ACSIA XDR Plus Architecture V6.x.x Technical Components

This document describes the design and architecture of the ACSIA XDR Plus security solution and its major components.

1. Base Platform

The ACSIA application engine runs on major Linux distributions and can be deployed on a physical, virtual, container or cloud platform. For the minimum/recommended hardware/software requirements, we suggest you check our latest installation guide, which you can find inside our knowledge base.

2. ACSIA Engine

The core analytical engine of ACSIA is written in the Java Spring framework that includes Spring Boot, Spring Data & Spring Rest.

3. ACSIA Frontend

The front end is a REST API interface that makes backend calls using React.js and the Chakra UI library.

4. ACSIA XDR Plus Security

ACSIA comes with the following integrated security components:

- OAuth2 - Secure delegated access
- Multi-factor authentication - 2-factor authentication
- TLS for secure communication across clients
- Pwgen - Strong random password generator

5. ACSIA Open Source Toolstack

There are several open-source tools used by the ACSIA analytical engine, including the following:

Ansible	Bc	Binds-utils	Docker
Dsnif	ElasticBeats	Falco	Httpd-tools
Logstash	Lucene	MySQL	OpenDashboard
OpenSearch	OSQuery (Agent)	OSSec	Postfix
Python pip	RabbitMQ	Sysmon (Agent)	Sysstat
Unbound (Agent)	Wazuh	Wget	Whois

6. ACSIA Virtualization Method

ACSIA uses Docker (Linux Containers), an operating-system-level virtualization method for running multiple isolated Linux systems (containers) on a control host using a single Linux kernel.

7. ACSIA Configuration Management System

ACSIA uses Ansible playbooks for automating software provisioning, configuration management, and application deployment.

8. ACSIA Databases

ACSIA uses two separate databases, which are segregated for security design reasons as follows:

- MariaDB/MySQL
- MongoDB (OpenSearch)

9. ACSIA Message Broker

ACSIA architecture includes RabbitMQ, the most widely deployed open-source message broker.

10. SIEM Environment

ACSIA uses Elastic Stack tools for its SIEM (Security Information and Event Management) centralized log collector along with the following tools:

- OpenSearch
- OpenDashboard
- Lucene
- Logstash
- Wazuh
- ElasticBeats (log shippers)
- OSSEC agents (log shipper)

11. ACSIA Firewall

ACSIA comes with an embedded host-based firewall. The firewall has innovative features such as killing established TCP connections, banning IP addresses at the routing table, blacklisting and whitelisting IP addresses and locking individual users (host-based).
The "Kill Connection" (host-based) feature is implemented using Berkeley Packet Filter (BPF) that provides a raw interface to data link layers, permitting raw link-layer packets to be sent and received and, therefore, the ability to stop undesired packets at TCP stack.

12. Client Agent

A lightweight Client Agent has been deployed in ACSIA since v4 and uses ElasticBeats to collect the following data:

- System Logs
- Web Application Logs
- Audit Logs

- Network Traffic
- Kernel/Registry Logs
- Compliance Related Logs

The aforementioned ElasticBeats is bundled into a single Client Agent for Windows, Linux and MAC OS operating systems, and can be downloaded from the ACSIA UI (see the User and Administration Guide) and will auto-install once the downloaded executable is run. The ACSIA agent consolidates all communication ports into a single port which is 443 (HTTPS) and 444 (TCP/UDP).

13. Log Shipping

Beats provided by Open Search are used as the log shipper to transfer the various log files to the Security Information Event Manager in ACSIA. The data volumes involved are minuscule (measured in kilobits), so there is no performance overhead on the client or network when deploying ACSIA.
Client-server certificates encrypt all client traffic using Transport Layer Security (TLS V1.2 or later).

Architectural Design

14. High-Level Architectural Block View

The ACSIA XDR Plus product incorporates Endpoint Detection and Response (EDR) capabilities with Intrusion Prevention (IPS) and Intrusion Detection Systems (IDS) into a single platform, all of which are supported by our real-time Security Information and Event Management (SIEM) system.
We have then enhanced this Extended Detection and Response (XDR) by including a predictive Threat Intelligence feed that bans wrong IP Addresses, anonymous exit nodes, and sources of malware from accessing the network. Eliminating these sources of threats from being able even to view your network infrastructure. This is why we refer to the product as ACSIA XDR Plus.

