PROJECT

Multi-tenant cybersecurity operations centre designed
as an open cloud service with machine learning elements

V6 description of dashboards, the monitoring and reporting system

KPB2 – Design of the SOCaaS service

type – documentation

submission month – M7

Funded by the European Union
NextGenerationEU

RECOVERY AND RESILIENCE PLAN

GOVERNMENT OFFICE OF THE SLOVAK REPUBLIC

CONTENTS

1. Introduction ... 3
2. INNOSOC SOCaaS system architecture ... 3
3. Roles and processes in the SOC ... 4
3.1. Organizational roles in the SOC ... 4
3.2. Processes in the SOC ... 4
3.2.1. Detection and monitoring (SIEM – ELK Stack) ... 5
3.2.2. Incident response (SOAR, Incident handling) ... 5
3.2.3. Asset inventory and management ... 6
3.2.4. Threat Intelligence (CTI) ... 6
3.2.5. Vulnerability management ... 6
3.2.6. Reporting and retrospective analysis ... 6
3.2.7. Summary ... 6
3.3. Operational scenarios and practical deployment ... 6
4. Monitoring and analytical system (SIEM – ELK Stack) ... 9
4.1. ELK architecture and components ... 9
4.2. Processing of data flows ... 9
4.3. Correlation rules and threat detection ... 10
4.4. Integration with other components ... 12
5. Creating dashboards in Kibana ... 14
5.1. Dashboard design principles ... 14
5.2. Structure of typical dashboards ... 15
5.3. Examples of dashboards and visualizations ... 15
5.3.1. System security overview ... 15
5.3.2. Correlation and incident dashboards ... 15
5.3.3. Visualization of network traffic and access ... 16
5.3.4. Monitoring of authentications and user behaviour ... 16
5.3.5. Integrated overview for the SOC analyst ... 16
6. Reporting and data export ... 17
6.1. Reporting objectives and types ... 17
6.2. Reporting options in Kibana ... 17
6.2.1. Manual export of reports ... 17
6.2.2. Automated reporting ... 17
6.2.3. Customization of outputs ... 17
6.3. Integration of the reporting system ... 18
6.4. Challenges and recommendations ... 18
7. Conclusion and recommendations ... 19
7.1. Benefits of the proposed architecture ... 19
7.2. Reasons for selecting the ELK Stack as the SIEM core ... 19
7.3. Dashboard and reporting solution ... 19
7.4. Challenges and recommendations for further development ... 19
Annexes ... 21

1. INTRODUCTION

This document constitutes output V6 of the INNOSOC project, the aim of which is to describe in detail the architecture design of the SOCaaS (Security Operations Center as a Service) service, the specification of the main roles and processes in the SOC, as well as the design and examples of the monitoring and reporting system built on the open-source ELK platform (Elasticsearch, Logstash, Kibana). The selected architecture and components are based on an analysis of the needs of a multi-customer, modular and scalable SOC, testing of individual open-source solutions and the experience of the project team.

The output reflects the practical needs of operating a SOC in a hybrid and distributed environment, takes into account the requirements for automation, integration, security and reporting, and builds on the recommended components identified in previous outputs of the project.

2. ARCHITECTURE OF THE INNOSOC SOCAAS SYSTEM

The architecture of the proposed SOCaaS solution consists of several modular components which together form a coherent and scalable ecosystem. The architecture was designed to support security monitoring, detection, response, incident management, as well as the creation of overviews and knowledge bases.

The basic architectural areas can be divided as follows:

a) Processing and detection components

- SIEM (Security Information and Event Management): the central point for log collection and analysis.
- EDR/XDR (Endpoint/Extended Detection and Response): threat detection on endpoints.
- CTI (Cyber Threat Intelligence): external threat data and its integration into processes.
- Vulnerability scanner: automated identification of weaknesses in the infrastructure.

b) Incident response and management components

- SOAR (Security Orchestration, Automation and Response): coordination and automation of responses.
- Ticketing system: recording and management of requests and incidents.
- Incident management: case management of security events.
- Reporting: overview of outputs from detection, response and the security status.
- Knowledge portal: documentation and knowledge sharing among team members.

c) Supporting components and integrations

- Asset management: recording of assets and their relationships.
- Integration with external APIs, sandboxes, IOC/CTI databases and notification channels.
- Access control and audit: logging of activities and securing of operations.

