5Genesis

5TH GENERATION END-TO-END NETWORK, EXPERIMENTATION, SYSTEM INTEGRATION, AND SHOWCASING

[H2020 - Grant Agreement No. 815178]

Deliverable D3.13

5G Security Framework (Release A)

Editor A. Kapodistria (SHC)

Contributors SHC (Space Hellas), INF (INFOLYSIS P.C.)

Version 1.0

Date October 15th, 2019

Distribution PUBLIC (PU)

List of Authors

SHC	Space Hellas (Cyprus) Ltd.
A. Kapodistria, G. Gardikis, D. Lioprasitis	
INF	INFOLYSIS P.C.
V. Koumaras, G. Theodoropoulos	

© 5GENESIS Consortium

Disclaimer

The information, documentation and figures available in this deliverable are written by the 5GENESIS Consortium partners under EC co-financing (project H2020-ICT-815178) and do not necessarily reflect the view of the European Commission.

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The reader uses the information at his/her sole risk and liability.

© 5GENESIS Consortium

Copyright

Copyright © 2019 the 5GENESIS Consortium. All rights reserved.

The 5GENESIS Consortium consists of:

NATIONAL CENTER FOR SCIENTIFIC RESEARCH "DEMOKRITOS"	Greece
AIRBUS DS SLC	France
ATHONET SRL	Italy
ATOS SPAIN SA	Spain
AVANTI HYLAS 2 CYPRUS LIMITED	Cyprus
AYUNTAMIENTO DE MALAGA	Spain
COSMOTE KINITES TILEPIKOINONIES AE	Greece
EURECOM	France
FOGUS INNOVATIONS & SERVICES P.C.	Greece
FON TECHNOLOGY SL	Spain
FRAUNHOFER GESELLSCHAFT ZUR FOERDERUNG DER ANGEWANDTEN FORSCHUNG E.V.	Germany
IHP GMBH – INNOVATIONS FOR HIGH PERFORMANCE MICROELECTRONICS/LEIBNIZ-INSTITUT FUER INNOVATIVE MIKROELEKTRONIK	Germany
INFOLYSIS P.C.	Greece
INSTITUTO DE TELECOMUNICACOES	Portugal
INTEL DEUTSCHLAND GMBH	Germany
KARLSTADS UNIVERSITET	Sweden
L.M. ERICSSON LIMITED	Ireland
MARAN (UK) LIMITED	UK
MUNICIPALITY OF EGALEO	Greece
NEMERGENT SOLUTIONS S.L.	Spain
ONEACCESS	France
PRIMETEL PLC	Cyprus
RUNEL NGMT LTD	Israel
SIMULA RESEARCH LABORATORY AS	Norway
SPACE HELLAS (CYPRUS) LTD	Cyprus
TELEFONICA INVESTIGACION Y DESARROLLO SA	Spain
UNIVERSIDAD DE MALAGA	Spain
UNIVERSITAT POLITECNICA DE VALENCIA	Spain
UNIVERSITY OF SURREY	UK

This document may not be copied, reproduced or modified in whole or in part for any purpose without written permission from the 5GENESIS Consortium. In addition to such written permission to copy, reproduce or modify this document in whole or part, an acknowledgement of the authors of the document and all applicable portions of the copyright notice must be clearly referenced.

Version History

Rev. N	Description	Author	Date
1.0	Release of D3.13	A. Kapodistria (SHC), A. Díaz-Zayas (UMA)	15/10/2019

© 5GENESIS Consortium

LIST OF ACRONYMS

Acronym	Meaning
API	Application Programming Interface
CA	CONSORTIUM AGREEMENT
DARE	Data Analysis and Remediation Engine
GA	GRANT AGREEMENT
GUI	Graphical User Interface
HDFS	Hadoop Filesystem
HW/SW	Hardware / Software
MEC	Mobile Edge Computing
ML	Machine Learning
NFV	Network Functions Virtualisation
RAN	Radio Access Network
SDN	Software-Defined Networking
YARN	Yet Another Resource Negotiator

© 5GENESIS Consortium

Executive Summary

5G comes with a new rich set of features and capabilities, which, in addition to their obvious technical and business value, are challenged with various side-effects. Such as side-effect is the drastic increase in the attack surface, compared to legacy cellular network infrastructures.

One of the objectives of 5GENESIS is to facilitate and enhance the security of 5G networks via its Security Framework that is built around a Security Analytics platform. 5G Security Analytics refers to the collection and joint analysis of massive heterogeneous data from multiple points of the 5G infrastructure utilized for integrated monitoring. The ultimate aim is the detection and classification of anomalies associated with security incidents, using state-of-the-art ML techniques.

The ambition of 5GENESIS is to realise an effective and efficient 5G Security Analytics framework by:

- Ingesting and adapting heterogeneous data coming from multiple sources
- Selecting and adapting the proper Machine Learning algorithms, also combining them into complex workflows
- Enabling intuitive visualization and query for facilitating security operations.

5GENESIS builds on well-established technologies from the Big Data / Machine Learning realm, such as Apache Hadoop, Spark and Kafka, while it heavily relies on the capabilities of Apache Spot project on cybersecurity analytics.

The baseline for the 5GENESIS developments is the DARE (Data Analysis and Remediation Engine) platform, developed in the frame of the EU SHIELD project, which in turn is based on Apache Spot. The 5GENESIS Security Analytics platform operates in three stages, referred to as (a) Data Acquisition, Transformation and Storage; (b) Data Analysis; and (c) Visualisation and Export.

The main extensions which are underway in 5GENESIS are:

- Adaptation of the data acquisition and storage phase in order to integrate information from multiple sources and multiple formats as well as to support data anonymization.
- Extension of the data analysis phase to include multiple ML algorithms and consolidate their output, selection and adaptation of the appropriate algorithms (currently Apache Spot only uses one algorithm at a time)
- Adaptation of the GUI and API components to suit the 5GENESIS visualization needs and data models.
- Integration with already existing security analytics platform, such as Splunk, so as to give better insights to data visualization and overall data handling.

The current implementation (Release A) features a fully functional ingestion and storage chain (for NetFlow data), a single ML algorithm (LDA) as well as the native GUI of Apache Spot.

Over the next phases of the project, the platform will be finalized and integrated in the Facility, while its assessment will be evaluated over representative usage scenarios. The next version of this deliverable, D3.14, will present the final release of the platform and discuss its operational evaluation and the generated results.

