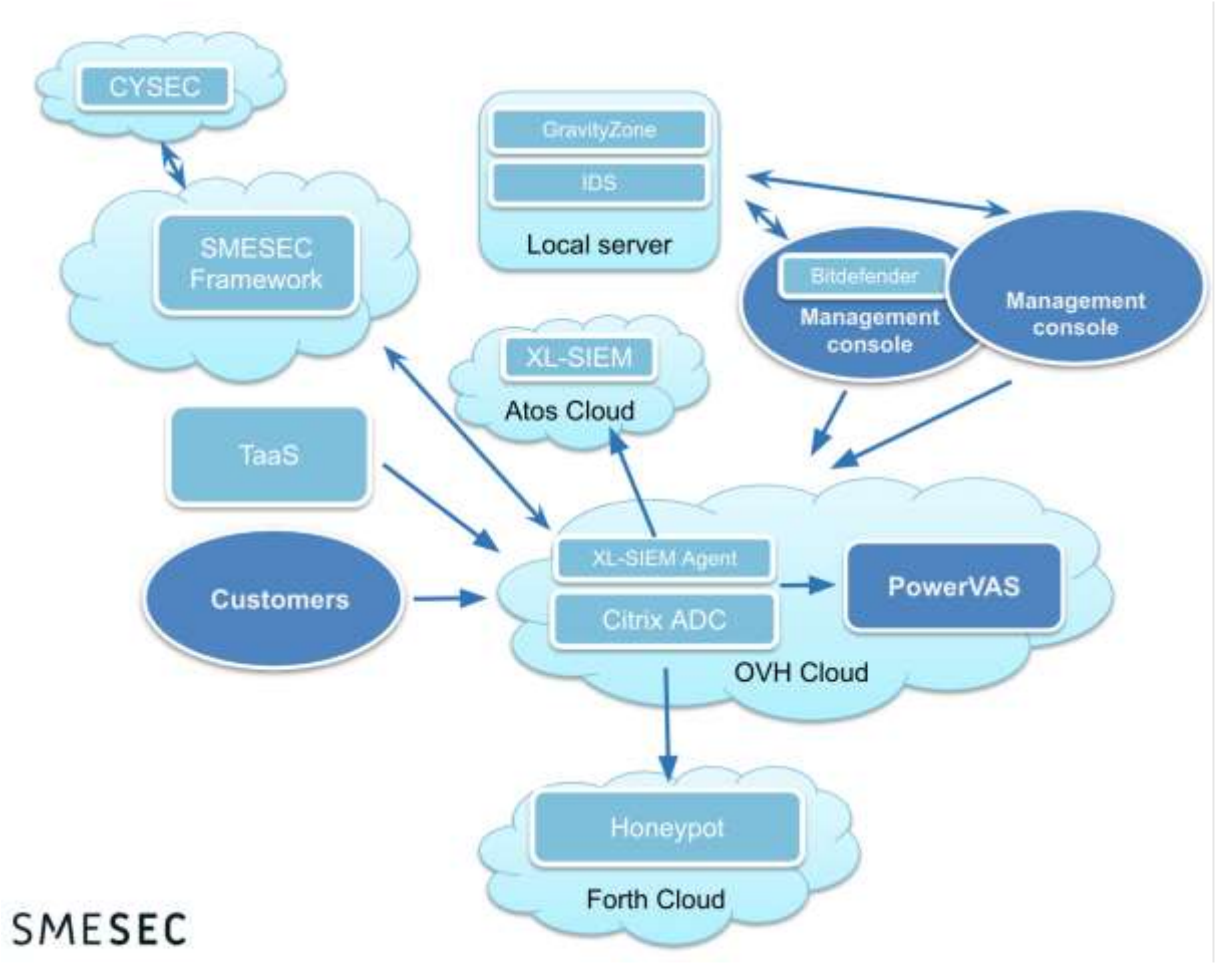


Figure 9: Pilot architecture - Smart Grid

4.4.2 Transition from lab testing system to demonstrator system

In case of Smart Grid Pilot, lab testing system architecture is very similar, or even identical to demonstrator system. Our systems are always hosted on OVH cloud, and consist of several servers, which take different roles - Database, backend API servers and User Interface servers. Only possible change is scalability of the system, but it should not affect implementation of SMESEC framework in any way.

4.4.3 Security tools used

Main component of the Smart Grid Pilot - PowerVAS platform is hosted on OVH cloud. In the same cloud resides Citrix ADC WAF with XL-SIEM agent and Honeypot connected to it. All traffic travels through ADC firewall, XL-SIEM agent sends logs to XL-SIEM cloud, and any suspicious traffic is redirected to Honeypot hosted on FORTH cloud. Authentication security is validated by TaaS. Developers are connecting to Pilot infrastructure using computers secured by Bitdefender Antivirus, which in turn is connected to GravityZone Console. The same server hosts Cloud-IDS￼￼ the SMESEC Framework, which also gives an access to SMESEC framework.

At the current time, final Citrix ADC and Honeypot implementation is incomplete, but GRID hopes to finish it with the help of Citrix before end of June.