No Endpoint data of any type is exported or shared from ACSIA to third parties (including Dectar). The ACSIA application is installed on the customer's infrastructure without sharing, distributing or copying data outside the customer's premises.

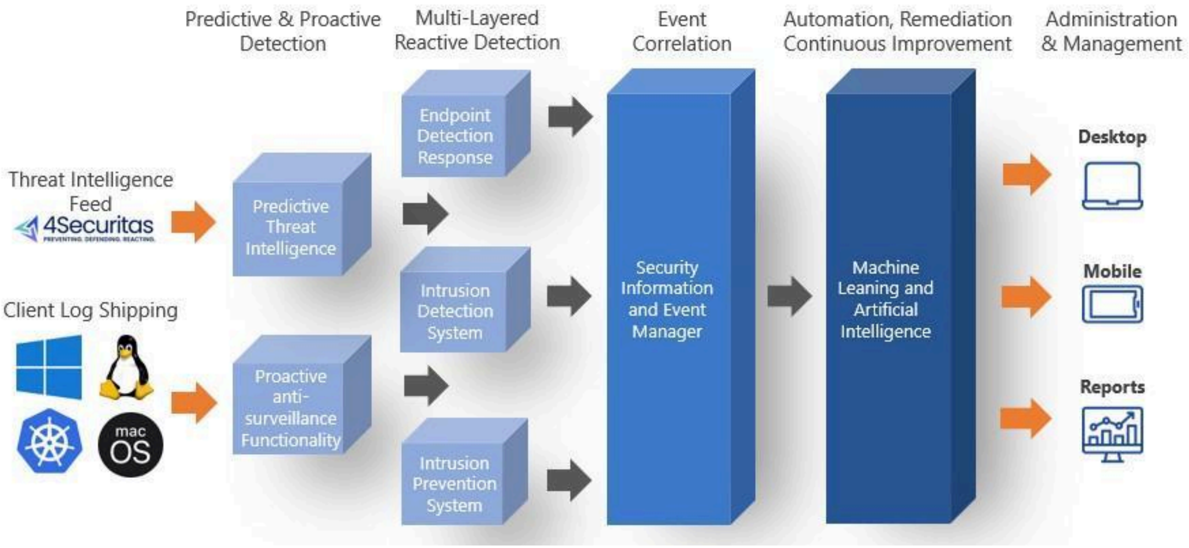
Predictive & Proactive Detection
Multi-Layered Reactive Detection
Event Correlation
Automation, Remediation Continuous Improvement
Administration & Management
Threat Intelligence Feed
4Securitas
Client Log Shipping
mac OS
Predictive Threat Intelligence
Proactive anti-surveillance Functionality
Endpoint Detection Response
Intrusion Detection System
Intrusion Prevention System
Security Information and Event Manager
Machine Leaning and Artificial Intelligence
Desktop
Mobile
Reports

Fig 1: Architectural Block View

Integrating the Predictive, Proactive, EDR, IDS and IPS functions into a single Application Engine facilitates the capture and correlation of multiple event types for superior analytical detection, automated remediation, and continuous improvement through the application of Artificial Intelligence and Machine Learning functions.

15. ACSIA XDR Plus Internal Workflow Diagram

The workflow diagram (fig 2) depicts ACSIA's architectural workflow and integration points with open-source components described earlier in this document.
It also shows the ACSIA server workflow where data from clients are received by "logstash" and stored in an unstructured form in the "OpenSearch" database. The data stream is queued and managed by the message broker "RabbitMQ", which serves the analyzer containing the ACSIA core engine.
The analyzed data is enriched and visualized in the ACSIA Web UI (such as alerts, messaging, notifications etc.) and stored in a MySQL database.
OpenDashboard formats all the OpenSearch data and presents it to the end user.

