Automated Cybersecurity Interactive Application Design and Architecture Overview

This document describes the design and architecture ACSIA security solution.
ACSIA Basic Requirements

ACSIA runs on CentOS Linux 7 system or Red Hat 7 (RHEL 7) regardless of platform deployment type (physical, virtual, container or cloud. When it comes to cloud platforms such as Azure and Google Cloud the process is almost identical as with the on-premise deployment. With AWS the process is a bit diferent as ACSIA is on the 'AWS Marketplace', so a few lines of instructions on using the marketplace will sufce to activate ACSIA.

For non-cloud installations, the minimum resource allocation for ACSIA is 2-4vCPU, 8-16GB RAM with the amount of storage required depending on the volume of logs and the retention period required; a suggested 200GB allocation would be sufcient for most environments wishing to retain logs for a one month period.
The above resource requirements would typically support up to 200 servers connected to a single ACSIA instance. For more than 200 clients an additional ACSIA subscription is required with a similar resource allocation.
Automated Cybersecurity Interactive Application

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 is a REST API interface that makes calls to the backend using a combination of Vue.js and the 'Material Design' design language.
ACSIA Security

ACSIA comes with its own integrated 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 used by the ACSIA analytical engine. Below is the list of the tools:

- Curl
- Whois
- Binds-utils
- Bc
- Python pip
- Dsnif
- Postfix
- 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 uses Ansible for automating software provisioning, configuration management, and application deployment.

ACSIA Databases

ACSIA uses two separate databases which are segregated for security design reasons

- MariaDB/MySQL
- MongoDB (Elastic)

ACSIA Message Broker

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

SIEM Environment

ACSIA uses predominantly Elastic Stack tools for our SIEM centralized log collector
- Elastic Search
- Lucene
- Logstash
- Kibana
- ElasticBeats (log shippers)
- OSSEC agents (log shipper)

ACSIA Firewall

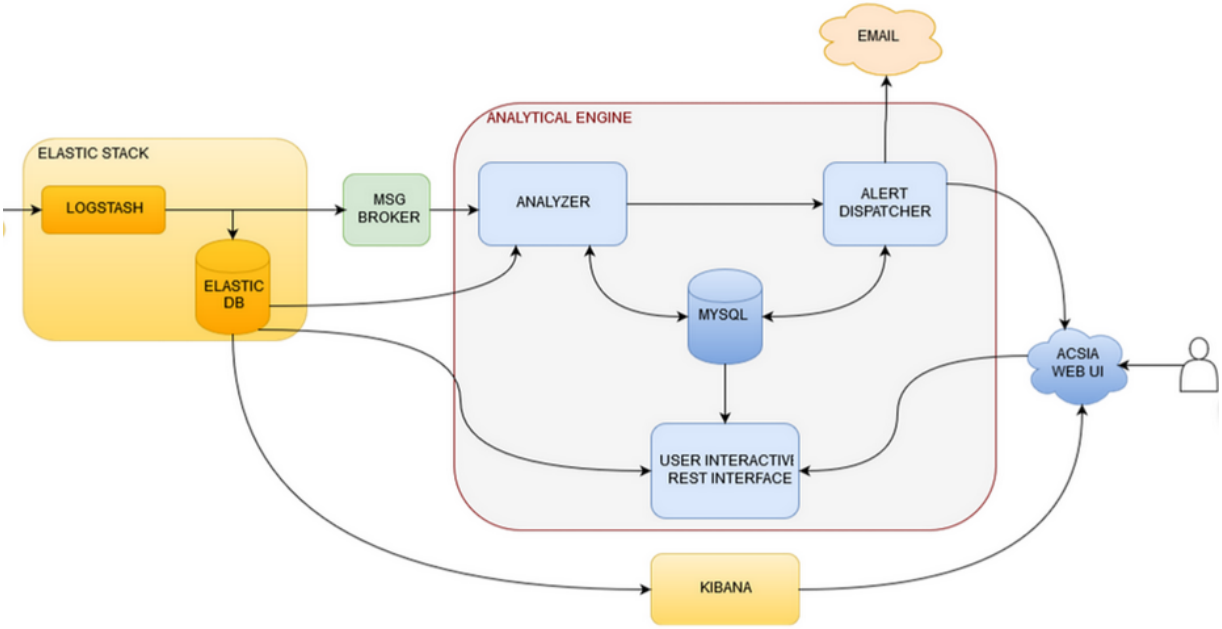
ACSIA comes with an embedded host-based firewall. The firewall covers the features such as killing established TCP connections, banning IP addresses at routing table, therefore blacklisting and whitelisting IP addresses as well as locking individual users (host based).
The “Kill Connection” (host-based) 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.

