SOCCRATES

SOC & CSIRT Response to Attacks & Threats based on attack defense graphs Evaluation Systems

# D5.3

# Business Logic Modelling and Impact Analyser & Response Planner - Final prototype

| | |
|---|---|
| Deliverable type: | DEM |
| Contributing work packages: | WP5 Impact Quantification |
| Due date of deliverable: | 28/02/2022 |
| Submission date: | 28/02/2022 |
| Dissemination level: | PU |

| | |
|---|---|
| Responsible organisation: | IMT |
| Editor: | Christophe Kiennert (IMT) |
| Revision: | 1.0 |

| | |
|---|---|
| Abstract | This deliverable describes the final prototype of the Business Impact Analyser, Business Logic Modelling and Response Planner components. It describes their functionalities, their integration with the SOCCRATES platform, as well as the final version of the graphical interfaces. |
| Keywords: | Business Logic Modelling, Business Impact Analyser, Response Planner, Containement CoA Generator |

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 833481
Call H2020-SU-ICT-2018 • Innovation Action • Start date: September 1st, 2019

**Classification level: Public**

# Management summary

The Business Impact Analyser and the Responser Planner are the two main components of the SOCCRATES platform that aim at performing impact quantification. The Business Impact Analyser performs an analysis of the operational impact of events such as attacks or mitigation deployment, while the Response Planner evaluates the financial impact of CoAs, on top of generation specific containment CoAs to handle the case of ongoing attacks, for which the ADG analyser cannot provide short term mitigations.

This deliverable describes the final state of the development of these two components. It presents the functionalities, architecture and interfaces of the two components, with illustrations of the graphical user interface. While both components reached a version with all the expected functionalities to be integrated in the SOCCRATES platform, one extra feature for automated reconfiguration is still planned to be added to the Response Planner. Feedback from the second pilot might also lead to adjustments in the computation of the metrics used for CoA ranking.

| | |
|---|---|
| Contributor(s) | Christophe Kiennert (IMT)<br>Marwan Lazrag (IMT)<br>Paul-Henri Mignot (IMT) |

| | |
|---|---|
| Reviewer(s) | Frank Fransen (TNO) |

| | |
|---|---|
| **Security Assessment** | **See deliverables table for security assessment requirements** |
| Approval Date | 19/02/2022 |
| Remarks | None |

# TABLE OF CONTENTS


1 INTRODUCTION – RATIONALE OF THIS DOCUMENT ..... 5

1.1 The SOCCRATES project ..... 5
1.2 This deliverable ..... 6

2 BUSINESS IMPACT ANALYSER AND BUSINESS LOGIC MODELLING ..... 8

2.1 Main additions to the Business Impact Analyser ..... 8
*2.1.1 Business Impact Analyser backend ..... 8*
*2.1.2 Business Impact Analyser graphical interface ..... 8*
2.2 Component role and functionalities ..... 8
*2.2.1 Impact computation ..... 9*
*2.2.2 Asset criticality evaluation ..... 10*
2.3 Component architecture ..... 10
*2.3.1 Business Impact Analyser API and Cortex analysers ..... 11*
*2.3.2 Business Logic Modelling inputs ..... 11*
*2.3.3 Requests to the Business Impact Analyser ..... 14*
2.4 Graphical interface of the Business Impact Analyser ..... 15
*2.4.1 Graphical interface features ..... 15*
*2.4.2 Probability graph ..... 17*
*2.4.3 Temporal graph ..... 18*
*2.4.4 Business Impact Analyser functionalities ..... 18*
*2.4.5 Graph building strategy ..... 20*
2.5 Business Impact Analyser deployment ..... 21
2.6 Final remarks ..... 22

3 RESPONSE PLANNER ..... 23

3.1 Main additions to the Response Planner ..... 23
*3.1.1 Metrics computation update ..... 23*
*3.1.2 Containment CoA generation ..... 23*
*3.1.3 CACAO Playbook generation ..... 23*
3.2 Component role and functionalities ..... 23
*3.2.1 Containment CoA generation ..... 24*
*3.2.2 CoA ranking ..... 25*
3.3 Component architecture ..... 30
3.4 Graphical interface of the Response Planner ..... 31
3.5 Response Planner deployment ..... 33
3.6 Next steps ..... 33

4 REFERENCES ..... 35

5 ABBREVIATIONS ..... 36

# 1 Introduction – Rationale of this document

This section introduces the SOCCRATES project and defines the goals of this deliverable.

## 1.1 The SOCCRATES project

SOCCRATES (SOC & CSIRT Response to Attacks & Threats based on attack defence graphs Evaluation Systems) is an EU funded project under the Horizon2020 programme that has the following main challenge:

***How can SOC and CSIRT operations effectively improve their capability in detecting and managing response to complex cyber-attacks and emerging threats, in complex and continuously evolving ICT infrastructures while there is a shortage of qualified cybersecurity talent?***

The main objective of SOCCRATES is to develop and implement a security automation and decision support platform (**'the SOCCRATES platform')** that will significantly improve an organisation's capability (usually implemented by a SOC and/or CSIRT) to quickly and effectively detect and respond to new cyber threats and ongoing attacks.

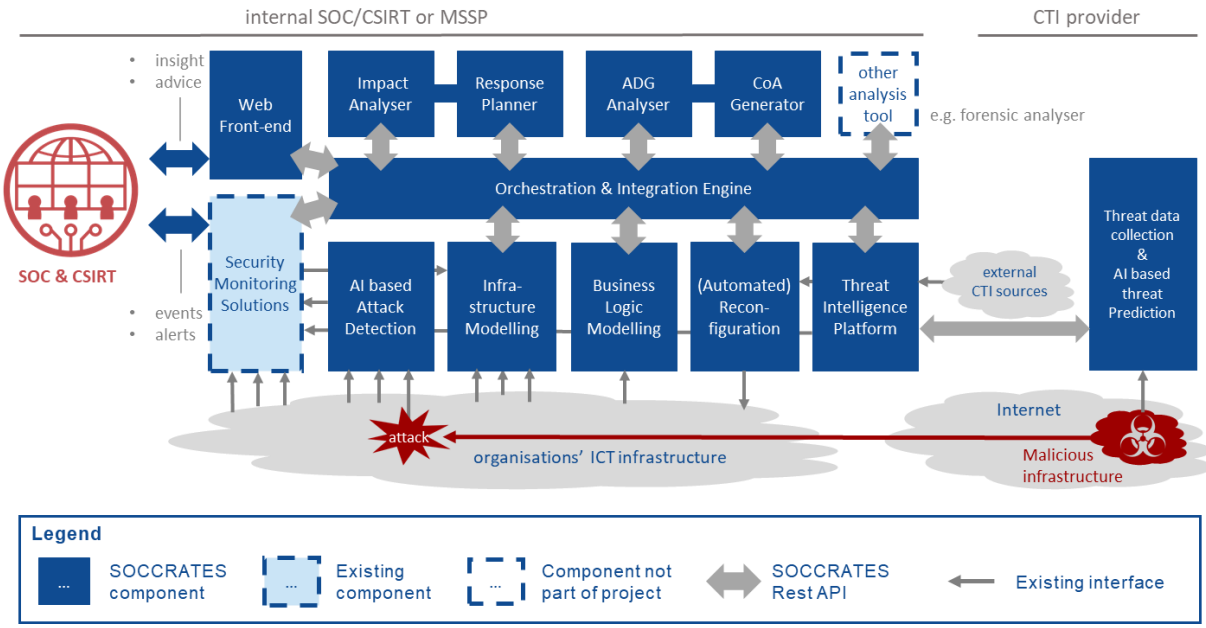


**Figure 1-1 – The SOCCRATES platform**

The SOCCRATES platform (see **Figure 1-1**) consists of an orchestrating function and a set of innovative components for automated infrastructure modelling, attack detection, cyber threat intelligence utilization, threat trend prediction, and automated analysis using attack defence graphs and business impact modelling to aid human analysis and decision making on response actions, and enable the execution of defensive actions at machine-speed.

SOCCRATES has the following concrete project objectives:

1. Deliver the SOCCRATES platform consisting of an orchestration function and a unique integration of innovative background solutions that seamlessly work together.
2. Show that the SOCCRATES platform can improve SOC operations by evaluating the SOCCRATES platform in two diverse real-life pilot environments.
3. Examine and illustrate the benefits of automation for selected SOC activities to help manage the cyber security skills gap in organizations.
4. Prepare for successful exploitation by the SOCCRATES partners of the individual innovated components and the integrated SOCCRATES platform in commercial products that are offered to the market and are available for the European (business) community.

Please visit www.soccrates.eu for more information on the SOCCRATES project.

## 1.2 This deliverable

The objective of this deliverable is to describe the final prototype of the Business Impact Analyser (and Business Logic Modelling it relies upon), as well as the Response Planner. The theoretical background of the Business Logic Modelling was already presented in Deliverable D5.1 [D5.1], and an initial prototype of these two components were described in Deliverable D5.2 [D5.2]. Since this document aims to be self-contained, it contains a substantial amount of content similar to D5.2. However, a summary of the main progress and of the main additions to D5.2 is provided at the beginning of each of the two main sections of the document, respectively focused on the Business Impact Analyser and the Response Planner.

The current prototype of these two components provides all the functionalities necessary for their integration to the SOCCRATES platform, as well as an interactive graphical interface allowing the analyst to get additional information on the outputs provided by the components. The Business Impact Analyser can compute the impact probability, the critical time, as well as the asset criticality. The graphical interface aims at making it easy for an operator to identify the most critical assets, as well as the most impacted business entities when an attack targets one or several assets. The component can successfully build a model based on the data provided by the IMC, and can successfully process the business impact requests.

The Response Planner also reached a version that can be integrated to the SOCCRATES platform. All the main expected functionalities have been implemented, with only a few improvements still planned. The Response Planner fulfills its two main functionalities: generating containment CoAs and ranking CoAs. It can compute the Return on Response Investment (RORI) of non containment CoAs, the business impact of containment CoAs, and displays the results in a graphical interface that allows the Security Operations Center (SOC) analyst to quickly identify the most efficient CoAs, based on the appropriate metric. After ranking the CoAs, the Response Planner generates a CACAO playbook for each CoA. Currently, these playbooks are only comprised of manual commands. However, in conjunction with work on the Automated Reconfiguration component, the generation of OpenC2 commands will soon be added. The evaluation of the RORI parameters was defined in accordance with the available outputs from the other SOCCRATES components, which required substantial adaptations from the original definition of the parameters. Therefore, it might be necessary to tweak the parameter evaluation based on the results obtained from the second pilot.

## 1.3 Structure of this deliverable

Section 2 of this document describes the final prototype of the Business Impact Analyser, specifying the implemented functionalities, the component architecture, and the graphical interface. The section starts with a summary of the main additions to the description provided in Deliverable D5.2. Similarly, Section 3 describes the final prototype of the Response Planner, with a structure similar to Section 2.

# 2 Business Impact Analyser and Business Logic Modelling

## 2.1 Main additions to the Business Impact Analyser

This section describes the major additions and new functionalities of the Business Impact Analyser. As described in deliverable D5.2 [D5.2], the development of the Business Impact Analyser was already at an advanced stage, and the component is now fully implemented. The major additions are focused on the BIA backend and the finalization of the graphical interface.