Table 23: Security tools - SmarGrid - GRIDP

Tool	Partner	Status
XL-SIEM	ATOS	Integrated and working
Bitdefender GravityZone	Bitdefender	Integrated and working
CITRIX ADC	CITRIX	Installed, final configuration stage.
EWIS Honeypot	FORTH	Under development
TaaS	EGM	Integrated and working
Cloud-IDS	FORTH	Integrated and working
CYSEC	FHNW	Used

4.4.4 Requirements evaluation

Table 24: Requirements evaluation 1 - SmartGrid - GRIDP

Business-and Platform requirements	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	GravityZone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Availability	✓				/		/	/	/	/	
Usability			/								/
Privacy	✓	/			/		/				
Cost	✓										
Alerting	✓				/		/				
Scalability	✓		/		/	/	/	/	/	/	/
System integrity	✓			/		/		/	/	/	/
Confiden-tiality	✓	/	/		/		/				/
Non-repudiation	✓	/		/	/	/	/				

Authen-tication	✓	/			/		/	/	/	/	/

Table 25: Requirements evaluation - SmartGrid - GRIDP

Protection capabilities	Required	Evaluation (Fulfilled in % 0 - 100)									
		XL-SIEM	Gravity Zone	Citrix ADC	EWIS Honeypot	CY SEC	TaaS	Anti ROP	Angel Eye	Expli SAT	Cloud-IDS
Web application servers	4	/			/	/	/	/	/	/	
Database servers		/			/	/	/	/	/	/	
Network traffic			/		/	/	/	/	/	/	
Web servers		/			/	/	/	/	/	/	
Email servers		/			/	/	/	/	/	/	
DDoS	1		/		/	/	/	/	/	/	
Access abuse			/			/					
Software misuse		/		/		/					
Zero-day attacks				/	/	/	/				/
Code injection	2	/	/	/	/	/	/				/
Man-in-the-Middle attacks	3	/	/	/	/	/	/				/

4.4.5 Testing strategy

The most important parts of testing strategy for Smart Grid are user authentication, data flow protection and developer device protection. This is why initial tests focus on testing TaaS and Bitdefender Antivirus. Except that, all previously installed tools were initially tested, as it was described in D4.8.

4.4.6 Validation protocol for social tools

Cybersecurity expert of Gridpocket tested and evaluated CYSEC tools in the presence of FHNW representative. He found CYSEC very useful in the process of assessing general level of security in the company. He noted that some questions would be too specific for not-expert in IT security. Overall rating of the tool was positive.

To test the social tool CYSEC, the project-wide protocol described in Table 11 will be applied.

5 Definition of proof of concept scenarios

Following table provides an overview of the trial scenarios which can be used to evaluate the components which are integrated in the SMESEC framework. The tests are only for the external SMESEC tools, and not include pure framework modules, such as SMESEC HUB. Trials are associated with both individual and cooperative with other modules. The idea of these tests is to check if modules are efficient and they have suitable inputs and outputs from and to other modules.

Table 26: Planned tests for SMESEC framework

Trials			
Test-Codes	**I/ J**	**Provider**	**Description**
IT_01_XL-SIEM	Individual	ATOS	General test of relevant alerts
IT_01_2_XL-SIEM	Individual	ATOS	Test of test plugin
IT_01_3_XL-SIEM	Individual	ATOS	Test of SSH plugin
IT_01_4_XL-SIEM	Individual	ATOS	Test of FORTH EWIS plugin
IT_01_5_XL-SIEM	Individual	ATOS	Test of ADC plugin
IT_02_1_GravityZone	Individual	Bitdefender	Malware detection in clients and servers, deployment and detection of test malware, alerts in relation to detected malware send and represented in GravityZone
IT_02_2_GravityZone	Individual	Bitdefender	Detection of downloaded malware
IT_02_3_GravityZone	Individual	Bitdefender	Accessing a blacklisted URL
IT_02_4_GravityZone	Individual	Bitdefender	Inserting an USB stick with a malicious file
IT_02_5_GravityZone	Individual	Bitdefender	Detection of port scanning
IT_03_1_Honeypot	Individual	FORTH	Detection of DDoS attack
IT_03_1_Honeypot	Individual	FORTH	Detection of SQL-Injection attack
IT_04_1_AntiROP	Individual	IBM	Validate that antiROP unique copies do not change executable functionality
IT_04_2_AntiROP	Individual	IBM	Validate that antiROP unique copies defend against ROP attack
IT_05_1_TaaS	Individual	EGM	Lora testing
IT_05_2_TaaS	Individual	EGM	API testing