3. ROLES AND PROCESSES IN THE SOC

3.1. Organizational roles in the SOC

The effective functioning of a security operations centre (SOC) is conditional on clearly defined roles of the individual team members, who cooperate on the detection, analysis, escalation and mitigation of cyber threats. In the context of the SOCaaS design for the INNOSOC project, the basic organizational roles were defined as follows:

- **Level 1 (L1) analyst**: The first line of defence, which monitors alerts, performs initial validation of incidents and ensures their documentation. Typically works with SIEM dashboards, the ticketing system and the knowledge base.
- **Level 2 (L2) analyst**: A more experienced analyst who takes over cases escalated from L1. Performs deeper analysis, uses correlation rules and threat intelligence, and cooperates on the development of response measures. Often uses the interfaces of SOAR tools and forensic tools.
- **Level 3 (L3) specialist/engineer**: An expert in complex incidents, advanced threat hunting, the creation of detection rules and the configuration of tools such as SIEM or EDR. Also involved in the architecture and in the automation of responses.
- **SOC lead (SOC manager)**: Responsible for the operation and capacities of the SOC team, SLAs, cooperation with customers, reporting and performance control. Coordinates both human and technological resources, including change planning, monitoring of response quality and team development.
- **Incident handler/incident response lead**: The role responsible for coordinating incident response – task planning, evidence collection, communication with stakeholders and the preparation of final reports.
- **Threat Intelligence analyst**: Responsible for working with threats from CTI sources, creating attacker profiles, enriching incidents with IOCs and TTPs, tracking trends and sharing alerts.
- **SOC systems and architecture administrator**: A technical role responsible for the deployment, administration and updating of all components of the SOCaaS architecture, including the ELK Stack, SOAR, asset management and other tools.

These roles are not rigid – they may vary depending on the size of the SOC team, the structure of customers and SLAs. In a multi-tenant SOCaaS architecture environment, the ability to assign analysts to individual customers, or to delegate them to different types of tasks according to priority and capacity, is also crucial.

3.2. Processes in the SOC

The security operations centre (SOC) functions as the central unit for managing the security of an organization's IT environment. Its activity is built on defined processes which ensure effective monitoring, detection, analysis and response to cyber threats. These processes must be precisely defined, well documented and regularly evaluated in order to guarantee consistent and coordinated SOC operation.

Within the INNOSOC SOCaaS project, the following key processes were defined, which represent the operational framework of the SOC's functioning:

- Security event monitoring process – ensures the collection and tracking of logs, metric data and alerts from all integrated systems. In our case, it relies mainly on the ELK component (Elasticsearch, Logstash, Kibana).
- Incident detection process – is responsible for identifying anomalies, attack patterns and other threats by means of correlation rules and heuristic mechanisms.
- Incident response process – defines the way in which the team responds to identified incidents. It includes escalation, assignment of responsibility, execution of measures,

as well as communication with the affected parties. It is implemented in conjunction with the SOAR and Ticketing components.
- Incident management process – includes the recording, classification, prioritization, documentation and closing of incidents. In the project, it relies on tools such as TheHive, RTIR or Zammad.
- Threat management process (Threat Intelligence) – enables systematic work with threat information. CTI tools such as OpenCTI, MISP and IntelMQ are used, which provide both external and internal information sources.
- Reporting and visualization process – ensures regular outputs for management, auditors and SOC analysts. It is based on the Kibana and Grafana components and builds on the ELK monitoring system.
- Knowledge management process – ensures the maintenance of the knowledge base of playbooks, documentation and recommended responses. It relies on tools such as Wiki.js and BookStack.

Each of these processes was defined within the INNOSOC project with regard to standards such as NIST 800-61, ISO/IEC 27035 and recommendations from current MITRE ATT&CK frameworks, which enables a high level of interoperability with state, commercial and academic SOC initiatives.

The processes in the SOC represent the basic pillar of its effective functioning. Each component designed in the INNOSOC SOCaaS architecture is linked to specific processes which ensure comprehensive handling of cyber threats throughout their entire life cycle – from detection, through response, to the evaluation and improvement of security policies.

3.2.1. Detection and monitoring (SIEM – ELK Stack)

The detection processes are implemented by means of the ELK Stack platform (Elasticsearch, Logstash, Kibana), which forms the basis of the monitoring system. They include:

- Collection and centralization of logs from systems, networks, endpoints and applications.
- Normalization and enrichment of data using Logstash and Beats.
- Storage, indexing and retention of data in Elasticsearch.
- Creation of correlation rules and alerts.
- Visualization of outputs in Kibana.
- Integration with other tools such as Wazuh (EDR), CTI systems and incident management.

These processes form the basis for identifying incidents in real time and enable a rapid response.

3.2.2. Incident response (SOAR, Incident handling)

The response processes are implemented by means of:

- **The TheHive + Cortex SOAR tool**: used to automate response scenarios, process alerts into cases and analyse them.
- **Ticketing and incident management system (TheHive, RTIR)**: provides a workflow for incident handling, escalation, audit and resolution tracking.
- **Knowledge portal (Wiki.js, BookStack)**: supports the response processes by sharing playbooks, internal policies and guides.