© 5GENESIS Consortium

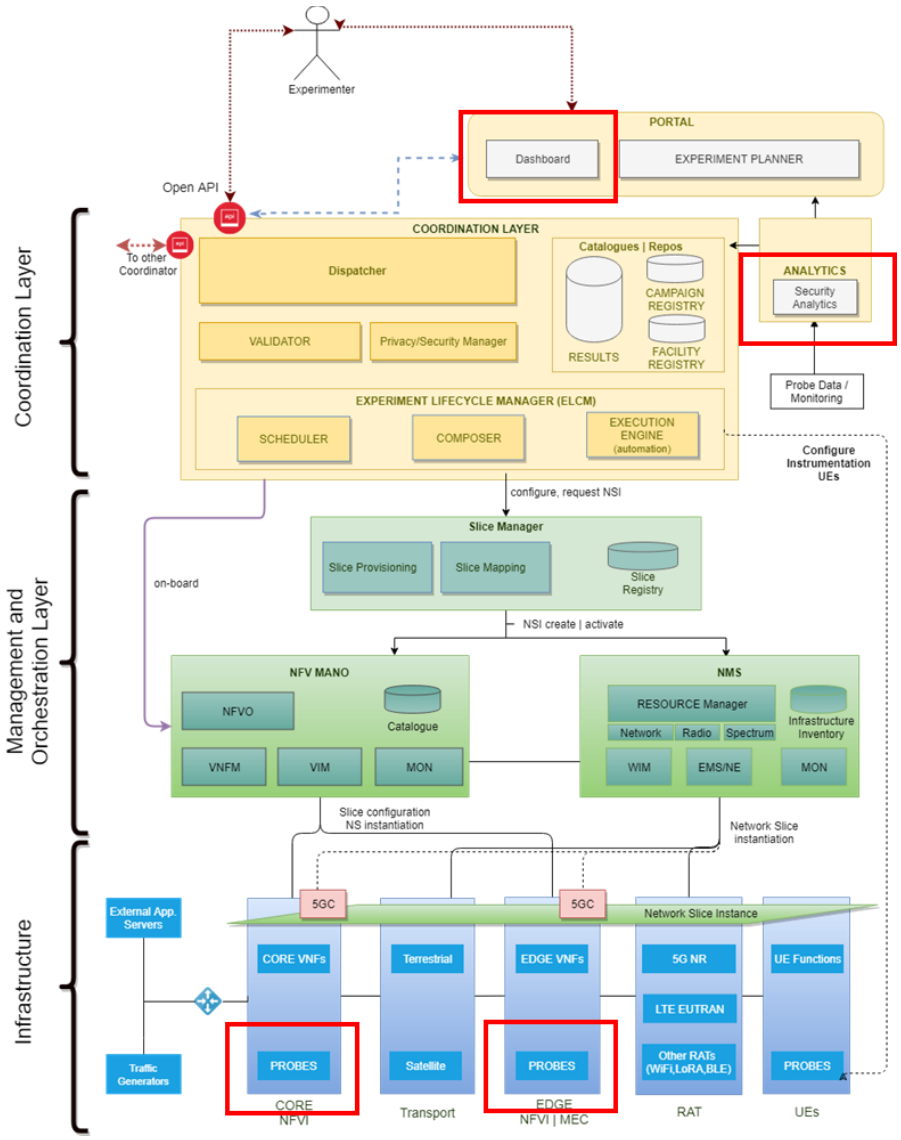

Figure 1 Security framework components

Figure 1 shows the components of the Security framework as part of the 5GENESIS architecture.

© 5GENESIS Consortium

Table of Contents

© 5GENESIS Consortium

1. INTRODUCTION

1.1. Purpose of the Document

The emergence of 5G networks is admittedly accompanied by a significant increase of the attack surface of the telecom infrastructure. New vulnerabilities are identified, in particular associated with the capabilities related to network softwarisation and slicing.

In this context, among its other features, 5GENESIS includes a Security Analytics platform as a contribution towards hardening the security of next-generation networks. This document presents the technical approach for the 5GENESIS Security analytics platform, as well as the relevant work that has been done and is planned to be done under WP3/Task 3.7.

1.1.1. Document dependencies

This document is based on specifications, requirements and assumptions as discussed in the first release of the Architecture related deliverables. The table below summarizes the relevance towards the deliverables produced by WP2.

id	Document title	Relevance
D2.1 [1]	Requirements of the Facility	The document sets the ground for the first set of requirements related to supported features at the testbed for the facilitation of the Use Cases.
D2.2 [2]	5GENESIS Overall Facility Design and Specifications	The 5GENESIS facility architecture is defined in this document. The list of functional components to be deployed in each testbed is defined.
D2.3 [3]	Initial planning of tests and experimentation	Testing and experimentation specifications that influence the testbed definition, operation and maintenance are defined.

1.2. Structure of the Document

The document is structured as follows:

- Chapter 1, *Introduction* (the present section)
- Chapter 2, *Motivation and Scope*, presents the motive behind this work and the scope of it and gives details about security in 5G and 5G security analytics.
- Chapter 3, *Foundation Technologies*, refers to the key technologies that are used to implement the 5GENESIS security framework.
- Chapter 4, *Overview of the 5GENESIS Security Analytics Platform,* presents the high-level architecture of the security platform to be used in the project.

© 5GENESIS Consortium

- Chapter 5, *Work so far, Release A features and Future Work*, refers to the progress that has been made in the course of the project, the features of Release A, as well as the steps ahead
- Chapter 6, *Conclusions*, concludes the document.

1.3. Target Audience

The document is initially targeted to the 5GENESIS project team, towards establishing a common understanding of the architecture and functionalities of the Security Analytics platform and identifying the technical steps needed for its smooth integration in the 5GENESIS Facility.

However, since most of the document is not specifically tied to the project, it can also be addressed to the wider 5G community. Since the outcome of this activity will mostly be open-sourced, this deliverable can be used by essentially anyone who either wishes to replicate the entire 5GENESIS architecture, or just plans to individually exploit and integrate the Security Analytics platform in another 5G system.