IT_05_3_TaaS	Individual	EGM	Check if user is authorized to access the TaaS platform
IT_05_4_TaaS	Individual	EGM	Show all reports
IT_06_CITRIX-ADC	Individual	CITRIX	Detects malicious or improper network traffic and blocks it before reaching the backend application servers, potentially causing service downtime. stops it
IT_07_1_IDS	Individual	FORTH	Scanning detection
IT_07_2_IDS	Individual	FORTH	DDoS attack detection
IT_08_1_Virtual_Patching	Individual	IBM	Validate that the predictive model provides reasonable FPR/TPR rates on input-samples
IT_08_2_Virtual_Patching	Individual	IBM	Validate that the Integration into custom log file analysis produces the same results as in T_08_01_Virtual_Patching
IT_09_1_CYSEC	Individual	FHNW	Validation of the installation and login functionality of the CYSEC tool
IT_09_2_CYSEC	Individual	FHNW	Validation of the on boarding, assessment, learning, control and practice implementation, reporting, and recommendation functionalities of the CYSEC tool
IT_09_3_CYSEC	Individual	FHNW	Validation of CYSEC coaches
IT_09_4_CYSEC	Individual	FHNW	Validation of the insight stream functionality of the CYSEC tool
IT_10_1_ExpliSAT	Individual	IBM	Validate that testing platform does not produce false alerts
IT_10_2_ExpliSAT	Individual	IBM	Validate that testing platform covers common vulnerability families
JT_01_XL-SIEM_GravityZone	Joint	ATOS & Bitdefender	Malware detection, reporting on the XL-SIEM system and alerts rising
JT_02_XL-SIEM _Honeypot	Joint	ATOS & FORTH	Possible attacks on the honeypot reported on the XL-SIEM system
JT_03_CITRIX-ADC_Honeypot	Joint	CITRIX & FORTH	Citrix ADC is deployed in front of an application server and intercepts all inbound traffic. Traffic is inspected based on predefined policies and discarded if found inappropriate. Inappropriate traffic is forwarded to the Honeypot while generic reports are issued to

			the XL-SIEM.
JT_04_XL-SIEM_IDS_Honeypot	Joint	ATOS & FORTH	The Cloud-IDS and Honeypot detect a DoS attack and reports the XL-SIEM about it

5.1 Definition of experiments

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

ATOS_XL-SIEM

Regarding the XL-SIEM tests, all tests can be performed with the following command:

```
> java -jar xl-siem-logger-6.0.0.jar --command=test-all
```

IT_01_1_XL-SIEM	
Objective:	Test whether the XL-SIEM agent is well connected to the XL-SIEM server
Test definition:	The XL-SIEM logger will send a test message via syslog to the agent
Success criteria:	The event is shown in the events panel of the XL-SIEM server

IT_01_2_XL-SIEM	
Objective:	Test whether the ssh plugin is well configured
Test definition:	The XL-SIEM logger will send a test message via syslog to the agent
Success criteria:	The event is shown in the events panel of the XL-SIEM server

IT_01_3_XL-SIEM	
Objective:	Test whether the FORTH EWIS plugin is well configured
Test definition:	The XL-SIEM logger will send via syslog examples of all the type of events that FORTH EWIS can send to the XL-SIEM agent server.
Success criteria:	The events are shown in the events panel of the XL-SIEM server

IT_01_4_XL-SIEM	
Objective:	Test whether the Gravity Zone plugin is well configured
Test definition:	The XL-SIEM logger will simulate all possible log types of the Gravity Zone console to the XL-SIEM agent server.

Success criteria:	The events are shown in the events panel of the XL-SIEM server

IT_01_5_XL-SIEM	
Objective:	Test whether the ADC plugin is well configured
Test definition:	The XL-SIEM logger will simulate all possible log types of ADC to the XL-SIEM agent server.
Success criteria:	The events are shown in the events panel of the XL-SIEM server

BITDEFENDER

IT_02_1_GravityZone	
Objective:	Detect the presence of malware within one of the protected hosts. Provide appropriate reaction to the attack, send alert to GravityZone.
Test definition:	A malware is introduced in the host using some known exploitation. The BD Endpoint protection is meant to detect the attack and perform appropriate actions, log entries etc. and sends an alert to the GravityZone.
Success criteria:	Detection of the malware and proper reaction as deletion of the file, quarantine or similar.

IT_02_2_GravityZone	
Objective:	Test if the endpoints are protected from malware downloaded from the Internet.
Test definition:	During the test, a "malware" file will be downloaded from the Internet on one of the endpoints. To prevent the risk of an actual infection, the downloaded file is not actually malware, but a specially crafted, benign file, that should be, by standard, detected by all anti-virus products. This file is the EICAR test file and can be accessed from this url: http://eicar.org/download/eicar.com. In order to perform the test, we have the following steps: ● Download the EICAR test file from the link above, using a browser or a command-line tool like wget or curl ● Check that the file is blocked ● Check that the detection is reported in the GravityZone dashboard Check that the detection is reported in the XL-SIEM dashboard
Success criteria:	● The download of the file is blocked ● The detection is reported in the GravityZone dashboard ● The detection is reported in the XL-SIEM dashboard

IT_02_3_GravityZone	
Objective:	Test if Bitdefender prevents the protected endpoints from accessing blacklisted URLs.
Test definition:	During the test, some blacklisted URLs will be accessed, either from a browser or from the command line. We will use some benign URLs that are blacklisted by Bitdefender for testing purposes, in order to avoid the risk of an actual infection: http://bitdefender-testing.com/malware http://bitdefender-testing.com/phishing In order to perform the test, we have the following steps: ● Access the links above using a browser or a command-line tool like wget or curl ● Check that the URLs are blocked ● Check that the detection is reported in the GravityZone dashboard ● Check that the detection is reported in the XL-SIEM dashboard
Success criteria:	● The URLs are blocked ● The detection is reported in the GravityZone dashboard ● The detection is reported in the XL-SIEM dashboard

