VSS SW components
ANNEX 1 – Recommendations of the workshop from 1 – list / security of architecture – APPS

Funded by the European Union NextGenerationEU | RECOVERY AND RESILIENCE | GOVERNMENT OFFICE OF THE SLOVAK REPUBLIC | UNIVERSITY OF ŽILINA Faculty of Management Science and Informatics | BRAIN:IT

List of components

Component name	Application	Version	Licence	Primary deployment	Application layer
ELK Stack	SIEM	Release 8.x	MIT License	Docker / OpenStack (PoC)	Docker / Docker Compose
Shuffle	SOAR	Release 2.1.0	GNU Affero General Public License	Docker / OpenStack (PoC)	Docker
Wazuh	XDR	Version: 4.14	GPL-v2	Docker / OpenStack (PoC)	Docker
Greenbone Community edition	Vulnerability scanner	Version: 24.10	Creative Commons Attribution Share Alike 4.0	Docker / OpenStack (PoC)	Docker
OpenCTI	CTI	Version: 6.9.0	Apache 2.0 license	Docker / OpenStack (PoC)	Docker Compose
MISP	CTI	Version: 2.5.28	AGPL-3.0 license	Docker / OpenStack (PoC)	Docker Compose
GLPI	Asset management	Version: 10.0.16	GPL-3.0	Docker / OpenStack (PoC)	Docker
NetBox	Asset management / IPAM/DCIM	Version: 4.0	Apache 2.0 license	Docker / OpenStack (PoC)	Docker
DFIR-IRIS	Ticketing	Version: 2.4.20	LGPL-3.0 license	Docker / OpenStack (PoC)	Docker
DFIR-IRIS	Incident recording and management	Version: 2.4.20	LGPL-3.0 license	Docker / OpenStack (PoC)	Docker
Kibana (ELK)	Reporting	Version: 8.2.3	Apache 2.0 license	Docker / OpenStack (PoC)	Docker
Wiki.js	Knowledge database	Version: 2.5	AGPL-v3 License	Docker / OpenStack (PoC)	Docker

Description of tools

Area / Function (SOC)	Framework (product)	Edition / Version	Prerequisite SOC unit	Main functions (capabilities)	Input data (data sources)	Protocols / interfaces	Interactions / API	Environment	Data structure / collection	Access & RBAC	Operation (notes)	Notes / Risks
SIEM	ELK Stack (Elasticsearch + Logstash/Beats + Kibana)	depending on the implementation	Central aggregation, indexing and correlation of logs + search and analysis	Data collection (Logstash/Agent/protocol/plugin/etc.), normalization, event correlation, alerting, indexing and search (Elasticsearch database), visualization (Kibana dashboards), scalability, log retention and archiving	System logs, network logs, logs from cloud systems, endpoint logs, logs from API systems, application logs, NetFlow, IPFIX, EDI systems, syslog, Windows events, Wazuh agents	Indexes/normalized logs, detection rules, alerts, dashboards, reports, data exports	REST API, Beats/Agent, syslog, webhooks (depending on the source)	self-hosted (VM/Docker/K8s)	Local DB (Elasticsearch cluster), JSON/logs (indices), configurable retention	User roles, rights to indices/tasks/dashboards, rights to manage/status, rules	SOC (main (security analysts) + SOC manager	Storage/performance demands, index/life cycle and backups needs to be addressed
SOAR	Shuffle	depending on the implementation	Orchestration and automation of responses (playbooks), interconnection of the individual components in SOC/ecosystem	Tool orchestration (API integrations with the other SOC components), automation of steps and responses (workflows/playbooks)	Alerts from SIEM/Wazuh/OpenCTI/MISP/etc., alerts from EDR systems, data from CTI systems (OpenCTI/MISP), tickets from DFIR-IRIS, IRIS webhooks	Triggered actions (responses), execution records, created and resolved incidents/tickets, playbook outputs	REST API, webhooks, connectors	self-hosted (Docker/K8s)	Database (SOAR format), configurable retention	Roles for creating/running workflows and playbooks, managing integrations	SOC security engineer + SOC manager	Important to secure access to API keys; higher load with parallel playbooks, single point of failure for automation