© 5GENESIS Consortium

2. MOTIVATION AND SCOPE

2.1. Security aspects in 5G

5G comes with a new rich set of features and capabilities, which, in addition to their obvious technical and business value, as expected, are accompanied with various side-effects, one of the most important of which is the drastic increase in the attack surface, compared to legacy cellular network infrastructures. Some of these 5G-specific capabilities, which, under certain circumstances, may introduce new vulnerabilities and increase the probability of a security incident, are:

- Software-defined infrastructures. The dynamic nature of service-oriented infrastructures introduced in the 5G landscape (be it SDN/NFV, containerised services and applications, etc.) makes it easy to deploy and configure services at the click of a button. In addition, the heterogeneity of HW/SW components and technologies, combined with the fast development cycles associated with the agile paradigm campaigned by most commercial adopters, can drastically increase attack surfaces and the related security/privacy risks of individual components as a whole. Software-based virtual appliances and SDN rules can be subject to malicious modifications, associated with various risks, from data breach to interruption of critical services and applications.
- Slicing and multi-tenancy. The vision of 5G is to enable end-to-end virtualization and true multi-tenancy by dividing the network to slices to be provided to multiple users. However, while in theory slices are expected to be fully isolated, in practice this isolation will be much weaker as expected. This implies the potential of either accidental or malicious inter-slice incidents, while operations and events in one slice affect the neighbouring ones.
- Multi-actor service paradigms. In addition to multi-tenancy, 5G promises the further decoupling of service, network and infrastructure providers, allowing multiple actors to be engaged in the 5G ecosystem and share resources, often in a dynamic manner. This has obvious security implications, mostly associated with privacy.
- Complex, multi-tier architectures. Apart from the traditional core/backhaul/RAN segmentation, 5G networks introduce additional architectural blocks, such as NFV/SDN Management and Orchestration, in-network compute resources (for NFV/MEC enablement), integration of heterogeneous backhaul/access technologies (such as satellite), cross-domain management functions etc. This increased complexity of the data and, most importantly, the control plane, naturally multiplies the overall system vulnerabilities.

In 5G, not only the probability of a security incident increases, but also the expected impact and severity. The connection of more and more devices in a 5G network, many of which track personal data, while others support critical operations (as in Intelligent Transport Systems/connected cars or e-health), implies that security incidents can lead to severe privacy breach and/or even life-threatening situations.

© 5GENESIS Consortium

It is evident that the emergence of 5G calls for significantly stricter security controls compared to legacy network. A more thorough investigation of the 5G security landscape is out of the scope of the present document and can be found in white papers and reports such as the one[1] produced by the Security Working Group of the 5G PPP, which 5GENESIS is attending and closely following.

However, at the same time, as a counterbalance to increased risks, 5G technologies also provide new capabilities to establish mitigation measures and contingencies. A key capability is network softwarisation (through SDN and NFV) which can be at the same time a strength and a weakness (as explained above). While SDN/NFV indeed increases the attack surface, at the same time it offers the capability to dynamically deploy virtual security appliances in the network, at the core and also at the edge, and selectively divert traffic through them for enhanced detection/prevention. This capability has been promoted via several projects which leverage software-based networks for security, including the EU SHIELD project[5].

Another interesting opportunity stems from the fact that 5G network components are highly heterogeneous and distributed across the network, thus creating an enormous amount of diverse data (mostly logs and monitoring information), whose timely analysis can lead to effective inference of security incidents. This is the concept of 5G security analytics, which is targeted in 5GENESIS and detailed in the next subsection.

2.2. 5G Security Analytics

5G Security Analytics refers to the collection and joint analysis of massive heterogeneous data from multiple points of the 5G infrastructure for integrated monitoring, with specific focus on detecting and classifying anomalies associated with security incidents.

Big Data and Machine Learning technologies are the ideal foundations towards this goal. Big Data infrastructures enable the scalable ingestion, storage and analysis of massive data, also in real-time. At the same time, state-of-the-art Machine Learning (ML) algorithms enable the identification of incidents, which will go unnoticed using traditional rule-based detection. This enables i) the detection of zero-day attacks, whose exact digital fingerprint is unknown and ii) the proactive identification of threats even at their very early stages, where detection thresholds of traditional methods have not been crossed.

ML techniques can currently counter Cyber-attacks [6] in two different ways: first, with anomaly detection, where unsupervised algorithms are trained to learn the trusted behaviour in order to detect any irregular ones; second, with threat classification, where supervised learning is used to classify known attacks. The former aims to detect 0-day attacks while the latter is more effective for known attacks. Nowadays, existing cybersecurity solutions combine both techniques: first, they apply anomaly detection, looking for malicious flows, and then use threat classification to identify and classify the type of attack. Current unsupervised anomaly detection algorithms can be sorted in three categories [7]:

- Reconstruction: is based on data compression and posterior reconstruction. Behaviours reconstructed with low error rate are considered “normal” while high error rate ones are considered anomalous.
- Boundary: focuses on finding boundaries around the normal behaviour and every behaviour outside the defined range is considered as anomalous.

© 5GENESIS Consortium

- Density Estimation: estimates the probability density function of the training data, it discriminates anomalous behaviours using a threshold value.

The most used supervised classification algorithms in cybersecurity are Support Vector Machines (SVM), Decision trees and Naïve Bayes classifiers. More specifically, [8] has shown that SVMs can be successful in the task of traffic classification, and [9] shows that tree-based methods exhibit very high accuracy measures, while also reducing the need of feature pre-processing. Algorithms based on neural networks are also being considered, but their use seems to be mostly restricted to data of high dimensionality.

The ambition of 5GENESIS is to realise an effective and efficient 5G Security Analytics framework by:

- Ingesting and adapting heterogeneous data coming from multiple sources
- Selecting and adapting the proper ML algorithms, also combining them into complex workflows
- Enabling intuitive visualization and query for facilitating security operations.

The next chapter presents the foundation technologies, which are going to be exploited in 5GENESIS towards the development of the 5G Security Analytics framework.

© 5GENESIS Consortium

3. FOUNDATION TECHNOLOGIES

3.1. Apache Spark, Kafka and the Hadoop ecosystem

3.1.1.1. Hadoop Ecosystem

The Apache Hadoop [10] software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than relying on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly available service on top of a cluster of computers, each of which may be prone to failures. There are ***four major elements of Hadoop*** i.e. *HDFS, MapReduce, YARN and Hadoop Common*. Along with these elements, there are extra tools that support/supplement them and together they constitute Hadoop Ecosystem, as shown in Figure 2. Combined together, these tools provide services such as absorption, analysis, storage and maintenance of data etc.

