D3.5 Advanced User Interface

This project has received funding from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement No 101021274

Protection of Critical Infrastructures from advanced combined cyber and physical threats

Deliverable nº:	D3.5
Deliverable name:	Advanced User Interface
Version:	1.0
Release date:	29/07/2022
Type* - Dissemination level**	Report - Public
Status:	Final version
Editors	UPVLC
Contributing WP	WP3

Abstract

This deliverable presents the Human-Machine Interface (HMI) solution developed as part of the Cyber Situation Awareness (CSA) component of PRAETORIAN's project, its architecture and functioning. It details the different advanced visualization techniques implemented in the application. It describes the information validation functions provided by the tool to the CSA operator.

**Type. Report; Demonstrator; Ethics*

***Dissemination Level. Public; Confidential (Confidential, only for members of the consortium (including the Commission Services)); RESTREINT UE (Classified information, RESTREINT UE (Commission Decision 2015/444/EC)).*

Disclaimer

This document contains material, which is the copyright of certain PRAETORIAN beneficiaries, and may not be reproduced or copied without permission.

The information appearing in this document has been prepared in good faith and represents the views of the authors. Every effort has been made to ensure that all statements and information contained herein are accurate; however, the authors accept no statutory, contractual or other legal liability for any error or omission to the fullest extent that liability can be limited in law.

This document reflects only the view of its authors. Neither the authors nor the Research Executive Agency nor the European Commission are responsible for any use that may be made of the information it contains. The use of the content provided is at the sole risk of the user. The reader is encouraged to investigate whether professional advice is necessary in all situations.

No part of this document may be copied, reproduced, disclosed, or distributed by any means whatsoever, including electronic without the express permission of the PRAETORIAN project partners. The same applies for translation, adaptation or transformation, arrangement or reproduction by any method or procedure whatsoever.

PRAETORIAN

PRAETORIAN's strategic goal is to increase the security and resilience of European CIs, facilitating the coordinated protection of interrelated CI against combined physical and cyber threats. To that end, the project will provide a multidimensional (economical, technological, policy, societal) yet installation-specific toolset comprising: (i) a Physical Situation Awareness system, (ii) a Cyber Situation Awareness system; (iii) a Hybrid Situation Awareness system, which will include digital twins of the infrastructure under protection; and (iv) a Coordinated Response system. The PRAETORIAN toolset will support the security managers of Critical Infrastructures (CI) in their decision making to anticipate and withstand potential cyber, physical or combined security threats to their own infrastructures and other interrelated CIs that could have a severe impact on their performance and/or the security of the population in their vicinity.

The project will specifically tackle (i.e. prevent, detect, response and, in case of a declared attack, mitigate) human-made cyber and physical attacks or natural disasters affecting CIs. It will also address how an attack or incident in a specific CI can jeopardise the normal operation of other neighbouring/interrelated CIs, and how to make all of them more resilient, by predicting cascading effects and proposing a unified response among CIs and assisting First Responder teams.

PRAETORIAN is a CI-led, user-driven project, which will demonstrate its results in three international pilot clusters, some of them cross border -Spain, France and Croatia-, involving 9 outstanding critical infrastructures: 2 international airports, 2 ports, 3 hospitals and 2 power plants.

Document history:

Version	Date of issue	Content and changes	Partner
0.1	25/02/2022	ToC.	UPVLC
0.2	08/03/2022	ToC updated version.	UPVLC, THALES
0.3	15/04/2022	Chapter 1, Section 2.1	UPVLC
0.4	7/05/2022	Sections 2.2 and 2.3	UPVLC
0.5	8/07/2022	Sections 3, 4, 5 and 6, and major changes all over.	UPVLC
0.9	13/07/2022	Quality check. Version for peer review.	THALES
1.0	29/07/2022	Final version	UPVLC, THALES

List of Authors:

Partner	Author
UPVLC	Israel Pérez, Javier Hingant, Alfonso Climente
THALES	Stéphane Paul (quality check only)

Peer reviewed by:

Partner	Reviewer
ICCS	Antonios Karteris, Lazaros Papadopoulos
KONČAR	Morana Loncar

Table of Contents

Index of Tables

Index of Figures

Abbreviations and Acronyms

API	Application Programming Interface
AR	Augmented Reality
CAL	Cybersecurity Assessment Lab
CDT	Cyber Digital Twin
CFE	Cyber Forecaster Engine
CI	Critical Infrastructure
CSA	Cyber Situation Awareness
COTS	Commercial off-the-shelf
DDP	Distributed Data Protocol
DoA	Description of Action
H2020	Horizon 2020. The EU Framework Programme for Research and Innovation
HMI	Human-Machine Interface
HSA	Hybrid Situation Awareness
IOP	Interoperability Platform
IT	Information Technology
JSON	JavaScript Object Notation
MQTT	MQ Telemetry Transport
MVC	Model-View-Controller
NATS	Neural Autonomic Transport System
OT	Operational Technology
PHP	Hypertext Preprocessor
PRAETORIAN	Protection of Critical Infrastructures from advanced combined cyber and physical threats
REST	Representational State Transfer
SA	Situational Awareness
TLS	Transport Layer Security
VR	Virtual Reality
WP	Work Package

Executive summary

As part of the PRAETORIAN project's WP3 "Cyber Situation Awareness (CSA)", this deliverable D3.5 "Advanced User Interface" is the result of the efforts done in the scope of the task T3.5 "Advanced visualization techniques". The outcome is the development of an innovative CSA's Human-Machine Interface (HMI) application directed at enhancing the situation understanding of the Critical Infrastructure (CI)'s cyber domain. This document presents the CSA HMI architecture and functionality, details both the information visualization mechanisms implemented in the tool and the information validation capabilities offered to the CSA operator to reach an adequate cyber situation awareness, and finally describes additional functions of the system.

The work here presented, along with deliverables [D3.2], [D3.3] and [D3.4], constitutes a main input for the final WP3 task – T3.6 "CYBER SA system Integration".

1. Introduction

1.1 Purpose of the document

This document constitutes the output of the task T3.5 “Advanced visualization techniques”, which focuses on the development and application of novel information visualization procedures with the goal of improving the Cyber Situation Awareness (CSA) provided by the Human-Machine Interface (HMI) application of the PRAETORIAN’s CSA system.

1.2 Scope of the document

This deliverable is aimed at enhancing the cyber domain’s situation understanding of the Critical Infrastructures (CI) under PRAETORIAN’s platform protection. To this end, a Cyber Situation Awareness (CSA) system’s Human-Machine Interface (HMI) application is developed which offers advanced visualization capabilities in order to represent the cyber information provided to the HMI by the CSA components developed in the context of both T3.2 “Cyber Forecaster Engine” (CFE) and T3.3 “Dedicated Cybersecurity Digital twin” (CDT).

1.3 Structure of the document

This document is structured as follows:

- Section 2 introduces the CSA HMI application, its architecture and functional flow.
- Section 3 focuses on both the implemented visualization mechanisms and the information validation functionalities.
- Section 4 provides further capabilities applied to the HMI of the CSA.
- Section 5 summarizes the main conclusions of the current work.
- Section 6 provides the references of the document.

2. Presentation of the CSA HMI

2.1 Introduction to the CSA HMI

As described in the Description of Action [DoA] of the project, the Work Package (WP) 3 of PRAETORIAN focuses on the development of an advanced and scalable Cyber Situation Awareness (CSA) system for the European Critical Infrastructures (CIs) protection capable of both preventing and detecting cyber threats, and anticipating problems to avoid or limit them if possible. As explained in depth in the deliverable "Cyber Situation Awareness system" [D3.6], the CSA solution is composed of several interrelated sub-systems: a) different Cyber Digital Twins (CDTs) that represent some CI's Information Technology (IT) and Operational Technology (OT) systems [D3.3]; b) a Cybersecurity Assessment Lab (CAL) which includes a set of technical tools to generate cybersecurity events by performing cyber-attacks on the CDT [D3.4]; c) a Cyber Forecaster Engine (CFE) which aims at predicting the potential goals of an attack through event-pattern recognition and hypothetical reasoning engines [D3.2]; d) and finally a Human-Machine Interface (HMI) that feeds from the other CSA sub-systems in order to provide to the security officer a real-time situation understanding of the cyber domain.

Regarding this last one in particular, the main purpose of the CSA HMI – developed in the framework of task T3.5 "Advanced visualization techniques" – is to enhance the cyber situation awareness of the CSA operator and, therefore, to facilitate the real time decision making by supporting the verification and validation of the information provided by the CSA CFE. To this end, the HMI of the CSA implements novel representation types to visualize the cyber information gathered. It also offers different functions to the security operator to come to the CI's real-time cyber situation acknowledgment (cf. Chapter 3).

2.2 CSA HMI architecture

In order to ensure its modularity and scalability, and as presented in Figure 1, the front-end side of the CSA component consists of a novel web-based tool designed and implemented following a decoupled approach, which is composed of different functional and interdependent modules that are described below.

Regarding the interoperability of the CSA HMI with external tools – and as explained in D3.6 "Cyber Situation Awareness system", the HMI application communicates with the PRAETORIAN's main database through the Interoperability Platform (IOP), instead of directly exchanging cyber-related data with the CSA back-end components – that is, the CDT, CFE and CAL.

To this end, as was already introduced in deliverable "PRAETORIAN toolset architecture implementation description" [D2.4], the PRAETORIAN's IOP offers four distinct communication mechanisms – MQTT, NATS, API REST, and DDP – to ensure the interoperability among the different systems and components implemented in the framework of the project. All of them are able to interact internally with the [MongoDB] that, as the main PRAETORIAN's database, stores all the data produced by the different modules of the overall project's solution in order to be available to all those sub-systems that may require it. The approach ensures the decoupling between the different actors and subsystems of the PRAETORIAN architecture, as the presence of the MongoDB and its associated message-passing infrastructure, inside the IOP, provides means to isolate different components.

In the particular case of the CSA HMI, the tool makes use of the API REST interface only during the operator authentication process, while the rest of the interactions are achieved through the Neural Autonomic Transport System [NATS] protocol following the publish-subscribe paradigm. The communications are end-to-end secured by the use of Transport Layer Security [TLS] ciphering, and the data exchange formats are based in JavaScript Object Notation [JSON] schemas.

