innoviris.brussels
empowering research

ULB UNIVERSITÉ LIBRE DE BRUXELLES

sopra steria

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

Scalable Machine Learning for Automated Defence Systems

Olivier Markowitch

December 17, 2015

Dimitrios Sisiaridis

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

Objectives

- to design and analyse mechanisms for automatic detection of threats and attacks, in:
  - *Managed File Transfer systems* (MFTs)
    - examples: IBM's MFT, Oracle's MFT, open-source MFTs
  - *Collaboration platforms and tools*
    - examples: MS Sharepoint, Alfresco, Huddle, OwnCloun

Objectives

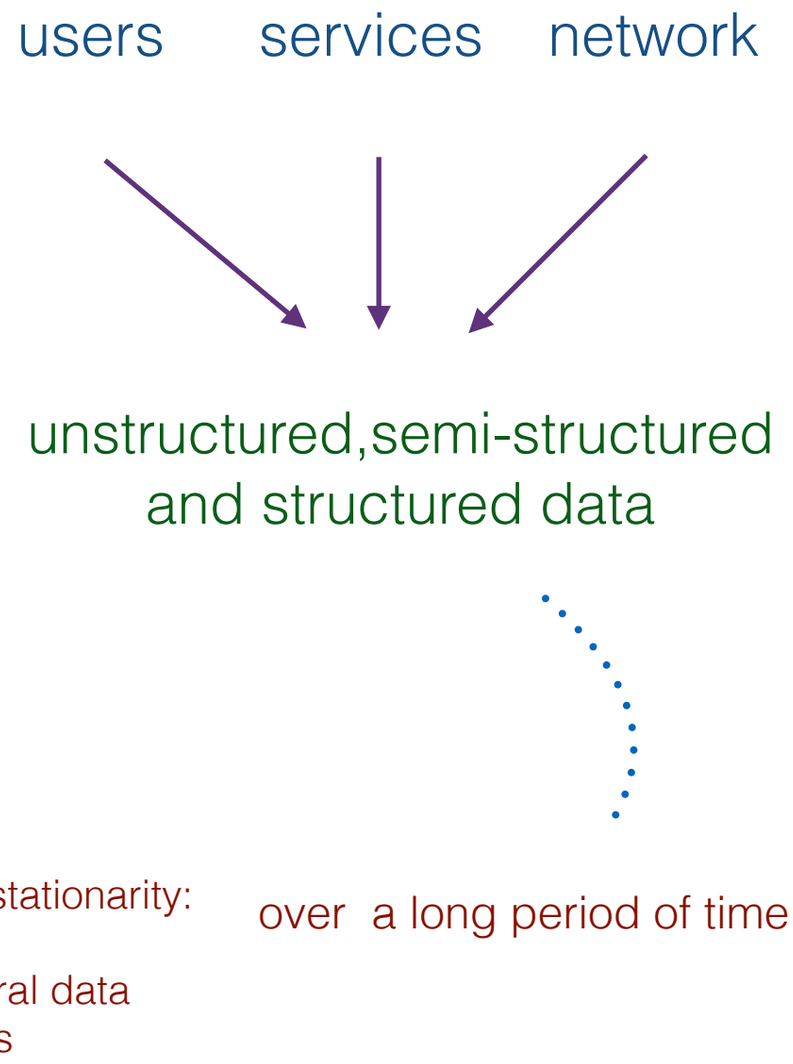
- to design efficient interfaces
  - to obtain information over intrusion kill chains
  - to report
    - automatically detected attacks
    - corresponding security measures automatically enforced by the system
- to promote centralised monitoring, processing and analysis of complex events

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

Key features

- a data-centric risk management modular approach for the detection of:
  - Advanced Persistent Threats
    - especially, for long-term targeted attacks
      - real / near-real time analysis
  - zero-day attacks
    - real / near-real time analysis
  - DDoS attacks
- allows attackers, to perform their attacks, without having access to any sensitive data:
  - in an observation area using *honeypots* and *sandboxes*
- in compliance with the *EC Guide for the identification of threats and attacks*