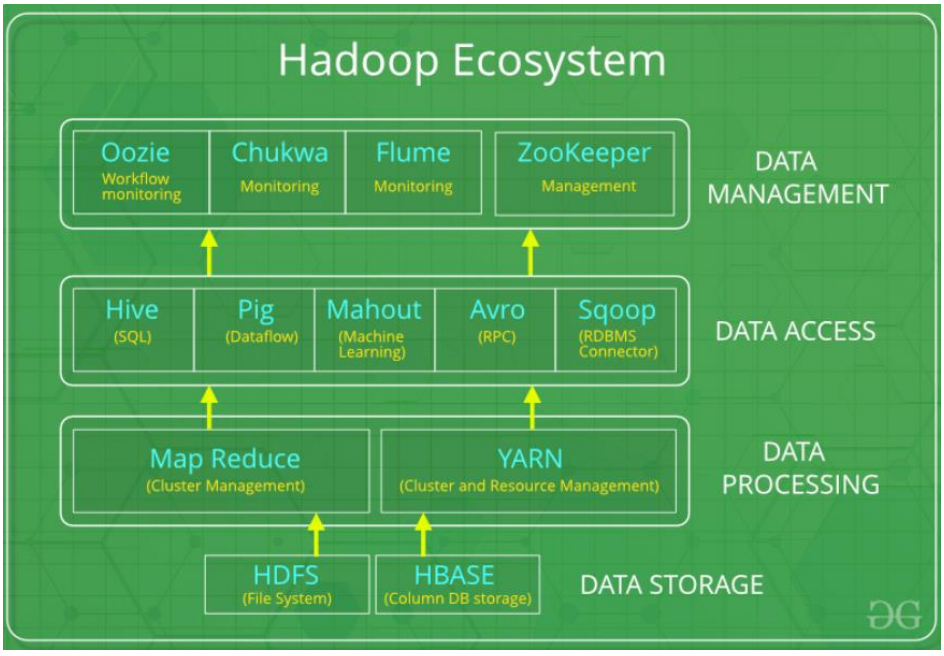

Figure 2 Hadoop Ecosystem (source: https://www.geeksforgeeks.org/hadoop-ecosystem/)

Hadoop is based on a storage component, Hadoop Distributed File System (HDFS), and on a processing part which follows the MapReduce programming model. Operational wise, Hadoop splits files into large blocks and distributes them across nodes in a cluster. It then transfers packaged code into nodes to process the data in parallel. This approach takes advantage of data locality, where nodes manipulate the data they have access to. This allows the dataset to be processed faster and more efficiently than it would be in a more conventional supercomputer architecture that relies on a parallel file system where computation and data are distributed via high-speed networking.

© 5GENESIS Consortium

The Hadoop framework, as already mentioned at the beginning of this section, is composed of the following modules:

- *Hadoop Distributed File System (HDFS)* – a distributed file-system that stores data on commodity machines, providing very high aggregate bandwidth across the cluster;
- *Hadoop MapReduce* – an implementation of the MapReduce programming model for large-scale data processing.
- *Hadoop YARN* – a platform responsible for managing computing resources in clusters and using them for scheduling users' applications;
- *Hadoop Common* – contains libraries and utilities needed by other Hadoop modules;

The 5GENESIS security analytics platform is based on Hadoop/HDFS for scalable and distributed storage of data (see Sec. 4.5).

3.1.1.2. Apache Kafka

Apache Kafka [11] is a distributed streaming platform, used for building real-time data pipelines and streaming apps. Kafka publishes and subscribes to streams of records, similar to a message queue or enterprise messaging system. It also stores streams of records in a fault-tolerant durable way and processes streams of records as they occur. Kafka can also connect to external systems (for data import/export) via Kafka Connect and provides Kafka Streams, a Java stream processing library. It can be used in two main categories of applications:

- Real time streaming data pipelines that get data between systems or applications.
- Real-time streaming applications that transform or react to the streams of data.

Kafka runs on cluster of one or more servers than can span multiple datacenters and stores streams of records in categories called topics. Each record is consisted of a key, a value and a timestamp.

Kafka has four core APIs:

- *Producer API*: allows an application to publish a stream of records to one or more Kafka topics.
- *Consumer API*: allows an application to subscribe to one or more topics and process the stream of records produced to them.
- *Streams API*: allows an application to act as a stream processor, consuming an input stream from one or more topics and producing an output stream to one or more output topics, effectively transforming the input streams to output streams.
- *Connector API*: allows building and running reusable producers or consumers that connect Kafka topics to existing applications or data systems. For example, a connector to a relational database might capture every change to a table.

As already discussed, Kafka topics are categories where streams of records are stored. Topics are always multi-subscriber; meaning that a topic can have zero, one, or many consumers that subscribe to the data written to it.

Two important APIs of Kafka that need to be explained further are producers and consumers.

© 5GENESIS Consortium

- *Producers*: publish data to the topics of their choice and are responsible for choosing which record to assign to which partition within the topic. This can be done in a round-robin fashion to balance load or it can be done according to some semantic partition function (say based on some key in the record).
- *Consumers*: assign themselves a consumer group name and each record published to a topic is delivered to one consumer instance within each subscribing consumer group. Consumer instances can be in separate processes or on separate machines. If all the consumer instances have the same consumer group, then the records will effectively be load balanced over the consumer instances. If all the consumer instances have different consumer groups, then each record will be broadcast to all the consumer processes.

The 5GENESIS security analytics platform, (see Sec. 4.4) uses Kafka to ingest data and transform/normalize them prior to their analysis, which is done using Spark (next section). In the project, Kafka is used as streaming platform in the sense that after the data collection, Kafka's role is to transfer these data into the HDFS for storage. Producers publish collected data to topics and then consumers are responsible for delivering each of these topics to a consumer group.

3.1.1.3. Apache Spark

Apache Spark [12] is a distributed and highly scalable cluster – computing framework. It provides an interface for programming entire clusters with implicit data parallelism and fault tolerance. There are ***four main modules***, which all are interoperable, meaning that data can pass between them:

- *MLib*: offers machine learning functionality.
- *GraphX*: offers big data in memory graph processing fast.
- *SQL*: provides the ability to process data in tabular forms and with tabular functions. It is integrated with Parquet and JSON formats in order for data to be represented in better formats and integrate with external systems. Moreover, Spark can be used by Hive as a processing engine.
- *Streaming*: data in Spark are processed as streams and cover a variety of topics such as transformations, output operations, etc.

