Detailed design of the solution

pursuant to MIRRI Decree No. 401/2023 Coll.

Obliged entity	
Project name	
Person responsible for the project	
Project implementer	
Project owner	
Supplier	
Supplier's PM	

Document approval

Item	Name and surname	Organization	Job position	Date	Signature (or electronic approval)
Prepared by					

Document history

Version	Date	Changes	Name
R2v0.3	*30.05.2025*		

Contents

1 Abbreviations and terms used 5

2 Executive summary 6

2.1 Solution intent 6

2.2 Solution objectives 6

2.3 Measurable and performance indicators (KPI) 6

2.4 Solution scope 6

2.5 Outside the project scope 6

2.6 Brief description of the proposed solution 7

2.7 Summary of output M-06 of the proposed solution 7

2.7.1 Overview of end services – future state 7

2.7.2 Scope of information systems – future state 7

2.7.3 Overview of planned integrations of ISVS with supra-ministerial ISVS – common modules pursuant to Act No. 305/2013 on e-Government – future state 7

2.7.4 Overview of planned use of other ISVS (integrations) – future state 7

2.7.5 Application services for the delivery of end services – future state 8

2.7.6 Application services for integration – future state 8

3 Requirements analysis 8

3.1 Requirements analysis procedure 8

3.2 User research 9

3.3 Motivation architecture 9

3.4 Motivation Viewpoints 9

3.5 Catalogue of requirements and its mapping to the design of their solution 9

3.6 Legislation, guidelines and methodological materials taken into account in the solution 9

3.7 Acceptance criteria 10

4 Business architecture 10

4.1 Business Viewpoints 10

4.1.1 ArchiMate 2 View 10

4.1.2 Asset inventory in SOCaaS – business view 10

4.2 Business architecture 10

4.3 Business Viewpoints 10

4.3.1 ArchiMate 2 View 10

4.3.2 Asset inventory in SOCaaS – business view 12

4.4 BPMN - Business process model 14

4.5 Use cases 15

4.5.1 Actors 15

4.5.2 General use cases 17

4.5.2.1 Use case diagrams 17

4.5.2.2 UC_00 - Initialization and configuration of SOCaaS 17

4.5.2.3 UC_01 - Monitor security events 22

4.5.2.4 UC_02 - Visualize and report security data 27

4.5.2.5 UC_03 - Manage detection and correlation rules 32

4.5.2.6 UC_04 – Notify and escalate a security incident 34

4.5.2.7 UC_05 – Resolve and close a security incident 36

4.5.2.8 UC_06 – Evaluate incidents and improve detection and awareness 40

4.5.2.9 UC_07 – Automate incident response 43

4.5.2.10 UC_08 – Proactive threat hunting 45

4.5.2.11 UC_09 – Manage Threat Intelligence and indicators of compromise 47

4.5.2.12 UC_10 – Manage compliance, metrics and management reporting 48
4.5.2.13 UC_11 – Manage assets and the customer environment 53
4.5.2.14 UC_12 – Manage vulnerabilities and security findings 54
4.5.2.15 UC_13 – Design and maintain the security architecture of the customer's SOCaaS environment 56
4.5.2.16 UC_14 – Work with the customer portal and collaborate 59
4.5.2.17 UC_15 – Manage SOCaaS platform operations 65
4.5.2.18 UC_16 – Customize reports, exports and documents to the organization's needs 67
4.5.2.19 UC_17 – Manage users and roles in the SOCaaS platform 68
4.5.2.20 UC_18 – Manage organizations, teams and case structure 71
4.5.2.21 UC_19 – Manage SOCaaS data retention, protection and auditing 73
4.5.2.22 UC_20 – Manage the knowledge base and playbook library 74
4.5.2.23 UC_21 – Manage and evaluate machine learning models for incident detection 79
4.5.2.24 UC_22 – Manage and deploy detection packages and rules 82
4.5.2.25 UC_23 – Manage and operate vulnerability scanning 84
4.5.2.26 UC_24 – Orchestration of integrations between SOCaaS and external systems 86
4.5.2.27 UC_25 – Manage threat intelligence (CTI) and IOC lifecycle 88
4.5.2.28 UC_26 – Configure report scheduling and export 93
4.5.2.29 UC_27 – Manage security posture and asset relationships 95
4.5.2.30 UC_28 – Deploy and maintain SOCaaS in different environments 97
4.5.2.31 UC_29 – Explain and analyse anomalies detected by ML models 99
4.5.2.32 UC_30 – Manage endpoint detection and response (EDR) within SOCaaS 102
4.5.2.33 UC_31 – Provide a workspace for manual threat analysis (CTI workspace) 107
4.5.2.34 UC_32 – Monitor vulnerability scanning and synchronize the status of related incidents . 108
4.6 Business architecture 110
4.7 Business Viewpoints 110
4.7.1 ArchiMate 2 View 110
4.7.2 Asset inventory in SOCaaS – business view 110
4.7.3 Identification of end-service design owners (UX) 110
4.7.4 Specification of end-customer needs 111
4.7.5 User interface prototype design 111
4.7.6 Screen visual design 111
5 Data architecture 111
6 Application architecture 112
6.1 Application architecture 112
6.2 Aplication Viewpoints 112
6.3 Aplication Viewpoints DELETE OLD 115
6.4 Solution design of application services 120
6.4.1 Existing application interfaces 120
6.4.2 Discontinued application interfaces 120
6.4.3 New application interfaces 121
6.5 Integrations between systems 121
6.5.1 List of existing integrations without change 121
6.5.2 List and description of proposed integrations 121
6.5.3 List and description of required integration interfaces 121
6.5.4 Design of the application interfaces provided by the system (module) 121
7 Technological architecture 121
7.1 Technological architecture 121
7.2 Technology Viewpoint INNOSOC Common 121
7.3 Technology Viewpoint Asset Inventory 121

7.4 Technology Viewpoint Kubernetes 123
7.5 Technology Viewpoint OpenStack 136
7.6 Technology Viewpoint OpenStack External 139
7.7 Technology Viewpoint Reporting 139
7.8 Technology Viewpoint Vulnerability Scanner 141
7.9 Technology Viewpoint Ticketing System / Incident Recording and Management 141
7.10 Technology Viewpoint Knowledge Portal for the SOC Team 144
7.11 Technological components of the solution and their relationship to application components . 147
7.11.1 Management of access, accounts and keys of operating environments 147
7.11.2 Containerization and orchestration of cloud services 147
7.11.3 Specification of the use of government cloud services 147
7.12 Description of the test environment 147
7.13 Description of the production environment 147
8 Service level (SLA) and performance requirements 147
9 Software licences and source code 147
10 Ensuring availability, backup and recovery of the solution 147
10.1 Ensuring availability 148
10.2 Backup 148
10.3 Recovery of the solution 148
11 Security - addressing security requirements 148
12 Data migration 151
12.1 Data migration procedure 151
12.2 Data migration verification 151
13 Implementation and deployment schedule, dependencies 152
13.1 Project phases and timeline - consolidated overview 152
13.2 Dependencies and solution design 152
13.3 Requirements for the Testing and Production Deployment phases 152
14 Open issues 152
15 Annexes 152

1 Abbreviations and terms used

ABBREVIATION	DESCRIPTION
API	Application Programming Interface
API GW	Process integration and data integration module - access part
CEP	ÚPVS central electronic registry module
CI/CD	Continuous integration/Continuous deployment – a series of steps for correct software deployment - Continuous integration and delivery
CUET	Central Official Electronic Notice Board
e-Government	Electronical Government, digitalization of public administration
eDesk	ÚPVS electronic communication mailbox module
eForm	ÚPVS electronic forms module
G2G	Process integration and data integration module - communication part
GDPR	General Data Protection Regulation
GovNET	Secure government network administered by the National Agency for Network and Electronic Systems. / Cross-sectoral public administration information network
GUI	Graphical user interface
IAM	Identity Access Management - User identity management. / Authentication module within ÚPVS
IDSK	A set of rules for creating the content and functionality of electronic services in line with users' needs.
KAV	Consolidated analytical layer
MIRRI	Ministry of Investments, Regional Development and Informatization
MoSCoW	M - Must have, S - Should have, C - Could have, W - Won't have. A technique for prioritizing requirements in project management and software development
NASES	National Agency for Network and Electronic Services
OVM	Public authority - a state body, a territorial self-government body, a public-law institution, a body of professional self-government, and a natural person or legal person entrusted by law with the exercise of public authority
PO/FO	Legal person/Natural person
SkTalk/Sk-Talk	Standard for communication via ÚPVS pursuant to Section 11(g) of Decree No. 78/2020 Coll.
SLA	Service Layer/Level Agreement - Agreement on the level/ parameters / quality of service provision
SvM	Slovensko v mobile (Slovakia in the mobile)
PROD	Production environment
UAT	Test environment
UIR	Universal integration interface – an asynchronous interface for receiving and sending messages from and to ÚPVS
USR	Universal synchronous interface for making other synchronous services of ÚPVS modules accessible – from the service provider's perspective, it is an external bus interface
ÚPVS	Central Public Administration Portal, central portal
XML	Extensible Markup Language
WSDL	Web Services Description Language
ZEP/KEP	Guaranteed electronic signature / Qualified electronic signature
ŽS	Life situation

TERM/REFERENCE	DESCRIPTION
ArchiMate	A standardized modelling language used primarily for the representation, description and subsequent analysis of enterprise architecture.
Authentication	The process of identifying and verifying the identity of a user requesting a service or resource in an information system.
Authorization	Represents verification of the authorization of persons or objects to work with data.
Documentation for e-Notify	Documentation for the ÚPVS e-Notify module for integration partners is available on the Partner Framework Portal (PFP); access to the PFP portal is granted on request.
Integration manual for e-Notify	Integration manual for e-Notify with a description of services in the current version on the PFP portal eNOTIFY - All documents (gov.sk) (gov.sk)
KAV	KAV - Consolidated analytical layer | Ministry of Investments, Regional Development and Informatization of the Slovak Republic (gov.sk)
MetaIS/METAIS	Central metainformation system of public administration
Notification	A notification is an information signal or message sent to a user in order to alert them to an event, an activity or a need for intervention.
eGOV,e-Government	Electronical Government, digitalization of public administration
OpenID Connect (OIDC), OAuth2	RFC 6749 (The OAuth 2.0 Authorization Framework), a standard for authentication and determining the scope of permissions when using the REST protocol pursuant to the Decree on Standards 78/2020, Section 11(i)
Rules for publishing electronic services	Rules for publishing electronic services in the multichannel environment of public administration https://mirri.gov.sk/wp-content/uploads/2018/10/Pravidla_Publikovania_Sluzieb_v1_0-1.pdf
Representational State Transfer (REST)	Network communication protocol used in the provision of electronic services required for processing electronic submissions or for the successful completion and preparation of an electronic submission via the process integration and data integration module pursuant to Decree on Standards 78/2020, §11(a)
Backward compatibility	In technical practice, a property of a product consisting in the product being able to replace its predecessor in such a way that it remains capable of interoperating with products that were created for this older product. The newer product is thus able to fully replace its predecessor.
ŽS	Priority life situations | Ministry of Investments, Regional Development and Informatization of the Slovak Republic (gov.sk)

2 Management summary

2.1 Solution intent

2.2 Objectives of the solution

2.3 Measurable and performance indicators (KPI)

2.4 Solution scope

2.5 Outside the project scope

2.6 Brief description of the proposed solution

2.7 Summary of output M-06 of the proposed solution

2.7.1 Overview of end services – future state

KS code (from MetaIS)	KS name	KS user (G2C/G2B/G2G/G2A)	Life situation (+ code from MetaIS)	KS level of electronization

Table 1 Overview of end services – future state

2.7.2 Scope of information systems – future state

ISVS code (from MetaIS)	ISVS name	ISVS module (tick if the ISVS is a module)	IS VS status	IS VS type	Code of the parent ISVS (if the checkbox for ISVS module is ticked)

Table 2 Scope of information systems – future state

2.7.3 Overview of planned integrations of the ISVS with cross-government ISVS – common modules pursuant to Act No. 305/2013 on e-Government – future state

IS code	ISVS name	Common modules pursuant to Act No. 305/2013 on e-Government

Table 3 Overview of planned integrations of the ISVS with cross-government ISVS

2.7.4 Overview of planned use of other ISVS (integrations) – future state

ISVS code *(from MetaIS)*	ISVS name	Code of the integrated ISVS *(from MetaIS)*	Name of the integrated ISVS

Table 4 Overview of planned use of other ISVS (integrations)

2.7.5 Application services for the implementation of end services – future state

AS code *(from MetaIS)*	AS name	ISVS/ISVS module *(code from MetaIS)*	Application service implements KS *(KS code from MetaIS)*

Table 5 Application services for the implementation of end services

2.7.6 Application services for integration – future state

AS (MetaIS code)	AS name	Implemented by ISVS (MetaIS code)	Providing or Consuming	Integration via CAMP	Integration with third-party IS	SaaS	Integration with the provider's AS (MetaIS code)

Table 6 Application services for integration

3 Requirements analysis

3.1 Requirements analysis procedure

In developing the architecture, this project uses the TOGAF framework and the ArchiMate language. The use of TOGAF framework components has been adapted for the purposes of this project. Primarily, the part of the Architecture Development Method concerning the development of the architecture in all four architecture domains was adopted from the framework. The Architecture Development Method is a process with recommended steps and activities. Although ADM is often associated with iterative and agile approaches, in view of the objectives and nature of this project it has been adapted to the Waterfall methodology, which is sequential and linear.

The project is developed in individual consecutive phases.

The first phase involves the analysis of requirements that define the expectations of users and stakeholders.

During the second phase, the requirements are transformed into a system architecture design that defines how the system will work. The design takes place at two levels. The high-level design determines the overall structure of the system, such as the main components and the way they are interconnected. The detailed specification describes each module or component in more detail, including data models, algorithms, interfaces and logical flows. This design is submitted for approval to ensure that it meets the requirements and respects technological, financial and legislative constraints.

After it has been reviewed and approved, the implementation phase begins. During this phase, the individual modules of the system are implemented according to the specification, and quality and security standards are applied when writing code. Continuous testing of each module will be ensured by UNIT testing. The implementation phase will take place in two steps. In the first step, a version of the application containing the functionality that covers the specified functional requirements will be deployed, and the first phase of

implementation will be checked and approved. In the second step, the remaining functionality will be implemented and integrated with the existing modules.

The next phase ensures that the system is tested. It is verified that all components work together correctly and that the system as a whole meets the requirements defined in the initial phase.

After successful testing, the system is deployed to the production environment. This process includes software installation and environment configuration. This phase also includes user training so that users are able to use the system effectively.

After deployment, the system is monitored and continuously updated according to user needs and technological changes. Maintenance includes bug fixing, implementation of new features and performance optimization. The maintenance process must respect all legislative requirements, for example ensuring personal data protection and cybersecurity.

The Enterprise Architect tool is used to design the application components being developed.

3.2 User research

3.3 Motivation architecture

3.3.1

3.4 Motivation Viewpoints

Figure 1 : Motivation Viewpoint

3.5 Requirements catalogue and its mapping to the design of their solution

The requirements catalogue is defined as an annex to this document. The design of the solution to the requirements is the content of this document and is described in detail in the following chapters. Based on the requirements catalogue, a Test Plan document has been created, which will serve as the basis for acceptance during UAT testing. The scope and plan of the tests are defined in the Test Strategy document and include:

- Functional testing (FAT) – on the supplier's side
- System and integration testing (supplier; NASES may participate)
- Load and performance testing
- Security testing
- User tests of the functional user interface (UX) and accessibility testing (may not be mandatory)
- User acceptance testing (UAT) – on the side of the contracting authority / recipient

The security tests will include penetration tests, from which a formal report will be produced.

Summary of the mapping of requirements to use cases and architecture objects:

3.6 Legislation, guidelines and methodological materials taken into account in the solution

3.7 Acceptance criteria

4 Business architecture

4.1 Business Viewpoints

4.1.1 ArchiMate 2 View

4.1.2 Asset inventory in SOCaaS – business view

4.2 Business architecture

4.3 Business Viewpoints

4.3.1 ArchiMate 2 View

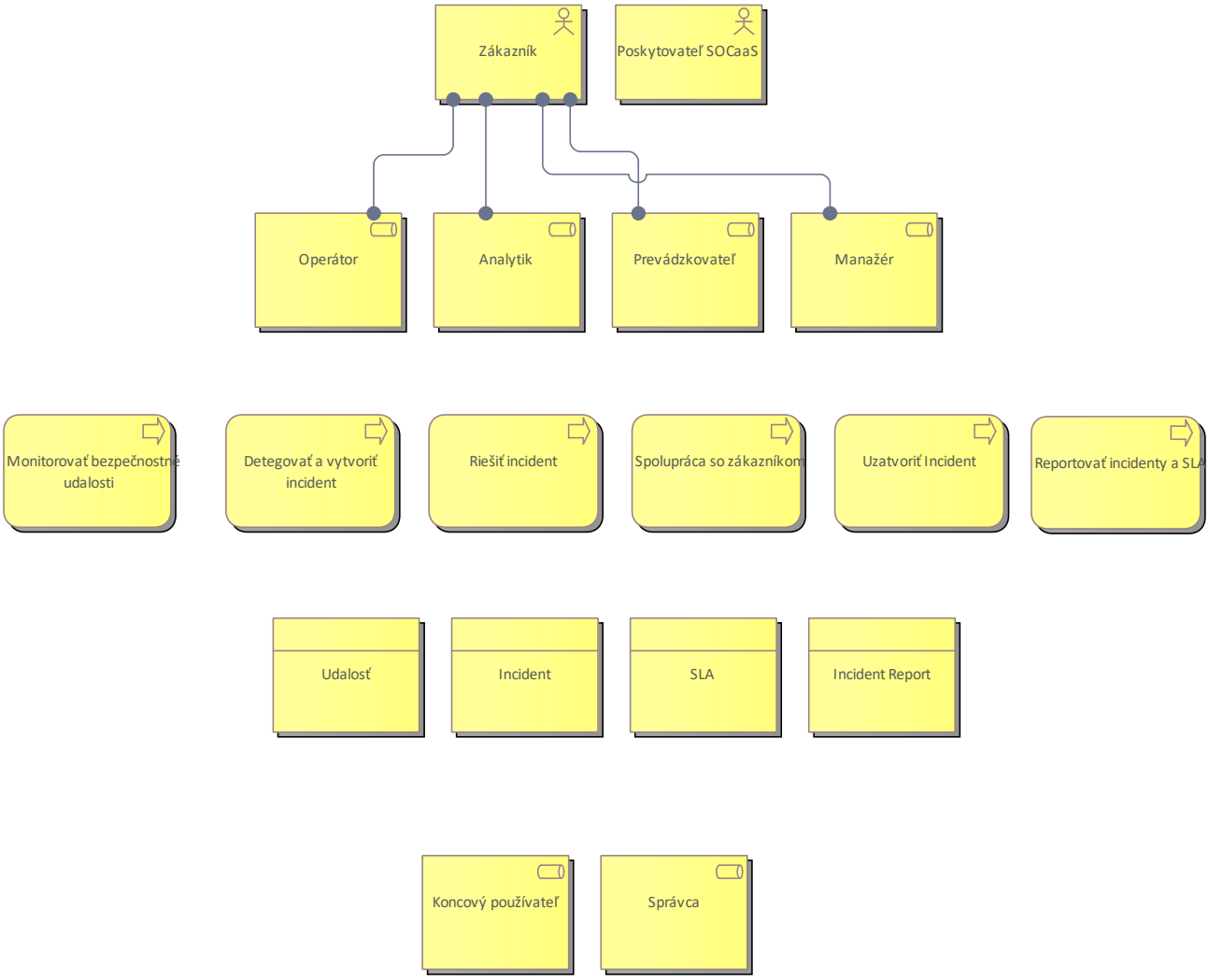


Figure 2 : ArchiMate 2 Business Layer Diagram

Description of the elements used:

Detect and create an incident - BusinessProcess

Monitor security events - BusinessProcess

Report incidents and SLA - BusinessProcess

Resolve incident - BusinessProcess

Cooperation with the customer - BusinessProcess

Close an Incident - BusinessProcess

Analyst - BusinessRole

Incident - BusinessObject

Incident Report - BusinessObject

End user - BusinessRole

Manager - BusinessRole

Operator - BusinessRole

SOCaaS provider - BusinessActor

Operator - BusinessRole

SLA - BusinessObject

Administrator - BusinessRole

Event - BusinessObject

Customer - BusinessActor

4.3.2 Asset inventory in SOCaaS – business view

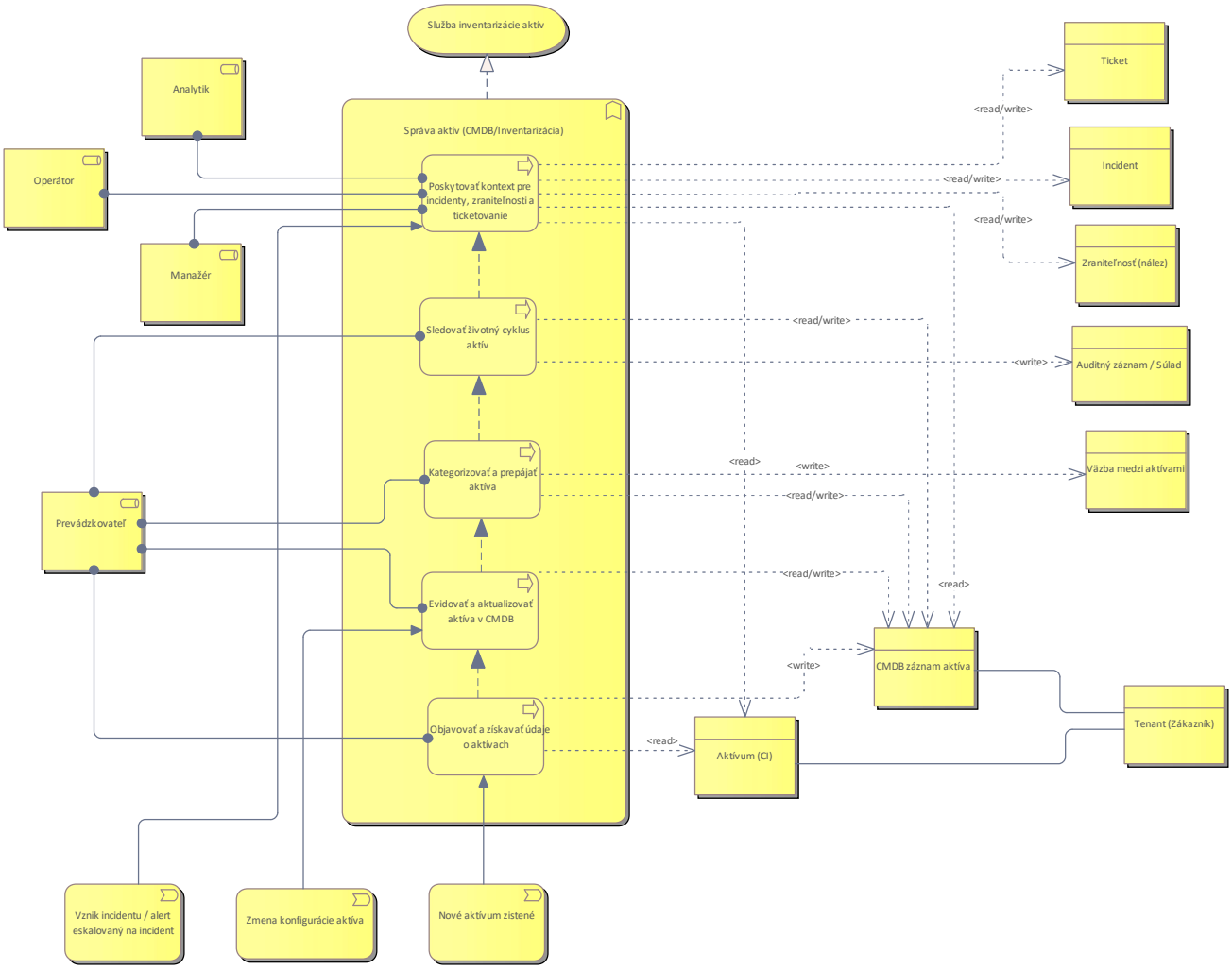


Figure 3 : Asset inventory in SOCaaS – business view

Description of the elements used:

New asset detected - BusinessEvent

Asset inventory service - BusinessService
Centralized recording and management of assets across tenants. It supports categorization, relationships, and tracking of security status and life cycle. It provides context for vulnerabilities, incidents, ticketing, audit and regulations.

Asset management (CMDB/Inventory) - BusinessFunction

Incident occurrence / alert escalated to an incident - BusinessEvent

Asset configuration change - BusinessEvent

Asset (CI)
- BusinessObject
Device/system/application/virtual or cloud resource.

Analyst - BusinessRole
Uses asset records to contextualize incidents and to analyse impact and priorities.

Audit record / Compliance - BusinessObject
Evidence of the status and changes of assets for audit and regulations.

CMDB asset record
- BusinessObject
Asset attributes, tenant, criticality, status, change history.

Incident
- BusinessObject
A security incident contextualized by asset records.

Manager - BusinessRole
Uses asset records for audit, recovery planning, compliance and reporting.

Operator - BusinessRole
Works with tickets and SLAs; uses asset information for operational handling and escalations.

Operator - BusinessRole
Is responsible for the quality and timeliness of asset records, categorization, life cycle management and data collection.

Tenant (Customer) - BusinessObject

Ticket
- BusinessObject
A work item for resolution; linked to an asset/incident/SLA.

Relationship between assets
- BusinessObject
Dependencies and interconnections (topology, service-to-component, network relationships).

Vulnerability (finding)
- BusinessObject
A finding linked to an asset; input for priority and impact.

Record and update assets in the CMDB - BusinessProcess
Maintaining asset records (identity, type, owner/tenant, criticality, status).

Categorize and link assets - BusinessProcess
Tagging and creating relationships between assets (dependencies, topology, relations).

Discover and obtain asset data - BusinessProcess
Obtaining data on devices, systems, applications and cloud resources; the aim is completeness and timeliness.

Provide context for incidents, vulnerabilities and ticketing - BusinessProcess
Contextualization of incidents, analysis of impact and priorities, support for efficient ticketing.

Track the asset life cycle - BusinessProcess
Status transitions, change history, recovery planning, auditability.

4.4 BPMN - Business process model

4.5 Use cases

The following chapter presents the individual user scenarios.

4.5.1 Actors

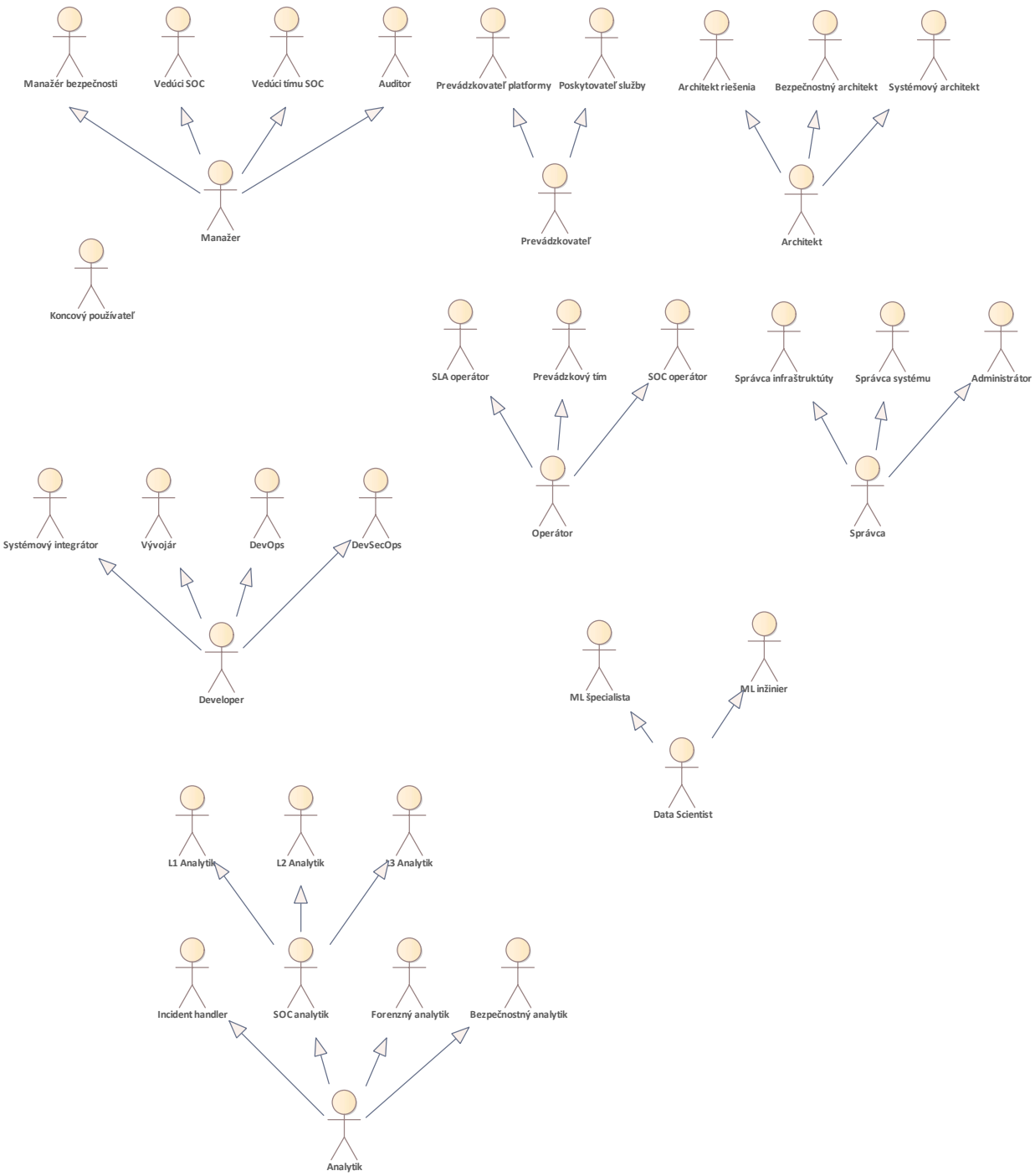


Figure 4 : Actors

Name of the actor/role	Description
Administrator	Responsible for: SIEM, log processing, data visualization, alerting, EDR/XDR, asset management, vulnerability scanner and security data sources.
Analyst	Responsible for areas such as: Log processing, SOAR, ML-based detection, Data visualization, Alerting, EDR/XDR, Asset management, Vulnerability scanner,

Name of the actor/role	Description
	knowledge base, Incident recording and management, and Reporting.
Architect	Responsible for areas such as: ML-based detection and EDR/XDR.
Solution architect	Responsible for areas such as: SIEM and Log processing.
Auditor	Responsible for areas such as: SIEM, SOAR, Asset management, Incident recording and management, Reporting, Optional extensions, System administration, CTI and Alerting.
Security analyst	Responsible for areas such as: SIEM, Log processing, SOAR, ML-based detection, Data visualization, Alerting, EDR/XDR, CTI, Asset management, Vulnerability scanner, Security data sources, Knowledge base, Ticketing, and Incident recording and management.
Security architect	Responsible for areas such as: SIEM, SOAR, ML-based detection, Data visualization, EDR/XDR, Secure data sources, Knowledge base, Incident recording and management, and System administration.
Data Scientist	Responsible for areas such as: ML-based detection.
Developer	Developer is a basic role focused on the design, development, modification and maintenance of software, scripts and integration solutions. It focuses on implementing logic, automating processes and providing technical support for development.
DevOps	Responsible for areas such as: SIEM, Log processing, ML-based detection, Vulnerability scanner, Security data sources, System administration and Optional extensions.
DevSecOps	Responsible for areas such as: SOAR, CTI, Vulnerability scanner and system administration.
Forensic analyst	Responsible for areas such as: optional extensions and EDR/XDR.
Incident handler	Responsible for areas such as: SOAR and Incident recording and management.
Incident Handler	
End user	Responsible for areas such as: Data visualization.
L1 Analyst	
L2 Analyst	
L3 Analyst	
Manager	Responsible for areas such as: SIEM, ML-based detection, data visualization, alerting, Asset management, vulnerability scanner, knowledge base, Ticketing, Reporting and Optional extensions.
Security manager	Responsible for areas such as: SIEM, SOAR, ML-based detection, Data visualization, Alerting, EDR/XDR, CTI, Asset management, Vulnerability scanner, Ticketing, Incident recording and management, Reporting, System administration and Optional extensions.
ML engineer	Responsible for areas such as: ML-based detection.
ML specialist	Responsible for areas such as: ML-based detection and EDR/XDR.
Operator	Responsible for areas such as: alerting, EDR/XDR, CTI, asset management, vulnerability scanner, security data sources, knowledge base, ticketing and system administration.
Service provider	Responsible for areas such as: SIEM.
Service operator	Responsible for areas such as: data visualization.
Platform operator	Responsible for areas such as: SIEM.
Operations team	Responsible for areas such as: alerting and optional extensions.
SLA operator	Responsible for areas such as: alerting.
SOC analyst	Responsible for areas such as: EDR/XDR, CTI, incident recording and management, reporting and optional

Name of the actor/role	Description
	extensions.
SOC operator	Responsible for areas such as: alerting, EDR/XDR, knowledge base and ticketing.
Administrator	The administrator is a basic technical role responsible for the operation, maintenance and functioning of IT infrastructure and systems. It covers competencies ranging from infrastructure, through systems management, to configuration administration.
Infrastructure administrator	Responsible for areas such as: Log processing, Asset management and System administration.
System administrator	Responsible for areas such as: SOAR, Data visualization, alerting, EDR/XDR, Asset management, Vulnerability scanner, Security data sources, Knowledge base, Ticketing, Incident recording and management, Reporting and System administrationdata visualization.
System architect	Responsible for areas such as: SIEM, Log processing, SOAR, ML-based detection, Vulnerability scanner and System administration.
System integrator	Responsible for areas such as: SIEM, Log processing, SOAR, ML-based detection, Data visualization, Alerting, EDR/XDR, CTI, Asset management, Vulnerability scanner, Security data sources, Knowledge base, Ticketing, and Incident recording and management.
SOC lead	Responsible for areas such as: alerting.
SOC team lead	Responsible for areas such as: data visualization.
Developer	Responsible for areas such as: SIEM, SOAR, ML-based detection and Security data sources.

4.5.2 General use cases

4.5.2.1 Use case diagrams

4.5.2.2 UC_00 - Initialization and configuration of SOCaaS

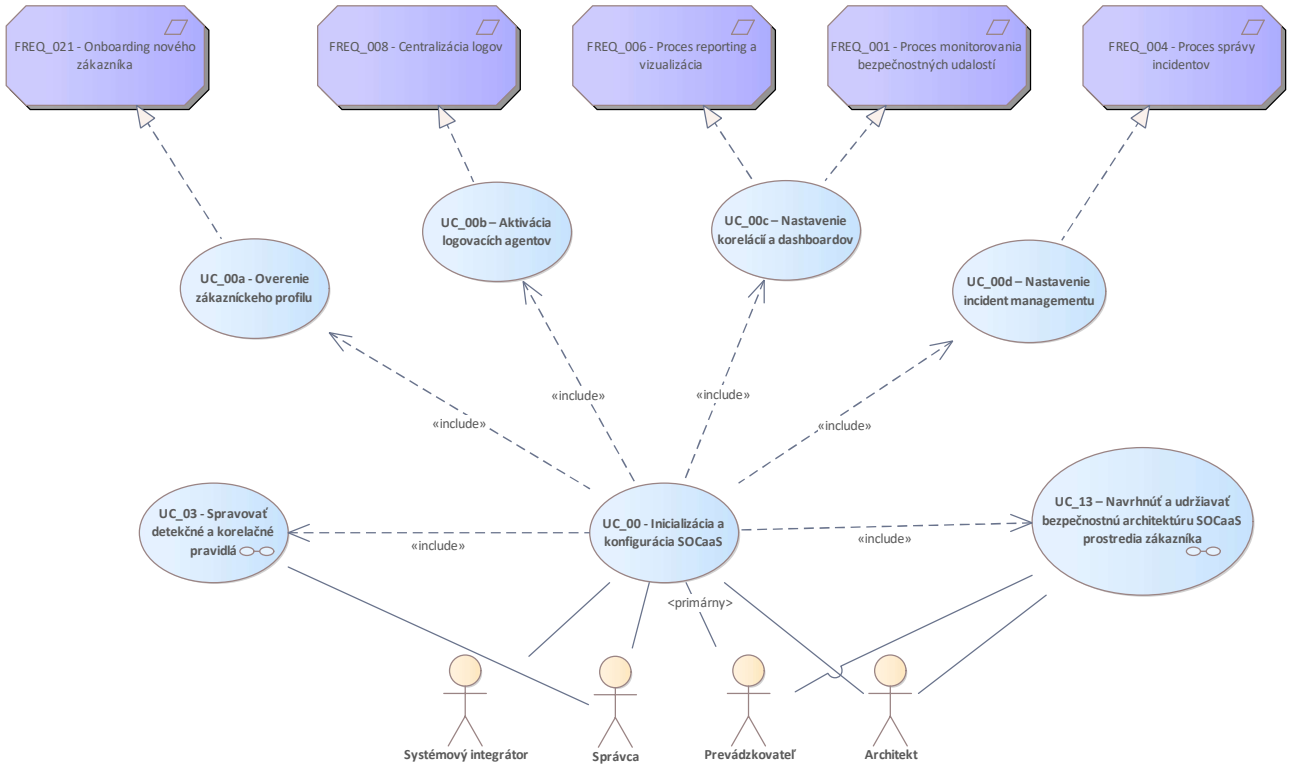


Figure 5 : UC_00 - Initialization and configuration of SOCaaS

Use case	UC_00 - Initialization and configuration of SOCaaS
Objective	The use case describes the initial initialization and set-up of SOCaaS for a new customer. It includes the activation of logging sources and the set-up of integration points, dashboards and basic security processes.

Precondition	The customer has a tenant created.
Precondition	The architect has configuration permissions.
Precondition	Basic network connectivity is available.
Postcondition	SOCaaS is successfully initialized and ready for monitoring and detection.
Postcondition	The basic modules are activated: logging, SIEM correlations, dashboards, incident management.
Postcondition	Logs from all mandatory sources contain the mandatory fields for detection.

Basic Path	Basic Path

Step	**Activity**
1	The architect starts the initialization process for a new customer.
2	The system loads the customer profile and prepares the configuration interface.
3	The architect selects the modules to be activated (SIEM, SOAR, CTI, Ticketing).
4	The system performs the initialization tasks (registration of sources, activation of agents, creation of

	dashboards).
5	The architect confirms the configuration.
6	The system saves the configuration and reports successful completion.

ALT_00_01 – Incomplete customer profile	Alternate

Step	**Activity**
1	The system detects missing data.
2	It notifies the architect to supply the missing data.
3	The process continues once the information has been supplied.

ALT_00_02 – Unavailable logging sources / infra	Alternate

Step	**Activity**
1	The system is unable to establish a connection with the logging agent or the customer's API.
2	It displays an error and offers the option to retry or skip the module.
3	The process continues with limited functionality.

Linked functional requirements	
UC_03 - Manage detection and correlation rules	This use case covers the activities of analysts and administrators in creating, managing and maintaining detection and correlation rules in the SOCaaS environment. These rules enable the detection of security threats based on IOCs, behavioural anomalies, event chaining or threshold values. The use case focuses exclusively on the rule management process – their definition, tuning, classification and mapping to security frameworks (e.g. MITRE ATT&CK).
UC_00b – Activation of logging agents	The system activates the customer's logging agents to send logs to SOCaaS.
UC_13 – Design and maintain the security architecture of the customer's SOCaaS environment	UC_13 addresses the design and ongoing maintenance of the technical and security architecture for a specific customer: how the customer is connected to SOCaaS, which systems, zones and logging sources are integrated, through which channels the data flow (agent, syslog, API, VPN, peering…), how SIEM, SOAR, TI, ticketing, EDR, vuln scanners and asset mgmt are interconnected. The output is an approved and maintained architectural design (topology, integration points, data flows) and a list of mandatory log sources and log types, which serves as the basis for onboarding (UC_00), monitoring (UC_01), asset management (UC_11), vulnerability management (UC_12) and automation (UC_07).
UC_00c – Setting up correlations and dashboards	The system creates predefined correlation rules and dashboards for monitoring security events.
UC_00a - Verification of the customer profile	The system verifies whether the customer tenant contains all mandatory data (identification, technical, integration).
UC_00d – Setting up incident management	The system activates the workflow for incident management and role assignment.

Use case	UC_00a - Verification of the customer profile
Objective	The system verifies whether the customer tenant contains all mandatory data (identification, technical, integration).

Precondition	The tenant exists in the system.
Postcondition	The customer profile is confirmed as complete.

Basic Path	Basic Path

Step	**Activity**
1	The platform operator starts the profile verification.
2	The system loads the customer data.
3	The system checks the mandatory fields.
4	The verification is successful.

ALT_00a_01 – Incomplete data	Alternate

Step	**Activity**
1	The system requests that the missing data be supplied

ALT_00a_02 – The tenant lacks technical configuration	Alternate

Step	**Activity**
1	The system stops the process and notifies the administrator.

Linked functional requirements	
FREQ_021 - Onboarding of a new customer	The solution must ensure the onboarding of a new customer, during which their infrastructure is integrated into the multi-tenant SOCaaS environment. This includes: • configuration of the logging agent • configuration of correlation rules • creation of separate dashboards • configuration of incident management for the customer
UC_00a - Verification of the customer profile	The system verifies whether the customer tenant contains all mandatory data (identification, technical, integration).

Use case	UC_00b – Activation of logging agents
Objective	The system activates the customer's logging agents to send logs to SOCaaS.

Precondition	The customer systems are available.
Precondition	The communication channels are open.
Postcondition	The logging agents are active and sending data.

Basic Path	Basic Path

Step	**Activity**
1	The platform operator starts the activation.
2	The system contacts the logging agents.
3	The availability of the API and certificates is verified.

4	The agents are activated.
ALT_00b_01 – The agent does not respond	Alternate
Step	**Activity**
1	The system reports an error and continues later.
ALT_00b_02 – Invalid API keys	Alternate
Step	**Activity**
1	The system requests that the keys be uploaded again.
Linked functional requirements	
FREQ_008 - Log centralization	The solution must ensure the collection and centralization of logs from systems, networks, endpoints and applications.
UC_00b – Activation of logging agents	The system activates the customer's logging agents to send logs to SOCaaS.
Use case	UC_00c – Setting up correlations and dashboards
Objective	The system creates predefined correlation rules and dashboards for monitoring security events.

Precondition	The SIEM module is active.
Precondition	Logs are arriving in the system.
Postcondition	The correlation rules and dashboards are configured.
Basic Path	Basic Path
Step	**Activity**
1	The platform operator starts the set-up.
2	The system loads the correlation template.
3	The system creates dashboards for monitoring.
4	The configuration is saved.
ALT_00c_02 – The dashboard cannot be created	Alternate
Step	**Activity**
1	The system creates only a minimal version. (the last executable one)
ALT_00c_01 – The correlation rule is invalid	Alternate
Step	**Activity**
1	The system skips it and reports an error.
Linked functional requirements	
FREQ_006 - Reporting and visualization process	Ensure the reporting and visualization process – ensures regular output for management, auditors and SOC analysts.
FREQ_001 - Security event monitoring process	Ensure the security event monitoring process – ensures the collection and tracking of logs, metric data and alerts from all integrated systems.

UC_00c – Setting up correlations and dashboards	The system creates predefined correlation rules and dashboards for monitoring security events.
Use case	UC_00d – Setting up incident management
Objective	The system activates the workflow for incident management and role assignment.

Precondition	The Ticketing or SOAR module is initialized.	
Postcondition	The incident lifecycle (triage → analysis → closure) is ready.	
Basic Path		Basic Path
Step	**Activity**	
1	The platform operator starts the configuration.	
2	The system loads the workflow templates.	
3	The system creates incident categories and priorities.	
4	It assigns roles (Analyst, Operator, Manager).	
ALT_00d_01 – No user roles exist		Alternate
Step	**Activity**	
1	The system stops the workflow and asks the administrator to add roles.	
ALT_00d_02 – The Ticketing module is unavailable		Alternate
Step	**Activity**	
1	The system activates only the basic SOAR fallback.	
Linked functional requirements		
FREQ_004 - Incident management process	Ensure the incident management process – includes recording, classifying, prioritizing, documenting and closing incidents.	
UC_00d – Setting up incident management	The system activates the workflow for incident management and role assignment.	

4.5.2.3 UC_01 - Monitor security events

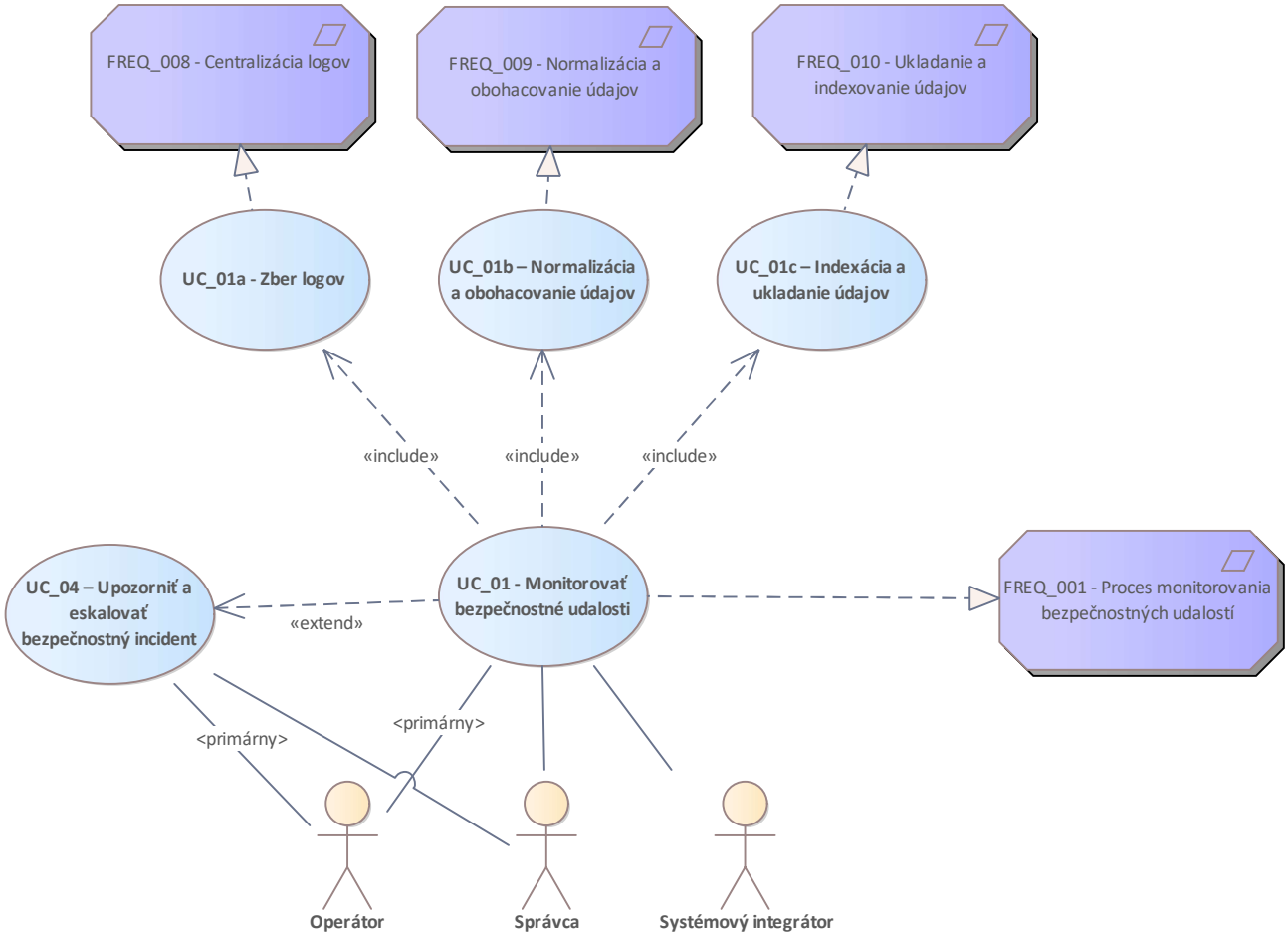


Figure 6 : UC_01 - Monitor security events

Use case	UC_01 - Monitor security events
Objective	The use case describes the continuous monitoring of security events within SOCaaS. It includes the collection of events, their normalization, enrichment and the preparation of alerts for the subsequent incident detection process.

Precondition	The logging agents are active.
Precondition	The SOCaaS SIEM module is functional.
Precondition	UC_00 (initialization) has been successfully completed.
Postcondition	The events are processed and passed on to the incident detection process.
Postcondition	The alerts are displayed to the operator.
Postcondition	Monitoring runs in real time.
Basic Path	Basic Path

Step	Activity
1	The operator monitors the control dashboard in SOCaaS.
2	UC_01a: The system receives events from logging sources.
3	UC_01b: The system normalizes and enriches the data.
4	UC_01c: The system indexes the stored data for fast searching.
5	SOCaaS generates alerts based on correlation rules.
6	The operator sees the resulting security events.
7	The use case ends; monitoring continues without restriction. The operator evaluates the alert.
8	If it is critical → activates UC_04 - Notify and escalate a security incident.
9	The Use Case ends – monitoring continues.

ALT_01_01 – Unavailable logs or agents	Exception

Step	Activity
1	SOCaaS is not receiving data from the source.
2	The system creates a “Missing logs” alert.
3	The administrator is notified and can resolve the problem manually.

