

**BI Data Lineage & excellence Tools**

**High Level Design**

**Version: 1.0**

**Confidential Document**

Author: PT Dua Empat Tujuh

Doc Version: 1.0.0

No. : TSELAGE-HLD-231201-0049-0100



**PT Dua Empat Tujuh**

Copyright © 2021 Solusi247. All rights reserved.

# 

# Document Location

Revision History

|  |  |
| --- | --- |
| Date of this revision: December 1, 2023 | Date of next revision |

|  |  |  |  |
| --- | --- | --- | --- |
| Revision Number | Revision Date | Summary of Changes | Changes marked |
| 1.0.0 | December 1, 2023 | Initial Document | (N) |

Approval

This document requires following approvals.

|  |  |
| --- | --- |
| Name | Title |
|  | Telkomsel – |
|  | Telkomsel – |
| Muhammad Anis | PT Dua Empat Tujuh – Project Manager |

Distribution

This document has been distributed to

|  |  |
| --- | --- |
| Name | Title |
|  | Telkomsel – |
| Muhammad Anis | PT Dua Empat Tujuh – Project Manager |

# Document Approval

|  |  | **Signature** | **Name/Position** | **Date Signed** |
| --- | --- | --- | --- | --- |
| **Approved (Telkomsel)** | **:** | .......................... |  | .................. |
| **Approved (Telkomsel)** | **:** | .......................... |  | .................. |
| **Approved (Solusi247)** | **:** | .......................... | Muhammad Anis  Project Manager | .................. |
|  |  |  |  |  |
|  |  |  |  |  |

# Contents

Document Location 2

Document Approval 3

Contents 4

1 Introduction 6

1.1 Purpose 6

1.2 Term & Definition 6

2 Overview 7

2.1 Background 7

2.2 Scope 7

3 Solution Design 9

3.1 Design Principles 9

3.2 Solution Architecture 9

3.3 Solution Components 11

3.3.1 Data Sources 11

3.3.2 Data Ingestion 11

3.3.3 Data Processing 11

3.3.4 Data Retention 12

3.4 Technology Stack 13

4 Data Ingestion Layer 17

4.1 HA for Data Ingestion 17

4.2 Pull framework. 18

4.3 Push framework. 18

5 Data Processing 19

5.1 Transaction Feed 19

5.1.1 Read Records from Kafka 19

5.1.2 Transformation 20

5.1.3 Write to Kafka 20

5.2 Reference Feed 20

5.2.1 Local file staging 21

5.2.2 Nifi read file 21

5.2.3 Mapping 21

5.2.4 Redis 21

6 Cyber Security Use Case Typology 22

6.1 Network 22

6.1.1 DDOS attack 22

6.1.2 Bad connection perimeter monitoring (Outbound) 22

6.1.3 Port and Vulnerability Scan 23

6.2 Application 23

6.2.1 Injection Attacks 24

6.2.2 Backdoor and Webshell 24

6.3 Endpoint 24

6.3.1 File changes (Servers) 25

6.3.2 Vulnerable and Outdated Components (Servers and workstations) 25

6.4 Access 25

6.4.1 Users’ login to servers not using PAM 26

6.4.2 Remote access brute-force 26

6.5 Fraud 26

6.5.1 Spam and Phishing: IP/Domain Phishing 27

6.6 Malware 27

6.6.1 Active Malware 28

7 Audit and Reconciliation 29

8 Application Environments 30

8.1 Development Environment 30

8.1.1 SIT Environment 30

8.1.2 UAT Environment 30

8.1.3 NFT Environment 30

9 Job Scheduling and Monitoring 32

10 Configuration & Deployment Management 33

11 Reference 34

12 Appendix 35

# Introduction

## Purpose

The purpose of this high-level design document is to define the scope of the requirements that are included in this design and to document the various components that will be integrated as part of this “BI Data Lineage & Excellence Tools” solution. The objective of this document is to provide high level details of all the components and form a baseline for the detailed design document that will be created during the program execution.

## Term & Definition

The following definitions are used in this document:

| Term | Definitions |
| --- | --- |
| AIML |  |
| HDFS | Hadoop Distributed File System |
| BI | Business Intelligence |
| SIT | System Integration Test |
| UAT | User Acceptance Test |
| NFT | Non-Functional Test |

# Overview

## Background

The objective of this project is to enhanced visibility of network security threats to mitigate current and emerging risks develop a business-driven, threat-based approach to conducting network security threat assessment based on Telkomsel's specific business, threats and capabilities.

Therefore, it needs to develop monitoring dashboard that can be assist executive, managerial, and operational stakeholders to better understand current network security relates threats and risks.

From these needs, the results to be obtained include:

* Comprehensive risk profile that covers IT network security related risk
* Enhanced monitoring dashboards and reporting (prompt detection of existing risks to allow for better response)
* Enhance threat prevention and detection capabilities within Telkomsel (improve functionality and usability of the existing network security monitoring tool and platform)
* Better configure current network security devices to minimize risk and free up bandwidth for better customer experience

## Scope

This document has the scope to define a high-level design for this project is based on the results of project team discussions. After going through the stage of gathering requirements, then through the process of analysis and discussion, it can be concluded with the scope of work as follows:

| No | Area | Details |
| --- | --- | --- |
|  | Platform Engine | 1. Sigma + ELK Platform Engine 2. TheHive Project Platform Engine |
|  | Additional Features | 1. Notable Event Handling Workspace 2. MTTD & MTTR Counter on Alert Handling 3. Incident Automation & Ticketing System Integration 4. Adopt Proven Framework for Incident Response 5. Forensic & Investigation 6. Free Search Workspace (Query Based) 7. Aggregator Capability 8. Reporting and Dashboard 9. Alerting |
|  | Additional IT Use Cases | 22 IT Use Cases |
|  | Phase 1 Alert | 21 Alert from Phase 1 Use Cases |
|  | CGNAT Adjustment | 1. Millisecond will be added to ‘timestamp’ column 2. Adding new 7 columns to CGNAT |