These tools support the team of analysts in effective incident handling, recording of observables, documenting responses and retrospective analysis.

3.2.3. Asset inventory and management

IT asset management is crucial for mapping the impacts of threats. We use a combination of:

- **NetBox**, **GLPI**, **Ralph**: for maintaining the CMDB database and linking assets to locations, services, owners and SLAs.

These tools provide support for the processes of asset discovery and the asset life cycle, and integration with SIEM and incident management.

3.2.4. Threat Intelligence (CTI)

CTI (Cyber Threat Intelligence) processes enable the enrichment of incidents with known threats, TTPs and IOCs:

- We use tools such as **OpenCTI**, **MISP** and **IntelMQ** for the collection, classification and distribution of threat intel data.

These processes provide analysts with up-to-date threat information and support proactive threat hunting.

3.2.5. Vulnerability management

Vulnerabilities identified in the environment are processed via:

- **Greenbone Community Edition (GVM)** as the tool for active scanning.

The results are linked to assets and sent to the SIEM/incident system for further processing.

3.2.6. Reporting and retrospective analysis

- Use of **Kibana** for the creation of regular reports, ad hoc overviews and audit outputs.

Exports and report scheduling cover the needs of internal SOC evaluation and communication with customers.

3.2.7. Summary

The process model of the INNOSOC SOCaaS reflects best practices, with an emphasis on:

- Automation and orchestration.
- Interconnection of tools.
- Transparency of incident handling.
- Documentation and the knowledge base.
- Effective detection, evaluation and response.

The interconnection of the above processes ensures a robust and scalable cybersecurity environment for multiple customers.

3.3. Operational scenarios and practical deployment

The operational scenarios describe the practical aspects of the day-to-day functioning of the SOCaaS, drawing on anticipated real-world situations that the SOC team may encounter. Their purpose is to ensure the operability of the service, effective cooperation between teams and an optimized response to threats.

1. Onboarding of a new customer

During the onboarding of a new customer, their infrastructure is integrated into the multi-tenant SOCaaS environment. This includes:

- setting up the logging agent (e.g. Filebeat, Winlogbeat),
- configuring correlation rules in ELK according to device types,
- creating separate dashboards in Kibana,
- setting up incident management for the customer in TheHive and Zammad.

2. Incident detection and escalation

Rules for detecting anomalies and known threats are configured in the ELK environment. After detection:

- an alert is generated (e.g. via Wazuh or ElastAlert),
- the alert is automatically transformed into an incident in TheHive,
- Zammad provides a ticket for the L1 team and supports escalation,
- the team evaluates the observables and uses enrichment via OpenCTI or MISP.

3. Vulnerability management and inventory

Regular scans by means of Greenbone (OpenVAS) are linked to the CMDB in NetBox or Ralph.

- Assets are updated on the basis of detection from LibreNMS and pi.Alert,
- incidents tied to vulnerabilities are linked into the management system (e.g. TheHive).

4. Proactive threat hunting (Threat Hunting)

Advanced queries in Kibana (KQL, Lucene) are used, often supplemented with:

- CTI data from OpenCTI or YETI,
- manual analysis in the case of suspicious entities,
- the results are documented as playbooks in BookStack or Wiki.js.

5. Creation of reports and dashboards

The following are regularly generated for management and customers:

- incident overviews in Kibana (Elastic Reports),
- visualizations of metrics and ticketing (linked with Grafana),
- exports for audit and SLA reporting.

6. Building the knowledge base

Each incident or new type of threat is documented in the form of:

- incident summary + lessons learned (TheHive),
- an entry in the knowledge portal (Wiki.js or BookStack),
- categorization according to MITRE ATT&CK and dates.

7. System maintenance and updating

The operations team has established:

- rules for component updates (Wazuh, ELK, TheHive),
- a test environment for verifying changes,
- centralized monitoring of the functionality of all modules.

These scenarios ensure the practical operation of SOCaaS as a service, with each of the components playing a clearly defined role in the architecture. At the same time, the documentation of incidents and analysis results contributes to the continuous improvement of detection and response.

4. MONITORING AND ANALYTICAL SYSTEM (SIEM – ELK STACK)

4.1. ELK architecture and components

The architecture of the ELK system (Elasticsearch, Logstash, Kibana) represents the core of the monitoring and analytical system designed within the INNOSOC SOCaaS. This system ensures centralized collection, processing, analysis and visualization of security events in real time.