### 2.1.1 Business Impact Analyser backend

This subsection details the finalization of the BIA backend and the integration of the Business Impact Analyser into the SOCCRATES platform. As detailed in deliverable D5.2, the BIA needs to build a model to compute the business impact of an external event on the business entities. To build a model, the BIA relies on the Business Logic Modelling, which is comprised of two parts:

- The mission dependency model, which describes the dependencies between business entities. One model was established for each pilot site, based on the pilot site's internal organization data. This model is represented as a matrix describing the interdependency values between the business entities. Two matrixes are needed to build the model: the impact probability matrix and the downtime tolerance matrix.
- The resource dependency model, which describes the dependencies between assets: a script that fetches the list of assets every 24 hours from the Infrastructure Modeling Component was added. Each asset is defined by a host name, an IMC identifier "UUID" and an IP address. A JSON file is generated, which contains all asset information, such as asset ID, anonymised asset name, UUID, host name and IP address.

The estimation of the downtime tolerances between assets has been updated, based on the downtime tolerances from the mission dependency model (see Section 2.2.1).

The deployment of the Business Impact Analyser at the pilot sites was handled with the creation of a Docker image (see Section 2.5).

### 2.1.2 Business Impact Analyser graphical interface

New features have been added to the graphical interface (see Section 2.4):

- The critical time can be computed,
- After computaton of the critical time, the shortest path is displayed in dotted on the downtime tolerance graph,
- Possibility to display asset information (UUID and IP address).

## 2.2 Component role and functionalities

This section describes the role and the functionalities of the Business Impact analyser component. The objective of the Business Impact Analyser is to provide the SOC analyst with an evaluation of operational impact of external events on the infrastructure, such as an attack or the deployment of a mitigation. To do so, two metrics were defined, and described in Deliverable D5.1 [D5.1]: the probability of impact on the company, and the critical time.

However, the Business Impact Analyser can also help identify the most critical assets in the infrastructure, i.e. those which contribute the most to the propagation of impacts. Therefore, on top of the evaluation of the operational impact of external events, the Business Impact Analyser can also compute a criticality value for each asset.

### 2.2.1 Impact computation

The impact computation is comprised of two metrics: the impact probability and the critical time.

Regarding the impact probability, the theoretical approach is based on the inclusion-exclusion principle, which is not tractable when the size of the graph grows. Therefore, the Monte-Carlo approximation, which considerably improves the computation time thanks to a tractable time complexity, was used instead. A more detailed description of the theoretical background of impact probability computation is provided in Deliverable D5.1 [D5.1].

The Monte-Carlo approximation was implemented to randomly explore the business graph a certain number of times, *ntimes,* while counting the number of times each node has been impacted by the shock events (which are the events occurring on the infrastructure, such as attacks or mitigation deployments). The shock events are the input list of perturbations whose operational impact we want to evaluate.

These are the steps of one graph exploration:

1. Initiate a queue *Q*, with the input list of Shock Events
2. Pop a node *n*, from *Q*
3. For each edge of the node *n*, try the impact probability of the edge by drawing a random number. If the drawn number is lower than the probability of impact of the tested edge, then the node at the other end of the edge is considered as impacted. If a node is impacted and was never added to the queue *Q*, then add it to *Q*, and increase the impact counter for this new node
4. While the queue *Q* is not empty, repeat from step 2

Eventually the approximated probability of business impact is computed by taking the mean of the impact counter (Impact counter divided by *ntimes*).

The precision of the approximated probability of impact depends on the number of graph exploration (*ntimes*). For the project, *ntimes* was set to 1000, giving an approximation with a $\pm 3\%$ precision.
The Monte Carlo algorithm was implemented in Python using standard libraries, among with multiprocessing libraries to speed up the computation of the asset criticality.

The API of the Business Impact Analyser is coded in Python, using the external libraries NetworkX[1], Numpy, Uvicorn[2], FastAPI[3] and Pandas[4].

[1] https://networkx.org
[2] https://www.uvicorn.org
[3] https://github.com/tiangolo/fastapi
[4] https://pandas.pydata.org

Regarding the critical time, the Dijkstra algorithm was implemented to find the shortest path from the shock event to the business company. The Dijkstra algorithm takes three arguments: graph, start_node and target_node. Then, the following steps are executed:

1. Find the shortest weighted path between the source node : Shock event and the target node: Business Company.
2. Return the list of nodes in the shortest path between the source and the target.
3. Return the length of the shortest path between the source and the target, which is the critical time.

### 2.2.2 Asset criticality evaluation

The Business Impact Analyser component will also focus on the assets criticality, in order to identify the assets that contribute the most to the propagation of impacts on the business functions and processes.

The criticality of an asset was estimated by computing the value of the impact probability on the business company (BC) node when a shock event occurs on this asset with a local conditional probability equal to 1. Therefore, the criticality of the assets is evaluated as a number between 0 and 1.

In order to provide a convenient visualization of the assets criticality, three levels of criticality have been defined: high, medium and low, based on the computed value. Details on the graphical interface of the component are provided in Section 2.4.

## 2.3 Component architecture

The internal architecture of the Business Impact Analyser is represented in Figure 2-1. A detailed description of the various elements of this architecture was provided in Deliverable D2.2 [D2.2].

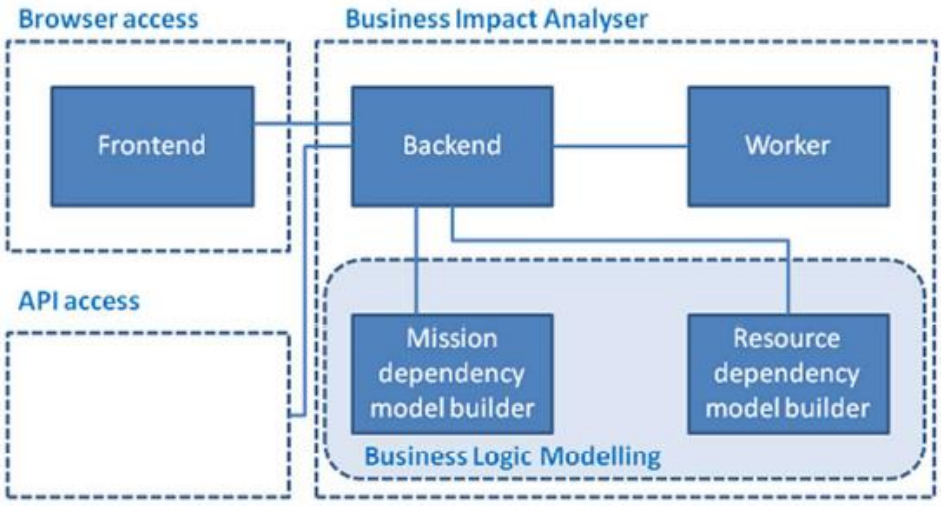


**Figure 2-1 – Architecture of the Business Impact Analyser**

Before performing any computation, the Business Impact Analyser needs to build a model. To do so, it relies on the Business Logic Modelling, which builds a mission dependency model and a resource dependency model. These two models are then fused into one graph that represents the current model.

### 2.3.1 Business Impact Analyser API and Cortex analysers

The Business Impact Analyser backend runs with a Python REST API. Two Cortex analysers and one Cortex responder have been developed for the integration in the SOCCRATES platform:

- One Cortex analyser to compute the business impact of an external event
- One Cortex analyser to retrieve the IMC identifier "UUID" of the critical assets (assets with a criticality value labeled as high).
- One Cortex responser to update the current model after the addition of a new system, specific to Use Case 5.

The details of the Cortex analysers and responder are provided in Deliverable D2.3 [D2.3]. Figure 2-2 displays the interactions of the Business Impact Analyser with the orchestrator (OIE).

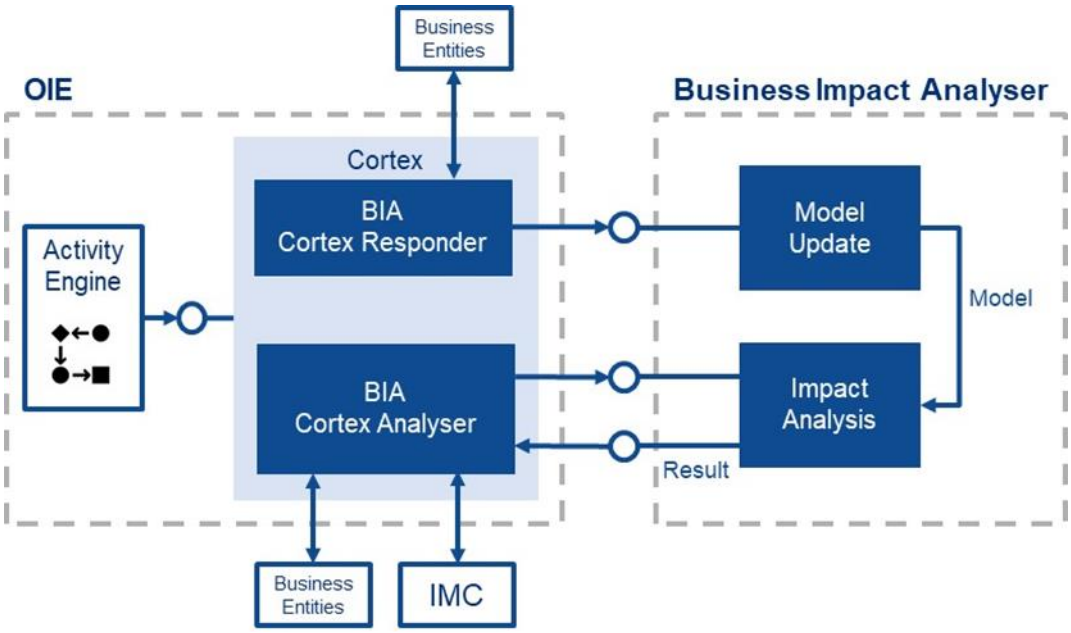


**Figure 2-2 – Business Impact Analyser Cortex interfaces**

When the BIA receives a business impact analysis request, the API will store the resulting graph inside a file, and will generate an URL. This URL will give the SOC operator the possibility to visualize and interact with the graph through the Graphical User Interface.

### 2.3.2 Business Logic Modelling inputs

This sub-section details the inputs needed to initialize and update a business logic model. As stated previously, and as detailed in Deliverable D5.1 [D5.1], a model is comprised of two parts :

- The mission dependency model, which contains the graph of the business functions and processes interdependencies
- The resource dependency model, which contains the graph of asset interdependencies

#### 2.3.2.1 Mission Dependency Model

To build the mission dependency model, the necessary input is a graph representing the company's business functions and processes along with their interdependencies. It is also necessary for this graph

to feature the assets that directly support business functions. An example of such graph is provided in Figure 2-3.

In the context of the SOCCRATES project, this graph has to be manually built, as there is no easy method to automate this, and developing one would require substantial work that would be out of the project scope. However, as advocated in Deliverable D5.1 [D5.1] this graph is not likely to change over time, or only with minor tweaks, since the business organization of a company is much more stable than its infrastructure.
The mission dependency model was established for the two pilot sites, based on the data describing the interdependencies between the company's business functions and processes and the assets that directly support those business entities.