innoviris.brussels
empowering research

ULB UNIVERSITÉ LIBRE DE BRUXELLES

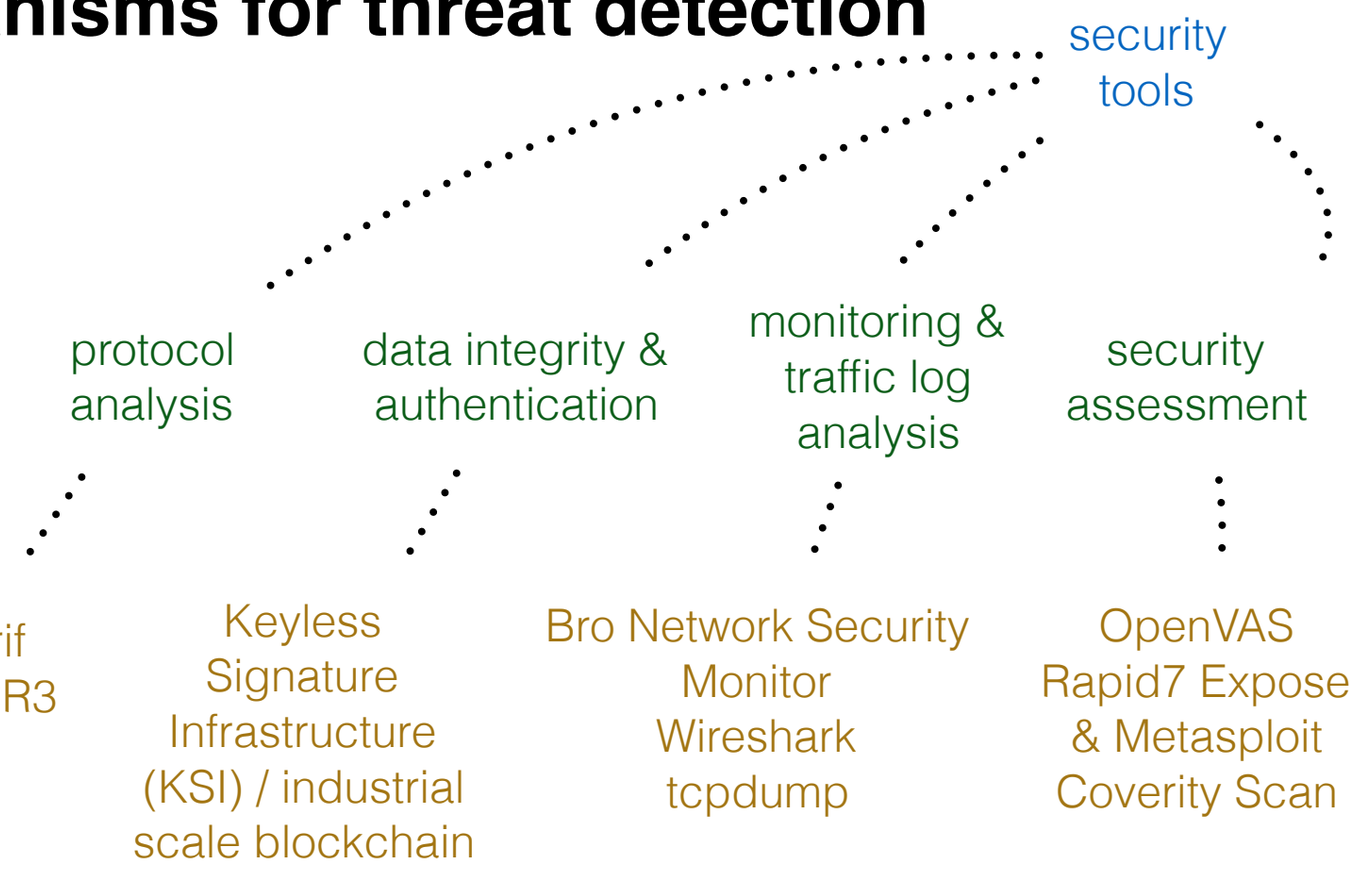
sopra steria

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

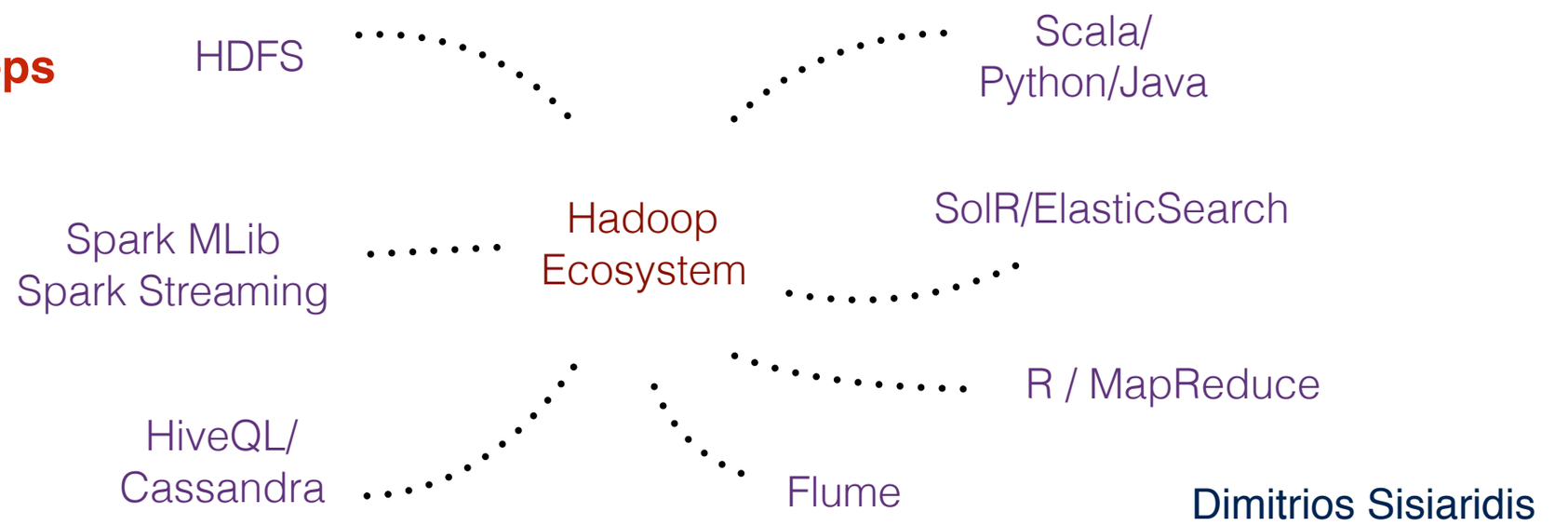
Security tools

- state-of-the-art industrial practice
  - using mainly open source software
- state-of-the-art research on the field
  - based on current findings and running projects