Components of the ELK system:

- **Elasticsearch** – a distributed search and analytics engine that stores and indexes all received logs and data. It enables fast and complex queries over the stored data and is designed for horizontal scalability.
- **Logstash** – responsible for data ingestion, parsing and normalization. It supports a wide range of inputs (e.g. syslog, beats, REST API), a number of filter plugins (grok, mutate, geoip) and output to Elasticsearch.
- **Kibana** – a visualization tool that provides an intuitive interface for creating dashboards, browsing logs and the outputs of correlation rules. It serves as the main interface for SOC analysts and operators.
- **Beats** – lightweight agents (e.g. Filebeat, Metricbeat) that are installed on endpoints and send logs and metrics to Logstash or Elasticsearch.
- **Fleet & Elastic Agent** – a centrally managed system for agent management that makes it easy to deploy and configure Elastic Agent on end devices. Elastic Agent provides a unified method of data collection, EDR capabilities and integrations with SIEM modules.

For the SOCaaS deployment, the ELK system was designed as a multi-tenant, scalable solution that can serve multiple customers with separate data flows, dashboards and analytics. Elasticsearch enables the use of so-called index patterns or namespaces to separate tenants, while Kibana provides access on the basis of RBAC (Role-Based Access Control).

An important property of the architecture is high availability – the individual components (Elasticsearch nodes, Logstash pipelines, Kibana instances) are deployed in redundant mode and monitored via the Watcher system and Alerting mechanisms.

In addition to the basic components, supplementary modules are also used, such as Elastic Security (the former SIEM module), which provides advanced capabilities for managing detections, rules and detection alerts with support for the MITRE ATT&CK framework.

4.2. Processing of data flows

Within the ELK Stack architecture in the SOCaaS system environment, the processing of data flows plays a key role in the consolidation and standardization of data originating from various sources. These flows may include logs from end devices, network elements and security tools (e.g. EDR, CTI, SOAR), as well as metadata from cloud or hybrid environments.

The main tasks of data flow processing in the ELK Stack are:

- Receiving logs and events from multiple heterogeneous sources
- Validation, normalization and enrichment of data
- Assignment of timestamps, identifiers and event categories

- Redirection of data to appropriate indices for efficient analysis
- Filtering out redundant or irrelevant data
- Implementation of pipeline rules and processing modules (Logstash, Ingest pipelines in Elasticsearch)

In the INNOSOC environment, the following components are used for data processing:

- Filebeat: a lightweight agent deployed on end devices that collects and forwards logs.
- Logstash: the central data processing unit, enabling advanced data transformation.
- Elasticsearch Ingest pipelines: lighter processing rules integrated directly in the database core.
- Wazuh Manager: in the case of XDR, processes security events (alerts) and ensures their structuring.
- Integrations with MISP and TheHive: provide CTI enrichment with observables, contextual data and reputation scoring.

The integrity and availability of data are ensured by means of TLS encryption between components and centralized configuration management of the pipeline modules. In the multi-customer SOCaaS solution environment, data are separated on the basis of namespaces (index prefixes) or via logical separation using tagging during the ingest process.

4.3. Correlation rules and threat detection

Effective threat detection within the security monitoring of the INNOSOC SOCaaS is built on the use of correlation rules and analytical techniques that make it possible to uncover anomalies, indicators of compromise (IOC) and other behaviour pointing to potential security incidents. Given the use of the ELK Stack platform, this detection is implemented by means of several layers of logical analysis and rules applied to the received and processed logs.

Types of detection techniques

In the INNOSOC environment, threat detection is based on a combination of the following approaches:

- **Static signature-based detection** – uses precisely defined patterns (e.g. hash values, domains, process names), which are compared with CTI data obtained from MISP or OpenCTI. These signatures are automatically loaded into Elastic Security by means of integrations.
- **Correlation rules (rule-based detection)** – defined by means of KQL (Kibana Query Language), Lucene or using Elastic Detection Rules. They make it possible to combine several conditions (e.g. a combination of IP address, time window and action type) into complex detection scenarios.
- **Anomaly detection** – based on the Machine Learning modules available in paid versions of the Elastic Stack, which analyse deviations from normal behaviour (e.g. the number of logins outside working hours, non-standard transfer volumes). In the open-source environment, these approaches are simulated to a certain extent through metrics and threshold alerts.
- **Behavioural detection and MITRE ATT&CK mapping** – rules are created on the basis of TTPs known from the MITRE ATT&CK database, which enables systematic detection according to the phases of the attack life cycle. For example, a rule

monitoring “Persistence via Scheduled Task” checks whether a new schtasks.exe with an unknown parameter has been created.