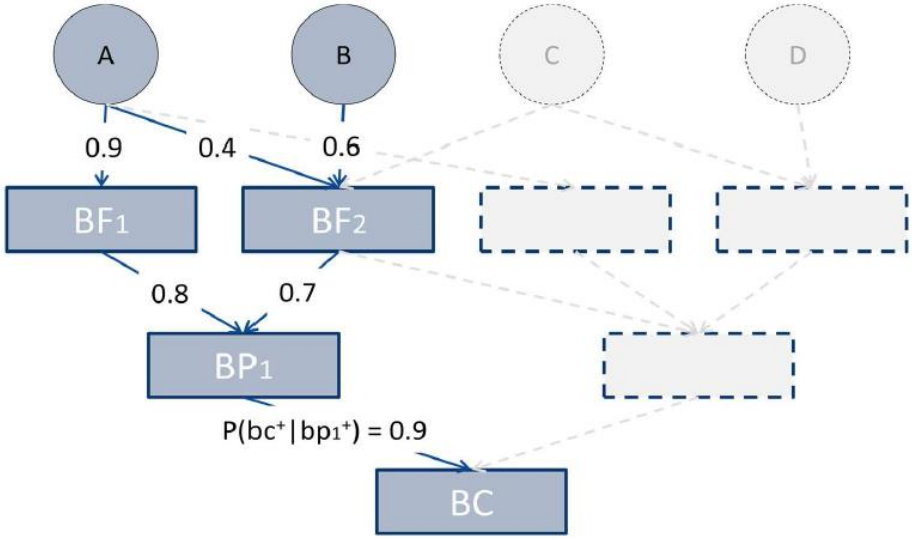


**Figure 2-3 – Mission dependency model**

To initialize the mission dependency model, the Business Logic Modelling requires such a graph, represented as a matrix describing the interdependency values, such as the one represented in Table 2-1 which contains the impact probabilities. Another matrix is required to provide the downtime tolerances between business entities.

As specified in Deliverable D5.1[D5.1], this graph has a few constraints, such as the need of a unique Business Company (BC) node acting as a sink for the graph. It must also be acyclic, in accordance with a tree structure. For the initialization phase, the matrix will be provided to the component beforehand, as a CSV file.

| Node | BF1 | BF2 | BP1 | BC |
|---|---|---|---|---|
| **A** | 0.9 | 0.4 | 0.0 | 0.0 |
| **B** | 0.0 | 0.6 | 0.0 | 0.0 |
| **BF1** | 0.0 | 0.0 | 0.8 | 0.0 |
| **BF2** | 0.0 | 0.0 | 0.7 | 0.0 |
| **BP1** | 0.0 | 0.0 | 0.0 | 0.9 |
| **BC** | 0.0 | 0.0 | 0.0 | 0.0 |

**Table 2-1 – Business Entity probability matrix**

In the context of the SOCCRATES project, the mission dependency model may only be updated during Use Case 5, where a new system is added to the infrastructure. However, if the newly added system does not bring any change to the business entities, then only the resource dependency model will be updated.

#### 2.3.2.2 Resource Dependency Model

The notion of dependency between assets is defined in relation with the quantity of data exchanged between assets, as described in more details in Deliverable D5.1 [D5.1]. The list of assets and the amount of exchanged data will be retrieved every 24 hours with a request to the Infrastructure Modeling Component (IMC) , where Netflow data will be gathered. Every 24 hours, the Business Impact Analyser fetches from the IMC the list of assets and all information related to these assets, such as the IMC identifier "UUID", the host name and IP address. The traffic matrix deduced from the Netflow data will be transformed into a probability matrix, from which the graph can be deduced.

Table 2-2 presents an example of probability matrix deduced from a traffic matrix, in such a way that the sum on each column is equal to 1.

| Node | A | B | C | D |
|---|---|---|---|---|
| **A** | 0.0 | 0.6 | 0.1 | 0.2 |
| **B** | 0.1 | 0.0 | 0.0 | 0.3 |
| **C** | 0.2 | 0.0 | 0.0 | 0.5 |
| **D** | 0.7 | 0.4 | 0.9 | 0.0 |

**Table 2-2 – Inter-assets probability matrix**

Once a mission dependency model and a resource dependency model are built, the matrixes are fused into a single adjacency matrix, which corresponds to the current business logic model. Table 2-3 presents the adjacency matrix built from the matrixes from Table 2-1 and Table 2-2.

| Node | A | B | C | D | BF1 | BF2 | BP1 | BC |
|---|---|---|---|---|---|---|---|---|
| **A** | 0.0 | 0.6 | 0.1 | 0.2 | 0.9 | 0.4 | 0.0 | 0.0 |
| **B** | 0.1 | 0.0 | 0.0 | 0.3 | 0.0 | 0.6 | 0.0 | 0.0 |
| **C** | 0.2 | 0.0 | 0.0 | 0.5 | 0.0 | 0.0 | 0.0 | 0.0 |
| **D** | 0.7 | 0.4 | 0.9 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **BF1** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.8 | 0.0 |
| **BF2** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.7 | 0.0 |
| **BP1** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.9 |
| **BC** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

**Table 2-3 – Adjacency matrix with conditional impact probabilities**

The Adjacency matrix with downtime tolerance is built from the Business Entity downtime tolerance matrix and the Inter-asset downtime tolerance matrix.
The impact propagation delay between assets is assumed to be much lower than the propagation delay between business entities, without being completely negligible. Therefore, the inter-asset downtime tolerances are set to 10% of the minimum interdependency value between business entities.

| Node | A | B | C | D | BF1 | BF2 | BP1 | BC |
|---|---|---|---|---|---|---|---|---|
| **A** | 0.0 | 0.1 | 0.1 | 0.1 | 2.5 | 1.0 | 0.0 | 0.0 |
| **B** | 0.1 | 0.0 | 0.0 | 0.1 | 0.0 | 4.0 | 0.0 | 0.0 |
| **C** | 0.1 | 0.0 | 0.0 | 0.1 | 0.0 | 0.0 | 0.0 | 0.0 |
| **D** | 0.1 | 0.1 | 0.1 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **BF1** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.0 | 0.0 |
| **BF2** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.4 | 0.0 |
| **BP1** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.2 |
| **BC** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

**Table 2-4 – Adjacency matrix with downtime tolerance**

Table 2-2 presents the adjacency matrix with downtime tolerance. The inter-asset downtime tolerances are calculated from the Business Entity downtime tolerance matrix. In this example, the inter-asset downtime tolerances are equal to 0.1 (10% x 1.0).

### 2.3.3 Requests to the Business Impact Analyser

After building a current model, comprised of a Mission Dependency model and a Resource Dependency model, the Business Impact Analyser can receive requests to:

- compute the business impact of an external event
- evaluate the asset criticality

The first type of request requires the following inputs:

- the node(s) impacted by the event
- the level of compromise in confidentiality, integrity and availability, evaluated as real numbers between 0 and 1

Figure 2-4 displays a shock event impacting node B of the current model, with a probability of 0.3 deduced from the level of compromise. The computation for the business impact will therefore trigger the calculation of the impact probability on the BC node. Similarly, a shortest path algorithm will be applied on the downtime tolerance graph to provide the critical time.

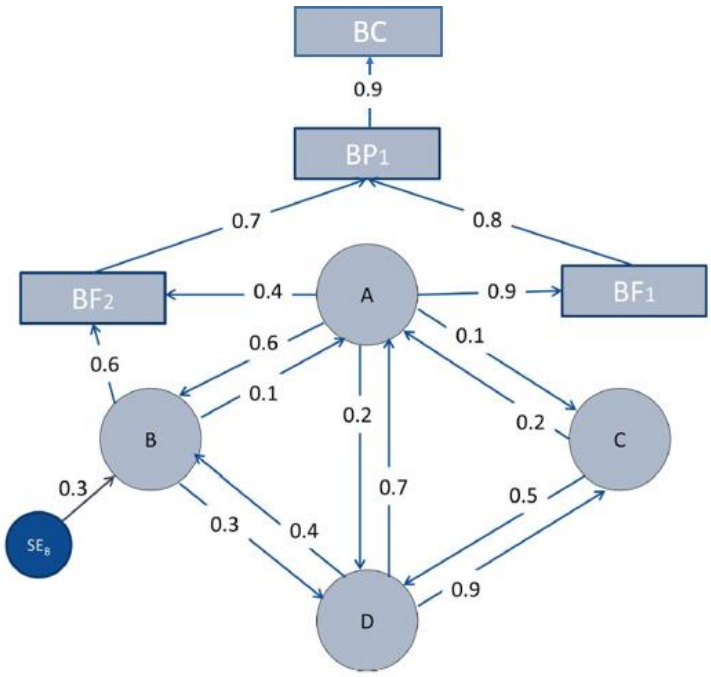


**Figure 2-4 – Business logic model with conditional impact probabilities**

The second type of request does not require any additional input. To evaluate the criticality of a node, an impact probability request is sent while applying a shock event on that node with an impact probability equal to 1.

## 2.4 Graphical interface of the Business Impact Analyser

This section describes the functionalities of the Business Impact Analyser component with its graphical interface. In particular, it displays the various functionalities of the component:

- The probability graph
- The temporal graph
- The computation of assets criticality
- The computation of the impact probability of a shock event on the business functions and processes
- The computation of the critical time
- The computation of the shortest path from a shock event to the business company

### 2.4.1 Graphical interface features

The Business Impact Analyzer Frontend is a Web Graphical User interface implemented in Javascript, running on a Python web server. Illustrations of this interface are presented in Sections 2.4.2 onwards.

The graphical interface of the Business Impact Analyser is running on a server to provide two functionalities :

- The calculation of the impact on the business function and process : impact probability and critical time
- The computation of the criticality of assets

Several buttons have been developed in the graphical interface to help the SOC analyst to assess the situation and identify the most vital assets and business processes:

- Graph type: probability graph or temporal graph
- Get impact button: compute the impact probability on the business functions and processes
- Get criticality button: compute the assets criticality
- Get critical time button : compute the critical time

#### 2.4.1.1 Interactions with the graph

A node can be added with *Shift + Click* on an empty space. Then it is possible to select the type of this new node between Asset, Business Function and Shock Event.

An edge between two nodes can be added by *Shift + Click* on a first node then hold and drag to a second node. The direction of the edge created is from the first node to the second one. Then it is possible to enter the forward and backward impact probability between the first node and the second node.

A node or an edge can be edited by *Ctrl + Click* on it. A node or an edge can be deleted by selecting it and using *Delete* key.

Asset information (UUID and IP address) can be displayed with *Ctrl + Click* on the corresponding node.

**Figure 2-5 – Node Information**

It is possible to import or export a graph in JSON format by clicking on the corresponding icons at the bottom of the screen.

#### 2.4.1.2 Sources

The implementation of the Business Impact Analyser (BIA) Frontend relies on several resources, coded in JavaScript.

It is mainly based on the *Mindcraft* code from edipa [https://github.com/eldipa/Mindcraft].

It also makes use of the following projects:

- d3: [https://d3js.org/d3.v3.min.js], Version: 3.5.17, BSD 3-clause "New" or "Revised" License
- FileSaver: http://purl.eligrey.com/github/FileSaver.js], Version: 1.3.2, MIT License
- Javascript-Undo-Manager: [https://github.com/ArthurClemens/Javascript-Undo-Manager], Version: 1.0.5., MIT License
- JavaScript Canvas to Blob: [https://github.com/blueimp/JavaScript-Canvas-to-Blob/blob/master/js/canvas-to-blob.js],Version: 3.2.0, License: MIT License
- AlertifyJS: [https://github.com/MohammadYounes/AlertifyJS], Version: 1.11.0,GNU GENERAL PUBLIC LICENSE Version 3
- Superformula Tweening: [https://bl.ocks.org/mbostock/1020902], GNU GENERAL PUBLIC LICENSE Version 3

### 2.4.2 Probability graph

Figure 2-6 displays the assets, the business functions and processes and the business company, as well as the impact probability dependencies between these nodes. The represented graph is the same as the example provided in Figure 2-4.

