4SECURITAS
SECURITY SIMPLIFIED

ACSIA - Architecture

This document describes the design and architecture i.e. flow-chart of the ACSIA security tool.

ACSIA Requirements

The environment in which ACSIA can be hosted is variable and depends on the number of connected resources. ACSIA can be installed on both bare metal or virtual environments such as cloud platforms. To install ACSIA you will need a CentOS 7 or equivalent i.e. RHEL 7 host while on the client side ACSIA supports any operating system.

A typical recommended environment to monitor 1-50 hosts would be:

- 2/4 vCPU
- 8/16 GB RAM
- Storage depending on the amount of data and the traffic

The product is scalable accordingly, so for example if the connected hosts are >50, i.e. 50-200 hosts, the recommended scale would be 8/12 vCPU, 32/64GB RAM and storage capacity enough to host the data depending on the amount and the retention period (see log/data retention in administration guide).

ACSIA Engine

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

ACSIA Frontend

The frontend which is REST interface to make call to backend is built in combination of Vue.js and Material Design.

ACSIA Algorithms

ACSIA algorithms are built on multi-layer following a specific escalation route.

- The primary layer contains **algorithms** that implements the **basics and the very fundamentals of security**. In this layer from basics to advanced level almost all of hacking (intrusion/penetration) techniques are covered to be detected and recognized (pattern, signature, automated tools, behavior etc.)
- The secondary layer contains algorithms implemented on **machine learning** concept. In this layers the product gets trained by what decision has been taken from each alerts

and learning new type of attacks by implementing event correlation in addition to the environment monitored (employees, external users etc)..

- The third layer contains algorithms that implements the principle of **artificial intelligence**. In this layer events are prioritized at severity level and an AI based decision making processes triggers each event to be handled or dispatched via notification.

Below is just a list of some threats that ACSIA is able to detect with our multi-layer algorithms:

- System level attacks threat detection
- Application level threat detection
- Malicious tools detection and identification
- BotNet detection
- SQL Injection detection
- Code Injection detection
- Brute-force detection
- XSS attack detection (Type-0, Reflected and Stored)
- CSRF detection
- LFI - File inclusion detection
- Malware detection (trojan, webshell, rootkit, etc.)
- Eavesdropping and information gathering detection
- Portscan detection (invasive only)
- Portscan detection
- Potential account compromises (geo-location based)
- Potential account compromises (UEBA)
- Linux container support (threat detection)
- User activity and session player within the notification
- Internal user and entity activity analysis (UEBA)
- Exploitation and 0-day detection
- Privilege escalation detection
- File integrity check
- Kernel level threat detection (Linux systems and containers)
- Automated cyber attack event handling and remediation
- Incident Response
- Advanced persistent threats detection
- Indicator of Compromises alerting

ACSIA Security

ACSIA comes with its integrated own security components:

- OAuth2 - Secure delegated access
- Multi factor authentication - 2 factor authentication via Google Authenticator

- TLS for secure communication across clients
- Pwgen - Strong random password generator

ACSIA Open Source Toolstack
There are several open source tools where the ACSIA main analytical engine combines them and applies all of its in-house implemented algorithms (totalling ~50) to get the best out of their functionalities. These are separately described in various documentation provided (some publicly disclosed others exclusive to customers and some others not disclosed at all).

Below is the list of the tools:

- Curl
- Whois
- Binds-utils
- Python pip
- Dsniff
- Postfix
- Bc
- Wget
- Sysstat
- Httpd-tools

ACSIA Virtualization Method
ACSIA uses LXC (Linux Containers) which is an operating-system-level virtualization method for running multiple isolated Linux systems (containers) on a control host using a single Linux kernel.

ACSIA Configuration Management System

ACSIA avails of Ansible which is the most popular software that automates software provisioning, configuration management, and application deployment.

ACSIA Databases

ACSIA uses more than one databases segregated for different purposes:

- MariaDB
- MongoDB

ACSIA Message Broker

ACSIA architecture includes RabbitMQ which is the most widely deployed open source message broker.

SIEM Environment

ACSIA is availing of a well known log collector collector such as Elastic stack:

- Elastic Search
- Lucene
- Logstash
- Kibana
- ElasticBeats (log shippers)

ACSIA Firewall

ACSIA comes with an embedded host-based firewall. The firewall covers the features such as killing established TCP connections, banning IP addresses at routing table, therefore blacklisting and whitelisting IP addresses as well as 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. This feature is available on most Unix-like operating systems.