Table 2 Project Scope

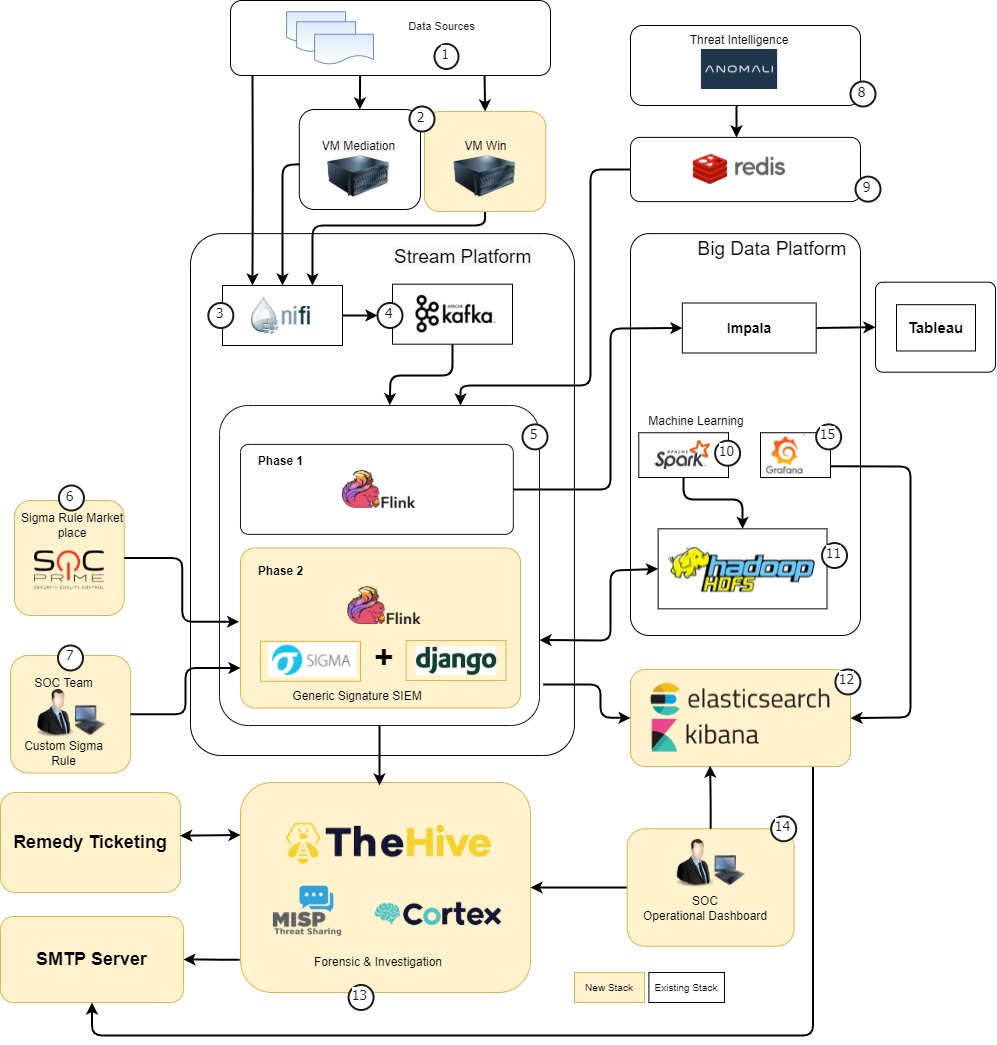
# Solution Design

## Design Principles

Basic design considerations under which solution has been designed have been stated here:

* Implementation of easily accessible, highly scalable but cost-effective data pipeline
* Linearly scalable solution to address high volume and fast data growth
* Framework capable of processing large volume of data

## Solution Architecture

**

1. Business Architecture Diagram
2. Telkomsel will provide the streaming data using UDP/TCP. The incoming data is already formatted in structured data and should have the same format as agreed during the development and based on raw device. Any changes on the data stream format/data structure will require reperform the profiling, update transformation rules, and reprocess the transformation.
3. Stream data sources must be passed through the syslog server if needed to be ingested to another system such as splunk
4. Then data will be streamed to the Nifi environment for splitting, mapping, enrichment and forwarded to the port that has been provided for each data sources. Nifi streaming platform will listen incoming data source on certain port and publish it to message broker kafka
5. Data will ingest from Nifi to Kafka for base layer
6. Incoming data on kafka will be consume by Flink, flink will create signature of events using predefined rule using SIGMA
7. Threat detection and mitigation rules can be obtained more quickly by utilizing the rules available in the market.
8. If the rules available on the market are not in accordance with Telkomsel's conditions, the SOC team can create their own rules for threat detection and mitigation
9. Nifi will get Threat Intelligence (CTI) from Anomaly server
10. The CTI data stored to Redis as cache for datasource correlation on Flink data processing
11. Machine learning is a component that was built in Phase 1
12. Correlated events will be store on HDFS as a backup
13. Elasticsearch for daily monitoring and tracking. The signed event that has match with CTI will be categorized as "Alert" and will be store also on the TheHive
14. As an incident and response management. The analyst can investigate the threat using available analyzer on the cortex application.
15. SOC team will use Kibana to see the dashboard and TheHiveProject as Incident response and analyzer
16. Grafana will be used to visualize ANR data from Elasticsearch for operational purposes

The following are the platforms from phase 1 (existing) that will still be used in phase 2, as well as what new platforms will be used in phase 2

**Existing Platform:**

* Nifi
* Kafka
* Redis
* Spark
* HDFS
* Flink

**New Platform:**

* Sigma
* Django
* TheHive
* MISP
* Cortex
* Elastic Search
* Kibana

## Solution Components

### Data Sources

Telkomsel will provide the streaming data using UDP/TCP. The incoming data is already formatted in structured data and should have the same format as agreed during the development and based on raw device. Any changes on the data stream format/data structure will require reperform the profiling, update transformation rules, and reprocess the transformation.

### Data Ingestion

Stream data sources must be passed through the syslog server if needed to be ingested to another system such as Splunk. Then data will be streamed to the Nifi environment for splitting, mapping, enrichment and forwarded to the port that has been provided for each data sources. Nifi streaming platform will listen incoming data source on certain port and publish it to message broker Kafka. Data will ingest from Nifi to Kafka for base layer

### Data Processing

Incoming data on Kafka will be consume by Flink, Flink will create signature of events using predefined rule using SIGMA. Threat detection and mitigation rules can be obtained more quickly by utilizing the rules available in the market. If the rules available on the market are not in accordance with Telkomsel's conditions, the SOC team can create their own rules for threat detection and mitigation. Signed event will be correlated with Threat Intelligence (CTI) from Anomali. The CTI data has been stored to Redis as cache for Flink data processing. Machine learning is a component that was built in Phase 1. Correlated events will be store on HDFS as a backup. Elasticsearch for daily monitoring and tracking. The signed event that has match with CTI will be categorized as "Alert" and will be store also on the TheHive. As an incident and response management. The analyst can investigate the threat using available analyzer on the cortex application.

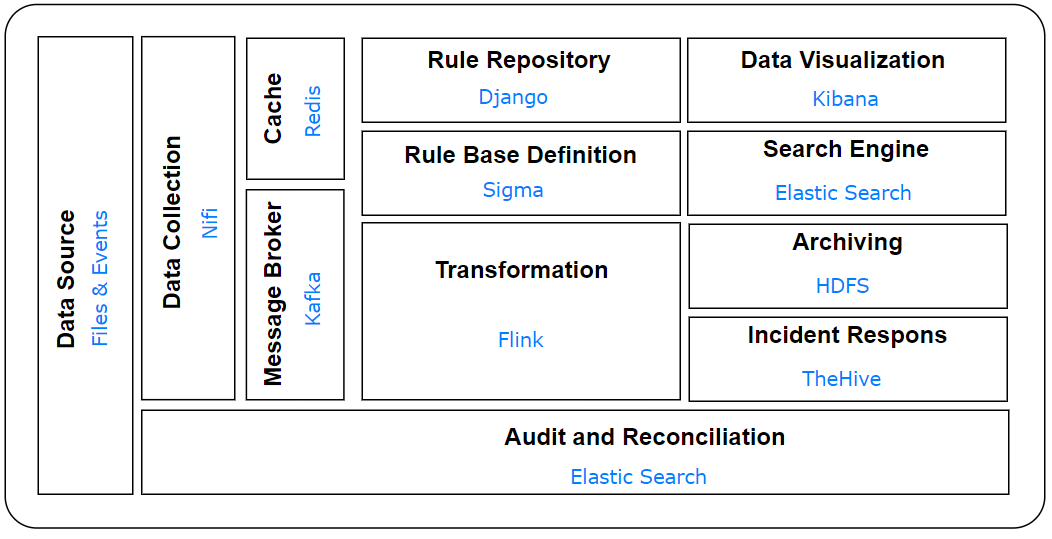
### Data Retention

|  |  |
| --- | --- |
| **Data** | **Retention Time** |
| Elasticsearch | 3 Months |
| HDFS | 1 Year |