T_02_4_GravityZone	
Objective:	Test if the endpoints are protected from malware distributed through USB drives.
Test definition:	During the test, a USB stick with a "malware" file will be inserted in one of the protected computers. To prevent the risk of an actual infection, we will use the EICAR test file, that can be downloaded from this url: http://eicar.org/download/eicar.com. The file needs to be downloaded and copied on the USB stick from a machine not protected by an antimalware solution, otherwise it will be detected and deleted. In order to perform the test, we have the following steps: ● Use a machine with no anti-malware protection or with the anti-malware protection turned off ● Download the EICAR test file from the link above ● Check that the file is detected ● Check that the detection is reported in the GravityZone dashboard ● Check that the detection is reported in the XL-SIEM dashboard
Success criteria:	● The EICAR test file from the USB stick is detected ● The detection is reported in the GravityZone dashboard ● The detection is reported in the XL-SIEM dashboard

IT_02_5_GravityZone	
Objective:	Test if port scanning attacks are detected by Bitdefender
Test definition:	An attacker machine will run a port scanning tool (e.g. *nmap*) in order to scan the tested machine's ports. In order to perform the test, we have the following steps: ● Prepare an "attacker" machine with *nmap* installed, that should be in the same

	network like the "victim" machine, that should run Windows and be protected by Bitdefender ● Run nmap on the "attacker" machine and scan the ports of the "victim" machine ● Check that the attack is detected by the victim machine ● Check that the detection is reported in the GravityZone dashboard
Success criteria:	● The port scanning attack is detected ● The detection is reported in the GravityZone dashboard ● The detection is reported in the XL-SIEM dashboard

FORTH - Honeypot

IT_03_1_Honeypot	
Objective:	The detection of DDoS attacks attempts will be tested.
Test definition:	Within this test we aim to evaluate the DDoS detection component of the EWIS solution. This will be achieved by simulating amplification DoS attacks, against the EWIS' honeypots, using UDP and ICMP flood attacks with the hping3 tool. Url: http://www.hping.org/ hping is a packet generator and can be used to test for various network security tests. ● Install hping in the attacker's machine ● Using hping generate large volumes of ICMP traffic against the victim's machine. ● Repeat the same attack with UDP against the victims machine
Success criteria:	● The DDoS attack is detected by the honeypot ● The DDoS attack is successfully reported (either graphically or through raw text format) in the XL-SIEM. ● Attack visualized in the SMESEC dashboard

IT_03_2_Honeypot	
Objective:	Test the detection of attacks targeting database services
Test definition:	The SMESEC framework will be able to detect SQL injection and other attacks aiming the dataset of the server. This will be done through the honeypot solution that is able to detect and report this kind of attack attempts back to the XL-SIEM. In addition, the results will be available to the system administrator via the SMESEC dashboard. To accomplish this attack, we will use SQLmap which is a known penetration testing tool. Url: http://sqlmap.org/ ● First, we must install sqlmap in the attacker's machine. ● Then will use sqlmap to scan for urls which are vulnerable to sql injection and similar attacks.
Success criteria:	● The database attack is detected by the Honeypot ● The database attack is successfully reported (either graphically or through raw text format) in the XL-SIEM. ● Attack visualized in the SMESEC dashboard

IT_03_3_Honeypot	
Objective:	The detection of brute force attacks attempts will be tested to our honeypot from an outside source
Test definition:	We want to test the detection of attacks from external sources against our ssh honeypot part of EWIS. To do that we will use hydra. Url: https://github.com/vanhauser-thc/thc-hydra Hydra is a pentesting tools used for brute force attacks. We will use it to commit a dictionary attack against the SSH port of our honeypot. ● Install the hydra-gtk package in the attacker's machine ● Perform brute force dictionary attack against our honeypot using hydra targeting our SSH port
Success criteria:	● The Brute force attack is detected by the Honeypot ● The Brute force attack is successfully reported (either graphically or through raw text format) in the XL-SIEM. ● Attack visualized in the SMESEC dashboard

IBM

IT_04_1_AntiROP	
Objective:	Validate that antiROP unique copies do not change executable functionality
Test definition:	● Compile two unique executables from the same source ● Run testing suit of the source ● Validate all results are identical
Success criteria:	Results are identical

IT_04_2_AntiROP	
Objective:	Validate that antiROP unique copies defend against ROP attack
Test definition:	● Insert a vulnerability into a given toy-example source code **(IBM to send example of a vulnerability)**, ● compile one executable from this source, ● create a ROP attack for this executable, ● launch attack and validate that it succeeds, ● compile a different unique copy of the source code, ● launch attack and validate that it fails
Success criteria:	Results are identical

EGM

IT_05_1_TaaS	
Objective:	Lora testing: This form of testing is focused around two key areas: The interoperability between and the End Device and the Network (Gateway and Network Server) and the conformance according to the predefined/standardized specification, including security functions for OTAA and BGP provisioning.
Test definition:	Browse to the LoRaWAN tab in the TaaS platform. Enter the LoRa device (SUT) configuration (direct input or an upload). Choose the "LoraWAN testcases" to be executed. Push the run button.
Success criteria:	Returns a test report: the user is notified that a new test report is available. So, he/she can navigate to report section to see the check report.