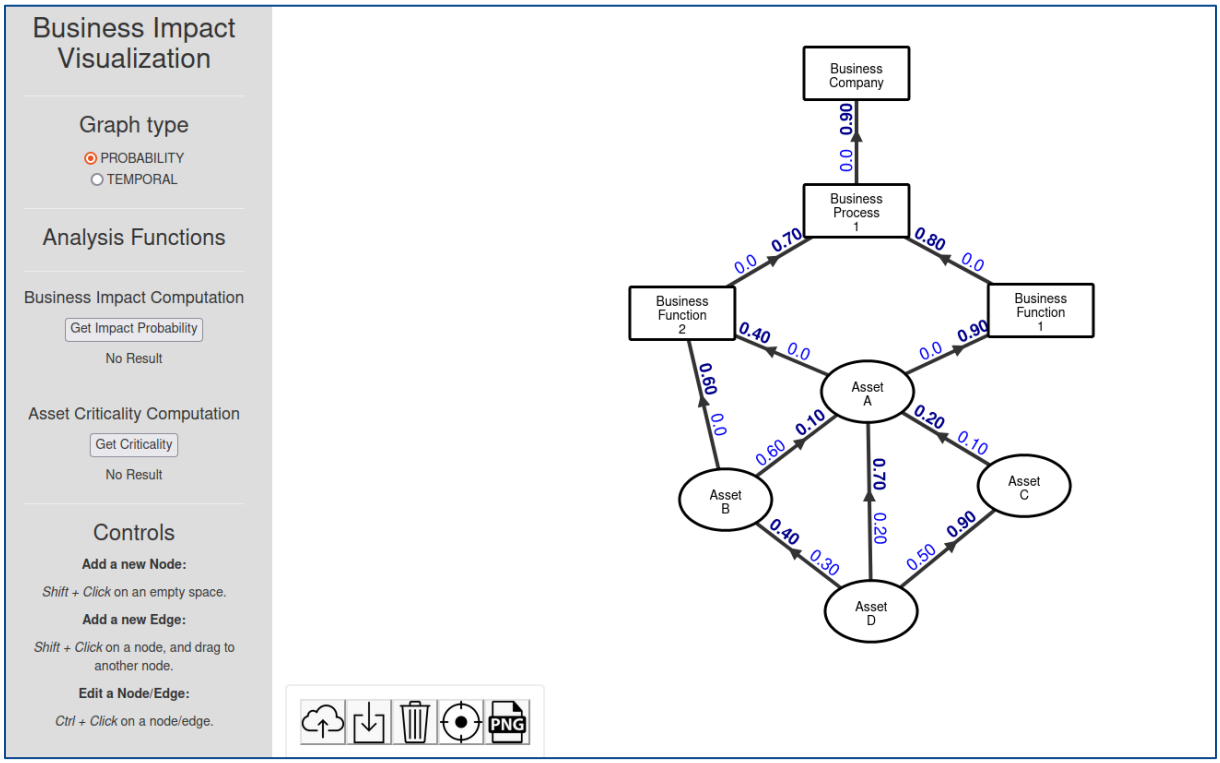


**Figure 2-6 – Probability graph**

In order to lighten the visual density of the graph, bidirectionnal impacts are represented with a single directed arc featuring two values: one colored in dark blue, and one colored in light blue. The dark blue value corresponds to the direction featured by the arrow on the arc, while the light blue value corresponds to the backward direction. For example, in Figure 2-6, the impact probability from node D to node B is 0.4, while the impact probability from node B to node D is 0.3. In accordance with the constraints specific to the mission dependency model, this graph contains no backward dependency on the arcs between business entities.

### 2.4.3 Temporal graph

Figure 2- presents the same graph but displaying the downtime tolerances instead of the conditional probabilities. The representation rules of the dependencies described in the previous section apply in this graph as well.

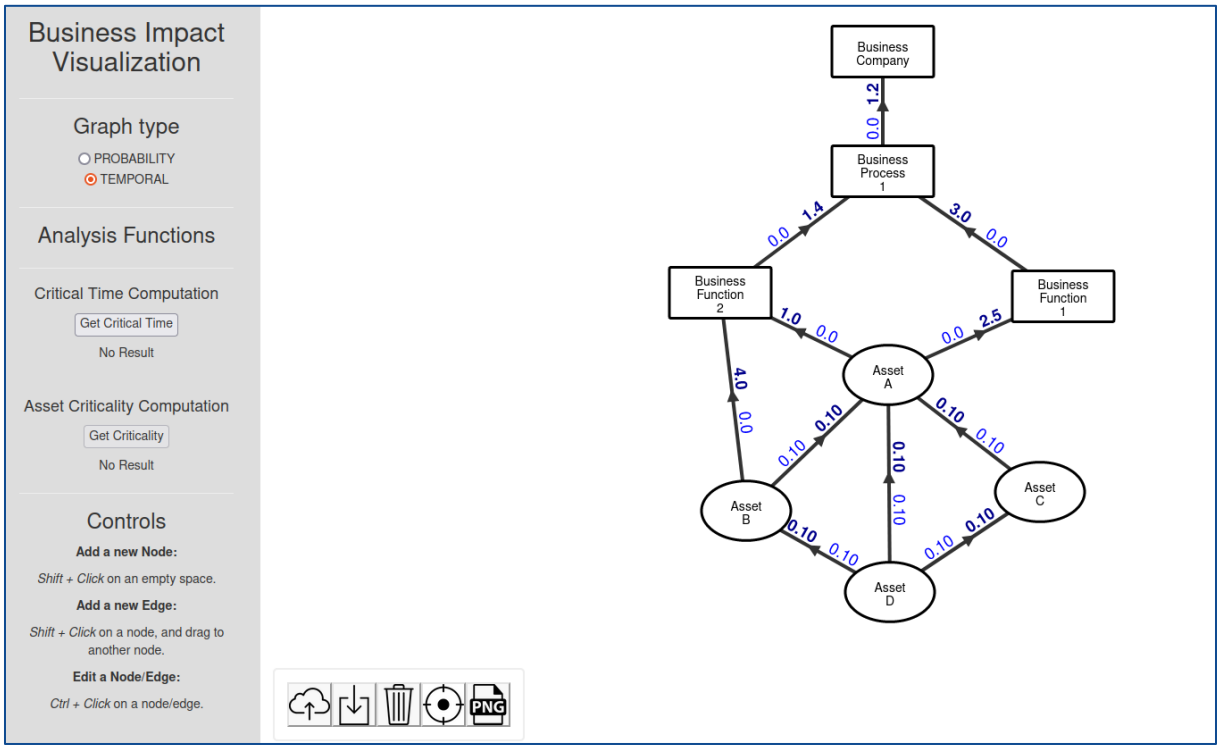


**Figure 2-7 – Temporal graph**

### 2.4.4 Business Impact Analyser functionalities

#### 2.4.4.1 Asset criticality computation

The BIA is able to calculate the criticality of assets, which returns the assets that contribute the most to propagate the impact on the most important business function (business company node).

The computed criticality is displayed in orange above each Asset node. Then, depending on the value of the impact, the node is colored:

- in green, if the value is between 0 and 1/3, because it is considered as a low critical asset
- in orange, if the value is between 1/3 and 2/3, it is considered as a medium critical asset
- in red, if the value is between 2/3 and 1, it is considered as a highly critical asset.

The value ranges for coloring can be freely configured.

Figure 2-8 displays a screenshot of the graphical interface after the computation of the asset criticality values.

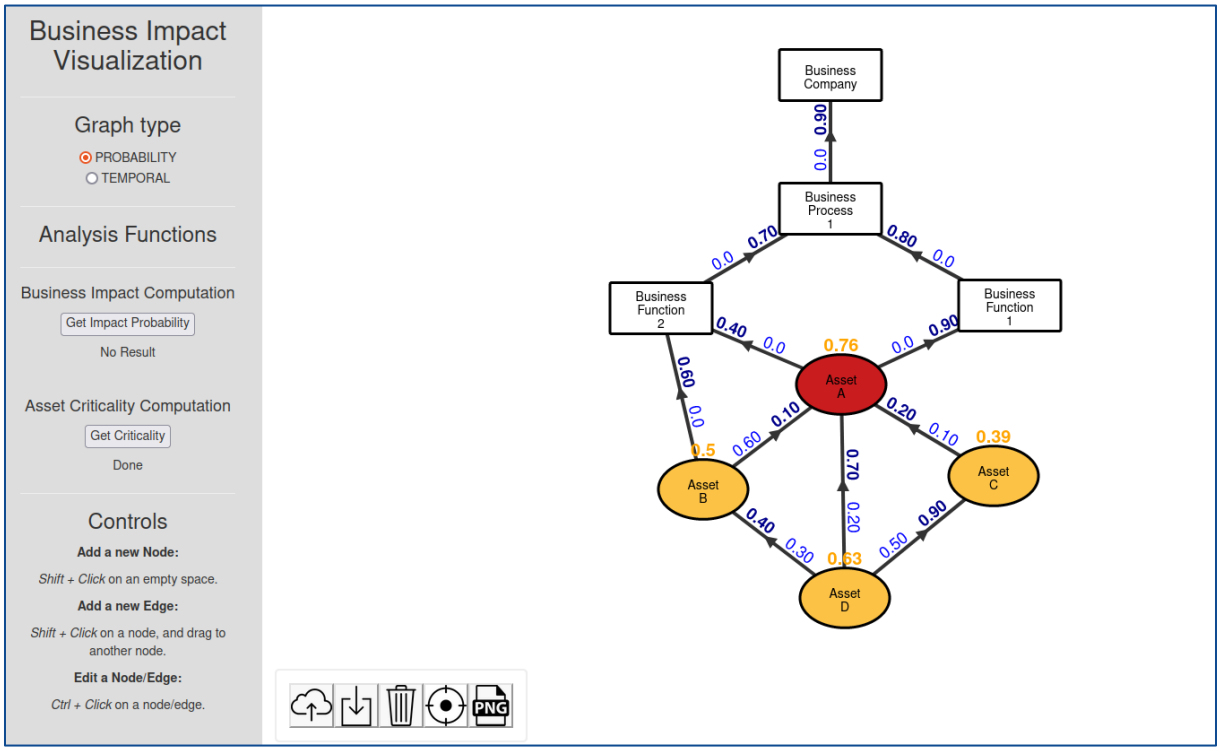


**Figure 2-8 – Asset criticality computation**

### 2.4.4.2 Impact probability computation

The Monte-Carlo approximation was implemented to compute the impact probability of a shock event on the business functions and processes.

The computed business impact is displayed in red above each Business Function node. Then, depending on the value of the impact, the node will be colored following the same rule as the one used for the asset criticality computation.