Big Data Security Analytics steps

- Data
- Analysis
- Security Intelligence
- Response



Olivier Markowitch

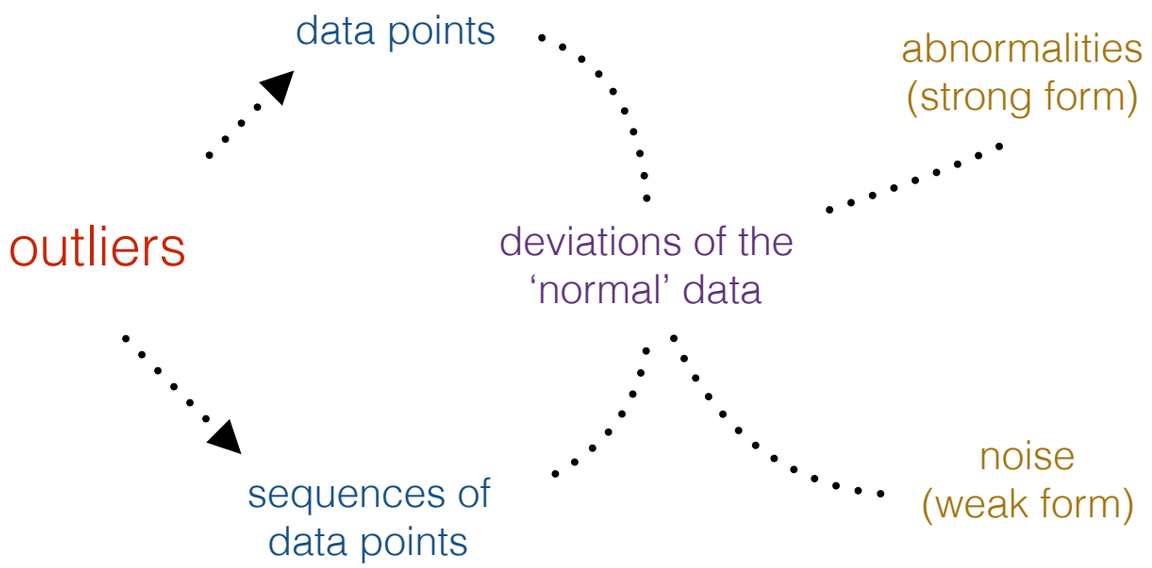
Dimitrios Sisiaridis

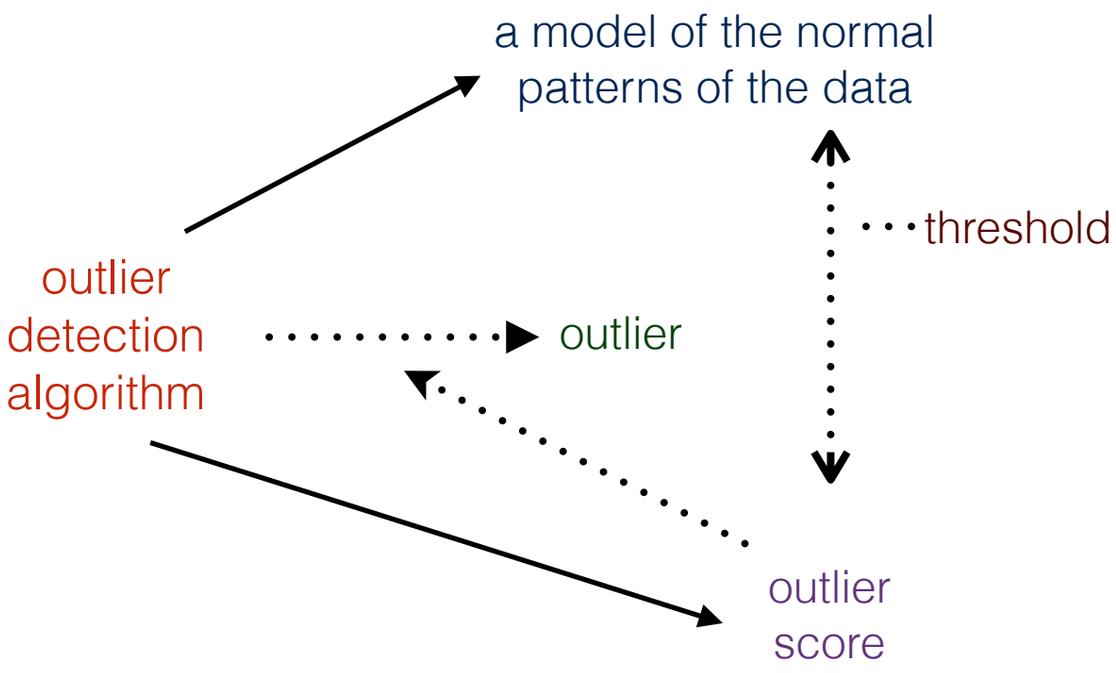
Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

Machine Learning for threat and attack detection

- based on *Outlier Detection Analysis*





Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

Discrete sequences in operation logs

- temporal data
  - categorical or time-series
- non-temporal data
  - based on their relative placement with respect to one another
- spatial data
  - where non-spatial attributes are measured at spatial locations
- spatiotemporal data
  - containing both continuous and categorical attributes

Change detection in the time-series

- slowly over time
  - known as *concept drift*
  - can be detected with a detailed analysis over long-time
- abruptly
  - sudden changes in time-series values
  - real or near-real time analysis

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

Tasks completed (months 1 - 6)

- WP2.1 : Market Intelligence
- WP2.2 : Analysis of existing techniques and products

Objectives for the next period (months 7 - 12)

- WP2.3: Security procurement
- WP2.4 : Design of analysis tools

On-going tasks

- security assessment of the basic building blocks
- Traffic log analysis
  - sensitive data anonymization
  - Datasets features extraction
    - data normalisation
    - data ingestion
      - on premises server and on a public cloud
- pattern analysis
- simulation analysis
- baseline behavioural profiles
  - including the use of honeypots and sandboxes
- data integrity constraints