The 5GENESIS security analytics platform , as detailed in Chapter 4, uses Spark to i) more quickly process and transform incoming data and ii) perform the core analytics functionality, detecting and classifying incidents. For the purposes of our work we mainly exploit SQL and Streaming modules.

3.2. Apache Spot

Apache Spot [13] is an open source software that leverages information from flow and packet analysis. It promotes threat detection and remediation using Machine Learning and integrates all security data into a comprehensive dashboard based on open data models. This ecosystem of ML-based applications can run simultaneously either on a single or a shared data set to provide enterprises analytic flexibility and thus the ability to discover suspicious connections and unseen attacks.

© 5GENESIS Consortium

Spot architecturally is consisted of three parts, as shown in Figure 3.2:

- Data Sources: all the possible sources that provide data, either a streaming source or batch files.
- Spot GUI: all the services that constitute Spot's GUI and are responsible for visualizing results.
- Spot Landscape: the back-end part, which is responsible for operations like ingestion, transformation, machine learning, etc.

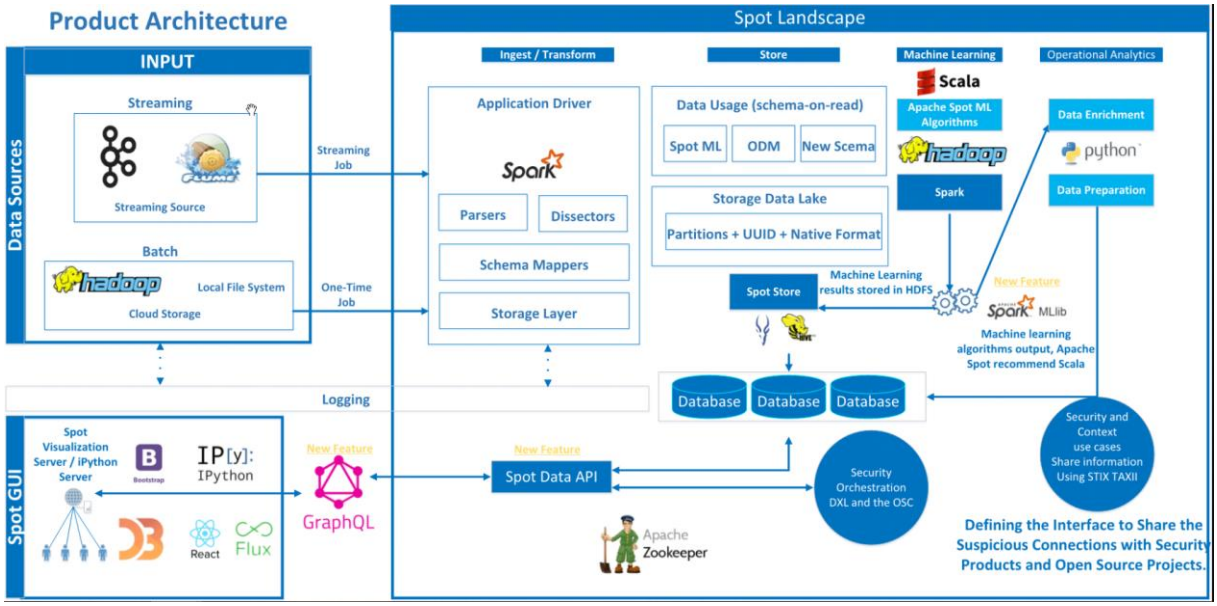

Figure 3: Architecture of Apache Spot (source: https://spot.apache.org/get-started/architecture/)

In turn, Spot is based on Hadoop, Kafka and Spark (see previous sections) for data storage, ingestion and processing. As explained in more detail under Chapter 4, 5GENESIS inherits most of the architectural concepts of Spot and re-uses its data model, basic ML algorithm (Latent Dirichlet Allocation, LDA), as well as GUI for basic visualization.

Based on Spot's data model and concepts, it is feasible to pursue high accuracy in anomaly detection, along with an informative visualization of the results. The ML algorithm that Spot uses, LDA, is a well-known algorithm for its performance and the quality of its results, with proven performance for anomaly detection in the cybersecurity domain. Moreover, Spot offers an integrated solution meaning that it includes operations for data collection, storage, analysis and results presentation. Taken into consideration the area that is covered from 5GENESIS, this solution is way more useful than implementing a component for each task from scratch.

The following chapter describes in detail the architecture and functional components of the 5GENESIS security analytics solution.

© 5GENESIS Consortium

4. Overview of the 5GENESIS Security Analytics platform

4.1. Overall architecture

The 5GENESIS Security Analytics platform is based on the Data Analysis and Remediation Engine (DARE) developed in the SHIELD project [5], which in turn is using Apache Spot, HDFS, Kafka and Spark, as presented in the previous chapter.

The Security Analytics platform consists of a central data analytics engine and a distributed set of data collection components. Following a Big Data approach, the data value elicitation is divided in three categories, as shown in Figure 4 below:

1. Data acquisition, transformation and storage
2. Data analysis
3. Visualisation and export

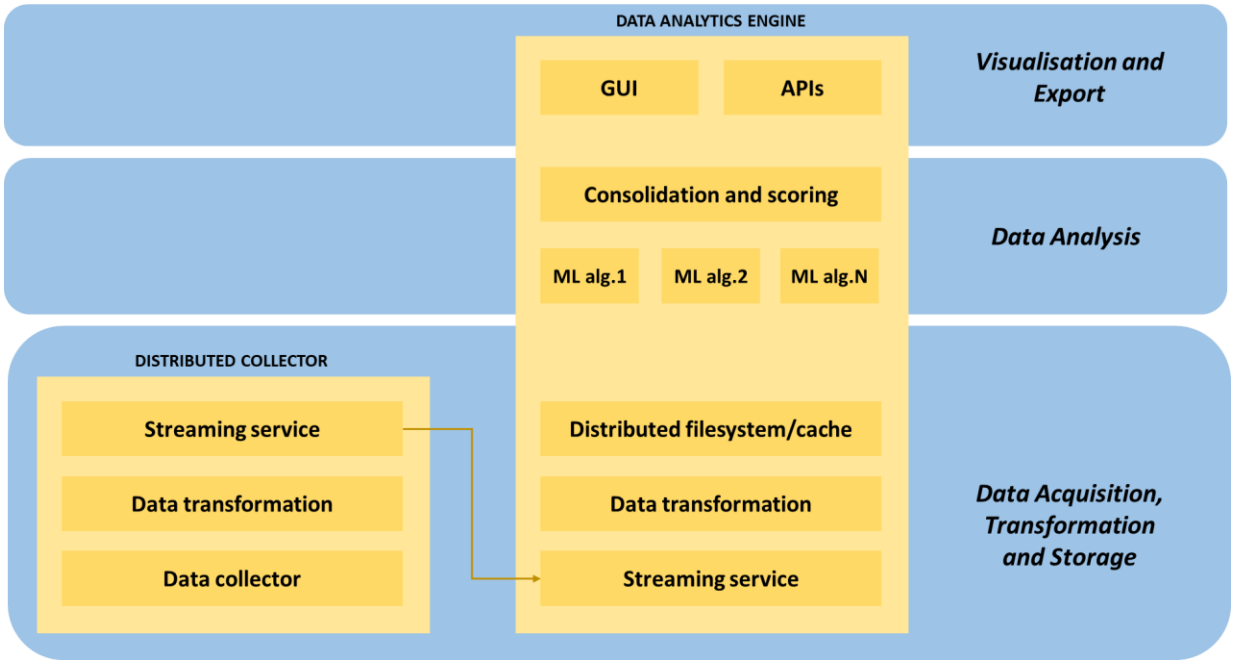

Figure 4: Functional components of the 5GENESIS Security Analytics platform

Below is a brief description of the involved subcomponents of the 5GENESIS Security Analytics platform.

© 5GENESIS Consortium

4.2. Data collection

The data collector is responsible for acquiring the data generated from remote endpoints (VNFs, network elements, etc.). Currently, data collection focuses on network flow (NetFlow) information, however in the frame of the project more data sources will be supported and additional interfaces will be developed. NetFlow data is either captured locally (e.g. in a VNF using dedicated collector agents), or received from a network element (e.g. firewall, router), which is capable of transmitting such information; most medium/high-end models found in the market already support this feature.

4.3. Data transformation

The data transformation stage is responsible for transforming the format-specific data into a structured, generic format. Data collection (Sec. 4.2) and transformation can follow two approaches:

- Option 1 – Centralised architecture: only the collection of the data is distributed, while all the other functionalities are centralised in the Data analysis phase.
- Option 2 – Distributed architecture (as seen in Figure 4): the data collection and the data transformation are distributed at the collector agent and hence, the data is sent to the central engine in a standard format (e.g. CSV).

Data collection and transformation follows the procedures supported by Apache Spot, mainly for collecting NetFlow data and converting them to a structured format (CSV, tables). However, while the centralized architecture (Option 1) is the only method supported by Apache Spot, in 5GENESIS we take advantage of the distributed architecture (Option 2), as initially developed in the SHIELD project and further extended in 5GENESIS. It has been shown that the distributed approach can achieve, up to 10-fold decrease in processing time, compared to the centralized one.

4.4. Streaming service

The streaming service sends the information from the monitoring vNSF to the data analytics central engine, assuring reliability on the communication. The streaming service is based on Apache Kafka (Sec. 3.1.1.2.), thus achieving both scalability and versatility.

The streaming service splits the network data into smaller specific topics and smaller partitions, while creating a data pipeline for each topic. The use of Kafka also allows the streaming stage to be reliable and fault tolerant for ensuring the integrity of the data and their quality in further processing steps.

4.5. Distributed filesystem/cache

The Distributed File System / Cache is responsible for storing the collected data for both, batch (i.e. hard disks) or real-time (i.e. cache) processing. Once the network data has been transformed, the input is stored in a distributed file system in both the original and modified

© 5GENESIS Consortium

formats (in the case of centralized processing) or only the modified/preprocessed (in the case of distributed processing). The distributed file system is responsible for storing the collected data and making them available, so that it can be accessible by search queries.

For storing data, the Hadoop filesystem (HDFS) (See Sec. 3.1.1.1.) is used, achieving scalability, integrity, and better performance for bulk data exchanges. Raw data are saved directly in the filesystem, while structured (preprocessed) information are stored in Hive tables. Hive[1] uses the Hadoop filesystem and facilitates reading, writing, and managing large datasets residing in distributed storage using SQL.

4.6. ML algorithms/ consolidation and scoring

The Data Analysis phase features cognitive and analytical functionalities capable of detecting network anomalies that are associated with specific vulnerabilities or threats. The processing and analysis of large amounts of data is carried out by using Big Data analytics and machine learning techniques. By processing data and logs from probes/collectors deployed at specific strategic locations of the network, the data analytics framework can link traffic logs that are part of a specific activity in the network and detect any possible anomaly. In case malicious activity is detected, the relevant alarms are produced.

The data analysis phase consists of two components, the ML family of algorithms as well as the consolidation and scoring stage.

The set of ML algorithms is responsible for the detection of anomalies in network traffic that will lead to the prevention or mitigation of potential threats. The machine learning engine works not only as a filter for separating bad traffic from benign, but also to characterise the unique behaviour of network traffic. It contains routines for performing suspicious connections analytics on data (currently NetFlow) gathered from the Data Acquisition phase and the built-in Distributed storage system subcomponent. These analytics consume a collection of network events to produce a list of the events that are considered to be the least probable, and these are considered the most suspicious.

The statistical model that is currently used for discovering abstract topics of these events and ultimately discovering normal and abnormal behaviour is a topic modelling algorithm called Latent Dirichlet Allocation [14]. LDA is a generative probabilistic model used for discrete data that is applied to network traffic by converting network log entries into words through aggregation and discretisation, to discover hidden semantic structures. LDA is the built-in algorithm of Apache Spot (See Sec. 3.2). Spot executes LDA routines using a Scala Spark implementation from MLlib, Apache Spark's scalable machine learning library. It should be noted that Spot's current capabilities do not include any anomaly classification algorithms that would interpret the detected outliers as specific threats/attacks, thus such an algorithm will be originally developed to meet this requirement. The module will exploit Spot's existing batch processing capabilities, coupled with the development of streaming analytics functionalities currently missing from Spot, to achieve real-time (or near real-time) visibility for threat detection.

[1] http://hive.apache.org/

© 5GENESIS Consortium

5GENESIS will leverage LDA and also employ other algorithms for anomaly detection and classification, such as autoencoders, CNNs etc. When multiple ML algorithms are used, a consolidation phase integrates their outputs. The threat scores of each data flow pipeline is aggregated for the extraction of the final incident report, which will be used as a platform output.