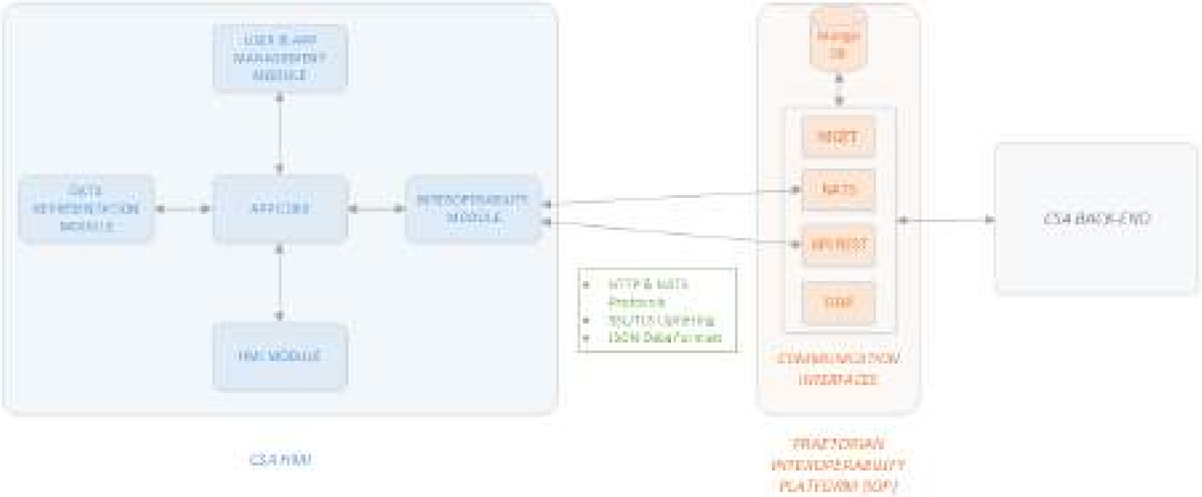


Figure 1 - CSA HMI Architecture

2.2.1 Application Core

As the kernel of the system, the application core module is responsible for the central tasks and processes of it as well as managing and controlling the well-functioning of the rest of the CSA front-end modules. It is the entry point of the user interactions, through the HMI module with the rest of the tool components.

2.2.2 User & Application Management Module

This module is in charge of all the configuration tasks of both the CSA front-end application and its corresponding users. On the one hand, this part of the system is at the helm of setting up the communication parameters to carry out successfully the data exchange of the cyber information at the interoperability module (cf. Section 4.2). Specifically, it sets the end-points and connectivity credentials with the main PRAETORIAN database – where all the relevant information provided by the CSA back-end components is stored – and manages the activation/deactivation of the cyber data ingestion and sending. On the other hand, the module also handles the management of the application user's details, that is, the modification of his/her user password.

2.2.3 Data Representation Module

This module is responsible for carrying out the visualizations of the cyber information (cf. Chapter 3) produced by the CSA back-end modules – in particular, from the Cyber Digital Twin (CDT) and the Cyber Forecaster Engine (CFE) modules – and available at the PRAETORIAN IOP's database. To this end, this module makes use of advanced visualization techniques – such as multidimensional graphs and diagrams, or 3D on-screen interactive visualizations – to achieve the representation of a set of different cyber data types like primary and supporting assets, the relations among them, and their related CFE alerts.

2.2.4 HMI Module

As its name implies, the Human-Machine Interface (HMI) module constitutes the interface between the user actions and the CSA front-end application. It transmits to the application core module the inputs gathered from the operator. On its purpose of providing an enhanced cyber situation awareness, from this module – and in combination with the data representation module – the user is able to both interact with the different visualization components, to manage the currently displayed cyber information representations, and to validate different CFE alerts information (cf. Section 3.3). Additionally, it also permits to access and manage – in combination with the user & application management module – the overall system configuration previously described (cf. Sections 4.2 and 4.3).

2.2.5 Interoperability Module

As the front-end's interface with any external component, the interoperability module is responsible for the exchange – that is, both receiving and sending – of any kind of data necessary for the adequate functioning of the CSA front-end application. In particular, it mainly interacts with the PRAETORIAN database, through the IOP communication interfaces, for the consumption of the cyber post-processed information generated by the CSA back-end as well as for the publication of possible CFE alerts updates once validated by the CSA operator.

2.3 CSA HMI operational flow

The sequence diagrams described in the next Figures (i.e., Figure 2 to Figure 6) represent the whole operational flow of the CSA HMI application, from the CSA operator authentication until the cyber information validation process. The design of the HMI solution has been conceived in such a generic and flexible manner that its overall functioning is largely transparent both to the PRAETORIAN's use case to which it is applied and to the person in the role of the CSA operator.

Each of the running steps of the CSA HMI tool are explained in the following.

2.3.1 Authentication process

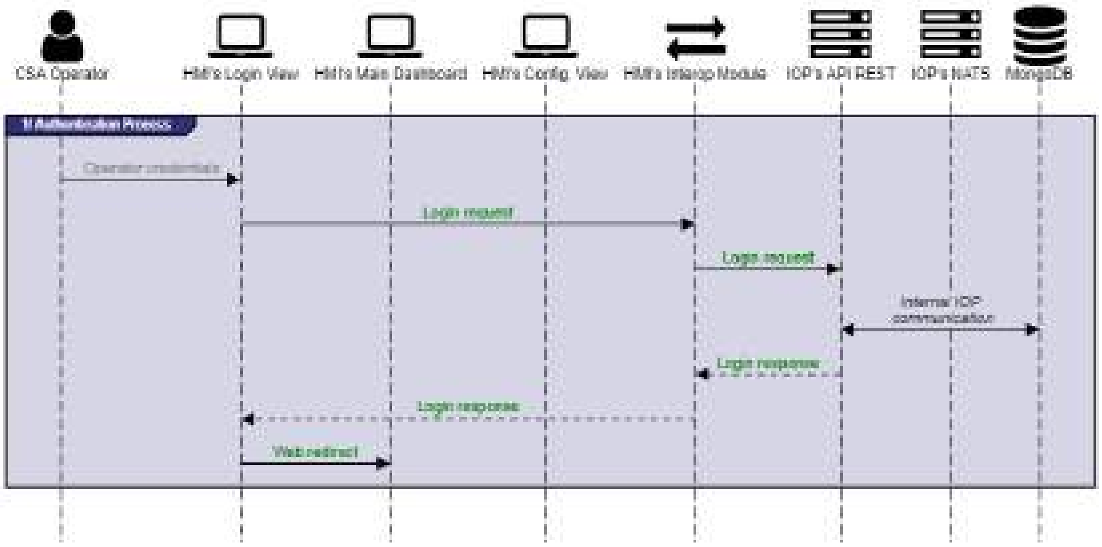


Figure 2 - Authentication process

The first step, once the CSA HMI tool is deployed and running, is the CSA operator's authentication process in the application. As shown in Section 4.1, the HMI solution's log-in relies on a username/password authentication mechanism. Once the CSA user introduces their credentials, a login request is triggered from the HMI's login view and transmitted from the HMI's interoperability module to the PRAETORIAN's IOP through its API REST interface. The credentials provided by the operator, transferred cyphered, are then compared with those salt-stored in the MongoDB database. A successful or failure login response is accordingly returned to the CSA user and, in positive cases, the user is redirected to the HMI's main view.

To adequately achieve the user login process, it is therefore mandatory to previously register the credentials of the user in the role of CSA operator at the PRAETORIAN's main database.

2.3.2 Interoperability configuration

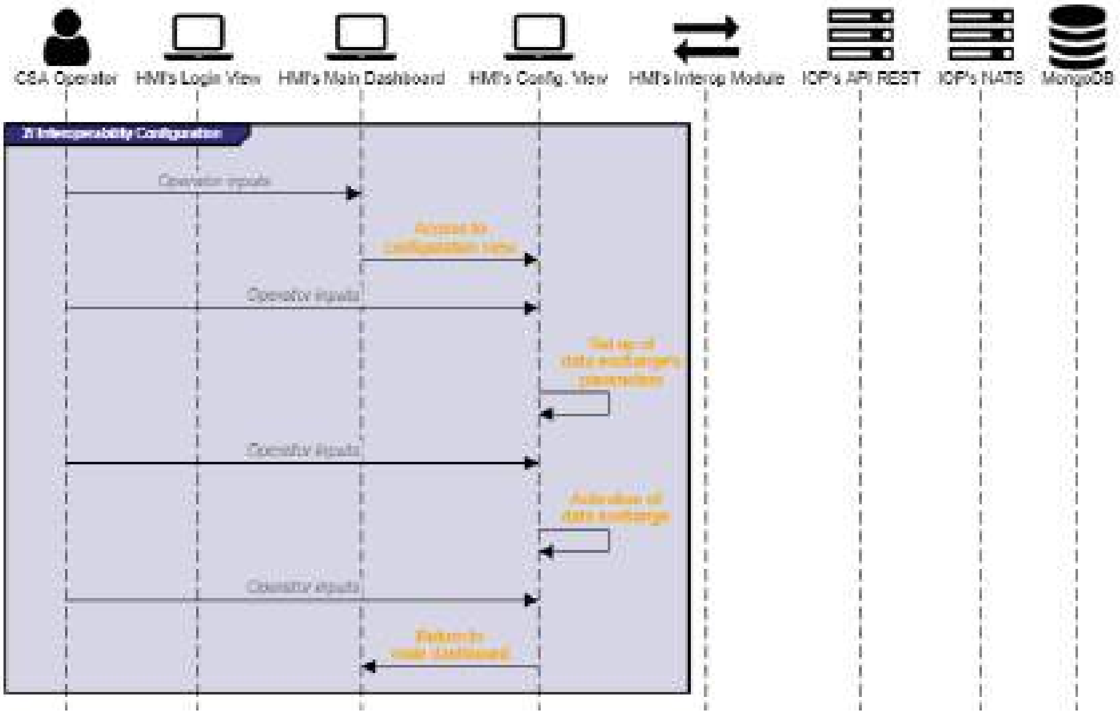


Figure 3 - Interoperability configuration

Once the CSA operator is logged in the CSA HMI, the communication among the application and external components – more specifically, the IOP's MongoDB – may be configured and activated in order to start exchanging data, that is, gathering and representing cyber assets and their related CFE alerts and detections, as well as publishing any CFE alert update generated by the operator. To this end, the CSA user should access the HMI configuration view, from where he/she is able to set up the connection parameters to the IOP's NATS interface – hostname, port, username, and password. This communication configuration is only necessary once, as it is locally saved for upcoming sessions.