IT_05_2_TaaS	
Objective:	API testing: This form of security testing concentrates on using software to make API calls in order to receive an output before observing and logging the system's response.
Test definition:	Browse to the API test tab in the TaaS platform. Enter the web server (system under test) configuration (direct input or an upload). Choose the "API testcases" to be executed. Push the run button.
Success criteria:	Returns a test report: the user is notified that a new test report is available. So he/she can navigate to report section to see the check report.

IT_05_3_TaaS	
Objective:	Check if user is authorized to access the TaaS platform.
Test definition:	Put the confidential in the keycloack login page. Push the button login
Success criteria:	The user is authenticated and redirected to the TaaS platform. The username appears in the navigation bar on the TaaS frontend.

IT_05_4_TaaS	
Objective:	Show the all reports
Test definition:	Browse to the reports tab in the TaaS platform.
Success criteria:	A Web Interface with all generated reports.

CITRIX ADC

IT_06_1_CITRIX_ADC	
Objective:	Verify proper configuration of all internal Citrix ADC services, entities and networking topologies.
Test definition:	This test intends to verify that all internal Citrix ADC services, entities and networking topologies are properly configured according the provided provisioning material. This test assumes that a network topology as well as scenario definition is available before starting the test. The necessary steps and CLI commands are listed in Annex 8.1
Success criteria:	Citrix ADC vservers, internal and external IPs, and services are available and properly configured.

IT_06_2_CITRIX_ADC	
Objective:	Initiate malicious traffic towards a backend server protected by Citrix ADC and verify that the later is able to intercept the traffic (despite being encrypted or not) and enforce proper actions according predefined policies.
Test definition:	An end-to-end testing of the overall Citrix ADC functionality deployed under the auspices of SMESEC, can be done through verifying that malicious traffic is not forwarded to systems protected by the specific solution. A virtualized Citrix ADC node is collocated with the server it protects, intercepts all ingress traffic streams and categorizes requests based on pre-defined policy rules. All incoming requests that are not aligned with the aforementioned policy rules are considered as malicious, are effectively blocked and finally discarded by the system thus pose no threat to the backend infrastructure. This test assumes that service owners have (i) upload all necessary certificates to Citrix ADC, rendering capable of “legally” intercepting encrypted traffic and (ii) have defined specific policies based on which traffic is categorized into malicious or benevolent as well as the necessary action in each case. The necessary steps and CLI commands are listed in Annex 8.2
Success criteria:	Citrix ADC intercepts all traffic and efficiently categorizes it to malicious or not, as well as take necessary actions in each case.

FORTH- Cloud-IDS

IT_07_1_IDS	
Objective:	Test the detection of network scan between VMs in the cloud.

Test definition:	Perform Xmas tree scan with nmap from one VM in the cloud to the other and see if it gets detected by snort running on the hypervisor. Then make sure that it gets reported to the SMESEC dashboard and the XL-SIEM. Url: https://nmap.org/ ● Both VMs need to be connected to the network ● First install nmap on one of the VMs in the cloud ● Then run nmap against a second VM which is the target and use the -sX flag to perform a Xmas tree scan.
Success criteria:	● The network scan attack is detected by snort ● The network scan is successfully reported (either graphically or through raw text format) in the XL-SIEM. ● Scan visualized in the SMESEC dashboard

IT_07_2_IDS	
Objective:	Detect DDoS attacks between VMs in the cloud
Test definition:	DDoS attack designed and executed by One VM in the cloud to the other. The DDoS attack must be detected and reported to the XL-SIEM and the SMESEC dashboard. Url: http://www.hping.org/ hping is a packet generator and can be used to test for various network security tests. ● Both VMs need to be connected to the network ● First install hping in one of the VMs in the cloud. ● Then generate large bursts of high traffic (ICMP or UDP or TCP) from the VM doing the attack to the other using hping At least 1000 packets in --flood mode. $ hping3 --icmp --flood -c 1000 [target]
Success criteria:	● The DDoS attack is detected by snort ● The DDoS is successfully reported (either graphically or through raw text format) in the XL-SIEM. ● Attack visualized in the SMESEC dashboard

IBM

IT_08_1_Virtual_Patching	
Objective:	Validate that the predictive model provides reasonable FPR/TPR rates on input-samples
Test definition:	● Create a large set of sample input to an application (using the same techniques for data generation) ● Run the predictive model server for this application ● Make sure FPR/TPR rates are as expected/configured
Success criteria:	FPR/TPR rates are expected / configured

IT_08_2_Virtual_Patching	
Objective:	Validate that the Integration into IDS produces the same results as in T_08_01_Virtual_Patching
Test definition:	● Create a large set of sample input to an application ● Send those samples to log file analyzes ● Make sure FPR/TPR rates are as expected/configured
Success criteria:	FPR/TPR rates are expected / configured

FHNW

All tests related to CYSEC tool are valid only for CYSEC installed on premise. In case of using cloud-based CYSEC, no tests are required.