ALT_01_02 – Overloaded SIEM or pipeline	Exception

Step	Activity
1	SOCaaS cannot keep up with processing the stream of events.
2	The system enables protective mechanisms (rate-limiting, buffering).
3	The operator is notified of the risk of event loss.

Linked functional requirements	
UC_01c – Data indexing and storage	The use case handles the storage of normalized security events in storage and their indexing for fast searching and correlation mechanisms.
UC_04 – Notify and escalate a security incident	After the system detects and creates a security incident (UC_01, UC_03), the objective of UC_04 is to ensure that the right person (1st level) learns about it in time, that someone takes over the incident and that, in the event of inactivity or increasing severity, the incident is automatically escalated to a higher level
UC_01b – Data normalization and enrichment	The use case ensures the normalization of logs into a uniform format and their enrichment with information from CTI, the asset database or user data.
UC_01a - Log collection	It ensures the technical reception of logging data from systems, applications, devices and sources into the SIEM platform. It is the foundation of the monitoring process.
FREQ_001 - Security event monitoring process	Ensure the security event monitoring process – ensures the collection and tracking of logs, metric data and alerts from all integrated systems.

Use case	UC_01a - Log collection
Objective	It ensures the technical reception of logging data from systems, applications, devices and sources into the SIEM platform. It is the foundation of the monitoring process.

Precondition	The communication channel between the system and the agent is functional.

Precondition	The logging agent is active.
Postcondition	The logs are ready for the next step – normalization.
Postcondition	SOCaaS has received new logs.

Basic Path	Basic Path

Step	**Activity**
1	The logging agent sends security events.
2	SOCaaS receives the data.
3	The format and integrity of the logs are verified.
4	The logs are passed on to the normalization process.

ALT_01a_01 – Logs are not being delivered	Alternate

Step	**Activity**
1	SOCaaS does not record any data from the source.
2	The system creates a “Missing logs” notification.
3	The administrator receives a notification.

ALT_01a_02 – Invalid log format	Alternate

Step	**Activity**
1	The log has an incorrect format or missing fields.
2	SOCaaS rejects it and stores it in the quarantine buffer.
3	A notification is sent to the integrator.

Linked functional requirements	
FREQ_008 - Log centralization	The solution must ensure the collection and centralization of logs from systems, networks, endpoints and applications.
UC_01a - Log collection	It ensures the technical reception of logging data from systems, applications, devices and sources into the SIEM platform. It is the foundation of the monitoring process.
Use case	UC_01b – Data normalization and enrichment
Objective	The use case ensures the normalization of logs into a uniform format and their enrichment with information from CTI, the asset database or user data.

Precondition	A configuration of normalization rules exists.
Precondition	UC_01a (log collection) was successful.
Postcondition	The log is normalized and enriched.
Postcondition	The data are ready for storage and indexing.

Basic Path	Basic Path

Step	**Activity**

1	SOCaaS loads the received logs.
2	A normalization profile is applied according to the source type.
3	Data enrichment (IP reputation, hostname, user).
4	The resulting event is ready for indexing.

ALT_01a_01 – Unknown log type	Alternate

Step	**Activity**
1	The system uses a fallback normalization profile.

ALT_01a_02 – CTI feed is missing	Alternate

Step	**Activity**
1	The system performs normalization without enrichment and marks the event as “unenriched”.

Linked functional requirements	
FREQ_009 - Data normalization and enrichment	The solution must ensure data normalization and enrichment.
UC_01b – Data normalization and enrichment	The use case ensures the normalization of logs into a uniform format and their enrichment with information from CTI, the asset database or user data.

Use case	UC_01c – Data indexing and storage
Objective	The use case handles the storage of normalized security events in storage and their indexing for fast searching and correlation mechanisms.

Precondition	Data normalization (UC_01b) has been completed.
Precondition	The storage and the cluster are available.
Postcondition	The events are stored in the SIEM database.
Postcondition	The indexes are ready for query systems.

Basic Path	Basic Path

Step	**Activity**
1	The system takes over the normalized data.
2	It writes them to storage (DB or distrib. index).
3	It creates/updates the indexes.
4	The event is available for correlation rules.

ALT_01c_01 – Storage unavailable	Alternate

Step	**Activity**
1	The system stores the data in a buffer and creates an alert.

ALT_01c_02 – Storage space exhausted	Alternate

Step	**Activity**

1	The system activates a clean-up or alerts operations.
Linked functional requirements	
FREQ_010 - Data storage and indexing	The solution must ensure the storage, indexing and retention of data.
UC_01c – Data indexing and storage	The use case handles the storage of normalized security events in storage and their indexing for fast searching and correlation mechanisms.

4.5.2.4 UC_02 - Visualize and report security data

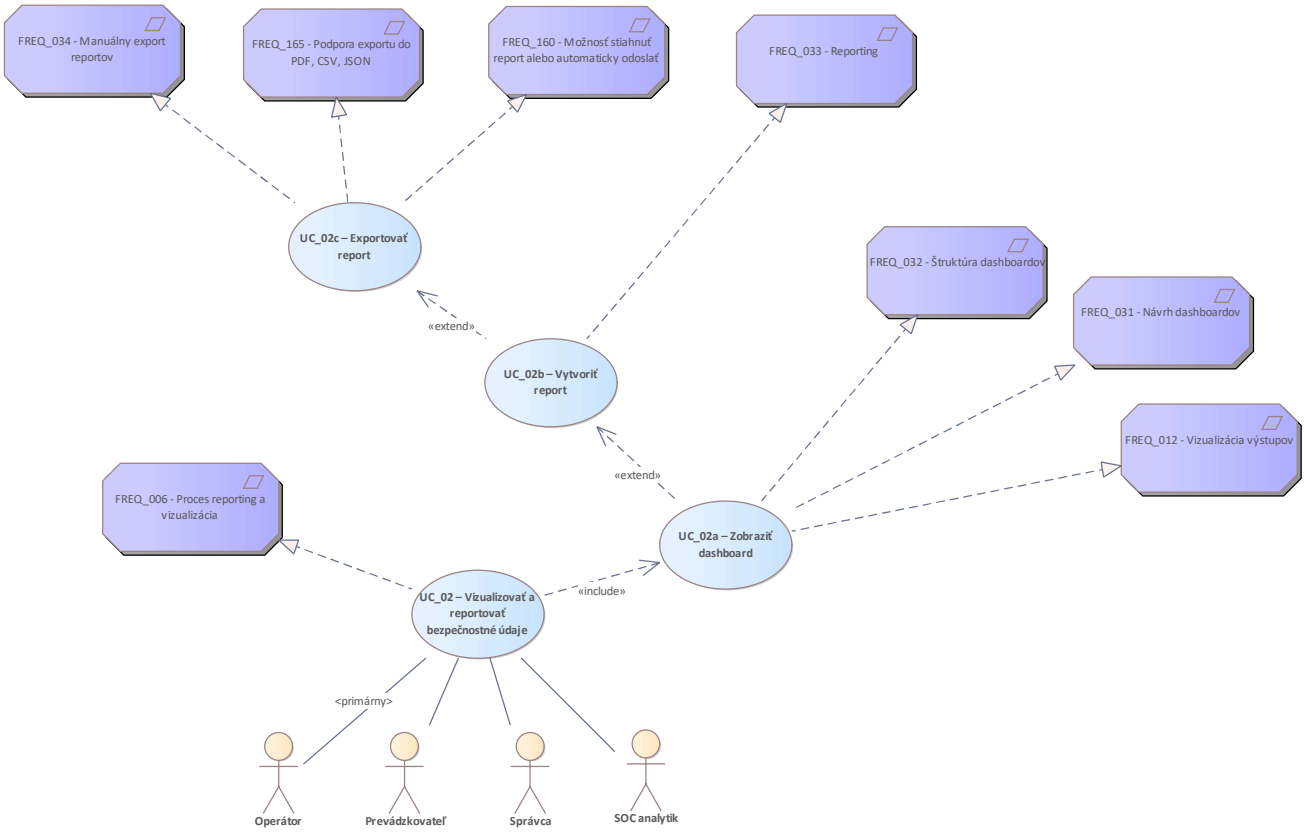


Figure 7 : UC_02 - Visualize and report security data

Use case	UC_02 – Visualize and report security data
Objective	The system enables SOC operators and the administrator to display security events using dashboards, filters, charts and tabular outputs. It also enables the generation of reports and the export of data for management, audit or customers.

Precondition	The dashboards are defined in the configuration (UC_00c).
Precondition	UC_01c – The data are indexed and available for visualization.
Precondition	UC_01 – Monitoring is running and providing data.
Postcondition	The activity is recorded in the audit log.
Postcondition	The report has been generated or exported (if initiated).

Postcondition	The user sees updated visualizations.

Basic Path	Basic Path

Step	**Activity**
1	The actor opens the “Visualization and reporting” module.
2	The system loads the available dashboards and reports according to the actor's role.
3	The actor opens a dashboard → the system displays charts, metrics, tables.
4	The actor can apply filters (time, event type, source).
5	The actor can click on “Create report”.
6	The system compiles the report according to the selected format and template.
7	The actor can use the “Export report” function (PDF, CSV, JSON).

ALT_02_01 – Unavailable data	Alternate

Step	**Activity**
1	The dashboard returns an empty dataset.
2	The system displays a warning.
3	The Operator and the Service operator can change the filtering criteria.

ALT_02_02 – Incorrect permissions	Alternate

Step	**Activity**
1	An actor (e.g. the Service operator) opens a dashboard outside their tenant.
2	The system displays the error “Insufficient permissions”.

ALT_02_03 – Report generation failed	Alternate

Step	**Activity**
1	It displays an error message.
2	It saves the error in the audit log.
3	An actor (e.g. the Service operator) opens a dashboard outside their tenant.

Linked functional requirements	
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
FREQ_006 - Reporting and visualization process	Ensure the reporting and visualization process – ensures regular output for management, auditors and SOC analysts.
UC_02 – Visualize and report security data	The system enables SOC operators and the administrator to display security events using dashboards, filters, charts and tabular outputs. It also enables the generation of reports and the export of data for management, audit or customers.

Use case	UC_02a – Display a dashboard
Objective	The user opens a dashboard that visualizes security events, metrics and system states in real time.

Precondition	The system is initialized and configured (UC_00).
Precondition	All log sources are active and data are being received (UC_01).
Precondition	The user is authenticated and has a role that allows dashboards to be displayed.
Postcondition	The dashboard has been displayed with current data.
Postcondition	The user can save filters, views or sections (if authorized).

Basic Path	Basic Path

Step	**Activity**
1	The actor opens the Dashboards module.
2	The system loads the available dashboards according to the role.
3	The actor selects a dashboard to display.
4	The system loads and renders the visualizations (charts, heatmaps, tables).
5	The system updates the visualizations in real time or at defined intervals.

ALT_02a_02 – The user does not have permission for a specific dashboard	Alternate

Step	**Activity**
1	The system hides the dashboard or displays “Access Denied”.
2	The actor must ask the administrator for access.

ALT_02a_01 – Data fail to load	Alternate

Step	**Activity**
1	The system detects that the sources are unavailable.
2	Instead of charts, it displays the message: “Data are not available.”
3	It offers to refresh or switch the source.

Linked functional requirements	
FREQ_012 - Visualization of outputs	The solution must ensure the visualization of outputs
FREQ_031 - Dashboard design	The solution must ensure dashboard design. To maintain clarity, dashboards within SOCaaS will be divided according to functional layers as follows: 1. Global overviews (SOC Overview) Intended for management and SLA reporting – they include daily/monthly incident statistics, threat types, the top 10 IP addresses, countries of origin of attacks, and ticket resolution time. 2. Alert monitoring dashboards Focused on operational monitoring – they display new alerts, breakdown by severity (Critical, High, Medium), time trends, and correlations between multiple sources. 3. Thematic dashboards They cover specific areas: • EDR activities: recorded files, processes, attempts to execute malicious code. • Vulnerabilities and scans: results from Greenbone or OpenSCAP. • Firewall and network logs: accesses, ports, detected scanning activities. 4. Detailed analytical dashboards – They serve L2 analysts and

	threat hunting. They contain heatmaps, time slices, distributions by hour, country, ASN, IOC occurrence and the like. 5. Ad-hoc and incident-specific dashboards – They are created during the investigation of a specific case, where the analyst builds their own visualization based on data from Elasticsearch.
UC_02b – Create a report	The user creates a one-off report or saves a reporting template based on data from dashboards or events.
FREQ_032 - Dashboard structure	The solution must ensure a dashboard structure in which dashboards are composed of individual panels that may contain charts, tables, heatmaps, distribution histograms, metrics or maps. Each dashboard within SOCaaS is designed as a composition of these visualizations focused on a specific aspect of monitoring or analysis. System security overview Correlation and incident dashboards Visualization of network traffic and access Monitoring of authentications and user behaviour Integrated overview for the SOC analyst
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
Use case	UC_02b – Create a report
Objective	The user creates a one-off report or saves a reporting template based on data from dashboards or events.

Precondition	The dashboards are available and the data are indexed (UC_01).
Precondition	The user has the “Create Report” permission.
Postcondition	The report is saved in the system.
Postcondition	It can be displayed, edited or exported (UC_02c).

Basic Path	Basic Path

Step	**Activity**
1	The actor opens the Reports module.
2	Selects “Create a new report”.
3	Selects: report type, time range, template or custom design.
4	The system loads the necessary data.
5	The user confirms the generation.
6	The system generates the report and saves it in the report database.

ALT_02b_02 – Unavailable data for the report	Alternate

Step	**Activity**
1	The system displays the message “Data for the selected period are not available”.
2	The Operator, Service operator or Administrator selects a different period.

	Alternate

ALT_02b_01 – Invalid report parameter		
Step	**Activity**	
1	The system detects a missing parameter (time period, data source...).	
2	It displays an error and requests that it be supplied.	
Linked functional requirements		
FREQ_033 - Reporting		The solution must ensure reporting. Reporting in SOCaaS will fulfil several basic objectives: • Operational overview – display of the status of systems, the number of incidents, agent activity. • Security reporting – outputs on detected threats, attack types, trends. • Compliance reporting – reports according to regulatory requirements (e.g. GDPR, NIS2). • Management reporting – aggregated statistics and indicators for the organization's management. • Customer reporting – individual overviews for individual clients of the multi-tenant SOCaaS.
UC_02c – Export a report		The user exports an existing report to PDF, CSV, JSON or PNG and can send it by email or via a webhook.
UC_02b – Create a report		The user creates a one-off report or saves a reporting template based on data from dashboards or events.
Use case	UC_02c – Export a report	
Objective	The user exports an existing report to PDF, CSV, JSON or PNG and can send it by email or via a webhook.	
Precondition	There is a generated report (UC_02b).	
Precondition	The user has the “Export Report” rights.	
Postcondition	The system has created a file in the selected format.	
Postcondition	The file has been downloaded or sent to an external endpoint.	
Basic Path		Basic Path
Step	**Activity**	
1	The user opens the report detail.	
2	Selects the Export option.	
3	Selects the format (PDF/CSV/JSON/PNG).	
4	The system generates the export.	
5	The file is downloaded or sent according to the settings.	
ALT_02c_02 – Insufficient permissions		Alternate
Step	**Activity**	
1	The system displays “Access Denied”.	
2	It offers to request access.	
ALT_02c_01 – Export fails (timeout, large dataset)		Alternate

Step	Activity
1	The system reports an export error.
2	It offers an offline export or splitting of the data.
Linked functional requirements	
FREQ_160 - Option to download a report or send it automatically	Exporting reports manually or their scheduled automatic distribution. Possibility of downloading a report or sending it automatically (e-mail, webhook)
FREQ_034 - Manual export of reports	The solution must ensure manual export of reports. It will be possible to export reports as: • PDF document – a generated snapshot of the current dashboard. • PNG image – static images of individual visualizations. • CSV files – raw data from tables and data fields. • JSON – for automated analysis and further processing.
FREQ_165 - Support for export to PDF, CSV, JSON	The possibility of exporting reports to various common formats. Support for export to PDF, CSV, JSON
UC_02c – Export a report	The user exports an existing report to PDF, CSV, JSON or PNG and can send it by email or via a webhook.

4.5.2.5 UC_03 - Manage detection and correlation rules

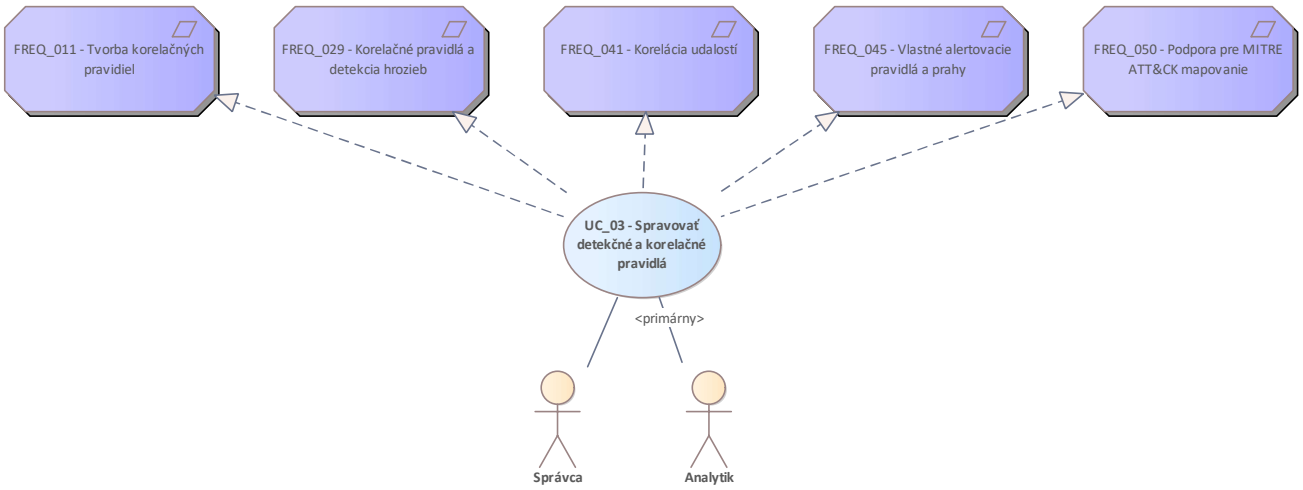


Figure 8 : UC_03 - Manage detection and correlation rules

Use case	UC_03 - Manage detection and correlation rules
Objective	This use case covers the activities of analysts and administrators in creating, managing and maintaining detection and correlation rules in the SOCaaS environment. These rules enable the detection of security threats based on IOCs, behavioural anomalies, event chaining or threshold values. The use case focuses exclusively on the rule management process – their definition, tuning, classification and mapping to security frameworks (e.g. MITRE ATT&CK).

Precondition	The actors have access to the administration interface for creating rules.
Precondition	The system supports the definition, management and versioning of detection rules.

Postcondition	A new rule is created or an existing rule is modified, deactivated or deleted.
Postcondition	The rule is versioned, validated and ready for deployment to the production environment.
Postcondition	The rule is available for the evaluation of security events (UC_04).

Basic Path	Basic Path

Step	**Activity**
1	The analyst or administrator opens the interface for managing detection rules.
2	Creates a new rule or searches for an existing one.
3	Defines the conditions: event type, IOC/indicator, frequency of occurrence, source/destination entities,
4	Assigns the severity, threat type and MITRE ATT&CK category to the rule.
5	The rule is saved and versioned.
6	It can be validated on historical data or marked as active.
7	The rule begins to be used in the evaluation of events (UC_04).

ALT_03_01 – The rule is tested on historical data (debugging)	Alternate

Step	**Activity**
1	The analyst uses a tool for testing rules against historical logs.
2	The results are displayed; the rule is then modified or approved.

ALT_03_02 – The rule is created via a no-code interface	Alternate

Step	**Activity**
1	The analyst uses a visual interface (drag & drop) to create conditions without writing syntax.
2	The platform automatically generates the underlying code of the rule.

Linked functional requirements	
FREQ_041 - Event correlation	The possibility of linking related events using correlation rules. Event correlation (predefined and custom rules)
FREQ_011 - Creation of correlation rules	The solution must ensure the creation of correlation rules and alerts
FREQ_050 - Support for MITRE ATT&CK mapping	The ability to map detections to tactics and techniques according to MITRE ATT&CK. Support for MITRE ATT&CK mapping
FREQ_029 - Correlation rules and threat detection	The solution must provide correlation rules and threat detection. • Alerts on basic IOCs – comparison with a database of known indicators (IP, URL, hash). • Anomalous user behaviour – e.g. multiple failed logins within a short time, a login from a new geographical region. • Non-standard network communications – e.g. DNS queries for dynamically generated domains (DGA), deviations in time and data volume. • Activities with administrative privileges – e.g. adding a user to

	the "Administrators" group, ACL changes, running netsh, powershell. • Event chaining – correlation between multiple logs over time: e.g. login ? file transfer ? execution of a malicious process. These rules will be managed in a way that makes it possible to: • monitor detection alerts, • set threshold values (thresholds), • configure exceptions and ignored values (exceptions), • export and import rules as JSON objects.
FREQ_045 - Custom alerting rules and thresholds	The ability to define custom rules and conditions for generating alerts. Custom alerting rules and thresholds
UC_03 - Manage detection and correlation rules	This use case covers the activities of analysts and administrators in creating, managing and maintaining detection and correlation rules in the SOCaaS environment. These rules enable the detection of security threats based on IOCs, behavioural anomalies, event chaining or threshold values. The use case focuses exclusively on the rule management process – their definition, tuning, classification and mapping to security frameworks (e.g. MITRE ATT&CK).
UC_03 - Manage detection and correlation rules	This use case covers the activities of analysts and administrators in creating, managing and maintaining detection and correlation rules in the SOCaaS environment. These rules enable the detection of security threats based on IOCs, behavioural anomalies, event chaining or threshold values. The use case focuses exclusively on the rule management process – their definition, tuning, classification and mapping to security frameworks (e.g. MITRE ATT&CK).
UC_03 - Manage detection and correlation rules	This use case covers the activities of analysts and administrators in creating, managing and maintaining detection and correlation rules in the SOCaaS environment. These rules enable the detection of security threats based on IOCs, behavioural anomalies, event chaining or threshold values. The use case focuses exclusively on the rule management process – their definition, tuning, classification and mapping to security frameworks (e.g. MITRE ATT&CK).

4.5.2.6 UC_04 – Notify and escalate a security incident

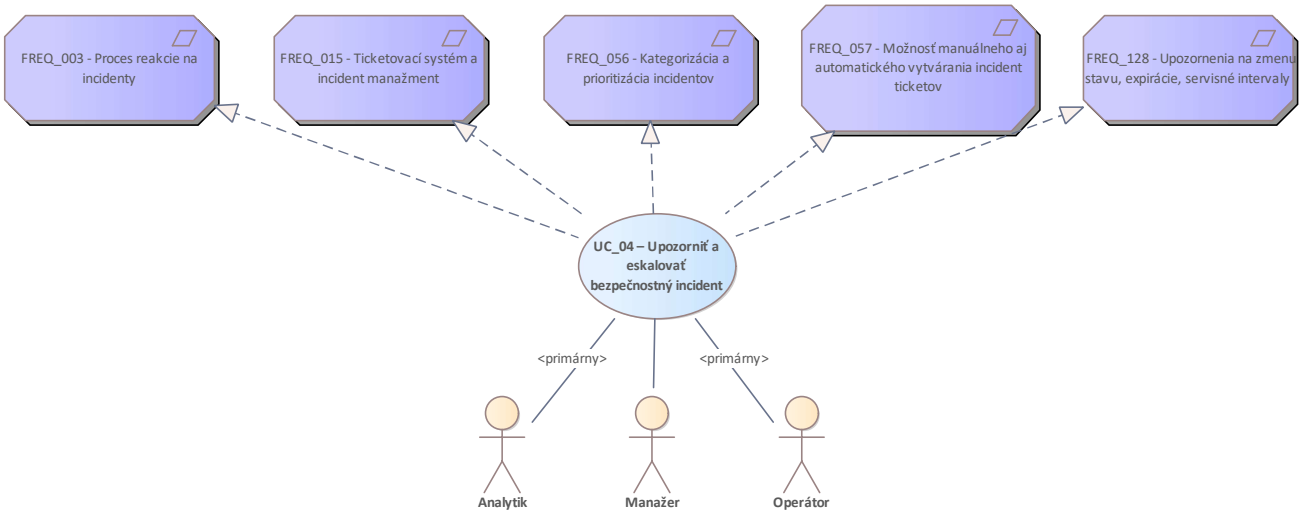


Figure 9 : UC_04 – Notify and escalate a security incident

Use case	UC_04 – Notify and escalate a security incident
Objective	After the system detects and creates a security incident (UC_01, UC_03), the objective of

	UC_04 is to ensure that the right person (1st level) learns about it in time, that someone takes over the incident and that, in the event of inactivity or increasing severity, the incident is automatically escalated to a higher level

Precondition	An incident exists – the incident has been created and has an assigned category and severity
Precondition	Notification and escalation policies are set
Precondition	The contact channels and the on-call schedule are known
Postcondition	The relevant recipient has been notified
Postcondition	The incident has a recorded history of notifications/escalations
Postcondition	In the event of inactivity, escalation has taken place in accordance with the SLA

Basic Path	Basic Path

Step	**Activity**
1	Creation or change of an incident
2	Evaluation of the notification policy
3	Selection of primary recipients
4	Generation of the notification message
5	Sending of the notification via primary channels
6	Waiting for confirmation (acknowledgement)
7	Acknowledgement of the incident by the Analyst/Operator
8	Stopping further escalation
9	Update of the incident history

ALT_01 – Failure of the primary notification channel	Alternate

Step	**Activity**
1	Sending the notification fails (e.g. the e-mail server is unavailable).
2	The system automatically tries an alternative channel (e.g. SMS, another mail gateway, chat).
3	Both the unsuccessful attempt and the subsequent fallback are recorded in the incident history.
4	If all channels fail, the incident is flagged “notification unsuccessful” and the system automatically creates an internal ticket for the infrastructure administrator.

ALT_02 - The incident is not acknowledged within the SLA time - escalation	Alternate

Step	**Activity**
1	After the SLA interval has elapsed, the incident has not been acknowledged.
2	The system automatically escalates the incident to a higher level (Manager, SOC Lead): increasing the priority / severity of the incident, sending an escalation notification to the Manager via their preferred channels.
3	A record of the failure to take over within the SLA and of the escalation performed is added to the incident history.

4	Optionally, a further playbook is launched (e.g. partial automated measures, pre-prepared actions).
Linked functional requirements	
FREQ_003 - Incident response process	Ensure the incident response process – defines the way in which the team responds to identified incidents. It includes escalation, assignment of responsibility, execution of measures and communication with affected parties.
FREQ_015 - Ticketing system and incident management	The solution must ensure a ticketing and incident management system that will provide a workflow for incident handling, escalation, auditing and tracking of resolution.
FREQ_128 - Notifications of status changes, expirations, service intervals	Automatic notifications upon a change of status or the need for service intervention. Alerts on status changes, expirations, service intervals
FREQ_057 - Option of both manual and automatic creation of incident tickets	Support for both manual and automatic creation of incident tickets from input data. Possibility of both manual and automatic creation of incident tickets
FREQ_056 - Incident categorization and prioritization	The possibility of sorting incidents by severity, type, category and importance. Incident categorization and prioritization
UC_04 – Notify and escalate a security incident	After the system detects and creates a security incident (UC_01, UC_03), the objective of UC_04 is to ensure that the right person (1st level) learns about it in time, that someone takes over the incident and that, in the event of inactivity or increasing severity, the incident is automatically escalated to a higher level

4.5.2.7 UC_05 – Handle and close a security incident

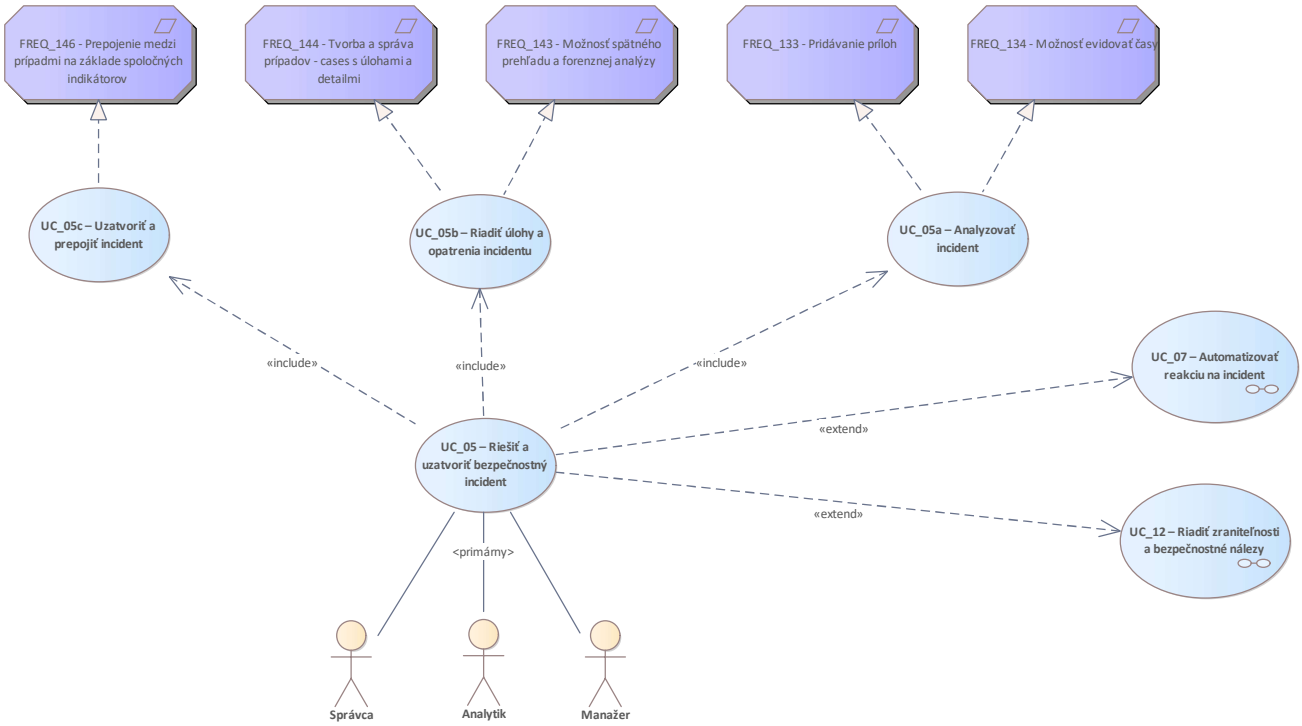


Figure 10 : UC_05 – Resolve and close a security incident

Use case	UC_05 – Handle and close a security incident
Objective	After the detection of an incident and the sending of notifications (UC_01, UC_03, UC_04), this UC covers the entire incident handling process – from takeover, analysis and implementation of corrective measures to the closure of the incident and complete

	documentation of its course (timeline, evidence, linked cases). The result is an incident that has a clear owner, implemented steps, recorded times and prepared supporting materials for any “lessons learned”

Precondition	The incident has been created and notified
Precondition	The incident has an assigned owner / team
Precondition	Relevant data and logs are available
Postcondition	The incident has a documented course and outcome
Postcondition	The agreed set of corrective measures has been carried out
Postcondition	The incident is closed or reclassified

Basic Path	Basic Path

Step	**Activity**
1	Takeover of the incident by the Analyst
2	Collection and supplementing of evidence
3	Analysis and identification of causes
4	Proposal of corrective measures
5	Implementation of measures by the Administrator
6	Tracking of times and the course of handling
7	Review and approval of the outcome
8	Closure of the incident
9	Preparation of supporting materials for lessons learned

ALT_ 01 – The incident is evaluated as a false positive	Alternate

Step	**Activity**
1	After the analysis, the Analyst finds that no incident occurred (incorrect correlation, test traffic).
2	The Analyst updates the incident to the “false positive” status and briefly documents the reason.
3	The incident is closed but remains in the history, including the evidence – it can later be used to adjust detection rules

ALT_02 – Corrective measures cannot be implemented immediately	Alternate

Step	**Activity**
1	The Administrator finds that the proposed measure has a high impact on operations (e.g. shutting down a critical service).
2	In the incident, they create scheduled tasks with deadlines and comments, or a request for approval to the Manager.
3	The incident remains in the “open – awaiting scheduled measures” status with recorded times and risk.

Linked functional requirements	
UC_05a – Analyse an incident	The Analyst examines the incident in detail – reviews the timeline of events, logs and IOCs, adds evidence and builds a comprehensive picture of the cause and scope of the incident.

UC_05b – Manage incident tasks and measures	The Analyst and the Administrator create, assign and track tasks that represent specific corrective or preventive measures related to the incident.
UC_12 – Manage vulnerabilities and security findings	UC_12 covers vulnerability management in SOCaaS: from receiving the outputs of vulnerability scanners, through their assignment to specific assets (UC_11) and prioritization according to risk, to the planning and tracking of corrective measures. The aim is to have a comprehensive overview of where we have security gaps, what is most important to fix and within what time.
UC_05c – Close and link an incident	After the corrective measures have been carried out, the Analyst (or the Manager) decides on the final status of the incident, documents the conclusion and links the incident with other related cases.
UC_07 – Automate incident response	UC_07 addresses the situation in which an incident has already been detected and is being handled (UC_05), but instead of purely manual steps, automated or semi-automated playbooks are used. Based on the incident type and TTPs, the system proposes suitable playbooks, orchestrates technical interventions (blocking, isolation, account disabling, ticketing) and records everything for audit purposes.

Use case	UC_05a – Analyse an incident
Objective	The Analyst examines the incident in detail – reviews the timeline of events, logs and IOCs, adds evidence and builds a comprehensive picture of the cause and scope of the incident.
Precondition	The incident has been created and assigned to the Analyst.
Precondition	Basic logs and events from the monitoring system are available for the incident.
Postcondition	The incident contains supplementary evidence (logs, attachments, IOCs).
Postcondition	The preliminary cause of the incident and the scope of the impact are described.

Basic Path	Basic Path

Step	**Activity**
1	The Analyst opens the incident detail.
2	Displays the timeline of related events and logs.
3	Identifies suspicious events and indicators of compromise.
4	Downloads or requests additional logs/attachments and attaches them to the incident.
5	Based on the collected data, describes the preliminary cause and scope of the incident.
6	Updates the incident priority if necessary.

ALT_01 – Unavailable or incomplete logs	Alternate

Step	**Activity**
1	The Analyst finds that the logs are incomplete or missing; in the incident, they create a task for the Administrator to remedy this (supplementing logging, export from the source system) and continue working with what is available.

ALT_02 – The incident turns out to be a false positive	Alternate

Step	**Activity**
1	After the analysis, the Analyst concludes that it is not a real incident; they mark the incident as a “false positive”, briefly document the reason and UC_05 ends.

Linked functional requirements	
FREQ_134 - Option to record times	Tracking and recording of time data for the entire ticket life cycle. Possibility of recording times (creation, update, escalation, closure)
FREQ_133 - Adding attachments	The possibility of adding attachments such as logs, screenshots and other evidence. Adding attachments (logs, evidence, screenshots, tool outputs)
UC_05a – Analyse an incident	The Analyst examines the incident in detail – reviews the timeline of events, logs and IOCs, adds evidence and builds a comprehensive picture of the cause and scope of the incident.

Use case	UC_05b – Manage incident tasks and measures
Objective	The Analyst and the Administrator create, assign and track tasks that represent specific corrective or preventive measures related to the incident.
Precondition	A basic analysis exists for the incident (UC_05a).
Precondition	It is clear which systems or services the incident affects.
Postcondition	The incident has a created list of tasks with assigned responsible persons and deadlines.
Postcondition	The status of individual tasks reflects the actual progress of handling (planned, in progress, done).

Basic Path	Basic Path
Step	**Activity**
1	Based on the analysis, the Analyst proposes specific corrective measures.
2	In the incident, creates tasks and assigns them to specific Administrators or teams.
3	Sets the priority and deadline for each task.
4	The Administrators continuously update the status of tasks according to the implementation of the measures.
5	The system records the times of task creation, start and completion.
6	The Analyst monitors the status of tasks and, as needed, sends reminders about or escalates delayed tasks.
7	After the key tasks have been completed, the Analyst updates the incident status.

ALT_01 – The task has a high impact on operations	Alternate
Step	**Activity**
1	The Administrator finds that the proposed measure may cause an outage; they mark the task as “requires approval” and the Manager approves or rejects it with a comment.

ALT_02 – The task cannot be implemented by the planned deadline	Alternate
Step	**Activity**
1	The Administrator updates the deadline and the reason in the task; the Analyst adjusts the incident priority or informs the Manager as needed.

Linked functional requirements	
FREQ_143 - Option of retrospective review and forensic analysis	The ability to retrospectively analyse the course and history of incidents. Possibility of retrospective review and forensic analysis
FREQ_144 - Creation and management of cases - cases with tasks and details	The possibility of creating incidents as cases with tasks, a description and attributes. Creation and management of cases with tasks and details

UC_05b – Manage incident tasks and measures	The Analyst and the Administrator create, assign and track tasks that represent specific corrective or preventive measures related to the incident.
Use case	UC_05c – Close and link an incident
Objective	After the corrective measures have been carried out, the Analyst (or the Manager) decides on the final status of the incident, documents the conclusion and links the incident with other related cases.

Precondition	The key tasks of the incident have been completed or a decision has been made to accept the remaining risk.
Precondition	Monitoring shows no persisting indicators of attack, or their occurrence has been explained.
Postcondition	The incident is in one of the final states (closed, false positive, risk accepted, part of a campaign).
Postcondition	The incident is linked with other relevant incidents/cases or campaigns, if any exist.

Basic Path	Basic Path

Step	**Activity**
1	The Analyst checks that all important tasks have been completed and that the systems are stable.
2	Verifies in monitoring that no new related events are appearing.
3	If the incident is part of a wider campaign, searches for and links related incidents.
4	Fills in the final summary: cause, impact, measures taken.
5	Sets the final status of the incident (e.g. “closed”).
6	For critical incidents, asks the Manager to approve the closure.
7	The incident is moved to the archive and is ready for the “lessons learned” process.

ALT_01 – New related events detected during closure	Alternate

Step	**Activity**
1	During the review, the Analyst detects new suspicious events; they leave the incident open, create new tasks and return to UC_05a/05b.

ALT_02 – The incident is part of an extensive campaign	Alternate

Step	**Activity**
1	The Analyst identifies that the incident belongs to a larger campaign; they set it as subordinate to the parent case and adjust the final summary.

Linked functional requirements	
FREQ_146 - Linking between cases based on shared indicators	Automatic or manual identification of related incidents. Linking between cases based on common indicators
UC_05c – Close and link an incident	After the corrective measures have been carried out, the Analyst (or the Manager) decides on the final status of the incident, documents the conclusion and links the incident with other related cases.

4.5.2.8 UC_06 – Evaluate incidents and improve detection and awareness

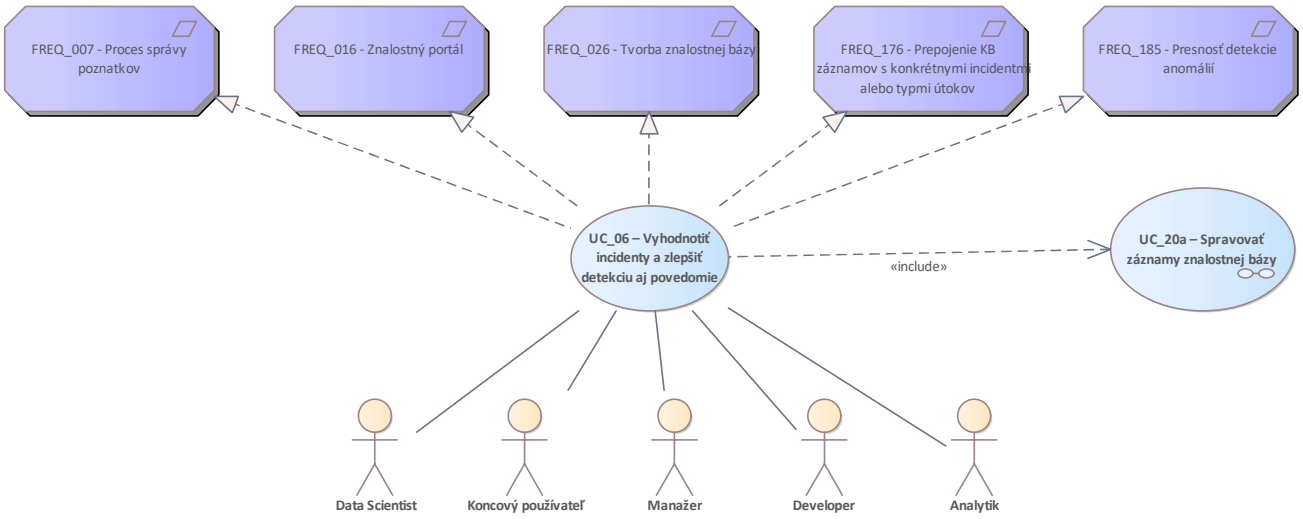


Figure 11 : UC_06 – Evaluate incidents and improve detection and awareness

Use case	UC_06 – Evaluate incidents and improve detection and awareness
Objective	After the incident has been resolved and closed (UC_05), this UC ensures that both the system and people learn from the incident: the incident is reviewed (post-incident / post-mortem), the findings are reflected in adjustments to detection rules and ML models, training / guidance for employees is prepared or updated.

Precondition	The incident is closed
Precondition	Data on the incident are available
Precondition	At least a basic “lessons learned” process exists
Postcondition	The incident has a processed “lessons learned” record
Postcondition	Adjustments to detection/ML have been proposed or carried out
Postcondition	Training / guidance for users has been prepared or updated

Basic Path	Basic Path

Step	**Activity**
1	Selection of incidents for evaluation
2	Preparation of supporting materials for evaluation
3	Post-incident meeting / review
4	Recording of lessons learned and recommendations
5	Proposal of adjustments to detection and ML models

6	The team evaluates how the incident affects detection accuracy (e.g. the number of false positives/false negatives, the suitability of thresholds, the behaviour of the model in the case of anomalies) and incorporates these findings into the proposed adjustments.
7	The knowledge base entry is linked to the specific incident (or attack type) so that it can be found quickly in similar cases.
8	The outputs are stored in the knowledge portal / knowledge base and communicated to the relevant people.
9	Approval and planning of the deployment of changes
10	Preparation of training / awareness materials
11	Dissemination of knowledge
12	Conclusion of the UC

ALT_01 – Insufficient data for a quality evaluation	Alternate

Step	**Activity**
1	During the analysis it turns out that not all necessary logs or historical data are available; the team does carry out a basic lessons learned, but at the same time creates tasks to improve logging and data collection for the future.

ALT_02 – The proposed adjustments to detection/ML are rejected	Alternate

Step	**Activity**
1	The Data Scientist or Developer prepares a proposal for changes, but the Manager does not approve them on the basis of risks or costs; the recommendations are stored in the knowledge base and the changes may be reconsidered later.

ALT_03 – A need for targeted employee training has been identified	Alternate

Step	**Activity**
1	The evaluation shows that the main weakness was human error or lack of knowledge; the Manager decides on the preparation or modification of a specific training course, which is assigned to a selected group of users.

Linked functional requirements	
FREQ_026 - Creation of the knowledge base	The solution must ensure the creation of a knowledge base. Each incident or new type of threat will be documented in the form of: • incident summary + lessons learned • entry in the knowledge portal • categorization according to MITRE ATT&CK and dates
UC_20a – Manage knowledge base records	UC_20a addresses in detail the life cycle of KB articles – creation, editing, structure, tagging and linking to incidents.
FREQ_007 - Knowledge management process	Ensure the knowledge management process – ensures the maintenance of a knowledge base of playbooks, documentation and recommended responses.
FREQ_176 - Linking KB entries with specific incidents or attack types	The possibility of linking knowledge articles with specific attack types and incidents in other tools. Linking KB records with specific incidents or attack types
UC_20 – Manage the knowledge base and playbook library	UC_20 ensures that the SOC has, in a centralized and managed manner: a knowledge base (KB) – articles, guides, lessons learned, procedures, a playbook library – both manual and automated response procedures, a process for approving, versioning and publishing this content.
FREQ_185 - Anomaly detection accuracy	The solution must ensure evaluation of anomaly detection accuracy (Precision, Recall, F1). How accurately the algorithm detects anomalies regardless of class. Evaluation of overall accuracy using metrics such as

	Precision, Recall, F1-score or AUC-ROC.
FREQ_016 - Knowledge portal	The solution must ensure a knowledge portal that will support response processes by sharing playbooks, internal policies and guides.

4.5.2.9 UC_07 – Automate incident response

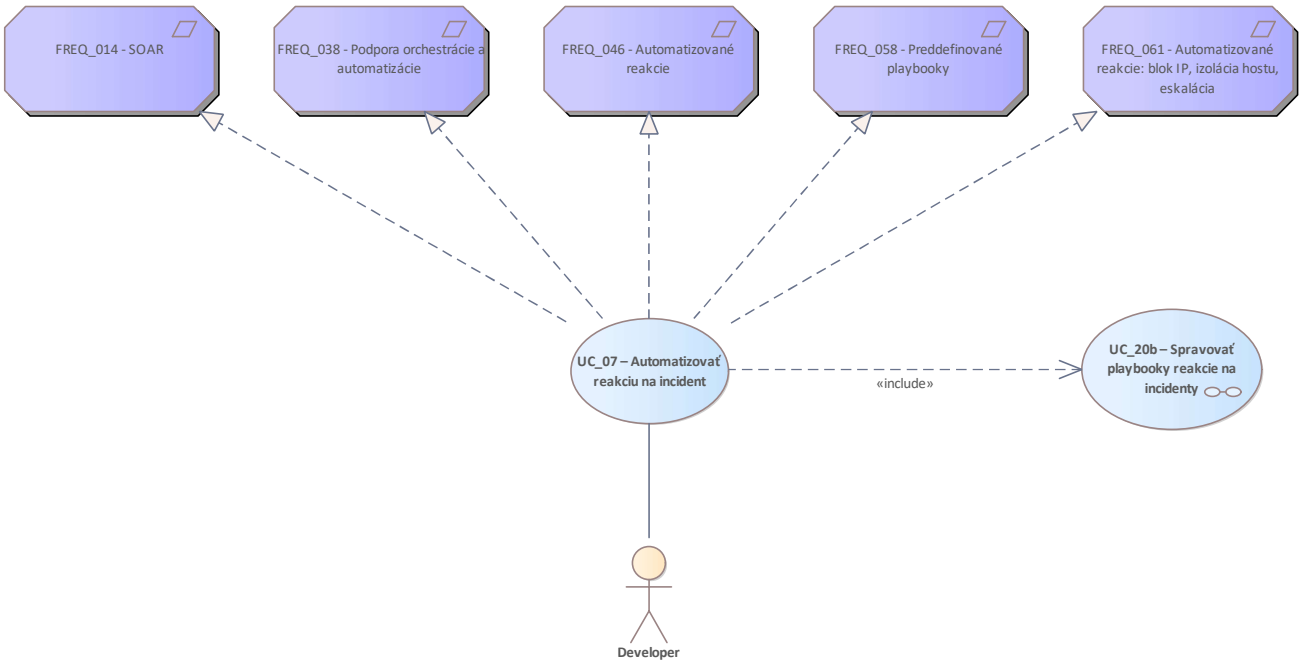


Figure 12 : UC_07 – Automate incident response

Use case	UC_07 – Automate incident response
Objective	UC_07 addresses the situation in which an incident has already been detected and is being handled (UC_05), but instead of purely manual steps, automated or semi-automated playbooks are used. Based on the incident type and TTPs, the system proposes suitable playbooks, orchestrates technical interventions (blocking, isolation, account disabling, ticketing) and records everything for audit purposes.

Precondition	The incident is in the “in progress” status
Precondition	Playbooks are available for the given incident type
Precondition	Integrations with the target systems are functional
Postcondition	The selected playbooks have been executed (auto / semi-auto)
Postcondition	All automated interventions are recorded in the incident for audit purposes
Postcondition	The incident is in a status that reflects the outcome of the automated response
Basic Path	Basic Path

Step	**Activity**

1	The Analyst opens an incident in the “in progress” status and the system displays a list of proposed playbooks according to the incident type and the techniques used.
2	The Analyst selects a suitable playbook and chooses the execution mode (e.g. simulation, semi-automatic mode, fully automatic at low risk).
3	The system displays an overview of the planned playbook steps (e.g. IP block, endpoint isolation, account disabling, ticket creation).
4	In semi-automatic mode, before executing each step, the system requests the Analyst to confirm or skip the step.
5	The system gradually executes the approved steps via integrations with the target systems and continuously records the status (success/failure).
6	After the playbook finishes, the system summarizes the outcome: which actions were successful, what failed and where manual intervention is needed.
7	The results of the executed actions and the playbook timeline are saved in the incident as part of the handling history.
8	Based on the outcome, the Analyst updates the incident status (e.g. containment done, awaiting monitoring, awaiting further tasks according to UC_05).
9	The incident continues in UC_05 (detailed analysis, addition of tasks, closure).

ALT_01 – No suitable playbook was found	Alternate

Step	**Activity**
1	The system is unable to assign any relevant playbook to the given incident; the Analyst continues handling the incident manually according to UC_05 and marks the incident as a candidate for the creation of a new playbook in the future.

ALT_02 – A critical playbook step fails	Alternate

Step	**Activity**
1	An error occurs during the execution of an automated step (e.g. blocking on the firewall); the system marks the step as unsuccessful, records the error details and the Analyst decides whether to repeat the step, replace it with another measure or assign a manual task to the Administrator.