As displayed in Section 4.2, and once the interoperability configuration is done, an adjacent toggle button permits the CSA operator to turn on and off the receiving and sending of cyber data between the HMI tool and the PRAETORIAN's database.

2.3.3 HMI configuration

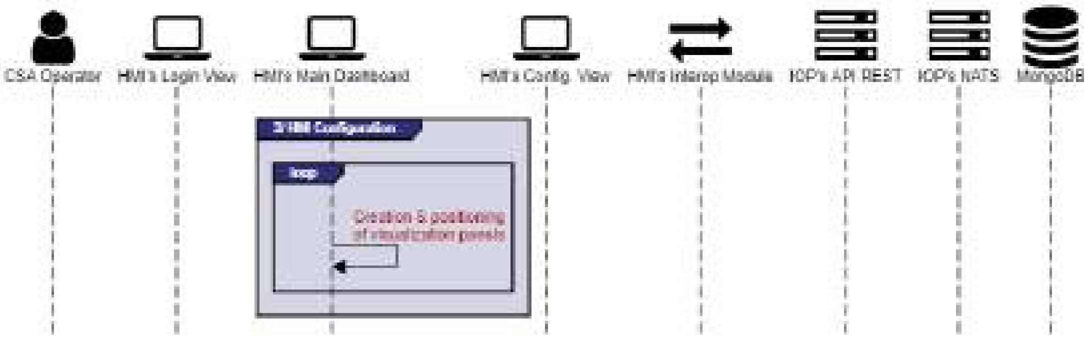


Figure 4 - HMI configuration

As described in Section 4.3, the CSA HMI has been designed to allow the user to widely customize the layout of the representational panels according to his/her own preferences or needs. Starting from a totally empty canvas, the CSA operator is able to create as many data visualization panels as desired, which can be freely positioned and docked at his/her criteria. Any of these information panels can be relocated and/or removed at any time. The user's HMI setup is locally saved at runtime, so that the CSA operator can preserve the current configuration at any new session.

The CSA user is able to set individually which cyber information is represented in each of the visualization panels. As will be presented in Chapter 3 in depth, the CSA HMI implements several enhanced representation techniques in order to offer a variety of advanced means to visualize cyber-related information in an innovative and useful manner.

2.3.4 CSA data reception

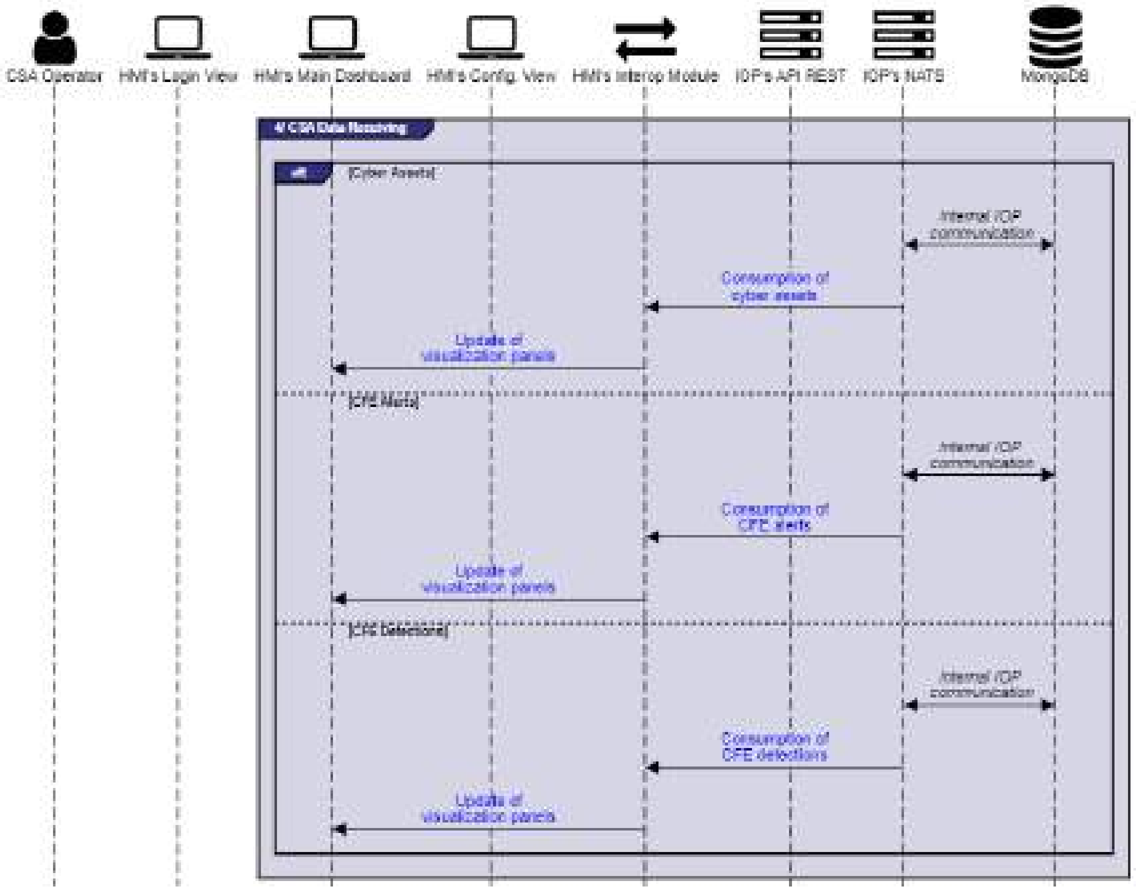

CSA Operator
HMI's Login View
HMI's Main Dashboard
HMI's Config. View
HMI's Interop Module
IOP's API REST
IOP's NATS
MongoDB
4/ CSA Data Receiving
alt
[Cyber Assets]
Internal IOP communication
Consumption of cyber assets
Update of visualization panels
[CFE Alerts]
Internal IOP communication
Consumption of CFE alerts
Update of visualization panels
[CFE Detections]
Internal IOP communication
Consumption of CFE detections
Update of visualization panels

Figure 5 - CSA data receiving

Once the CSA operator has set up the HMI layout at his/her criteria and activated the HMI's interoperability module, gathering and representing cyber information at the HMI application occurs in an automated way without needing any additional user intervention.

To achieve this and always following the publish-subscribe communication pattern, the CSA HMI tool firstly requests for all the currently existing cyber assets – and the relations between them (differentiating between primary and their supporting assets) – in the PRAETORIAN's MongoDB. This query is launched periodically to be aware of any updates related to cyber asset information and their status.

Additionally, the CSA HMI is able to keep track of creations and updates of detections and alerts generated by the CSA CFE by being subscribed to a queue related to the corresponding MongoDB's collections. Thus, whenever a data object's modification in the referred database collections occurs, a message containing its current content is automatically received at the HMI application.

Each time that new or updated cyber information – either cyber assets, CFE detections or CFE alerts – is received at the CSA HMI through its interoperability module, the data representation module reloads the representation of all the visualization panels to integrate the received data changes.

2.3.5 CSA data validation

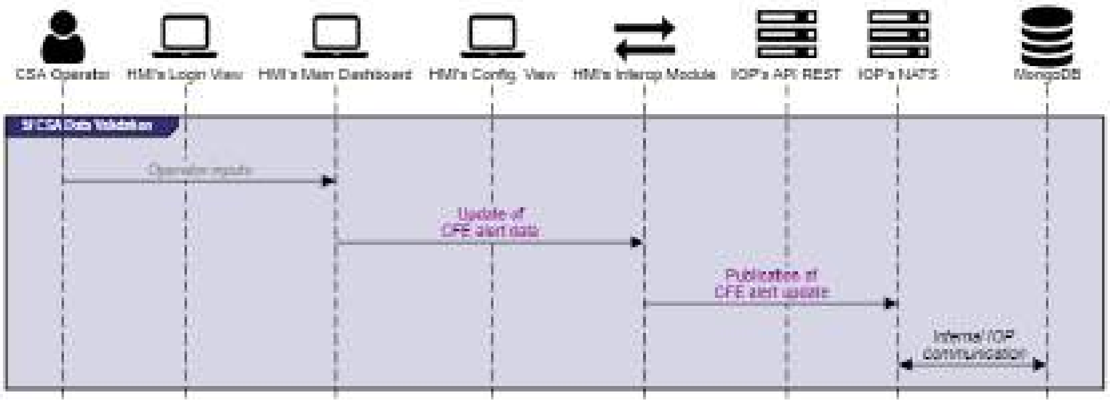


Figure 6 - CSA data validation

On its final goal of supporting the verification and validation of the cyber information produced by the CSA CFE, the HMI application permits the user, in CSA operator role, to acknowledge and validate the veracity of the CFE alerts data. To do so, and as shown in Section 3.3, the CSA HMI offers, in a single information validation dashboard, a set of functions that allows the CSA user to confirm the truthfulness of a CFE alert or to update its current status, among others.

Each time that a modification is confirmed, a CFE alert update message is forwarded to the HMI's interoperability module, which publishes it through the IOP's NATS interface in order to be updated in the PRAETORIAN main database and, therefore, consumed by the CSA CFE for its internal corresponding processing.

3. Information visualization and validation

3.1 Prior considerations

In order to ensure the adequate functioning of the CSA HMI application on its whole and, in particular, to manage to achieve a successful representation of the CSA information in a manner that enhances the overall operator's situation awareness of the cyber domain, some relevant considerations should be mentioned:

1. The IOP's MongoDB, as the main PRAETORIAN database, needs to include specific and independent collections for storing cyber assets information, CFE detections, and CFE alerts. An additional collection to store the CSA user credentials is also needed.
2. The IOP should implement internal services to interact with the different MongoDB collections and provide the requested information to the concerned PRAETORIAN sub-systems.
3. The CSA HMI is completely stateless and will represent data currently on the database.
4. The more enriched information and detailed level provided to the CSA HMI, the better Situational Awareness will be offered to the operations to enhance their Situation Understanding.