Table 3 Data Retention

## Technology Stack

Based on functional components as articulated in section 3.2, below is proposed technology landscape for each component:



1. Technology Stack Diagram

Each of the mentioned elements is detailed below:

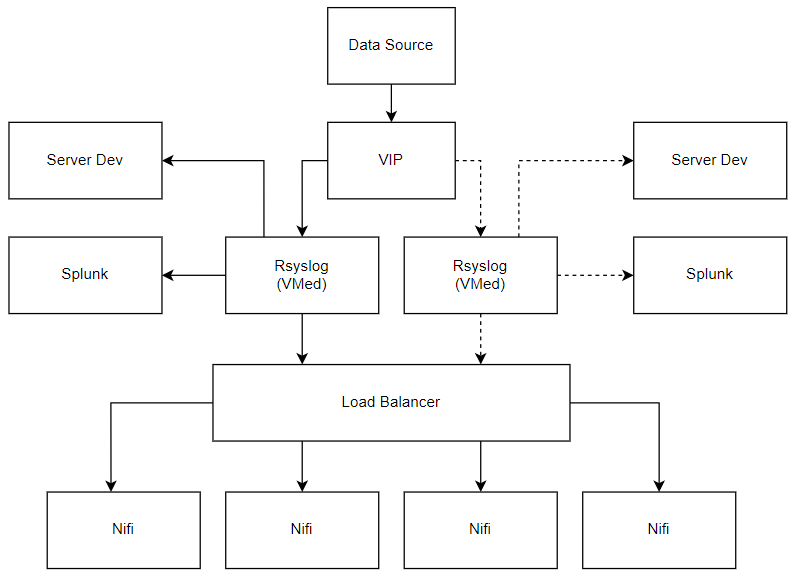
| **No** | **Category** | **Solution** | **License** |
| --- | --- | --- | --- |
|  | Nifi | * NiFi is a system that designed to automate the flow of data between software systems. Data that comes from syslog will be ingested to NiFi using UDP (ListenUDP). NiFi is capable to perform transformation such as extract data, filtering and data manipulation. In processing data, NiFi will have separate pipeline for each port and data sources. NiFi will merge the data from pipeline and load the data into Kafka. | * Telkomsel’s Cloudera Subscription |
|  | Kafka | * Kafka is a distributed event streaming platform. Kafka publishes (write) and subscribes to (read) streams of events. Kafka can store streams of events durably and reliably for as long as wanted. | * Telkomsel’s Cloudera Subscription |
|  | Flink | * Flink is a data processing framework with stream-processing and batch-processing capabilities. In this project, Flink will be used to process data as unbounded stream and provide data as it is generated. The stateful Flink applications will also be utilized for generating alert where the state is stored locally in memory. | * Telkomsel’s Cloudera Subscription |
|  | Sigma | * Sigma is a generic and open signature format that allows you to describe relevant log events in a straightforward manner. The rule format is very flexible, easy to write and applicable to any type of log file. The main purpose of this project is to provide a structured form in which researchers or analysts can describe their once developed detection methods and make them shareable with others. | * Community |
|  | Redis | * Redis is in-memory data structured store. Redis is a key-value where key can contain data structure such as string, hash, list, set, and sorted set. Data that ingested to Redis is stored as reference such as CTI anomali. | * Community |
|  | Elastic Search | Elasticsearch is a NoSQL database with a focus on search engine databases. Elasticsearch is powered by Apache Lucene which is also a database search engine that has low level queries. Elasticsearch has easy-to-use queries because it's RESTful based. | Community |
|  | Kibana | * Kibana is a data visualization and management tool for Elasticsearch. The visualized data can be in the form of graphs, metrics, or tables. Kibana also provides a dashboard feature, this feature serves to collect data that has been visualized into one dashboard page. * To successfully log in to Kibana, basic authentication requires a username and password. Basic authentication is enabled by default, and is based on LDAP or Active Directory security realm that is provided by Elasticsearch. LDAP, Active directory authentication available in Platinum or Enterprise license. | * Community |
|  | TheHive Project | TheHive is free Security Incident Response Platform which is integrated with Cortex and MISP. TheHive is designed to make life easier for SOCs, CSIRTs, CERTs and any information security practitioner dealing with security incidents that need to be investigated and acted upon swiftly. In simple terms, TheHive acts as a front-end application to the SOC to aid in the three fundamental phases (Detection, Analysis and Response) as well as the case/alert management from creation to closure.  TheHive authentication consists of a set of modules. Each one tries to authenticate the user. If it fails, the next one in the list is tried until the end of the list. There are two modules that apply for this project:   * 1. Basic   Authenticates HTTP requests using the login and password provided in authorization header using basic authentication format (Base64). Password is checked from the local user database.   * 1. LDAP   Use LDAP directory server to authenticate the user.  **Cortex**  Cortex is another software that was created by the same team as TheHive and works closely with TheHive. TheHive and Cortex could be used together to make life in a SOC much easier. In such a solution, any Indicator of Compromise (IoC) or pieces of forensic data that could help in mitigating threat actors are classified as an observable. Cortex is a powerful Observable Analysis & active Response Engine. It has analyzers which help in the analysis of these observables. The Analyzers also help in enriching the alerts on TheHive with valuable information. Using this valuable information analysts could then run Responders available on Cortex for easy and automated resolution of the security alert, which these responders will be used as a tool to integrate tickets into the Remedy System.  The Hive analyzer can use third party service to analyze the IoC. The following is a list of some of the indicator types.   * IP Address * Domain * URL * File * Email * Host * Threat Actor * Intrusion Set * Malware * Campaign * Registry Key * CVE CVSS Score   **MISP**  MISP or Malware Information Sharing Platform is an open-source software solution for collecting, storing, distributing and sharing cybersecurity indicators and threats about cybersecurity incidents and malware analysis. This has been developed and maintained by CIRCL. Apart from the many uses of MISP one of the main uses which this solution could leverage is the subscription to many of the open-source threat intel feeds. The true power of TheHive can be seen when it is integrated with Cortex and MISP. All these 3 tools work hand in hand to form a comprehensive solution for Incident Analysis/Response and Case Management in a Security Operation Centre. | Community |
|  | Django | Django framework used to build Sigma UI which is used as an interface for managing repository of sigma rules and data pipeline.  Django is a full-stack framework for creating web applications in the Python programming language. Django is also a high-level python web framework that can do application development quickly and has a clean pragmatic design.  There are two authentication modules that apply for this project:   1. Basic   Password is checked from the local user database.   1. LDAP   Use LDAP directory server to authenticate the user. | Community |
|  | HDFS | HDFS is a distributed storage system, which performs the process of splitting large files into smaller parts and then distributing them to clusters of computers. Used as archiving of parsed output and transformation of flink. | Telkomsel’s Cloudera Subscription |

Table 4 Technology Definition

# Data Ingestion Layer

Data Ingestion layer will build using the Cloudera Data Flow product. Cloudera Dataflow (CDF) is a scalable, real-time streaming data platform that ingests, curates, and analyzes data for key insights and immediate actionable intelligence.