IT_09_1_CYSEC	
Objective:	Validation of the installation and login functionality of the CYSEC tool on-premise. This test validates the installation of the appliance, the installation of the coaches, the email server configuration, and proper working of the CySeC infrastructure.
Prerequiste:	CySeC is installed on-premise and self-subscription is activated. All Tests during the installation phase were successful
Test definition:	1. User opens up the login page → A login prompt for username and password is shown. → If self-subscription was enabled, an enrol link is show 2. User enters username and clicks on “forgot password” →An email is sent to the user with a recovery link 3. User clicks on the recovery link in the email →A recovery page with a password prompt (2x) is shown 4. User enters a password fulfilling the complexity mentioned on the recovery page twice. → User receives confirmation of password change → User is redirected to the login page 5. User enters username and password and presses login → User is redirected to the dashboard → dashboard shows either the SMESEC company coach (when working with the SMESEC appliance) or an introductory coach (when having installed a bare bone CySeC)
Success criteria:	All criterias marked with “→” in the test definition have been met

IT_09_2_CYSEC	
Objective:	Validation of the onboarding, assessment, learning, control and practice implementation, reporting, and recommendation functionalities of the CYSEC tool. Please note: This onboarding process is specific to on-premise installation. The cloud installation has a different, joint onboarding process for new users.
Prerequisite:	CySeC installed and tested. An Admin account for the installed instance. Email of a user to be onboarded and access to that post box.
Test definition:	1. Login with the admin user → Dashboard is shown 2. Press the admin button on the top right → A list of users already entered is shown 3. Press the add button and enter firstname, lastname, email, and username into the fields and submit → The user is added to the list → The user receives an onboarding email with a password recovery link 4. User clicks on the recovery link in the email →A recovery page with a password prompt (2x) is shown 5. User enters a password fulfilling the complexity mentioned on the recovery page twice. → User receives confirmation of password change → User is redirected to the login page 6. User enters username and password and presses login → User is redirected to the dashboard → User is shown with exactly the same recommendations as the admin
Success criteria:	All criteria marked with "→" in the test definition have been met

IT_09_3_CYSEC	
Objective:	Validation of installation of CYSEC coaches
Prerequisite:	CySeC installed and tested. SMESEC coaching package is installed (always when working with appliance). A username and password tuple.
Test definition:	1. Login with the user → Dashboard is shown → The Dashboard shows at least the SMESEC Company Coach as available coach for the user (more is OK)
Success criteria:	All criteria marked with "→" in the test definition have been met

IT_09_4_CYSEC	
Objective:	Validation of the insight stream functionality of the CYSEC tool (only cloud version). This Test validates that MQTT messages are being delivered and that CySeC is

	integrated into the SMESEC Framework
Prerequisite:	Console access to CySeC server.
Test definition:	1. Open two consoles to the CySeC sever 2. On console one open CySeC log file in follow mode with the following command: tail –f /var/log/cysec/cysec.log |egrep –i “(starting|set|succeeded)” 3. Go to the second console and restart tomcat (systemctl restart tomcat8) 4. Go back to the first console and look for the line indicating that CySeC is starting → should find “Starting CySeC” → Should find after the previous line “setting up MQTT” → Should find after the previous line “selftest MQTT succeeded” 5. Log into the SMESEC framework page → You should see the CySeC recommendations (at least one)
Success criteria:	All criteria marked with “→” in the test definition have been met

IBM

IT_10_1_ExpliSAT	
Objective:	Validate that testing platform does not produce false alerts
Test definition:	Insert supported vulnerabilities into a source code. Run testing-platform to get alerts on this code
Success criteria:	Validate that most of the vulnerabilities introduced in the code are found

IT_10_2_ExpliSAT	
Objective:	Validate that testing platform covers common vulnerability families
Test definition:	Insert supported vulnerabilities into a source code. Run testing-platform to get alerts (tests leading to vulnerability) on this code.
Success criteria:	Validate that no alerts are false-positive (run each test on the code)

Combined tests

JT_01_XL-SIEM_GravityZone	
Objective:	Bitdefender detects a malware attack and forwards log entries to the XL-SIEM to generate alerts and notifications
Test definition:	Introduce a Malware in the test platform and generate log entries through Bitdefender

	endpoint protection. The entries are captured by the XL-SIEM tool and produce alert messages that are visually evident to the security administrators
Success criteria:	Simultaneous detection and alarm triggering

JT_02_XL-SIEM_Honeypot	
Objective:	Honeypot detects a network attack (e.g. A DDoS or intrusion attack) and forwards log entries to the XL-SIEM to generate alerts and notifications
Test definition:	An external entity performs a network security attack (e.g. A DDoS or intrusion attack) on the platform servers. The Honeypot should detect the attack and generates log entries. The entries are captured by the XL-SIEM and produces alert messages that are visually evident to the security administrators
Success criteria:	Simultaneous detection and alarm triggering