EDR	Wazuh	depending on the implementation	Endpoint monitoring/detections and basic response/+ sending events to the SIEM	Endpoint protection and monitoring, HIPS/HIDS functionality (FIM, sometimes), log analysis, system auditing, taking basic responses/steps	Endpoint telemetry (processes, services), OS logs, application logs, FIM, integrity status	Security alerts, endpoint/agent/incident events, telemetry to the SIEM, information on the status of end devices	Wazuh API, integration with Elastic/Logstash, syslog	self-hosted (manager + agents, VM)	Wazuh indices (JSON format), database, configurable retention (aligned with the SIEM)	Roles/Users (Wazuh/ULM/Elastic), rights to read alerts, investigation, rights to manage agents, rules	SOC team (SOR security specialist) + SOC security engineer	Higher load on end devices (endpoints), large volume of logs, privilege/settings on end hosts
Vulnerability scanner	Greenbone Community Edition (GVM)	Community	Regular scanning of infrastructure/systems and vulnerability reporting, identification of weaknesses in systems + supporting prioritizing/risk (patching)	Network scanning, CVE scanning and detection, vulnerability reporting, prioritization, vulnerability database (NVT/CVE)	Asset information (IP addresses, ranges, hostnames), CVE/NVT feeds, credentials (if used)	Vulnerability findings, reports, exports, CVE/CVSS, hosts, risk evaluation (CVSS)	GMP (IP), exports (CSV/XML), connection via automation	self-hosted (VM/Docker)	Database/local scan results, (SCAN/XML format configurable retention	Users/roles in GVM, rights to read reports, rights to run scans,	SOC team (vulnerability specialist) + SOC manager	The Community edition may have limitations in updates/convenience, scans/windows and network impact must be tested, possible load on
CTI	OpenCTI	depending on the implementation	Collection and analysis of threat information, provision of context for individual incidents, Management, correlation and enrichment of alerts/incidents.	Threat modelling (TTX), data enrichment, analysis and visualization of relationships (attack chains/graphs), relationships/threat actors/campaign tracking, TAXII/API	External CTI feeds (TAXII, API), MISP, manual entry from external sources, OSINT sources	Threat model (CTI entities, links, relationships), enriched IOCs, reports (STIX, TAXII), visualizations (graphs)	REST/GraphQL API, connectors, STIX/TAXII (depending on modules)	self-hosted (Docker/K8s)	Database (e.g. Elastic/Neo4j depending on the setup), configurable retention	Roles (RBAC)+ TLP restrictions	SOC team (CTI specialist/ CTI analyst)	Important to set source quality and deduplication, alerts storage overload.
CTI	MISP	depending on the implementation	Collection and analysis of threat information, provision of context for individual incidents, Management, correlation and enrichment of alerts/incidents.	Sharing of attributes/IOCs, real-time event based CTI, automatic import/export of information (CTI feeds), tagging, organization and community sharing of information	IoC feeds, external sources (OpenCTI platform), malware samples, manually entered inputs	REST API, exports (CTI), , possibly TAXII (depending on configuration)	REST API, exports (CTI), possibly TAXII (depending on configuration)	self-hosted (VM/Docker)	Database of events/attributes (IoC events - JSON format), configurable retention	Role-based access (RBAC), sharing rules, TLP restrictions	SOC team (CTI specialist/ CTI analyst)	Sensitive to correct sharing settings (what may be shared externally),
Asset management	GLPI	depending on the implementation	Asset records, CMDB/ITSM records, links to scanners and sensors	Inventory of hardware/software, CMDB, asset life cycle management, ability to search for 'new' assets (discovery, plugins), connection to third-party system, connections, user, ownership and organizational relationships	Discovery tools, manual inputs/scanners, network discovery, software asset management systems (Optional)	List of assets, owners, relationships (hardware/software/CMDB data, attributes for associations/links)	REST API, import/export	self-hosted (Docker/K8s)	Database relational data (CMDB records), retention configurable according to internal rules	Roles for individual access types and permissions, rights to manage assets	IT/asset team/ SOC security engineer (admin)	CMDB quality is key for scans and incident correlation, need for integration/synchronization problems and its integrations.