## HA for Data Ingestion



1. Business Architecture Diagram

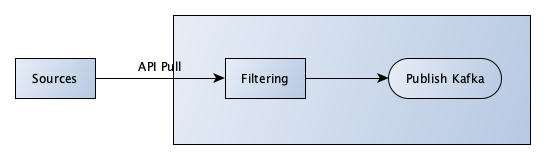
Standby

Active

Logs coming from data source will be sent to Virtual IP of VM Mediation. VM Mediation then will forward data sources to Splunk, Cyber Big Data Development Server, and Load Balancer.

## Pull framework.

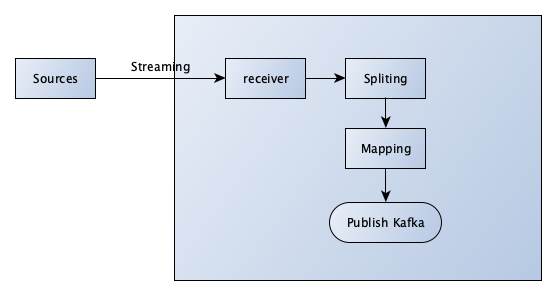
Few of the sources in Telkomsel are able to integrated based on file pull design. Data should be got via specific API. Below mentioned is the high-level flow of the file pull process.



1. Pull Framework
2. NIFI processor would get the data via API based on the configuration.
3. The data that receive will be filtered by some condition that needs to apply
4. Valid records would be published in Kafka

## Push framework.

Most of the data sources in Telkomsel are integrated based on push design. These are transferred via UDP protocol stream. Below mentioned is the high-level flow of the push process.



1. Push Framework
2. NIFI processor would get the data via receiver based on the configuration.
3. Data will be split based on data type on each data sources
4. Data will be mapped based on reference data center, and internal destination port
5. Data will be published to Kafka based on data source and data center

# Data Processing

The feeds collected from external source system, must pass through from different data processing components/functionalities. The treatment on the feed is applied based on the category to which the feed belongs.

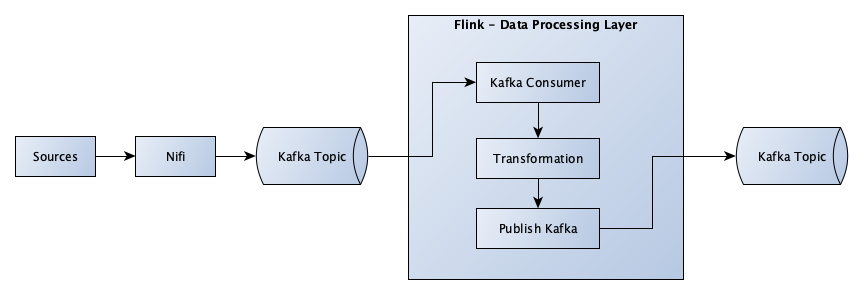
The Data Processing layer is going to be implemented on Apache Flink, it’s an open-source platform, unified stream-processing and batch-processing framework with core is a distributed streaming data-flow engine written in Java and Scala. The platform provides a scalable, distributed, fault-tolerant, and stateful stream processing capabilities. Its runtime supports low-latency processing at extremely high throughputs in a fault-tolerant manner.

We can divide the feeds into 2 main categories:

1. Transaction Feed
2. Reference Feed

## Transaction Feed

The standard data feeds are considered as text file format arriving daily or any other defined/specific period. The diagram below shows a standard data feed. This approach will be used for all the NRT (near real time) feeds and some of the daily batch feeds that are going to be available at defined/specific period.



1. Transaction Feed

Standard data feeds will require the following reusable functions/utilities/components that can support the Data processing functionalities.

### Read Records from Kafka

The file collection layer Nifi, will insert the records to Kafka topic. The Flink must consume the input records from Kafka topic.

### Transformation

This section documents transformations done to the data in the data processing/Flink layer.

* ***Lookup and Enrichment***

The function / utility will perform logic to transform the required attribute/indicator column based on the business requirement. The column will be correlate to CTI Anomali data source to get Threat Information, if the indicator column is match with CTI is\_alert column will be True, if not match will be False.

* ***Timestamp Normalization***

The function/utility will perform timestamp format conversion if the column is specified as Timestamp data type.

* ***Data source Correlation***

The function/utility will perform logic to transform the required attribute/indicator column based on the business requirement as specified on Source to Target Mapping document.

* ***Record Split***

In some of the feeds, the incoming record’s attribute will have Json format or a string with specific delimiter. The function/utility must perform logic to split the specific attribute/column to multiple columns based on the business requirement specified in source to target mapping document.

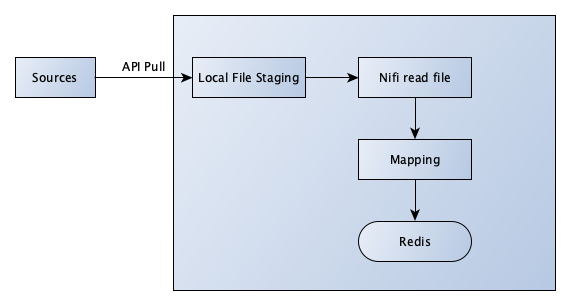
### Write to Kafka

The records must be inserted/publish to Kafka topic using Avro compression method, once all the deduplication, transformation and lookup is applied to the records.

There might be possibility of publishing the same record to different Kafka topic based on the requirement.

## Reference Feed

The diagram below shows a reference feed collection and ingestion. This approach will be used for all the reference feeds that are collected daily.



1. Reference Feed

### Local file staging

The reference feeds are pulled via API and writes the content of a flow file.

### Nifi read file

File that already generated in previous process then will be read by Nifi processor and loaded based on configuration.

### Mapping

The data received then will be mapped into respective required columns as specified.