JT_03_CITRIX_ADC_Honeypot_XL-SIEM	
Objective:	CITRIX ADC detects traffic from an unauthorised range and forward it to the honeypot. Honeypot detects type of attack and give alert to the administrator and both send information to the XL-SIEM.
Test definition:	SQL injection attack designed and executed by external to an entity in the protected network area.
Success criteria:	Simultaneous detection and alert triggering reported to the XL-SIEM

JT_04_XL-SIEM_IDS_Honeypot	
Objective:	Cloud-IDS and Honeypot detects a network attack and report it to the XL-SIEM
Test definition:	Generate an internal DoS attack on the Honeypot and wait that also the Cloud-IDS react to this type of attack and send the report about it to the XL-SIEM.
Success criteria:	XL-SIEM is representing the two reports about the DoS attack.

5.2 Refine planning for trial

This document will be used for the tasks T5.1 “System readiness for validation activities” and T5.2 “Prototype Demonstration: Field trial results” as a guideline tool for the evaluation of the SMESEC framework.

6 Conclusions

This document describes the activities fulfilling requirements of deliverable D5.1 “Trial scenario definitions and evaluation methodology specification”. It covers definitions of testing strategy for each tool used by SMESEC framework: XL-SIEM and XL-SIEM agent, Bitdefender Endpoint Security and GravityZone server, CITRIX ADC, EWIS Honeypot, Cloud-based IDS, TaaS, Virtual Patching (AngelEye), Testing Platform (ExpliSat), Moving Target (AntiROP) and CYSEC Cybersecurity Coach.

The different tests target the specificities of the four different pilots and cover a very large area of security vulnerabilities such as: Information Gathering, Social Engineering, Phishing, Scanning, Sniffing, Spoofing, Man-in-the-middle, Exploitations: Buffer Overflow, SQL – Injection, XSS, …

Testing methodologies are defined in detail in the document with the final goal to provide a correct validation of the SMESEC framework in rich and realistic pilot environments.

References

[1] Runeson, Per, and Martin Höst, (2009), Guidelines for conducting and reporting case study research in software engineering, Empirical Software Engineering, 14(2), 131-164.

[2] Creswell, John W., Cheryl N. Poth, (2017), Qualitative inquiry and research design: Choosing among five approaches, Sage publications.

Annexes

Citrix ADC test IT_06_1_Citrix_ADC – necessary steps

STEP 1

Verify that proper licensing is available and the core features (i) SSL Offloading, (ii) Load Balancing and (iii) Content Switching are enabled

Login to the Citrix ADC CLI and execute the following command

```
> show license
        License status:
                          Web Logging: YES
                       Load Balancing: YES
                    Content Switching: YES
                    Cache Redirection: YES
                    ...
                       SSL Offloading: YES
           ...
                    Content Filtering: YES
                  ....
                              Rewrite: YES
                      Model Number ID: 20
                         License Type: Standard License
                       Licensing mode: Express
 Done
```

STEP 2

Verify that proper networking configuration is available (this topology must be identical to the one existing in the provisioning diagram)

Login to the Citrix ADC CLI and execute the following command

```
> show ns ip
        Ipaddress       Traffic Domain  Type            Mode     Arp      Icmp     Vserver  State
```

```
     ---------      --------------  ----          ----    ---     ----    -------  ------
1)    172.31.118.17   0           NetScaler IP    Active   Enabled  Enabled  NA      Enabled
2)    172.31.108.216   0           SNIP          Active   Enabled  Enabled  NA      Enabled
3)    172.31.98.168   0           VIP          Active   Enabled  Enabled  Enabled  Enabled
 Done
```

STEP 3

Verify that three (3) different vservers are configured in the specific topology. One (1) Content Switching vserver and two (2) Load Balancing vservers

Login to the Citrix ADC CLI and execute the following command

```
> show vserver
1)    scytl-ssl (0.0.0.0:0) - SSL     Type: ADDRESS
      State: UP
      …..
2)    honeypot-ssl (0.0.0.0:0) - SSL  Type: ADDRESS
      State: UP
      …..
1)    cs-ssl (172.31.98.168:443) - SSL       Type: CONTENT
      State: UP
     …..
```

Citrix ADC test IT_06_2_Citrix_ADC – necessary steps

STEP 1

Verify that proper licensing is available and the core features (i) SSL Offloading, (ii) Load Balancing and (iii) Content Switching are enabled

Login to the Citrix ADC CLI and execute the following command

```
> show license
      License status:
                  Web Logging: YES
                Load Balancing: YES
              Content Switching: YES
              Cache Redirection: YES
              ...
```

```
                  SSL Offloading: YES
         ...
                Content Filtering: YES
               ....
                        Rewrite: YES
                 Model Number ID: 20
                   License Type: Standard License
                  Licensing mode: Express
 Done
```

STEP 2

Verify that proper networking configuration is available (this topology must be identical to the one existing in the provisioning diagram)