Asset management / IPAM/DCIM	NetBox	depending on the implementation	Recording and management of assets, enriched with network information/asset management	Management of network assets and sensors, IPAM, DCIM, up-to-date centralized inventory (Source of truth), network topology modelling and visualization, exports	Discovery tools, DHCP, manual inputs, integration (Ansible, Terraform), CMDB (GLPI sync)	IP ranges, network topology, infrastructure inventory, infrastructure mapping for the SOC, API value source (GLPI sync)	REST API, webhooks	self-hosted (Docker)	Database relational data (IP, devices, topology), retention configurable according to internal rules	Roles for individual access types and permissions, rights to manage assets	Network/Active team, IT/asset team/ SOC security engineer (admin)	Must be kept up-to-date - otherwise inaccurate/misleading data from target
Ticketing	DFIR-IRIS	depending on the implementation	Management of SOC tickets, requests and tasks (triage, escalations, SLA), assignment of tasks and responsibilities	Case/ticket management, incident workflow, task management, incident assignment and escalation, integration with SIEM and SOAR, inputs to audit	Incident data (alerts) from SIEM/SOAR, evidence (logs), CTI context, EDR alerts, manual creation of inputs	Tickets, escalation state, audit trail, tasks and workflows	API, webhooks (when possible), integrations with SOAR	self-hosted (VM/Docker)	Case database, JSON format + relational data, configurable retention	Roles (analyst/lead/admin), rights to read incidents, rights to work on an incident case, approval rights	SOC team (security analysts) + SOC manager	Consistent naming and workflows (state models) are important, load with a large number of tickets
Incident recording and management	DFIR-IRIS	depending on the implementation	Central record of incidents and evidence (case management), management of timeline/alert life cycle	Incident recording, incident life cycle management, retention of evidence and artefacts, incident timeline, chain of custody, incident overview	Reports from SOAR/Wazuh/Greenbone, tickets, evidence from investigations, CTI context, EDR verdicts	Incident write-ups, records, evidence, reports for audit	API, import/export	self-hosted (VM/Docker)	Database/+ attachment storage (incident records, + evidence), retention according to policy	Roles (analyst/lead/admin), rights to read incidents, rights to work on an incident case, approval rights	SOC team (security analysts) + SOC manager	Role full recording evidence and sensitive data must be/not, possible overload from large volumes of data
Reporting	Kibana (ELK)	as per ELK	Visualizations, overviews, dashboards and reports for the SOC/team, management and audit	Real-time and historical visualizations and dashboards (SIEM/XDR), report generation, trend analysis, alert/incident statistics, role-based access, multi-tenant	Indices in Elasticsearch, alerts (SIEM), EDR data, CTI data, vulnerability scans,	Visualizations and dashboards (shared/ read-only), , report exports	REST API (Kibana/Elastic), integration within ELK	part of ELK (VM/Docker/K8s)	Data in the Elastic search database: retention according to the SIEM policy	Roles and permissions (inherited in Elasticsearch, dashboard rights	SOC team (security analysts) + SOC manager	Beware of access to sensitive indices, access to SIEM data, separate views for different