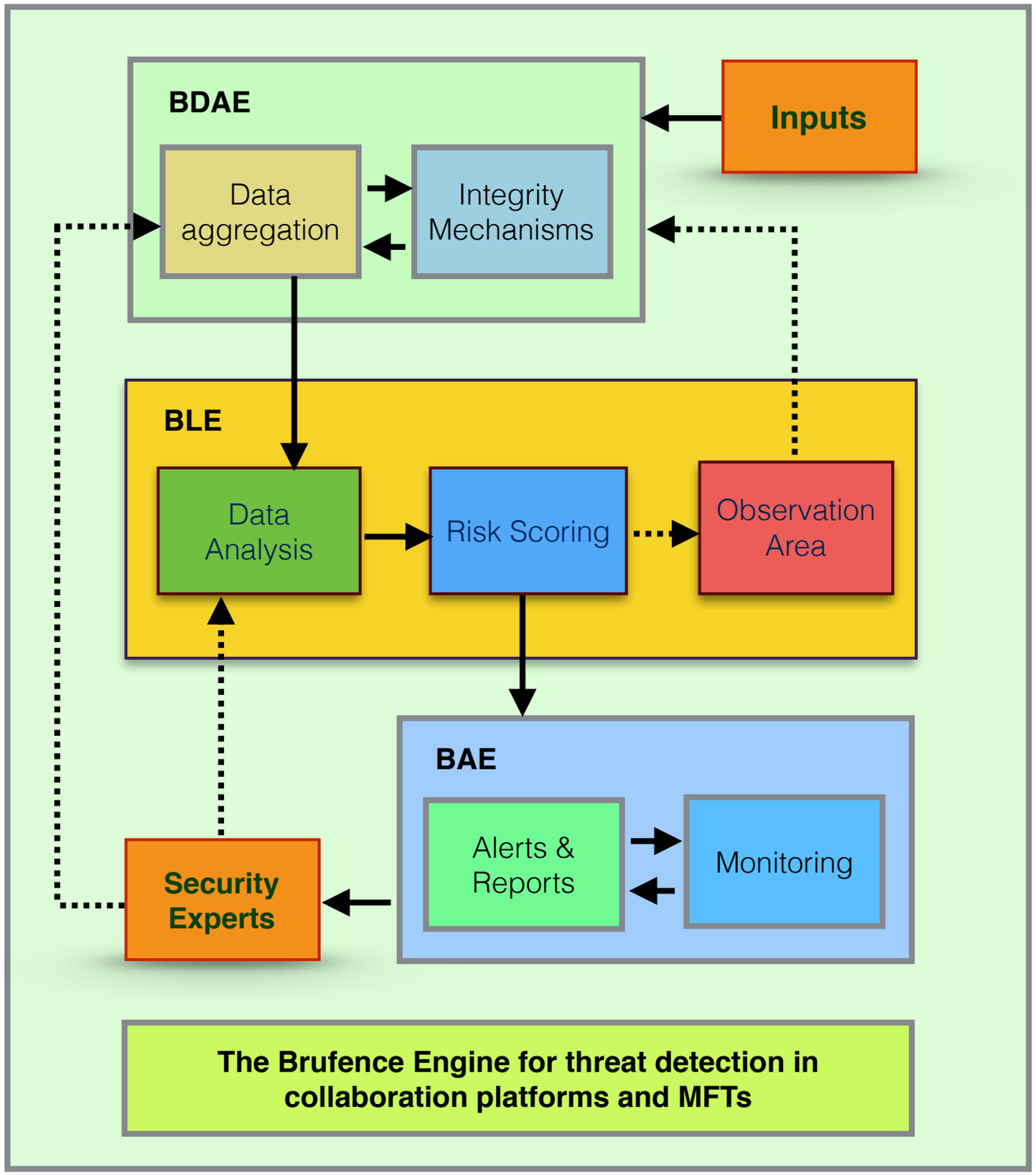
Olivier Markowitch

Dimitrios Sisiaridis

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

- **Inputs:**
  - sources (collaboration platform / MFT):
    - Authentication, Authorization and Auditing Services (AAA)
    - Network
    - Middleware
  - data format**:**
    - un-structured data (e.g. files of different formats)
    - structured data (e.g. RDB/ORDB)
    - semi-structured spatial and temporal data
- **BDAE: Data Aggregation Engine**
  - data aggregation/correlation
  - integrity mechanisms
- **BLE: Learning Engine**
  - BigData Analytics
  - Behavioural Analysis
  - Complex Event Processing
  - Risk scoring for threats/attacks
  - Observation area (Honeypot/Sandbox)
- **BAE: Alert Engine**
  - near real-time monitoring
  - reports for security experts



Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

State-of-the art methods and techniques

A. Behavioural Analytics & Pattern Matching

- ✓ A1. Behavioural analysis
- ✓ A.2 Detection of misuse of system access
- ✓ A.3 Data analytics with python scripts
- ✓ A.4 Security analytics with text mining
- ✓ A.5 Threat intelligence

B. Analysis of unstructured, semi-structured and structured data using Hadoop ecosystem

(CDH, Cloudera's open source distribution for Apache Hadoop)

- ✓ B.1 Uploading structured data onto HDFS, using Scoop
- ✓ B.2 Logs ingestion: unstructured or semi-structured data
- ✓ B.3 Data analysis with Spark
- ✓ B.4 Interactive log analysis: indexing and data ingestion using Flume

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A1. Behavioural analysis of system users