3.2 Information visualization techniques

Visualization techniques will be the key element inside the Cyber Situational Awareness (CSA) HMI with the goal of providing a proper situation understanding to the tool operator's and PRAETORIAN analysts. To achieve that goal, these techniques must provide means to enhance Situational Awareness at a glimpse. In the following sections, some overview on existing techniques for web-based development and visual analytics are going to be shown. Then, the techniques themselves and their application to the PRAETORIAN CSA HMI will be shown.

3.2.1 Overview of existing tools and methodologies for web-based CSA visualizations

The field of visualizations for Cyber Situational Awareness has been attracting a big amount of efforts at the research area for the last years. There is no clear, concise and agreed definition, between all the involved actors, on what the cyber space is, and how it should be mapped and represented.

In the following sections, most of the outstanding efforts in the areas related to PRAETORIAN project CSA implementation are shown. Those areas are: (i) web-based development frameworks, as the CSA HMI tool is a web-based tool, (ii) web-based advanced visualization frameworks, as those are the ones upon which the CSA HMI will be constructed, (iii) a survey on visual analytics for the CSA, as those lay foundations for key concepts and ideas in developing a CSA HMI tool.

3.2.1.1 Web-based development frameworks

There have been development frameworks since the inception and wide usage of the World Wide Web (www), since 1995. In this section, most recent trends, besides more widely used frameworks are described, to determine which were the most suited when selecting proper environments for the CSA HMI development.

Nowadays, the usage of web technologies for implementing the front-end of a Cyber situational awareness tool can be considered the number one ranked option due to several reasons. This approach ensures that every existing platform the client has, regardless of its underlying operating system, accesses the system seamlessly.

Therefore, the same development can be used from any kind of client, such as desktop, laptop, smartphone, etc., without installing any extra software, besides the typical web browser, that every system comes with.

Moreover, the costs of development are reduced by a big factor, as every new update of the software is tested once and known to be working properly with every system supporting a given set of browsers.

In addition, security is enhanced by several degrees: instead of installing a piece of software on a given machine, the client isolates the critical elements residing in the back-end to the access of the users.

That is why the main purpose of these sections is centred in surveying existing technologies and frameworks for data analytics in a web browsing approach for front-ends.

For the development of web applications, there are several solutions. All of them fall in one of the following categories:

- In-server rendering: in this approach, information is processed by the server when replying to the client requests and returns to the later an HTML page to visualize.
- Rendering done at the client; in this kind, the server returns to the client web browser the set of assets needed and the procedure to generate the HTML code. The browser generates the HTML page.

There are several languages and frameworks to work with in-server rendering. To point out some of the most used and widely spread:

- Laravel [LAR]: it is a very well-known framework based on the PHP programming language. It follows the Model-View-Controller (MVC) approach, which decomposes the code in segments for the Model, segments for the View and segments for the Controller management. It also provides means for creating middleware and even APIs to interact with front-ends created with JavaScript. Currently it is compatible with the latest release of PHP, version 8.0, despite being in a continuous development.
- Blazor [BLA]: is a technology developed by Microsoft inside the suite ASP.NET. It provides means to develop both web servers and clients relying on the C# programming language and .NET platforms. It has also its own web Sockets implementation for real-time communications avoiding page reloads or pollings.

These systems do provide the benefit of centralizing both the backend and the frontend, easing the data validation processes. It must be pointed out that, to generate active pages, developers must make use of JavaScript based frameworks.

Following this approach, nowadays, the most used technology by far is the rendering done at the client. This is mostly for frontends where content must be dynamic, which happens to be a widely spread requirement. This kind of approaches are also called API-first, as the development processes is led by the definition of the API structure, first, and then follows the development of the contents to be consumed.

In this paradigm, the server can be developed with any programming language, as there are several libraries for communicating all the pieces in real-time and in a decoupled way.

On the other hand, most solutions make use of the JavaScript programming language, as it is the one that is understood by all the web browsers and upon which are developed nearly all of today's most used frameworks. There is a huge variety of them, to name the most outstanding:

- AngularJS [AJS]: It is a very complete technology based on JavaScript, very modular with lots of packages, though many things can be done just with the baseline distribution. Currently is on Long Term Support.
- Angular [ANG]: It is an evolution of AngularJS allowing for the usage of TypeScript, a programming language that extends JavaScript providing strong typing of variables.
- React [REA]: Is is a JavaScript library for developing advanced and dynamic client-side applications. As it happens with Angular, it provides means for developing using TypeScript. Noteworthy, React Native is a feature allowing the development of Android and iOS applications from a given React code.
- Vue.JS [VJS]: Vue.JS is a JavaScript framework that also allows for Typescript. However, Vue.JS only incorporates by default the basic and strictly necessary packages, so it is the decision of the developer to add new packages on demand, therefore providing means for developing lighter applications if needed.
- EmberJS [EMB] is a robust and easy to learn JavaScript framework for building modern web applications. It works on any device.
- MeteorJS [MET] is another widely used framework that enforces simplicity in development and in HMI look & feel, in order to transmit clear and concise messages. Can be used in conjunction with React or Vue.js as it is a full-stack framework.

3.2.1.2 Web-based advanced visualization frameworks (advanced and immersive)

There are several frameworks for advanced visualizations in the context of web development. All those frameworks provide means for developing advanced (and immersive) visualizations and all of them are based on the JavaScript programming language for web client environments.

One of the most outstanding and used APIs is D3.js [D3j], which also, in a very remarkable manner, can be used in conjunction with other frameworks.

There are alternatives for advanced visualizations such as vis.js [VIS], graphViz [GRV] or Gephi [GEP] which do have a lesser quality, less rich API and lower capabilities for implementing advanced visualizations as those needed at the PRAETORIAN CSA HMI.

Here are other relevant graph-generating libraries and APIs for web development, which are mostly based on JavaScript technologies:

- A remarkable example is highCharts [HIG]. It provides means for several kinds of advanced charting and has wrappers for the most popular programming languages (.Net, PHP, Python, R, and Java) as well as iOS and Android, and frameworks like Angular, Vue, and React. It also provides means for charting on maps, with the solution highCharts for maps [HIM]. Free and commercial versions are available.
- Google charts [GOC] is a complete and capability-rich charting framework in JavaScript. Supported by Google, it provides many features for data visualization and new charts creation.
- SyncFusion [SYN] also provides web-based charting APIs and libraries, with free trials and commercial subscriptions. There are APIs for: Blazor, Flutter, ASP.NET Core, ASP.NET MVC, ASP.NET Web Forms, vanilla JavaScript, Angular, React, Vue and jQuery. There are also libraries for Android and iOS.
- Vega-lite [VEG] is a high-level grammar of interactive graphics. It provides a concise, declarative JSON syntax to create an expressive range of visualizations for data analysis and presentation.
- React-vis, a visualization library for JavaScript inside the REACT HMI framework [REV] is an interesting approach for visualization, not very mature, but remarkable for being integrated inside the widely used React framework.
- Graphicsjs [GRJ] is a complete JavaScript OSS framework with a simple API to provide 2D/3D graphics capabilities.
- playCanvas [PLC] is a suite of OSS (Open Source Software) and paid tools for providing WebGL environments for 3D visualizations in the browser. The playCanvas engine is one of the best OSS WebGL 3D graphics engine.
- Three.js [THR] is a high performance 3D modelling library for JavaScript that allows for high-end, resource-aware 3D complex visualizations on browsers.
- Babylon.js [BAB] is powerful web-based 3D engine to develop 3D applications in JavaScript. It provides one of the most powerful and simple Web rendering engines currently existing.
- It is important to state also the relevance of the Kibana [KIB] platform, in conjunction with elastic search [ELA], in order to provide advanced visualizations for cyber situational awareness data.

On the subject of graph-generating immersive solutions, there are several libraries and frameworks available in order to create solutions on web browsers. Please notice that immersive and virtual reality are terms used indiscriminately from now on.

Up until 2017, one of the most outstanding libraries has been WebVR [WVR]. Now it has been deprecated in favour of WebXR Device API [WXR]. One of main reasons that WebVR was superseded by WebXR is that the latter can also support augmented reality (AR) devices and not only virtual reality (VR) ones. Nevertheless, both of them are full of capabilities and support nearly any of the existing VR Headset devices in the market like those developed by Microsoft, HTC, Google or Facebook (prev. Oculus).

These manufacturers usually provide libraries to develop Virtual Reality experiences with other development tools. For example, these libraries can be used in conjunction with a cross-platform game engine development tool, known as Unity [UNI]. Unity is one of the most used game engines in the world, being open-source, and having a vast active community. Other game engines capable of developing with VR libraries are Unreal Engine [UNE], licensed, or OpenGL [OGL], open source but difficult to program as it is in low-level language.

On the other hand, Unity uses its own framework for developing WebGL experiences but it is not as developed for supporting WebVR technologies. Additional libraries provided by the VR Headset manufactures can be integrated into its framework to upgrade the WebGL compiler to include the AR and VR functionalities, but proves very challenging. As such, many Unity developers are moving in favour of WebXR developed code by means of intermediate community-based frameworks, like those provided by De-Panther [PAN]. It is important to notice, that setting up the development platform to work with these community-based solutions can be quite complex. Nevertheless, once obtained, the outcome provides a much smoother solution for Web immersive solutions. The front-end provides all the necessary code to run the VR experience, and in capable browsers and computers, the system can be ready with only 3 clicks required to install the driver of the VR Headset.

On the other hand, there are COTS (Commercial off-the-shelf) devices on the market that provide a cheap and high quality immersive experience to a regular computer user.

It widely opens the spectrum of applications and opportunities for the data visualization community. The immersive experience allows the user to get into the visualization changing dramatically the experience. For instance, allowing a better inspection when there is an overwhelming number of elements on display. It is not only limited to visualization but also to actuation (interaction) with the coupled haptic devices.

3.2.1.3 Visual analytics for CSA

Visual analytics, applied to Cyber Situational Awareness is a very wide area of development. At the moemnet and work is not yet finished, as seen in the previous section.. Despite that, several key ideas and concepts have been consolidated in the literature and in the community, academia and industry.

These concepts are mandatory elements for tools and techniques to provide proper SA, and help analysts and commanders in their everyday work.