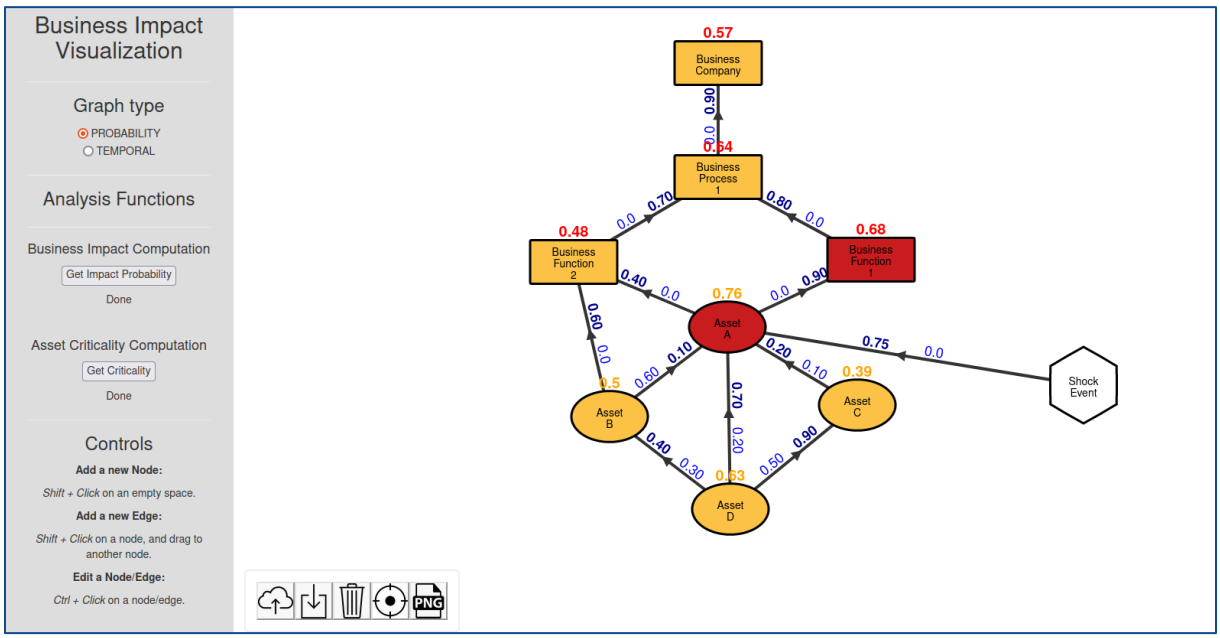


**Figure 2-9 – Impact probability computation**

In the example of Figure 2-9, the shock event is presented as follows : (Shock Event, Asset B, 0.8). Two levels of impact of this shock event can be observed on the business functions and processes and the business company. A high impact on Business Function 1 and a moderate impact on Business Function 2, Business Process 1 and Business Company.

### 2.4.4.3 Critical time computation

The computation of the shortest path from the shock event to the business company was implemented using the Dijkstra algorithm. It also provides the critical time, which is the sum of the downtime tolerances between the nodes in the shortest path.

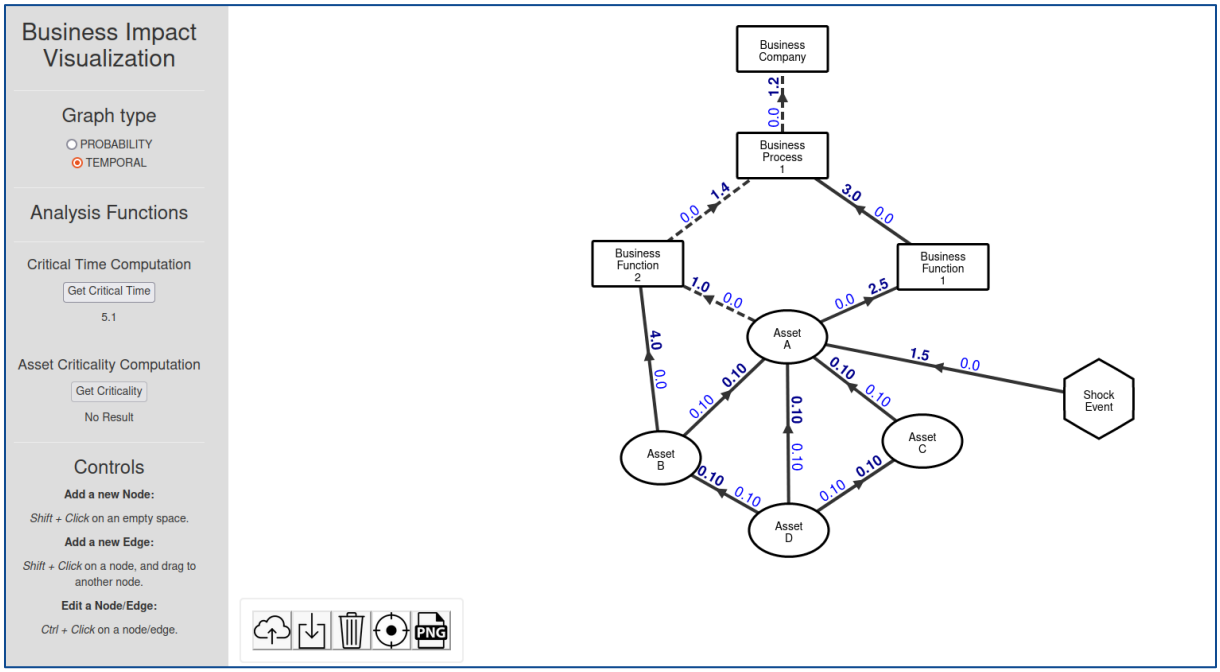


**Figure 2-10 – Critical time computation**

The shortest path is displayed on the temporal graph as a dotted line.
In this example, the critical time is equal to 5.1 hours, and the shortest path is :
Shock Event → Asset A → Business Function 2 → Business Process 1 → Business Company

## 2.4.5 Graph building strategy

In the graphical user interface, a graph can be built from the adjacency matrix constructed beforehand by the BIA backend. However, the generated graph can be quite difficult to read in the case of large graphs. Although the user interface makes it possible to drag and drop the nodes to improve readability, it was deemed necessary to think of a strategy so that the generated graph naturally highlights the important information, i.e. the business entities.

Therefore, the BIA displays the nodes with a hierarchy, starting with the Business Company node on the top, and then successively display business function nodes and assets in several circular arcs.
When adding an edge between two nodes from the adjacency matrix, the direction of the edge is chosen according to the highest impact probabilty between the two nodes, in case there is bidirectional impact. For example, considering two assets A and B with the probability of impact from A to B

being 0.2 and the probability of impact from B to A being 0.8, then the edge will be drawn with an arrow from B to A, with the value 0.8 in bold dark blue, and the value 0.2 in light blue above the edge.

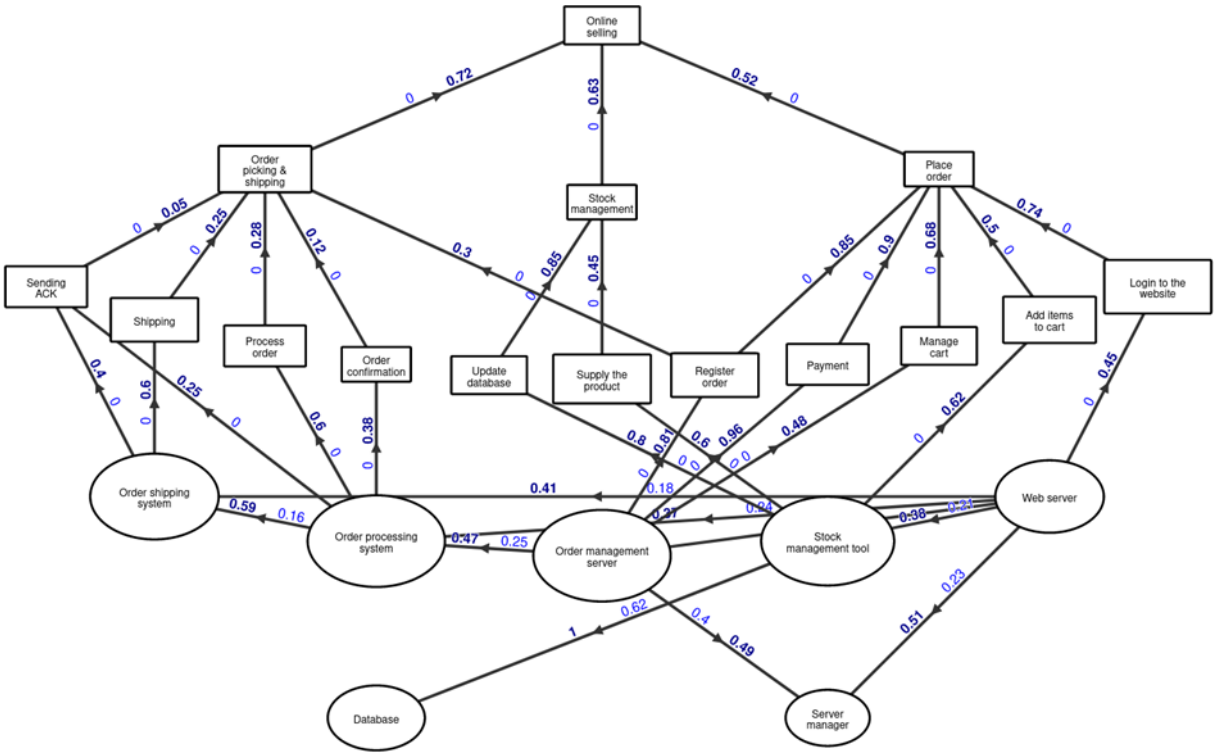


**Figure 2-11 – Graph generated from an adjacency matrix**

The graph must have a Business Company node, which is a Business entity with no outgoing edge. In other words, it is impacted by other nodes, but does not impact any node. Without a Business Company node, the GUI will not be able to compute the criticality values.

Figure 2-11 displays an example of generated graph. As shown in this figure, the hierarchical structure of the business functions, business processes and business company can be easily understood even without any manual interaction with the nodes.

## 2.5 Business Impact Analyser deployment

The deployment of the Business Impact Analyser relies on a docker image that runs the API and the graphical interface of the component.

These steps are necessary to build the image of the Business Impact Analyser on a pilot site:

1. Prepare the impact probability matrix and the downtime tolerance matrix, and put them in a specific directory
2. Edit a JSON file that makes the association between the assets that support the business entities (as they appear in the mission dependency model) and their UUID values
3. For each pilot site, configure the requests to the Infrastructure Modelling Component to retrieve the necessary data to build the model.
4. Set up a periodic update the model. By default, the model isupdated every day at midnight
5. Build the docker image and deploy the component

## 2.6 Final remarks

The development of the Business Impact Analyser is fully completed and the component is ready to be integrated into the SOCCRATES platform. The scalability of the computations has been tested on reasonably large graphs, leading to very satistying results. At this point, further improvements and optimizations might be decided based on the results of the second pilot. In particular, the Netflow data, which are meant to be used to compute the resource dependency model, appear to be extremely large. The feasability of the real-time parsing of these data has yet to be verified, and will be reported in Deliverable D5.4.

# 3 Response Planner

## 3.1 Main additions to the Response Planner

This section aims at presenting a summary of the new features and functionalities of the Response Planner, compared to the description provided in Deliverable D5.2 [D5.2].

### 3.1.1 Metrics computation update

While the initial plan was to evaluate every CoA with both the RORI metric and the Business Impact metric, the containment CoAs will be evaluated with only the Business Impact, since the financial impact of short-term CoAs has no relevancy. Likewise, the other CoAs will only be evaluated with the RORI metric. On top of being the most relevant metric for such CoAs, many of the defenses (such as Supply Chain Auditing or Security Awareness) are not compatible with the Business Logic Modelling developed in this project.