- ✓ *data aggregation* from different sources
  - ➡ complex event processing analysis
  - ➡ deployment of Indicators of Compromise (IoCs)
- ✓ *post-breach analysis*
  - ➡ thorough observation of interesting events
  - ➡ including honeypots and sandboxing
- ✓ *lateral movement analysis*
  - ➡ logging in from one account on one resource to another e.g. a user logs in as a local administrator on another computer
  - ➡ sudden changes in behaviour are highly suspicious
  - ➡ special care for pass-the-hash (PtH) attacks
    - ◉ such attacks are usually used in APTs to authenticate in a remote authentication server

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A2. Detection of misuse of system access (I): insider and outsider threats

- ✓by tracking anomalous behaviour in order to detect attacks/threats, either in:
  - ➡system-centric file transfer
  - ➡people-centric file transfer
  - ➡extreme file transfer across proprietary or enhanced protocols over UDP or in parallel over TCP

- ✓we use:
  - ➡access log files
    - ◉produced by AAA services as well as by web/application servers and DB transactional logs
    - ◉ including unfiltered data logs from all the firewalls implemented in the underlying network
  - ➡IoCs (Indicators of Compromise) as they are defined in STIX, for long-term APTs as well as for potential zero-day attacks

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A2. Detection of misuse of system access (II): insider & outsider threats

- ✓we are looking for finding potentially suspicious connections in spatiotemporal data
  - ➡by using a critical factor, the distance between locations of the connections, expressed in the form of haversine distances
    - ◉A *haversine distance* is a formula for finding the great-circle distance between a pair of latitude/longitude coordinates
      - ‣ It is a calculation of geographical distance (latitude/longitude) in terms of measuring a spherical distance
      - ‣ is used already in navigations to define a certain geographical area

- The greater the haversine distance:
  - the greater the distance between the sources of remote logins of one particular user in a given time
    - thus, the greater are the chances that this was a potentially anomalous user access

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A2. Detection of misuse of system access (III) : insider & outsider threats

- ✓In next steps, machine learning algorithms in terms of outlier detection analysis will be deployed:
  - ➡to handle more efficiently detection of threats and attacks on the platform/system
  - ➡to evaluate those cases that will need further analysis
    - ◉using honeypots and sandboxing for monitoring

- ✓ areas of interest:
  - ➡ time series analysis of temporal and spatial multidimensional data
  - ➡ streaming outlier detection
  - ➡ change detection and concept drift
    - ◉ the latter, shows the importance of binding a more active collaboration with the *Spices* project.

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A3. Data analytics with python (I) : pattern matching

Step 1:

✓logs are transformed to .csv format

✓these datasets are then imported in python scripts to be analysed for potential threats

➡We have written the needed parsers for processing logs of different format and of different structure with a variety of features

☆Still we have to work more on this, to finalise a more automated way of features parsing

☆We are also aware to employ the new EU General Data Protection Regulation (GDPR), which is expected to be adopted by early 2016

★the regulation requirements focus heavily on the individual's right to privacy

★state-of-the-art anonymization techniques ensuring privacy is one of our major concerns in our research

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A3. Data analytics with python (II) : pattern matching

Step 2:

- ✓ datasets are normalised and anonymised
- ✓ we use the ‘csv’ module, to extract the features we are interested by parsing thus the logs into 'structural events’. Then we calculate:
  - ➡ the coordinates values for a given IP address
  - ➡ the haversine distances using the MaxMind geoIP module, one of the most popular geolocation databases
- ✓ finally, we generate results, by determining a threshold for what is unusual for a certain amount of time (time series analysis)

⭐we are interested in finding a greater haversine distance in a shorter log-in frequency span

- ★ as being more suspicious

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A3. Data analytics with python (III) : pattern matching

Step 3:

- ✓ Further analysis on the logs can be done with other tools which are consuming data in the .json format or in other semistructured forms (e.g. .xml)
  - ➡ the 'json' module can take python hierarcies and convert them to string representations (data serialization)
  - ➡ The reverse process, would be to construct the data from the string representation (data deserialization)
    - ◉ avro schemas (.avsc) are implicitly defined in json format

- ✓ Logs in semi-structured and structured form can be imported to the MISP system
  - ➡ the Malware Information Sharing Platform, a H2020 funded project
    - ◉ for showing relations with other observables and indicators by creating correlations between malware, events and attributes in the case of APTs
  - ➡ outputs from MISP are in the form of Snort/Suricata IDS rules, xml/xsd for STIX IoCs, openIOC, .csv or plain text

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A4. Security analytics with text mining (I)

- ✓ Textual data, imported in a .csv format, are analyzed to find hidden relations between features of interest
  - ➡ pattern analysis on unstructured data combined with n-grams analysis
    - ◉ call center recordings as well speech used in webinars and any other form of oral data in a collaboration platform can be converted to text
    - ◉ websites can be scraped and analyzed to find trends in security-related themes, such as botnets, malware, etc.