Visual Analytics is a wide sub-area of data analytics, which focuses on the usage of the information visualization techniques and methods to boost effective analysis of data by means of the employment of visual and graphical representations. As stated by Thomas et al., "Visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces" [Tho05]. There is a very relevant work done by A. Kott et al. "Cyber Defense and Situational Awareness", which states that: "Visual analytics focuses on analytical reasoning using interactive visualizations" [Kot15]. That is one of the cornerstones for visual analytics particularized to Cyber Situational Awareness (CSA), interactivity. Several authors enforce this interactivity element for visual analytics to be effective in the context of CSA. The user must be able to interact with the data but, more than this, must be able to go several steps further.

As stated in that relevant work [Kei08], there are huge differences in terms of military doctrine, required resources, threat characterization and detection, pace of events, etc. between the physical or kinetic domain and the cyber domain. With regards to data visualization and analysis, those differences aggravate for the visualization and representation of information. Moreover, there is no agreement on how to represent the cyber space and how it should look like.

As such, for the physical domain, the common operational picture and counter insurgency operations require network analysis, and dependency graphs are mostly used for the visualization. In the cyber domain, attack graphs, dependency graphs and cyber terrain are used.

Regarding representation, for the physical domain, it tends to have a commonly accepted, widely used organization paradigm – the physical terrain of the battlefield (e.g. a map). For the cyber domain, no map-like common reference has emerged.

As expressed by the famous Shneiderman's information seeking mantra: "Overview first, zoom/filter, details on demand" [Shn96]. Keim extended and particularized to the visual analytics domain with his own mantra: "Analyze first, show the important, zoom/filter, analyze further, details on demand" [Kei08].

Interactivity and iterativity in the process are clearly necessary. Every tool must be designed with those concepts in mind in order to provide proper CSA, and help analysts and commanders, each one with its own needs and responsibilities, in finding out what is going on in the cyber domain, how it affects a given mission, and how to tackle any occurring issues.

Interaction enables the user to fully traverse all the data paths and interrelations by means of exploring the data, try out and check hypotheses, drill into data (zoom in and zoom out), gain insight, discover hidden or non-evident patterns and collect knowledge.

Another very relevant dimension in this process is time. As shown by Keim et al. [Kei08], the pace and rhythm of operations is very different from the physical domain to the cyber domain. Considering that in the physical world, events occur at a rate that may (or may not) be processed by a human operator.

In the cyber domain, it is impossible, as events occur at, at least, two orders of magnitude above that of the physical world. To cope with this, as stated by [Hee12], ‘’To be most effective, visual analytics tools must support the fluent and flexible use of visualizations at rates resonant with the pace of human thought’’.

For this purpose, several efforts such as a taxonomy of interactive dynamics that contribute to successful analytic dialogues was proposed in the same work. The approach is similar to other efforts and can be summarized as:

- Data & view specification: (1) Visualize data by choosing visual encodings, (2) Filter out data to focus on relevant items, (3) Sort items to expose patterns, (4) Derive values or models from source data;
- View manipulation: (1) Select items to highlight, filter or manipulate them, (2) Navigate to examine high-level patterns and low-level detail, (3) Coordinate views for linked, multi-dimensional exploration, (4) Organize multiple windows and workspaces;
- Process & provenance: (1) Record analysis histories for post-processing, review and sharing, (2) Annotate patterns to document findings, (3) Share views and annotations to enable collaboration, (4) Guide users through analysis tasks or stories.

Despite that, due to the huge amounts of data and the pace generated in the cyber domain, automatic processing of information and aiding tools are mandatory to help the operator or analyst. Those range from Intrusion Detection Systems (IDS) or Intrusion Prevention Systems (IPS), Security Information and Event Management Systems (SIEMS) or event machine learning based tools and systems for threat hunting.

Therefore, visual analytics should actively engage the users in an exploratory, continuously backtracking hypothesis-checking process of discovery. As stated in [Jia22], there will be a discourse and interaction between the analyst and the data with three main goals: (i) situation assessment by understanding current situation, (ii) forecasting in order to estimate future capabilities, threats, vulnerabilities and opportunities; and (iii) projection or planning in order to manage possible scenarios and prepare reactions to potential events.

Another important aspect in visual analytics for cyber space is the data overload and cognitive overwhelm that the operator can experience. Data overload can be defined as ‘’a condition where a practitioner, supported by artefacts and other practitioners finds it extremely challenging to focus in on, assemble, and synthesize the significant subset of data for the problem context into a coherent situation assessment, where the subset is a small portion of a vast data field’’ [Pat01]. As defined by Bowden [Bow05], insight is ‘’thought to arise when a solver breaks free of unwarranted assumptions, or forms novel, task-related connections between existing concepts or skills’’.

Obviously visual analytics in the scope of cyber situational awareness is a complex and multidisciplinary domain that encompasses experts from several disparate areas such as psychology, cognitive science, social science areas, computer engineering, etc. Visual analytics is a highly interdisciplinary field of research [Tho09]. There is a comprehensive and complete survey on the cognitive foundations of visual analytics done by Greitzer et al. [Gre11]. At the end, visual analytics is meant to facilitate high-quality analysis with limited user's time, combing huge amounts of disparate and heterogeneous data and in harsh cognitive conditions.

With regards to those psychological and cognitive aspects of visualization, pre-attentive visual features is an approach to catch the attention at a glimpse [Hea09]. Our brains recognize patterns some orders of magnitude quicker than it understands situations. Moreover, the Gestalt laws of organization describe how people perceive visual components as organized patterns or wholes, instead of many different parts [Wer38].

Other key factors are dimensionality reduction and complexity decomposition, assuming inherent nonlinearities and couplings.

At the end, tools and visualization techniques are needed to help in the iterative process: find one pattern with that information, make a model, which will lead to a new insight, then make us refine or transform the model…, until we find what is going on.

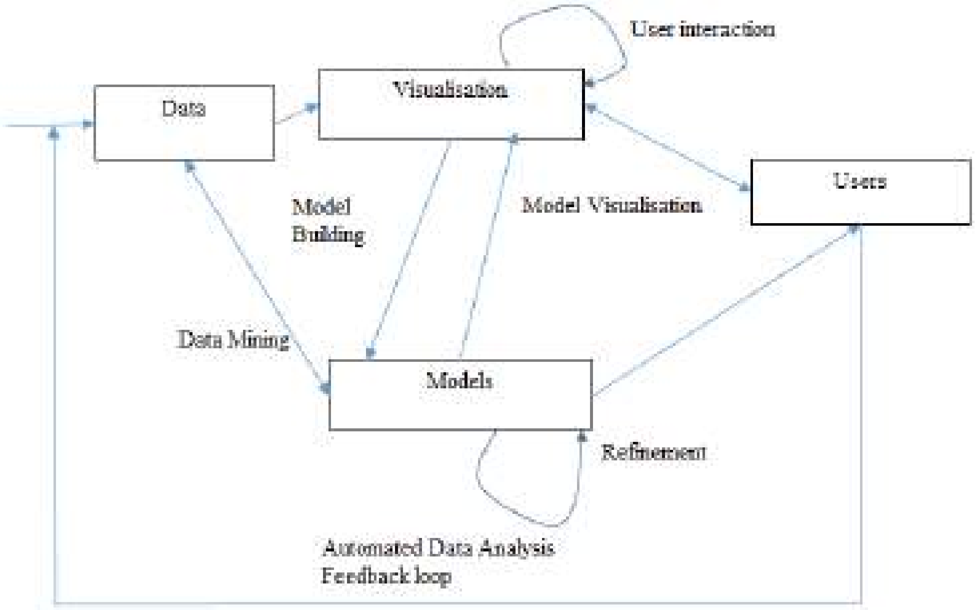


Figure 7 – Visual Analytics process by Keim et al. (2010)

3.2.2 HMI General Overview

To access information visualization in the system, the user must log in and select a given panel or create a new one, if no panels are currently in the system. This process is shown in the “Additional CSA HMI functionalities” section.

It is important to highlight that the CSA HMI system makes a clear distinction between queries and visualizations. By means of the queries drop down menu, the user selects which queries he is interested in viewing. By selecting the visualization, the user selects which view to apply to a given query.

To select between the different visualizations, the user must click on the visualization select control, which is in the top navigation bar, as seen in the following figure.

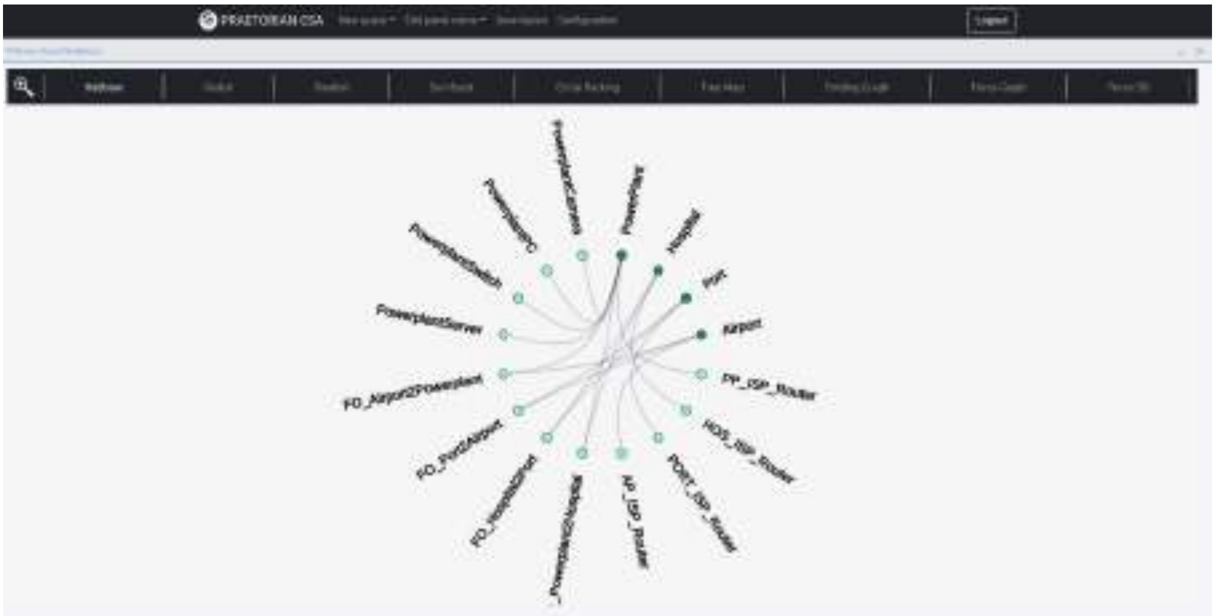


Figure 8 – Types of visualizations selection

If the panel width is not large enough, then the user will see this capability as follows, the functionality remaining the same.