Implementation of detection rules

In the INNOSOC environment, correlation rules are designed according to several scenarios:

- **Alerts based 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 data timing and volume.
- **Activities with administrative privileges** – e.g. adding a user to the “Administrators” group, ACL changes, running netsh or powershell.
- **Event chaining** – correlation between multiple logs over time: e.g. login → file transfer → execution of a malicious process.

These rules are managed via the **Elastic Security Detection Engine** module, where it is possible to:

- track detection alerts,
- set threshold values (thresholds),
- configure exceptions and ignored values (exceptions),
- export and import rules as JSON objects.

Automation of detection and response

Detection rules are connected to the response system by means of:

- **Elastic Actions & Connectors** – automatic sending of alerts to SOAR tools (TheHive), e-mail notifications or webhooks.
- **Integration with Wazuh** – makes it possible to generate alerts on the end-device side and transfer them to the Elastic environment, where they are further processed and visualized.
- **Integration with Cortex analysers** – makes it possible to run automatic enrichment tasks for observables.

Practical examples (from the INNOSOC laboratory environment)

During testing, dozens of detection rules were created, for example:

- A rule identifying an attempt at lateral movement via PsExec to multiple devices within a short time.
- Correlation between DNS queries for suspicious TLDs (e.g. .xyz, .top) and subsequent connection to unverified IP addresses.
- Creation of a “honeypot” dashboard with alerts on unauthorized access to specially prepared services.

Challenges and recommendations

- Correlation rules must be regularly updated in line with current threat trends.
- In a multi-tenant SIEM environment, it is necessary to ensure the separation of rules by customer, which requires working with tags, namespaces and separate pipelines.
- The operation of high-accuracy detection rules depends on the quality of the input data and a correct normalization process (Logstash, Beats).

4.4. Integration with other components

The effective functioning of the SOCaaS solution depends on the ability of the individual components to cooperate with one another. The SIEM system, as the central analytical core (in our case the ELK stack), must be interconnected with the other tools for data collection, incident management, threat intelligence and reporting in order to create a unified ecosystem for threat detection and response.

Connection to data sources

The SIEM obtains data from various infrastructure components by means of agents and integrations that ensure uniform formatting and transport of data. In our architecture, the following are mainly used:

- **Wazuh**, as the XDR agent, ensures the collection of logs from workstations, servers and devices, and sends these data to Logstash.
- **Logstash and Beats**, as the transport layer, perform pre-processing and normalization of data before they are stored in Elasticsearch.
- **The asset management system (e.g. NetBox or Ralph)** provides contextual information about the infrastructure and identified devices. These data can be integrated via an API or synchronized as part of alert enrichment.
- **The vulnerability scanner (e.g. Greenbone)** shares scan results and identified vulnerabilities, which are imported as records into the SIEM and serve for the creation of correlation rules.
- **CTI platforms (e.g. MISP, OpenCTI)** provide external IOCs and TTPs, which are used in correlation rules and in alert enrichment.

Integration with incident management and SOAR tools

Alerts detected in Elasticsearch are, as a rule, automatically or semi-automatically transformed into incidents by means of integration with systems such as:

- **TheHive** – as the main incident management system, it receives alerts via webhooks or by means of connectors (e.g. ElastAlert → TheHive Connector). A case contains observables (e.g. IPs, domains, hashes) and enables both their manual and automatic enrichment.
- **Cortex** – serves as the SOAR component for response automation. After an incident arrives in TheHive, actions such as a WHOIS query, a VirusTotal scan or device isolation can be triggered automatically.

- **Zammad or RTIR** – if more advanced ticket recording is required, alerts can also be converted into tickets in these systems, while the link to the analytical data from Kibana is preserved.
- **Cooperation with reporting and dashboards**
- **Kibana** provides an interface for tracking alerts, analysing events, creating dashboards and reporting. Users can interactively filter data according to various criteria (time, incident type, source, IP address).
- Reporting is served by scheduled exports (PDF/CSV) and dashboard snapshots, which can be forwarded by e-mail or published on internal portals.
- The results of incident management and enrichment from CTI are often visualized precisely through Kibana – e.g. as incident timelines, a map of the geographical origin of IPs, or the top threats over the last month.

Knowledge base and playbooks

- Incidents and alerts that have arisen, and their resolutions, are documented and archived in the **knowledge portal** (e.g. BookStack, Wiki.js), which ensures retrospective traceability and the creation of so-called SOC playbooks.
- The link between SIEM alert → incident management → knowledge base creates a consistent flow of information, thereby shortening response time and reducing the workload of analysts.

5. CREATING DASHBOARDS IN KIBANA

5.1. Dashboard design principles

Dashboards are a key tool enabling analysts to quickly find their way around data on the security situation. In the SOCaaS environment, they act as the visual layer above the SIEM system, which makes it possible to track trends, alerts, attacker activity and the performance of security components in real time. In our design, the dashboards are implemented in the **Kibana** environment, which is a native part of the ELK Stack.