### Redis

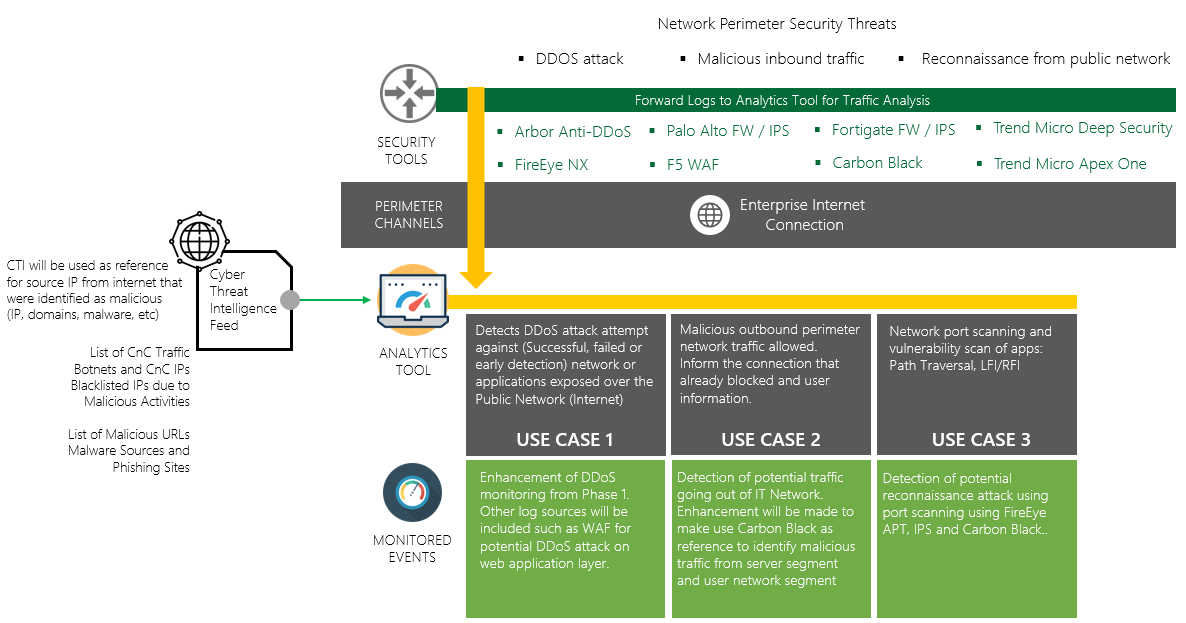
Redis is an in-memory data structure store, used as a database, cache, and message broker. The output of reference feed is inserted to Redis with key columns. Data processing platform will read from this Redis table as an input.

# Cyber Security Use Case Typology

Below are the 22 Cyber Security Use Cases that are categorized by Network, Application, Endpoint, Access, Data Loss, Fraud, Malware, and Security Intelligence.

## Network

Network has 3 use cases with description as follows:



1. Description of IT Network Use Cases

### DDOS attack

Detects DDoS attack attempt against (Successful, failed or early detection) network or applications exposed over the Public Network (Internet). Tools that can detect the malicious activity/traffic are:

* Arbor Anti-DDoS\*\*
* FireEye NX\*\*
* Perimeter FW / IPS\*\*
* WAF\*\*
* Anomali CTI (for potential DDoS detection)

### Bad connection perimeter monitoring (Outbound)

Malicious outbound perimeter network traffic allowed. Inform the connection that already blocked and user information. Tools that can detect the malicious activity/traffic are:

* FireEye NX\*\*
* Perimeter FW / IPS\*\*
* DMZ FW / IPS\*\*
* Carbon Black\*\*
* Anomali CTI
* Trend Micro Deep Security
* Trend Micro Apex One

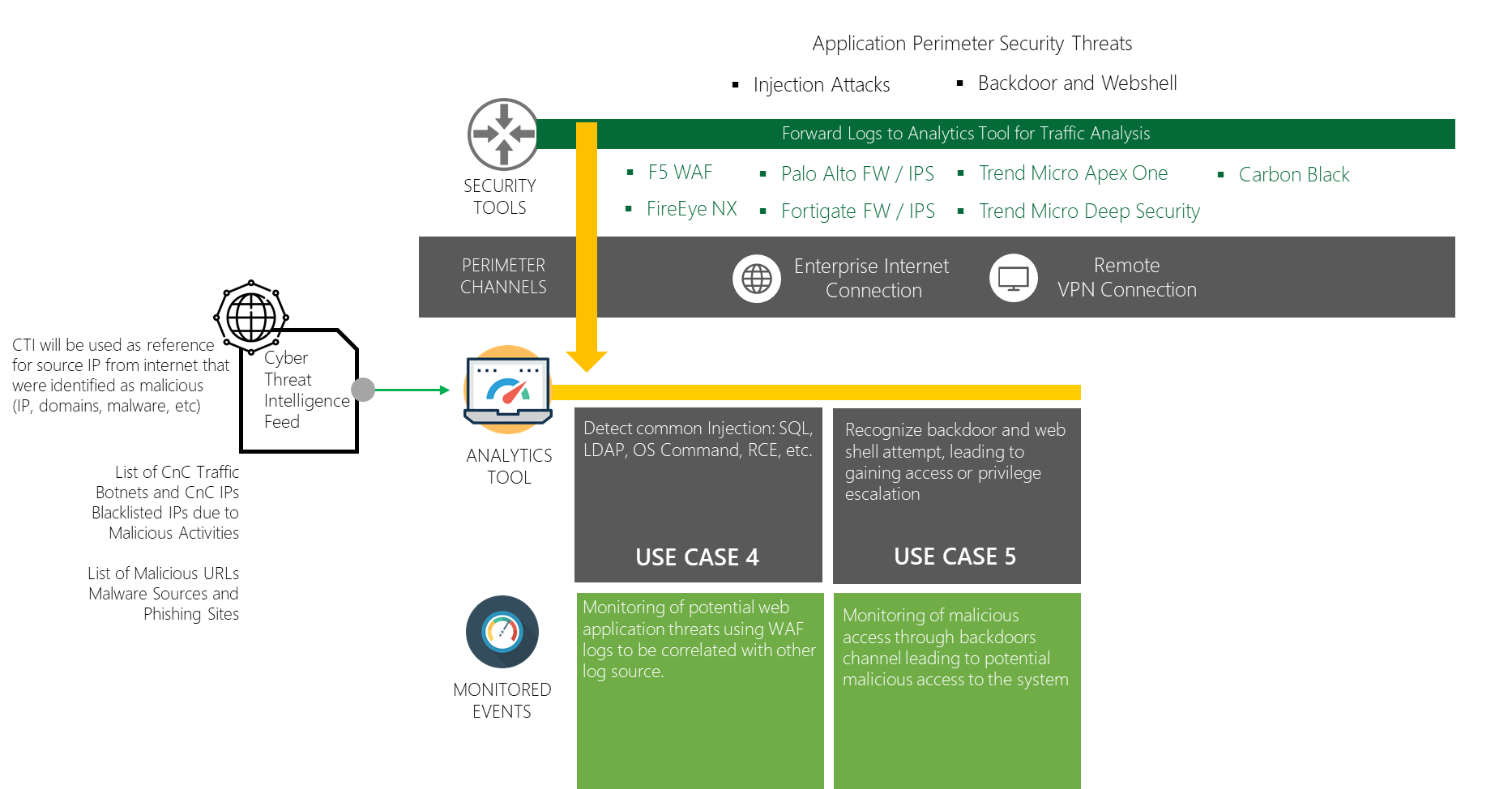
### Port and Vulnerability Scan

Network port scanning and vulnerability scan of apps: Path Traversal, LFI/RFI. Tools that can detect the malicious activity/traffic are:

* FireEye APT\*\*
* Perimeter FW / IPS\*\*
* WAF\*\*
* Internal FW / IPS\*\*
* Carbon Black EDR\*\*
* Trend Micro Deep Security
* Trend Micro Apex One

## Application

Application has 2 use cases with description as follows:



1. Description of IT Application Use Cases

### Injection Attacks

Detect common Injection: SQL, LDAP, OS Command, RCE, etc. Tools that can detect the malicious activity/traffic are:

* WAF\*\*
* FireEye NX
* Perimeter FW / IPS
* Internal FW / IPS
* CTI
* Trend Micro Deep Security
* Trend Micro Apex One

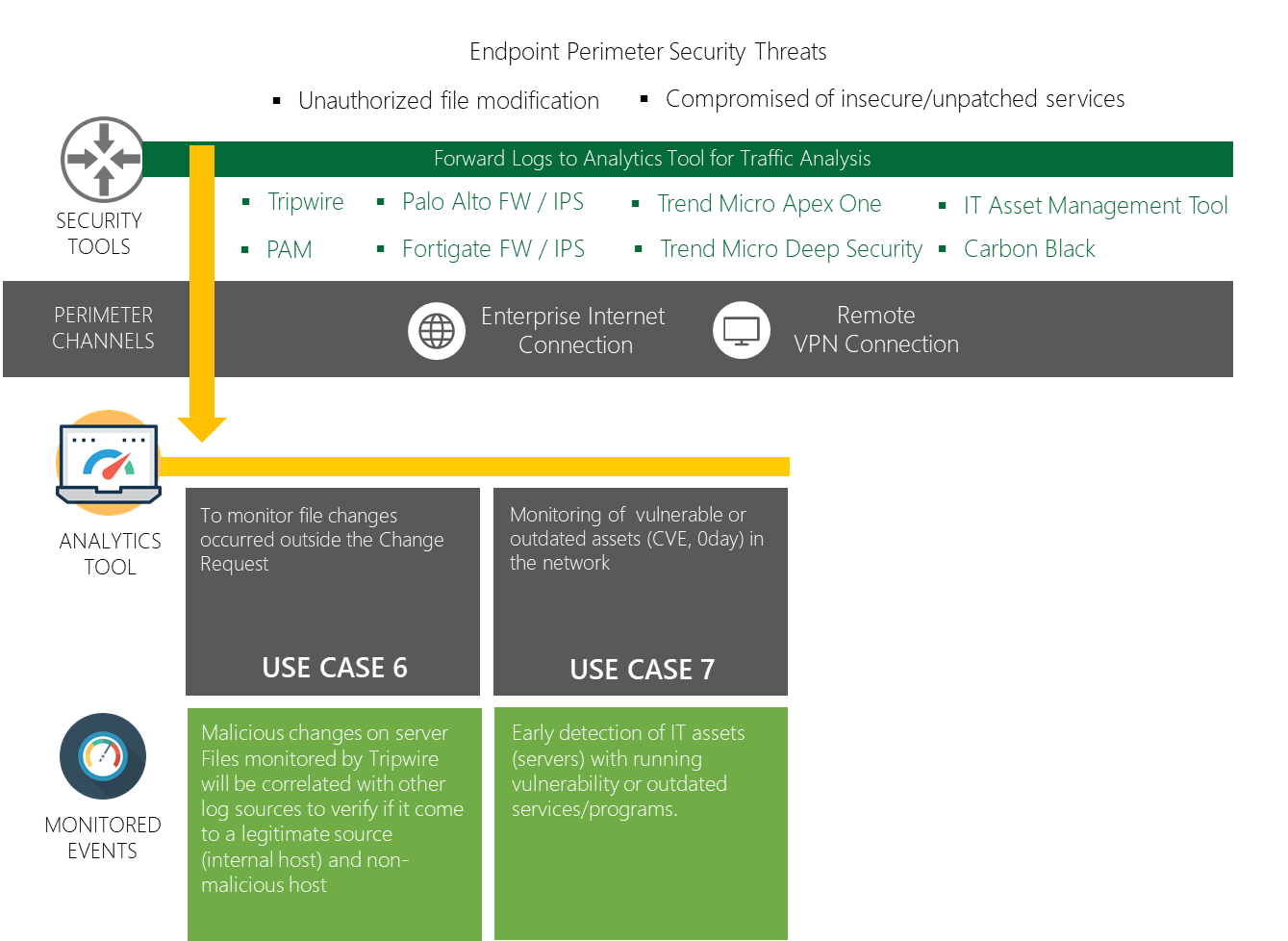
### Backdoor and Webshell

Recognize backdoor and web shell attempt, leading to gaining access or privilege escalation. Tools that can detect the malicious activity/traffic are:

* FireEye NX
* Perimeter FW / IPS
* Internal FW / IPS
* Carbon black
* Anomali CTI
* Trend Micro Deep Security
* Trend Micro Apex One

## Endpoint

Endpoint has 2 use cases with description as follows:



1. Description of IT Endpoint Use Cases

### File changes (Servers)

To monitor file changes occurred outside the Change Request. Tools that can detect the malicious activity/traffic are:

* Tripwire\*\*
* PAM - for admin access to server
* Carbon black
* Perimeter FW / IPS
* Internal FW / IPS
* Remedy for verification of authorized change

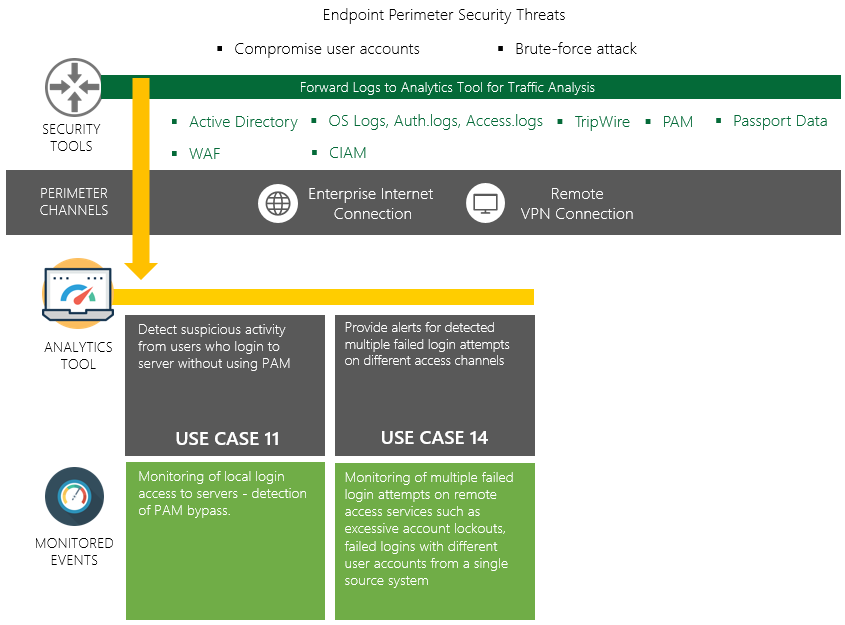
### Vulnerable and Outdated Components (Servers and workstations)