ALT_03 – The playbook is run only in simulation mode	Alternate

Step	**Activity**
1	Based on the Manager's policy, selected playbooks are permitted only in simulation mode; the Analyst runs the simulation, the system shows which actions would be executed and what the impact would be, and only then does the Analyst decide which of them will be executed manually or through partial automation.

Linked functional requirements	
FREQ_014 - SOAR	The solution must ensure SOAR, which will be used to automate response scenarios, process alerts into cases and analyse them.
UC_20 – Manage the knowledge base and playbook library	UC_20 ensures that the SOC has, in a centralized and managed manner: a knowledge base (KB) – articles, guides, lessons learned, procedures, a playbook library – both manual and automated response procedures, a process for approving, versioning and publishing this content.
UC_20b – Manage incident response playbooks	UC_20b addresses the design, editing and maintenance of playbooks that the SOC uses for automated or semi-automated response (UC_07).
FREQ_046 - Automated responses	The possibility of triggering a response (blocking, isolation, notification) automatically. Automated responses (e.g. IP blocking, SOAR integration)
FREQ_038 - Support for orchestration and automation	The solution is to support tools such as Heat (OpenStack orchestration), Terraform (infrastructure as code), Ansible (automation) or Kubernetes Operator for the efficient management and deployment of complex

	infrastructure scenarios. This capability will enable rapid scaling of services, automated roll-out of environments, repeatable testing and recovery from outages. In a SOC environment, where requirements and network topology change frequently, orchestration is essential for ensuring flexibility, sustainability and operational efficiency.
FREQ_061 - Automated responses: IP block, host isolation, escalation	The ability to automatically perform responses such as blocking, isolation or notification. Automated responses: IP block, host isolation, escalation
FREQ_058 - Predefined playbooks	Availability of basic response scenarios and automations prepared in advance in the tool. Predefined playbooks
UC_07 – Automate incident response	UC_07 addresses the situation in which an incident has already been detected and is being handled (UC_05), but instead of purely manual steps, automated or semi-automated playbooks are used. Based on the incident type and TTPs, the system proposes suitable playbooks, orchestrates technical interventions (blocking, isolation, account disabling, ticketing) and records everything for audit purposes.

4.5.2.10 UC_08 – Proactive threat hunting

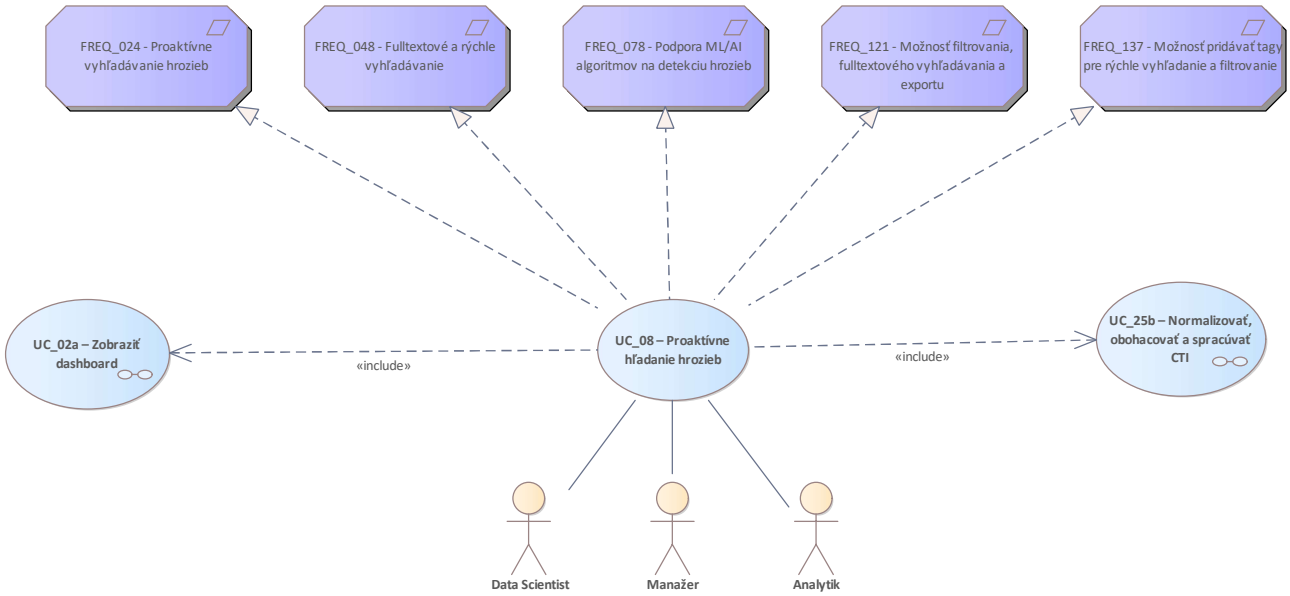


Figure 13 : UC_08 – Proactive threat hunting

Use case	UC_08 – Proactive threat hunting
Objective	UC_08 addresses proactive threat hunting in historical and current data beyond regular alerts. The Analyst purposefully searches for suspicious behaviour patterns (TTPs, IOCs) that have not yet triggered an incident or an alert, saves hunting scenarios and can create new incidents directly from the findings.

Precondition	Historical and current logs/telemetry are available
Precondition	At least a basic set of hunting queries or scenarios exists
Precondition	The Analyst has access to dashboards and tools for ad-hoc queries
Postcondition	At least one hunting run has taken place

Postcondition	The findings are documented and linked to incidents or the knowledge base
Postcondition	The hunting scenarios are saved for reuse

Basic Path	Basic Path

Step	**Activity**
1	Selection of the hunting target
2	Opening of the hunting workspace
3	Creation or selection of a query / scenario
4	Launching a hunting run over the data
5	Analysis of findings
6	Conversion of findings into incidents or enrichment of existing incidents
7	Saving of the hunting scenario
8	Documentation of results
9	Planning of repeated hunting

ALT_01 – Hunting reveals nothing significant	Alternate

Step	**Activity**
1	The hunting query does not return any suspicious findings, or all findings are legitimate after review; the Analyst saves the hunting record as “no significant findings” and may adjust the query for future use.

ALT_02 – Hunting reveals an extensive campaign	Alternate

Step	**Activity**
1	Hunting identifies several related findings that indicate a more extensive attack; the Analyst creates a parent incident/campaign, assigns all related findings and incidents to it and informs the Manager of the need for increased attention.

ALT_03 – The query is too demanding or slow	Alternate

Step	**Activity**
1	The launched hunting query runs over a large volume of data and the systems evaluate it as too demanding; the system notifies the Analyst, stops or limits the query and offers the option to adjust the time range or granularity.

Linked functional requirements	
FREQ_121 - Option of filtering, full-text search and export	Efficient work with data – searching by various parameters and exporting data. Possibility of filtering, full-text search and export
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
FREQ_024 - Proactive threat hunting	The solution must ensure proactive threat hunting (Threat Hunting). Advanced queries will be used, often supplemented with: • CTI data • manual analysis in the case of suspicious entities • the results are documented as playbooks
UC_25c – Distribute CTI and IOCs to detection and response systems	UC_25c addresses the push of CTI to the rest of the ecosystem: forwarding IOCs to SIEM/IDS/EDR/firewalls, use of CTI in detection, playbooks and incidents,

FREQ_137 - Option to add tags for fast searching and filtering	Use of tags to facilitate searching and categorization. Possibility of adding tags for quick searching and filtering
FREQ_078 - Support for ML/AI algorithms for threat detection	Use of artificial intelligence algorithms to detect anomalies and threats. Support for ML/AI algorithms for threat detection
FREQ_048 - Full-text and fast search	The ability to search logs using operators, tags and filters. Full-text and fast search
UC_25b – Normalize, enrich and process CTI	UC_25b addresses the CTI processing pipeline: normalization of various formats into a unified model, enrichment of IOCs and artefacts from external services, workflow for automatic/semi-automatic processing of CTI.

4.5.2.11 UC_09 – Manage Threat Intelligence and indicators of compromise

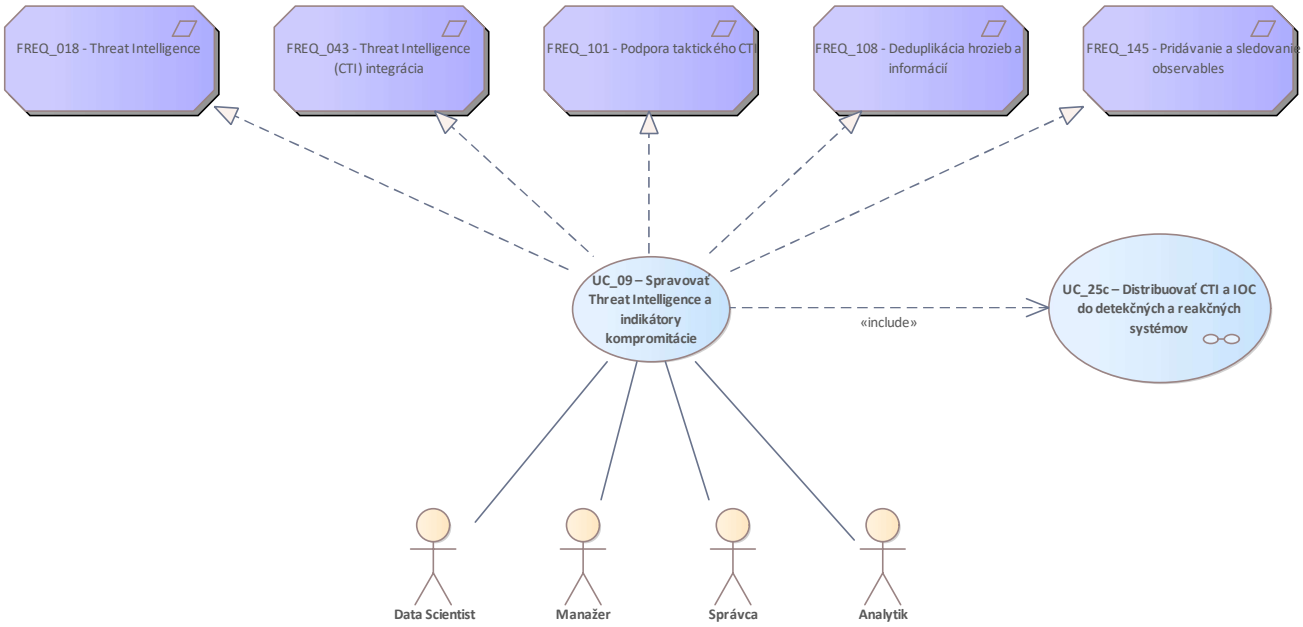


Figure 14 : UC_09 – Manage Threat Intelligence and indicators of compromise

Use case	UC_09 – Manage Threat Intelligence and indicators of compromise
Objective	UC_09 addresses the management of Threat Intelligence: the reception and integration of CTI feeds, work with indicators of compromise (IOCs), their deduplication and tracking over time. The aim is for the SOC to be able to centrally manage threats, enrich incidents and logs with TI information and keep its observables in order.

Precondition	TI sources are available
Precondition	Integrations with TI sources are configured
Precondition	The system has storage for TI and IOCs
Postcondition	TI data are loaded, normalized and stored
Postcondition	Redundant or conflicting IOCs are deduplicated or merged

Postcondition	IOCs/observables are linked to incidents, cases or assets and have a traceable history

Basic Path	Basic Path

Step	**Activity**
1	Loading of TI from external sources
2	Normalization and storage of TI
3	Deduplication and merging of IOCs
4	Management of the IOC life cycleManagement of the IOC life cycle
5	Linking of IOCs to incidents and cases
6	Publication of TI to other parts of the system
7	Monitoring of TI quality

ALT_01 – Failure or unavailability of a TI source	Alternate

Step	**Activity**
1	When loading TI, the source is temporarily unavailable or returns errors; the system records an integration incident, uses the last valid set of IOCs and notifies the Administrator of the problem.

ALT_02 – Conflicting information about the same threat	Alternate

Step	**Activity**
1	Several TI sources provide different or contradictory information about the same IOC; the system retains all variants with their sources, marks the IOC as conflicting and leaves it to the Analyst to select the more trustworthy source.

ALT_03 – The IOC is evaluated as outdated or unreliable	Alternate

Step	**Activity**
1	The Analyst finds that the given IOC is no longer relevant or has been marked as a mistake; they change its status to inactive, add a comment and the system ensures that it is no longer used in new detections.

Linked functional requirements	
FREQ_101 - Support for tactical CTI	Working with indicators of compromise (IOC) and their recording, processing and distribution. Support for tactical CTI (IOC, IP, hash, URL, domains)
FREQ_108 - Deduplication of threats and information	Removal of redundant IOCs and indicators. Deduplication of threats and information
FREQ_145 - Adding and tracking observables	The ability to track IOCs as part of a case. Adding and tracking observables (IP, domains, URL, hash values)
FREQ_018 - Threat Intelligence	The solution must ensure Threat Intelligence. CTI (Cyber Threat Intelligence) processes enable the enrichment of incidents with known threats, TTPs and IOCs
UC_25c – Distribute CTI and IOCs to detection and response systems	UC_25c addresses the push of CTI to the rest of the ecosystem: forwarding IOCs to SIEM/IDS/EDR/firewalls, use of CTI in detection, playbooks and incidents,
FREQ_043 - Threat Intelligence (CTI) integration	Integration with threat sources (IOC, TTP), automatic data enrichment. Threat Intelligence (CTI) integration

4.5.2.12 UC_10 – Manage compliance, metrics and management reporting

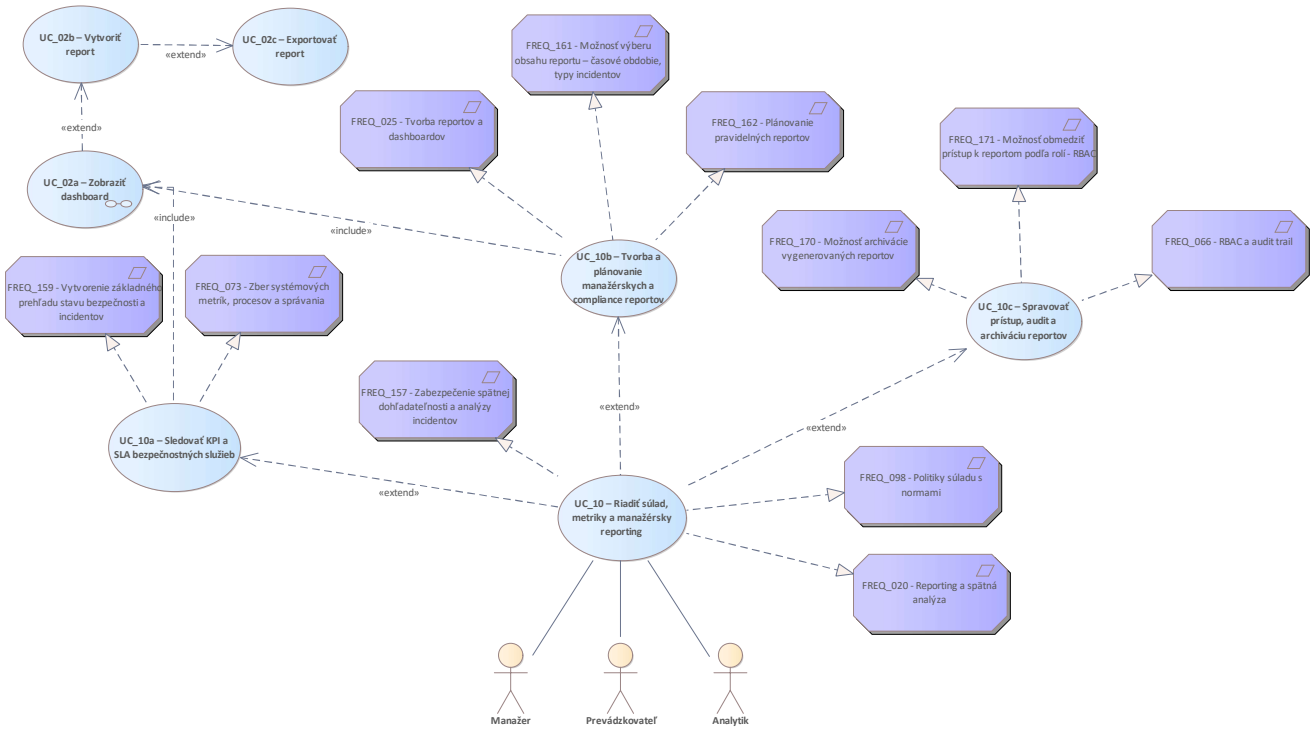


Figure 15 : UC_10 – Manage compliance, metrics and management reporting

Use case	UC_10 – Manage compliance, metrics and management reporting
Objective	UC_10 addresses the “top floor” of SOCaaS – so that management and auditors have an overview of: how well the SOC is performing (KPIs, SLAs, incident trend), whether the system complies with standards/regulations, whether all important matters are traceable retrospectively through reports, logs and audits.

Precondition	The system collects data on incidents, responses, playbooks and TI over the long term
Precondition	Defined metrics/KPIs and SLAs exist
Precondition	At least a basic compliance policy is set (standards, internal guidelines)
Postcondition	The Manager and the Auditor have clear metrics and reports on the state of security and SOC services at their disposal
Postcondition	Identified problems in KPIs/SLAs or compliance are recorded and have assigned recommendations/measures
Postcondition	The created reports are archived and traceable for future audits

Basic Path	Basic Path

Step	**Activity**
1	The Manager opens the SOC management overview
2	Selection of the area of interest

3	Display of aggregated metrics and trends
4	Identification of deviations or problems
5	Detailed drill-down
6	Submission of a request for a report / analysis
7	Recording of findings and recommendations
8	Linking to measures

ALT_01 – Unavailable or incomplete metrics	Alternate

Step	**Activity**
1	When the management overview is displayed, the system indicates that some metrics cannot be reliably calculated (missing data, incomplete logs); the Manager receives information on which parts are incomplete and can request remediation.

ALT_02 – Identified breach of an SLA or a standard	Alternate

Step	**Activity**
1	The metrics and reports show that SLAs or the requirements of a standard have been breached; the Manager registers the finding as an official finding and launches internal measures (e.g. a service quality incident, corrective measures).

Linked functional requirements	
UC_10a – Monitor KPIs and SLAs of security services	UC_10a extends UC_10 with detailed monitoring of KPIs, SLAs and the overall state of security. The Manager, the Auditor or the Service operator view, in the dashboards, trend charts, overviews by team/customer and the overall "health status" of SOCaaS.
FREQ_098 - Policies for compliance with standards	The possibility of assessing vulnerabilities in relation to standards. Standards compliance policies (e.g. GDPR, ISO, CIS, BSI...)
FREQ_157 - Ensuring retrospective traceability and analysis of incidents	Makes it possible to obtain the complete timeline of an incident. Ensuring retrospective traceability and analysis of incidents
FREQ_020 - Reporting and retrospective analysis	The solution must ensure reporting and retrospective analysis.
UC_10b – Creation and scheduling of management and compliance reports	UC_10b addresses practical work with reports for management and audits: creation, configuration of content, scheduling of regular reports and their one-off generation.
UC_10c – Manage access, audit and archiving of reports	UC_10c addresses the security and audit aspect of reporting: who sees what, who made what available, whether reports are archived and whether it is possible to reconstruct who had what data at their disposal.

Use case	UC_10a – Monitor KPIs and SLAs of security services
Objective	UC_10a extends UC_10 with detailed monitoring of KPIs, SLAs and the overall state of security. The Manager, the Auditor or the Service operator view, in the dashboards, trend charts, overviews by team/customer and the overall "health status" of SOCaaS.

Precondition	From UC_10 – the management overview is available.
Precondition	KPIs and SLAs are defined (e.g. MTTR, MTTD, % of incidents within the SLA).
Postcondition	Identified deviations in KPIs/SLAs are documented.
Postcondition	The Manager has an overview of which areas/teams/customers require attention.

Basic Path	Basic Path

Step	**Activity**
1	In UC_10, the Manager selects the “KPIs and SLAs” area.
2	The system displays a dashboard with KPIs and SLAs (include UC_02a – Display a dashboard).
3	The Manager filters the metrics by period, team, incident type or customer.
4	The system displays trend charts, averages, SLA fulfilment percentages and comparisons with target values.
5	The Manager identifies problems (e.g. a specific team has long response times) and marks them for detailed analysis or corrective measures.

ALT_10a_01 – KPIs or SLAs are not yet defined	Alternate

Step	**Activity**
1	When an attempt is made to display KPIs/SLAs, the system indicates that they are not configured; the Manager receives a recommendation to define the metrics and the assigned data sources.

Linked functional requirements	
FREQ_073 - Collection of system metrics, processes and behaviour	Monitoring of system behaviour and metrics such as CPU, memory, services and processes. Collection of system metrics, processes and behaviour
FREQ_159 - Creation of a basic overview of the state of security and incidents	The possibility of creating summaries of the security status, incidents and system activity. Creation of a basic overview of the security status and incidents
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
UC_10a – Monitor KPIs and SLAs of security services	UC_10a extends UC_10 with detailed monitoring of KPIs, SLAs and the overall state of security. The Manager, the Auditor or the Service operator view, in the dashboards, trend charts, overviews by team/customer and the overall “health status” of SOCaaS.

Use case	UC_10b – Creation and scheduling of management and compliance reports
Objective	UC_10b addresses practical work with reports for management and audits: creation, configuration of content, scheduling of regular reports and their one-off generation.

Precondition	The system has aggregated data from the other UCs (incidents, TI, playbooks, KPIs) at its disposal.
Precondition	The user has sufficient permissions to create reports.
Postcondition	Report templates for management/the regulator are created.
Postcondition	Both regular and ad-hoc reports are set up according to needs.

Basic Path	Basic Path

Step	**Activity**
1	The Manager or the Auditor opens the “Reports” section within UC_10.
2	Selects the report type (management overview, incident overview, compliance report).
3	The system makes it possible to configure the report content – selection of metrics, time period, filters.
4	The user chooses whether it is a one-off report or a regular one (daily/weekly/monthly).
5	When generating the report, the system uses UC_02b (Create a report) and UC_02c (Export a

	report) – PDF/CSV/…
6	The generated report is saved and is available in the report archive.

ALT_01 – Incomplete report due to missing data	Alternate

Step	**Activity**
1	When generating the report, the system warns that some parts will be incomplete; the user can decide to generate the report anyway or to adjust the scope.

Linked functional requirements	
FREQ_025 - Creation of reports and dashboards	The solution must ensure the creation of reports and dashboards. The following will be regularly generated for management and customers: • incident overviews • visualizations from metrics and ticketing • exports for audit and SLA reporting
FREQ_162 - Scheduling of regular reports	The possibility of setting up report generation based on a schedule. Scheduling of regular (e.g. daily, weekly) reports
UC_02a – Display a dashboard	The user opens a dashboard that visualizes security events, metrics and system states in real time.
FREQ_161 - Option to select report content – time period, incident types	The possibility of filtering report content by time, type or source data. Possibility of selecting report content – time period, incident types
UC_10b – Creation and scheduling of management and compliance reports	UC_10b addresses practical work with reports for management and audits: creation, configuration of content, scheduling of regular reports and their one-off generation.

Use case	UC_10c – Manage access, audit and archiving of reports
Objective	UC_10c addresses the security and audit aspect of reporting: who sees what, who made what available, whether reports are archived and whether it is possible to reconstruct who had what data at their disposal.

Precondition	Generated reports exist in the system (UC_10b).
Precondition	Roles and access policies (RBAC) are defined.
Postcondition	Access to reports is restricted according to roles and responsibilities.
Postcondition	An audit trail exists of who worked with the reports and when.
Postcondition	Important reports are archived for the purposes of future audits.

Basic Path	Basic Path

Step	**Activity**
1	The Manager or the Auditor opens the management of reports and access.
2	For individual report types, sets which roles or specific users have access to them.
3	The system applies RBAC when reports are displayed and downloaded.
4	With each display, download or change of access settings, the system writes an event to the audit trail.
5	Older reports are archived automatically or manually so that they are available for audit but do not overload the standard interface.

ALT_01 – Attempt to access a report without authorization	Alternate

Step	**Activity**

1	A user attempts to open a report for which they do not have the role; the system denies access, records the event in the audit trail and may notify the Manager or the Auditor.
Linked functional requirements	
FREQ_066 - RBAC and audit trail	Role-based access control and logging of changes in the system. RBAC and audit trail
FREQ_170 - Option to archive generated reports	The possibility of retaining reports for retrospective analysis and audit. Possibility of archiving generated reports
FREQ_171 - Option to restrict access to reports by role - RBAC	Access to reports controlled by roles and permissions. Possibility of restricting access to reports by role (RBAC)
UC_10c – Manage access, audit and archiving of reports	UC_10c addresses the security and audit aspect of reporting: who sees what, who made what available, whether reports are archived and whether it is possible to reconstruct who had what data at their disposal.

4.5.2.13 UC_11 – Manage the customer's assets and environment

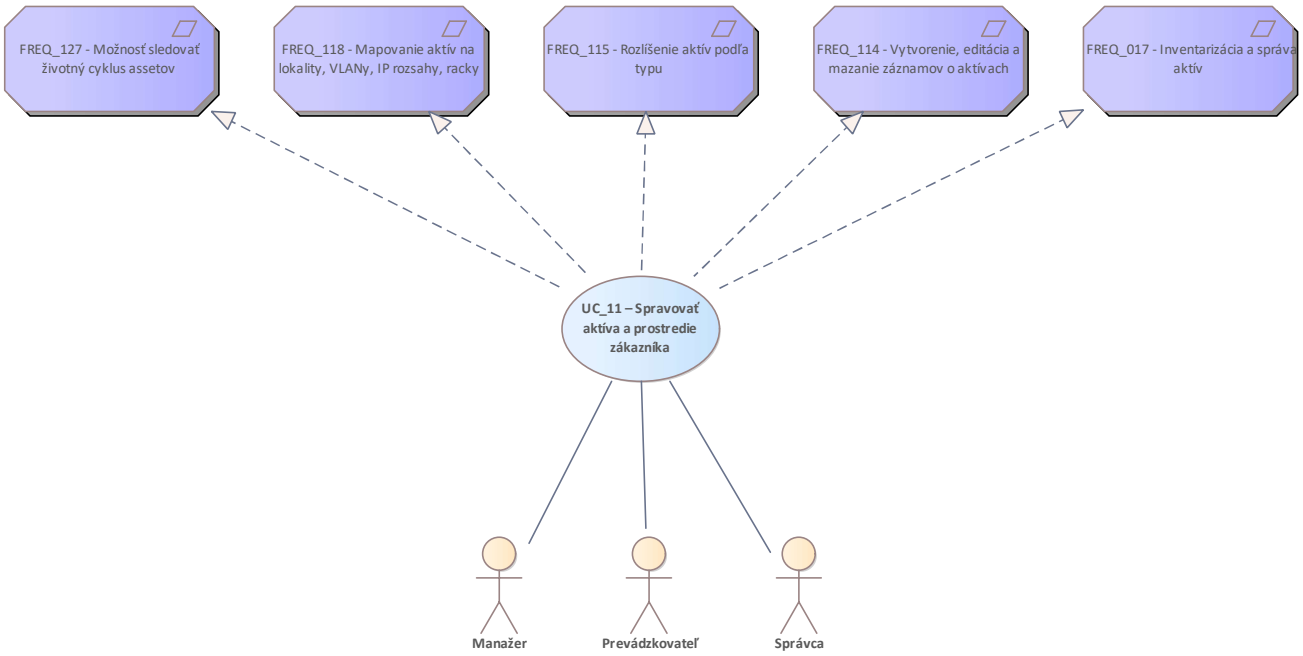


Figure 16 : UC_11 – Manage assets and the customer environment

Use case	UC_11 – Manage the customer's assets and environment
Objective	The objective of UC_11 is for SOCaaS to know what it is actually protecting – all relevant assets of the customer (servers, applications, networks, cloud services...) and their relationships. It makes it possible to record assets, distinguish their types, map them to locations/VLANs/IPs and track their life cycle so that incidents, threats and vulnerabilities can be tied to specific parts of the environment.

Precondition	The customer has been onboarded in the system (UC_00) and has a basic profile created.
Precondition	The system has a module / section for asset recording at its disposal.
Precondition	At least a basic list of systems/services that need to be included in the inventory exists.

Postcondition	The customer's assets are recorded in the system with their type, location and basic properties.
Postcondition	Each asset has a unique identifier and can be used in incidents, TI, vuln mgmt and reports.
Postcondition	Changes to assets (addition, modification, decommissioning) are recorded and the life cycle is traceable.

Basic Path	Basic Path

Step	**Activity**
1	The Service operator opens the “Assets” module for a specific customer.
2	Creates a new asset record and enters its basic data (name, description, owner, criticality).
3	Selects the asset type (e.g. server, application, cloud service, network element) according to predefined categories.
4	The Administrator adds technical data – IP addresses, hostname, environment (prod/test/dev).
5	The system makes it possible to map the asset to a specific location, VLAN, IP range or rack.
6	The Service operator sets the status of the asset in its life cycle (e.g. “in operation”, “being deployed”, “decommissioned”).
7	The system saves the asset record and makes it available to other modules (incident mgmt, TI, vuln mgmt, reporting).
8	With any further change (upgrade, migration, decommissioning), the Administrator updates the asset record and the system records the change in the history.

ALT_11_01 – Import of assets from an external CMDB	Alternate

Step	**Activity**
1	Instead of creating records manually, the Service operator selects import from an external CMDB or inventory tool; the system loads the list of assets, displays a preview and allows them to be saved in bulk or matched with existing records.

ALT_11_02 – Duplicate asset detected	Alternate

Step	**Activity**
1	When saving a new asset, the system detects that a record with the same key identifiers (e.g. IP/hostname) already exists; the user is given the option either to update the existing record or to create a separate asset with a justification.

ALT_11_03 – Change of asset criticality	Alternate

Step	**Activity**
1	The Manager decides that the asset has a higher/lower criticality; they modify the value in the asset record and the system flags the related incidents and vulnerabilities whose priority may change as a result of this decision.

Linked functional requirements	
FREQ_127 - Option to track the life cycle of assets	Management of the individual phases of an asset's life – from commissioning to decommissioning. Possibility of tracking the asset life cycle (deployment – decommissioning)
FREQ_115 - Distinguishing assets by type	The possibility of distinguishing asset types for better organization and management. Distinguishing assets by type – hardware, software, virtual, cloud
FREQ_118 - Mapping of assets to locations, VLANs, IP ranges, racks	The ability to map the physical and logical location of assets in the network. Mapping of assets to sites, VLANs, IP ranges, racks
FREQ_017 - Asset inventory and management	The solution must ensure asset inventory and management. IT asset management is key to mapping the impacts of threats.

FREQ_114 - Creation, editing and deletion of asset records	The ability of the system to record, modify and delete information on individual assets. Creation, editing and deletion of asset records

4.5.2.14 UC_12 – Manage vulnerabilities and security findings

UC_12 covers vulnerability management in SOCaaS:
from receiving the outputs of vulnerability scanners, through their assignment to specific assets (UC_11) and prioritization according to risk, to the planning and tracking of corrective measures.
The aim is to have a comprehensive overview of where we have security gaps, what is most important to fix and within what time.

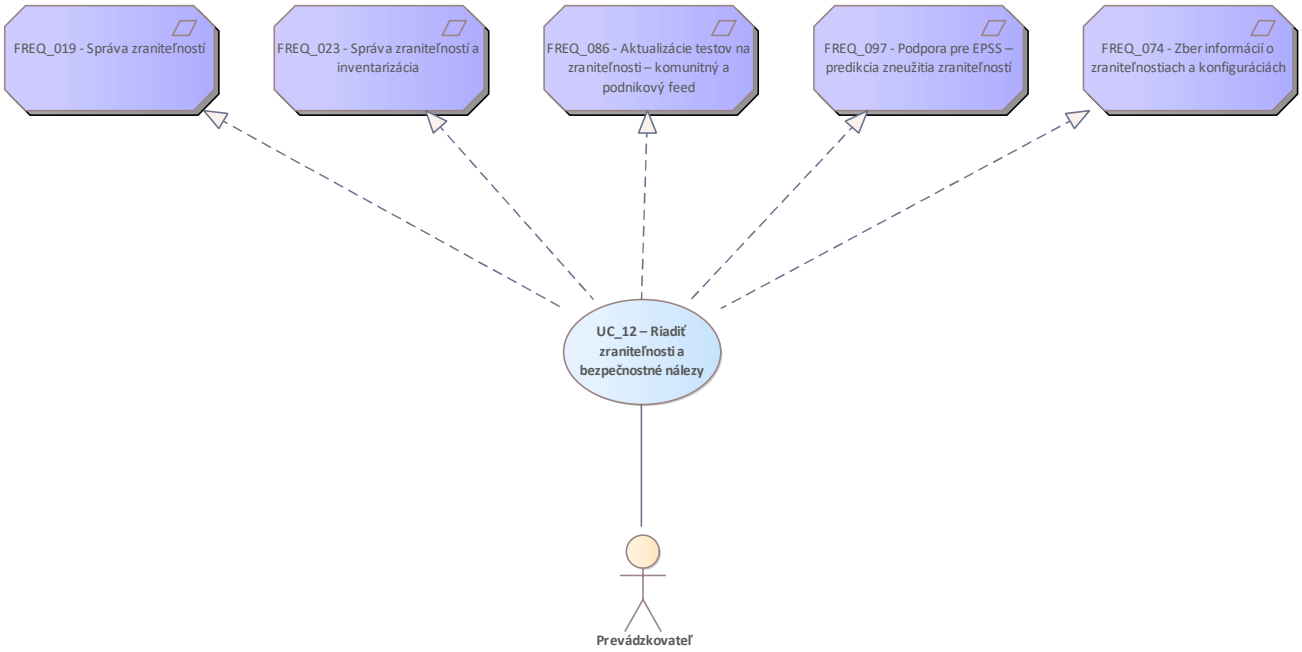


Figure 17 : UC_12 – Manage vulnerabilities and security findings

Use case	UC_12 – Manage vulnerabilities and security findings
Objective	UC_12 covers vulnerability management in SOCaaS: from receiving the outputs of vulnerability scanners, through their assignment to specific assets (UC_11) and prioritization according to risk, to the planning and tracking of corrective measures. The aim is to have a comprehensive overview of where we have security gaps, what is most important to fix and within what time.

Precondition	Vulnerability scan results or other security findings exist
Precondition	Assets are recorded in the system
Precondition	A basic vulnerability prioritization policy is defined
Postcondition	Vulnerabilities are recorded and assigned to specific assets.
Postcondition	Each vulnerability has a determined priority and a planned method of remediation.
Postcondition	The resolution status (unfixed / in progress / fixed / risk accepted) is tracked and traceable.
Basic Path	Basic Path

Step	Activity
1	Import or entry of vulnerabilities
2	Normalization and grouping of findings
3	Assignment of vulnerabilities to assets
4	Calculation and setting of vulnerability priorityCalculation and setting of vulnerability priority
5	Creation of a remediation plan
6	Implementation of corrective measures
7	Re-scan or status verification
8	Update of vulnerability status
9	Reporting and linking with incidents

ALT_12_01 – The vulnerability cannot be fixed within the required time	Alternate

Step	Activity
1	The Administrator finds that the vulnerability cannot be fixed (technical limits, no vendor patch exists, high impact on operations); after discussion, the Manager decides whether the risk is temporarily accepted or compensating measures are introduced (network segmentation, increased monitoring). The status of the vulnerability is set to “risk accepted” or “compensated”.

ALT_12_02 – Duplicate or obsolete finding detected	Alternate

Step	Activity
1	When importing scan results, the system detects that the same vulnerability has already been recorded and in the meantime marked as fixed/irrelevant; the finding is marked as duplicate or obsolete, while a reference record is retained for audit.

ALT_12_03 – The vulnerability leads to the opening of an incident	Alternate

Step	Activity
1	The Analyst evaluates a specific vulnerability (e.g. a critical, actively exploited one) as an acute threat; they create a security incident for it, which begins to be handled according to UC_05, while UC_12 is used to track the long-term remediation.

Linked functional requirements	
FREQ_097 - Support for EPSS – prediction of vulnerability exploitation	Use of models of the probability of vulnerability exploitation. Support for EPSS – prediction of vulnerability exploitation
FREQ_023 - Vulnerability management and inventory	The solution must ensure vulnerability management and inventory. Regular scans are to be linked to the CMDB. • Assets are updated on the basis of detection • incidents linked to vulnerabilities are connected to the management system
FREQ_086 - Updates of vulnerability tests – community and enterprise feed	Frequency and scope of updates of detection patterns. Vulnerability test updates – community and enterprise feed
FREQ_019 - Vulnerability management	The solution must ensure vulnerability management.
FREQ_074 - Collection of information on vulnerabilities and configurations	Obtaining information on vulnerabilities from configuration data and comparison with CVE databases. Collection of information on vulnerabilities and configurations
UC_12 – Manage vulnerabilities and security findings	UC_12 covers vulnerability management in SOCaaS: from receiving the outputs of vulnerability scanners, through their assignment to specific assets (UC_11) and prioritization according to

	risk, to the planning and tracking of corrective measures. The aim is to have a comprehensive overview of where we have security gaps, what is most important to fix and within what time.

4.5.2.15 UC_13 – Design and maintain the security architecture of the customer's SOCaaS environment

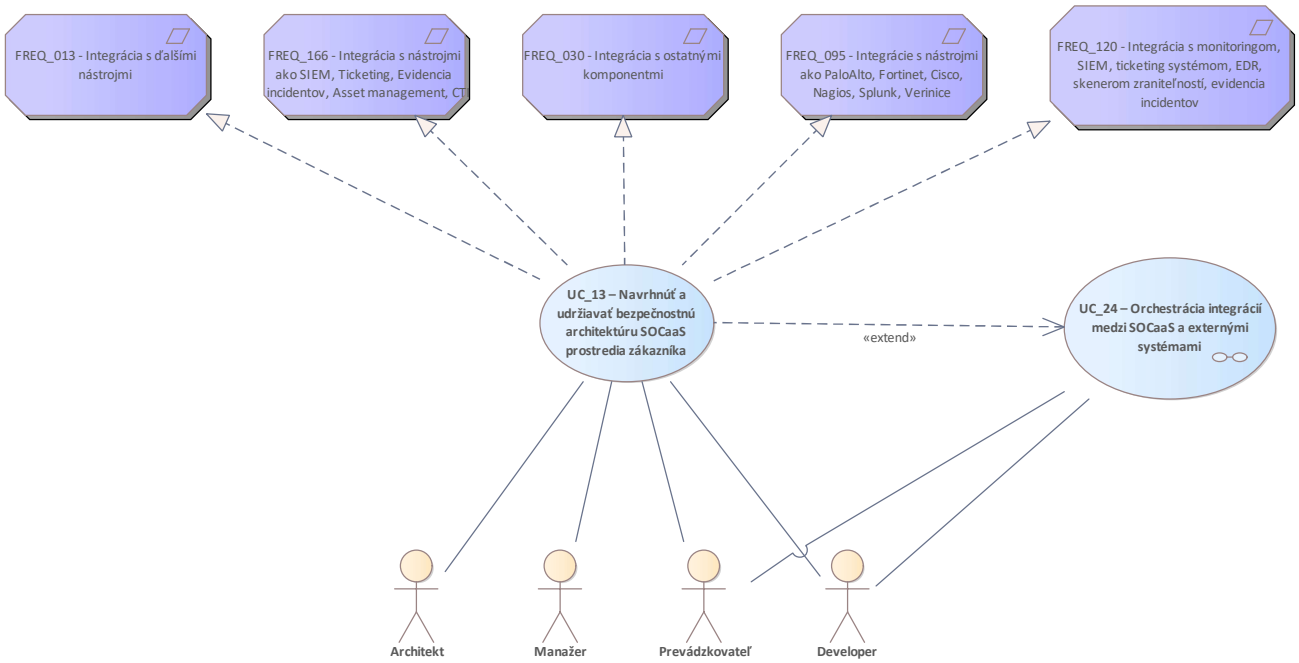


Figure 18 : UC_13 – Design and maintain the security architecture of the customer's SOCaaS environment

Use case	UC_13 – Design and maintain the security architecture of the customer's SOCaaS environment
Objective	UC_13 addresses the design and ongoing maintenance of the technical and security architecture for a specific customer: how the customer is connected to SOCaaS, which systems, zones and logging sources are integrated, through which channels the data flow (agent, syslog, API, VPN, peering…), how SIEM, SOAR, TI, ticketing, EDR, vuln scanners and asset mgmt are interconnected. The output is an approved and maintained architectural design (topology, integration points, data flows) and a list of mandatory log sources and log types, which serves as the basis for onboarding (UC_00), monitoring (UC_01), asset management (UC_11), vulnerability management (UC_12) and automation (UC_07).

Precondition	Cooperation has been agreed with the customer and the basic requirements are known (scope, regulations, criticality of services).
Precondition	The customer has provided the necessary overview of their environment (at least a high-level architecture, types of systems, networks).
Precondition	The SOCaaS platform has integration options available (connectors, APIs, agents, network paths) that can be used.

Postcondition	A documented and approved architectural design exists for connecting the customer's environment to SOCaaS (topology, integration points, data flow).
Postcondition	Integration channels are defined (which system → by what means → into which SOCaaS module).
Postcondition	Changes to the architecture are traceable (versions, who approved them, when) and usable as input for onboarding and operations.
Postcondition	A list of mandatory log sources and log types for onboarding the customer to SOCaaS is defined and documented (including the minimum fields required for threat detection).

Basic Path	Basic Path

Step	**Activity**
1	Collection of requirements and inputs
2	Analysis of the customer's existing architecture
3	Design of data flows and integration patterns, including identification of mandatory log sources and log types
4	Design of integrations with the customer's tools
5	Assessment of security principles and risks
6	Approval of the architectural design
7	Submission of the design for implementation
8	Ongoing maintenance of the architecture

ALT_13_01 – Constraints on the customer's side (network, security policies)	Alternate

Step	**Activity**
1	During the design, it is found that the customer has constraints (e.g. new ports cannot be opened, an agent or VPN cannot be used); the Architect adjusts the design to use alternative integration patterns or a narrowed scope of monitoring and resubmits the changes for approval.

ALT_13_02 – Changes in the customer's environment after deployment	Alternate

Step	**Activity**
1	Over time, the customer adds new systems, migrates to the cloud or changes the topology; the Architect creates a revision of the architectural design, adds new integration points and the Service operator plans and carries out the necessary changes to the SOCaaS configuration.

ALT_13_03 – The design does not pass the security or performance assessment	Alternate

Step	**Activity**
1	During the assessment of the design, it is found that the proposed solution is not satisfactory (excessive latencies, unacceptable risks, insufficient scaling); the design is returned to the Architect for rework and is resubmitted for approval only after the comments have been incorporated.

Linked functional requirements	
FREQ_120 - Integration with monitoring, SIEM, the ticketing system, EDR, the vulnerability scanner, incident recording	The possibility of interconnection with other SOC tools for automation and data sharing. Integration with monitoring, SIEM, ticketing system, EDR, vulnerability scanner, incident records
UC_24 – Orchestration of integrations between SOCaaS and external systems	UC_24 addresses the technical and process orchestration of integrations between SOCaaS and other systems: SIEM, SOAR, CTI, ticketing, communication tools, firewall, EDR, API integrations, webhooks, plug-ins and connectors,

	unified management of data flows and actions across these systems. The aim is for incident response, automation (UC_07) and incident management to function as a single interconnected ecosystem rather than as isolated tools.
FREQ_013 - Integration with other tools	The solution must ensure integration with other tools
FREQ_030 - Integration with other components	The solution must ensure integration with other components. Connection to data sources Integration with incident management and SOAR tools Knowledge base and playbooks
FREQ_166 - Integration with tools such as SIEM, Ticketing, Incident recording, Asset management, CTI	The possibility of collecting data from other systems and creating linked reports. Integration with tools such as SIEM, Ticketing, Incident records, Asset management, CTI, ...
FREQ_095 - Integrations with tools such as PaloAlto, Fortinet, Cisco, Nagios, Splunk, Verinice	The possibility of interconnection with other security and IT systems. Integrations with tools such as PaloAlto, Fortinet, Cisco, Nagios, Splunk, Verinice
UC_13 – Design and maintain the security architecture of the customer's SOCaaS environment	UC_13 addresses the design and ongoing maintenance of the technical and security architecture for a specific customer: how the customer is connected to SOCaaS, which systems, zones and logging sources are integrated, through which channels the data flow (agent, syslog, API, VPN, peering…), how SIEM, SOAR, TI, ticketing, EDR, vuln scanners and asset mgmt are interconnected. The output is an approved and maintained architectural design (topology, integration points, data flows) and a list of mandatory log sources and log types, which serves as the basis for onboarding (UC_00), monitoring (UC_01), asset management (UC_11), vulnerability management (UC_12) and automation (UC_07).

4.5.2.16 UC_14 – Work with the customer portal and collaborate

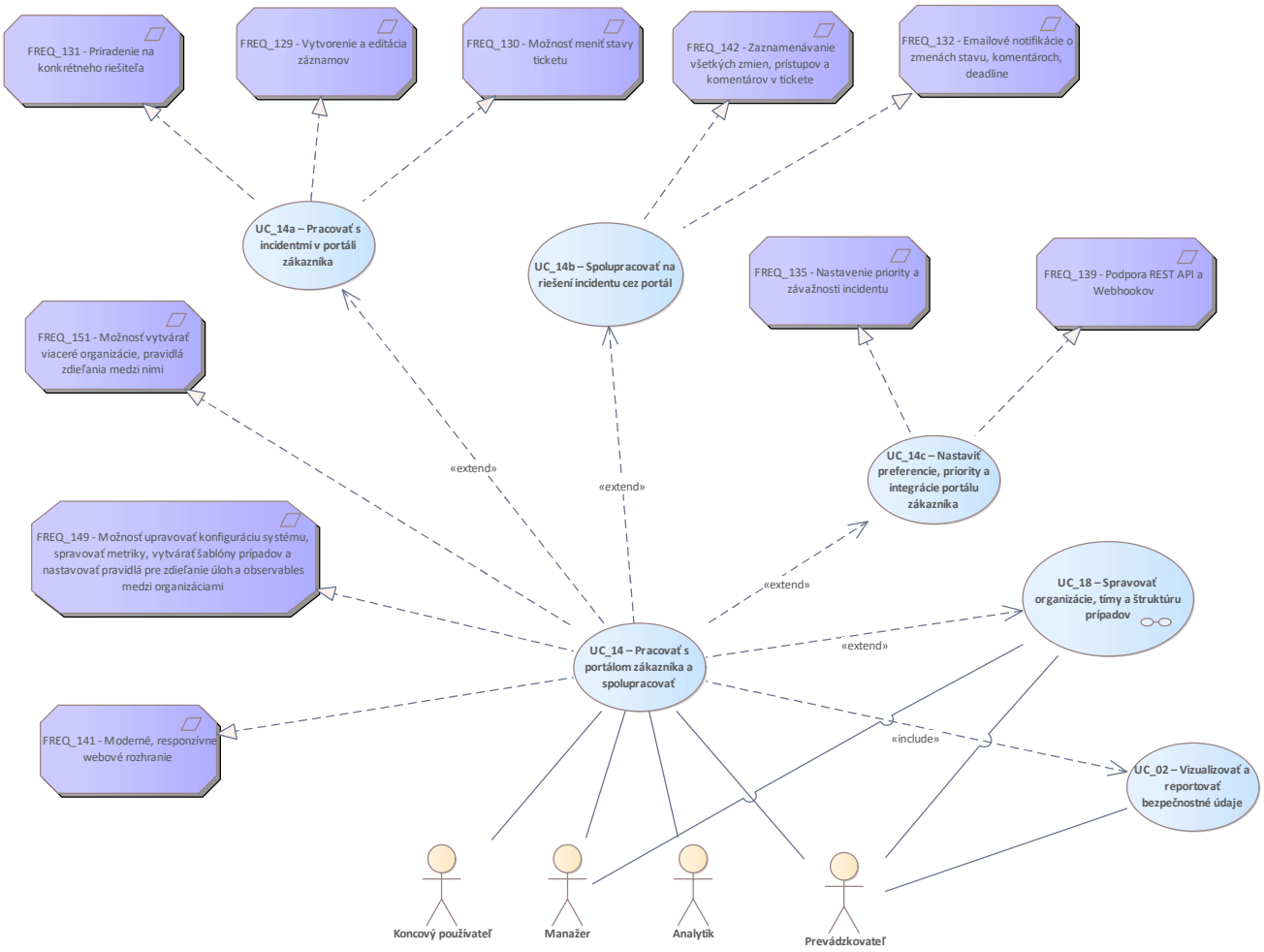


Figure 19 : UC_14 – Work with the customer portal and collaborate

Use case	UC_14 – Work with the customer portal and collaborate
Objective	UC_14 addresses the customer's view of SOCaaS – via a web portal, where the customer: sees their incidents, KPIs, reports and assets, can collaborate with the SOC (comments, attachments, approval of measures), can customize the portal for multiple organizations/units. The portal is a modern, responsive, multi-tenant web interface and also supports more advanced settings (templates, sharing, integration hooks in the sub-UCs).

Precondition	The customer has been onboarded (UC_00) and has at least one organization / tenant created.
Precondition	Accounts have been created for users and roles have been assigned (e.g. “Security manager”).
Precondition	The portal is accessible via a secured web interface (HTTPS, SSO/Federation if needed).
Postcondition	Through the portal, the user has an overview of their incidents, KPIs and assets.
Postcondition	It is possible to collaborate with the SOC directly in the context of incidents (comments, attachments, approval).