Design objectives and philosophy

In designing the dashboards within SOCaaS, we followed these principles:

- **Clarity and purposefulness** – each dashboard is designed for a specific purpose (e.g. alert monitoring, vulnerability overview, anomaly detection) and tailored to the target role (L1 analyst, SOC manager, auditor).
- **Minimalism and clarity** – we display only relevant metrics and indicators while avoiding visual overload. Each panel has a precisely defined function.
- **Interactivity and drill-down** – we enable clickable transitions between the aggregated overview and detailed events. Users can filter by time, source, IP address or incident type.
- **Real-time updating** – dashboards display current data with the option of live refresh (e.g. every 15 seconds), which is important during active incidents.
- **Unification of data from multiple sources** – via Beats and Logstash, we obtain data from various components (EDR, network devices, asset management systems) and display them in a unified visualization layer.
- **Export and sharing options** – dashboards are designed so that they can be exported to PDF or PNG or shared as a link, or embedded in reporting templates.

Typology and hierarchy of dashboards

To maintain clarity, the dashboards within SOCaaS were 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, the breakdown by severity (Critical, High, Medium), time trends, and correlations between multiple sources.
3. **Thematic dashboards** 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**: access, ports, detected scanning activities.
4. **Detailed analytical dashboards** – Used by L2 analysts and for threat hunting. They contain heatmaps, time slices, distributions by hour, country, ASN, occurrence of IOCs and the like.
5. **Ad hoc and incident-specific dashboards** – Created during the investigation of a specific case, where the analyst builds their own visualization based on data from Elasticsearch.

Templates and reusable panels

During development, **panel templates** (so-called reusable components) were also created, which enable:

- Reuse of common visualizations (e.g. timeline, top sources, IP distribution).
- Creation of dashboards based on a template for a new project or customer.
- A unified visual style for the entire SOCaaS.

5.2. Structure of typical dashboards

In the Kibana environment, dashboards are composed of individual panels, which 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.

The basic structure of dashboards is typically based on the following division into sections:

- Main panel with summary metrics: number of incidents, system status, alerts by severity.
- Time trends: histograms capturing the number of incidents or alerts over time.
- Breakdown by type: attack types, threat categories, detected IOCs (Indicators of Compromise).
- Interactive filters: enable selection by IP address, time range, source system or geolocation.
- World map (if geo-IP is enabled): display of the geographical distribution of attacks.
- Panels with drill-down capability: e.g. clicking on an IP opens a list of events or alerts related to the given source.

5.3. Examples of dashboards and visualizations

Effective data visualization in the SOC environment is crucial for the rapid identification of incidents, detection of trends and decision-making by analysts. Within the INNOSOC SOCaaS solution, several types of dashboards were designed and implemented that cover various aspects of monitoring and incident management. These dashboards were developed in the **Kibana** tool, which forms part of the selected SIEM solution based on the **Elastic Stack (ELK)**.

5.3.1. System security overview

An initial view of the overall security status of the system is provided by the so-called overview dashboard, which summarizes the main metrics:

- Number of logs received over time (log rate)
- Number of active alerts (detections)
- Status of Wazuh agents/Elastic Agents
- Number of detected IOCs over time

5.3.2. Correlation and incident dashboards

Correlation rules detect threats on the basis of logic that links events from multiple sources. The result is incident alerts, which are displayed in real time. The dashboards display:

- Time distribution of detected alerts by type
- Criticality and classification of incidents

- Geolocation distribution of attackers' IP addresses
- Trends of the most frequent attacks (e.g. brute-force, port scan)

5.3.3. Visualization of network traffic and access

An overview of the activity of users, endpoints and network activity is crucial. The relevant dashboards include:

- The most frequent destination ports and source IP addresses
- Anomalous requests (e.g. HTTP/HTTPS, DNS)
- Resource utilization by services and applications
- Access to protected segments

5.3.4. Monitoring of authentications and user behaviour

Suspicious behaviour (e.g. a login from an unusual geographical location, repeated failed logins) is identified via:

- Failed/success login attempts
- Anomalies in login frequency
- Account usage by roles and systems
- Detection of unauthorized activities

5.3.5. Integrated overview for the SOC analyst

The final type of dashboard combines metrics from several areas into a single interactive interface for the analyst's day-to-day work:

- Widgets for searching by IP/hash/domain
- Quick filters for time, criticality and incident type
- Aggregated summaries (incidents in 24h, top attackers)
- Click-through to detailed logs or cases in the ticketing system

6. REPORTING AND DATA EXPORT

Effective reporting is one of the key aspects of operating a security operations centre (SOC). It enables not only retrospective analysis of incidents and trends, but also transparent communication with stakeholders (internal management, customers, audit, supervisory authorities).