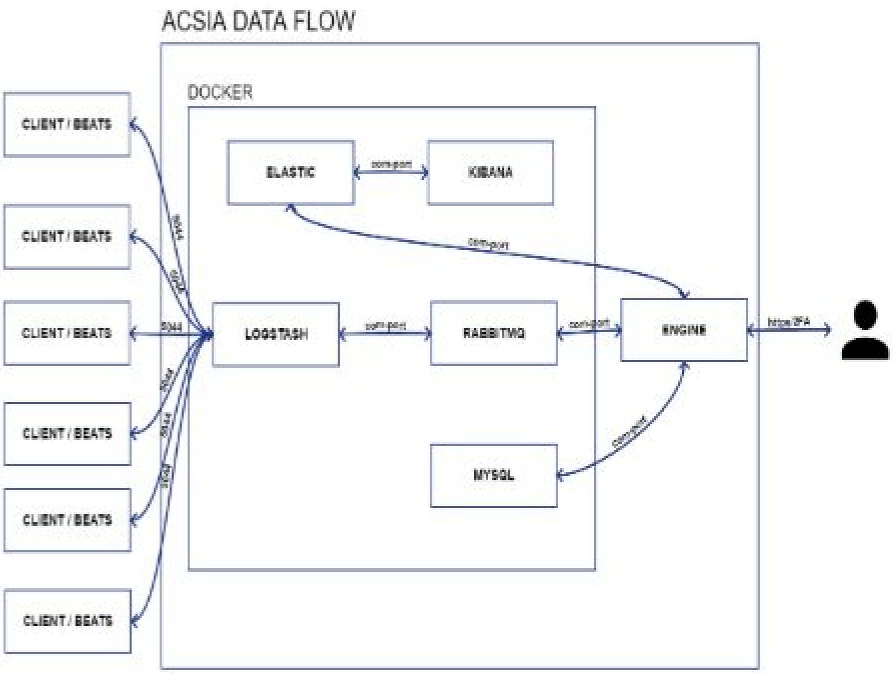
ACSIA Clients

ACSIA works at server level and at the same time it captures and analyzes network traffic as well. Therefore, it works on both network and end-point perimeters. For details about the clients implementation and connection please refer to user administration guide.

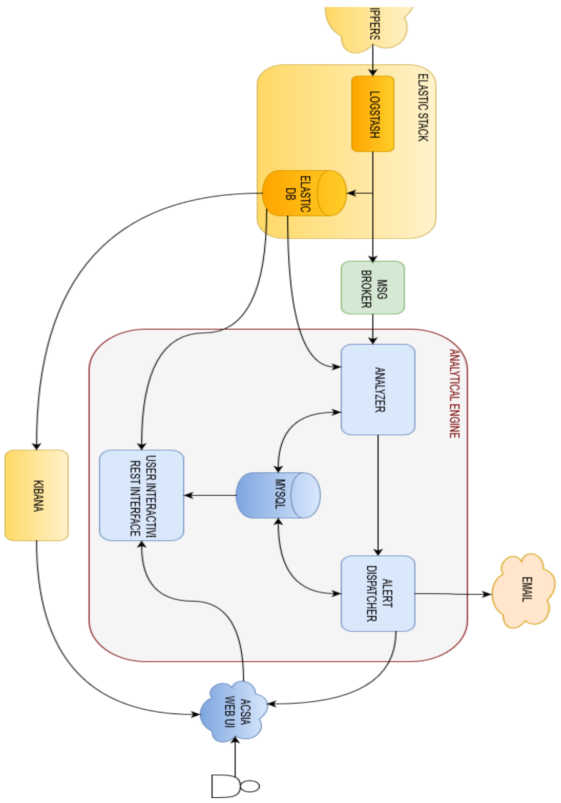
4SECURITAS

SECURITY SIMPLIFIED

ACSIA Architecture - Design - Flowchart



ACSIA High Level Architecture - Diagram


IPPERS
ELASTIC STACK
LOGSTASH
ELASTIC DB
MSG BROKER
ANALYTICAL ENGINE
ANALYZER
KIBANA
USER INTERACTION REST INTERFACE
MYSQL
ALERT DISPATCHER
EMAIL
ACSIA WEB UI