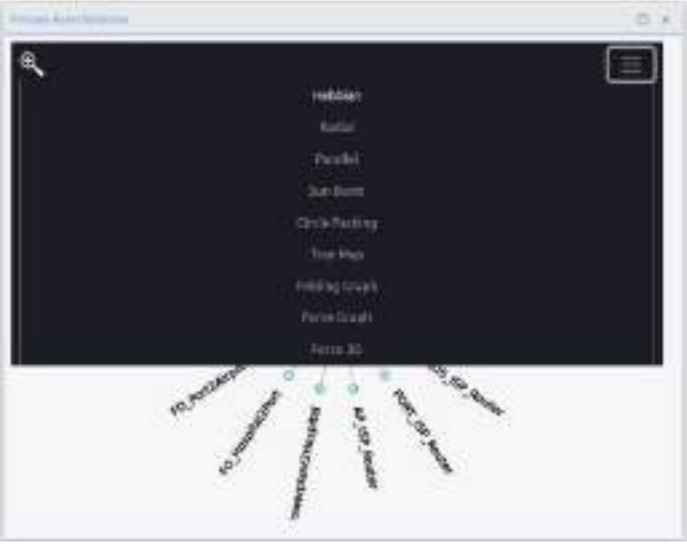


Figure 9 – Types of visualizations selection II

It must be stated that, once we set up on screen a query and its corresponding view, it will change dynamically every time related data on the IOP is modified.

On the other hand, once the user selects a query, he/she can switch from one view to the other, with the above-mentioned procedure, applying to the results of the given query those visualizations that are switched. We will see in the next sections the visualizations that the CSA HMI provides.

To emphasize that concept, the following screenshot depicts the same data subset (the same query at a given moment) for three different visualizations.

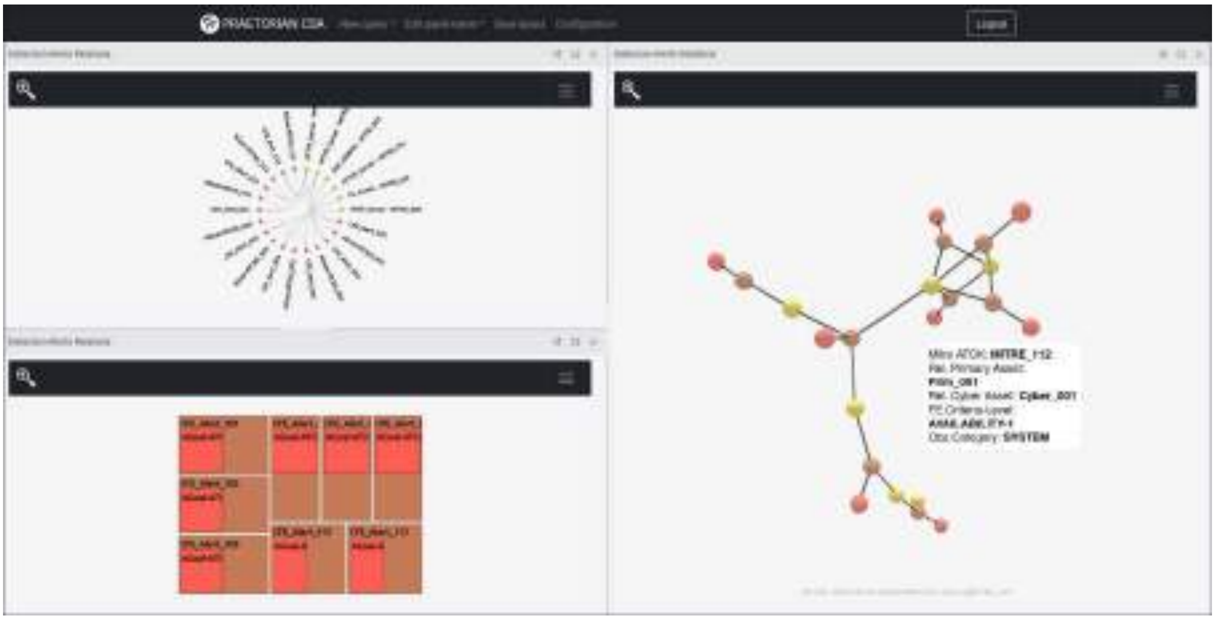


Figure 10 – Same data with 3 different kinds of visualization

3.2.3 Advanced graphs and diagrams

These views correspond to 2D advanced graphs and diagrams selected at the CSA HMI system to enhance the user's views. Those are consonant with the following visualizations:

- Hebbian visualization: based on Hebbian dynamics to represent the relationship between several variables.
- Radial visualization: which is an extension of the previous Hebbian dynamics views including branching highlights.
- Parallel visualizations: which show the relations between several dimensions in a column approach.

Then, we have another group of three views, which provides means for focusing on the representation of clusterings of data. Those are the following views:

- Sun burst: shows the structure of a set of grouped items in a circular fashion, allowing to hide / showelements.
- Circle Packing: groups data in circular clusters, also providing animation to traverse hierarchies.
- Tree map: provides a Hilbert map-like representation.

Next, there is a group of visualizations primarily oriented to a graph view of the data. Those are:

- Folding graph: graph view of the data and its relations, in a more static way.
- Force graph: dynamically reallocated force-equilibrium graph representation.

It is remarkable to show that there is a colour code in all the visualizations. As a consequence, all threats are shown with the same colour, all vulnerabilities with the corresponding one and so on per each of the categories of the data, which are:

- Primary Assets,
- Cyber Assets,
- Alarms,
- Detections,
- Hosts,
- Attack goals,
- Etc.

The following paragraphs shows some examples of the above-described visualizations.

For Hebbian dynamics, oriented to show the relationships among elements at a glimpse and in a very interactive approach, we have the following view.

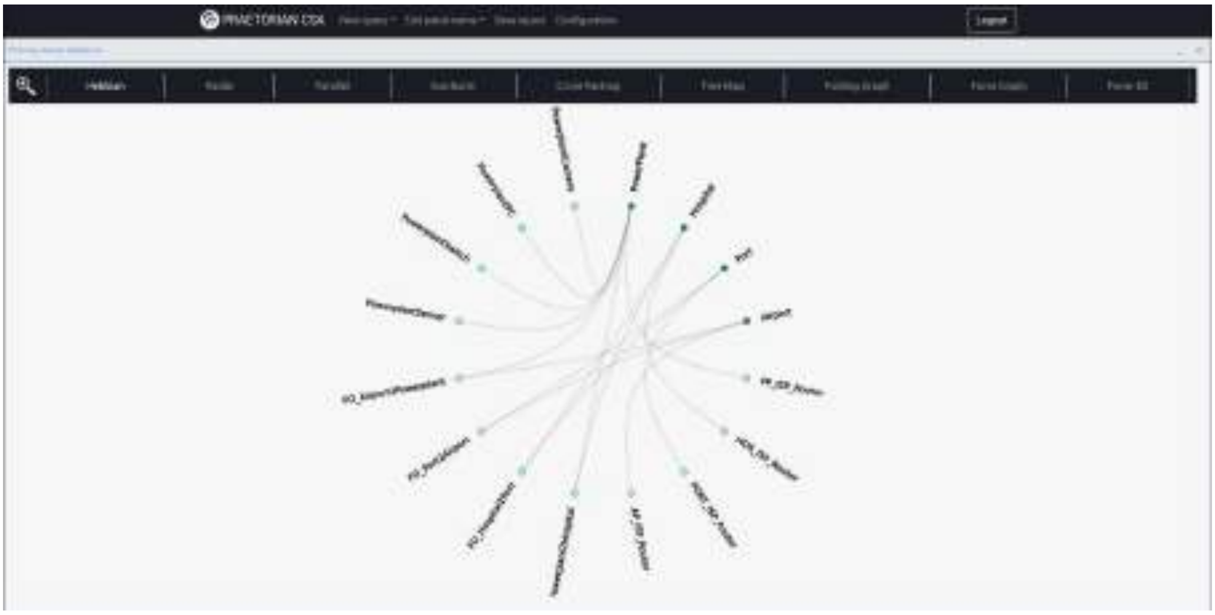


Figure 11 – Hebbian dynamics I

The interactivity of this view can be seen in the next figure. Once we locate the mouse over one item, the system shows automatically the other items which are related to it, highlighting the connections among them.