Regarding the RORI metric, all the parameters are now evaluated, as detailed in Section 3.2.2.1. New computation methods for the Annual Loss Expectancy and Risk Mitigation have been proposed based on the SIEM label, initial TTC, as well as the criticality of high-value assets.

Regarding the Business Impact of Containment CoAs, a new evaluation approach was designed and implemented through an interaction with the BIA, as detailed in Section 3.2.2.2

### 3.1.2 Containment CoA generation

The RP can now generate Containment CoAs based on the SIEM label, according to the strategy described in Section 3.2.1. In the context of the SOCCRATES project, two defenses were considered: host isolation, and traffic filtering.

### 3.1.3 CACAO Playbook generation

The RP can now generate CACAO playbooks for each of the evaluated CoA, following the model presented in Section 3.2.2.3. Currently, a manual command is describing each defense, but OpenC2 command generation will be added afterwards.

The graphical interface has been consequently updated, and can display the CACAO playbook for each CoA in the table. The CoA evaluation table is no longer editable, in order to avoid confusion on the analyst side while reviewing the CoAs.

## 3.2 Component role and functionalities

The Response Planner (RP) provides two main functionalities: containment CoA generation and CoA ranking.

The first functionality, containment CoA generation, is only used in the context of Use Case 1 (UC1): Response on Detected Ongoing Attack, in the *Containment - Preliminary response* and the *Containment – Final containment* steps. The aim of this functionality is to suggest a list of containment strategies, i.e. Courses of Actions (CoAs) to stop the progress of the current attack.

In all the Use Cases, the second functionality, CoA ranking, is called in order to evaluate the CoAs or Containment CoAs, with one different metric for each CoA type. The first metric, *RORI,* is a financial impact quantification and is used to evaluate the benefits of CoA deployment. The second metric, *BI*, is an operational impact quantification and can evaluate the short term impact of Containment CoAs. In addition, the RP will generate a CACAO Playbook for each CoA. The CoA metric will help the analyst to select the appropriate CoAs, and the playbooks will facilitate CoA deployment.

### 3.2.1 Containment CoA generation

The scenario of UC1 describes an ongoing attack on the company. In order to immediately stop the attack progression in the infrastructure, short-term CoAs meant for attack containment need to be generated.

#### 3.2.1.1 Strategy for containment CoA generation

In the context of the SOCCRATES project, the defenses of a Containment CoA can either be *Host Isolation* or *Traffic Filtering*. To generate Containment CoAs, the RP receives as input a SIEM event, one or more source IP addresses and one or more destination IP addresses. The SIEM event and the IP addresses will be provided by the REASENS module or by the pilot site SIEM. The analyst will have to validate the SIEM event and can manually add additional IP addresses to be contained. The range of SIEM events has been restrained to three labels that the RP is able to process.

The first label is *Initial Access.* This SIEM event means that an attacker managed to access one or more hosts, which will be labeled as compromised. The event will include the IP address of the compromised host. Depending on the attack technique detected, the compromised host could be given as a source or destination IP address. The RP will treat an internal IP address, in the source or destination IP address field, as a compromised host, and will generate a Containment CoA to isolate it.

The second label is *Lateral Movement.* This SIEM event alerts that suspicious communications between a compromised host and other hosts have been detected as lateral movement of the attacker, meaning that the attack is spreading to other hosts. The event will include one or more source IP addresses and one or more destination IP addresses. In the context of Containment CoAs, we assume that all hosts with reported sources and destination IP addresses are compromised, and should be contained. The generated Containment CoA will isolate the list of compromised host.

The third label is *Exfiltration.* This event means that one internal host is exfiltrating data outside of the company. The event will include one source IP address and one or more external destination IP addresses. The generated Containment CoAs will aim to block communication from the source IP address to destination IP address(es) and to isolate host with the source IP address.

#### 3.2.1.2 Isolation perimeter

Depending on the situation, the analyst may prefer to deploy different containment strategies. In that way, the RP should produce several Containment CoAs corresponding to these strategies. Two strategies have been established based on a perimeter of isolation. The first one, the default one, is to isolate all compromised hosts. The other one is more preventive and aims to isolate all compromised hosts plus neighbouring hosts, acting on a bigger perimeter. However, in the context of UC1, only the first isolation perimeter was implemented.

### 3.2.2 CoA ranking

The second functionality of the Response Planner is to generate an evaluation of the CoAs, in order to rank, compare and select the most suitable CoAs to deploy. Depending on the type of CoAs, the RP computes the appropriate metric. For non containment CoAs, we are reasoning on a long-term vision, thus a financial impact quantification metric is more appropriate. On the other hand, containment CoAs are expected to be deployed temporarily as short-term countermeasures. Therefore, an operational impact quantification metric is more relevant for such CoAs.

#### 3.2.2.1 RORI: financial impact quantification metric

The financial impact of a CoA is quantified by the Return on Response Investment (RORI) metric. The RORI metric evaluates the financial balance between the costs and the benefits of deploying a CoA, according to the size of the company.

This metric is based on four parameters: Annual Loss Expectancy (ALE), Risk Mitigation (RM), Annual Infrastructure Value (AIV), Annual Response Cost (ARC), and is computed as follows:

$$RORI = \frac{ALE \times RM - ARC}{ARC + AIV} \times 100$$

The ALE and RM parameter are one-dimensional arrays of the same size, we will use the inner product of the two arrays to compute the RORI.

##### *3.2.2.1.1 Annual Loss Expectancy (ALE)*

The Annual Loss Expectancy parameter is the expected loss from the attack if no CoA is deployed. It is dependent on two parameters, Single Loss Expectancy (SLE) and Annual Rate of Occurrence (ARO).

$$ALE = SLE \times ARO$$

The Single Loss Expectancy is the monetary loss we expect each time the attack occurs. It is the multiplication of the Exposure Factor (EF) by the monetary value of a high-value asset (AV). In our model, the Single Loss Expectancy is an array whose size is equal to the number of high value assets for which a TTC value has been computed.

$$SLE = \left[asset_{\text{monetary-value}} \times EF\right]$$

The Exposure Factor (EF) is the percentage of the value of a given asset that gets lost as a result of an incident. A loss can exceed the value of a given asset; in such cases, the EF would be greater than 1. EF could be determined for each asset impacted, but we can simplify it to a unique value of EF for an attack. For each SIEM label type, we can set an exposure factor. The choice that was made for the SIEM labels is as follows:

| **SIEM Label** | **Initial Access** | **Lateral Movement** | **Data Exfiltration** | **Others** |
|---|---|---|---|---|
| Exposure Factor | 0,1 | 0,4 | 0,8 | 0,3 |

The monetary value of an high-value asset is estimated by multiplying its Criticality Rate by the Mean Asset Value. The Criticality Rate of an high-value asset is an estimation of the relative weight of the asset inside the company. It is given by the BIA component, by weighting the high value asset criticality over the mean criticality of the assets of the company.

The Mean Asset Value is configurable by the company, but it can also be estimated as a fraction of the Annual Infrastructure Value.

The Annual Rate of Occurrence (ARO) is the number of time the threat vulnerability is likely to be exploited in a year. It is made of the product of the Annual Attack Rate and the Exploit Probability.

$$ARO = AAR \times EP$$

The Annual Attack Rate (AAR) is the estimation of the number of attacks that the company faces in one year. This estimation should be given by the company, for example based on the logs.

The Exploit Probability (EP) is the probability that an attacker will exploit the threat vulnerability. Assuming that the higher the TTC, the less likely the attack can compromise the system, we propose a formula to link the initial TTC of a high-value asset for a threat to the probability that this threat will be exploited by an attacker:

$$\text{Exploit_Probabilty}(asset) = \frac{1}{\frac{\text{initialTTC(asset)}}{TTC_0} + 1}$$

When the initial TTC grows large enough, the Exploit Probability approaches 0; the vulnerability is very hard to exploit. When the initial TTC approaches 0, the Exploit Probability approaches 1; the vulnerability is easily exploited.

$TTC_0$ is a reference Time to Compromise value for high value assets, indicating when there is 50% chance that the high value asset will be compromised if an attack occurs. $TTC_0$ can simply be estimated by having a couple (TTC, Exploit_Probability) value. For example, if a TTC value has been estimated for an asset, and a success rate of Exploit_probability has been observed for attackers to compromise that asset, then $TTC_0$ will be given by the following formula:

$$TTC_0 = \frac{TTC(asset) \times \text{Exploit_Probability}(asset)}{1 - \text{Exploit_Probability}(asset)} with\ TTC \neq 0, EP \neq 1, EP \neq 0$$

To sum it up, the Annual Loss Expectancy is an array of size equals to the number of high value assets for which a TTC value has been computed. It is obtained using the following formula:

$$ALE = SLE \times ARO = \left[asset_{\text{monetary-value}} \times EF \times AAR \times EP(asset)\right]$$

#### *3.2.2.1.2 Risk Mitigation (RM)*

The Risk Mitigation (RM) parameter evaluates how efficient the countermeasure is to mitigate the attack, and is expressed as a percentage. In the context of the SOCCRATES project, the estimation of this parameter is based on Time to Compromise (TTC) values, computed by the Attack Defense Graph Analyser. A set of high value assets will be used as a base to compute initial TTC values ($initialTTC$), and the same high value assets will be used to compute expected TTC values after the CoA is applied ($coaTTC$). These TTC values will be provided to the RP as an array of size equals to the number of high value assets for which a TTC value has been computed.

The Risk Mitigation is estimated as follows:

$$\text{Risk Mitigation} = \left[\frac{coaTTC(asset) - initialTTC(asset)}{coaTTC(asset) + TTC_0}\right]$$

The $TTC_0$ reference is the same as the one used in the Exploit Probability formula (see Annual Loss Expectancy). $TTC_0$ acts as a scaling time parameter and also allows to avoid issues when initialTTC = 0.

The Risk Mitigation formula can be written using the Exploit Probability expression:

$$\text{Risk Mitigation} = \left[1 - \frac{\text{Exploit_Probabilty}\big(\text{coaTTC}(asset)\big)}{\text{Exploit_Probabilty}\big(\text{initialTTC}(asset)\big)}\right]$$

If $coaTTC \leq initialTTC$, the Time to Compromise of the high value assets is worse after applying the CoA. In such situations, we consider that the CoA is harmful, and we set RM equal to 0.

Based on this formula, it is possible to check that *RM* approaches 1 when *coaTTC* grows, which reflects the fact that the risk is entirely mitigated. Conversely, when *coaTTC* is close to *initialTTC*, *RM* approaches 0, since the CoA does not improve the situation.

#### *3.2.2.1.3 Annual Response Cost (ARC)*

The Annual Response Cost parameter is the monetary deployment cost of the CoA, and will be provided by the CoA Generator as a multidimensional array, and the RP will sum over the dimension to obtain a global monetary cost for each CoA.

$$ARC = \sum CoA \text{ monetary cost}$$

#### *3.2.2.1.4 Annual Infrastructure Value (AIV)*

The Annual Infrastructure Value parameter will be provided beforehand by the pilot site during the initialization phase, by evaluating the monetary value of its infrastructure. This value should be evaluated once, and will then remain stable. In real world deployment, it might be updated periodically, but since it mainly acts as a scaling parameter in order to adapt the computed RORI values to the size of the company, it has no real impact on the ranking of the CoAs.