- ✓ Text is converted to numbers, which represents word frequencies, using either:
  - ➡ a document-term matrix, where each column represents a single word that appears in any one of a collection of documents (as rows)
  - ➡ a term-document matrix, where the above table is transposed, thus, each document name becomes a column header and each row represents a different word

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A4. Security analytics with text mining (II)

- ✓ The numerical values within the body of the table represent the number of times each word appears in a given document
  - ➡ These frequency numbers can be further transformed to be used for e.g. finding differences in documents sizes, to reduce words to their root forms, or inversely to weight the frequencies by how commonly certain words appear within a given language

- ✓ then, we clean the data from:
  - ➡ extra characters that are generally meaningless for analyzing word frequencies, including extra white spaces
  - ➡ frequently occurring words in any language that are meaningless in data analysis
    - ◉ These words are collected and form a 'stop-word list', also known as 'stop-word corpus' (e.g. 'the', 'of', 'a', 'is', etc.)
    - ◉ they should be removed and not included in the word frequency table

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A4. Security analytics with text mining (III)

Data analysis steps (I):

- ✓ applying data profiling with summary statistics, to look e.g. for possible anomalies
- ✓ find commonly occurring terms, to detect e.g. anomalies in words that could affect further analysis
- ✓ find word associations, by setting a balanced correlation threshold, to detect possible attacks (e.g. SQL injections)

➡ for SDN and generally multi-layer networks as they are commonly exist in collaboration platforms and managed file transfer systems in large enterprises, a Bayesian inference can be implemented by Monte Carlo Sampling or by deterministic methods employing Gaussian approximation.

➡ Promising vendors, as well as the new trends of using machine learning in well-known managed file transfer products, is in line with this direction

- ◉ utilizing also latent variable models as HMM and mixture models for behavioural analytics

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A4. Security analytics with text mining (IV)

Data analysis steps (II):

- ✓ we are looking for trends over time, for patterns of attacks e.g. by aggregating word frequencies over a period of time (e.g. across a year/month/week/day/hour)
- ✓ we apply correlation analysis of time series trends by creating a symmetrical correlation matrix on the data (values are mirroring to each other on opposite side of the diagonal)
  - ➡ each value represents the correlation coefficient
  - ➡ the closer a value is to 1.0 the more each term is related to each other. On the other hand, the closer a value is to -1.0, the more often one term is not present
  - ➡ in cases where there are many correlations to examine, the correlation matrix is converted into a graphical image for visualisation

- ✓ then, we create a term dictionary to analyze a selected subset of terms that are relevant to the analysis to reveal anomalies such as outliers

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A4. Security analytics with text mining (V)

Clustering analysis:

- ✓ hierarchical clustering, to find clusters that maximise the differences between them
- ✓ k-means clustering for grouping data by partitioning data elements
  - ➡ in a predefined number of clusters, by hierarchical clustering
  - ➡ the sum of square distances from each data element to each cluster's centroid is minimised
    - ◉ In this way we can assign to each data element, a cluster membership value:
      - ‣ to build a predictive model to classify new threats and attacks (e.g. in new phising attacs)
      - ‣ to generalize characteristics that tend to define each cluster, with a decision tree classification analysis, (e.g. by using the randomForest algorithm
      - ‣ to classify a cluster variable by all other variables
        - each word in the document-term matrix represents a predictive variable

- ✓ Statistics then will give each word's relative importance
  - ➡ The larger the number, the more important the word in characterizing each security-related incident within a cluster

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A5. Threat intelligence (I)

the aim:

✓we need to apply predictive analytics with real-time incidents in order to define priorities based on the classification of users, assets and threat severity

an example:

✓ in the case of APTs, in order to cope with false positives:

- ➡ we are looking for end-user DNS lookups to identify possibly malicious activity of an attacker who has compromised the system
- ➡ we are searching for evidence that DNS manipulation is being used:
  - ◉ to hide the IP addresses of remote servers
  - ◉ or, as a covert channel for data exfiltration
- ➡ assumption: an attacker would have a higher DNS lookup rate when compared to the average user's DNS lookup rate

☆ a *covert channel* is a mechanism for sending and receiving data between machines without alerting any firewalls and IDS's on the network

- ◆ in a stealthy way by using packets appearing to carry ordinary information when in fact they are concealing its actual data usually by using a variety of randomized signatures