Knowledge portal	Wiki.js	depending on the implementation	Knowledge base (runbooks, procedures, playbooks, onboarding), support for knowledge sharing within the	Management of documentation, page versioning, RBAC access, pre-prepared templates, full-text search,	Documentation of SOC processes, incident procedures, SOAR playbooks, reports	Documentation (runbooks/playbooks), information from previous incidents	GraphQL API, git sync (Format)	self-hosted (Docker)	Database/+ files (Markdown/HTML documents), retention according to internal rules	Roles (read/edit/admin), read and write rights	SOC team (security analysts) + SOC manager	Maintain a "single source of truth", introduce a review process (e.g. quarterly)

Integration flow

Area / Function (SOC)	Component (product)	Data flow	Communication flow (Source → Target)	Mechanism	Format / payload	Trigger / Frequency	Output
SIEM	ELK Stack (Elasticsearch + Logstash/Beats + Kibana)	Flow into SIEM: logs, network data, IoCs from CTI Flow from SIEM: alerts → SOAR/incident response (DFIR-IRIS), alerts → SOAR (reporting)	Flow via AGENT: Wazuh agents + Wazuh → ELK / CTI → ELK/ network devices → ELK Flow from SIEM: ELK → SOAR	Log forwarding (syslog/Beats), REST API, Elasticsearch index access, webhooks	JSON format, ECS (Elastic Common Schema), CSV, NetFlow/IPFIX	Real-time	Indexed events, centralized and correlated logs, alerts + dashboards
SOAR	Shuffle	Flow into SOAR: ELK + Shuffle(SOAR), Wazuh (XDR) → Shuffle, CTI → Shuffle, vulnerability scanner → Shuffle Flow from SOAR: Shuffle → SIEM/network infrastructure, action → incident/enriched data from CTI → incident response	REST API, webhooks, triggers	JSON/HTTP request, JSON,	Real-time/event-driven (when an alert is received from SIEM/XDR)	Automated responses across the entire SOC cycle
EDR	Wazuh	Flow into EDR: endpoint telemetry from end devices Flow from EDR: alerts → SIEM/SOAR, logs → SIEM,	Flow from EDR: Wazuh (EDR) → ELK (SIEM)	Agent-based communication, log forwarding, REST API	JSON (alert), syslog	On alert (rule triggered) - real-time monitoring	Endpoint alerts, response and automatic reaction on end devices
Vulnerability scanner	Greenbone Community Edition (GVM)	Flow into CTI: assets from the asset management system, CVE/NVT feeds Flow from SE: vulnerability reports → SIEM/Violating +	Flow into SE: GLPI/NetBox → GVM Flow from VS: GVM → ELK, GVM → Shuffle	REST API, , feed synchronization, report, scheduled scans	STIX like entity/IOC query, XML, JSON, CSV, CVE/NVT formats	On-demand via a playbook, scheduled scans	Enriched data for the incident, list of vulnerabilities, risk scores
CTI	OpenCTI	Flow into CTI: external feeds, IoCs from other sources/CTI feeds, incident data from SIEM/ incident response Flow from CTI: enriched data → SIEM/SOAR/ incident	Flow into CTI: MISP/ other CTI → OpenCTI, external sources → OpenCTI, ELK/ SIEM/ XDR → OpenCTI Flow from CTI: CTI → ELK, CTI → Shuffle, CTI → MISP/ other CTI	REST API, GraphQL API, STIX/TAXII, connectors	JSON, GraphQL, TAXII, STIX	On-demand via a playbook, on a relevant alert	Enriched data for the incident, reputation/tags/IOC created
CTI	MISP	Flow into CTI: external/other/ intel feeds, IoCs from other sources/CTI feeds, incident data from SIEM/ incident response Flow from CTI: enriched data → SIEM/SOAR/ incident	Flow into CTI: MISP/ other CTI → MISP, external sources → MISP, SIEM/ DFIR/XDR → MISP Flow from CTI: CTI → ELK, CTI → Shuffle, CTI → OpenCTI/ other CTI	REST API, TAXII, feed import/export	JSON, MISP format, STIX, CSV, OpenIOC	On-demand via a playbook, on a relevant alert	Enriched data for the incident, reputation/tags/IOC created