Monitoring of vulnerable or outdated assets (CVE, 0day) in the network. Tools that can detect the malicious activity/traffic are:

* Carbon black\*\*
* IT Asset Management tool (if available)

## Access

Access has 2 use cases with description as follows:



1. Description of IT Access Use Cases

### Users’ login to servers not using PAM

Detect suspicious activity from users who login to server without using PAM. Tools that can detect the malicious activity/traffic are:

* OS Logs - access log, authentication log
* PAM

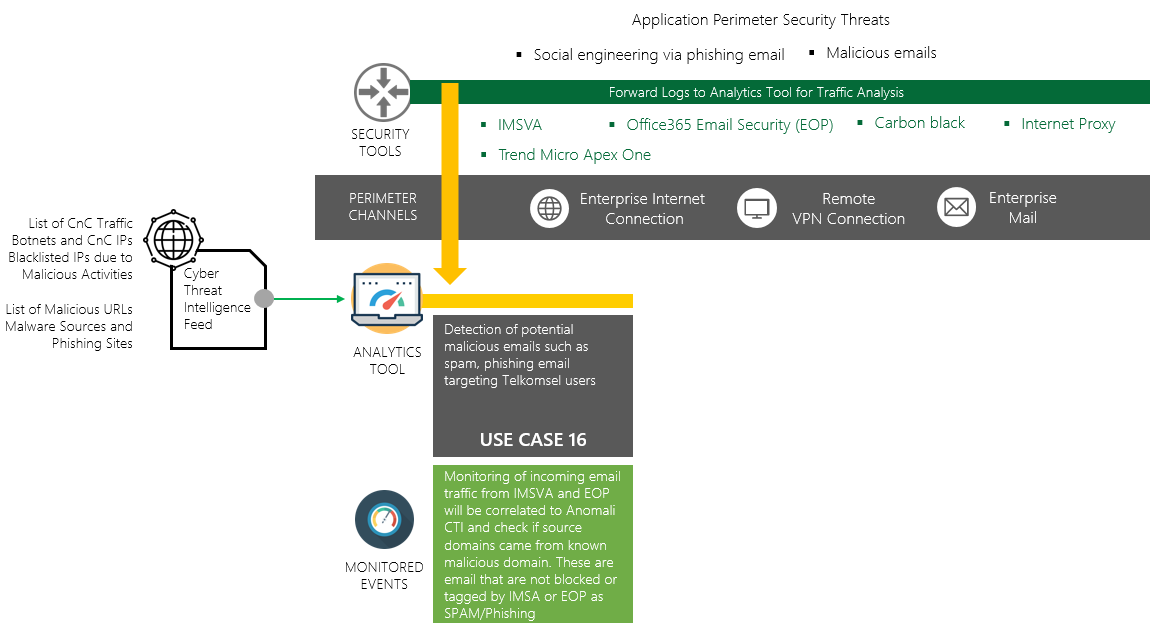
### Remote access brute-force

Provide alerts for detected multiple failed logins attempts on different access channels. Tools that can detect the malicious activity/traffic are:

* WAF - for injection attempts
* Active Directory
* PAM
* Passport Data
* OS Logs
* ForgeRock CIAM

## Fraud

Fraud has 1 use case with description as follows:



1. Description of IT Fraud Use Cases

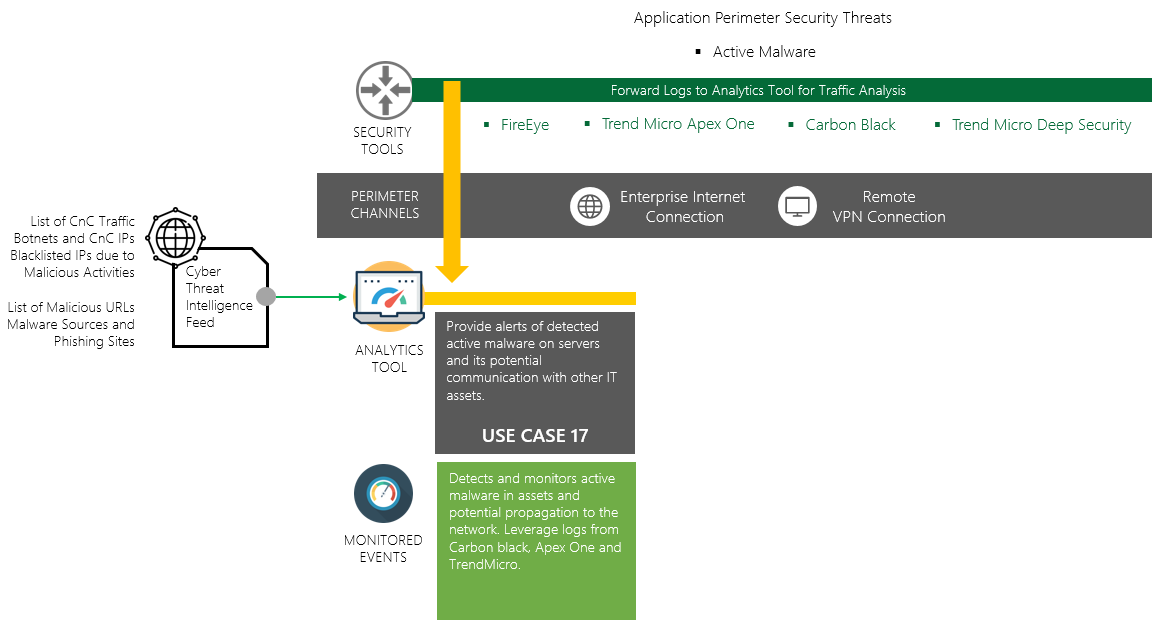
### Spam and Phishing: IP/Domain Phishing

Detection of potential malicious emails such as spam, phishing email targeting Telkomsel users IMSVA. Tools that can detect the malicious activity/traffic are:

* Office365 Email Security (EOP)
* Proxy
* Anomali CTI
* Carbon black - if it can detect IP/Domain

## Malware

Access has 1 use case with description as follows:



1. Description of IT Malware Use Cases

### Active Malware

Active Malware Provide alerts of detected active malware on servers and its potential communication with other IT assets. Tools that can detect the malicious activity/traffic are:

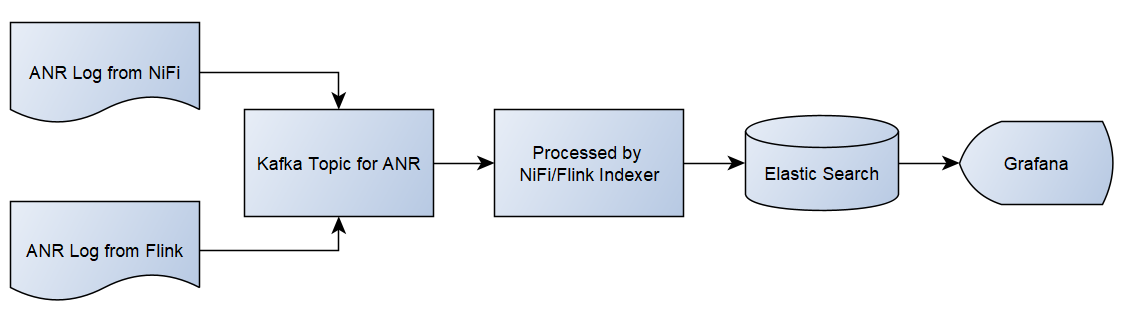
FireEye

* Carbon Black
* ApexOne
* TrendMicro Deep Security
* Anomali

The mentioned Use Cases are conducted in a discovery workshop to acquire understanding in the Telkomsel environment. Information gathered during these sessions will contribute to the Use Case dashboard design; as seen below as represented in a high level of design:

Analysis of the actual sample logs from log sources is required to properly formulate the use cases (Identify specific columns to be used for the use cases).