ACSIA Clients

ACSIA works at both network and end-point perimeters by operating at server level whilst simultaneously capturing and analysing network trafc. ACSIA supports both Windows and Linux systems as clients. For details about the clients implementation and connection please refer to the user administration guide.

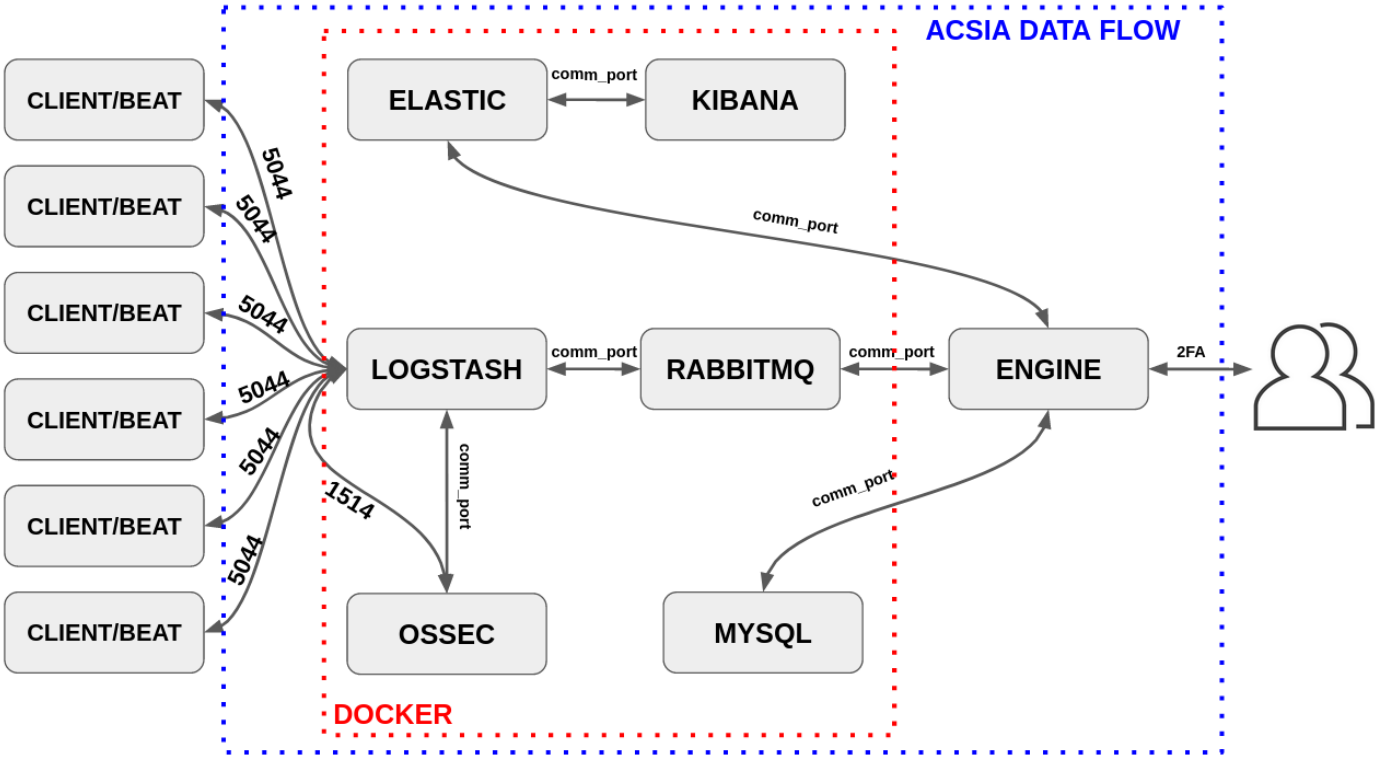
ACSIA High Level Architecture - Diagram

The following depicts ACSIA's architectural design and the integration with external open source components described above in this document. The red line square indicates ACSIA analytical engine.



ACSIA Architecture - Design - Flowchart

The following flow chart shows how clients interact with ACSIA by shipping their logs in real time. All open source components used by ACSIA are running within docker containers with the exception of the ACSIA analytical engine.



For full details about ACSIA administration and Usage please refer to ACSIA Administration Guide or contact us at support@acsia.io.

4Securitas

65 Ivy Exchange, Parnell Street D01 AW68 Dublin Registered in Ireland

CRO: 598914 VAT Number: 3463752OH

Contact: info@4securitas.com