Asset management	GLPI	Flow into SE/system: network devices/endpoints → GLPI Flow from SE/system GLPI → vulnerability scanner, GLPI → Shuffle/ DFIR-IRIS	Flow into SE/system: network devices/endpoints → GLPI Flow from GLPI/system GLPI → vulnerability scanner, GLPI → Shuffle/ DFIR-IRIS	REST API, discovery agents, ADLDAP integrations, import/export	JSON, CSV, LDAP/ADobjects	Real-time, synchronization, scheduled discovery	List of assets and their relationships, CMDB data
Asset management / IPAM/DCIM	NetBox	Flow into SE/system: current data on the status of the network and infrastructure, information on the Tenth/Lifecycle/data, information on users/ links to assets, topology) information Flow from SE/system: asset information → SIEM/SOAR/ vulnerability scanner/incident response	Flow into SE/system: network devices/endpoints → NetBox Flow from SE/system NetBox → vulnerability scanner, NetBox → ELK/Shuffle/ DFIR-IRIS	REST API, webhooks, , import/export, automated integrations (Ansible/Terraform)	JSON, YAML, , CSV	Real-time, synchronization, scheduled discovery	List of assets and their relationships, infrastructure mapping, topology, IP/DHCP/Contacted
Ticketing	DFIR-IRIS	Flow into Ticketing: alerts from SIEM, outputs from SOAR, enriched data from CTI, data from XDR/vulnerability scanner, data entered manually by the user Flow from Ticketing: incident/s/tickets/ tasks → incident	Flow into Ticketing: ELK → Shuffle → DFIR-IRIS Flow from Ticketing (DFIR-IRIS) (internally within the system), DFIR-IRIS → SIEM	REST API, webhooks, internal workflow	JSON, HTTP payloads, various/mixed/ attachments (various formats)	After receipt of an alert from SIEM/SOAR, or manual creation of a ticket	Created ticket, prioritization of individual tickets, assignment of responsibility for resolution
Incident recording and management	DFIR-IRIS	Flow into: Mixed alerts from SIEM, outputs from SOAR, enriched data from CTI, data from SIEM/vulnerability sources, trigger from the ticketing system Flow from Mixed: reports → reporting/ knowledge portal, updated status of the user ticket in the ticketing system, actions performed in the framework automated/performed manually by an administrator after the case has been evaluated by a SOC analyst	Flow into IR (user) DFIR-IRIS (internally within the system) ticketing → incident response Flow from/Mixed: DFIR-IRIS → Kibana, DFIR-IRIS → Wiki.js	REST API, workflow engine, case management system	JSON, PCAP, screenshots, logs	On ticket creation, SOAR trigger	Investigated incident, reports produced on the occurrence and investigation of the security incident
Reporting	Kibana (ELK)	Flow into: reporting: data from SIEM/XDR/vulnerability scanner/ incident response tool Flow from reporting: dashboards/reports → SOC management/ stakeholders	Flow into requesting: ELK (Elasticsearch) → ELK (Kibana), Wazuh → Kibana, SIEM → Kibana, DFIR-IRIS → Kibana Flow from reporting: Kibana → SOC team/ management	Interaction in the stack (Elasticsearch queries), REST API, WEB UI	Visualizations - JSON, Elasticsearch DLS, CSV/PDF reports	Continuous/at regular intervals, throughout the entire period of SOC operation	Visualizations and dashboards, operational reports, post-incident analysis reports
Knowledge portal	Wiki.js	Flow into: IR/ playbooks from SOAR, acquired knowledge/ procedures/ lessons learned from incident response Flow from IR/ procedures → SOC team (analysts),	Flow into IR: Shuffle → Wiki.js, DFIR-IRIS → Wiki.js Flow from IR: Wiki.js → SOC team (analysts), Wiki.js → Shuffle → DFIR-IRIS	REST API, WEB UI, GIT integration	Text/markdown - Markdown/Language: HTML, JSON	After closure of a security incident/security event	Updated incident handling procedures