Within the INNOSOC SOCaaS, reporting is implemented by means of the **Elastic Stack**, primarily using the **Kibana** tool, which enables the creation, management and distribution of overviews in various formats and levels of detail.

6.1. Reporting objectives and types

Reporting in SOCaaS fulfils several basic objectives:

- **Operational overview** – display of the status of systems, the number of incidents and agent activity.
- **Security reporting** – outputs on detected threats, attack types and trends.
- **Compliance reporting** – reports according to regulatory requirements (e.g. GDPR, NIS2).
- **Management reporting** – aggregated statistics and indicators for the organization's senior management.
- **Customer reporting** – individual overviews for the individual clients of the multi-tenant SOCaaS.

6.2. Reporting options in Kibana

Kibana provides several options for the creation and distribution of reports.

6.2.1. Manual export of reports

Each dashboard in Kibana can be exported 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.

These options enable flexible use of the outputs for various purposes – from attachments to reports, through technical documentation, to input for other analytical tools.

6.2.2. Automated reporting

Kibana (in conjunction with Elastic X-Pack or open-source extensions) supports the scheduling and automated sending of reports:

- **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.

6.2.3. Customization of outputs

- **Option to select metrics, fields and time ranges**
- **Design personalization – logos, header, colour scheme**
- **Inclusion of explanatory notes, remarks and contextual commentary**

Kibana enables each analyst to prepare reporting tailored precisely to their needs – according to the target group (SOC operator, manager, customer) and the type of information.

6.3. Integration of the reporting system

Reporting within SOCaaS is not an isolated component – it is interconnected with other parts of the architecture:

- **SIEM (ELK)** – provides the primary data, alerts and correlations.
- **SOAR and Ticketing** – make it possible to include statistics on incidents, resolution time and responses.
- **EDR/XDR (Wazuh)** – provides endpoint data and events from devices.
- **CTI and Threat Feed integrations** – enrichment with external threats.
- **CMDB / Asset Management** – extension of the context with relationships to specific assets.

6.4. Challenges and recommendations

In the design and operation of reporting, it is important to:

- Define the **target groups and their needs** – technical vs. non-technical reporting.
- Ensure **automation and regularity** – manual reporting is unsustainable.
- Create **interactive dashboards** as a basis for live presentations.
- Ensure **auditability and data integrity** – important for regulatory compliance.
- Distinguish between **internal and external reporting** – content, level of detail, form.

7. CONCLUSION AND RECOMMENDATIONS

The design of the security operations centre as a service (SOCaaS) in the INNOSOC project is based on a thorough analysis of the components, processes and tools that are essential for the effective operation of multi-tenant security monitoring. The knowledge gained from laboratory verifications, test deployments and the synthesis of scientific knowledge was transformed into the design of a comprehensive solution that respects the requirements for openness, flexibility and modularity.

7.1. Benefits of the proposed architecture

- **Modularity**: The architecture enables independent deployment and replacement of components without disrupting the overall flow.
- **Open-source technologies**: The selection of tools such as ELK, Wazuh, TheHive, MISP, Kibana, Grafana and others makes it possible to reduce costs while maintaining a high degree of configurability.
- **Integration**: The key components (SIEM, SOAR, CTI, EDR, Asset Management) are interconnected by means of APIs and integration mechanisms, which enables a smooth flow of data.
- **Adaptability for a multi-tenant environment**: The architecture is designed to support the management of multiple customers with different service levels.
- **Scalability**: All the selected tools enable horizontal scaling, which supports the growth and expansion of the service without the need for fundamental architectural changes.

7.2. Reasons for selecting the ELK Stack as the SIEM core

- The **Elastic Stack (ELK)** was selected as the core of the SIEM solution for its ability to process large volumes of data in real time, its extensive community support, the availability of supplementary tools and its excellent visualization layer (Kibana).
- Another advantage of ELK is its natural integration with Wazuh (EDR), TheHive (Incident Management) and CTI tools (e.g. MISP), which creates a unified data environment for analytics and response.

7.3. Dashboard and reporting solution

- The **Kibana** tool was used not only for data visualization, but also for implementing basic reporting and the presentation of incidents, infrastructure states, alerts and performance metrics.
- Dashboards were designed according to the roles in the SOC: L1 analyst, L2 analyst, lead operator, customer view, technical support.
- Reporting was implemented with the option of exports (PDF, CSV, JSON), as well as scheduled automatic sending using Elastic Stack extensions.