Login to the Citrix ADC CLI and execute the following command

```
> show ns ip
        Ipaddress        Traffic Domain  Type            Mode     Arp      Icmp     Vserver  State
        ---------        --------------  ----            ----     ---      ----     -------  ------
1)      172.31.118.17    0               NetScaler IP    Active   Enabled  Enabled  NA       Enabled
2)      172.31.108.216   0               SNIP            Active   Enabled  Enabled  NA       Enabled
3)      172.31.98.168    0               VIP             Active   Enabled  Enabled  Enabled  Enabled
 Done
```

STEP 3

Verify that three (3) different vservers are configured in the specific topology. One (1) Content Switching vserver and two (2) Load Balancing vservers

Login to the Citrix ADC CLI and execute the following command

```
> show vserver
1)      scytl-ssl (0.0.0.0:0) - SSL     Type: ADDRESS
        State: UP
        …..
2)      honeypot-ssl (0.0.0.0:0) - SSL  Type: ADDRESS
        State: UP
        …..
```

```
1)     cs-ssl (172.31.98.168:443) - SSL       Type: CONTENT
        State: UP
       …..
```

STEP 4

-------------------

Before initiating any ingress traffic, access all vservers and obtain some baseline metrics

Login to the Login to the Citrix ADC CLI and execute the following commands:

```
> stat lb vserver

Virtual Server(s) Summary
                      vsvrIP  port     Protocol       State   Req/s
scytl-ssl          0.0.0.0     0         SSL          UP      0/s

honeypot-ssl        0.0.0.0     0         SSL          UP      0/s

(A) - Metrics for LB vserver 1
> stat lb vserver scytl-ssl

Virtual Server Summary
                      vsvrIP  port     Protocol       State   Health  actSvcs
scytl-ssl          0.0.0.0     0         SSL          UP      100       1

           inactSvcs
scytl-ssl          0

Virtual Server Statistics
                                          Rate (/s)             Total
Vserver hits                                     0             20895
Requests                                        0             20895
Responses                                        0             20895
Request bytes                                    0            9955458
Response bytes                                    0          694236853
Total Packets rcvd                                0              21379
Total Packets sent                                0             530095
Current client connections                          --                  0
Current Client Est connections                        --                  0
Current server connections                --            0
Current Persistence Sessions               --            0
Current Backup Persistence Sessi            --            0
Requests in surge queue                  --            0
Requests in vserver's surgeQ               --            0
Requests in service's surgeQs              --            0
Spill Over Threshold                    --            0
Spill Over Hits                       --            0
Labeled Connection                      --            0
Push Labeled Connection                  --            0
Deferred Request                      0           0
Invalid Request/Response                  --            0
Invalid Request/Response Dropped               --            0
Vserver Down Backup Hits                   --            1
Current Multipath TCP sessions               --            0
Current Multipath TCP subflows               --            0
Apdex for client response times.             --          1.00
Average client TTLB                     --            0

Bound Service(s) Summary
                IP  port      Type      State    Hits   Hits/s
scytl      172.31.108.137   443       SSL        UP    20895     0/s

           Req    Req/s     Rsp    Rsp/s Throughp ClntConn   SurgeQ
scytl       20895      0/s    20895      0/s      0      0      0

        SvrConn   ReuseP  MaxConn ActvTran  SvrTTFB    Load
scytl           8      0      0      0      0      0
 Done
```

(B) - Metrics for LB vserver 2

Identical as above

(C) - CS vserver

[This command provides the vserver summary only]

```
> stat cs vserver
```

```
Vserver(s) Summary
                    IP  port    Protocol      State   Req/s
cs-ssl      172.31.98.168   443        SSL         UP     0/s
```

[The previous command accompanied by the cs vserevr name provides the actual metrics for the specific cs vserver]

```
> stat cs vserver cs-ssl


Vserver Summary
                    IP  port    Protocol      State
cs-ssl      172.31.98.168   443        SSL         UP

VServer Stats:
                                Rate (/s)            Total
Vserver hits                            0             20890
Requests                               0             20890
Responses                               0             20890
Request bytes                            0             210949
Response bytes                            0             7158230
Total Packets rcvd                        0              40527
Total Packets sent                       0              60118
Current client connections                  --                0
Current Client Est connections                --                0
Current server connections                  --                0
Spill Over Threshold                      --                0
Spill Over Hits                         --                0
Labeled Connection                       --                0
Push Labeled Connection                    --                0
Deferred Request                         0                0
Invalid Request/Response                    --                0
Invalid Request/Response Dropped                --                0
Vserver Down Backup Hits                    --                0
Current Multipath TCP sessions                --                0
Current Multipath TCP subflows                --                0
Apdex for client response times.              --              1.00
Average client TTLB                       --                0
```

STEP 5

Initiate some proper traffic from the interconnected clients, traffic which is in accordance with the policies that are defined in the Citrix ADC

STEP 6

Repeat Step 4 and verify that Content Switching vserver and Scytl-ssl Load Balancing vserver metrics increase as traffic flows and is properly redirected

STEP 7

Initiate some irregular traffic from the interconnected client, traffic which is not in accordance with the policies that are defined in the Citrix ADC

STEP 8

Repeat Step 4 and verify that Content Switching vserver and Honeypot-ssl Load Balancing vserver metrics increase as traffic flows and is properly redirected