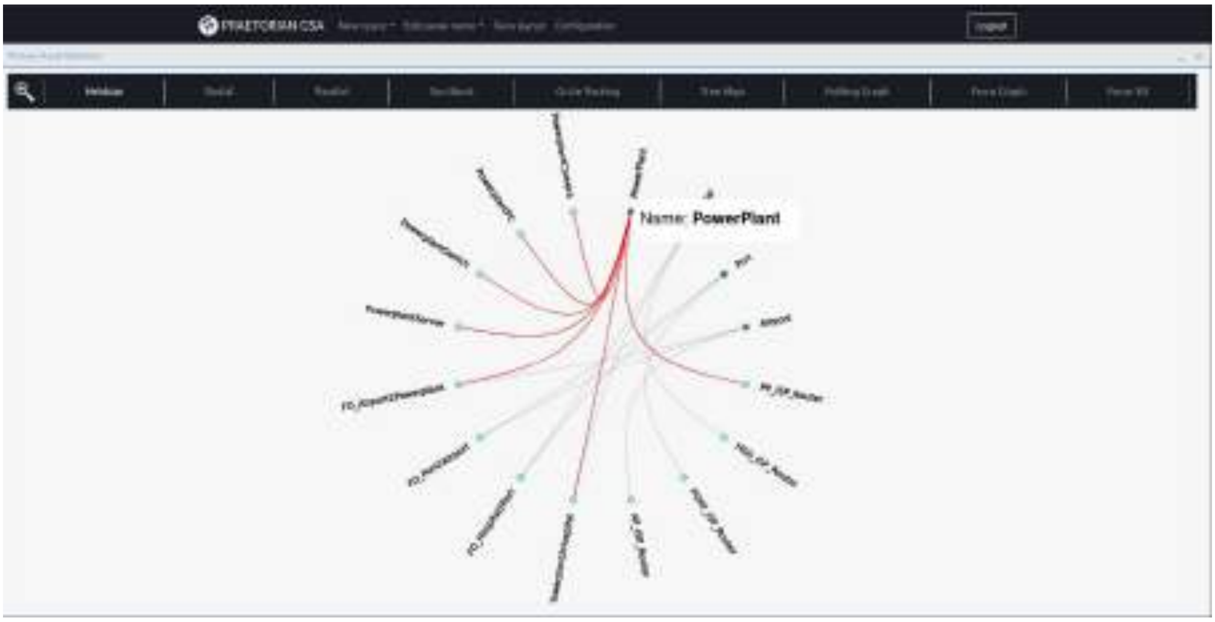
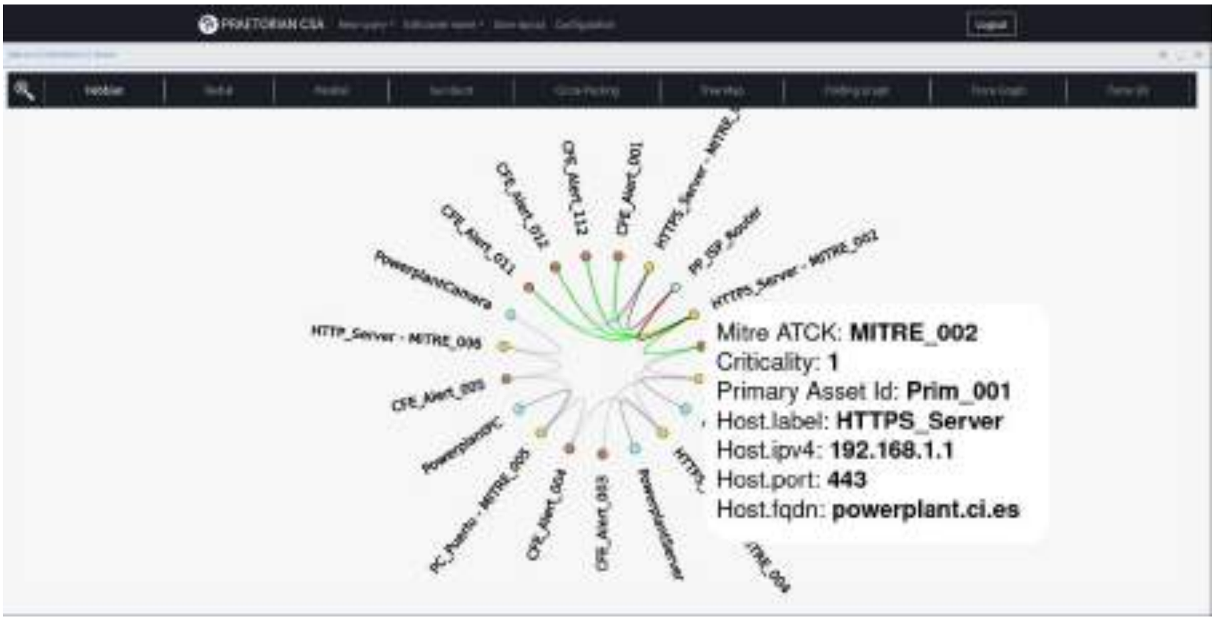


Figure 12 – Hebbian dynamics II

The colour codes, do not apply only to the nodes, but also to the links and relationships among them, as can be seen in the following figure.



Mitre ATCK: **MITRE_002**
Criticality: **1**
Primary Asset Id: **Prim_001**
Host.label: **HTTPS_Server**
Host.ipv4: **192.168.1.1**
Host.port: **443**
Host.fqdn: **powerplant.ci.es**

Figure 13 – Hebbian dynamics III

The same data set (primary assets relations) can be shown with a radial dendogram. It is represented in the next figure.

Figure 14 – Radial I

As stated above, the radial view shows all the branching processes in a given tree of data, or at least shows the elements that are included in a sequence. It is remarkable that the colour codes also apply in this case.

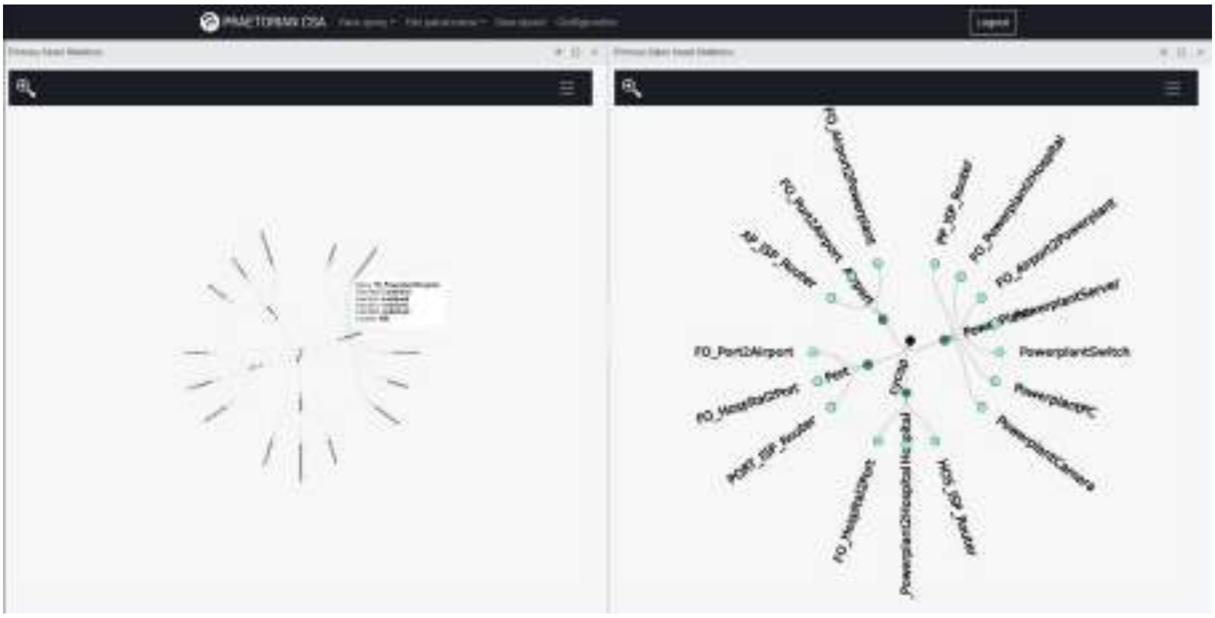


Figure 15 – Radial II

The next visualization in this group is 'Parallel'. It provides the capability to relate different elements in a column view, showing the relations between elements of the different dimensions.

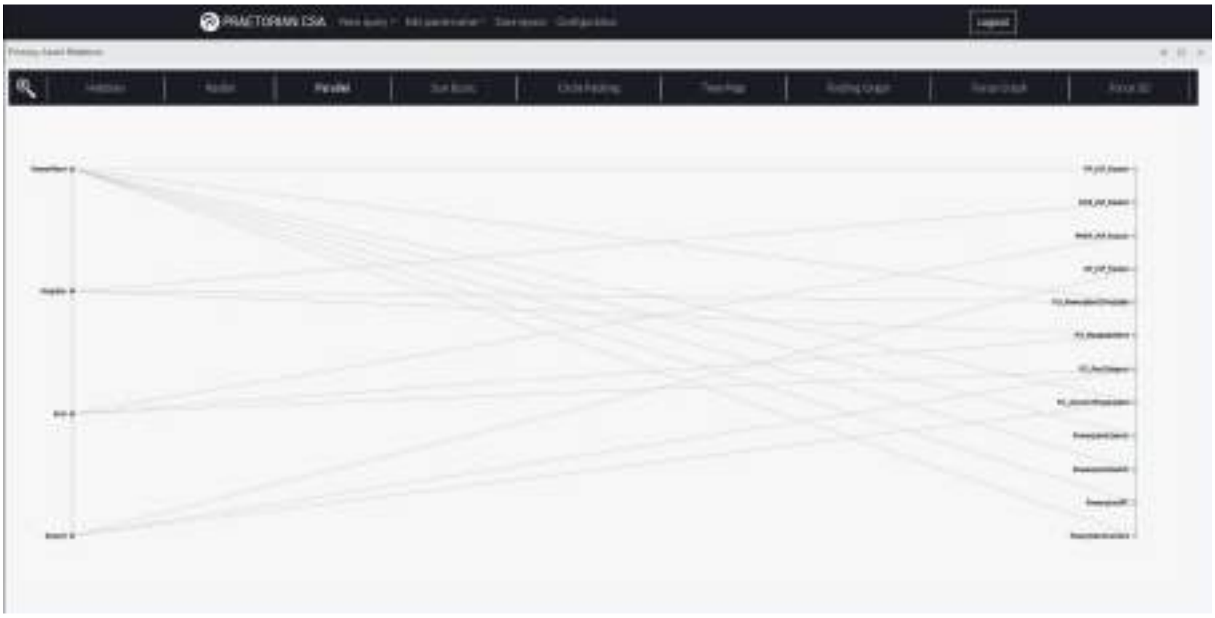


Figure 16 – Parallel coordinates I

This visualization enforces interactivity by of several mechanisms. First, the user can see which elements relate to which, just by leaving the mouse on the given element, as illustrated below.