4.7. GUIs and APIs

For the visualization of information, 5GENESIS will use the GUI of Apache Spot as a starting point.

The main view of the GUI is depicted in Figure 5 below and aims at the visualization of the suspicious flows. The latter are displayed as a list (with the most suspicious ones at the top) as well as in a graph.

The operator is also allowed to manually score the detected flows, thus helping the algorithm to learn and eventually minimize false positives.

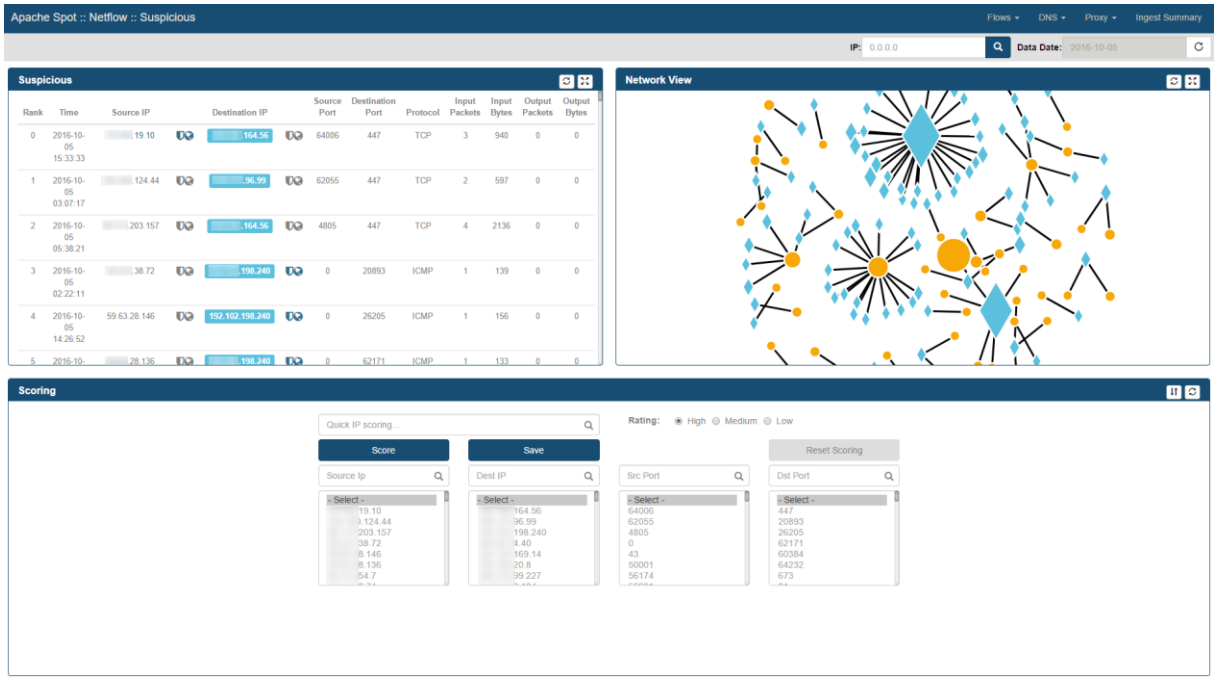

Figure 5. Apache Spot GUI (Suspicious connects view)

In 5GENESIS, the Spot GUI will be expanded in order to embrace also additional data coming from the architectural components (alerts etc.).

Also, an API will be developed in order to export data to third-party analytics platforms, as well as the 5GENESIS portal.

The next chapter details the progress so far on the implementation of the above-mentioned components.

© 5GENESIS Consortium

5. RELEASE A SUMMARY AND FUTURE PLANS

5.1. Work done so far

During the first months of T3.7, the first version of the Security Analytics (for the scope of 5GENESIS) was implemented using as baseline Apache Spot as well as results of the SHIELD project. The focus was on additional features to further optimize its operation.

In order to make the solution more efficient by optimizing the ingest chain, a distributed collector framework was created. The collector is responsible for data ingestion inside Spot and can be hosted in a virtual machine where all the traffic is directed to, to perform the collection. The distributed collector on the other side, can be hosted in several VMs and perform the same task. Having the distributed collector, the traffic does not have to be directed in a specific machine for capturing and it enables the collection of traffic from various sources. Since the distributed collector was created, the data transformation workers had to be modified as well, to "listen" to all collectors and gather data to be sent in Kafka for publishing in HDFS.

In addition, also to facilitate the mitigation of privacy aspects in its application, an add-on feature for IP anonymization was implemented. Due to restrictions, it may not be possible for an IP address to be visible to all users. The IP anonymizer transforms every IP of the collected data into a random IP, there is a one to one mapping between the two IP's, and the whole procedure can continue accordingly with no further interruption. Moreover, we exploited the dashboard offered from Apache Spot to visualize results and be able to observe any anomalies/threats that may occur.

Finally yet importantly, an internal trial was conducted, also in collaboration with the SHIELD project, in order to validate the current performance of Apache Spot in a pre-operational environment.

The data used for the trial were captured under real conditions from inside the SHC corporate network, naturally generated during the day-to-day operations of the company personnel (~300 employees). The traffic information (in NetFlow v.9) format was captured from the company's central firewall, which interconnects the company internal network to the Internet. The firewall was configured to send the traffic information in real time to a NetFlow collector, which in turn fed the Spot storage using the distributed collector pipeline described above. The incident detection and classification results, as well as the mitigation proposal, was visualised in the Spot dashboard.

5.2. Status and features of Release A

The current implementation status of the 5GENESIS Security Analytics platform, until M15 (Release A) is visualized in Figure 6 below. The components indicated with a solid outline have been already implemented and tested (yet several extensions are still foreseen in the frame of the project). The components indicated with a dotted outline are planned to be developed in the months to come.