Postcondition	The portal respects the multi-tenant model – data are separated between organizations and the settings are tailored to the needs of the specific organization.

Basic Path	Basic Path

Step	**Activity**
1	The end user logs in to the customer portal (SSO / name + password / MFA according to the policy).
2	The system displays the home overview – a dashboard with the status of incidents, basic KPIs and notifications
3	The user selects the organization/tenant on behalf of which they want to work (if they have more than one).
4	In the menu, the user chooses whether they want to handle incidents, view reports or manage settings and notifications.In the menu, the user chooses whether they want to handle incidents, view reports or manage settings and notifications.
5	When “Incidents” is selected, a list of incidents/tickets for the given organization is displayed; the user can click through to the detail
6	When “Reports” is selected, the user can view or download the management reports that are available for their organization
7	When “Settings” is selected, the user can (depending on their role) change notifications, preferences and the organizational settings of the portal
8	The system records all interactions (access to incidents, comments, changes to settings) for audit purposes and ensures that they are visible only within the given organization.

ALT_14_01 – The user has no organization assigned	Alternate

Step	**Activity**
1	After login, the system detects that the account is not linked to any organization; it displays information about the missing assignment and recommends contacting the Service operator or the Manager.

ALT_14_02 – Restricted view according to role	Alternate

Step	**Activity**
1	The user has only a role with restricted rights (e.g. “read-only”); the portal hides some action elements (editing, approval) and allows only viewing of incidents and reports.

Linked functional requirements	
UC_14a – Work with incidents in the customer portal	UC_14a extends UC_14 with specific work with incidents/tickets from the customer's perspective: viewing the list of incidents, opening the detail, creating a new “ticket” or adding information, changing statuses (New, In progress, Closed, Postponed), routing the incident to the correct resolver (L1/L2/L3, a specific team).
FREQ_149 - Option to modify the system configuration, manage metrics, create case templates and set rules for sharing tasks and observables between organizations	Option to customize the platform according to the needs of the organization. Option to modify the system configuration, manage metrics, create case templates and set rules for sharing tasks and observables between organizations?
UC_14c – Set preferences, priorities and integrations of the customer portal	UC_14c extends UC_14 with a configuration part for the customer: setting the priorities and severity of incidents from the customer's perspective, setting up integration channels (REST API / Webhooks) for connecting the portal to the customer's internal tools (ticketing, chat, ITSM).

FREQ_151 - Option to create multiple organizations and sharing rules between them	The ability to operate the system for multiple entities simultaneously. Possibility of creating multiple organizations and sharing rules between them
UC_02 – Visualize and report security data	The system enables SOC operators and the administrator to display security events using dashboards, filters, charts and tabular outputs. It also enables the generation of reports and the export of data for management, audit or customers.
UC_18 – Manage organizations, teams and case structure	UC_18 addresses the administration of organizations/tenants and the way incidents/cases are structured within them: creating and modifying organizations / teams, setting up categories, types and owners of cases, defining incident templates and the way they are shared between teams and external partners. It builds on UC_17 (users and roles) and UC_14 (customer portal) and complements them with an “organizational and process” layer.
UC_14b – Collaborate on incident resolution via the portal	UC_14b extends UC_14 with a collaboration layer for incident resolution: comments between the customer and the SOC, e-mail notifications about status changes, comments and deadlines, an audit trail of all changes and accesses in the ticket.
FREQ_141 - Modern, responsive web interface	Modern and responsive design for convenient use. Modern, responsive web interface

Use case	UC_14a – Work with incidents in the customer portal
Objective	UC_14a extends UC_14 with specific work with incidents/tickets from the customer's perspective: viewing the list of incidents, opening the detail, creating a new “ticket” or adding information, changing statuses (New, In progress, Closed, Postponed), routing the incident to the correct resolver (L1/L2/L3, a specific team).

Precondition	The precondition from UC_14 applies – the user is logged in and has an organization.
Precondition	The user has a role that allows working with incidents (not just read-only).
Postcondition	Incidents/tickets are created or updated according to the customer's actions.
Postcondition	The incident has its current status set and a resolver assigned.

Basic Path	Basic Path

Step	**Activity**
1	In the portal, the user opens the “Incidents / Tickets” section.
2	The system displays a list of incidents with a filter by status, type, priority or period.
3	The user selects a specific incident and opens its detail.

4	In the incident detail, the user can add information (description, impacts, contact person) or change the incident status (e.g. from “New” to “In progress”).
5	The user can assign the incident to a specific resolver or team (e.g. “SOC L1”, “SOC L2” , ...).
6	The system saves the changes, updates the view in the portal and makes them available to SOC analysts.

ALT_14a_01 – The user wants to create a new incident/ticket	Alternate

Step	**Activity**
1	The user clicks “Create incident/ticket”, fills in the form (description, criticality, affected systems) and submits it; the system creates a new record and assigns it to the SOC queue.

ALT_14a_02 – Status change is not permitted	Alternate

Step	**Activity**
1	The user attempts to change the status of an incident for which they do not have permission (e.g. to close an incident that only the SOC can close); the system rejects the change and displays information about insufficient rights.

Linked functional requirements	
FREQ_130 - Ability to change ticket statuses	The ability to switch between different ticket statuses during its life cycle. Possibility of changing ticket statuses (e.g. New, In progress, Closed, Postponed)
FREQ_131 - Assignment to a specific resolver	Assignment of tickets to specific resolvers, including escalation by level. Assignment to a specific resolver (L1 analyst, escalation to L2/L3)
FREQ_129 - Creation and editing of records	Basic functionality of a ticketing system – creating, editing and deleting tickets. Creation and editing of records (tickets)
UC_14a – Work with incidents in the customer portal	UC_14a extends UC_14 with specific work with incidents/tickets from the customer's perspective: viewing the list of incidents, opening the detail, creating a new “ticket” or adding information, changing statuses (New, In progress, Closed, Postponed), routing the incident to the correct resolver (L1/L2/L3, a specific team).

Use case	UC_14b – Collaborate on incident resolution via the portal
Objective	UC_14b extends UC_14 with a collaboration layer for incident resolution: comments between the customer and the SOC, e-mail notifications about status changes, comments and deadlines, an audit trail of all changes and accesses in the ticket.

Precondition	The incident exists and is available in the portal (UC_14a).
Precondition	The user has permission to view and comment on the given incident.
Postcondition	Comments and/or any status changes initiated by the customer are added to the incident.
Postcondition	All changes and accesses are recorded for audit purposes; the relevant parties receive notifications.

Basic Path	Basic Path

Step	Activity
1	The user opens the incident detail in the portal.The user opens the incident detail in the portal.
2	The system saves the comment and records who added it and when.
3	The system saves the comment and records who added it and when.
4	The system sends e-mail notifications to the relevant parties (Analyst, other customer representatives) about the new comment or status change.
5	The Analyst replies via their tool; the reply is synchronized to the portal and displayed to the customer as another comment.
6	All comments, status changes and accesses can be viewed retrospectively in the incident history.

ALT_14b_01 – Notifications are disabled for a specific user	Alternate

Linked functional requirements	
FREQ_132 - E-mail notifications about status changes, comments, deadlines	Automatic e-mail notifications of status changes, comments, deadlines. E-mail notifications of status changes, comments, deadlines
FREQ_142 - Recording of all changes, accesses and comments in the ticket	Recording of all actions, changes and comments in a ticket. Recording of all changes, accesses and comments in a ticket
UC_14b – Collaborate on incident resolution via the portal	UC_14b extends UC_14 with a collaboration layer for incident resolution: comments between the customer and the SOC, e-mail notifications about status changes, comments and deadlines, an audit trail of all changes and accesses in the ticket.

Use case	UC_14c – Set preferences, priorities and integrations of the customer portal
Objective	UC_14c extends UC_14 with a configuration part for the customer: setting the priorities and severity of incidents from the customer's perspective, setting up integration channels (REST API / Webhooks) for connecting the portal to the customer's internal tools (ticketing, chat, ITSM).

Precondition	The user has a role with the right to change portal settings for their organization.
Precondition	At least basic incident priorities and categories are defined for the organization.
Postcondition	Preferred incident priorities/severities are defined for the given organization.
Postcondition	Integration channels are configured (if necessary) – e.g. Webhooks for notifications to the internal ticketing system.

Basic Path	Basic Path

Step	Activity
1	The Manager opens the “Organization settings” section in the portal.
2	In the “Incidents” section, they set the priority/severity mapping (e.g. how SOC priorities are to be reflected in the customer's internal priorities).
3	In the “Integrations” section, they enter the endpoint URLs for Webhooks or API tokens for connecting to the internal ticketing/ITSM system.
4	The system verifies basic connectivity and saves the settings.

5	On subsequent incident status changes or for new incidents, the system sends notifications via REST API / Webhooks to the customer's systems according to the settings.
ALT_14c_01 – Invalid or unavailable integration interface	Alternate
Step	**Activity**
1	When the integration settings are saved, the test request fails (timeout, 4xx, 5xx); the system alerts the Manager to this and either saves the integration as inactive or requires the parameters to be corrected.
Linked functional requirements	
FREQ_135 - Setting incident priority and severity	Setting the priority and severity level of a ticket for better planning. Setting the priority and severity of an incident
FREQ_139 - Support for REST API and Webhooks	Support for REST API interfaces and Webhooks for communication with other systems. Support for REST API and Webhooks
UC_14c – Set preferences, priorities and integrations of the customer portal	UC_14c extends UC_14 with a configuration part for the customer: setting the priorities and severity of incidents from the customer's perspective, setting up integration channels (REST API / Webhooks) for connecting the portal to the customer's internal tools (ticketing, chat, ITSM).

4.5.2.17 UC_15 – Manage the operation of the SOCaaS platform

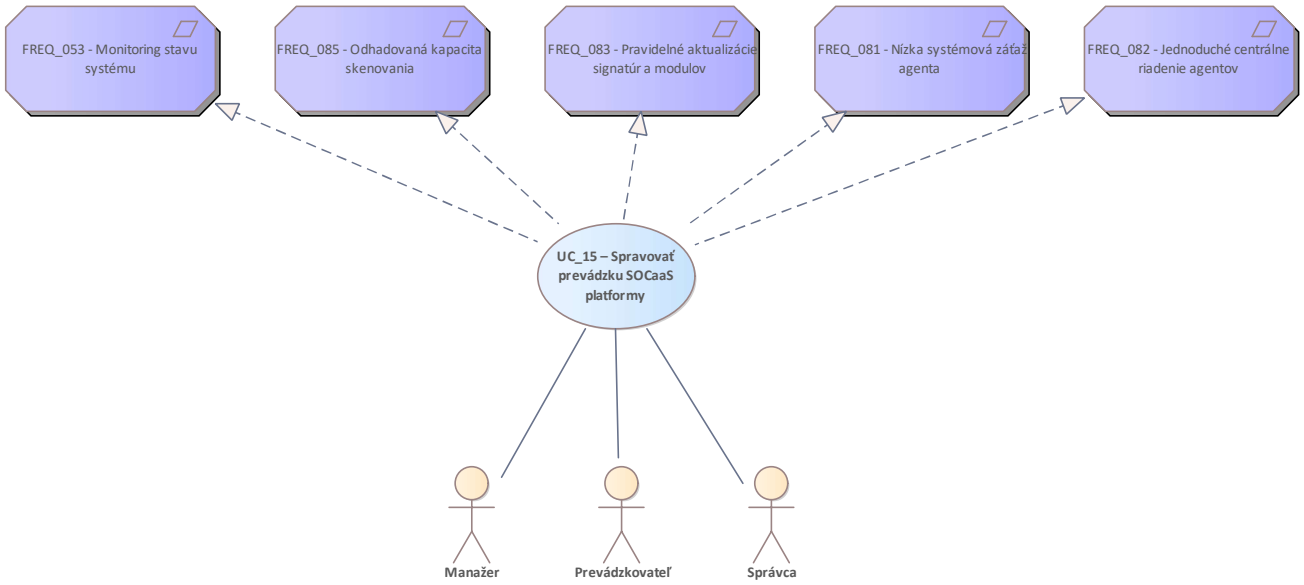


Figure 20 : UC_15 – Manage SOCaaS platform operations

Use case	UC_15 – Manage the operation of the SOCaaS platform
Objective	UC_15 addresses the technical operation of the SOCaaS platform so that: it is stable, performant and available, agents and sensors do not place unnecessary load on the customer's environment, the platform is continuously updated (signatures, modules),

	processing/scanning capacity is kept under control. The aim is for SOCaaS to run reliably, to be able to handle the expected volume of data and not to jeopardize the performance of the customer's systems.

Precondition	The SOCaaS platform is deployed in the production environment (on-prem/cloud/SaaS).
Precondition	Monitoring tools and mechanisms exist for tracking the status of the platform and the agents.
Precondition	Basic operational parameters are defined – acceptable load, processing capacity, maintenance windows.
Postcondition	The status of the platform and the deployed agents is continuously monitored and known to the Operator.
Postcondition	Updates of signatures and modules are performed in a controlled manner and without significant service outage.
Postcondition	Processing/scanning capacity is planned and the Operator is able to identify the need for expansion or optimization.

Basic Path	Basic Path

Step	**Activity**
1	The Operator monitors the status of the SOCaaS platform in the operational dashboard (component availability, queues, latencies, error rates).
2	The system continuously monitors the status of the agents deployed at customers – availability, version, CPU/RAM load.
3	When the number of customers or the volume of data grows, the Operator evaluates the processing/scanning capacity and compares it with the expected growth.
4	Based on the update policy, the Operator schedules regular updates of signatures and modules (e.g. outside peak hours) in order to minimize the impact on performance.
5	The system performs the updates and verifies whether the agents and components are functional after the update and are reporting within the expected scope.
6	If necessary, the Operator adjusts the agent configuration (estimates of scanning frequency, level of detail, load limits) so that they do not overload the customers' systems.
7	All operational interventions (updates, configuration changes, capacity interventions) are recorded for audit and diagnostic purposes.

ALT_15_01 – Overload of agents or the platform detected	Alternate

Step	**Activity**
1	Monitoring shows that the agents or the platform itself place excessive load on the system (CPU, RAM, disk, network); the Operator adjusts the configuration (scanning frequency, data scope) or proposes a capacity expansion or deployment optimization.

ALT_15_02 – Failure of signature or module update	Alternate

Step	**Activity**
1	After the update is launched, it is found that some of the agents or components did not complete the update or stopped reporting; the Operator identifies the affected systems and rolls them back to the last stable version or repeats the update with adjusted parameters.

Linked functional requirements	
FREQ_081 - Low system load of the agent	The agent has a low impact on system performance, which enables deployment at large scale. Low system load of the agent
FREQ_083 - Regular updates of signatures and modules	Regular updates of threat databases, signatures and detection rules. Regular updates of signatures and modules
FREQ_082 - Simple centralized management of agents	A central console for the management, configuration and supervision of all agents. Simple central management of agents
FREQ_053 - System status monitoring	Monitoring of system performance, availability and functionality. System health monitoring (healthcheck)

FREQ_085 - Estimated scanning capacity	The ability of the tool to scan large networks efficiently. Estimated scanning capacity (for example, min. 100 IP/24h)

4.5.2.18 UC_16 – Customize reports, exports and documents to the organization's needs

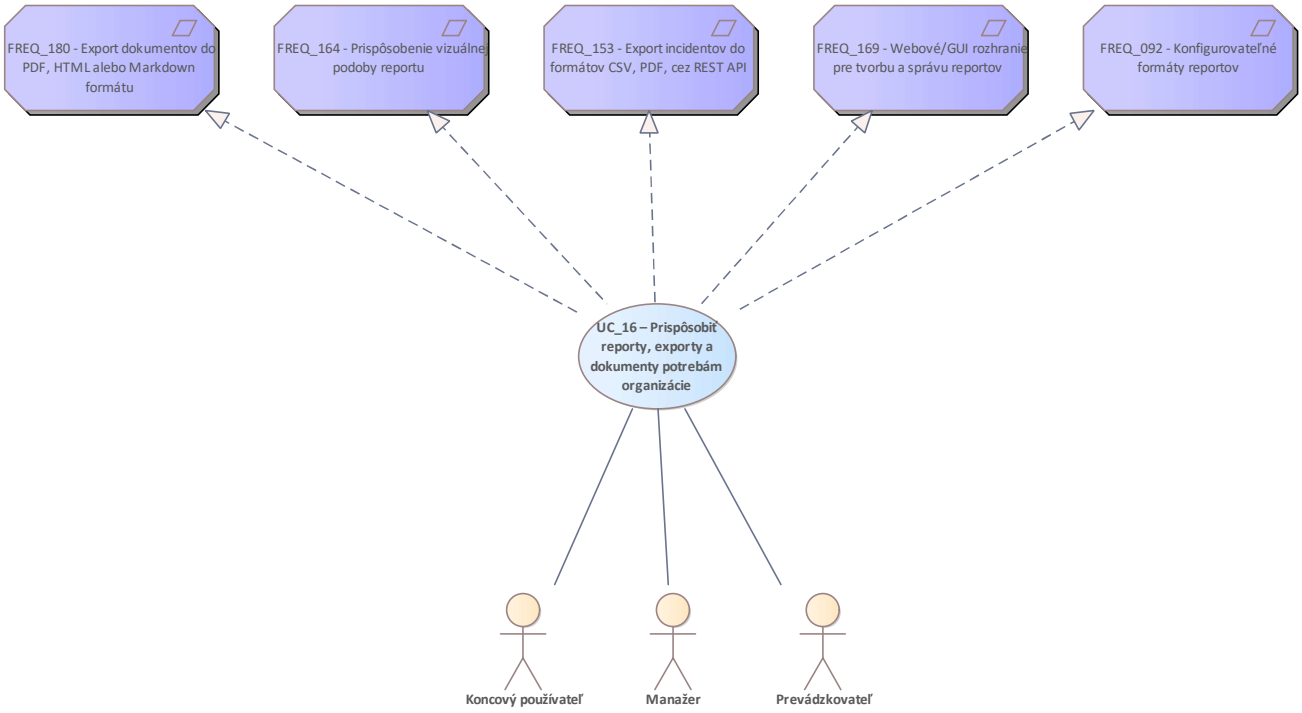


Figure 21 : UC_16 – Customize reports, exports and documents to the organization's needs

Use case	UC_16 – Customize reports, exports and documents to the organization's needs
Objective	UC_16 addresses the configuration of the form of SOCaaS outputs: the appearance of reports according to the corporate identity (logo, colours, header), export formats (CSV, XML, PDF, HTML, Markdown), export of incidents and documents for further processing or archiving. It builds on UC_10b (which addresses the content and scheduling of reports) by enabling detailed settings of presentation and export so that the outputs fit into the customer's environment and tools.

Precondition	There are defined reports and documents that can be generated (UC_10b, UC_02b).
Precondition	The user has a role that allows them to change the appearance and export settings of reports/documents.
Precondition	The SOCaaS platform supports multiple export formats and basic appearance templates.
Postcondition	Report appearance templates are defined for individual organizations (logo, colours, header).
Postcondition	Preferred export formats are set for specific types of reports/incidents.

Postcondition	Users are able to export reports, incidents and documents in the required formats for further processing or archiving.

Basic Path	Basic Path

Step	**Activity**
1	The Manager or Auditor opens the “Report and export settings” section in the portal.
2	They select the type of output they want to customize (e.g. “management report”, “incident overview”, “knowledge document”).
3	In the Appearance section, they select or modify the template:
4	In the Export formats section, they select the formats in which the report/document is to be available (e.g. CSV and PDF; for documents also HTML/Markdown).
5	The user displays a preview of the report with the new appearance and format and checks the layout and readability.
6	After confirmation, the system saves the appearance template and preferred export formats for the given type of report/document.
7	The next time a report is generated or an incident is displayed, the system allows one of the configured exports to be selected directly (e.g. “Download as PDF”, “Export to CSV”, “Export document to HTML”).
8	Exported files can be downloaded by the user or sent to other systems (e.g. via REST API), according to the organization's settings.

ALT_16_01 – The required export format is not supported	Alternate

Step	**Activity**
1	The user attempts to set a format that the platform does not support; the system alerts them to this and offers available alternatives (e.g. CSV instead of XLSX, PDF instead of DOCX).

ALT_16_02 – Error when exporting a large volume of data	Alternate

Step	**Activity**
1	When exporting a report with a large number of records, a limit or timeout is exceeded; the system aborts the export, informs the user and offers to narrow the scope (shorter period, filtered data) or an asynchronous export.

ALT_16_03 – Change of corporate identity	Alternate

Step	**Activity**
1	The organization changes its logo or corporate colours; the Manager opens the appearance settings, modifies the template (logo, colours) and, after saving, all newly generated reports will use the new identity, while older reports remain in their original form.

Linked functional requirements	
FREQ_180 - Export of documents to PDF, HTML or Markdown format	The possibility of exporting documents for offline use or publication. Export of documents to PDF, HTML or Markdown format
FREQ_092 - Configurable report formats	Flexibility of outputs for further processing. Configurable report formats (e.g. CSV, XML, PDF)
FREQ_169 - Web/GUI interface for creating and managing reports	A user-friendly interface for designing, editing and managing reports. Web/GUI interface for creating and managing reports
FREQ_153 - Export of incidents to CSV, PDF formats, via REST API	Data export for documentation or reporting. Export of incidents to CSV, PDF formats, via REST API
FREQ_164 - Customization of the visual form of the report	The possibility of adjusting the appearance of the report according to the corporate identity. Customization of the visual appearance of the report (logo, colours, header)

4.5.2.19 UC_17 – Manage users and roles in the SOCaaS platform

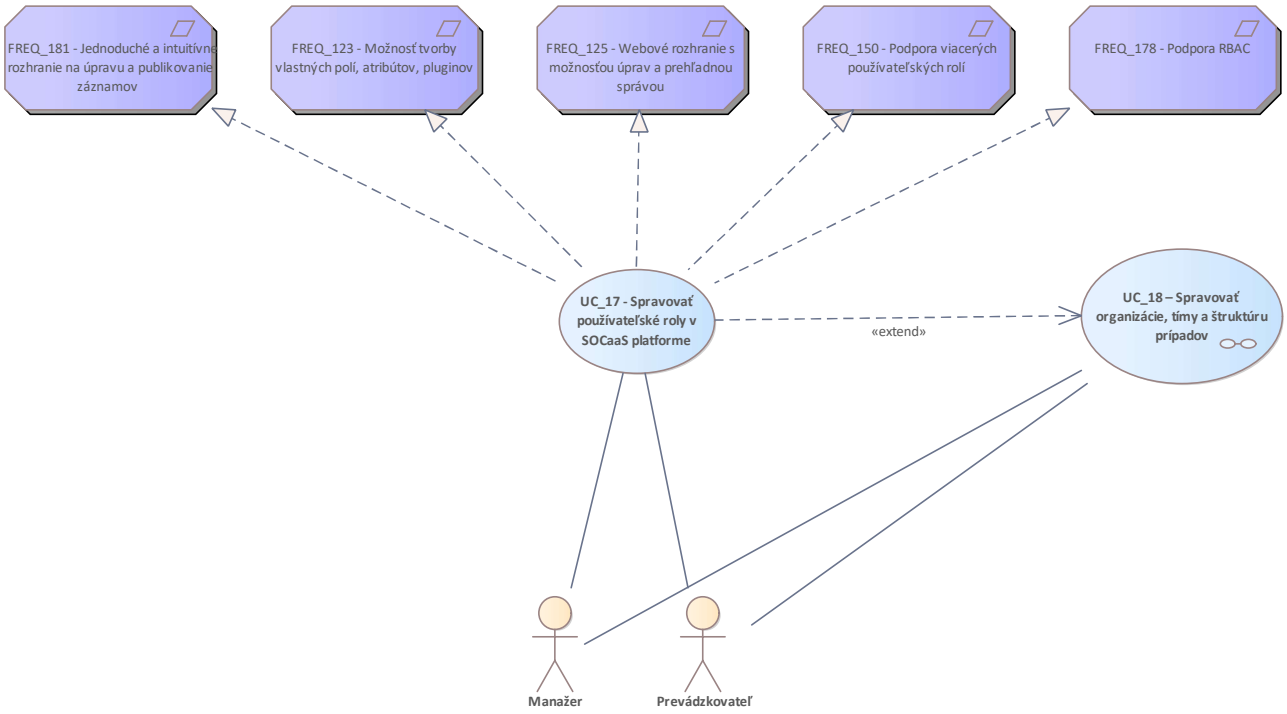


Figure 22 : UC_17 – Manage users and roles in the SOCaaS platform

Use case	UC_17 - Manage user roles in the SOCaaS platform
Objective	UC_17 addresses the administration of users and roles within SOCaaS: creating and modifying accounts, assigning roles (SOC, customer, auditor…), setting up access via RBAC, adding custom attributes where applicable. It is a system/administrative UC that supports all other UCs by determining who can see and do what in the platform.

Precondition	The SOCaaS platform is running and has an accessible administrative web interface.
Precondition	The currently logged-in user has administrative permissions to manage accounts and roles.
Postcondition	Created, modified or deactivated user accounts are stored in the system.
Postcondition	Users are assigned appropriate roles according to the RBAC model.
Postcondition	Any custom attributes/metadata are set and usable in other parts of the system.

Basic Path	Basic Path

Step	**Activity**
1	The Operator logs in to the administrative web interface.

2	They open the “Users and roles” section.
3	They choose whether they want to create a new user or modify an existing account.
4	When creating an account, they enter basic data (name, contacts, account identifier) and select one or more user roles (e.g. SOC analyst, Manager, Customer, Auditor).
5	As needed, they add custom fields/attributes (e.g. organizational affiliation, permission level, special tags).
6	The system saves the changes and displays the user in a clear list; the interface allows filtering and searching by name, role and other attributes.
7	The Operator can deactivate the account or change roles if the user's responsibilities change.

ALT_17_01 – Attempt to make a change for which the user does not have permission	Alternate

Step	**Activity**
1	A user without administrative rights attempts to modify accounts or roles; the system rejects the change and displays information that they do not have sufficient permission.

ALT_17_02 – Invalid combination of roles	Alternate

Step	**Activity**
1	When saving the account, the system detects that the selected combination of roles is in conflict with the RBAC policy (e.g. a conflict of interest); the administrator receives information and must adjust the selected roles.

Linked functional requirements	
FREQ_092 - Configurable report formats	Flexibility of outputs for further processing. Configurable report formats (e.g. CSV, XML, PDF)
FREQ_125 - Web interface with editing capabilities and clear management	User-friendly GUI for asset management. Web interface with editing capability and clear management
FREQ_150 - Support for multiple user roles	RBAC support for various roles. Support for multiple user roles – at least analyst and administrator
FREQ_153 - Export of incidents to CSV, PDF formats, via REST API	Data export for documentation or reporting. Export of incidents to CSV, PDF formats, via REST API
FREQ_123 - Ability to create custom fields, attributes, plugins	User flexibility to customize the system to their own needs. Possibility of creating custom fields, attributes, plugins (e.g. NetBox/Django)
FREQ_169 - Web/GUI interface for creating and managing reports	A user-friendly interface for designing, editing and managing reports. Web/GUI interface for creating and managing reports
FREQ_178 - RBAC support	Access control according to user roles. RBAC support – different access levels for reading, editing, approval
FREQ_164 - Customization of the visual form of the report	The possibility of adjusting the appearance of the report according to the corporate identity. Customization of the visual appearance of the report (logo, colours, header)
FREQ_180 - Export of documents to PDF, HTML or Markdown format	The possibility of exporting documents for offline use or publication. Export of documents to PDF, HTML or Markdown format
FREQ_181 - Simple and intuitive interface for editing and publishing records	User-friendly GUI for knowledge management. Simple and intuitive interface for editing and publishing records
UC_18 – Manage organizations, teams and case structure	UC_18 addresses the administration of organizations/tenants and the way incidents/cases are structured within them: creating and modifying organizations / teams, setting up categories, types and owners of cases, defining incident templates and the way they are shared between teams and external partners. It builds on UC_17 (users and roles) and UC_14 (customer portal) and complements them with an “organizational and process” layer.

4.5.2.20 UC_18 – Manage organizations, teams and case structure

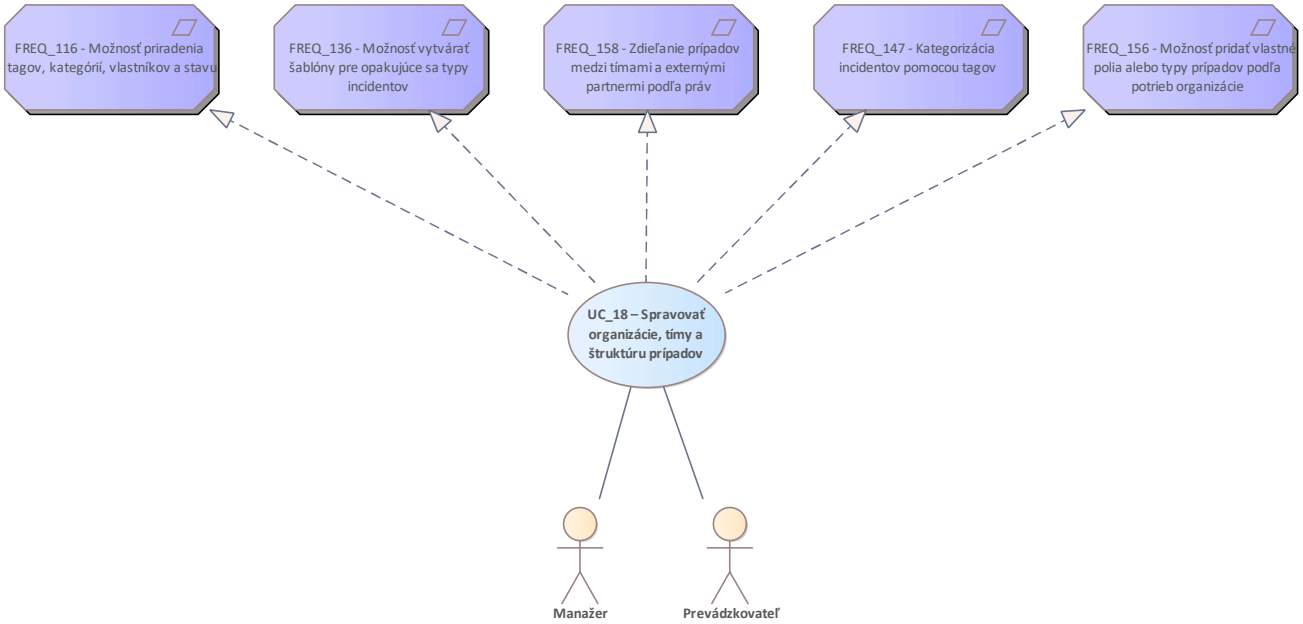


Figure 23 : UC_18 – Manage organizations, teams and case structure

Use case	UC_18 – Manage organizations, teams and case structure
Objective	UC_18 addresses the administration of organizations/tenants and the way incidents/cases are structured within them: creating and modifying organizations / teams, setting up categories, types and owners of cases, defining incident templates and the way they are shared between teams and external partners. It builds on UC_17 (users and roles) and UC_14 (customer portal) and complements them with an “organizational and process” layer.

Precondition	The user is logged in and has permissions to manage organizations/case structure.
Precondition	SOCaaS has at least a few users created (UC_17) and at least one organization/tenant (UC_14).
Postcondition	Organizations/teams have a case structure set up (types, categories, owners).
Postcondition	Templates for recurring incidents and the way they are shared between teams/partners are defined.
Postcondition	Incidents can be unambiguously assigned to an organization, team, category and owner.

Basic Path		Basic Path
Step	**Activity**	

1	The Manager or Operator opens the “Organizations and cases” section in the administration.
2	They select an existing organization (tenant) or create a new one and assign teams to it (e.g. “SOC L1”, “SOC L2”, “Customer IT”).
3	For the organization, they set the case/incident types (e.g. “Phishing”, “Malware”, “Policy”, “Vuln finding”) and basic categories.
4	For individual incident types, they create templates – prefilled fields, recommended items, required information.
5	They assign owners, tags and categories (e.g. “critical”, “regulatory”, “internal”) to incident types and templates so that they can be filtered and reported on.
6	They set which teams and, where applicable, external partners can view and edit the given cases (sharing between teams according to rights).They set which teams and, where applicable, external partners can view and edit the given cases (sharing between teams according to rights).
7	The system saves the organization structure, templates and sharing rules; new incidents are created according to these settings.

ALT_18_01 – Creating a custom case type for a specific organization	Alternate

Step	**Activity**
1	The organization has a specific case type (e.g. “Compliance finding”) that does not exist in the global schema; the Manager adds a new case type for this organization only and creates a custom template for it.

ALT_18_02 – Restricting case sharing with an external partner	Alternate

Step	**Activity**
1	Originally, cases are shared with an external supplier; the Manager decides that certain categories (e.g. HR incidents) will not be shared and adjusts the sharing rules so that these cases are accessible only to internal teams.

Linked functional requirements	
FREQ_156 - Ability to add custom fields or case types according to the organization's needs	Possibility of extending the incident schema. Possibility of adding custom fields or case types according to the organization's needs
FREQ_116 - Ability to assign tags, categories, owners and status	Adding metadata for efficient sorting and management of assets. Possibility of assigning tags, categories, owners and status (e.g. active, under maintenance, decommissioned)
FREQ_147 - Categorization of incidents using tags	The possibility of categorizing incidents using labels/tags. Categorization of incidents using tags
FREQ_136 - Ability to create templates for recurring incident types	Creation of templates for quick entry of frequently recurring incidents. Possibility of creating templates for recurring incident types
FREQ_158 - Sharing of cases between teams and external partners according to rights	Support for controlled sharing of cases. Sharing of cases between teams and external partners according to permissions
UC_18 – Manage organizations, teams and case structure	UC_18 addresses the administration of organizations/tenants and the way incidents/cases are structured within them: creating and modifying organizations / teams, setting up categories, types and owners of cases, defining incident templates and the way they are shared between teams and external partners. It builds on UC_17 (users and roles) and UC_14 (customer portal) and complements them with an “organizational and process” layer.
UC_18 – Manage organizations, teams and case structure	UC_18 addresses the administration of organizations/tenants and the way incidents/cases are structured within them: creating and modifying organizations / teams,

	setting up categories, types and owners of cases, defining incident templates and the way they are shared between teams and external partners. It builds on UC_17 (users and roles) and UC_14 (customer portal) and complements them with an “organizational and process” layer.

4.5.2.21 UC_19 – Manage retention, protection and auditing of SOCaaS data

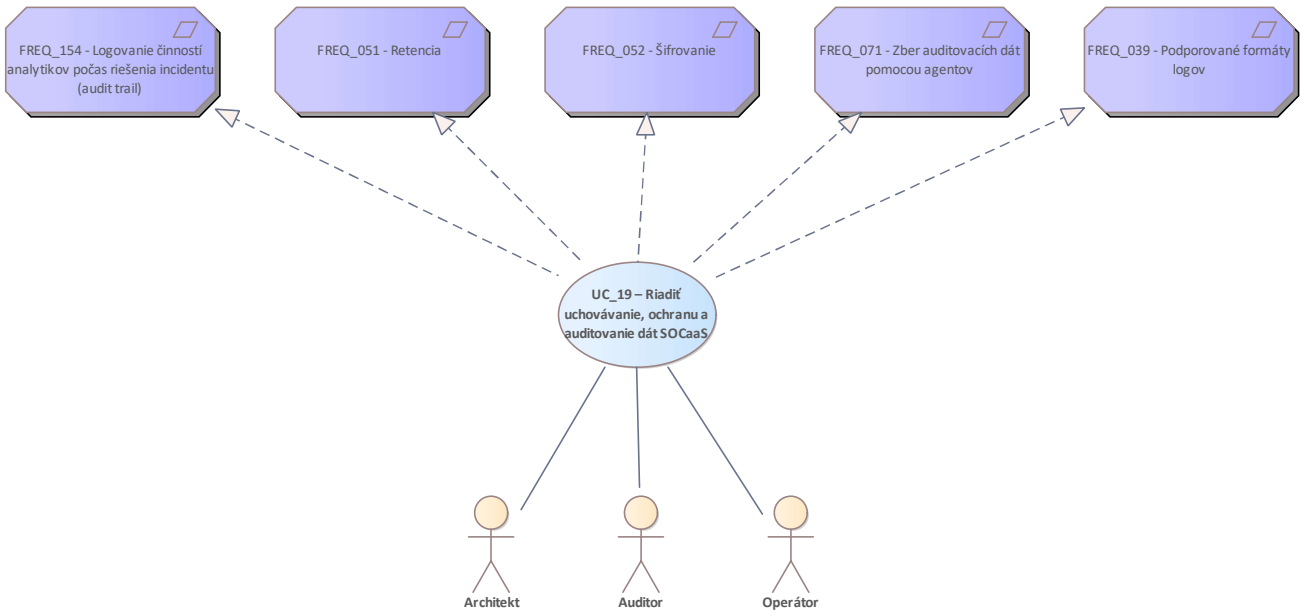


Figure 24 : UC_19 – Manage SOCaaS data retention, protection and auditing

Use case	UC_19 – Manage retention, protection and auditing of SOCaaS data
Objective	UC_19 addresses data management in SOCaaS from the platform perspective: where and how logs are stored, indexed and retained, which retention policies apply to individual log sources, how logs are protected and encrypted, how audit records of user and analyst activities are kept. The aim is for SOCaaS to meet the requirements for secure storage and auditability of data while also supporting forensic analysis.

Precondition	The SOCaaS platform is in operation (UC_15) and receives logs from various sources (UC_01).
Precondition	There is a log/data storage (SIEM / data lake) that supports the configuration of retention and encryption.
Precondition	At least a basic log retention and security policy is defined (e.g. according to log type or regulation).
Postcondition	Retention policies are set for logs/data according to source and importance.

Postcondition	Logs are protected both in transit and at rest (encryption configured).
Postcondition	User and analyst activities are audited so that forensic analysis can be performed and it can be traced who did what.

Basic Path	Basic Path

Step	**Activity**
1	Design of the storage and retention architecture
2	Setting retention policies according to source and importance
3	Enabling and verifying log encryption
4	Configuring the collection of audit records
5	Monitoring the status of retention and the audit trail
6	Evaluating auditors' requirements
7	Adjusting policies based on findings

ALT_19_01 – Insufficient capacity for the configured retention	Alternate

Step	**Activity**
1	With longer retention periods, the Operator finds that the storage does not have sufficient capacity; together with the Architect, they either adjust the retention (e.g. differently according to log type) or propose an expansion of the storage.

ALT_19_02 – Auditor's requirement for longer retention	Alternate

Step	**Activity**
1	An auditor or regulator requires longer retention for a specific type of logs (e.g. login logs); the Architect and the Operator adjust the relevant retention policy and record the change as part of the compliance documentation.

ALT_19_03 – Incident involving compromise of the log storage	Alternate

Step	**Activity**
1	An incident related to access to the log storage is detected; the audit trail makes it possible to determine who accessed the data and when. Based on the findings, the Architect and the Operator tighten encryption, access rights or segmentation of the storage.

Linked functional requirements	
FREQ_154 - Logging of analyst activities during incident handling (audit trail)	Recording of activities for forensic analysis. Logging of analysts' activities during incident handling (audit trail)
FREQ_039 - Supported log formats	The ability of the SIEM to receive and process logs in various standardized formats. Supported log formats (syslog, JSON, CSV, CEF, LEEF...)
FREQ_052 - Encryption	Protection of logs against unauthorized access in transit and at rest. Encryption (both in transit and at rest)
FREQ_071 - Collection of audit data using agents	Collection of audit records on user activities, processes and the system via the agent. Collection of audit data using agents
FREQ_051 - Retention	The ability to set log retention policies according to source and importance. Retention (flexible, by log type)

4.5.2.22 UC_20 – Manage the knowledge base and playbook library

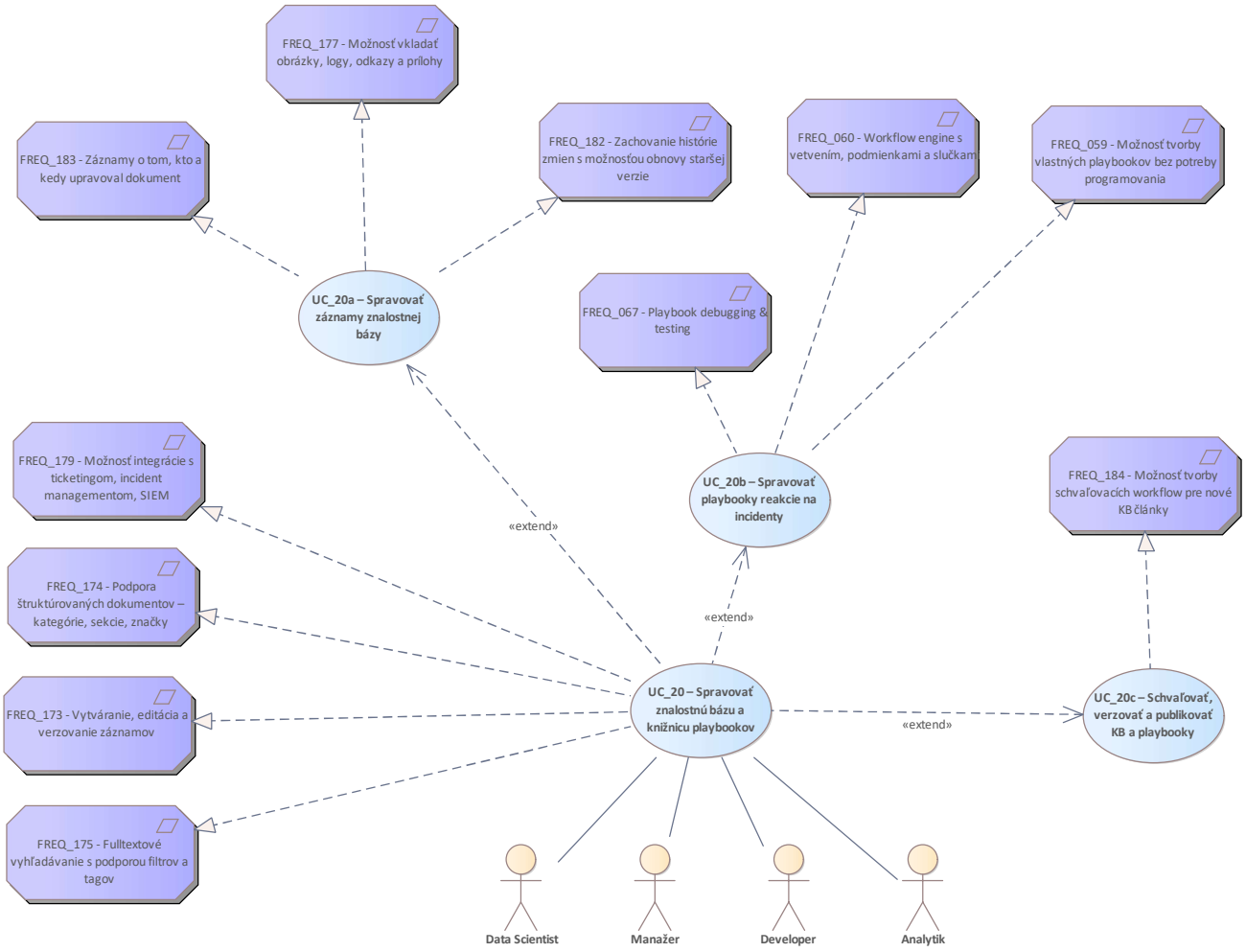


Figure 25 : UC_20 – Manage the knowledge base and playbook library

Use case	UC_20 – Manage the knowledge base and playbook library
Objective	UC_20 ensures that the SOC has, in a centralized and managed manner: a knowledge base (KB) – articles, guides, lessons learned, procedures, a playbook library – both manual and automated response procedures, a process for approving, versioning and publishing this content.

Precondition	There is a module for the knowledge base and playbooks available to the Analyst and the Manager.
Precondition	Roles and rights for creating, editing and approving content are defined.
Precondition	At least basic KB articles or playbooks already exist in the system (e.g. an initial set).
Postcondition	The knowledge base contains up-to-date, versioned and categorized records.
Postcondition	The playbook library contains published and usable playbooks linked to incident types.
Postcondition	Each record has a traceable history and information on who created and approved it.

Basic Path		Basic Path
Step	**Activity**	
1	The Analyst opens the “Knowledge base and playbooks” module and chooses whether they want to work with the KB or with playbooks.	
2	The Analyst creates a new record or opens an existing one and adds content based on current knowledge from incidents, TI or hunting.	
3	The Analyst assigns the record to incident types, categories and tags so that it is easy to find in other UCs.	
4	The Analyst saves the record as a draft and submits it for approval to the Manager or the area owner.	
5	The Manager reviews the content, compares the changes with the previous version and decides whether to approve it or return it for revision.	
6	After approval, the system marks the new version as valid and moves the old version to the history with a record of the date and author of the change.	
7	The valid record is made available in the relevant parts of the system (e.g. in the incident detail, in the playbook engine, in the customer portal).	
8	The system records all operations on the record (creation, edits, approval) for audit and forensic analysis.	
ALT_20_01 – The record is returned for further work		Alternate
Step	**Activity**	
1	The Manager does not approve the proposed content, adds a comment with remarks and the record is returned to the Analyst in the draft state.	
ALT_20_02 – The record is marked as obsolete		Alternate
Step	**Activity**	
1	The Manager or Analyst finds that the content is no longer valid; they change the status to “archived” and the system stops offering it as an active recommendation, but it remains available in the history.	

Linked functional requirements	
UC_20a – Manage knowledge base records	UC_20a addresses in detail the life cycle of KB articles – creation, editing, structure, tagging and linking to incidents.
FREQ_179 - Ability to integrate with ticketing, incident management, SIEM	Linking the KB with other systems for more efficient use of content. Possibility of integration with ticketing, incident management, SIEM
UC_20c – Approve, version and publish KB and playbooks	UC_20c addresses the governance layer over the KB and playbooks – approval, versioning, change auditing and the approval workflow.
FREQ_140 - Export to PDF, CSV, JSON formats for documentation and reporting	The possibility of exporting tickets for archiving or reporting purposes. Export to PDF, CSV, JSON formats for documentation and reporting
FREQ_175 - Full-text search with support for filters and tags	Searching the KB with the possibility of filtering by tags, categories and content. Full-text search with support for filters and tags
FREQ_173 - Creation, editing and versioning of records	The possibility of creating, editing and tracking versions of documents with a history of changes. Creation, editing and versioning of records (guides, playbooks, FAQ)
UC_20b – Manage incident response playbooks	UC_20b addresses the design, editing and maintenance of playbooks that the SOC uses for automated or semi-automated response (UC_07).
FREQ_174 - Support for structured documents – categories, sections, tags	Organization of KB documents according to structured rules (categories, tags, sections). Support for structured documents – categories, sections, labels (tags)
UC_20 – Manage the knowledge base and playbook library	UC_20 ensures that the SOC has, in a centralized and managed manner: a knowledge base (KB) – articles, guides, lessons learned, procedures, a playbook library – both manual and automated response procedures,

	a process for approving, versioning and publishing this content.
UC_20 – Manage the knowledge base and playbook library	UC_20 ensures that the SOC has, in a centralized and managed manner: a knowledge base (KB) – articles, guides, lessons learned, procedures, a playbook library – both manual and automated response procedures, a process for approving, versioning and publishing this content.
Use case	UC_20a – Manage knowledge base records
Objective	UC_20a addresses in detail the life cycle of KB articles – creation, editing, structure, tagging and linking to incidents.

Precondition	The preconditions of UC_20 apply.
Precondition	The user has permission to edit KB records.
Postcondition	The KB contains up-to-date, structured articles with rich content, change history and an audit trail.

Basic Path	Basic Path

Step	**Activity**
1	The Analyst opens the list of KB articles in the “Knowledge base” module.
2	The Analyst creates a new KB article or opens an existing record for editing.
3	The Analyst edits the article content and adds structure (sections, subsections), tags and categories.
4	The Analyst inserts relevant attachments, logs or screenshots that help in resolving similar incidents.
5	The Analyst saves the record; the system records the author and time of the change and, as needed, submits it for approval.

ALT_20a_01 - Merging two related KB records	Alternate

Step	**Activity**
1	The Analyst identifies two very similar articles in the knowledge base, creates one consolidated record and marks the original articles as archived with a link to the new consolidated article.

ALT_20a_02 - Attempt to edit without sufficient permissions	Alternate

Step	**Activity**
1	A user with read-only access attempts to edit a KB article; the system does not allow the change, displays information about insufficient permissions and leaves the record unchanged.

Linked functional requirements	
FREQ_183 - Records of who edited the document and when	Audit trail of document changes with information on authors and time. Records of who edited a document and when
FREQ_177 - Ability to insert images, logs, links and attachments	The ability to enrich documents with multimedia or data content. Possibility of inserting images, logs, links and attachments
FREQ_182 - Retention of change history with the ability to restore an older version	Tracking and archiving of changes for each document. Retention of the change history with the possibility of restoring an older version
UC_20a – Manage knowledge base records	UC_20a addresses in detail the life cycle of KB articles – creation, editing, structure, tagging and linking to incidents.
UC_20a – Manage knowledge base records	UC_20a addresses in detail the life cycle of KB articles – creation, editing, structure, tagging and linking to incidents.
Use case	UC_20b – Manage incident response playbooks

Objective	UC_20b addresses the design, editing and maintenance of playbooks that the SOC uses for automated or semi-automated response (UC_07).