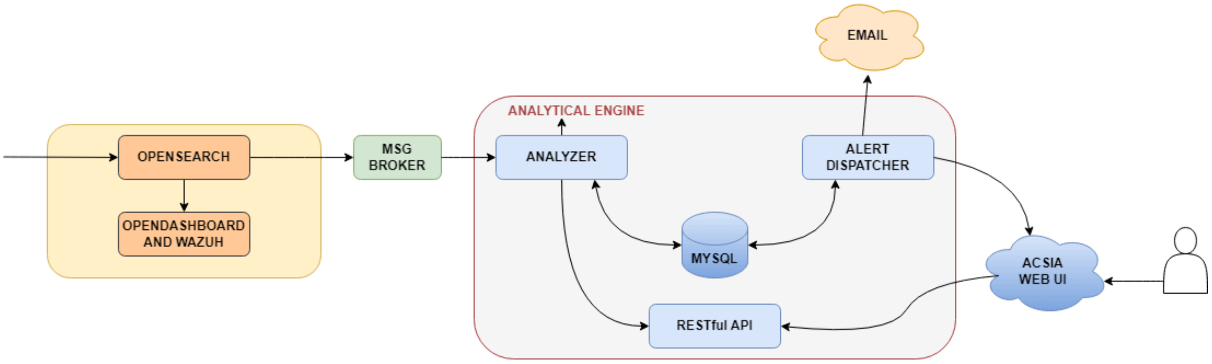

Fig 2: Internal Workflow Diagram

16. ACSIA XDR Plus Client Agent Data Flow Diagram

The following flow chart (fig. 4) shows how the agent deployment interacts with ACSIA by shipping their logs in real-time through a consolidated secure port such as 443. All open-source components used by ACSIA are running within docker containers except the ACSIA analytical engine.

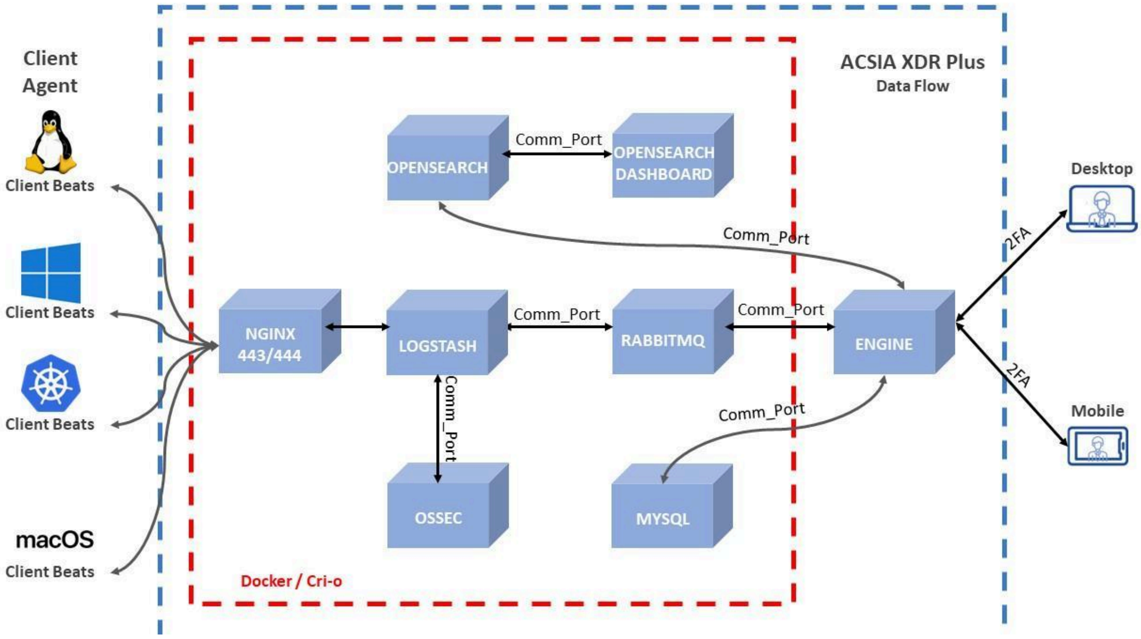

Fig 4: Client Agent Data flow Diagram

Related articles

- ACSIA XDR Plus Software Product Description v6.x.x
- Pre-ACSIA Installation Minimum Requirements Check with AMR
- Release Note v6.x.x
- ACSIA XDR Plus Installation and User Administration Guide - v6.x.x
- Troubleshooting and Common Issues