© 5GENESIS Consortium

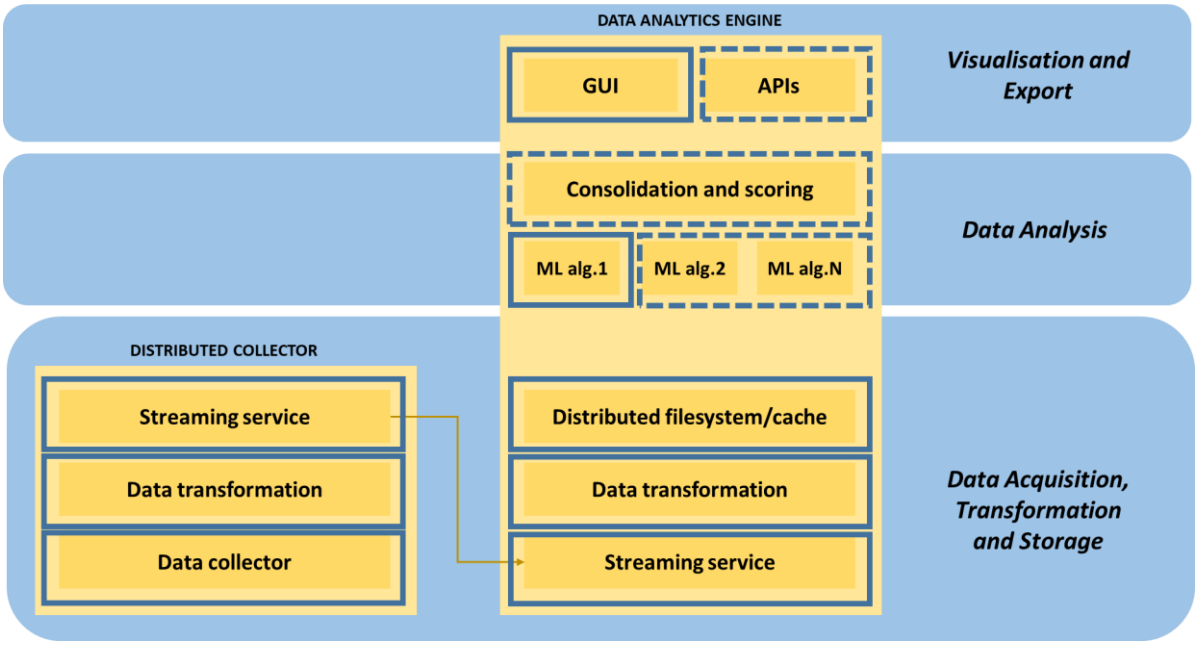

Figure 6. Status and features of Release A of the 5GENESIS Security Analytics platform

In specific:

- The ingestion chain (collection, transformation, streaming, storage) is fully functional, currently operating with NetFlow data. It will be extended in the frame of the project, also to integrate other types of information.
- The Machine Learning algorithm set currently includes the LDA algorithm, which is operational. It will be extended with more algorithms.
- The consolidation and scoring function is to be developed.
- The GUI currently uses the native Spot capabilities, to be extended in the frame of the project.
- APIs are to be developed.

5.3. Future work

The main extensions, which are planned for the next phases in 5GENESIS, are:

- Adaptation of the data acquisition and storage phase in order to integrate information from multiple sources of the 5GENESIS infrastructure, as well as multiple formats.
- Extension of the data analysis phase to include multiple ML algorithms and consolidate their output, selection and adaptation of the appropriate algorithms (currently Apache Spot only uses one algorithm at a time)
- Adaptation of the GUI components to suit the 5GENESIS visualization needs and data models.

© 5GENESIS Consortium

- Development of an API for data export. Integration with already existing security analytics platform, such as Splunk[2], to give better insights to data visualization and data handling in general.
- Deployment of the security analytics platform in (at least) the Limassol and Athens platforms.
- Evaluation in the frame of the rural connectivity UC (Limassol) and the Security-as-a-Service UC (Athens), using both real user traffic, as well as synthetic traffic replayed.

[2] https://www.splunk.com/

© 5GENESIS Consortium

6. CONCLUSIONS

Reinforcing security is crucial for the viability of next-generation 5G networks. The security analytics solution integrated in the 5GENESIS facility intends to be a significant contribution towards this direction. The 5GENESIS Security Analytics platform is established on proven technologies and has the potential to constitute a scalable and efficient solution for promptly detecting and classifying security incidents. Over the next phases of the project, the platform will be further evolved, finalized and integrated in the Facility, while its operational effectiveness will be evaluated over representative usage scenarios. The next version of this deliverable, D3.14, will present the final release of the platform and discuss/analyze its evaluation results.

© 5GENESIS Consortium

REFERENCES

[1] 5GENESIS Consortium, "D2.1 Requirements of the Facility," 2018. [Online]. Available: https://5genesis.eu/wp-content/uploads/2018/11/5GENESIS_D2.1_v1.0.pdf.

[2] 5GENESIS Consortium, "D2.2 Initial overall facility design and specifications," 2018. [Online]. Available: https://5genesis.eu/wp-content/uploads/2018/12/5GENESIS_D2.2_v1.0.pdf.

[3] 5GENESIS Consortium, "D2.3 Initial planning of tests and experimentation," [Online]. Available: https://5genesis.eu/wp-content/uploads/2018/12/5GENESIS_D2.2_v1.0.pdf.

[4] 5G Security Landscape, June 2017, 5G-PPP, https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP_White-Paper_Phase-1-Security-Landscape_June-2017.pdf

[5] SHIELD project, https://www.shield-h2020.eu/

[6] S. Omar, A. Ngadi, and H.H. Jebur, "Machine learning techniques for anomaly detection: An overview", International Journal of Computer applications, vol. 79, No 2, 2013

[7] M Umer, M. Sher, and Y. Bi, "Applying One-Class Classification Techniques to IP Flow Records for Intrusion Detection", Baltic Journal of Modern Computing, num 1, pp 70-86, http://dx.doi.org/10.22364/bjmc.2017.5.1.05

[8] H.Kim, K.Claffy, M.Fomenkov, D.Barman, M.Faloutsos, K.Lee: "Internet traffic classification demystified: myths, caveats, and the best practices." In Proc. of ACM CoNEXT, 2008 Madrid, Spain

[9] Y.Lim, H.Kim, J.Jeong, C.Kim, T.Kwon, Y.Choi: "Internet traffic classification demystified: on the sources of the discriminative power." 2010, In CoNEXT, pg. 9

[10] Apache Hadoop, https://hadoop.apache.org/

[11] Apache Kafka, https://kafka.apache.org

[12] M. Frampton, Mastering Apache Spark, Packt Publishing Ltd., 2015

[13] Apache Spot (incubating), https://spot.apache.org/

[14] D. Blei, A. Ng, M. Jordan, "Latent Dirichlet Allocation," Journal of Machine Learning Research, vol. 3, pp. 993-1022 , 2003.

© 5GENESIS Consortium