# Audit and Reconciliation



1. Process flow of Audit and Reconciliation

Any data-warehouse fetches records from several disparate systems and store them centrally in an enterprise-wide warehouse built on n-tier architecture and quality of data gets degraded in the process of centralization if proper audit and reconciliation framework does not exist. Failure due to any such issue can result into potential information loss leading to unreliable data quality for business process decision making.

Audit and Reconciliation framework is implemented using Grafana dashboard. The data source might be failed to be received by the NiFi because the Network does not send data. Grafana will check periodically for the log data generated by ANR and send the alert if the log contains no record. ANR log also will be produced by Flink in every data processing steps in order to measure & validate the accuracy of end results.

# Application Environments

From the environmental point of view, there is a physical separation between the development and production environments, below is the proposed environment used according to the application development life cycle.

## Development Environment

Each developer has access to this environment to do data exploration and test the compiled script to run a specific job. User should have access to the data sources.

### SIT Environment

SIT is the environment where the code is testing with all the surrounding system along with all the integrations for the system. All the integrated system communication and data flows are tested in this environment.

### UAT Environment

User acceptance is a key stage until the code is considered production ready. All the data objects are validated by the ITQA team to ensure the code quality and data accuracy before the production development. All the UAT test cases need to be passed in the UAT environment before the UAT sign-off is provided by the ITQA team which is mandatory gating approval required for implementation of any major solution as per Telkomsel process. To ensure there is no interference of any other activity in the system, UAT environment is segregated from all other environments.

### NFT Environment

This is environment is built with the key objective of performance testing and data security testing of the solution. To ensure proper performance test, this environment is closest to production in terms of resource allocations. Results publish in this environment are published. These results would be compared with the expected performance, benchmark with the existing system and ensure all key SLAs met.

# Job Scheduling and Monitoring

The operational scheduling and monitoring will be done via Airflow UI, Superset and Cloudera Manager which all are web-based application that can be easily accessed via web browser. These solutions provide a dashboard for administrators and managed-operations personnel to do these following tasks:

* Monitor utilized CPU and memory of analytic ops resource pool from Cloudera Manager
* For Datamart purpose : pre-requisite / pre-run will be added to validate the data source completeness
* Monitor ABT & Datamart jobs or pipelines (DAGs) from Airflow GUI including run status and elapsed time
* Monitor ABT data quality by measuring daily trend of size and high level count numbers from Superset dashboard

# Configuration & Deployment Management

All deployments will be done after getting approval from release manager. To ensure continuous development and parallel execution of various modules and future requirements, CICD pipeline is being setup on GitLab SCM repository.

* 1. ABT

Below are the deployment objects for FTP-ABT:

1. ./f2p-abt/run #all executable scripts are here (\*.sh)

<https://cicd-gitlab.telkomsel.co.id/f2p_s247/abt-pipelines/tree/master/run>

1. ./f2p-abt/ops #all Spark scripts contain ABT transformation logic are here (\*py)

<https://cicd-gitlab.telkomsel.co.id/f2p_s247/abt-pipelines/tree/master/ops>

1. ./f2p-abt/utils #all common shared function such helper and config files are here (\*.py)

<https://cicd-gitlab.telkomsel.co.id/f2p_s247/abt-pipelines/tree/master/utils>

1. ./f2p-abt/dags #all airflow DAG scheduler scripts are here (\*.py)

<https://cicd-gitlab.telkomsel.co.id/f2p_s247/abt-pipelines/tree/master/dags>

Below are proposed GitLab branch naming format for FTP-ABT:

1. master (production version code)
2. <username>-dev-<feed\_name> #this branch is for specific feature on existing feed or new feed development including adding function to executable files, pyspark transformation logic script and DAG scheduling script.
   1. Datamart
3. Contains all compiled Jar from Hgrid247:

<https://cicd-gitlab.telkomsel.co.id/f2p_s247/bi-datamart/tree/dev-sprint-3/Jar>

1. Contains all config file for shell script:

<https://cicd-gitlab.telkomsel.co.id/f2p_s247/bi-datamart/tree/dev-sprint-3/Config>

1. Contains all executable script:

https://cicd-gitlab.telkomsel.co.id/f2p\_s247/bi-datamart/tree/dev-sprint-3/Script

# Reference

Reference of this document is:

* Meeting and discussion
* Business Requirement Document
* Security Requirement

The following Telkomsel Information Security Standards are reformed as Security Requirement development of this application:

|  |  |  |
| --- | --- | --- |
| **#** | **Standard** | **File Location** |
| 1 | ISMS/S01 – API Security Standard | https://tsel.me/StandardKeamananInformasi |
| 2 | ISMS/S21 - Secure System Development Lifecycle | https://tsel.me/StandardKeamananInformasi |
| 4 | ISMS/S04 – Data Classification Standard | https://tsel.me/StandardKeamananInformasi |
| 3 | ISMS/S30 – Security System Standard | https://tsel.me/StandardKeamananInformasi |

# Appendix

BoQ on this project are listed as follow:

|  |  |  |
| --- | --- | --- |
| **Sprint** | **No** | **BoQ Item** |
| Sprint 1 |  | Project Management |
|  | Environment Setup (Dev + Prod) |
|  | Sigma + ELK Platform Engine |
|  | TheHive Project Platform Engine |
|  | Additional Features: IT Features 1-9 |
|  | Phase 1 Alert |
| Sprint 2 |  | Project Management |
|  | Data: Discovery, Ingestion, Identification, Exploration |
|  | Use Cases Scenario & BRD |
|  | IT UC # 1 |
|  | IT UC # 2 |
|  | IT UC # 8 |
|  | IT UC # 11 |
|  | IT UC # 16 |
|  | IT UC # 21 |
|  | IT UC # 12 |
|  | IT UC # 18 |
|  | IT UC # 20 |
|  | IT UC # 13 |
|  | IT UC # 17 |
|  | IT UC # 19 |
| Sprint 3 |  | Project Management |
|  | Data: Discovery, Ingestion, Identification, Exploration |
|  | Use Cases Scenario & BRD |
|  | IT UC # 14 |
|  | IT UC # 15 |
|  | IT UC # 9 |
|  | IT UC # 10 |
|  | IT UC # 22 |
|  | CGNAT Adjustment |
|  | IT UC # 3 |
|  | IT UC # 4 |
|  | IT UC # 5 |
|  | IT UC # 6 |
|  | IT UC # 7 |

Table 5 List of BoQ