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

A5. Threat intelligence (II)

- ✓ we aim to extend the work that has been done by the *MICIE* project:
  - ➡ to develop an alerting system, able to identify in real-time and near-real time the level of possible threats and attacks induced on a collaboration platform and in managed file transfer providing them with a real risk level
  - ➡ to promote active learning in the Brufence engine

- ✓ Other interesting projects:
  - ➡ the *Spacios* project for security provision and consumption:
    - ◉ it is more focused on the Internet of Things
    - ◉ we are working on adapting its applicability on our domain of interest
  - ➡ the *Vis-Sense* project for visual analytical representations of large datasets for enhancing network security:
    - ◉ we intend to extend their findings on the identification and prediction of complex patterns of abnormal behaviour
    - ◉ from a network security domain to the backbone network infrastructure of a collaboration platform (e.g. a collection of farms in MS Sharepoint)
    - ◉ including accounting for users, services and devices by correlating information from the underlying SDN with behavioural analytics

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

B. Working with the Hadoop ecosystem

B.1 Uploading structured data onto HDFS, using Scoop

Steps:

- ✓RDBMS transactional (structured) data are uploaded to HDFS using Scoop, in the .avro format
  - ➡Scoop creates the schema files (in .avsc) in our home directory
  - ➡we copy them to Hive, using the avro representation
- ✓we then create the external tables that hold the information from the avro files, including tables' metadata
- ✓after that, using Hue, a web server with a web-based interface:
  - ➡ we execute queries in Hive as MapReduce jobs (it is done implicitly)
    - ◉ it is a rather slow process although effective
  - ➡ Impala is a Massively Parallel Processing query engine that reads the data directly from the system itself
    - ◉ faster than Hive

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

B.2 Logs ingestion into HDFS (unstructured and semi-structured data)

Steps:

- ✓ first, we build a table in Hive in order to query the data
- ✓ data are transferred into the table by invoking the Beeline JDBC client:
  - ➡ we create an external table having the features in the log file
  - ➡ logs are parsed with SERDES (serializer/deserializer) using a well-defined regular expression
  - ➡ then we copy the data from this intermediate table to a new one that does not require any special SerDes
  - ➡ With the help of a jar (hive-contrib) we overwrite this table and invalidate metadata in order to *refresh* them for real-time analytics

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

B.3 Data analysis with Spark

- ✓ we are working with Scala
- ✓ data are ingested in the .avro format, from HDFS
- ✓ features are extracted for applying analytics

➡ a more detailed report on this will follow soon

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

B.4 Exploring logs interactively (I)

Step 1: indexing data

- ✓ we create a search index
- ✓ we generate the solr_config and schema files in the cluster
  - ➡ we customize the resulted skeleton configuration to our needs, based on the format of the normalized data logs
  - ➡ we edit the schema.xml file, defining e.g. those fields that are present and searchable in our index
  - ➡ then, we upload the configuration
- ✓ finally, we create our collection

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

B.4 Exploring logs interactively (II)

Step 2: Data ingestion

- ✓ data are imported to the collection/index by using Flume and Morphines in *real-time*
  - ➡ *Flume* is a system for collecting, aggregating and moving large amounts of log data from many different sources to a centralised data source
    - ◉ we employ a *Flume agent* defined in flume.conf to load the data into the Solr index
  - ➡ *Morphlines* is a Java library for doing ETL (Extract-Transform-Load) on-the-fly
    - ◉ A *morphline* is defined in the relevant configuration file (morphline.conf) that reads records from Flume
      - ‣ it breaks them into the fields we would like to search on
      - ‣ then loads them into Solr for quering the index
  - ➡ after that, indexed data can be explored in Solr, by creating a dashboard for visualizing searches, including time series analysis

Brufence WP2: Communication Systems Security

Design of mechanisms for threat detection

Conclusions

What we have, so far:

- ✓ data analytics on log files, of raw format, or in semi-structured and structured form
  - ➡ in terms of detecting threats and attacks
- ✓ using state-of-the-art techniques for pattern matching

Question:

- Why we are interested in these particularly?

Answer:

- ❖ these techniques have been introduced in this research in order:
  - ✓ to integrate state-of-the-art techniques in traffic log analysis in terms of data analytics on the log files
    - ➡ as they are used partially in a piecemeal basis
  - ✓ to be used for evaluating the techniques that we will introduce in the next stage
    - ➡ while employing machine learning for the detection of Advanced Persistent Threats, zero-day attacks and DDoS attacks utilising Big Data analytics
    - ➡ mainly by using Spark analytics

Thank you