Precondition	A workflow/playbook engine is available in the system.
Precondition	The user has permission to create and modify playbooks.
Postcondition	Playbooks have defined steps, branching and integration points and are tested before deployment to production.

Basic Path	Basic Path

Step	**Activity**
1	The Analyst or Developer opens the “Playbooks” section in the SOCaaS interface.
2	The user creates a new playbook or opens an existing one for editing.
3	In the graphical editor, they define the individual steps of the playbook, their order, branching and execution conditions.
4	They assign integration actions to the steps (e.g. calling the API of the firewall, EDR, ticketing) and set the necessary parameters.
5	The user runs a test of the playbook in debug/simulation mode and verifies that the steps proceed as expected.
6	After a successful test, they submit the playbook for approval to the Manager and, after approval, it is marked as “production” for UC_07.

ALT_20b_01 - The playbook fails the test in simulation mode	Alternate

Step	**Activity**
1	After the playbook test is run in simulation mode, the system detects an error in one of the steps or missing parameters; the test ends with an error and the playbook remains in the draft state until the Analyst or Developer modifies it.

ALT_20b_02 - Saving a playbook with an incomplete configuration	Alternate

Step	**Activity**
1	The user attempts to save a playbook in which mandatory fields are missing or not all integration steps are configured; the system refuses to save it and highlights the missing or invalid settings.

Linked functional requirements	
FREQ_059 - Ability to create custom playbooks without the need for programming	The ability to create and manage playbooks using a visual interface without the need for coding. Possibility of creating custom playbooks without the need for programming
FREQ_067 - Playbook debugging & testing	The possibility of testing and debugging playbooks before their live deployment. Playbook debugging & testing
FREQ_060 - Workflow engine with branching, conditions and loops	The possibility of defining complex logic in playbooks using decision points. Workflow engine with branching, conditions and loops
UC_20b – Manage incident response playbooks	UC_20b addresses the design, editing and maintenance of playbooks that the SOC uses for automated or semi-automated response (UC_07).
UC_20b – Manage incident response playbooks	UC_20b addresses the design, editing and maintenance of playbooks that the SOC uses for automated or semi-automated response (UC_07).

Use case	UC_20c – Approve, version and publish KB and playbooks
Objective	UC_20c addresses the governance layer over the KB and playbooks – approval, versioning, change auditing and the approval workflow.

Precondition	Drafts or modified versions of KB articles and playbooks exist (UC_20a, UC_20b).
Precondition	The Manager has permission to approve and publish content.

Postcondition	Each record has a status (draft, in approval, published, archived) and a tracked history of changes and authors.

Basic Path	Basic Path

Step	**Activity**
1	The Manager opens the list of KB articles and playbooks awaiting approval.
2	The Manager selects a specific record and displays the current version together with an overview of the changes compared with the previous version.
3	The Manager decides whether to approve the record, return it for further work or mark it for archiving.
4	Upon approval, the system sets the new version as valid, moves the older version to the history and records the date and author of the approval.
5	According to the configured workflow, the system informs the relevant users that a new article or playbook has been published and is available for use.

ALT_20c_01 - Error in content detected after publication	Alternate

Step	**Activity**
1	After a new version of a KB article or playbook is published, the Analyst notices a factual error, creates a corrective version, submits it for expedited approval and, after approval, the corrective version becomes valid.

ALT_20c_02 - The primary approver is not available	Alternate

Step	**Activity**
1	The record is awaiting approval, but the primary Manager is unavailable for a long period; after a defined time has elapsed, the system offers the approval to a substitute approver according to the configured approval workflow.

Linked functional requirements	
FREQ_184 - Ability to create approval workflows for new KB articles	Approval processes for the publication of new articles. Possibility of creating approval workflows for new KB articles
UC_20c – Approve, version and publish KB and playbooks	UC_20c addresses the governance layer over the KB and playbooks – approval, versioning, change auditing and the approval workflow.

4.5.2.23 UC_21 – Manage and evaluate machine learning models for incident detection

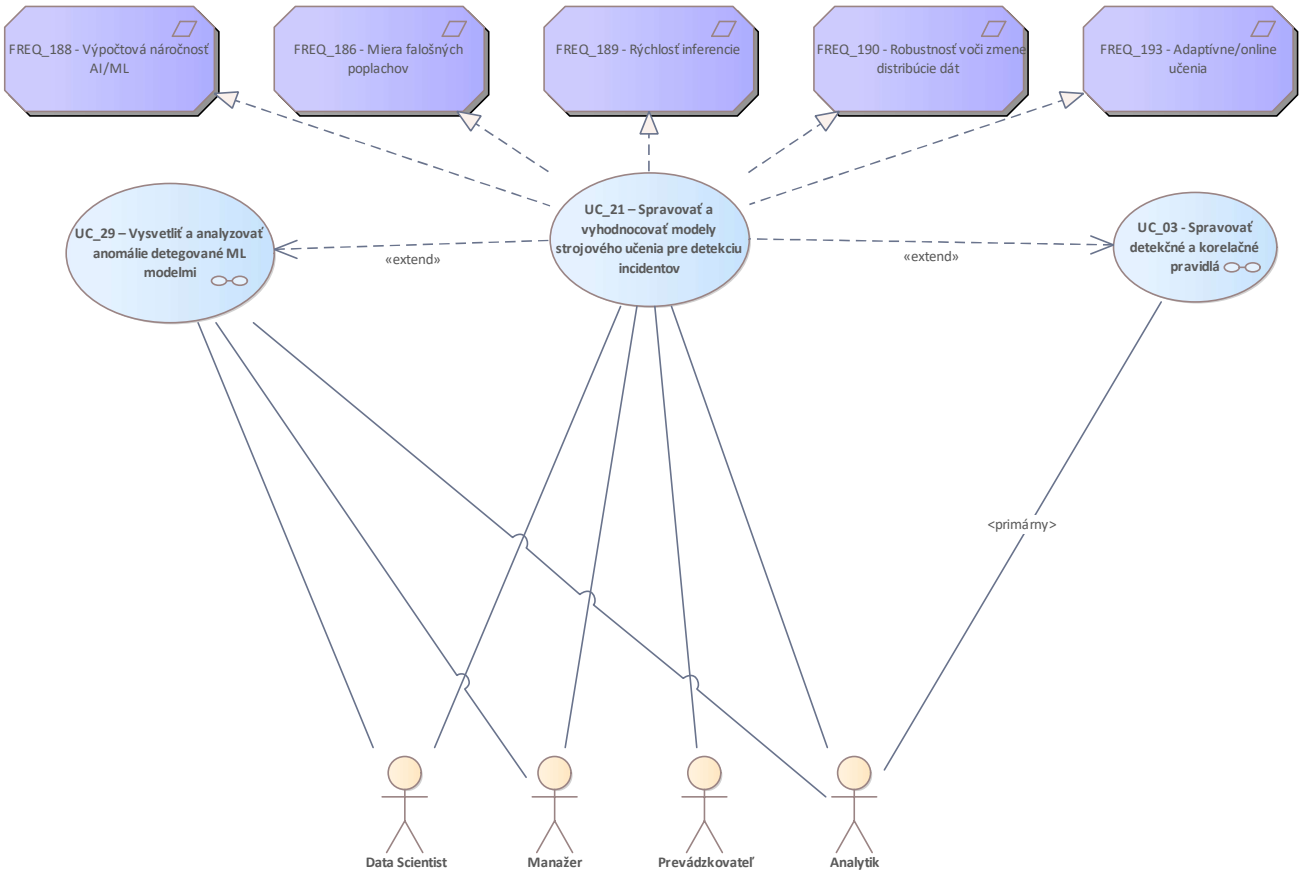


Figure 26 : UC_21 – Manage and evaluate machine learning models for incident detection

Use case	UC_21 – Manage and evaluate machine learning models for incident detection
Objective	UC_21 addresses the life cycle of the ML models that we use to detect incidents and anomalies in SOCaaS: design and training of models, quality measurement (FP/FN, accuracy, speed), robustness to changes in data, adaptive/online learning while the system is running.

Precondition	SOCaaS has ML-based detection component(s) (anomalies, classifiers, scoring).
Precondition	Historical data/logs are available for training and testing the models.
Precondition	Basic quality metrics that we want to achieve are set (e.g. a limit on FP/FN).
Postcondition	The ML models have had their quality evaluated (FP/FN, speed, robustness) and are either deployed or rejected.
Postcondition	The system is able to respond to changes in data (drift) by means of retraining or online learning.
Postcondition	The results of model evaluation are recorded for further decision-making and audit.
Basic Path	Basic Path

Step	**Activity**
1	The Data Scientist selects a dataset (historical logs, incidents) as the basis for training or testing the model.
2	The Data Scientist trains a new model or modifies an existing model based on current detection requirements.
3	The system evaluates the quality of the model on the validation set, including the metrics of false alarms and missed incidents (FP/FN).
4	The system evaluates the computational demands of the model during training (CPU, GPU, RAM, training time) and compares them with the available resources.
5	The system measures the inference speed of the model – the time needed to process new events under conditions close to real operation.
6	The Data Scientist evaluates the robustness of the model to changes in the data distribution (e.g. new log types, changes in behavioural patterns) and the impact on detection quality
7	If the model meets the required metrics, the Manager approves its deployment to production, and the Operator ensures the deployment of the model in the detection engine.
8	After deployment, the system continuously collects feedback from real operation (actual FP/FN) and prepares the basis for further tuning or adaptive learning.

ALT_21_02 – The model is too slow for real time	Alternate
Step	**Activity**
1	The inference test shows that the model is unable to process incoming data within the required time; the Data Scientist selects a more efficient architecture, simplifies the model or proposes offloading to another environment (e.g. a dedicated GPU server).

ALT_21_03 – Significant data drift appears after deployment	Alternate
Step	**Activity**
1	After some time, it is found that new types of events or behaviour have reduced the detection quality; the system alerts to the drift, and the Data Scientist prepares an updated training set and launches adaptive/online learning so that the model reflects the new data.

ALT_21_01 – The model has too high a number of false alarms (FP)	Alternate
Step	**Activity**
1	After evaluation, it turns out that the model generates an unacceptably high number of FPs; the Data Scientist adjusts the architecture, hyperparameters or threshold values and repeats the training and evaluation process until an acceptable level is achieved.

Linked functional requirements	
FREQ_193 - Adaptive/online learning	The solution must have the capability of adaptive/online learning. Support for updating the model during operation without retraining. The ability to update the model without the need for complete retraining, e.g. by means of online learning.
FREQ_186 - False alarm rate	The solution must ensure evaluation of the false alarm rate (False Positives/Negatives). How often the algorithm generates false positives/negatives. Reducing the false alarm rate is key to usability in the SOC. Ideal models have a very low number of FP/FN.
UC_29 – Explain and analyse anomalies detected by ML models	UC_29 ensures that SOCaaS is able to explain comprehensibly the anomalies detected by machine learning models: display the context of the data flow (logs, metrics, network flows), show the time course of the signals that led to the anomaly, work with unsupervised models (without labels), and give the analyst an interpretation of “why the model flagged it”.

FREQ_188 - Computational demands of AI/ML	The solution must be scalable with respect to computational demands (CPU/GPU, memory). Resources required for model training and inference. Varying load on computing resources, including GPU and RAM demands and training duration.
UC_03 - Manage detection and correlation rules	This use case covers the activities of analysts and administrators in creating, managing and maintaining detection and correlation rules in the SOCaaS environment. These rules enable the detection of security threats based on IOCs, behavioural anomalies, event chaining or threshold values. The use case focuses exclusively on the rule management process – their definition, tuning, classification and mapping to security frameworks (e.g. MITRE ATT&CK).
FREQ_190 - Robustness to changes in data distribution	The solution must ensure robustness to changes in data distribution (drift, noise). Resilience to noise, changes in data or attacksThe model must be resilient to unexpected changes, fluctuations or noise, otherwise it will stop working correctly.
FREQ_189 - Inference speed	The solution must ensure evaluation of inference speed (detection time). Response speed in detection. How quickly the model can provide a result on new data. Critical for real-time deployment.

4.5.2.24 UC_22 – Manage and deploy detection packages and rules

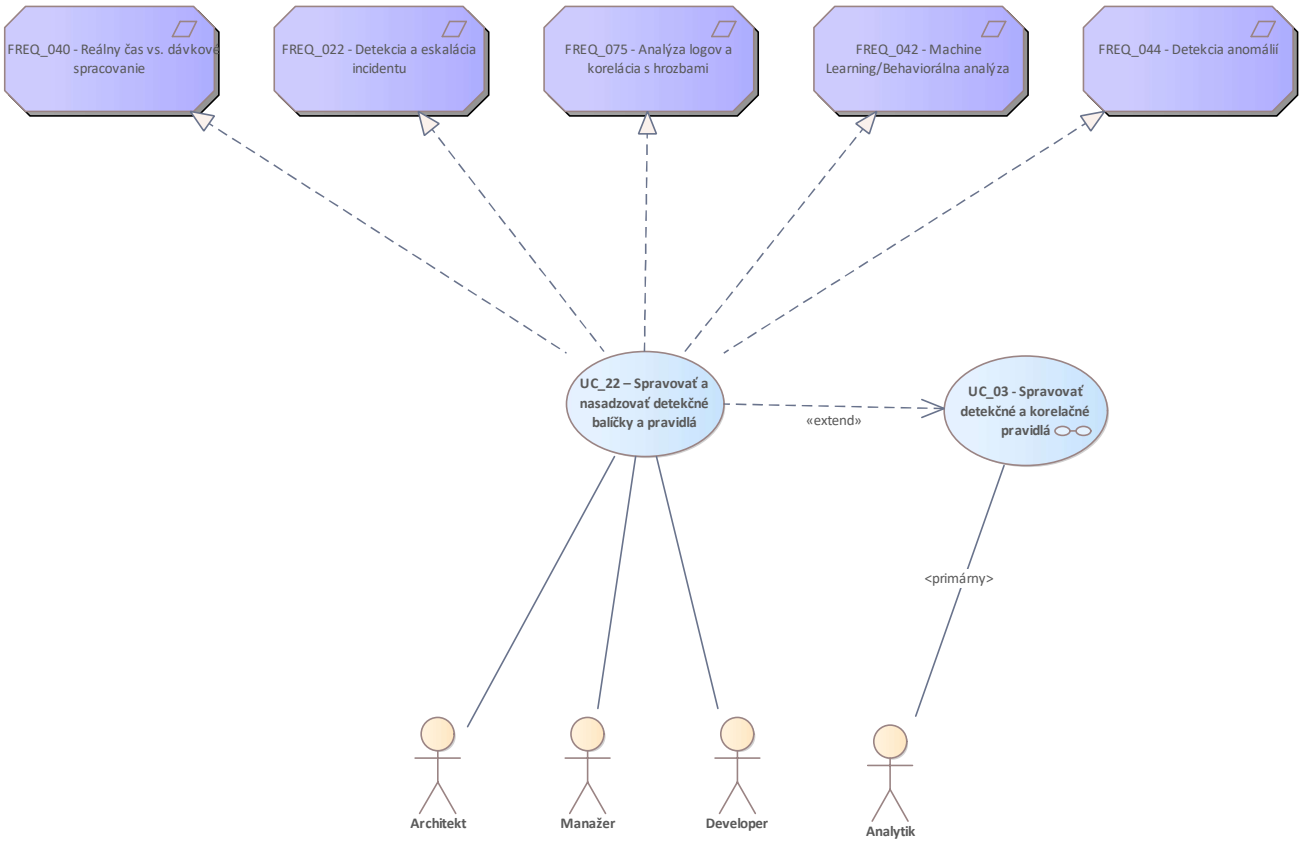


Figure 27 : UC_22 – Manage and deploy detection packages and rules

Use case	UC_22 – Manage and deploy detection packages and rules
Objective	UC_22 addresses the life cycle of detection content (rules, correlation rules, signatures, “content packs”) in SOCaaS: design and modification of detection rules and packages,

	testing on historical or synthetic data, controlled deployment to production, including the possibility of rollback, restricted deployment for selected tenants/customers. UC_22 does not address the detection runtime itself (that is covered in UC_03), but rather the process of designing, testing and the controlled deployment of detection rules and packages to the engine.

Precondition	There is a detection engine (UC_03) that supports the configuration of rules/packages.
Precondition	The platform has a mechanism for testing rules (a separate test environment or test mode).
Precondition	Roles and competences are defined as to who can create/modify, and who must approve, detection rules.
Postcondition	New or modified detection rules are tested, approved and deployed to production, or rejected.
Postcondition	Rule changes are versioned and auditable, and in case of problems it is possible to revert to the previous version.

Basic Path	Basic Path

Step	**Activity**
1	The Analyst opens the “Detection rules and packages” module and displays the list of existing rules.
2	The Analyst creates a new rule or modifies an existing one, defining conditions, threshold values and the mapping to an incident type.
3	The Analyst saves the change as a proposal (draft) and runs a test of the rule on historical or synthetic data in test mode.
4	The system evaluates the test results – the number of hits, potential false alarms and the impact on existing detection.
5	Based on the test results, the Analyst modifies the rule if necessary and repeats the test until acceptable behaviour is achieved.
6	The Analyst submits the rule for approval to the Manager or to the defined owner of the detection content.
7	The Manager assesses the impact of the rule (in particular on the number of alerts, threat types and affected customers/tenants) and decides whether to approve it or return it for modification.
8	After approval, the Developer / DevSecOps launches the deployment of the rule to production, or only for selected tenants or in restricted (“canary”) mode.
9	The system records all changes (rule version, author, approver, deployment time) and monitors the impacts on alerts and performance.

ALT_22_01 – The rule test reveals an extremely high number of hits	Alternate

Step	**Activity**
1	After step 4, the Analyst finds that the rule would generate a disproportionate number of alerts; they modify the rule (tightening conditions, adding filters) and repeat the test until an acceptable number of hits is achieved.

ALT_22_02 – The deployed rule negatively affects production	Alternate

Step	**Activity**
1	After deployment in step 8, the system detects a significant increase in the number of alerts, latency or errors; the Operator or DevSecOps initiates a reversion to the previous version of the rules and the new rule is marked as “requires modification/test”.

ALT_22_03 – The rule is to be deployed only for selected customers	Alternate

Step	**Activity**

1	During approval in step 7, the Manager decides that the rule is relevant only for a certain type of customer (e.g. a specific sector or tenant); during deployment, only a subset of tenants to which the rule is applied is selected

Linked functional requirements	
FREQ_040 - Real-time vs. batch processing	The ability to process logs in real time (streaming) or only in batches. Real-time vs. batch processing
FREQ_044 - Anomaly detection	Detection of non-standard behaviour or deviations from expected behaviour. Anomaly detection
FREQ_075 - Log analysis and correlation with threats	Log analysis with the aim of detecting correlated events and potential threats. Log analysis and correlation with threats
FREQ_042 - Machine Learning/Behavioural analysis	The ability of the SIEM system to detect anomalies using machine learning. Machine Learning/Behavioural analysis
FREQ_022 - Incident detection and escalation	The solution must ensure incident detection and escalation. Rules for detecting anomalies and known threats must be configured in the environment. After detection: • an alert is generated • the alert is automatically transformed into an incident • Zammad provides a ticket for the L1 team and escalation support, • the team evaluates the observables and uses enrichment
UC_03 - Manage detection and correlation rules	This use case covers the activities of analysts and administrators in creating, managing and maintaining detection and correlation rules in the SOCaaS environment. These rules enable the detection of security threats based on IOCs, behavioural anomalies, event chaining or threshold values. The use case focuses exclusively on the rule management process – their definition, tuning, classification and mapping to security frameworks (e.g. MITRE ATT&CK).

4.5.2.25 UC_23 – Manage and operate vulnerability scanning

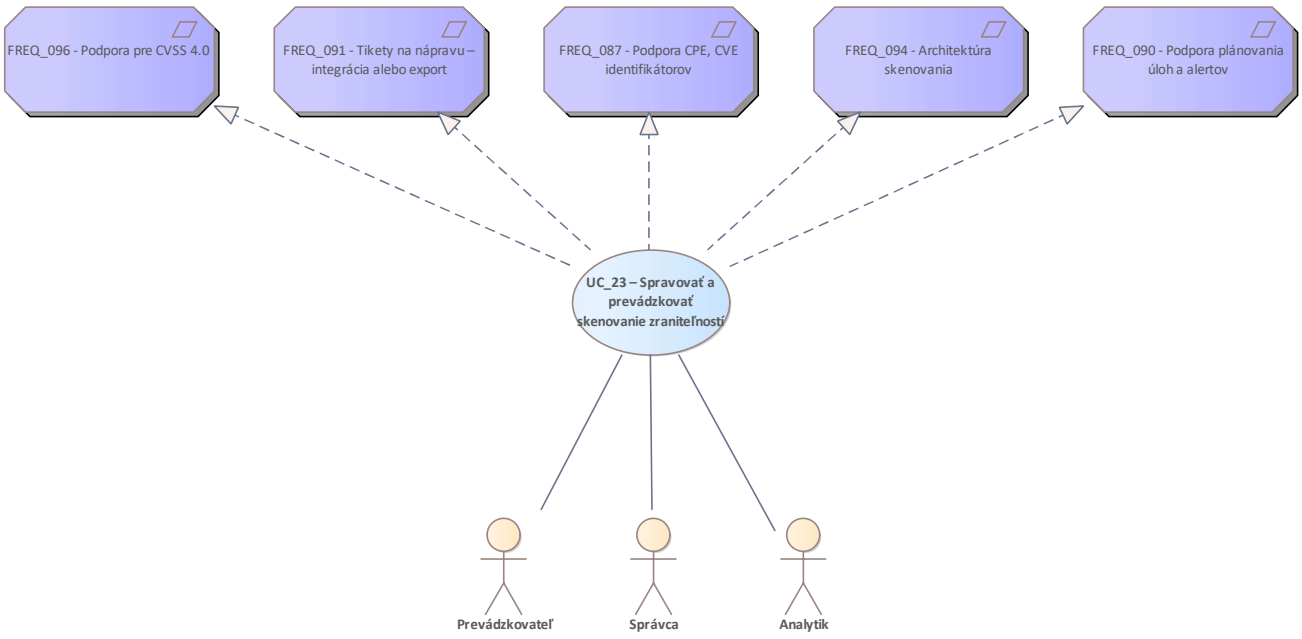


Figure 28 : UC_23 – Manage and operate vulnerability scanning

Use case	UC_23 – Manage and operate vulnerability scanning

Objective	UC_23 addresses the operation and management of vulnerability scanning within SOCaaS: planning and launching scans, assigning scans to assets and networks, the technical architecture of scanning (master/sensor), integration of findings into tickets and into the vulnerability management process (UC_12).

Precondition	At least the basic assets and network ranges (UC_11) intended for scanning are known.
Precondition	A vulnerability scanning tool is deployed in the environment (on-prem, VM or SaaS).
Precondition	The Administrator and the Operator have permissions to configure scanners and schedule scanning.
Postcondition	Vulnerability scanning for the selected ranges has been carried out according to the schedule or on demand.
Postcondition	Findings are enriched with CPE/CVE and CVSS scores and are available for UC_12.
Postcondition	Relevant findings are exported or linked to remediation tickets.

Basic Path	Basic Path

Step	**Activity**
1	The Operator or Administrator opens the “Vulnerability scanning” module and displays an overview of the existing scanning profiles.
2	The Operator selects the ranges (IP, networks, asset groups) and chooses the scan type (full, quick, authenticated).
3	The Operator schedules regular execution of scans (e.g. weekly at night) or launches a one-off ad hoc scan.The Operator schedules regular execution of scans (e.g. weekly at night) or launches a one-off ad hoc scan.
4	The system assigns the scans to the available scanning nodes according to the architecture (master/sensor) and optimizes the load distribution.The system assigns the scans to the available scanning nodes according to the architecture (master/sensor) and optimizes the load distribution.
5	After the scan is completed, the system processes the results, assigns CPE/CVE identifiers to the vulnerabilities and calculates CVSS scores.After the scan is completed, the system processes the results, assigns CPE/CVE identifiers to the vulnerabilities and calculates CVSS scores.
6	The Analyst or Administrator displays the scan results, filters the findings by severity and scope and marks those that are to be addressed.The Analyst or Administrator displays the scan results, filters the findings by severity and scope and marks those that are to be addressed.
7	The system creates or updates remediation tickets in the integrated ticketing system or exports the findings in the selected format for further processing.

ALT_23_01 – Scanning cannot be completed on part of the infrastructure	Alternate

Step	**Activity**
1	During the scan, some systems are unavailable or are blocked by a firewall; the system marks these parts as “not scanned”, the Operator receives information and can schedule a replacement scan with an adjusted configuration or in a different time window.

ALT_23_02 – Overload of the scanning infrastructure	Alternate

Step	**Activity**
1	After several scans are launched at once, the system detects excessive load on the scanning nodes; the scans are queued or spread over time so as not to exceed the permitted limits, and the Operator is alerted that it would be advisable to strengthen the scanning architecture.

Linked functional requirements

FREQ_087 - Support for CPE, CVE identifiers	The possibility of identifying vulnerabilities according to standardized identifiers. Support for CPE, CVE identifiers
FREQ_096 - Support for CVSS 4.0	Support for the latest version of the CVSS scoring system. Support for CVSS 4.0
FREQ_090 - Support for scheduling tasks and alerts	The possibility of automating repeated tasks and the generation of alerts. Support for task and alert scheduling (optional)
FREQ_091 - Remediation tickets – integration or export	The possibility of exporting findings to a ticketing system or to CSV, JSON, PDF. Remediation tickets – integration or export
FREQ_094 - Scanning architecture	Scalability and customization of scan distribution. Scanning architecture (monolithic or master/sensor)

4.5.2.26 UC_24 – Orchestration of integrations between SOCaaS and external systems

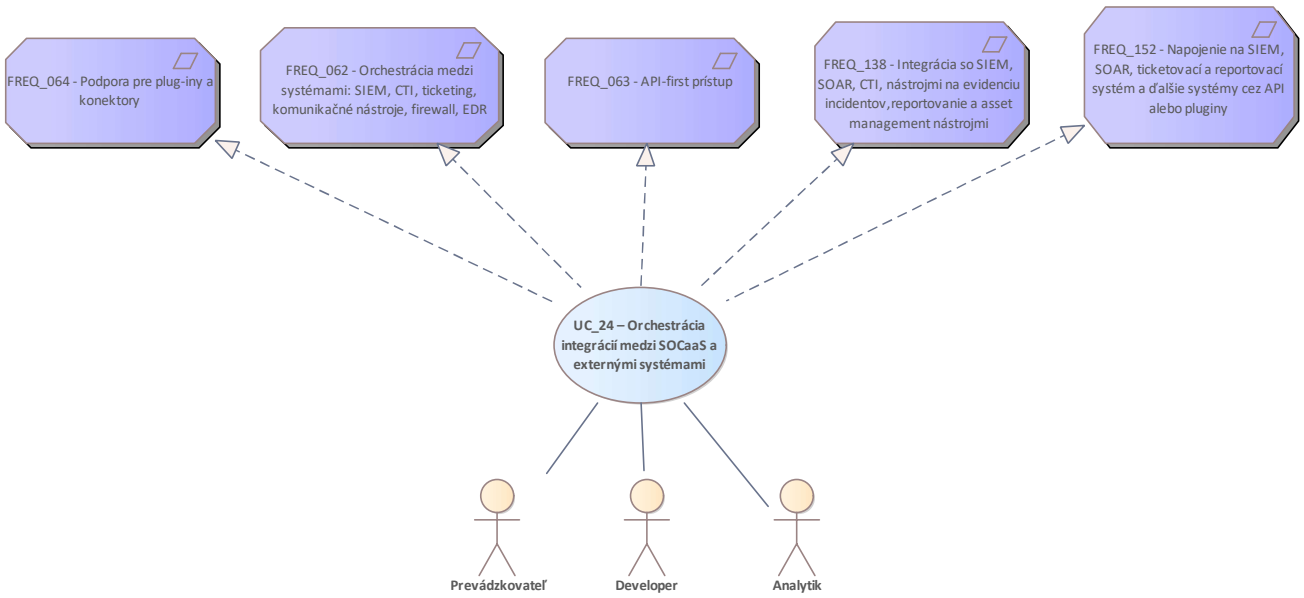


Figure 29 : UC_24 – Orchestration of integrations between SOCaaS and external systems

Use case	UC_24 – Orchestration of integrations between SOCaaS and external systems
Objective	UC_24 addresses the technical and process orchestration of integrations between SOCaaS and other systems: SIEM, SOAR, CTI, ticketing, communication tools, firewall, EDR, API integrations, webhooks, plug-ins and connectors, unified management of data flows and actions across these systems. The aim is for incident response, automation (UC_07) and incident management to function as a single interconnected ecosystem rather than as isolated tools.

Precondition	The architecture of integrations between the customer and SOCaaS is designed (UC_13) and at least a basic connection to SIEM/SOAR/ticketing exists.
Precondition	The SOCaaS platform supports REST API, webhooks and a plug-in/connector mechanism.
Precondition	The Operator and the Developer have permissions to configure integrations and orchestrations between systems.

Postcondition	The defined integration links and flows between SOCaaS, SIEM, SOAR, CTI, ticketing, firewalls, EDR and other systems are active and monitored.
Postcondition	Incidents and events are consistently synchronized between systems (status, ID, links).
Postcondition	An audit and overview are available of which integration channels are active, which actions they trigger and how they fail.

Basic Path	Basic Path

Step	**Activity**
1	The Operator opens the “Integrations and orchestration” module and displays an overview of the existing integration links (SIEM, SOAR, CTI, ticketing, EDR, firewall, communication tools).
2	The Developer or Operator creates a new integration link or modifies an existing one, selects the target system (e.g. ticketing system, CTI, firewall) and the integration type (API, webhook, plug-in).
3	The user configures authentication, endpoints, data formats and field mapping between SOCaaS and the target system.
4	The user defines which events in SOCaaS are to trigger actions in external systems (e.g. incident creation → create ticket, incident status change → update ticket, IOC from TI → update block list in the firewall).
5	The system saves the configuration and allows a test of the integration link to be run on a small data sample or a test incident.
6	After a successful test, the Operator activates the integration link and sets whether it is to be applied globally or only for selected customers/tenants.
7	In real operation, SOCaaS begins to use the configured orchestration – it generates webhooks, calls APIs or launches plug-ins according to the defined rules and mappings.
8	The system continuously logs successful and unsuccessful integration calls, displays statistics and alerts the Operator to errors or outages of integration channels.

ALT_24_01 – Failure of an integration call	Alternate

Step	**Activity**
1	During a test or production API/webhook call, the external system returns an error or is unavailable; SOCaaS marks the record as unsuccessful, saves the error detail and, depending on the settings, either repeats the call using a retry policy or creates an internal incident/report for the Operator.

ALT_24_02 – Conflict in data mapping between systems	Alternate

Step	**Activity**
1	When setting up field mapping between SOCaaS and an external system, it is found that some mandatory fields in the target system cannot be unambiguously populated from SOCaaS data; the Operator must add a transformation or default values, or restrict the type of events for which the integration is used.

Linked functional requirements	
FREQ_062 - Orchestration between systems: SIEM, CTI, ticketing, communication tools, firewall, EDR	Interconnection and data exchange with other systems within the incident response process. Orchestration between systems: SIEM, CTI, ticketing, communication tools, firewall, EDR
FREQ_063 - API-first approach	Accessibility of the tool's functions via API and support for webhooks for notifications and interactions. API-first approach (REST, Webhooks)
FREQ_138 - Integration with SIEM, SOAR, CTI, incident tracking tools,reporting and asset management tools	Integration with other security tools for automation and workflow. Integration with SIEM, SOAR, CTI, incident recording tools, reporting tools and asset management tools
FREQ_152 - Connection to SIEM, SOAR, ticketing and reporting systems and other systems via API or plugins	Possibility of bidirectional data exchange. Connection to SIEM, SOAR, ticketing and reporting systems and other systems via API or plugins
FREQ_064 - Support for plug-ins and connectors	The ability to extend the tool's functionality through modules or connectors. Support for plug-ins and connectors

UC_24 – Orchestration of integrations between SOCaaS and external systems	UC_24 addresses the technical and process orchestration of integrations between SOCaaS and other systems: SIEM, SOAR, CTI, ticketing, communication tools, firewall, EDR, API integrations, webhooks, plug-ins and connectors, unified management of data flows and actions across these systems. The aim is for incident response, automation (UC_07) and incident management to function as a single interconnected ecosystem rather than as isolated tools.

4.5.2.27 UC_25 – Manage threat intelligence (CTI) and the IOC lifecycle

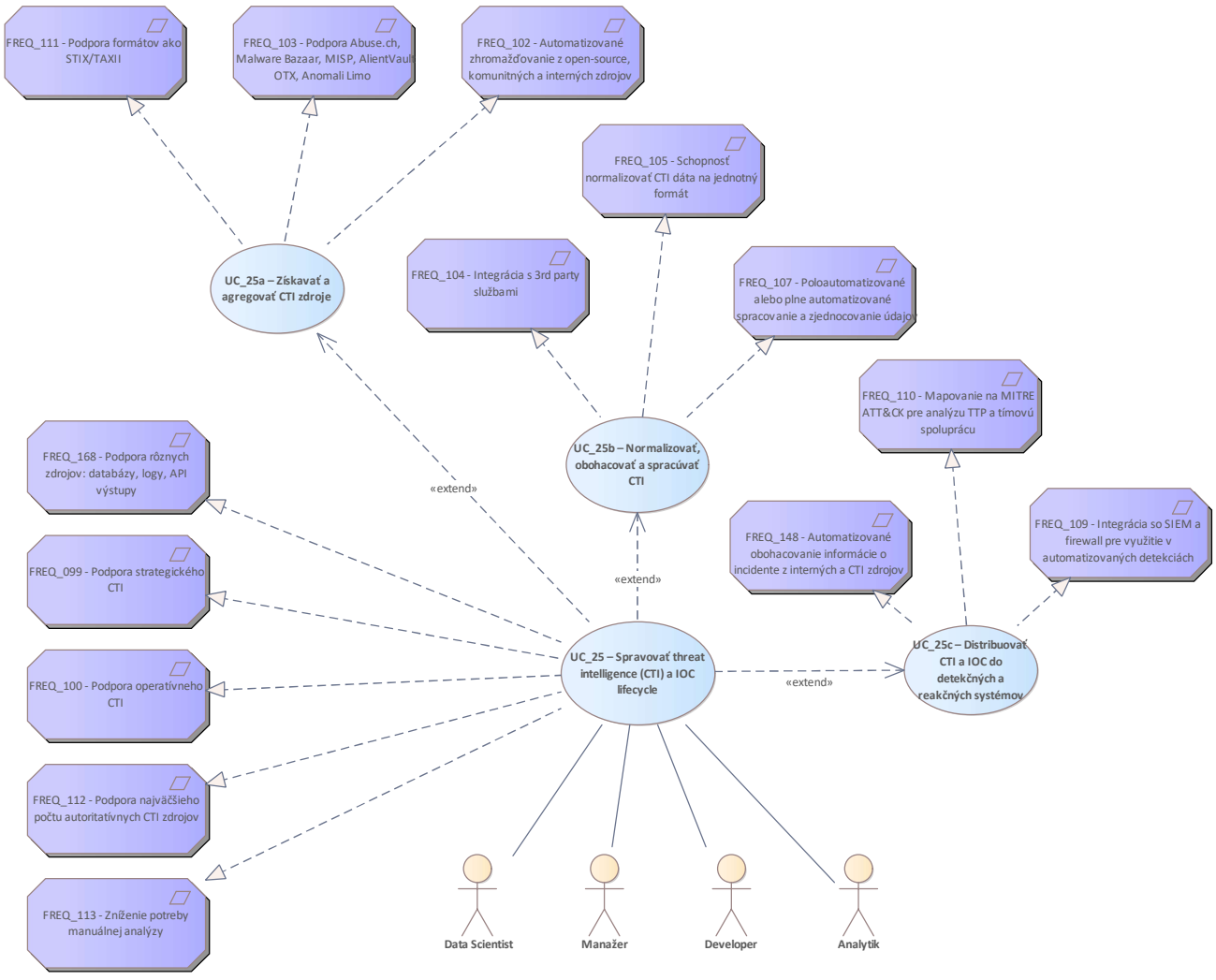


Figure 30 : UC_25 – Manage threat intelligence (CTI) and IOC lifecycle

Use case	UC_25 – Manage threat intelligence (CTI) and the IOC lifecycle
Objective	UC_25 addresses the complete CTI lifecycle in SOCaaS: obtaining CTI from various sources (feeds, MISP, open-source, vendor), normalization, enrichment and prioritization of CTI data,

	distribution of IOCs/TTPs to detection (SIEM, EDR, firewall, SOAR) and to incidents.

Precondition	SOCaaS has a CTI module (or is integrated with an external CTI tool).
Precondition	CTI sources (community, commercial, internal) are defined, along with a basic policy on which sources are “trusted”.
Precondition	Detection systems (SIEM, EDR, firewall, IDS, SOAR) support the receipt of IOCs/TTPs from CTI.
Postcondition	CTI data are systematically collected, normalized, enriched and aggregated.
Postcondition	Threats are prioritized and inputs for detection, hunting and incidents are created from the relevant IOCs/TTPs.
Postcondition	IOCs/TTPs are distributed to detection systems and incident management in a consistent manner (with an audit trail).

Basic Path	Basic Path

Step	**Activity**
1	The Analyst opens the “Threat Intelligence” module and displays an overview of active CTI sources and the latest updates.
2	Periodically or on demand, the system downloads new CTI data from all configured sources (open-source, community, vendor, internal).Periodically or on demand, the system downloads new CTI data from all configured sources (open-source, community, vendor, internal).
3	The CTI pipeline normalizes, enriches and aggregates the data – it merges IOCs, adds context and links them with historical incidents and logs.
4	The Data Scientist defines or adjusts the rules for scoring, prioritization and categorization of threats (criticality, relevance for specific customers).
5	Based on the scoring, the system automatically identifies the most relevant threats and IOCs that are to be used in detection, hunting and incidents.
6	For selected IOCs/TTPs, the CTI module creates outputs for detection (e.g. block lists, SIEM rules, hunting queries) and prepares them for distribution to the connected systems.
7	In incidents and hunting tools, the Analyst has enriched CTI available – they can see campaigns, TTPs, groups and related IOCs.
8	The system continuously records which CTI sources and IOCs/TTPs were used in real incidents and hunting cases and uses this information for further fine-tuning of scoring and relevance.

ALT_25_01 – A CTI source ceases to be trustworthy or available	Alternate

Step	**Activity**
1	Over time, it becomes apparent that a specific CTI source has low-quality or duplicate data, or is unavailable for a long period; the Manager or CTI specialist deactivates the source or reduces its weight in the scoring.

ALT_25_02 – CTI surplus and analyst overload	Alternate

Step	**Activity**
1	The system detects that the quantity of new IOCs/threats significantly exceeds the capacity of the analysts; the CTI specialist adjusts the scoring, filters and thresholds so that only the most relevant threats come to the fore (e.g. according to sector, geolocation, asset exposure).

ALT_25_03 – CTI mismatch between customers/tenants	Alternate

Step	**Activity**
1	A certain IOC is relevant only for a specific customer (e.g. according to its technologies or geography); the system supports per-tenant filtering of CTI, so IOCs/TTPs are distributed only to the customers they concern.

Linked functional requirements

FREQ_112 - Support for the largest number of authoritative CTI sources	An extensive selection of community, public and paid sources. Support for the largest number of authoritative CTI sources
FREQ_168 - Support for various sources: databases, logs, API outputs	The ability to aggregate data from multiple technical and analytical sources. Support for various sources: databases, logs, API outputs
UC_25b – Normalize, enrich and process CTI	UC_25b addresses the CTI processing pipeline: normalization of various formats into a unified model, enrichment of IOCs and artefacts from external services, workflow for automatic/semi-automatic processing of CTI.
UC_25a – Obtain and aggregate CTI sources	UC_25a addresses the ingest and connection of CTI sources: defining sources (feeds, MISP, API, vendor portals), automated downloading, use of standardized formats.
FREQ_099 - Support for strategic CTI	Obtaining and analysing information on long-term trends, attack impacts and strategic recommendations. Support for strategic CTI (reports, impact analyses, attack trends)
FREQ_113 - Reducing the need for manual analysis	Automation and prioritization of threats so that analysts deal only with the relevant ones. Reducing the need for manual analysis – saving analysts' time
FREQ_100 - Support for operational CTI	The ability to map techniques, tactics and procedures (TTP) and to identify attacker groups and their behaviour. Support for operational CTI (TTP, group profiling)
UC_25c – Distribute CTI and IOCs to detection and response systems	UC_25c addresses the push of CTI to the rest of the ecosystem: forwarding IOCs to SIEM/IDS/EDR/firewalls, use of CTI in detection, playbooks and incidents,

Use case	UC_25a – Obtain and aggregate CTI sources
Objective	UC_25a addresses the ingest and connection of CTI sources: defining sources (feeds, MISP, API, vendor portals), automated downloading, use of standardized formats.

Precondition	The CTI module is available in SOCaaS.
Precondition	The CTI sources to which we have access are known (API keys, URLs, authentication).
Postcondition	CTI sources are configured, activated and regularly download new data in supported formats.

Basic Path	Basic Path

Step	**Activity**
1	The Developer opens the “CTI sources” section and displays the list of existing sources.
2	The Developer adds a new CTI source (feed, MISP, vendor API) and enters the necessary credentials.
3	They configure the download interval (periodic, on demand) and the type of data (IOCs, TTPs, campaigns, reports).
4	The system tests the connection, downloads a data sample and verifies the format and consistency.
5	After a successful test, the source is marked as active and included in the regular CTI ingest.

ALT_25a_01 – Error when connecting to a CTI source		Alternate
Step	**Activity**	
1	The test in step 4 fails (incorrect API key, expired token, unavailable endpoint); the system displays an error, deactivates the source and leaves it in the state “requires configuration correction”.	
ALT_25a_02 – Limit or quotas on the CTI source		Alternate
Step	**Activity**	
1	The CTI source has a limited number of requests per day; the system adjusts the download interval so that the quotas are not exceeded and alerts the SOC analyst	
Linked functional requirements		
FREQ_103 - Support for Abuse.ch, Malware Bazaar, MISP, AlientVault OTX, Anomali Limo	Ready-made connectors or integrations for well-known CTI sources. Support for Abuse.ch, Malware Bazaar, MISP, AlientVault OTX, Anomali Limo	
FREQ_102 - Automated collection from open-source, community and internal sources	Automatic downloading of data from various sources (e.g. GitHub, feeds, MISP, API systems). Automated collection from open-source, community and internal sources	
FREQ_111 - Support for formats such as STIX/TAXII	Use of standardized formats for the exchange of CTI data. Support for formats such as STIX/TAXII	
UC_25a – Obtain and aggregate CTI sources	UC_25a addresses the ingest and connection of CTI sources: defining sources (feeds, MISP, API, vendor portals), automated downloading, use of standardized formats.	
Use case	UC_25b – Normalize, enrich and process CTI	
Objective	UC_25b addresses the CTI processing pipeline: normalization of various formats into a unified model, enrichment of IOCs and artefacts from external services, workflow for automatic/semi-automatic processing of CTI.	

Precondition	CTI data have been downloaded from the sources (UC_25a).	
Precondition	Integration services for enrichment are configured (VirusTotal, Shodan, …).	
Postcondition	CTI data are in a unified format, enriched and vetted, ready for scoring and distribution.	
Basic Path		Basic Path
Step	**Activity**	
1	The system receives new CTI data from the sources and stores them in a buffer for processing.	
2	The CTI pipeline transforms various formats into a unified structured model (IOCs, TTPs, campaigns, meta).	
3	For each IOC or relevant artefact, the pipeline optionally performs enrichment via external services (VirusTotal, Shodan, DomainTools, …).	
4	The workflow engine applies the defined processing steps – deduplication, filtering, first-seen/last-seen logic and technical validations.	
5	The resulting CTI records are stored in a unified form and marked with a status (new, processed, unconfirmed, discarded).	

ALT_25b_01 – External enrichment is temporarily unavailable		Alternate
Step	**Activity**	
1	In step 3, an external service (e.g. VirusTotal) does not respond in time; the system marks the enrichment status as “pending” and schedules a retry later without blocking the entire pipeline.	
ALT_25b_02 – Conflict during normalization of CTI data		Alternate
Step	**Activity**	
1	Different sources provide conflicting information for the same IOC; the pipeline flags the record for manual review by a CTI specialist, who selects or combines the correct values.	
Linked functional requirements		
FREQ_104 - Integration with 3rd party services	The ability to enrich IOCs and other artefacts using external services. Integration with 3rd party services such as VirusTotal, Shodan, DomainTools, URLScan.io	
FREQ_105 - Ability to normalize CTI data into a unified format	Consolidation of data from various formats into a structured unified output. The ability to normalize CTI data to a unified format	
FREQ_107 - Semi-automated or fully automated processing and unification of data	Workflow capability for data processing without the need for manual intervention. Semi-automated or fully automated processing and unification of data	
UC_25b – Normalize, enrich and process CTI	UC_25b addresses the CTI processing pipeline: normalization of various formats into a unified model, enrichment of IOCs and artefacts from external services, workflow for automatic/semi-automatic processing of CTI.	
UC_25b – Normalize, enrich and process CTI	UC_25b addresses the CTI processing pipeline: normalization of various formats into a unified model, enrichment of IOCs and artefacts from external services, workflow for automatic/semi-automatic processing of CTI.	
Use case	UC_25c – Distribute CTI and IOCs to detection and response systems	
Objective	UC_25c addresses the push of CTI to the rest of the ecosystem: forwarding IOCs to SIEM/IDS/EDR/firewalls, use of CTI in detection, playbooks and incidents,	

Precondition	CTI data are processed and prioritized (UC_25, UC_25b).
Precondition	Integration links to detection systems exist (UC_24).
Postcondition	Relevant IOCs/TTPs are deployed in detection systems.
Postcondition	Incidents and hunting cases are enriched with CTI metadata.

Basic Path		Basic Path
Step	**Activity**	
1	The CTI module selects the IOCs/TTPs that are marked as “relevant for distribution” according to the scoring and policy.	
2	The Developer selects the detection systems to which the IOCs/TTPs are to be distributed (SIEM, IDS/IPS, EDR, firewall, SOAR).	

3	The system generates output artefacts (block lists, SIEM rules, feeds) and sends them via integration channels (API, plug-in, CTI interface).
4	The detection systems begin to use the new IOCs/TTPs in detection; the CTI module tracks the success rate and the manner of use in real incidents.
5	When an incident arises, SOCaaS automatically enriches the incident with information from CTI – it links IOCs, TTPs and MITRE ATT&CK techniques.

ALT_25c_01 – The detection system does not accept some IOCs	Alternate

Step	**Activity**
1	During distribution in step 3, the SIEM or firewall rejects some of the IOCs (e.g. due to format or a limit); the system marks these records as “not applied” and offers the CTI specialist the option of adjusting the format or scope.

ALT_25c_02 – An IOC is later marked as a false positive or irrelevant	Alternate

Step	**Activity**
1	Based on feedback from incidents, it is found that some IOCs cause spam in detection; the CTI module withdraws these IOCs from distribution and adjusts the scoring/policy so that similar threats are evaluated differently in the future.

Linked functional requirements	
FREQ_110 - Mapping to MITRE ATT&CK for TTP analysis and team collaboration	Linking incidents, IOCs and TTPs with the ATT&CK framework. Mapping to MITRE ATT&CK for TTP analysis and team collaboration
FREQ_109 - Integration with SIEM and firewall for use in automated detections	Forwarding of IOCs and data directly to detection systems (SIEM, IDS, firewall). Integration with SIEM and firewall for use in automated detections
FREQ_148 - Automated enrichment of incident information from internal and CTI sources	Integration for enriching IOCs and metadata from CTI or other systems. Automated enrichment of incident information from internal and CTI sources
UC_25c – Distribute CTI and IOCs to detection and response systems	UC_25c addresses the push of CTI to the rest of the ecosystem: forwarding IOCs to SIEM/IDS/EDR/firewalls, use of CTI in detection, playbooks and incidents,
UC_25c – Distribute CTI and IOCs to detection and response systems	UC_25c addresses the push of CTI to the rest of the ecosystem: forwarding IOCs to SIEM/IDS/EDR/firewalls, use of CTI in detection, playbooks and incidents,
UC_25c – Distribute CTI and IOCs to detection and response systems	UC_25c addresses the push of CTI to the rest of the ecosystem: forwarding IOCs to SIEM/IDS/EDR/firewalls, use of CTI in detection, playbooks and incidents,

4.5.2.28 UC_26 – Configure scheduling and export of reports

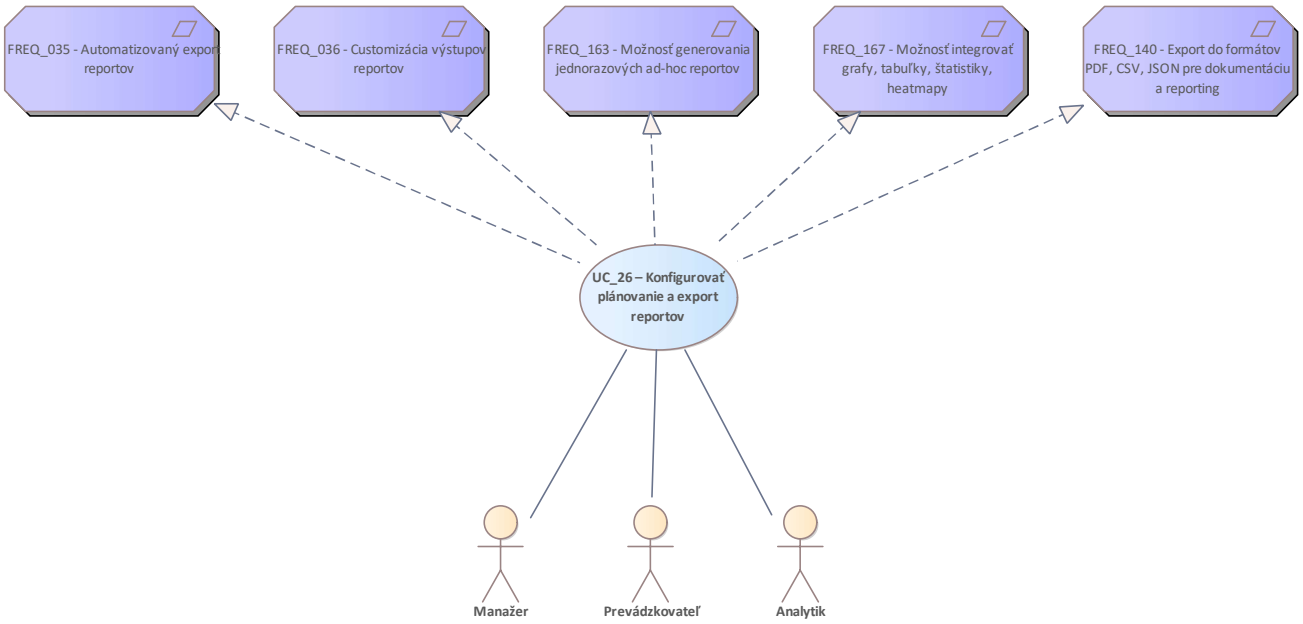


Figure 31 : UC_26 – Configure report scheduling and export

Use case	UC_26 – Configure scheduling and export of reports
Objective	UC_26 addresses the practical management of reports in SOCaaS: setting up automatic generation and distribution of reports, customization of the content and appearance of reports, generation of one-off ad hoc reports for managers or audit.