### 3.2.2.2 Business Impact: Operational impact quantification metric

The main goal of Containment CoAs is to stop an attack and protect the infrastructure, but it can temporarily alter the proper functioning of the company. In order to assess the negative operational impact of a Containment CoA, the RP relies on the Business Impact (BI) evaluated by the Business Impact Analyser component. With this estimation, the RP ranks the Containment CoAs by lowest BI value, so that the analyst can make the best decision in a short-term view, knowing the balance between system protection and impact on the business processes.

The Containment CoAs consider two types of defenses: *Host Isolation* and *Traffic Filtering*.

The *Host isolation* defense is represented by a shock event of probability 1 on the targeted asset. Since the targeted asset will be isolated from the rest of the network, it can be modeled as if the asset was compromised.

For the *Traffic Filtering* defense, if the connection interruption is between two internal assets, then it is possible to compute a Business Impact. It is done by deleting, in the BIA model, the link between the two assets, and by adding one Shock Event on each node with an impact probability equal to the probability of the deleted link (taking into account the direction) as illustrated in Figure 3-1.

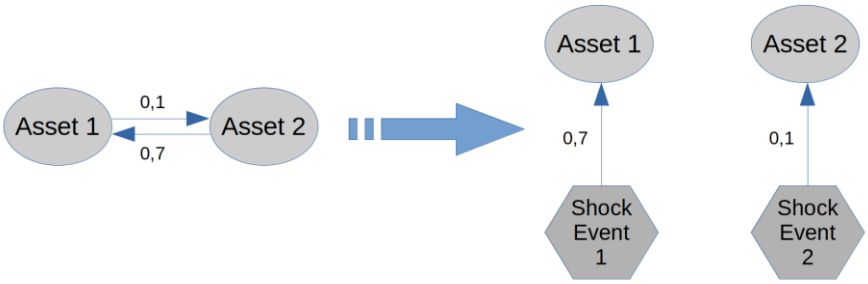


**Figure 3-1 – Modelling of Traffic Filtering defense in Business Logic**

### 3.2.2.3 CACAO Playbooks

The SOCCRATES project perspective is to automate the Incident Response. Therefore, the output of the RP consist of playbooks based on the Collaborative Automated Course of Action Operations (CACAO) standard [CACAO]. This standard is relevant as it can integrate various command types in a single playbook, in particular manual and automated commands. The RP will automatically transform the defenses list of CoAs into a workflow of several commands forming a CACAO playbook.

An example of CACAO playbook, containing only manual commands, is provided in Figure 3-2. Manual commands are human readable version of the defense deployment, containing a short description of the defense.

```
{"type": "playbook",
 "id": "playbook--soccrates-coa-123456123456",
 "name": "Soccrates CoA Playbook",
 "description": "This playbook will apply the CoA in the system",
 "playbook_types": [
   "prevention"
 ],
 "created_by": "identity--uuid",
 "created": "2021-07-19T23:32:24.399Z",
 "labels": [
  "Patch",
  "2FA",
  "CoA"
 ],
 "features": {
  "parallel_processing": true,
 },
 "workflow_start": "start--soccrates-coa-123456123456",
 "workflow": {
  "start--soccrates-coa-123456123456": {
   "type": "start",
   "on_completion": "parallel--soccrates-coa-123456123456"
  },
  "parallel-soccrates-coa-123456123456": {
   "type": "parallel",
   "name": "Apply defenses of the Coa",
   "next_steps": [
    "defense1--soccrates-coa-123456123456",
    "defense2--soccrates-coa-123456123456"
   ]
  },
  "defense1--soccrates-coa-123456123456": {
   "type": "single",
   "name": "Patch Application",
   "description": "This step will Patch the application on selected hosts",
   "on_completion": "end--soccrates-coa-123456123456",
   "target": "112233",
   "commands": [
    {
     "type": "manual",
     "command": "Download and install the patch for the application in selected hosts"
    }
   ]
  },
  "defense2--soccrates-coa-123456123456": {
   "type": "single",
   "name": "Two Factor Authentication",
   "description": "This step will enable Two Factor Authentication",
   "on_completion": "end--soccrates-coa-123456123456",
   "target": "445566",
   "commands": [
    {
     "type": "manual",
     "command": "Enable Two Factor Authentication on the selected hosts"
    }
   ]
  },
  "end--soccrates-coa-123456123456": {
   "type": "end"
  }
 }
}
```


**Figure 3-2 – Example of a CACAO playbook for a CoA**

Automated commands will take the form of OpenC2 commands. OpenC2 [OpenC2] is a language for Command and Control between machines, and will be used for reconfiguration. The generation of OpenC2 commands has yet to be implemented.

Figure 3-3 and

Figure 3-4 illustrate examples of OpenC2 commands for the containment CoA defenses, *Host Isolation* and *Traffic Filtering*.

```
{
   "action": "contain",
   "target": {
      "device": {
         "device_id": "123e4567-e89b" <- IMC UUID
         "description": "computer_name"
      }
   }
}
```

**Figure 3-3 – Host Isolation OpenC2 command example**

```
{
   "action": "deny",
   "target": {
      "ipv4_connection": {
         "protocol": "tcp",
         "src_addr": "1.2.3.4",
         "src_port": 10996,
         "dst_addr": "198.2.3.4",
         "dst_port": 80
      }
   },
   "args": {
      "start_time": 1642769450012560273,
      "duration": 500,
   },
}
```

**Figure 3-4– Traffic Filtering OpenC2 command example**

## 3.3 Component architecture

The Response Planner will run with a Python REST API. This API has been developed using the external libraries Uvicorn[5] and FastAPI[6].

The Response Planner has two Cortex Analyzers, one for each of its functions:

- SOCCRATES_RP_CONTAINMENT_COA_GENERATOR
- SOCCRATES_RP_COA_RANKING_ANALYZER

These Cortex Analysers have been developed using the external libraries *CortexUtils*[7] and *Requests*[8]. The first one aims at generating containment CoAs based on attack detection data provided by REASENS, which is part of the AI-based Anomaly Detection (AAD), or by the pilot site SIEM. The second Cortex Analyser aims at computing the financial and operational impacts of CoAs in order to obtain a

5 https://www.uvicorn.org/
6 https://github.com/tiangolo/fastapi
7 https://cert-bund-cortex-analyzers.readthedocs.io/en/latest/cortexutils3.html
8 https://docs.python-requests.org/en/master/

CoA ranking that will help the SOC operator in decision making. The details of these analysers are provided in Deliverable D2.3 [D2.3].

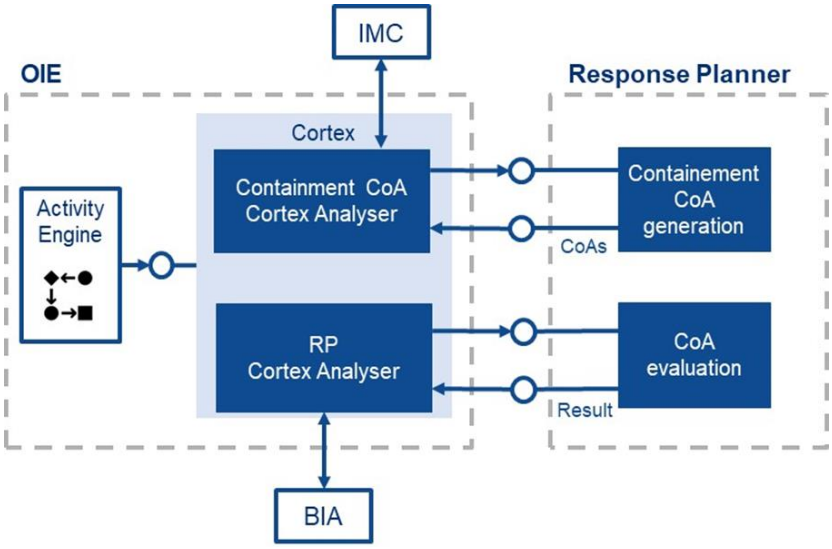


**Figure 3-5 – Response Planner API interaction with Cortex Analyzers and other components**

The orchestrator (OIE) interacts with the Response Planner API through the Response Planner Cortex Analysers, as shown in Figure 3-5.

The OIE will only use the Containment CoA Cortex Analyser in the Use Case 1. In the *Containment – Preliminary response* step, in order to generate "stop exfiltration" CoAs, and in the *Containment – Final containment* step, to generate "contain host" CoAs.

The OIE will regularly call the RP Cortex Analyser for assessing CoAs or Containment CoAs. The RP will interact with the BIA in order to obtain an estimation of the criticality of high value assets or to obtain an evaluation of the operational impact of Containment CoAs. The RP will return to the OIE a list of ranked CoAs with CACAO playbooks to deploy them.

The RP can request from the IMC the UUID corresponding to an IP address. The RP may also interact with the IMC to obtain a list of security controls, in order to provide specific OpenC2 commands in the CACAO playbooks.

When the CoA Ranking functionality is called, the RP API will store the resulting ranked CoAs inside a file, and will generate an URL. This URL will give the SOC operator the possibility to visualize the results inside the RP Graphical User Interface.

## 3.4 Graphical interface of the Response Planner

The Response Planner has a JavaScript Web Graphical User Interface that summarizes all the parameters relevant to establish the CoA ranking. The table inside the Web GUI is built using the JavaScript library *Tabulator*[9].

There are three display box outside of the table, which contain parameters that do not depend on a specific CoA : The Annual Infrastructure Value (AIV), the Annual Loss Expectancy (ALE) and the type of

9 https://github.com/olifolkerd/tabulator

CoA (Containment or not). Figure 3-6 presents the interface with the compact view of the CoAs (as opposed to the full display presented in Figure 3-7).

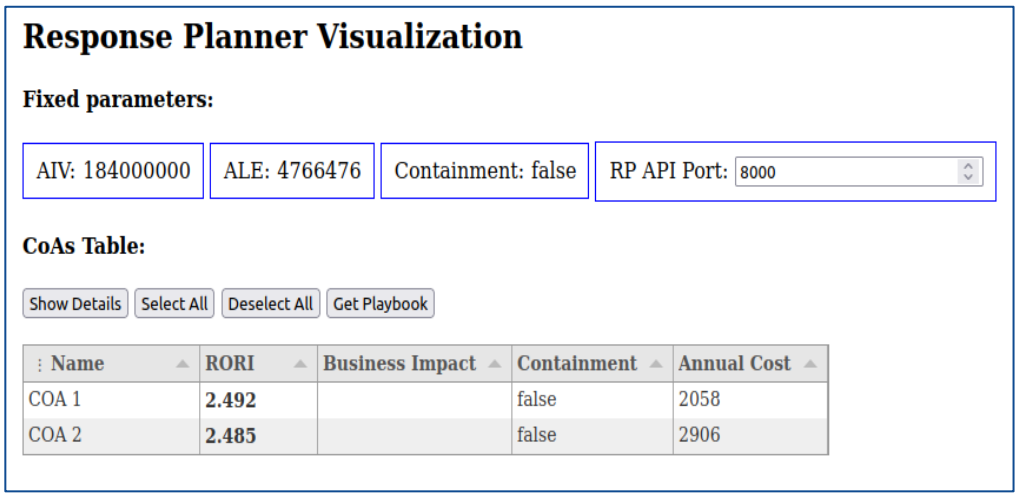


**Figure 3-6 – Response Planner Graphical User Interface, compact CoA display**

The RORI column displays the RORI value computed for each CoA, and is empty for Containment CoA. By default the table will be sorted by decreasing RORI values.