7.4. Challenges and recommendations for further development

- **Introducing CI/CD for security rules**: It is recommended to introduce a process of continuous integration and verification of correlation rules and detections.
- **Increase the automation of responses (SOAR)**: Deepen the use of TheHive + Cortex and the connection of playbooks in StackStorm or Shuffle.

- **Make knowledge management more efficient**: Fully integrate Wiki.js or Bookstack as the SOC knowledge base, linked to incidents and cases.
- **Extend reporting for management and customers**: Prepare predefined reports focused on SLAs, response durations and incident categorization.

SOURCES

Primary documentation of tools and platforms

1. Elastic Stack Documentation. Elastic.co. https://www.elastic.co/guide/
2. Wazuh Documentation. Wazuh Inc. https://documentation.wazuh.com/
3. OpenCTI Documentation. https://www.opencti.io/en/docs/
4. MISP Project Documentation. https://www.misp-project.org/documentation/
5. IntelMQ Project. CERT.at. https://github.com/certtools/intelmq
6. TheHive Project Documentation. https://docs.thehive-project.org/
7. Cortex Analyzers. https://github.com/TheHive-Project/Cortex-Analyzers
8. NetBox Documentation. NetBox Labs. https://docs.netbox.dev/
9. LibreNMS Documentation. https://docs.librenms.org/
10. GLPI Documentation. https://glpi-project.org/
11. Greenbone Community Edition (GCE). https://docs.greenbone.net/
12. OpenSCAP Official Site. Red Hat Security. https://www.open-scap.org/
13. BookStack Documentation. https://www.bookstackapp.com/docs/
14. Wiki.js Documentation. https://docs.requarks.io/

Supplementary professional and academic sources

15. Bhat, M. A., & Kumar, P. (2022). *Advanced SIEM for Enhanced Threat Detection Using Machine Learning*. In 2022 IEEE International Conference on Cyber Security and Resilience (CSR), IEEE. https://doi.org/10.1109/CSR54599.2022.9850371
16. Nguyen, K., et al. (2021). *A Review of Security Information and Event Management (SIEM) Systems*. Future Internet, 13(8), 204. https://doi.org/10.3390/fi13080204
17. Munoz, M., Garcia, C., & Valdes, J. J. (2023). *ML-based Optimization for SOC Event Triage*. Proceedings of the 2023 ACM Workshop on AI in Cybersecurity. https://doi.org/10.1145/3576888.3602857
18. Akinrolabu, O. A., et al. (2020). *Security Operations Centre: A systematic study and open challenges*. Computers & Security, Volume 92, 101747. https://doi.org/10.1016/j.cose.2020.101747
19. Al-Shaer, E., et al. (2021). *Autonomic Security Operations: Challenges and Research Opportunities*. In IEEE Transactions on Dependable and Secure Computing. https://doi.org/10.1109/TDSC.2021.3127814
20. Kertesz, A., & Kiss, B. (2025). *Cloud-native Architectures for SOC-as-a-Service Platforms*. Journal of Cloud Computing, 14(2), 117–139.
21. Papadopoulos, K., & Vlachos, I. (2025). *A Federated Model for Sharing Threat Intelligence Across SOCs*. Journal of Cybersecurity and Privacy, 5(1), 54–79.
22. Zhang, Y., et al. (2025). *Visualization and Reporting Techniques in Elastic SIEM Environments*. Proceedings of the 2025 International Conference on Networked Systems. [DOI pending]

ANNEXES

This chapter brings together supplementary materials and illustrations that complement the main parts of the document and provide an overview of the practical aspects of the design, testing and verification of the individual components of the INNOSOC SOCaaS system. It

contains visualizations, diagrams, dashboard screenshots, configuration examples and other supporting outputs from the laboratory deployment and the research activities of the INNOSOC project team.

These annexes serve as supporting documentation for all the main parts of the document and demonstrate the readiness of the components for real-world deployment in the operation of a multi-tenant SOCaaS environment within the INNOSOC project. At the same time, they confirm that the choice of technologies and the design of the integration architecture were validated not only theoretically but also practically during iterative testing of the solutions in a controlled environment.

List:

1. podklad pre V6 v rámci KPB2_Koncept SOC centra.docx
2. podklad pre V6 v rámci KPB2_Dashboardy - vizualizácia dát z nástrojov.docx
3. podklad pre V6 v rámci KPB2_Dashboardy - ukážka tvorby vizualizácií v Kibane.docx
4. podklad pre V6 v rámci KPB2_Dashboardy - analýza bezpečnostných udalostí.docx
5. podklad pre V6 v rámci KPB2_Dashborady - ukážka vizualizácií pre analýzu sieťových tokov.docx