Precondition	A reporting module (UC_10) with defined report types (incidents, SLA, KPIs…) exists in the system.
Precondition	Distribution channels are configured (at least e-mail, ideally also webhook/integration endpoints).
Postcondition	Generation schedules, recipients and export formats are set for individual report types.
Postcondition	Users are able to generate one-off ad hoc reports outside the schedule.
Postcondition	Reports contain the selected metrics, charts and visualizations in the required design.

Basic Path	Basic Path

Step	**Activity**
1	The Manager or Operator opens the “Reports and export” module and displays the available report types.
2	The user selects a specific report type and chooses which metrics, time ranges and sections are to be included in the report.
3	The user configures the report generation schedule (daily, weekly, monthly) and the list of recipients or target endpoints.
4	The user adjusts the appearance of the report – organization logo, header, notes, or the structure of sections.
5	The user selects the export format (PDF, CSV, JSON) and the delivery method (e-mail, webhook, saving to storage).

6	At the defined times, the system automatically generates the report, adds the selected charts, tables and statistics and sends it to the selected channels.
7	The system keeps a history of generated reports so that they can be downloaded retrospectively or used as a basis for an audit.

ALT_26_01 – Creating a one-off ad hoc report	Alternate

Step	**Activity**
1	The Manager needs an extraordinary report outside the scheduled cycle; in the module, they select the report type, set filters and the time period and launch a one-off generation, whereby the system generates the report only once without changing the existing schedules.

ALT_26_02 – Unsuccessful delivery of an automatically generated report	Alternate

Step	**Activity**
1	During a regular report export, an error occurs when sending the e-mail or webhook; the system saves the report, marks the distribution as unsuccessful and alerts the Operator, who can repeat the delivery or modify the target addresses.

Linked functional requirements	
FREQ_036 - Customization of report outputs	The solution must ensure customization of report outputs. It will include: • Option to select metrics, fields, time ranges • Design personalization – logos, header, colour scheme • Inclusion of explanatory notes, remarks and contextual commentary
FREQ_167 - Ability to integrate charts, tables, statistics, heatmaps	Visualization of data directly in reports in various forms. Possibility of integrating charts, tables, statistics, heatmaps
FREQ_163 - Ability to generate one-off ad hoc reports	Creation of reports outside the regular schedule as needed. Possibility of generating one-off ad-hoc reports
FREQ_035 - Automated export of reports	The solution must ensure automated export of reports. It will be possible to export reports automatically: • Periodic reports – outputs generated and distributed daily, weekly or monthly. • Distribution via e-mail or webhook – reports can be automatically sent to specified addresses or endpoints. • Support for templates and custom filters – the option to set conditions, time period selections and formats as needed.
FREQ_140 - Export to PDF, CSV, JSON formats for documentation and reporting	The possibility of exporting tickets for archiving or reporting purposes. Export to PDF, CSV, JSON formats for documentation and reporting

4.5.2.29 UC_27 – Manage the security status and relationships of assets

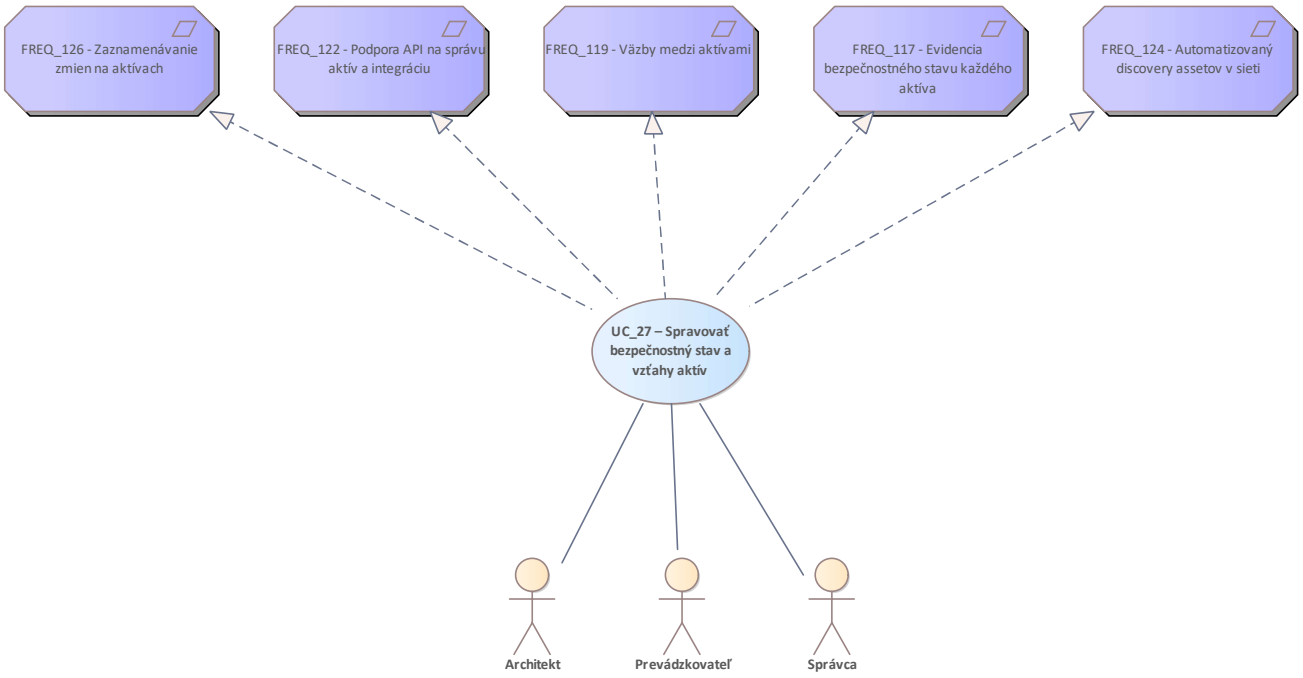


Figure 32 : UC_27 – Manage security posture and asset relationships

Use case	UC_27 – Manage the security status and relationships of assets
Objective	UC_27 extends asset management with a “smart CMDB”: it records the security status of each asset (patch level, vulnerabilities, recent changes), it displays and manages the links between assets (server–service–application), it uses automated discovery and API integrations, it keeps an audit of changes to assets.

Precondition	A basic asset inventory has been established (UC_11) – at least the key servers, networks and applications exist.
Precondition	Data sources for discovery are available (scans, CMDB, cloud API…).
Precondition	Users have defined roles for reading/editing assets.
Postcondition	The current security status (versions, patch level, vulnerabilities) is recorded for each asset.
Postcondition	Links between assets are defined (topology / dependencies).
Postcondition	Changes to assets are auditable (who, what, when) and available for forensic and operational analysis.
Basic Path	Basic Path

Step	**Activity**
1	The Administrator or Operator opens the “Assets” module and displays an overview of assets with their current security status.
2	The Operator configures automated discovery (e.g. network scanning, cloud API, CMDB integration) for selected segments of the environment.

3	The system launches discovery and identifies new or changed assets, comparing them with the existing inventory.
4	The Administrator reviews newly found or changed assets, assigns them to the correct categories and, as needed, adjusts the details (owner, criticality, location).
5	The Administrator or Architect adds or modifies the links between assets (e.g. server → service → application → database) so that they reflect the actual dependencies.
6	The system automatically updates the security status of assets based on inputs from scanning, patch management or other integration sources.
7	Every change to an asset (parameters, status, links) is saved with information on the author and time of the change so that the history can be analysed later.

ALT_27_01 – Conflict between discovery and the existing inventory	Alternate

Step	**Activity**
1	Discovery finds an asset that already exists in the inventory but with a different designation or parameters; the system flags the conflict, and the Administrator chooses whether to merge the record, overwrite it or keep it as a new asset.

ALT_27_02 – Failure of automated discovery	Alternate

Step	**Activity**
1	The discovery task fails (incorrect authentication, unavailable network, missing permissions); the system stops the discovery, records the error details and informs the Operator so that they can adjust the configuration or launch the discovery again.

Linked functional requirements	
FREQ_124 - Automated discovery of assets in the network	Automated collection and updating of asset data from the network and other systems. Automated discovery of assets in the network (e.g. SNMP, nmap, APIs from other systems)
FREQ_119 - Links between assets	The possibility of visualizing and managing interconnections between infrastructure components. Relationships between assets (e.g. server – service – application)
FREQ_126 - Recording of changes to assets	Auditability of all changes for forensic and operational transparency. Recording changes to assets (who changed what and when)
FREQ_122 - API support for asset management and integration	The possibility of managing and updating asset data via API. API support (REST, GraphQL) for asset management and integration
FREQ_117 - Recording of the security status of each asset	Storing information on vulnerabilities, patch level and system updates. Recording the security status of each asset (e.g. OS versions, last update)

4.5.2.30 UC_28 – Deploy and maintain SOCaaS in various environments

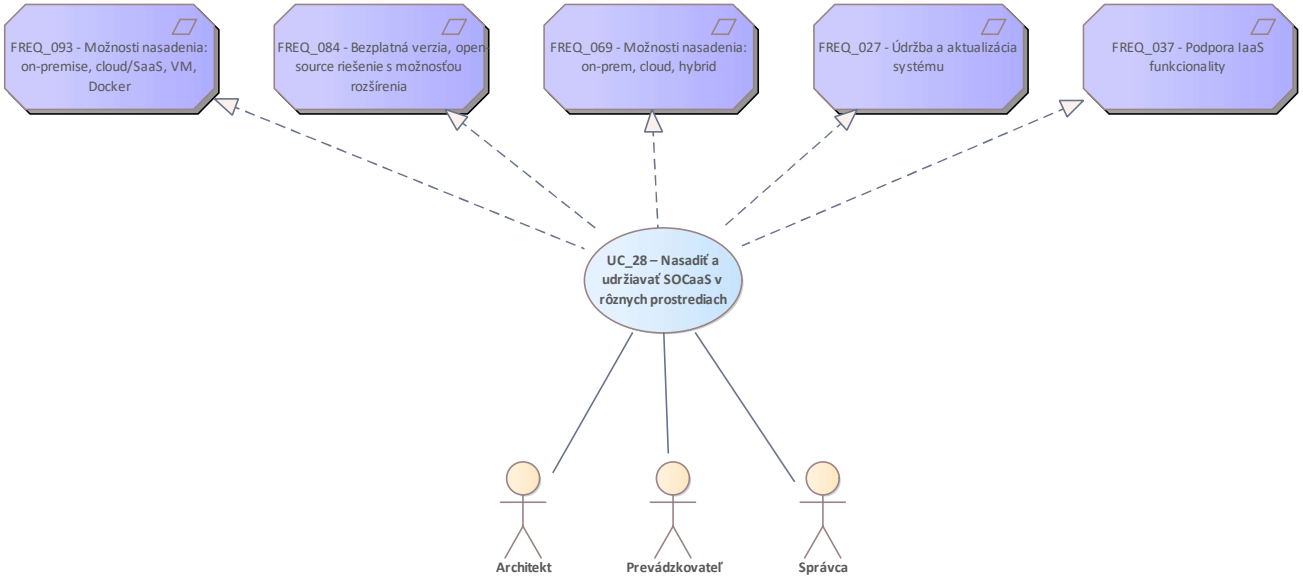


Figure 33 : UC_28 – Deploy and maintain SOCaaS in different environments

Use case	UC_28 – Deploy and maintain SOCaaS in various environments
Objective	UC_28 addresses the deployment and technical operation of the SOCaaS platform in various environments: on-premise at the customer, in the cloud, hybrid, as a dedicated IaaS environment for SOCaaS, with clear rules for maintenance, updates and monitoring, including the use of an open-source/free edition as the base platform. It is a technical and operational UC that extends platform operation (e.g. the UC on SOCaaS operation) with a multi-deployment / IaaS perspective.

Precondition	The architecture of the SOCaaS solution has been designed (topology, modules, integration points).
Precondition	Infrastructure (on-prem, cloud or IaaS) in which SOCaaS can be deployed is available.
Precondition	The Operator has defined processes for monitoring, maintenance and updates of components.
Postcondition	SOCaaS is deployed in the selected environment (on-prem/cloud/hybrid/IaaS) and is monitored.
Postcondition	Rules for maintenance, updates and testing of changes in a separate environment are set.
Postcondition	The platform is ready to scale according to the needs of tenants / customers.

Basic Path	Basic Path

Step	**Activity**
1	The Architect, together with the Operator, decides whether SOCaaS will be deployed on-premise, in the cloud or in a hybrid model.
2	The Operator prepares the target environment – creates the necessary virtual machines, networks,

	storage and other resources in IaaS.
3	The Operator installs or deploys the SOCaaS components in the selected form (e.g. VM, Docker containers, cloud/SaaS instance).
4	The Operator sets up centralized monitoring of module functionality (metrics collection, health-checks, status logging).
5	The Operator creates a test environment or staging instance in which updates and configuration changes will be verified.
6	The Operator defines the rules for maintenance and updates – maintenance windows, the order of deploying changes and the rollback procedure.
7	The Administrator or Operator continuously monitors the status of the deployment and, as needed, adds further resources (VMs, storage, network elements) to ensure performance and availability.

ALT_28_01 – Need to change the deployment model (on-prem → cloud or hybrid)	Alternate

Step	**Activity**
1	Based on growth or customer requirements, it turns out that the original on-prem deployment is no longer suitable; the Architect proposes a transition to the cloud or a hybrid model, and the Operator prepares the new environment and a gradual migration of components so that service availability is maintained.

ALT_28_02 – An update causes problems in the production environment	Alternate

Step	**Activity**
1	After deployment of an update according to steps 5–6, it is found that a module does not work correctly; the Operator uses the prepared rollback procedure, reverts to the previous version and first fixes and tests the update in the staging environment before the next attempt.

Linked functional requirements	
FREQ_084 - Free version, open-source solution with the possibility of extension	Availability of basic functionality without a licence and the possibility of extension. Free version, open-source solution with the possibility of extension
FREQ_069 - Deployment options: on-prem, cloud, hybrid	The possibility of deploying the solution in various environments, including an on-premises, cloud or hybrid model. Deployment options: on-prem, cloud, hybrid
FREQ_037 - Support for IaaS functionality	The solution requires full IaaS functionality – management of virtual machines (VMs), disks, networks, images, snapshots and standalone resources. The platform must support the creation and management of various instance types, allow resources to be allocated according to tenants' needs and at the same time offer a robust API for programmatic access. The ability to provide complete IaaS functionality is key to ensuring the flexible and reliable operation of SOC as a service.
FREQ_093 - Deployment options: on-premise, cloud/SaaS, VM, Docker	Flexibility of installation in various environments. Deployment options: on-premise, cloud/SaaS, VM, Docker
FREQ_027 - System maintenance and updates	The solution must ensure system maintenance and updates. The operations team must have established: • rules for component updates • a test environment for verifying changes • centralized monitoring of the functionality of all modules

4.5.2.31 UC_29 – Explain and analyse anomalies detected by ML models

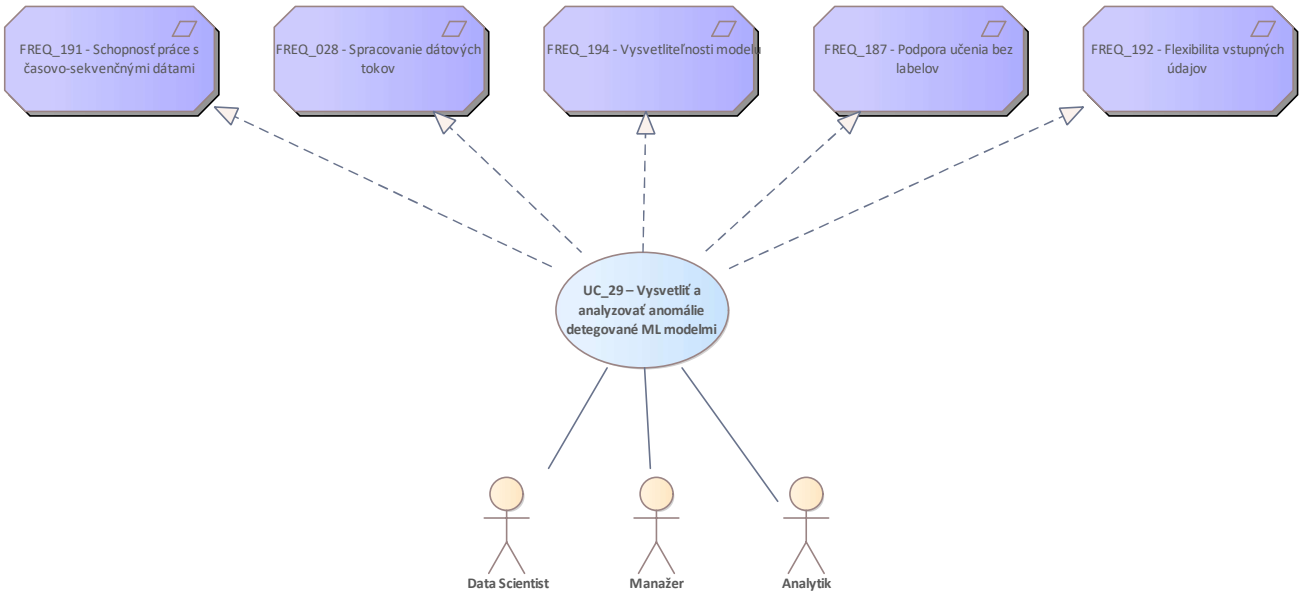


Figure 34 : UC_29 – Explain and analyse anomalies detected by ML models

Use case	UC_29 – Explain and analyse anomalies detected by ML models
Objective	UC_29 ensures that SOCaaS is able to explain comprehensibly the anomalies detected by machine learning models: display the context of the data flow (logs, metrics, network flows), show the time course of the signals that led to the anomaly, work with unsupervised models (without labels), and give the analyst an interpretation of "why the model flagged it".

Precondition	ML models for anomaly detection are deployed in SOCaaS (UC_21) and generate anomalous events/alerts.
Precondition	Data flows (logs, metrics, network flows) are processed via the pipeline – validated, normalized, enriched and stored.
Precondition	The user (Analyst, Data Scientist) has access to the "ML anomalies and explanations" module.
Postcondition	For the selected anomaly, an explanation of the detection exists – over time, over the relevant inputs and with a textual comment.
Postcondition	The Analyst can label the model output (TP/FP) and add a comment that is used as feedback for the models.
Basic Path	Basic Path

Step	**Activity**
1	The Analyst opens the "ML anomalies" module and selects a specific anomaly or incident that was detected by an ML model.
2	From the data pipeline, the system loads all related events – logs, metrics and network flows – that have passed validation, normalization and enrichment and have been assigned to the correct indices and timestamps.
3	The system identifies which ML model detected the anomaly (supervised/unsupervised) and loads its outputs (score, or cluster membership).

4	The Analyst launches the “explain detection” function and the system generates an explanation – the key signals, time segments and features that contributed most to the model's decision.
5	The system displays the time course of the selected metrics and log events (time series), highlighting the interval in which the anomaly was detected.
6	The system displays a breakdown of inputs by type (logs, metrics, flow data) and shows which groups of inputs most influenced the detection, including the case where they come from different sources.
7	The Analyst adds their own interpretation, labels the detection as “true positive”, “false positive” or “unclear” and saves the feedback; the Data Scientist can later use these labels when fine-tuning the models (including unsupervised ones).

ALT_29_01 – Insufficient data to explain the anomaly	Alternate

Step	**Activity**
1	When attempting an explanation, the system finds that some logs or metrics are missing in the selected time segment (the data were not stored or processed); the system alerts the analyst to the limited explanation, displays only the available data and marks the explanation as “incomplete”.

ALT_29_02 – The Analyst re-evaluates trust in a specific model	Alternate

Step	**Activity**
1	The Analyst repeatedly labels anomalies detected by one model as “false positive”; the system aggregates this feedback and displays to the Data Scientist a recommendation to review the model (or replace it with another type, including a change of threshold or architecture).

Linked functional requirements	
FREQ_194 - Model explainability	The solution must have an explainable model (interpretability). The ability to explain the detection output to the analyst. The ability to provide the analyst with a comprehensible explanation of why the anomaly was detected.
FREQ_191 - Ability to work with time-sequence data	The solution must be capable of working with time-sequential data. Suitability for data evolving over time (e.g. logs, network traffic). The ability to capture temporal patterns and anomalies in time series. Architectures such as LSTM and Transformer are an advantage.
FREQ_192 - Flexibility of input data	The solution must be capable of ensuring flexibility of input data (logs, metrics, flow data). How many data types the model can handle without the need to modify the architecture. The model must be able to work with both structured and unstructured inputs – logs, metrics, network flows.
FREQ_028 - Processing of data flows	The solution must ensure data stream processing. The main tasks of data stream processing are: • Ingestion of logs and events from multiple heterogeneous sources • Data validation, normalization and enrichment • Assignment of timestamps, identifiers and event categories • Redirection of data to appropriate indices for efficient analysis • Filtering out of redundant or irrelevant data • Implementation of pipeline rules and processing modules
FREQ_187 - Support for learning without labels	The solution must ensure support for learning without labels (unsupervised). The ability to learn from input data without the need for annotations. Not all models require labelled data. Unsupervised models are advantageous particularly with large amounts of unlabelled data.
UC_29 – Explain and analyse anomalies detected by ML models	UC_29 ensures that SOCaaS is able to explain comprehensibly the anomalies detected by machine learning models: display the context of the data flow (logs, metrics, network flows), show the time course of the signals that led to the anomaly, work with unsupervised models (without labels),

	and give the analyst an interpretation of “why the model flagged it”.

4.5.2.32 UC_30 – Manage endpoint detection and response (EDR) within SOCaaS

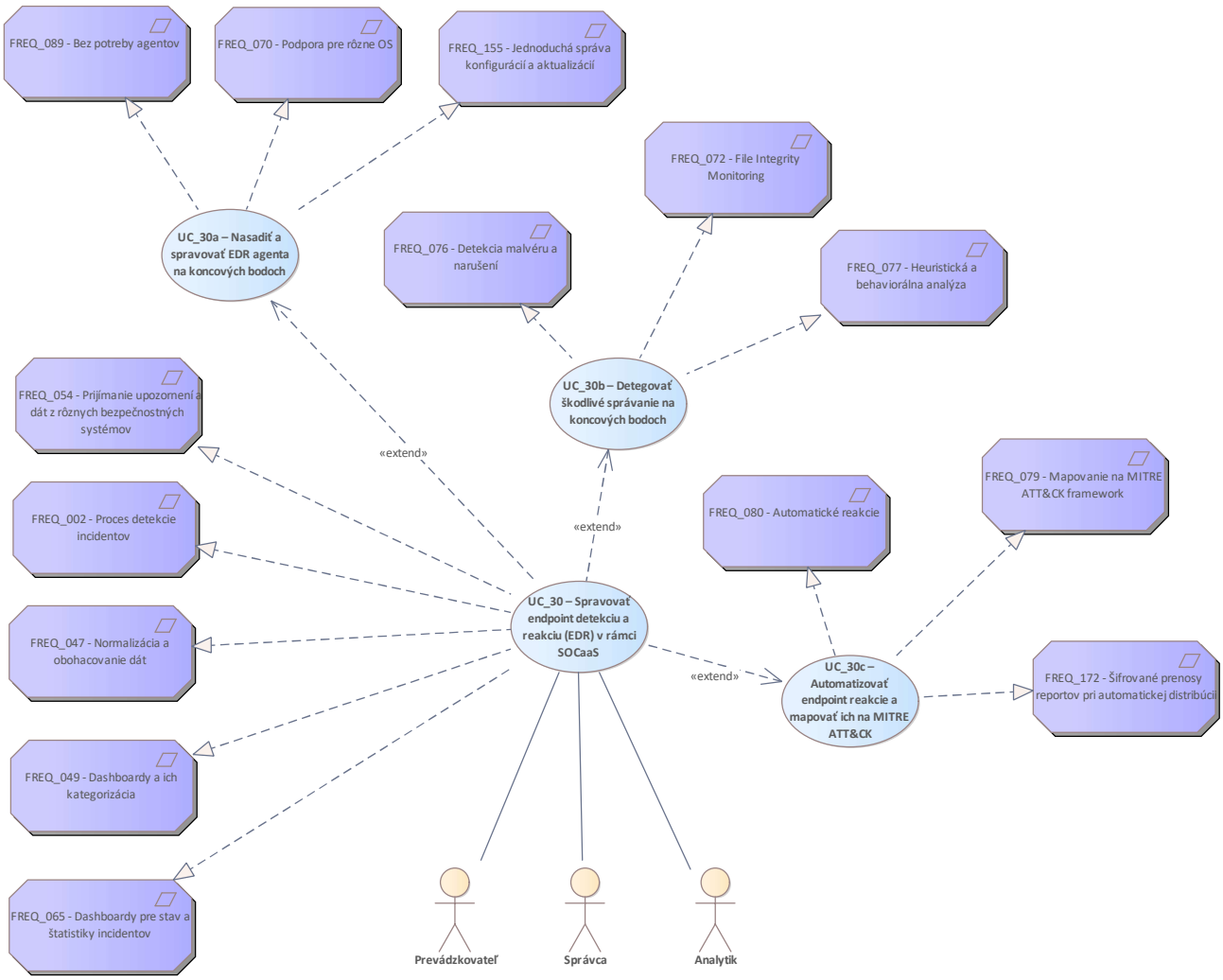


Figure 35 : UC_30 – Manage endpoint detection and response (EDR) within SOCaaS

Use case	UC_30 – Manage endpoint detection and response (EDR) within SOCaaS
Objective	UC_30 addresses how SOCaaS centrally manages the EDR/XDR layer: it receives alerts and logs from EDR/antivirus/OS tools, it normalizes and enriches them, it displays the status of incidents on dashboards, it feeds them into the overall incident detection (UC_03) and response (UC_07) process.

Precondition	An EDR/AV/host-based sensor (or integration agent) is deployed on the customer's endpoints.

Precondition	SOCaaS has integration connectors for collecting logs and alerts from EDR and other security tools.
Precondition	Basic correlation rules and heuristics for detecting endpoint incidents are defined.
Postcondition	Alerts and logs from endpoint tools are received, normalized and enriched and used in incident detection.
Postcondition	The status of endpoint incidents is clearly displayed on dashboards for analysts and management.
Postcondition	Endpoint incidents are connected to automated responses (UC_07) if permitted by policy.

Basic Path	Basic Path

Step	**Activity**
1	The Operator configures the integration of the EDR/AV tool with SOCaaS and defines which types of alerts and logs are to be sent.
2	The system receives alerts and data from various security tools (EDR, AV, OS logs) and routes them into a unified processing pipeline.
3	The pipeline transforms various logs from endpoints into a unified format, adds WHOIS/GEOIP and other context and stores them in suitable indices for correlation.
4	The detection engine (UC_03) uses normalized endpoint data in correlation rules and heuristics to identify potential incidents.
5	When an incident originating at an endpoint arises, the system displays it on the dashboards together with context (host, user, threat type, timeline).
6	The Analyst works with the endpoint incident directly in SOCaaS – filters, sorts, drills down into the details of logs and enrichment, and, where applicable, manually escalates or closes it.
7	If a playbook is defined for the given incident type, SOCaaS invokes UC_07 and offers automated or semi-automated responses (isolation, quarantine, blocking).

ALT_30_01 – Incomplete or faulty EDR integration	Alternate

Step	**Activity**
1	The EDR tool does not send all expected types of alerts or logs; the system alerts the Operator that only partial data are arriving from the given source and recommends checking the configuration or permissions.

ALT_30_02 – Surplus of endpoint alerts (alert fatigue)	Alternate

Step	**Activity**
1	After the EDR is connected, an extremely large number of low-severity alerts arrives; the Analyst and the Operator adjust the filters, correlation rules and thresholds so that only relevant incidents are displayed on the dashboards and the rest remain as low-priority events in the background.

Linked functional requirements	
FREQ_049 - Dashboards and their categorization	Data visualization using charts, filters and widgets. Dashboards (including interactive ones) and their categorization
UC_30c – Automate endpoint responses and map them to MITRE ATT&CK	UC_30c addresses automated/semi-automated responses to endpoint incidents and their mapping to MITRE ATT&CK: device isolation, file quarantine, process blocking, linking with SOAR playbooks, secure transfer of forensic data and reports.
FREQ_065 - Dashboards for incident status and statistics	A clear overview of the system status and ongoing incidents. Dashboards for incident status and statistics
UC_30b – Detect malicious behaviour on endpoints	UC_30b addresses the actual detection of malware, intrusions and suspicious behaviour on endpoints: FIM (File Integrity Monitoring),

	malware detection, heuristic/behavioural analysis.
UC_30a – Deploy and manage the EDR agent on endpoints	UC_30a addresses the deployment and life cycle of the EDR agent / sensor: installation on various OSs, updates and configuration, where applicable, use of agentless mode where an agent is not possible
FREQ_002 - Incident detection process	Ensure the incident detection process – is responsible for identifying anomalies, attack patterns and other threats by means of correlation rules and heuristic mechanisms.
FREQ_047 - Normalization and enrichment of data	Transformation of various logs into a unified format and their enrichment (e.g. WHOIS, GEOIP). Data normalization and enrichment
FREQ_054 - Receiving alerts and data from various security systems	The ability to receive alerts and data from various security sources (SIEM, EDR, CTI...). Receiving alerts and data from various security systems

Use case	UC_30a – Deploy and manage the EDR agent on endpoints
Objective	UC_30a addresses the deployment and life cycle of the EDR agent / sensor: installation on various OSs, updates and configuration, where applicable, use of agentless mode where an agent is not possible

Precondition	The EDR/endpoint tool that will be used in SOCaaS has been selected.
Precondition	Lists of endpoints and their OSs are available.
Postcondition	The EDR agent is deployed or configured (including in agentless mode) on the relevant endpoints.
Postcondition	Agent configurations and updates are managed centrally without code changes.

Basic Path	Basic Path

Step	**Activity**
1	The Administrator obtains an overview of the endpoints and their operating systems that are to be monitored.
2	The Operator prepares installation packages or policies for deploying the EDR agent on supported OSs.
3	The Administrator deploys the agent (or configures an agentless connection) on selected endpoints according to the customer's policy.
4	The Operator centrally sets the agent configuration – which logs to collect, which modules are active and how often to send data.
5	The system ensures the distribution of agent updates and configurations without the need to change code or intervene manually on each device.

ALT_30a_01 - The agent cannot be installed	Alternate

Step	**Activity**
1	On some devices (e.g. special servers, OT devices) the agent cannot be installed; the Operator activates agentless mode (e.g. log collection via syslog, WMI, API) and marks these devices in the system as “agentless”.

Linked functional requirements	
FREQ_070 - Support for various OSs	Support for installing and running the agent on various operating systems. Support for various OS (Windows, Linux, macOS)

FREQ_155 - Simple management of configurations and updates	The ability to manage the platform without the need to intervene in the code. Simple management of configurations and updates
FREQ_089 - Without the need for agents	The possibility of performing scans without the need to install an agent on devices. No need for agents (agentless)
UC_30a – Deploy and manage the EDR agent on endpoints	UC_30a addresses the deployment and life cycle of the EDR agent / sensor: installation on various OSs, updates and configuration, where applicable, use of agentless mode where an agent is not possible

Use case	UC_30b – Detect malicious behaviour on endpoints
Objective	UC_30b addresses the actual detection of malware, intrusions and suspicious behaviour on endpoints: FIM (File Integrity Monitoring), malware detection, heuristic/behavioural analysis.

Precondition	An EDR/agent is running on the endpoints (UC_30a).
Precondition	Policies are defined as to which types of behaviour are to be considered suspicious.
Postcondition	Suspicious behaviour (malware, intrusions, FIM events) is detected and mapped to incidents in SOCaaS.

Basic Path	Basic Path

Step	**Activity**
1	The EDR agent monitors changes to files, processes and system calls on the endpoint.
2	In the event of a suspicious file change (e.g. a system binary, critical configuration), it creates an FIM event and sends it to SOCaaS.
3	The EDR engine analyses process and memory behaviour in order to identify signs of malware or security breaches.
4	Heuristic and behavioural rules (or models) evaluate the combination of events and determine whether it is suspicious activity.
5	When thresholds are exceeded or a matching pattern occurs, the EDR creates an alert, which is subsequently processed in UC_30/UC_03 as an incident or incident candidate.

ALT_30b_01 - Suspicious behaviour without a clear verdict	Alternate

Step	**Activity**
1	The heuristics evaluate the activity as suspicious, but not strongly enough for automatic classification; the EDR sends a “low confidence” alert, which is displayed on the dashboard as a low-priority event for manual assessment by an analyst.

Linked functional requirements	
FREQ_077 - Heuristic and behavioural analysis	Basing detection on behaviour and heuristics rather than on signatures alone. Heuristic and behavioural analysis
FREQ_072 - File Integrity Monitoring	Real-time detection of file changes for the purposes of forensic analysis and integrity. File Integrity Monitoring (FIM)
FREQ_076 - Detection of malware and intrusions	Identification of malicious software and security breach incidents. Malware and intrusion detection
UC_30b – Detect malicious behaviour on endpoints	UC_30b addresses the actual detection of malware, intrusions and suspicious behaviour on endpoints: FIM (File Integrity Monitoring),

	malware detection, heuristic/behavioural analysis.
Use case	UC_30c – Automate endpoint responses and map them to MITRE ATT&CK
Objective	UC_30c addresses automated/semi-automated responses to endpoint incidents and their mapping to MITRE ATT&CK: device isolation, file quarantine, process blocking, linking with SOAR playbooks, secure transfer of forensic data and reports.

Precondition	Response playbooks are defined for endpoint incident types (UC_07).
Precondition	The EDR tool supports automatic actions on endpoints (isolation, quarantine, blocking).
Precondition	SOCaaS has a mapping of activities to MITRE ATT&CK.
Postcondition	Endpoint incidents are mapped to ATT&CK TTPs and used in reports and CTI.
Postcondition	Automated responses are performed securely and auditably, including the secure transfer of reports and forensic data.

Basic Path	Basic Path

Step	**Activity**
1	The Analyst or the system identifies an endpoint incident for which an automated response is appropriate (e.g. malware on a workstation).
2	SOCaaS assigns the relevant MITRE ATT&CK techniques to the incident and displays them in the incident detail.
3	The Analyst launches a playbook, or the system automatically evaluates the conditions and performs the action (device isolation, file quarantine, process blocking).
4	The EDR sends confirmation of the result of the action back to SOCaaS, where the incident status is updated and the steps performed are recorded.
5	The system generates a summary report on the course of the incident and the responses performed and sends it securely via an encrypted channel (e.g. to the manager or the customer).

ALT_30c_01 - The automated response is blocked by policy	Alternate

Step	**Activity**
1	The Manager has set a policy that fully automatic isolation is not permitted on critical servers; for such an incident, SOCaaS offers only a “semi-automatic” action requiring explicit confirmation by an analyst.

Linked functional requirements	
FREQ_172 - Encrypted transfers of reports during automatic distribution	Secure transfer of sensitive data via encrypted channels. Encrypted transfers of reports during automatic distribution
FREQ_079 - Mapping to the MITRE ATT&CK framework	Classification of detected activities and techniques within the MITRE ATT&CK framework for the purposes of TTP analyses. Mapping to the MITRE ATT&CK framework
FREQ_080 - Automatic responses	Automatic response to incidents, such as endpoint isolation or file quarantine. Automatic responses (e.g. device isolation, file quarantine)
UC_30c – Automate endpoint responses and map them to MITRE ATT&CK	UC_30c addresses automated/semi-automated responses to endpoint incidents and their mapping to MITRE ATT&CK: device isolation, file quarantine, process blocking, linking with SOAR playbooks,

	secure transfer of forensic data and reports.

4.5.2.33 UC_31 – Provide a workspace for manual threat analysis (CTI workspace)

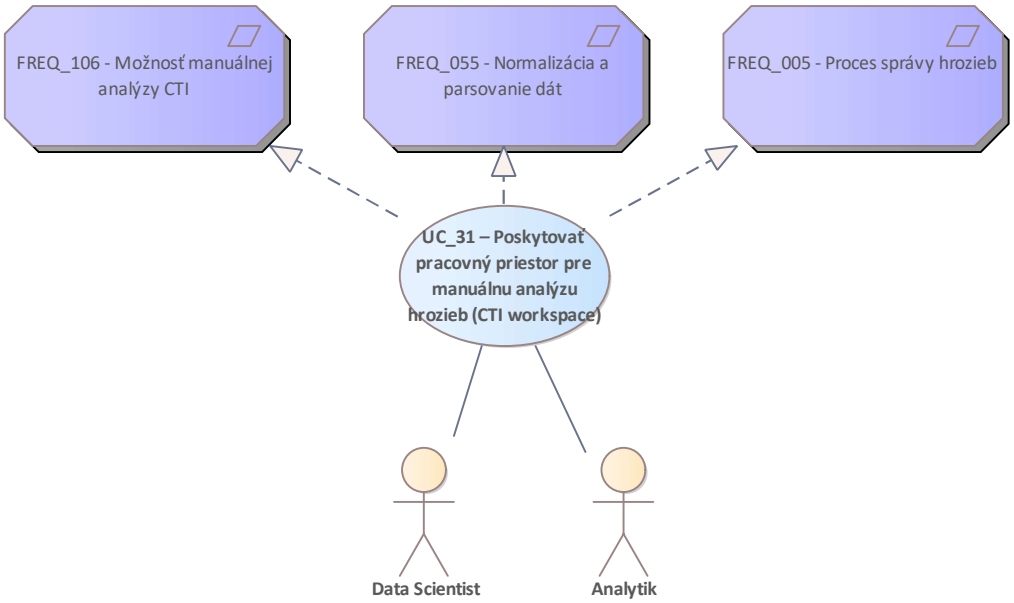


Figure 36 : UC_31 – Provide a workspace for manual threat analysis (CTI workspace)

Use case	UC_31 – Provide a workspace for manual threat analysis (CTI workspace)
Objective	UC_31 ensures that SOCaaS provides a dedicated workspace for manual threat analysis (CTI): the analyst is able to open a specific threat, IOC or campaign, they see normalized data from multiple CTI sources, they have tools for manual investigation, filtering and adding comments.

Precondition	The CTI pipeline is functional (UC_25) – CTI data have been downloaded and technically processed.
Precondition	The user has a role that allows them to work with CTI data (reading, comments, labels).
Postcondition	Manually added analyses and comments from the analyst exist for selected threats / IOCs.
Postcondition	The results of the manual analysis are stored so that they can be used in incidents, hunting cases and further CTI processing.

Basic Path	Basic Path

Step	**Activity**
1	The Analyst opens the “CTI workspace” module and searches for the threat, IOC or campaign they want to analyse.
2	From the CTI repositories, the system loads all available information on the given threat – IOCs, descriptions, source feeds – and displays it in a unified, normalized format.

3	The Analyst uses the available tools (filtering, pivoting on IOCs, linking to incidents and logs) to manually examine the data and the context of the threat.
4	The Analyst adds their own comments, relevance assessment and recommendations and, where applicable, flags the threat for specific customers/segments.
5	The system saves the result of the manual analysis as part of the CTI record and makes it available in related incidents, hunting cases and reports.

ALT_31_01 – CTI data for the threat are incomplete or inconsistent	Alternate

Step	**Activity**
1	In step 2, the Analyst finds that the information on the threat from different sources is contradictory or incomplete; they mark the record as “requires completion”, add manual notes, and the CTI specialist later adjusts the sources or the scoring.

ALT_31_02 – The Analyst opens the CTI workspace directly from an incident	Alternate

Step	**Activity**
1	While investigating an incident in UC_06, they click on the linked threat/IOC; the system opens the CTI workspace already pre-filtered to the relevant IOCs and campaign so that the analyst does not have to search manually.

Linked functional requirements	
FREQ_005 - Threat management process	Ensure the threat management process (Threat Intelligence) – enables systematic work with threat information.
FREQ_055 - Normalization and parsing of data	Transformation of input data into a standardized format for further processing. Data normalization and parsing
FREQ_106 - Ability to perform manual CTI analysis	An interface and tools for security analysts to examine data manually. Possibility of manual CTI analysis

4.5.2.34 UC_32 – Monitor vulnerability scanning and synchronize the status of related incidents

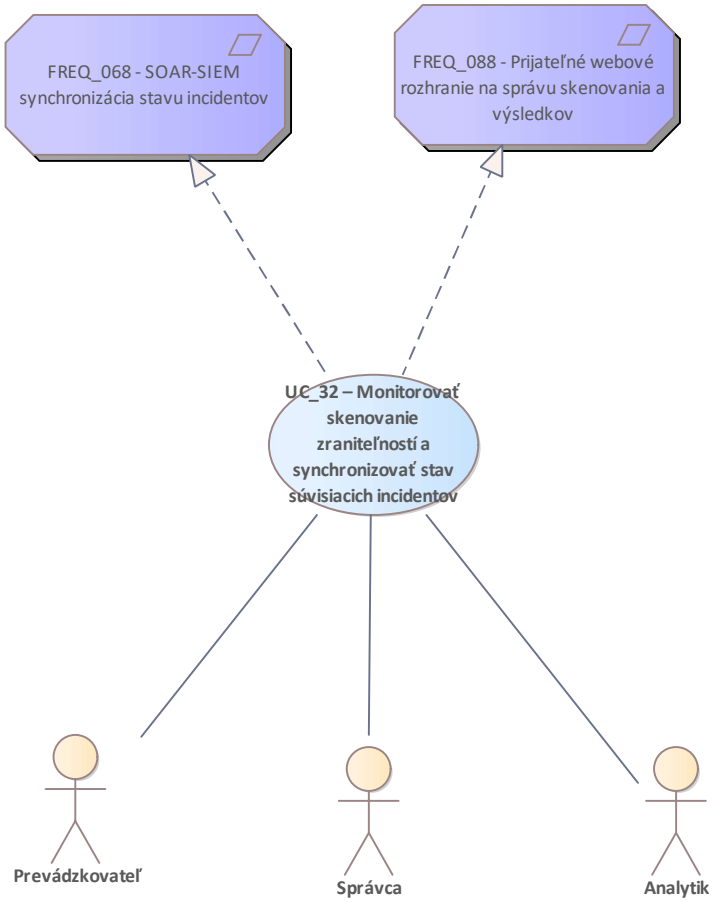


Figure 37 : UC_32 – Monitor vulnerability scanning and synchronize the status of related incidents

Use case	UC_32 – Monitor vulnerability scanning and synchronize the status of related incidents
Objective	UC_32 addresses the operational view of vulnerability scanning and related incidents: it provides a web interface for managing scans and results (on top of the functions from UC_23), it displays which findings have already been converted into incidents, and it ensures synchronization of incident status between SIEM and SOAR so that analysts have a consistent status everywhere.

Precondition	Vulnerability scanning is technically set up and running according to UC_23.
Precondition	An integration connection between SIEM and SOAR exists (according to UC_24).
Precondition	The user has permission to access the “Scanning and incidents” module.
Postcondition	Users have a clear web interface available through which they can see scans, their status and results.
Postcondition	Scan findings that are handled as incidents have their status synchronized between SIEM and SOAR (e.g. “open / in progress / resolved”).
Basic Path	Basic Path

Step	Activity
1	The Analyst or Administrator opens the “Scanning and incidents” module in the SOCaaS web interface and sees an overview of launched and scheduled scans.
2	The user selects a specific scan and displays its detail – scope, launch time, status, number of vulnerabilities found and basic statistics.
3	In the same interface, the user displays the list of scan findings, with each finding linked to its current status (new, in progress, closed) and information on whether an incident already exists for it in SIEM/SOAR.
4	The system displays for which findings an incident has been created in SOAR and automatically verifies the status of this incident in SIEM (e.g. “open”, “assigned”, “resolved”).
5	When the incident status changes in SOAR (e.g. after a playbook is executed), the UC_32 module also updates the incident status in SIEM via the integration layer so that analysts and management have consistent information everywhere.
6	From the GUI, the Analyst can filter findings by incident status (e.g. “critical findings without an incident” or “incidents in progress”) and thereby prioritize the next steps.

ALT_32_01 – An incident exists in SOAR, but SIEM does not know its status	Alternate

Step	Activity
1	In step 4, the system finds that an incident exists in SOAR for a specific finding, but there is no corresponding record in SIEM (e.g. an integration error or a historical incident); the system marks the finding as “inconsistent” and alerts the Operator to check the integration rules or align the incidents manually.

ALT_32_02 – The web interface cannot display some of the scans or results	Alternate

Step	Activity
1	When loading the overview in step 1 or 2, an API error occurs (e.g. the scanner is unavailable); the UI displays a warning indicating which scans/results could not be loaded and keeps the rest of the overview functional, while allowing the view to be refreshed later.

Linked functional requirements	
FREQ_088 - Acceptable web interface for managing scanning and results	Usability of the GUI for managing scans and results. Acceptable web interface for managing scanning and results
FREQ_068 - SOAR-SIEM synchronization of incident status	Linking of incident status between SIEM and SOAR for up-to-date visibility. SOAR-SIEM incident status synchronization

4.6 Business architecture

4.7 Business Viewpoints

4.7.1 ArchiMate 2 View

4.7.2 Asset inventory in SOCaaS – business view

4.7.3 Identification of the design owners of end services (UX)

Name and surname	Job position	Telephone	E-mail

Table 7 : Identification of the design owners of end services (UX)

4.7.4 Specification of the end customer's needs

Description of the user need	Reference to the research from which the need arises	Proposed solution to the need	Quantity	Impact on customer experience	Technical complexity	Business benefit	Priority

Table 8 : Specification of user needs

Explanation of **Chyba! Neznámy argument prepínača.** (table cells from left to right):

- Description of the user need: a statement of a specific user need in the first person that does not describe the solution.
- Reference to the research from which the need arises: describes the specific interviews with respondents in which the requirement was expressed.
- Proposed solution to the need: one of the possible solutions that has been quantitatively validated.
- Quantity: output of the questionnaire in which users rated the proposed solution to the need on a scale from 1 to 10, where 1 is "It is not important to me" and 10 is "It is very important to me."
- Impact on customer experience: it is determined by a combination of qualitative and quantitative information from the research and is rated on the MoSCoW scale[1].
- Technical complexity is the input from the technical solution provider.
- Business benefit is the input from the business sponsor.
- Priority is the resulting importance of implementing this solution after considering the columns Impact on customer experience, Technical complexity and Business benefit.

4.7.5 User interface prototype design

4.7.6 Screen visual design

5 Data architecture

The following chapters describe the design of the entities of the module's persistence layer.
Each entity contains:

- Definition of the entity
- List of basic attributes

[1] https://en.wikipedia.org/wiki/MoSCoW_method

- Attribute name
- Data type
- Mandatory – the attribute must be populated (not null)
- Description – explanation of what the attribute represents, if this is not obvious from the name, or a description of the validation

6 Application architecture

The application architecture of the CNM module is fully modelled in the Sparx Enterprise Architect (EA) tool in ArchiMate 3.0 notation.

The following chapter shows the application architecture diagram with descriptions of the objects, which were generated from EA as an output.

All descriptions of application architecture objects, such as application services, application components, functional components and data objects, are part of the model in EA, recorded in the object's "Description" (note).

Since the output for the modelling of the DNR application architecture is generated from EA, where a given application architecture object is used more than once, its description is identical.