The Annual Cost field correspond to the Annual Response Cost of the CoA (RORI parameter), and is empty for Containment CoAs.

The table can be sorted by decreasing or increasing value for each column by clicking on the arrow next to the column name.

The *Hide* and *Show* buttons are used to reveal more details about each CoA. The table grows to show the defense list of each CoA (Ref, Defense Name, Defense Info, Mitre Ref from the CoA Generator) and also the list of hosts impacted by the CoA. Figure 3-7 shows the graphical interface of the RP with full details of the CoAs.

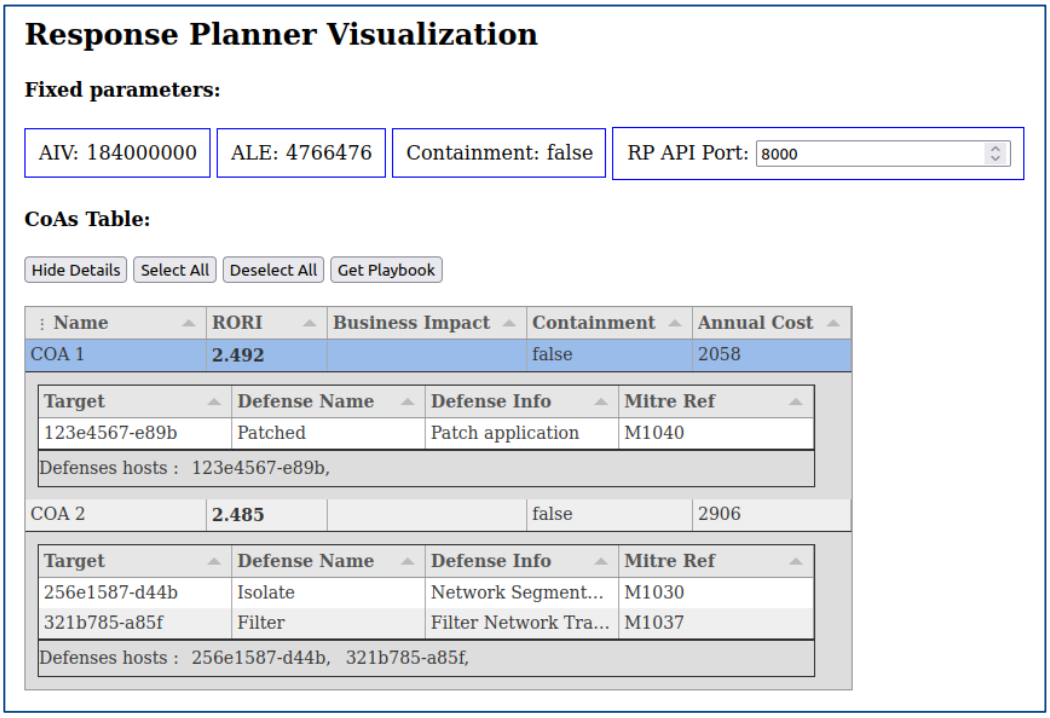


**Figure 3-7 – Response Planner Graphical User Interface, full CoA display**

The Business Impact field is the value returned by the BIA when evaluating the Containment CoA Business Impact on the company.

Figure 3-8 displays an example of the graphical interface of the RP containing a Containment CoA.

The BI values are colored following the same coloring rule applied in the BIA (in Section 2.4.4.1):

- green if the value is between 0 and 1/3, considering that the CoA has a low operational impact
- orange if the value is between 1/3 and 2/3, meaning a medium operational impact
- red, if the value is between 2/3 and 1, meaning a high operational impact

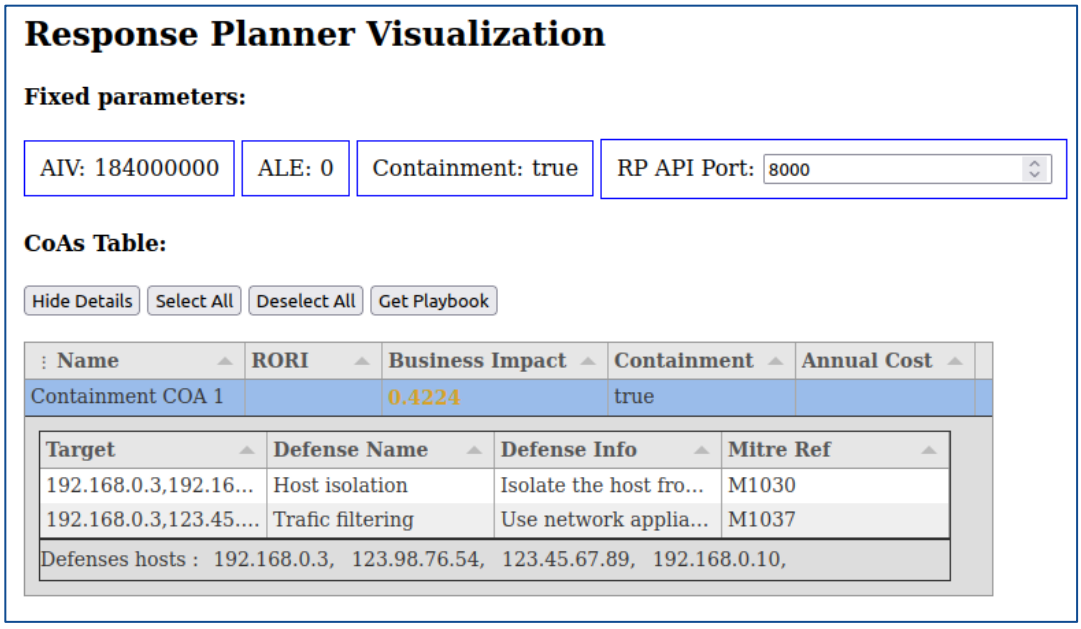


**Figure 3-8 – Response Planner Graphical User Interface, Containment CoA example**

Each row of the table is selectable, leaving the possibility for the analyst to select the CoAs to deploy. Then, by clicking the "Get Playbook" button, the CACAO playbook for each of the selected CoAs will be generated and displayed in different tabs.

## 3.5 Response Planner deployment

A docker image has been set up in order to properly deploy the Response Planner along with the other SOCCRATES components. It will run the API and the graphical interface of the component, and leave them accessible on two different ports.

The RORI constants (Annual Attack Rate, Annual Infrastructure Value ,$TTC_o$, Mean Asset Value) can be initialized in a configuration file of the RP API.

## 3.6 Next steps

The Response Planner is almost fully implemented, and can be integrated to the SOCCRATES platform to be part of the second pilot. However, one extra feature still needs to be added in conjunction with the Automated Reconfiguration component. More specifically, the next steps of the development will focus on the following points:

- Fine tuning of the parameter evaluation of the RORI metric, based on the feedback of the second pilot
- Generation of CACAO playbooks with OpenC2 commands

# 4 References

[CACAO] OASIS Collaborative Automated Course of Action Operations, Security Playbooks Version 1.0, available at https://docs.oasis-open.org/cacao/security-playbooks/v1.0/security-playbooks-v1.0.html

[D2.2] Deliverable D2.2, "System architecture & Interface specification; Initial version", SOCCRATES project

[D2.3] Deliverable D2.3, "System architecture & Interface specification; Final version", SOCCRATES project

[D5.1] Deliverable D5.1, "Definition of the business model structure, inputs and interfaces", SOCCRATES project

[D5.2] Deliverable D5.2, "Business Logic Modelling and Impact Analyser & Response Planner - Initial prototype", SOCCRATES project

[OASIS] The Organization for the Advancement of Structured Information Standards, available at https://www.oasis-open.org/

[OpenC2] Open Command and Control standard, available at https://openc2.org/

# 5 Abbreviations

This glossary serves as inventory of abbreviations used in the document.

*This is a standard glossary, used for all SOCCRATES report deliverables; it will be expanded when necessary.*

| Acronym | Description |
|---|---|
| AAD | AI-based Anomaly Detection |
| AAR | Annual Attack Rate |
| ACT | semi-Automated Cyber Threat intelligence |
| ADG | Attack Defence Graph |
| AEF | Argus Event Format |
| AI | Artificial Intelligence |
| AIT | AIT Austrian Institute of technology |
| AIV | Annual Infrastructure Value |
| ALE | Annual Loss Expectancy |
| API | Application Programming Interface |
| APT | Advance Persistent Threat |
| ARC | Annual Response Cost |
| ARO | Annual Rate of Occurrence |
| ATOS | ATOS Spain |
| AV | AntiVirus |
| BC | Business Company |
| BI | Business Impact |
| BIA | Business Impact Analyser |
| BF | Business Function |
| BP | Business Process |
| BPMN | Business Process Model and Notation |
| CSIRT | Computer Security Incident Response Team |
| CC | Command and Control |
| CERT | Computer Emergency Response Team |
| CMDB | Configuration Management Database |
| CSIRT | Computer Security Incident Response Team |
| CoA | Course of Action |
| CTI | Cyber Threat Intelligence |
| DC | DataCentre |
| DGA | Domain Generated Algorithm |
| DNS | Domain Name System |
| EDR | Endpoint Detection and Response |
| EF | Exposure Factor |
| ELK | Elasticsearch/Logstash/Kibana |
| EP | Exploit Probability |
| FRS | Foreseeti |
| FSC | F-secure |
| ICT | Information and Communication Technology |
| IDS | Intrusion Detection System |
| IMC | Infrastructure Modelling Component |

| | |
|---|---|
| IMT | Institut Mines Télécom – Télécom SudParis |
| INTF | Interface |
| IoC | Indicators of Compromise |
| IP | Internet Protocol |
| IPS | Intrusion Prevention System |
| IRM | Incident Response and Management |
| ITIL | Information Technology Infrastructure Library |
| KTH | Kungliga Tekniska högskolan - Royal Institute of Technology |
| LAN | Local Area Network |
| LDAP | Lightweight Directory Access Protocol |
| $M_n$ | Infrastructure Model (at time *n*) |
| MNM | mnemonic |
| MSSP | Managed Security Service Provider |
| MTTD | Mean Time To Detection |
| NOC | Network Operations Centre |
| OIE | Orchestration & Integration Engine |
| OpenC2 | Open Command and Control |
| OT | Operational Technology |
| OS | Operating System |
| RORI | Return on Response Investment |
| RM | Risk Mitigation |
| RP | Response Planner |
| SDN | Software Defined Network |
| SHS | Shadowserver |
| SIEM | Security information and event management |
| SLE | Single Loss Expectancy |
| SOAR | Security Orchestration, Automation and Response |
| SOC | Security Operation Centre |
| SOCCRATES | SOC & CSIRT Response to Attacks & Threats based on attack defence graph Evaluation Systems |
| SSL | Secure Sockets Layer |
| TAP | Test Access Point |
| TIP | Threat Intelligence platform |
| TLS | Transport Layer Security |
| TNO | Nederlandse Organisatie voor toegepast natuurwetenschappelijk onderzoek |
| TTC | Time To Compromise |
| UC | Use Case |
| URL | Uniform Resource Locator |
| UUID | Universally Unique IDentifier |
| VLAN | Virtual LAN |
| VM | Virtual Machine |
| VTF | Vattenfall |