6.1 Application architecture

6.2 Aplication Viewpoints

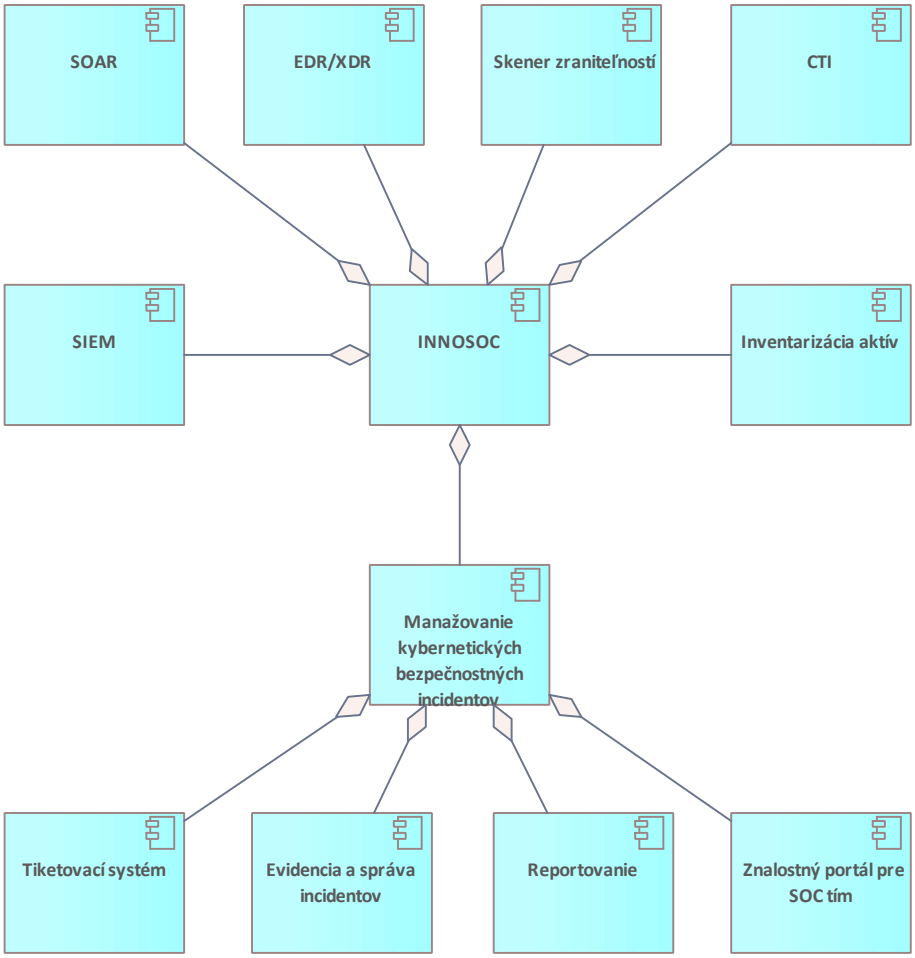


Figure 38 : Aplication Viewpoints Components

The diagram presents a high-level (HL) list of the solution components.

Description of the elements used:

CTI

CTI (Cyber Threat Intelligence) is a key building block of the SOCaaS architecture, as it provides the capability to enrich detection rules, correlation mechanisms and playbooks with external threat information – including IOCs, TTPs, attacker profiling, campaigns and related vulnerabilities. In the environment of the multi-customer SOCaaS service, it plays a role in:

• Distribution and management of threat feeds across tenants,
• Automatic enrichment of SIEM and EDR data with known threats,
• Creation of attack timelines and mapping to MITRE ATT&CK techniques,
• Support of threat hunting and forensic analysis processes,
• Prioritization of incidents according to threat context.

For the solution design, open-source platforms were chosen that cover various aspects of CTI – from IOC collection through visualization to STIX/TAXII compatibility.

EDR/XDR

EDR (Endpoint Detection and Response) and XDR (Extended Detection and Response) are the fundamental technological pillars for threat detection at the endpoint level. In the SOCaaS environment, they serve for the collection and correlation of, and response to, events that would otherwise escape traditional perimeter detections. XDR extends the EDR concept with the ability to integrate data from networks, clouds, e-mail systems and other layers.

Incident recording and management

Incident recording and management is a core element of every security operation (SOC). This subcomponent ensures systematic documentation, tracking and management of security events and incidents throughout their life cycle – from detection through analysis to closure. A high-quality recording system must enable teams to collaborate in a coordinated manner, document steps, record related artefacts (observables) and link incidents

with relevant threats, systems or historical cases. Ideally, it also supports advanced data enrichment, auditing, multi-tenancy and the possibility of retrospective investigation of incidents.

INNOSOC

Asset inventory

Asset inventory is a fundamental pillar of every security solution. It enables the SOC team to obtain an overview of all devices, system entities, applications and virtual or cloud resources that make up the organization's infrastructure. In the SOCaaS environment, it plays an even more significant role, as it enables centralized management, categorization, monitoring and tracking of the security status of the infrastructure across different customers.

Management of cybersecurity incidents

Effective handling of cyber incidents is a key pillar of the SOCaaS architecture. It includes not only the recording of events itself, but a comprehensive cycle of their registration, management, response, reporting and knowledge retention. For the SOC team it is essential that incidents are managed systematically, with an emphasis on traceability, analytical work, coordination and knowledge support. An effective and coordinated response to security incidents is an indispensable pillar of every SOCaaS solution. Within the INNOSOC architecture, four subcomponents have been identified which together form an integrated system for the management, registration, documentation and handling of incidents:

1. Ticketing system
2. Incident registration and management
3. Reporting
4. Knowledge portal for the SOC team

Reporting

The reporting component within the SOCaaS architecture serves as the central means for aggregating, visualizing and distributing security information. Its role is to provide both regular and ad hoc outputs on the state of cybersecurity for various target groups – from SOC analysts, through team leaders, to the management of the organization or clients. Such a reporting system must cover:

• Generation of metrics for incidents, alerts, SLAs and KPIs,
• Overviews of trends over time and assessment of the performance of security measures,
• Adaptation of content according to the level of detail (technical vs. strategic),
• Automation and scheduling of the distribution of outputs,
• Retrospective analysis and archiving of historical data.

SIEM

SIEM is the heart of the SOCaaS solution. It provides aggregation, processing, correlation and analysis of security events from various sources in real time. It enables analysts to identify anomalies, generate alerts and escalate incidents, while increasing visibility over the client's infrastructure.

Vulnerability scanner

The vulnerability scanner is a fundamental building block for ensuring a proactive approach in the SOCaaS architecture. It makes it possible to regularly identify weaknesses in the infrastructure, assess the status of updates, configurations and settings, and provide information for further prioritization and planning of corrective measures. The scanner outputs are often a source for SOAR systems, SLA metrics and reporting, as well as for assessing the client's security level.

SOAR

SOAR (Security Orchestration, Automation and Response) is the component responsible for automating incident responses, orchestrating security tools and supporting analysts' decision-making. In SOCaaS, it is a fundamental element enabling the scaling of response capacities without the need to increase staff numbers. It is indispensable especially when processing large volumes of alerts, where it reduces the MTTR and MTTC metrics.

Ticketing system

The ticketing system serves as the central communication and organizational channel for the management of security incidents throughout their entire life cycle. It enables the creation, assignment and tracking of tasks and communication between analysts, while supporting audit, SLA monitoring, escalation and notifications.

Knowledge portal for the SOC team

A knowledge portal for the SOC team is an important building block of every modern security operations centre. It serves as a central repository of know-how, playbooks, procedures, recommendations, analytical guides and often also historical lessons learned from incidents. The knowledge portal:

• Increases the efficiency and consistency of incident response,
• Enables rapid onboarding of new team members,
• Supports the standardization of processes and methodologies,
• Enables the team to retain and share proven procedures and its own analytical experience,
• Strengthens the ability to learn collectively from incidents and to build internal "threat intel" knowledge.

6.3 Aplication Viewpoints DELETE OLD

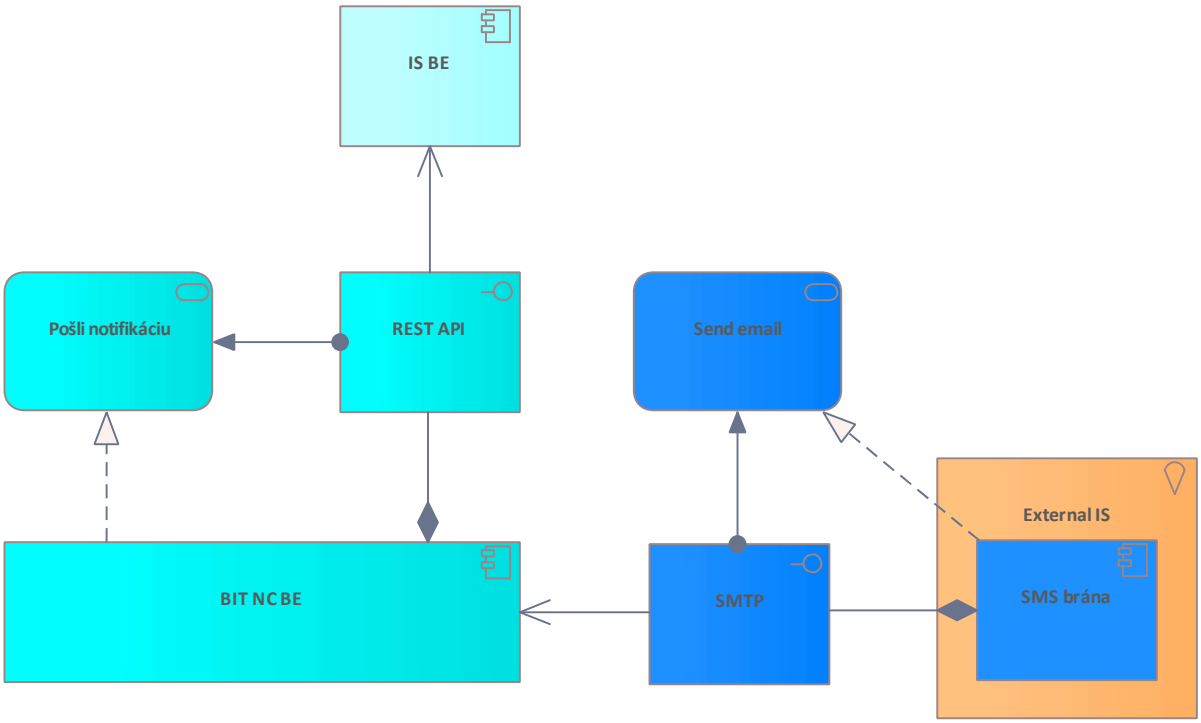


Figure 39 : Application Cooperation Viewpoint Empire and BIT NC

The diagram presents the integration of the IS BE with the notification-sending services of the BIT NC module using REST API, and the integration of BIT NC with the mail server using SMTP and its e-mail sending service.

Description of the elements used:

Send notification
Represents the HTTP POST, GET, PUT and DELETE methods of the service for sending a notification.

Send email
Service for sending an e-mail.

External IS
External information systems.

BIT NC BE
BIT NC BE represents the BE module of BIT Notify Center, which provides central management of notifications together with the corresponding BIT NC FE.

IS BE
The main BE component of the IS, which uses the services of NFQES Enterprise IAM and BIT NC and provides an interface for the IS FE module.

REST API
Represents the REST API of the BIT NC module.

SMTP
SMTP service of the mail server. Port 587 for Secure SMTP, alternatively 465, and in the case of an internal mail server port 25 is also possible (not recommended).

SMS gateway
Represents an e-mail server connected using a username and password via SMTP.

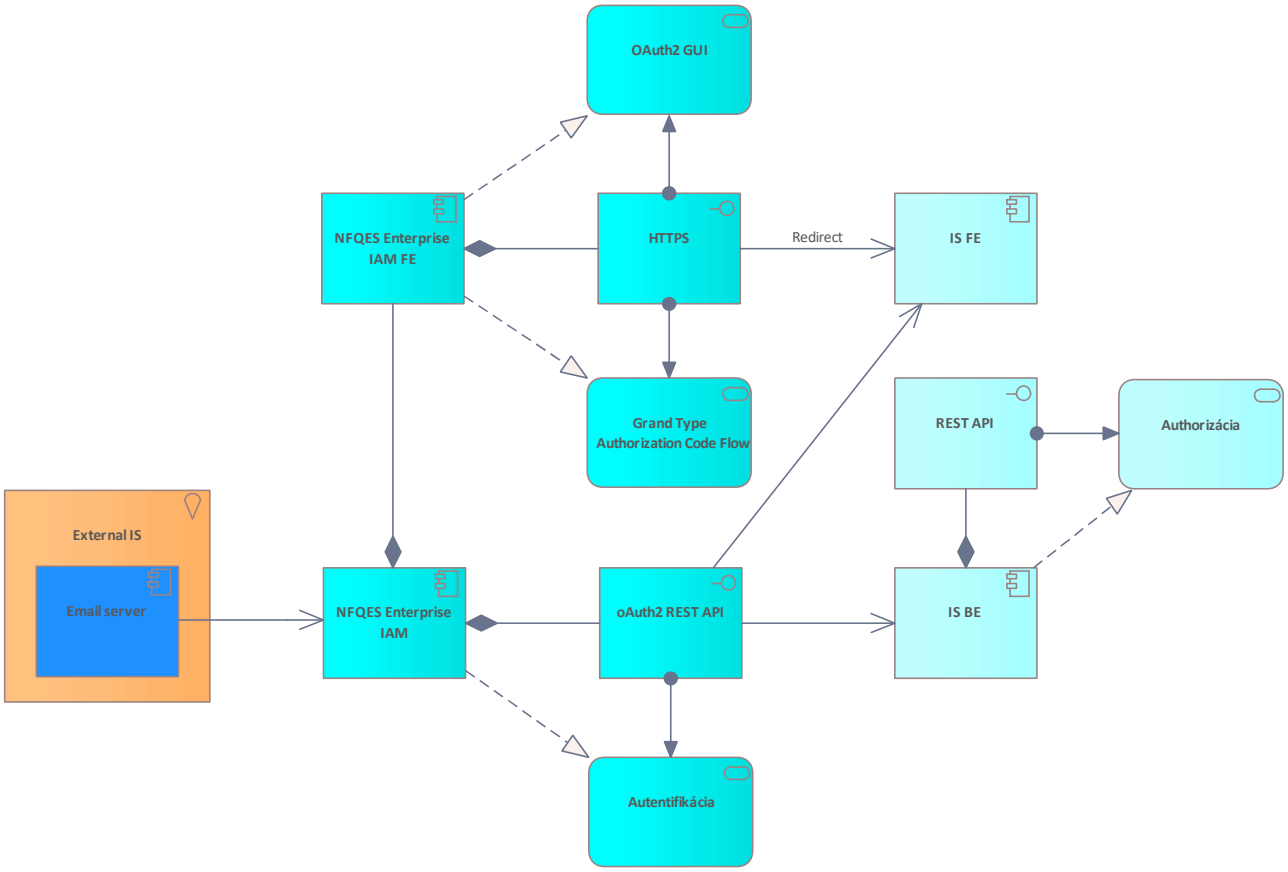


Figure 40 : Application Cooperation Viewpoint IS and BIT IAM

The diagram presents the mutual integration between the IS being built and the existing NFQES Enterprise IAM module.

Description of the elements used:

Authentication
Represents the HTTP POST, GET, PUT and DELETE methods of the service for Authentication.

Authorization
Represents the authorization application service.

Grand Type Authorization Code Flow
Grand Type - Authorization Code Flow.
Note - The second flow, which we are not implementing, is the Pasword Flow (signing in directly from the app - the client id and password must not leak).

OAuth2 GUI
Represents the HTTP pages of NFQES Enterprise IAM.

External IS
External information systems.

IS BE
The main BE component of the IS, which uses the services of NFQES Enterprise IAM and BIT NC and provides an interface for the IS FE module.

IS FE
The main FE component of the IS, which uses the services of IS BE and NFQES Enterprise IAM.

NFQES Enterprise IAM
NFQES Enterprise IAM represents the BIT Identity Access Management module, which provides authentication and role management with the corresponding NFQES Enterprise IAM FE.

NFQES Enterprise IAM FE
IAM FE represents the FE submodule of the NFQES Enterprise IAM FE module, which provides the basic screens required for the authentication flow.

HTTPS
Represents the HTTPS interface of the NFQES Enterprise IAM module.

oAuth2 REST API
Represents the oAuth2 REST API of the NFQES Enterprise IAM module.

REST API
Represents the REST API of the IS BE module.

Email server
Represents the accounting program into which data will be imported in an agreed format. This is a process integration.

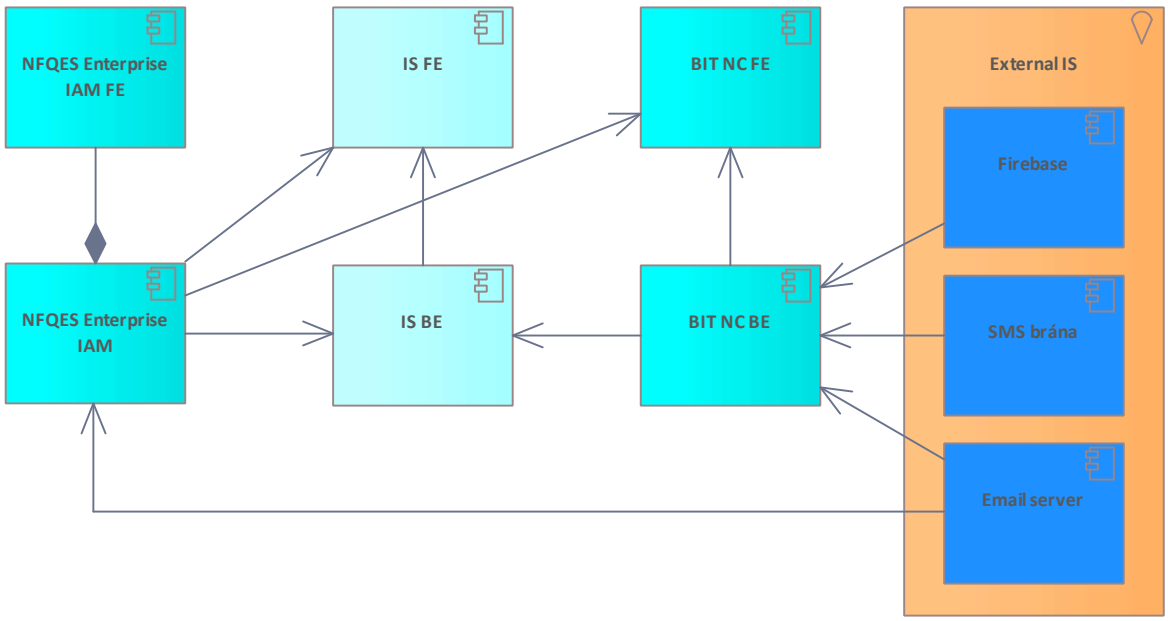


Figure 41 : Application Structure Viewpoint

The diagram presents a high-level architectural scheme of the interconnection of components.

Description of the elements used:

External IS
External information systems.

BIT NC BE
BIT NC BE represents the BE module of BIT Notify Center, which provides central management of notifications together with the corresponding BIT NC FE.

BIT NC FE
BIT NC FE represents a standalone FE module using the BIT NC module.

IS BE
The main BE component of the IS, which uses the services of NFQES Enterprise IAM and BIT NC and provides an interface for the IS FE module.

IS FE
The main FE component of the IS, which uses the services of IS BE and NFQES Enterprise IAM.

NFQES Enterprise IAM
NFQES Enterprise IAM represents the BIT Identity Access Management module, which provides authentication and role management with the corresponding NFQES Enterprise IAM FE.

NFQES Enterprise IAM FE
IAM FE represents the FE submodule of the NFQES Enterprise IAM FE module, which provides the basic screens required for the authentication flow.

Email server
Represents the accounting program into which data will be imported in an agreed format. This is a process integration.

Firebase
Represents the accounting program into which data will be imported in an agreed format. This is a process integration.

SMS gateway
Represents an e-mail server connected using a username and password via SMTP.

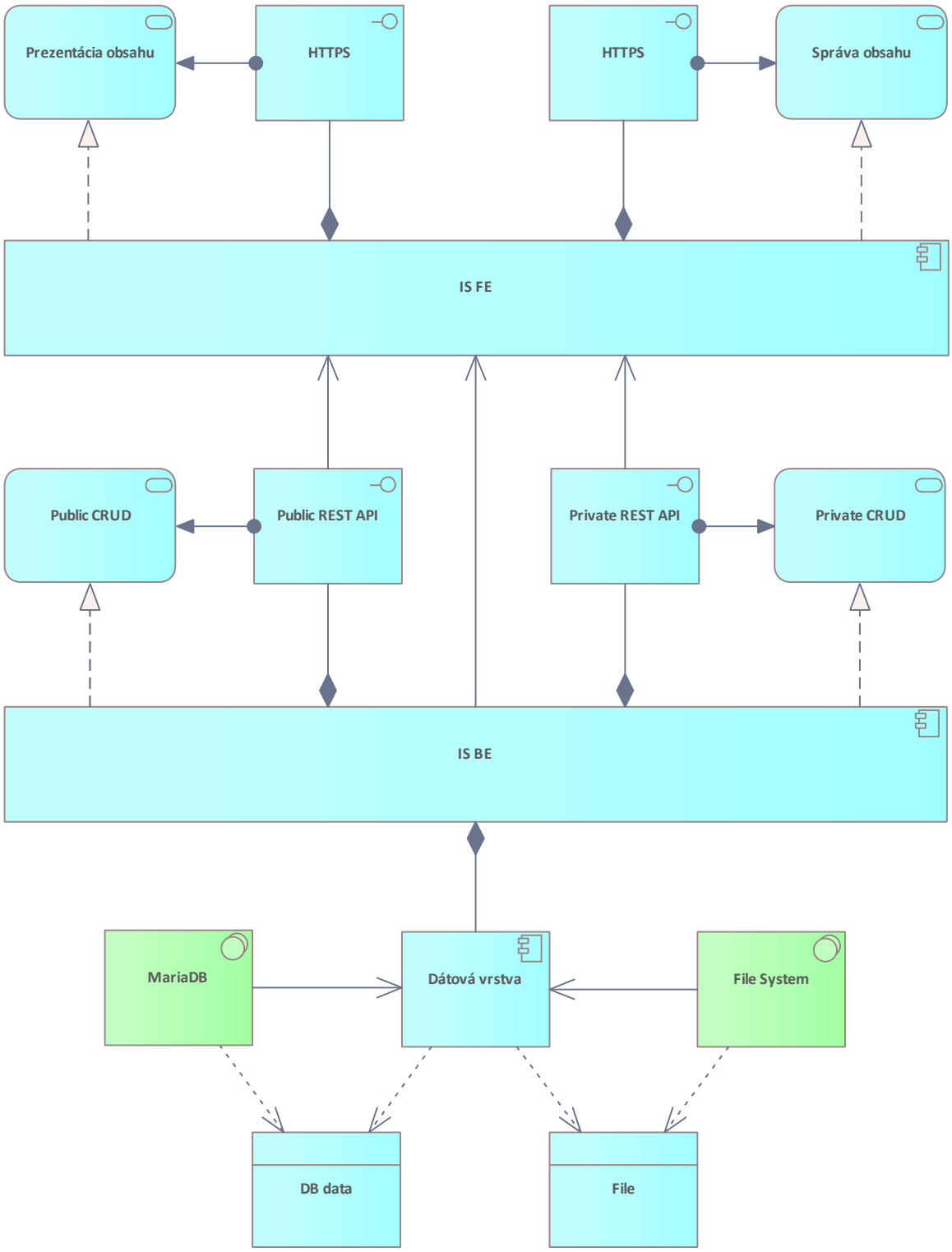


Figure 42 : Implementation and Deployment Viewpoint with bussines objects - Empire

The diagram presents the application architecture of the IS and its interconnection by means of REAST API interfaces. It also depicts the use of the persistence layer in the form of the relational SQL database MariaDB and the file system.

Description of the elements used:

Content presentation
Represents the content presentation application service.

Private CRUD
Represents the private HTTP POST, GET, PUT and DELETE methods of the service.

Public CRUD
Represents the public HTTP POST, GET, PUT and DELETE methods of the service.

Content management
Represents the content management application service.

DB data
Represents database data.

File
A standard file representing any required data such as images, documents and others.

File System
Local file system for standard work with files.

MariaDB
A relational SQL database providing the persistence layer for data other than files.

Data layer
Represents the component of the Empire BE module that performs operations on the persistence layer in the form of the file system, the database and possibly others.

IS BE
The main BE component of the IS, which uses the services of NFQES Enterprise IAM and BIT NC and provides an interface for the IS FE module.

IS FE
The main FE component of the IS, which uses the services of IS BE and NFQES Enterprise IAM.

HTTPS
Represents the HTTPS interface for accessing content.

HTTPS
Represents the HTTPS interface for content management.

Private REST API
Represents the private REST API of the WCM module.

Public REST API
Represents the public REST API of the WCM module.

6.4 Solution design of application services

6.4.1 Existing application interfaces

The CNM project envisages retaining the services of the eNotify module - SendNotification, SendNotificationUsingTemplate. For this purpose, a LegacyService service will be created, implementing an interface that receives asynchronous messages in the structure of SendNotification and SendNotificationUsingTemplate requests. A more detailed description is given in the following subchapter.

6.4.2 Discontinued application interfaces

6.4.3 New application interfaces

6.5 Integrations between systems

6.5.1 List of existing integrations without change

6.5.2 List and description of proposed integrations

6.5.3 List and description of required integration interfaces

6.5.4 Design of the application interfaces provided by the system (module)

7 Technological architecture

The chapter describes the physical architecture and defines the requirements for the test and production environments.

The proposed operation is planned on two identical environments differing only in the amount and type of resources:
- **PROD** environment – production environment with real data and operation of the public portal.
- **TEST** environment – testing, training, pre-production environment. The entire TEST environment is accessible via VPN for privileged users.

7.1 Technological architecture

7.2 Technology Viewpoint INNOSOC Common

7.3 Technology Viewpoint Asset Inventory

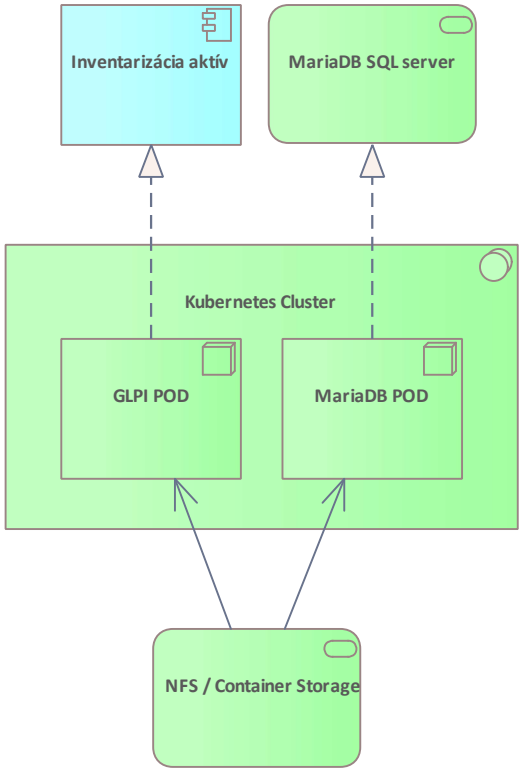


Figure 43 : Technology Usage Viewpoint Asset Inventory alt 1

The diagram shows a simplified assignment of the individual PODs to the corresponding application components.
Alt 1 represents the use of the database directly within the module setup. The design envisages a containerized version of the SQL server.

Description of the elements used:

Asset inventory
Asset inventory is a fundamental pillar of every security solution. It enables the SOC team to obtain an overview of all devices, system entities, applications and virtual or cloud resources that make up the organization's infrastructure. In the SOCaaS environment, it plays an even more significant role, as it enables centralized management, categorization, monitoring and tracking of the security status of the infrastructure across different customers.

NFS / Container Storage
Represents the NFS service or another Container Storage

Kubernetes Cluster
Represents the Kubernetes Cluster

MariaDB SQL server
Represents the technology service of the relational MariaDB SLQ database.

GLPI POD
Represents the GLPI POD for asset inventory.

MariaDB POD
Represents the MariaDB POD for the relational SQL DB.

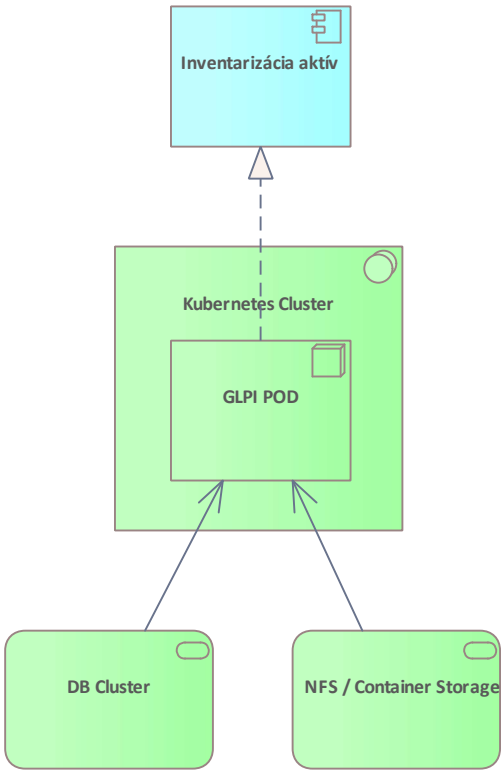


Figure 44 : Technology Usage Viewpoint Asset Inventory alt 2

The diagram shows a simplified assignment of the individual PODs to the corresponding application components. Alt 2 represents the use of the database directly in the form of a centralized DB cluster.

Description of the elements used:

Asset inventory
Asset inventory is a fundamental pillar of every security solution. It enables the SOC team to obtain an overview of all devices, system entities, applications and virtual or cloud resources that make up the organization's infrastructure. In the SOCaaS environment, it plays an even more significant role, as it enables centralized management, categorization, monitoring and tracking of the security status of the infrastructure across different customers.

DB Cluster
Represents the clustered relational database service

NFS / Container Storage
Represents the NFS service or another Container Storage

Kubernetes Cluster
Represents the Kubernetes Cluster

GLPI POD
Represents the GLPI POD for asset inventory.

7.4 Technology Viewpoint Kubernetes

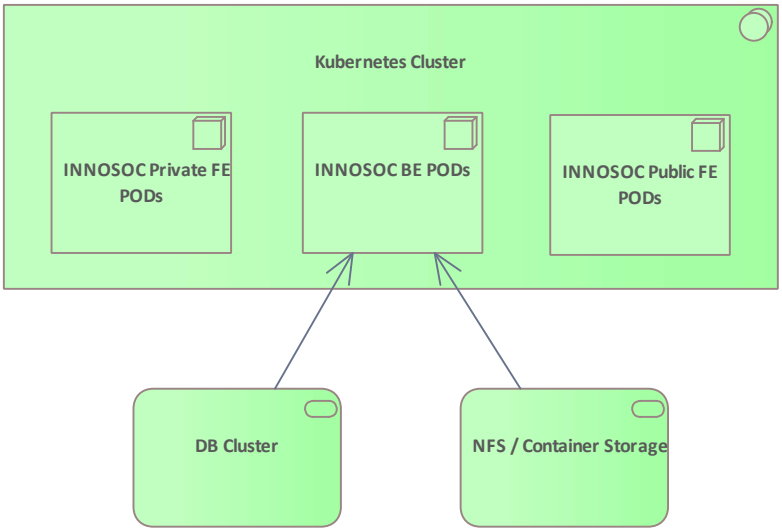


Figure 45 : Technology Usage Viewpoint CNM and Containers maping

The diagram shows the assignment of the individual PODs to the corresponding application components.

Description of the elements used:

DB Cluster
Represents the clustered relational database service

NFS / Container Storage
Represents the NFS service or another Container Storage

Kubernetes Cluster
Represents the Kubernetes Cluster

INNOSOC BE PODs
Represents the CNM BE PODs

INNOSOC Private FE PODs
Represents the CNM FE Private PODs

INNOSOC Public FE PODs
Represents the CNM FE Public PODs

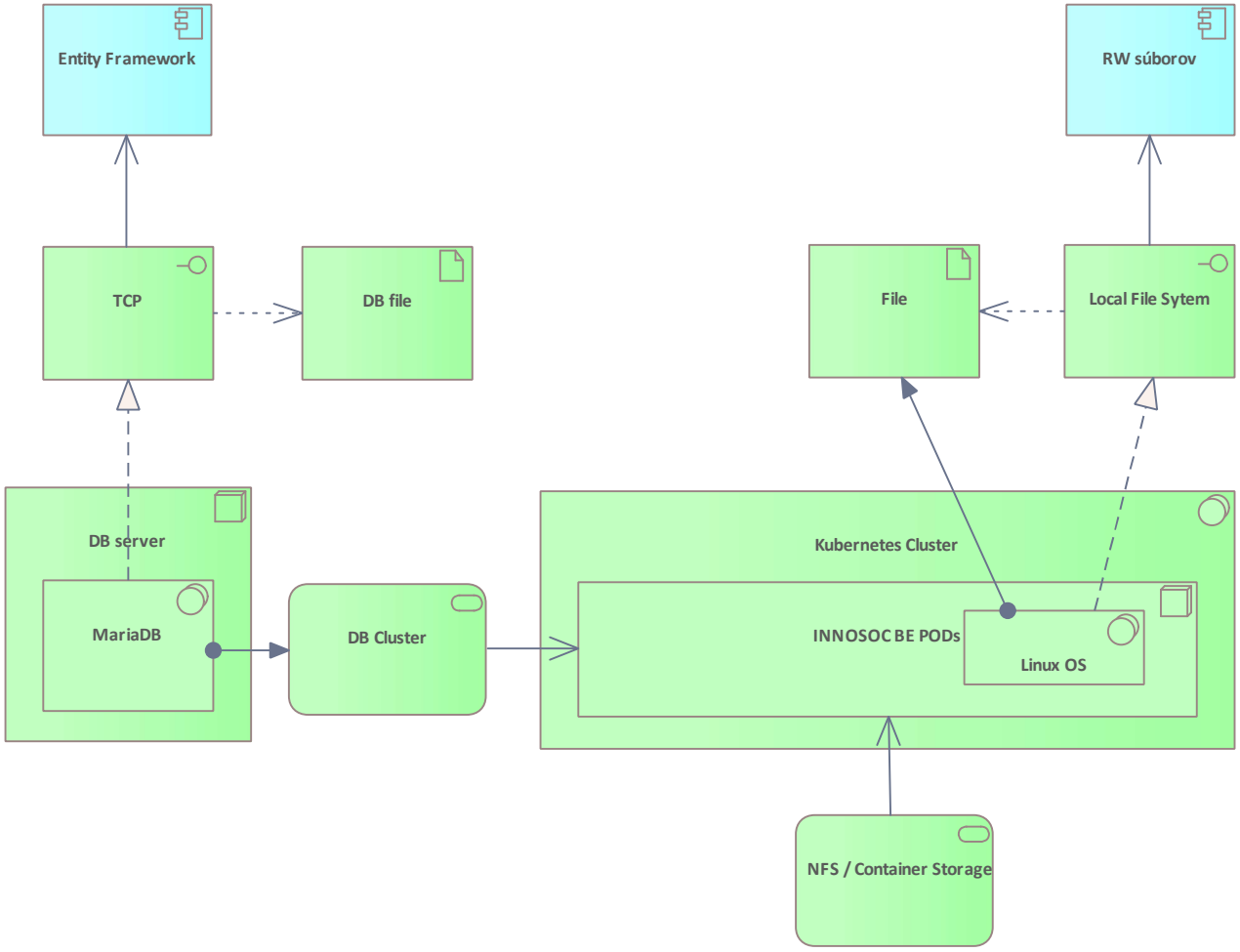


Figure 46 : Technology Usage Viewpoint CNM BE, DB and File System

The diagram represents and describes in more detail the use of the persistence layer for the needs of the BIT NC and CNM BE module by means of its Data layer component, which handles work with PostgreSQL and the Local File System.

Description of the elements used:

DB Cluster
Represents the clustered relational database service

NFS / Container Storage
Represents the NFS service or another Container Storage

DB file
DB file - a file representing the MariaDB database file

DB server
Represents the MariaDB server

File
File - a file representing data such as attachments, images, documents and others

Entity Framework
Represents the Entity Framework application module

File RW
Represents the RW module for files

Local File Sytem
Local file system providing work with files, implemented by the operating system under which the BE module is run

TCP
Represents client-server communication using the TCP protocol

Kubernetes Cluster
Represents the Kubernetes Cluster

INNOSOC BE PODs
Represents the CNM BE PODs

Linux OS

MariaDB
Represents the MariaDB relational database

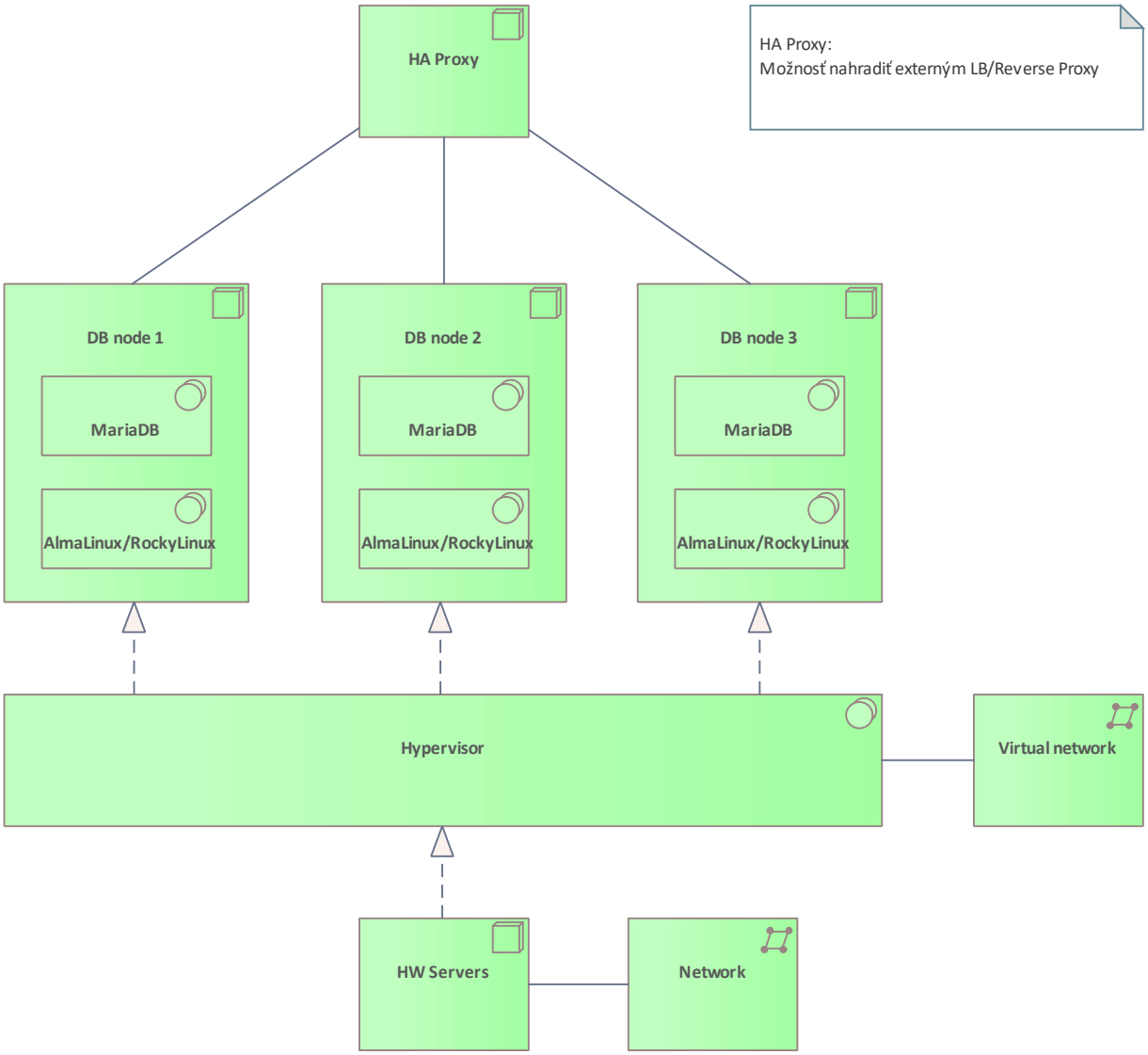


Figure 47 : Technology Viewpoint DB Cluster

The diagram shows the technological architecture of the PostgreSQL Cluster.

Description of the elements used:

DB node 1
Represents Node 2 of the MariaDB Galera Cluster

DB node 2
Represents Node 2 of the MariaDB Galera Cluster

DB node 3
Represents Node 3 of the MariaDB Galera Cluster

HA Proxy
Represents an internal reverse proxy built on HA Proxy

HW Servers
Represents the HW server infrastructure

Hypervisor
Represents a Type 1 or Type 2 hypervisor

Network
Represents the network/subnet of the HW infrastructure

Virtual network
Represents the virtual network/subnet of the hypervisor

Note
HA Proxy:
Can be replaced by an external LB/Reverse Proxy

AlmaLinux/RockyLinux
Represents the AlmaLinux/RockyLinux Linux distribution

MariaDB
Represents the MariaDB relational database

AlmaLinux/RockyLinux
Represents the AlmaLinux/RockyLinux Linux distribution

MariaDB
Represents the MariaDB relational database

AlmaLinux/RockyLinux
Represents the AlmaLinux/RockyLinux Linux distribution

MariaDB
Represents the MariaDB relational database

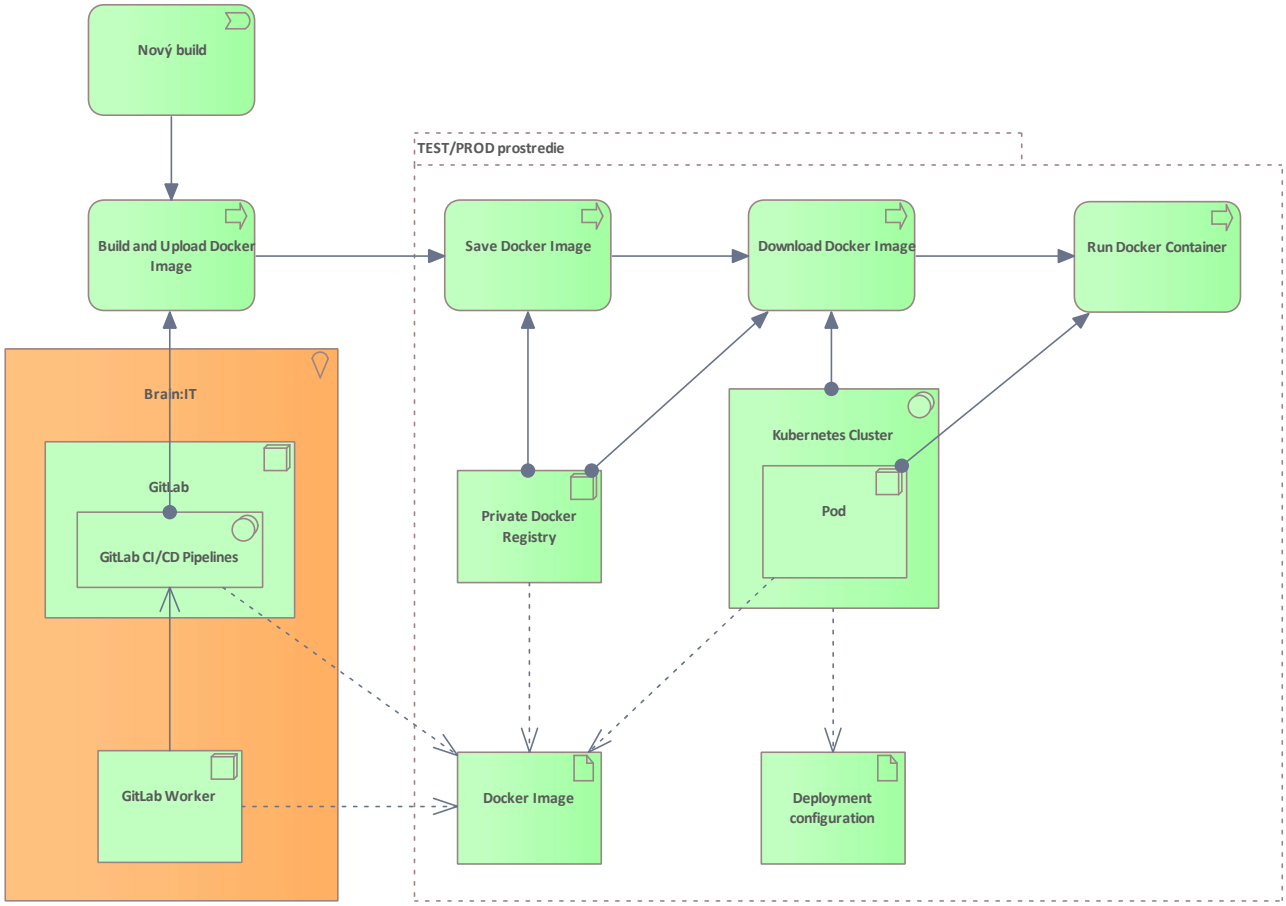


Figure 48 : Technology Viewpoint Development CI/CD TEST/PROD

The diagram shows the process of building and deploying a new version of a package in the form of a Docker image to the TEST/PROD environment. The diagram is applicable to the case of in-house packages; for other existing ones, the public DockerHub source is used.

Description of the elements used:

Build and Upload Docker Image
Build process and upload of the Docker Image to the Docker Registry

Download Docker Image
Process of storing the Docker Image in the Docker Registry

New build
Launch of a new build process

Run Docker Container
Process of launching a Docker Container

Save Docker Image
Process of storing the Docker Image in the Docker Registry

TEST/PROD environment
Represents the internal DEV and TEST environment

Brain:IT
Brain:IT site

Private Docker Registry
Represents a private Docker Registry

Deployment configuration
Representation of the Kubernetes Deployment artefact.

Docker Image
Representation of the Docker Image artefact.

GitLab
Represents a GitLab instance.

GitLab Worker
Represents a GitLab Worker instance

Kubernetes Cluster
Represents the Kubernetes Cluster

GitLab CI/CD Pipelines
GitLab CI/CD Pipelines are a series of tasks providing compilation, tests, configuration, deployment and other activities within GitLab.

Pod
Represents a Pod instance in which a new Docker Image is run in the form of a Docker Container

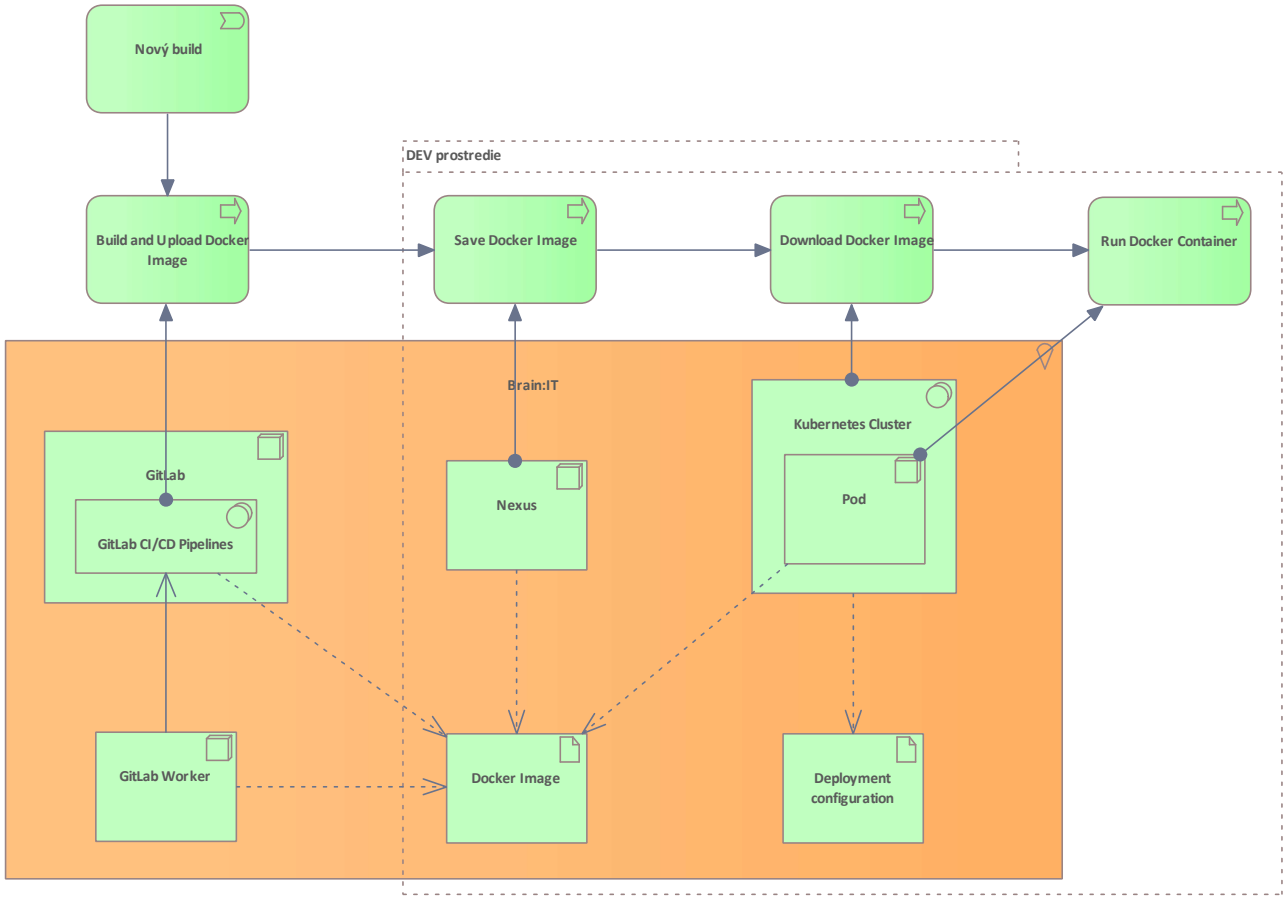


Figure 49 : Technology Viewpoint Developmnent CI/CD DEV

The diagram shows the process of building and deploying a new version of a package in the form of a Docker image to the DEV environment. The diagram is applicable to the case of in-house packages; for other existing ones, the public DockerHub source is used.

Description of the elements used:

Build and Upload Docker Image
Build process and upload of the Docker Image to the Docker Registry

Download Docker Image
Process of storing the Docker Image in the Docker Registry

New build
Launch of a new build process

Run Docker Container
Process of launching a Docker Container

Save Docker Image
Process of storing the Docker Image in the Docker Registry

DEV environment
Represents the internal DEV and TEST environment

Brain:IT
Brain:IT site

Deployment configuration
Representation of the Kubernetes Deployment artefact.

Docker Image
Representation of the Docker Image artefact.

GitLab
Represents a GitLab instance.

GitLab Worker
Represents a GitLab Worker instance

Kubernetes Cluster
Represents the Kubernetes Cluster

Nexus
Represents a private Docker Registry

GitLab CI/CD Pipelines
GitLab CI/CD Pipelines are a series of tasks providing compilation, tests, configuration, deployment and other activities within GitLab.

Pod
Represents a Pod instance in which a new Docker Image is run in the form of a Docker Container

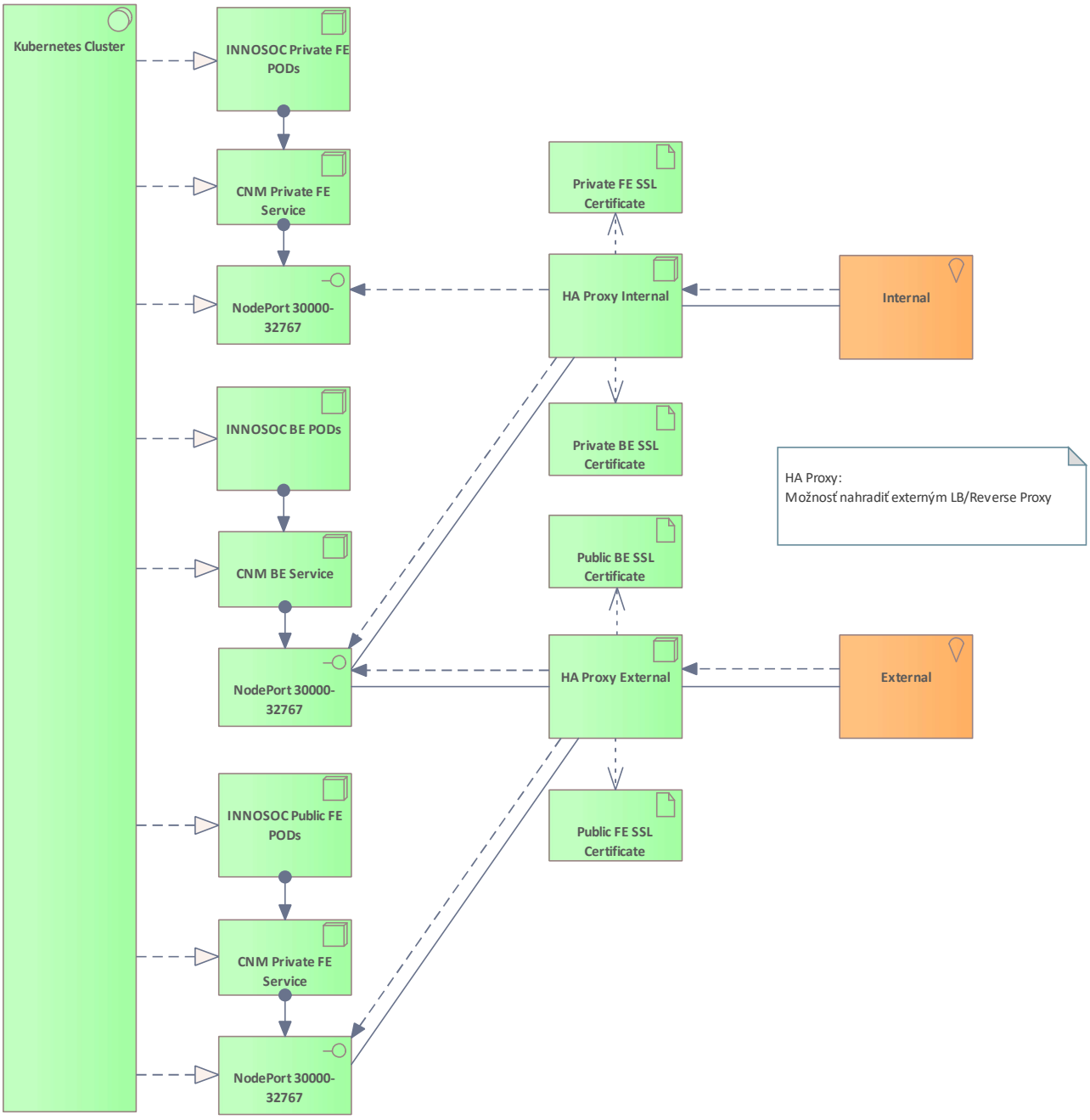


Figure 50 : Technology Viewpoint Interconnect

The diagram shows the deployment and interconnection of exposed Kubernetes services and the LB/Reverse Proxy for two zones.

Description of the elements used:

CNM BE Service
Represents CNM BE Service

CNM Private FE Service
Represents the CNM FE Private Service

CNM Private FE Service
Represents the CNM FE Private Service

External

Represents the external Internet environment

HA Proxy External
Represents an external reverse proxy built on HA Proxy

HA Proxy Internal
Represents an internal reverse proxy built on HA Proxy

Internal
Represents the internal BRAIN:IT environment

Private BE SSL Certificate
Represents a valid private SSL certificate for HTTPS purposes

Private FE SSL Certificate
Represents a valid SSL certificate for HTTPS purposes

Public BE SSL Certificate
Represents a valid public SSL certificate for HTTPS purposes

Public FE SSL Certificate
Represents a valid public SSL certificate for HTTPS purposes

NodePort 30000-32767
Represents NodePort-type routing from the range 30000-32767

NodePort 30000-32767
Represents NodePort-type routing from the range 30000-32767

NodePort 30000-32767
Represents NodePort-type routing from the range 30000-32767

Note
HA Proxy:
Can be replaced by an external LB/Reverse Proxy

Kubernetes Cluster
Represents the Kubernetes Cluster

INNOSOC BE PODs
Represents the CNM BE PODs

INNOSOC Private FE PODs
Represents the CNM FE Private PODs

INNOSOC Public FE PODs
Represents the CNM FE Public PODs

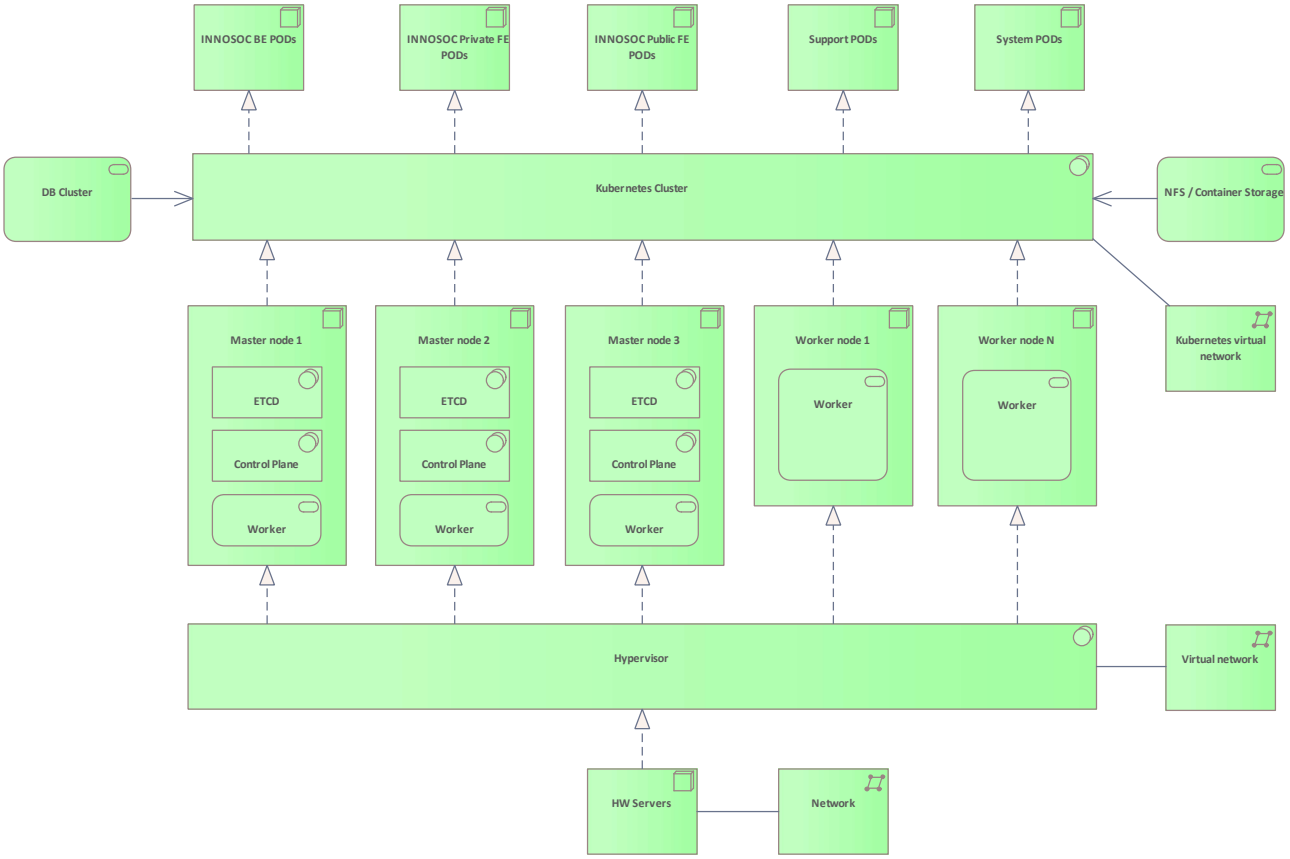


Figure 51 : Technology Viewpoint Kubernetes Cluster

Represents the role and service of the Worker node .

Description of the elements used:

DB Cluster
Represents the clustered relational database service

NFS / Container Storage
Represents the NFS service or another Container Storage

HW Servers
Represents the HW server infrastructure

Hypervisor
Represents a Type 1 or Type 2 hypervisor

Kubernetes virtual network
Represents the virtual network/subnet of the Kubernetes Cluster

Master node 1
Represents the Master node of the Kubernetes cluster

Master node 2
Represents the Master node of the Kubernetes cluster

Master node 3
Represents the Master node of the Kubernetes cluster

Network
Represents the network/subnet of the HW infrastructure

Support PODs
Represents the PODs of supporting services

System PODs
Represents the PODs of system services

Virtual network
Represents the virtual network/subnet of the hypervisor

Worker node 1
Represents a Worker node of the Kubernetes cluster

Worker node N
Represents a Worker node of the Kubernetes cluster

Kubernetes Cluster
Represents the Kubernetes Cluster

INNOSOC BE PODs
Represents the CNM BE PODs

INNOSOC Private FE PODs
Represents the CNM FE Private PODs

INNOSOC Public FE PODs
Represents the CNM FE Public PODs

Worker
Represents the role and service of the Worker node

Control Plane
Represents the Control Plane for managing the Kubernetes cluster

ETCD
Represents the ETCD database

Worker
Represents the role and service of the Worker node

Control Plane
Represents the Control Plane for managing the Kubernetes cluster

ETCD
Represents the ETCD database

Worker

Control Plane
Represents the Control Plane for managing the Kubernetes cluster

ETCD
Represents the ETCD database

Worker

Worker

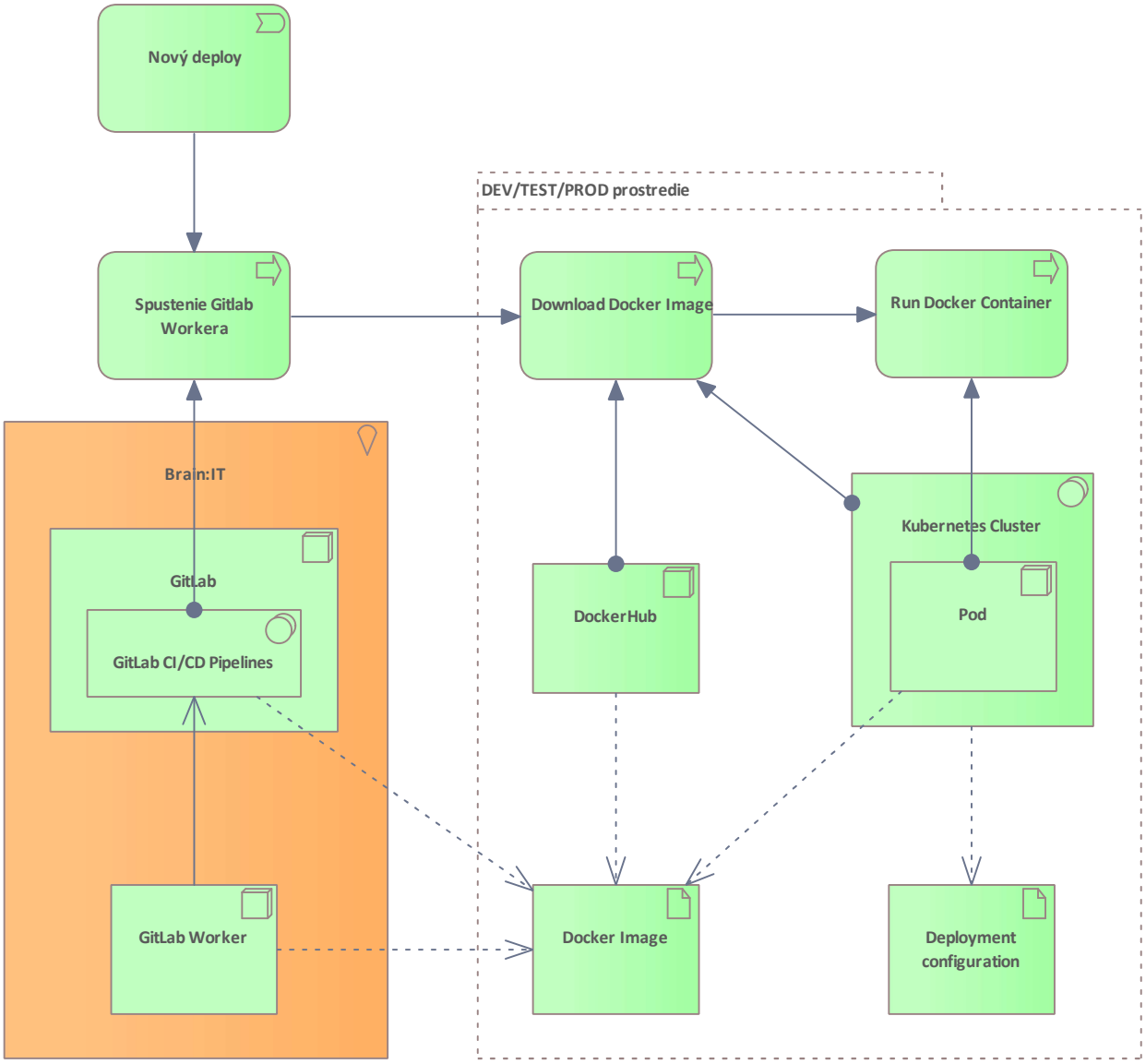


Figure 52 : Technology Viewpoint standard CI/CD DEV/TEST/PROD

Description of the elements used:

Download Docker Image
Process of storing the Docker Image in the Docker Registry

New deploy
Launch of a new deploy process

Run Docker Container
Process of launching a Docker Container

Starting the Gitlab Worker
process of launching the Gitlam Worker to deploy a new version of the package.

DEV/TEST/PROD environment
Represents the internal DEV and TEST environment

Brain:IT
Brain:IT site

DockerHub
Represents the public Docker Registry DockerHub

Deployment configuration
Representation of the Kubernetes Deployment artefact.

Docker Image
Representation of the Docker Image artefact.

GitLab
Represents a GitLab instance.

GitLab Worker
Represents a GitLab Worker instance

Kubernetes Cluster
Represents the Kubernetes Cluster

GitLab CI/CD Pipelines
GitLab CI/CD Pipelines are a series of tasks providing compilation, tests, configuration, deployment and other activities within GitLab.

Pod
Represents a Pod instance in which a new Docker Image is run in the form of a Docker Container

7.5 Technology Viewpoint OpenStack

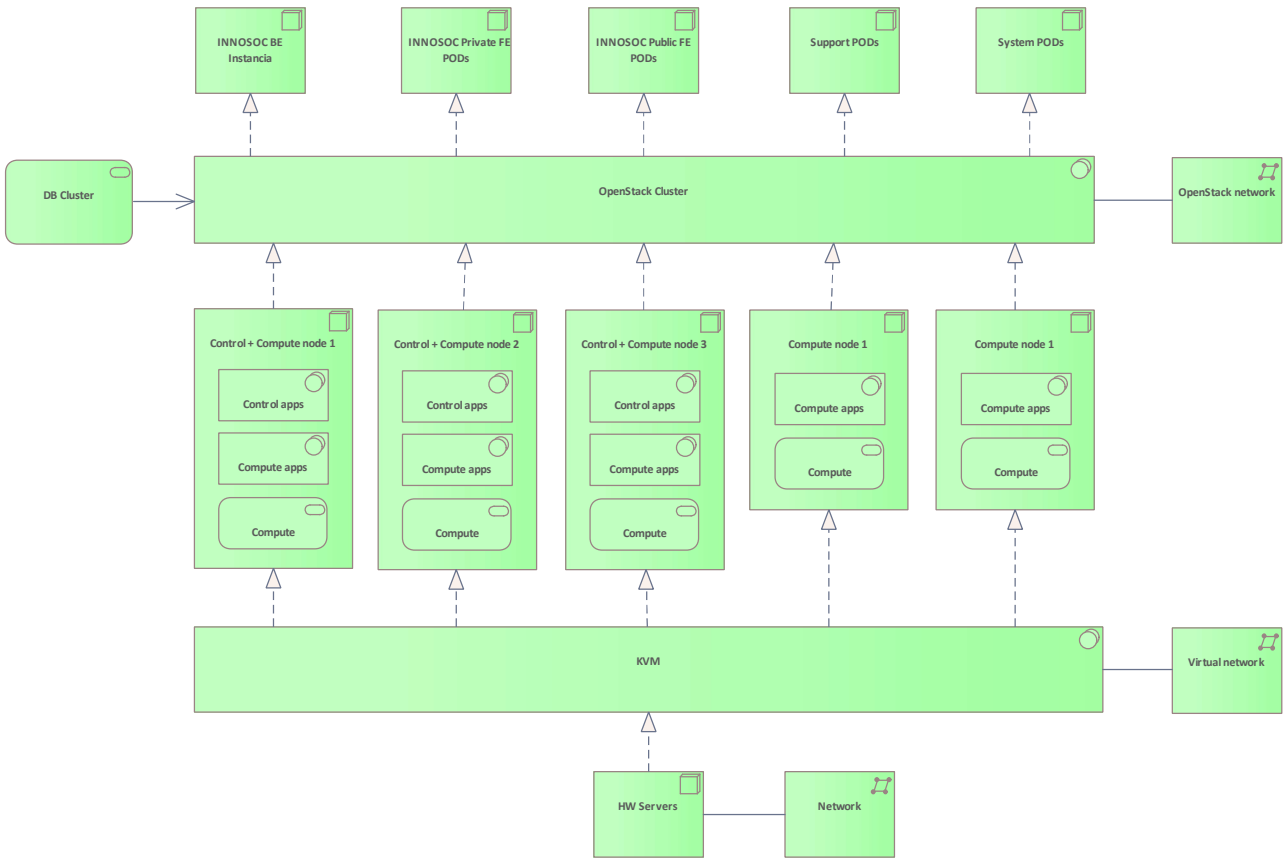


Figure 53 : Technology Viewpoint OpenStack Cluster

Description of the elements used:

DB Cluster
Represents the clustered relational database service

Compute node 1
Represents a Compute node of the OpenStack cluster

Compute node 1
Represents a Compute node of the OpenStack cluster

Control + Compute node 1
Represents Control and Compute node 1 of the OpenStack cluster

Control + Compute node 2
Represents Control and Compute node 2 of the OpenStack cluster

Control + Compute node 3
Represents Control and Compute node 3 of the OpenStack cluster

HW Servers
Represents the HW server infrastructure

KVM
Represents a type 1 KVM hypervisor.

Network
Represents the network/subnet of the HW infrastructure

OpenStack Cluster
Represents the OpenStack Cluster

OpenStack network
Represents the network/subnet of the OpenStack cluster, specifically:
Corporate network
OpenStack Tenant Data Network
OpenStack Internal Admin / API Subnet

Support PODs
Represents the PODs of supporting services

System PODs
Represents the PODs of system services

Virtual network
Represents the virtual network/subnet of the hypervisor

INNOSOC BE Instance
Represents an OpenStack VM instance or a container

INNOSOC Private FE PODs
Represents the CNM FE Private PODs

INNOSOC Public FE PODs
Represents the CNM FE Public PODs

Compute
Represents the role and service of the Compute node

Compute apps
Represents the set of Compute applications

Control apps
Represents the set of applications for the Control node required for managing the cluster

Compute
Represents the role and service of the Compute node

Compute apps
Represents the set of Compute applications

Compute
Represents the role and service of the Compute node

Compute apps
Represents the set of Compute applications

Control apps
Represents the set of applications for the Control node required for managing the cluster

Compute
Represents the role and service of the Compute node

Compute apps
Represents the set of Compute applications

Control apps
Represents the set of applications for the Control node required for managing the cluster

Compute
Represents the role and service of the Compute node

Compute apps
Represents the set of Compute applications

7.6 Technology Viewpoint OpenStack External

7.7 Technology Viewpoint Reporting

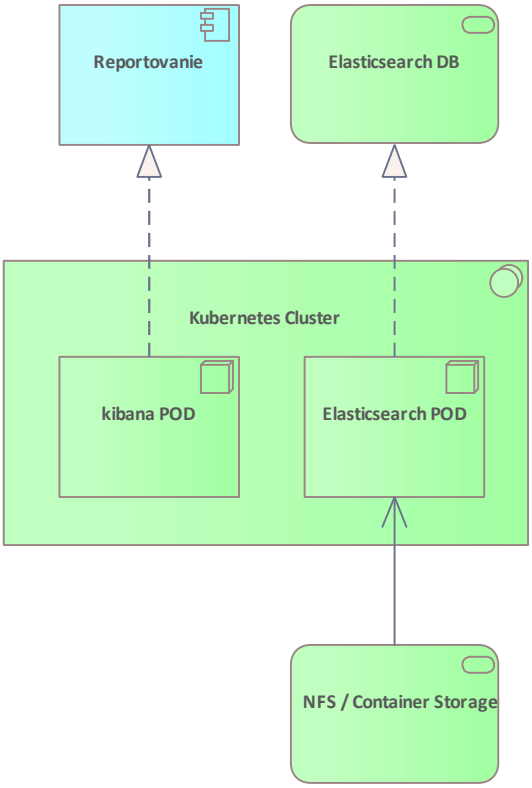


Figure 54 : Technology Usage Viewpoint Reporting alt 1

The diagram shows a simplified assignment of the individual PODs to the corresponding application components. Alt 1 represents the use of the Elasticsearch database directly within the module setup. The design envisages a containerized version of the NoSQL database for Elasticsearch.

Description of the elements used:

Reporting
The reporting component within the SOCaaS architecture serves as the central means for aggregating, visualizing and distributing security information. Its role is to provide both regular and ad hoc outputs on the state of cybersecurity for various target groups – from SOC analysts, through team leaders, to the management of the organization or clients. Such a reporting system must cover:
• Generation of metrics for incidents, alerts, SLAs and KPIs,
• Overviews of trends over time and assessment of the performance of security measures,
• Adaptation of content according to the level of detail (technical vs. strategic),
• Automation and scheduling of the distribution of outputs,
• Retrospective analysis and archiving of historical data.

NFS / Container Storage
Represents the NFS service or another Container Storage

Kubernetes Cluster
Represents the Kubernetes Cluster

Elasticsearch DB
Represents the technology service of the NoSQL Elasticsearch database.

Elasticsearch POD
Represents the Elasticsearch POD for the NoSQL database.

kibana POD
Represents the Kibana POD for Reporting.

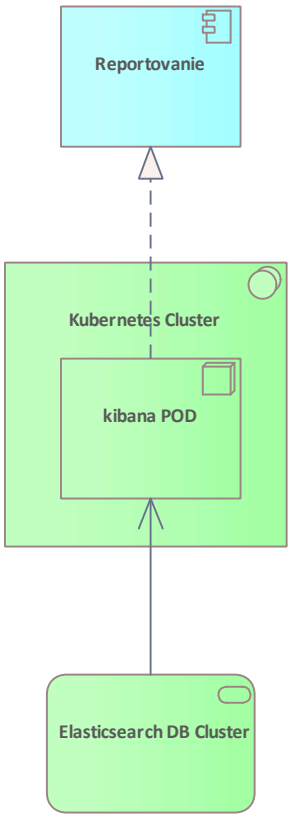


Figure 55 : Technology Usage Viewpoint Reporting alt 2

The diagram shows a simplified assignment of the individual PODs to the corresponding application components. Alt 2 represents the use of the Elasticsearch database directly in the form of a centralized DB cluster.

Description of the elements used:

Reporting
The reporting component within the SOCaaS architecture serves as the central means for aggregating, visualizing and distributing security information. Its role is to provide both regular and ad hoc outputs on the state of cybersecurity for various target groups – from SOC analysts, through team leaders, to the management of the organization or clients. Such a reporting system must cover:
• Generation of metrics for incidents, alerts, SLAs and KPIs,
• Overviews of trends over time and assessment of the performance of security measures,
• Adaptation of content according to the level of detail (technical vs. strategic),
• Automation and scheduling of the distribution of outputs,

• Retrospective analysis and archiving of historical data.

Kubernetes Cluster
Represents the Kubernetes Cluster

Elasticsearch DB Cluster
Represents the technology service of the NoSQL Elasticsearch database in the form of a cluster.

kibana POD
Represents the Kibana POD for Reporting.

7.8 Technology Viewpoint Vulnerability Scanner

7.9 Technology Viewpoint Ticketing system / Incident registration and management

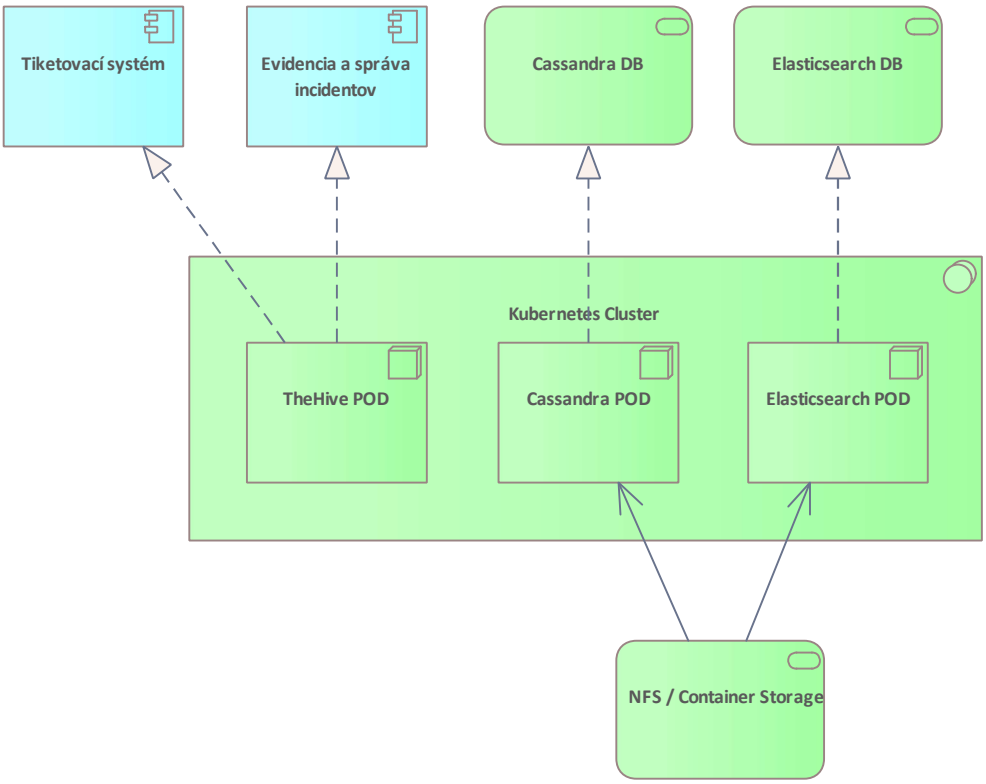


Figure 56 : Technology Usage Viewpoint Ticketing System / Incident Recording and Management alt 1

The diagram shows a simplified assignment of the individual PODs to the corresponding application components. Alt 1 represents the use of the Elasticsearch database directly within the module setup. The design envisages a containerized version of the NoSQL databases for Cassandra and Elasticsearch.

Description of the elements used:

Incident recording and management

Incident recording and management is a core element of every security operation (SOC). This subcomponent ensures systematic documentation, tracking and management of security events and incidents throughout their life cycle – from detection through analysis to closure. A high-quality recording system must enable teams to collaborate in a coordinated manner, document steps, record related artefacts (observables) and link incidents with relevant threats, systems or historical cases. Ideally, it also supports advanced data enrichment, auditing, multi-tenancy and the possibility of retrospective investigation of incidents.

Ticketing system

The ticketing system serves as the central communication and organizational channel for the management of security incidents throughout their entire life cycle. It enables the creation, assignment and tracking of tasks and communication between analysts, while supporting audit, SLA monitoring, escalation and notifications.

NFS / Container Storage
Represents the NFS service or another Container Storage

Kubernetes Cluster
Represents the Kubernetes Cluster

Cassandra DB
Represents the technology service of the NoSQL Cassandra database.

Elasticsearch DB
Represents the technology service of the NoSQL Elasticsearch database.

Cassandra POD
Represents the Apache Cassandra POD for the NoSQL database.

Elasticsearch POD
Represents the Elasticsearch POD for the NoSQL database.

TheHive POD
Represents the TheHive POD for asset inventory.

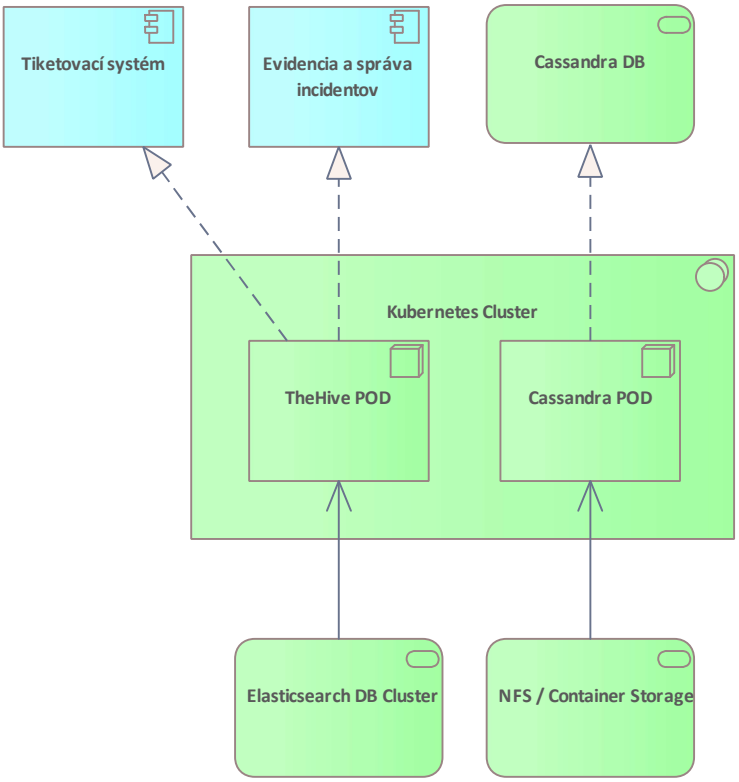


Figure 57 : Technology Usage Viewpoint Ticketing System / Incident Recording and Management alt 2

The diagram shows a simplified assignment of the individual PODs to the corresponding application components. Alt 2 represents the use of the Elasticsearch database directly in the form of a centralized DB cluster. The design envisages a containerized version of the NoSQL database for Cassandra.

Description of the elements used:

Incident recording and management
Incident recording and management is a core element of every security operation (SOC). This subcomponent ensures systematic documentation, tracking and management of security events and incidents throughout their life cycle – from detection through analysis to closure. A high-quality recording system must enable teams to collaborate in a coordinated manner, document steps, record related artefacts (observables) and link incidents with relevant threats, systems or historical cases. Ideally, it also supports advanced data enrichment, auditing, multi-tenancy and the possibility of retrospective investigation of incidents.

Ticketing system
The ticketing system serves as the central communication and organizational channel for the management of security incidents throughout their entire life cycle. It enables the creation, assignment and tracking of tasks and communication between analysts, while supporting audit, SLA monitoring, escalation and notifications.

NFS / Container Storage
Represents the NFS service or another Container Storage

Kubernetes Cluster
Represents the Kubernetes Cluster

Cassandra DB
Represents the technology service of the NoSQL Cassandra database.

Elasticsearch DB Cluster

Represents the technology service of the NoSQL Elasticsearch database in the form of a cluster.

Cassandra POD
Represents the Apache Cassandra POD for the NoSQL database.

TheHive POD
Represents the TheHive POD for asset inventory.

7.10 Technology Viewpoint Knowledge Portal for the SOC Team

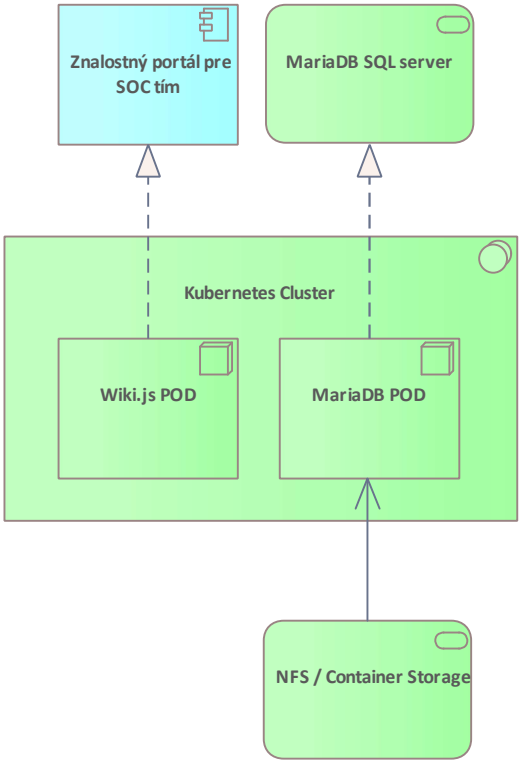


Figure 58 : Technology Usage Viewpoint Knowledge Portal for the SOC Team alt 1

The diagram shows a simplified assignment of the individual PODs to the corresponding application components.
Alt 1 represents the use of the database directly within the module setup. The design envisages a containerized version of the SQL server.

Description of the elements used:

Knowledge portal for the SOC team
A knowledge portal for the SOC team is an important building block of every modern security operations centre. It serves as a central repository of know-how, playbooks, procedures, recommendations, analytical guides and often also historical lessons learned from incidents. The knowledge portal:
• Increases the efficiency and consistency of incident response,
• Enables rapid onboarding of new team members,
• Supports the standardization of processes and methodologies,

• Enables the team to retain and share proven procedures and its own analytical experience,
• Strengthens the ability to learn collectively from incidents and to build internal "threat intel" knowledge.

NFS / Container Storage
Represents the NFS service or another Container Storage

Kubernetes Cluster
Represents the Kubernetes Cluster

MariaDB SQL server
Represents the technology service of the relational MariaDB SLQ database.

MariaDB POD
Represents the MariaDB POD for the relational SQL DB.

Wiki.js POD
Represents the Wiki.js POD for the Knowledge portal for the SOC team.

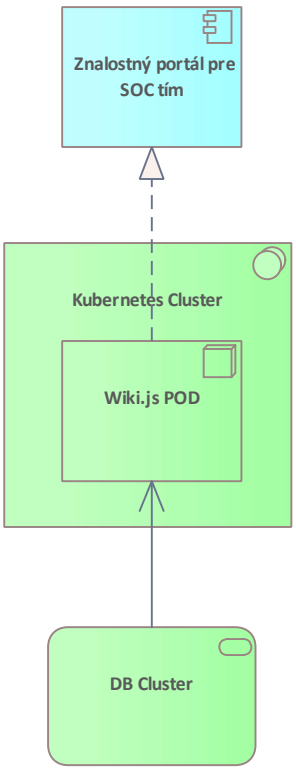


Figure 59 : Technology Usage Viewpoint Knowledge Portal for the SOC Team alt 2

The diagram shows a simplified assignment of the individual PODs to the corresponding application components. Alt 2 represents the use of the database directly in the form of a centralized DB cluster.

Description of the elements used:

Knowledge portal for the SOC team

A knowledge portal for the SOC team is an important building block of every modern security operations centre. It serves as a central repository of know-how, playbooks, procedures, recommendations, analytical guides and often also historical lessons learned from incidents. The knowledge portal:
• Increases the efficiency and consistency of incident response,
• Enables rapid onboarding of new team members,
• Supports the standardization of processes and methodologies,
• Enables the team to retain and share proven procedures and its own analytical experience,
• Strengthens the ability to learn collectively from incidents and to build internal "threat intel" knowledge.

DB Cluster
Represents the clustered relational database service

Kubernetes Cluster
Represents the Kubernetes Cluster

Wiki.js POD
Represents the Wiki.js POD for the Knowledge portal for the SOC team.

7.11 Technological components of the solution and their relationship to application components

7.11.1 Management of access, accounts and keys of operating environments

7.11.2 Containerization and orchestration of cloud services

7.11.3 Specification of the use of government cloud services

7.12 Description of the test environment

Given in chapter 7.1.

7.13 Description of the production environment

Given in chapter 7.1.

8 Service level (SLA) and performance requirements

9 Software licences and source code

List of SW licences that will be used during development:

No.	Name of the SW used	Type of the SW used	SW licence
....			

10 Ensuring availability, backup and recovery of the solution

The description of the approach to ENSURING AVAILABILITY, Backup and recovery of the solution is precisely set out and addressed in the valid **Contract for the Provision of Information System Support and Maintenance Services**, which takes effect on the date of acceptance of the Final Acceptance Protocol of the delivered work.

10.1 Ensuring availability

The basic pillar of the availability of critical systems in a cloud environment is system clustering, which ensures automatic failover of an application to a standby node (virtual server) in the event of a failure of the primary node and the execution of all operations associated with it. Critical applications or network services are implemented in high availability (HA) of systems by means of a software cluster..., which represents the standard in the field of high availability.

The reliability of critical systems is implemented at two (up to four) levels:

1. High availability of the system through the use of software clustering. This ensures a fast, operator-independent restart of the system after a hardware or software crash or a system fault.
2. Cyclical (and automatic) data backup (system logs, application data, the applications themselves...). In this way, we protect the system against any loss of valuable data.
3. The purpose of a service intervention is to return the system to its original state and to repair hardware or network faults whose occurrence caused part of the system to stop functioning, thus preparing it for another potential outage. If the designed "redundancy" of the system were not sufficient to ensure the functioning of the system in the event of larger-scale outages, or in the case of outages caused by incorrect operation (administrator errors, etc.), it is sometimes necessary to restore systems to operation using data backups as well.
4. Replication of data to another location for the resilience of the system (computer centre) against large-scale outages, natural disasters, etc. However, this matter is fully ensured by the cloud solution provider.

Backups are created of the server application environment, system configuration, operation records and the databases themselves.

All important central components of the system and infrastructure are deployed in production operation multiple times as needed, so that the failure of any single component does not jeopardize the failure or unavailability of certain system services.

All internal network interfaces of workstations are protected by a local firewall, have an access control policy set with ports blocked by default (default deny all), and communication is permitted only for designated connections according to the communication matrix.

10.2 Backup

The PostgreSQL database needs to be backed up. It is not necessary to deal with file backup.
The method of backup and recovery is the responsibility of the contracting authority; the recommended backup frequency is once every 12 hours with a retention policy of one week.

10.3 Recovery of the solution

11 Security - addressing security requirements

Secure development principles
The requirements laid down in this document must be met for all components that are used during development and all artefacts that are considered products of development.
The system will be developed in accordance with internationally recognized recommendations and best practices in the field of secure software development. The development activities will reflect the requirements defined within the CISA Secure by Design initiative (https://www.cisa.gov/resources-tools/resources/secure-by-design) as well as the OWASP Developer Guide methodology (https://owasp.org/www-project-developer-guide/release/).

By products of development we mean in particular:
- High-level Design (HLD), Low-level Design (LLD), Entity Relationship Diagram (ERD), process diagrams and the like
- software source code
- executable programs that are part of the delivery
- configurations
- database schemas
- test data sets

It is advisable to divide the security requirements into three parts:
- Requirements for the development environment
- Requirements for the developed product itself – the application
- Requirements for the production environment in which the product is operated.

The aim of these measures is to reduce the likelihood and impact of threats associated with software development and delivery, in particular:
- introduction of vulnerabilities as a result of poor programming techniques,
- introduction of vulnerabilities as a result of using insecure and unsupported frameworks,
- introduction of undesirable functionality into the code, whether through negligence or intentionally,
- delay in delivery due to an outage of development environments,
- breach of personal data protection principles due to its improper handling,
- introduction of vulnerabilities due to incorrect configuration of environments,
- criminal liability as a result of copyright infringement,
- leakage of sensitive data, loss of sensitive data,
- inability to secure evidence in the event of a security incident, both in development and in operation, and the like.

During the development of the project, it is necessary to prevent and avoid the introduction of the most common implementation errors, vulnerabilities and risks by adhering to a secure development life cycle, best practices and an appropriately chosen security standard and/or methodology. During the implementation of the proposed solution, the software is verified during development against well-known vulnerabilities and security recommendations, which form part of secure development practice to prevent the occurrence of the most common security risks for web applications.

By components used during development we mean in particular:

- workstations used in development
- virtualization environment
- servers used in development (development, test)
- network environment used in development, including home networks
- printers and other active elements
- physical environment in which development takes place, including home offices
- supporting services used in development, such as communication tools (MS Teams, Jira, email...)
- development tools (editors, compilers)
- development frameworks
- middleware (web servers, proxy servers, application servers)
- source code and configuration repositories (github)
- backup applications/services/systems and backed-up data

authentication mechanisms (AD, AAD, LDAP, Radius, Kerberos, SAML, cloud access security broker and the like)

Securing the development environment

Development follows security best practice principles, including secure administration, configuration, patching and hardening of workstations and servers, anti-malware protection, segregation of physical and logical environments, account policies, credential management and secure login procedures, e.g. in accordance with ISO/IEC 27001.

All devices used in development will be protected by a firewall, both at the network perimeter (the interface between networks) and locally by means of a host firewall that permits only necessary connections. Only devices that are needed for development may be connected to the networks in which development takes place.
Endpoints on which development takes place must not be accessible to or used by other persons. In the case of computers, automatic screen locking protected by a password will be ensured and required.
The computer will not automatically connect to Wi-Fi networks other than the protected networks used in development. The Wi-Fi SSID to which a developer's computer is connected must have access controlled by WPA2 and a complex password, ideally authentication via WPA2 Enterprise. Where a physical LAN is used, access protection must be used.
All devices used in development, including the OS of network and security devices (routers, switches, firewalls, Wi-Fi access points), must have current security patches installed, and the firmware version must be at least N-1.

All network communication to and from the internet must pass through a device that filters dangerous and malicious content. All communication from computers used for development that are located outside protected internal networks (for example, when working from home) must pass through the corporate VPN gateway, which inspects and filters dangerous content.
Only legal, supported and up-to-date software will be used in development. The disks of all portable devices (laptops) must be encrypted. It is also necessary to encrypt the disks of non-portable workstations and servers that are not located in sufficiently protected data centre and server room premises.

All configurations must be protected against unauthorized changes. All configuration changes will be automatically detected and reported.

Only personal accounts may be used to log in to systems. Personal accounts used for ordinary work must be separate from personal accounts for administration. The use of generic and shared accounts (root, administrator) is prohibited except for disaster recovery. All accounts must be protected by SSH keys or, in exceptional cases where necessary, by a sufficiently strong password or passphrase with a high level of entropy.
On all systems used for development, only the minimum necessary set of software required for the specific activity may be installed.

All test data, sensitive data, passwords and the like are stored in defined locations.

End user management - The rights themselves, whether for access to information, permissions to work with data or access to individual modules and functionalities, will be governed by permissions defined in central user management and identity management, known as IAM (Identity and Access Management), using oAuth/SAML with the possible use of 2FA, on the basis of which individual user roles will be assigned.

Ensuring integrity

Validation of all user inputs and actions will be performed on the server side and, for suitable operations, also on the client side. All user inputs must be checked on the server side.
Both the web server process and the backend database process must be configured to run under a unique user account with a limited set of privileges.
The web server will be configured so that files with web content are accessible to the process belonging to the web server service for reading only, not for writing (unless the service and process themselves require it).

The web content and all logs generated by it will be located on a separate hard disk or on a logical partition other than that of the OS and the web server. In the case of containerization, the logging method will be adapted in line with the orchestration platform concerned.
To meet the security requirements associated with the development of the project, a suitable tool will be used that provides continuous code quality checks, validation of the secure life cycle and checks that detect bugs, duplicated code and security holes.

Securing communication

All communication channels will be known and described. The individual system components will be separated according to layers that perform a specific role.
The basic layers are:
- Access layer
- Application layer
- Database layer

The individual layers will be located in separate, logically isolated network segments or independently executable blocks, or by means of defining access on the basis of access groups, with communication between them permitted on the basis of firewall rules or orchestration platform rules, and in addition monitoring and filtering will be possible.
The administration and monitoring interfaces of network devices and servers are not accessible directly from the external network, the internet. The administration and monitoring interfaces are accessible only from pre-specified networks (e.g. the administrator network, a designated internal network).
The web server will be externally accessible exclusively via the HTTPS protocol. In line with best practices, the web portal is not to be accessed externally via HTTP.
The identity of the web portal will be ensured by a valid, trusted certificate issued for the domain on which the portal in question is available. For this purpose, a certificate and a technical certificate from an internal CA may be used. The web portal must not use untrusted or expired SSL/TLS certificates towards the external environment.
Data that are sensitive in terms of integrity or confidentiality must be transmitted only via an encrypted SSL/TLS connection. Sensitive data (especially credentials) must be transmitted exclusively via an encrypted channel.
The web server will support TLS 1.2, TLS 1.3 and later. By default, it will not support the old versions SSLv2, SSLv3, TLS 1.0 and TLS 1.1, unless otherwise required.
The web server is not to support ciphers with a key shorter than 112 bits and a block shorter than 64 bits. The web server must not support (they must be disabled) the HTTP methods OPTIONS, TRACK and TRACE.
The web application will, to the greatest possible extent, take into account and apply recommendations for the elimination of well-known vulnerabilities according to best practices.
The basic logical topology of the system solution is shown within the physical architecture.

Access control
Access control to the application uses IAM, which assigns user roles on the basis of permissions defined in central user management based on oAuth/SAML and, where applicable, 2FA.
Access control between individual zones or subnets is ensured by means of firewalls or another security mechanism.

System logging and collection of audit records
Logging and monitoring of the system will be ensured by technology supplied by the Contracting Authority. It is recommended to set appropriate metrics for monitoring resource utilization and capacity for each monitored device/system.
System monitoring and logging should include alerts that should be provided to the system administrator in cases such as: storage capacity approaching full, abnormally high system load, an unusual security event detected.
Logs and audit records will contain specific information on the date, time and time zone used, should be transmitted via a secure channel, and the granularity of logging will correspond to the requirements of the given service. Log retention should be set so as to prevent the logging storage from filling up and not to affect the stability of resources.
Logging is subject to a centrally defined retention policy that determines the retention period of individual types of log records depending on their nature (e.g. operational, security, audit). Retention periods will be configurable and set in accordance with legislative requirements and the operational needs of the Contracting Authority. The retention policy recommended by the supplier is at least 1 month for logging and at least one year for audit logs. After the retention period has expired, the records will be automatically archived or securely deleted.

User management methodology
The rights themselves, whether for access to information, permissions to work with data or access to individual modules and functionalities, will be governed by permissions defined in central user management and identity management, known as IAM (Identity and Access Management), using oAuth/SAML with the possible use of 2FA, on the basis of which individual user roles will be assigned. IAM will also be used to define the way in which individual users may interact with the system (CRUD operations).

12 Data migration

No data migration will be performed within the solution.

12.1 Data migration procedure

12.2 Data migration verification

13 Implementation and deployment schedule, dependencies

13.1 Project phases and timeline - consolidated overview

13.2 Dependencies and solution design

13.3 Requirements for the Testing and Production Deployment phases

The work will be implemented as a project managed in accordance with the PRINCE2 methodology. The delivery of the solution will go through the basic project stages within the meaning of Decree No. 401/2023 on project management. The outputs of each stage will be approved by the Project Steering Committee.

14 Open issues

15 Annexes

List of DNR annexes:

1. Catalogue of requirements and mapping to their implementation in the proposed solution
2. List of figures