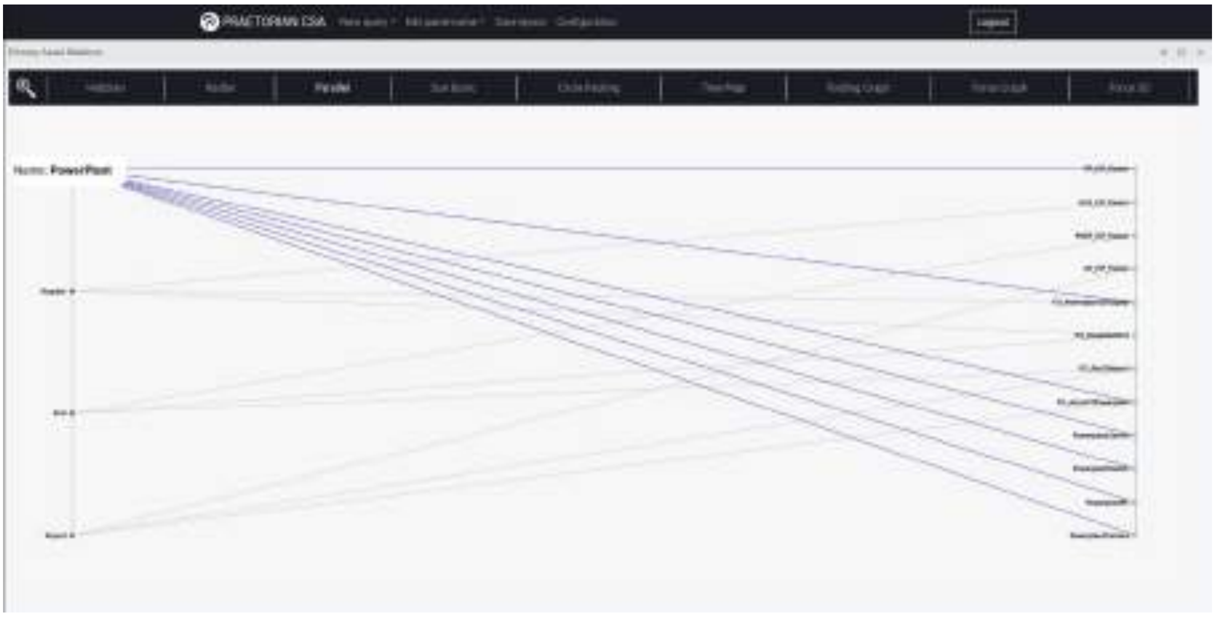


Figure 17 – Parallel coordinates II

On the other hand, the user can filter which elements are interesting and should be shown in the interface by selecting them on the vertical bars.

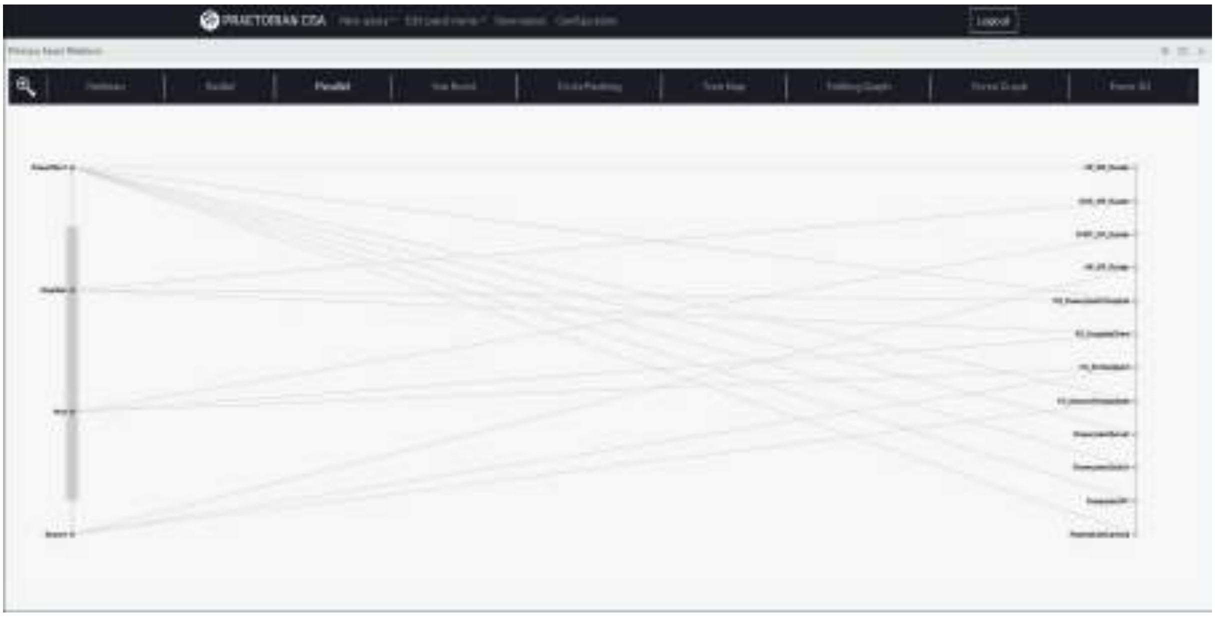


Figure 18 – Parallel coordinates III

This view can relate several dimensions between them, in a M-to-N-to-P, and so on, relation, as shown in the next figure.

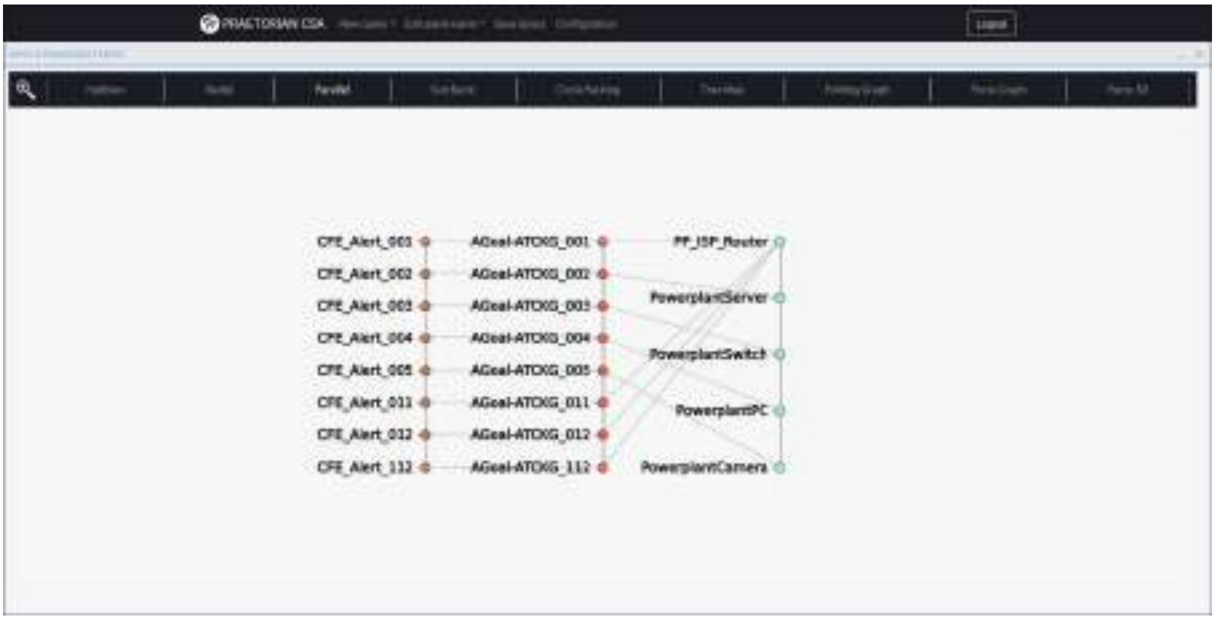


Figure 19 – Parallel coordinates IV

The next group of visualizations tries to represent the groupings of information in a hierarchical and/or clustered view.

For the sunburst visualization, a set of data, organized in categories and subcategories, is shown in a clockwise radial aspect, with the inner circles including the outer ones and using homogenized colour codes per each of the categories. This way, the user can see all the elements at level N, each of them including several elements at level N+1, which correspondingly include each element at level N+2 and so on.

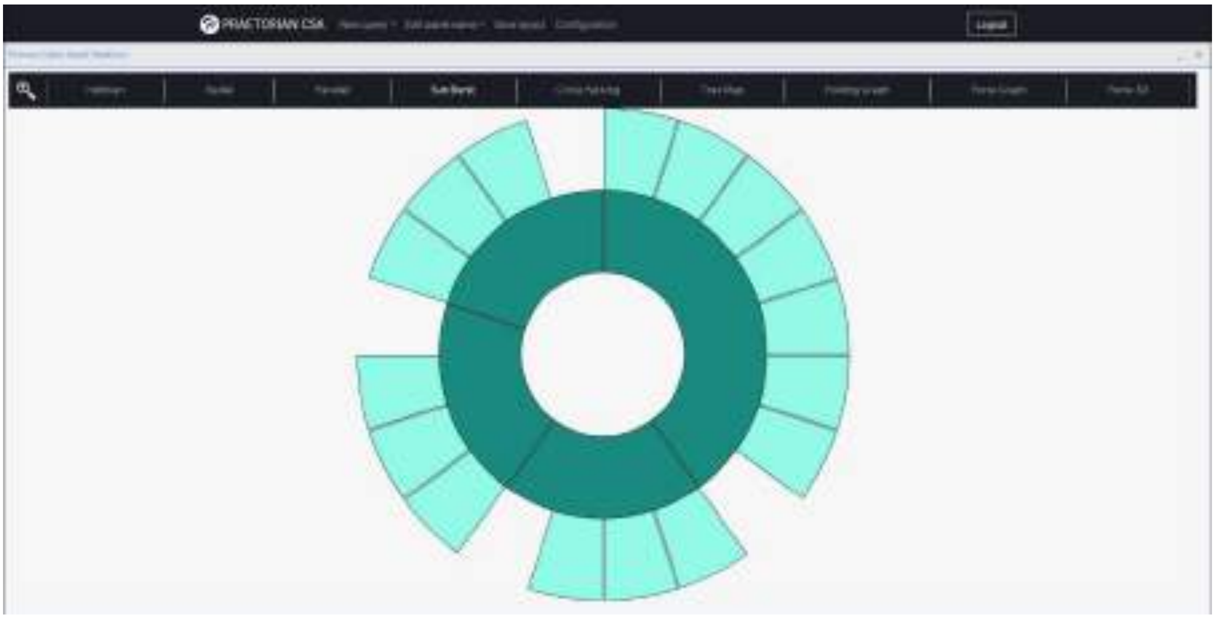


Figure 20 – Sunburst view I

Interactivity is enforced by letting the user click on a specific information to group/ungroup sets of data and particularize them into a given feature of interest. Once the relevant pattern is detected, the user can backtrack, following the theoretical aspects seen in precedent sections, for the information seeking mantras.

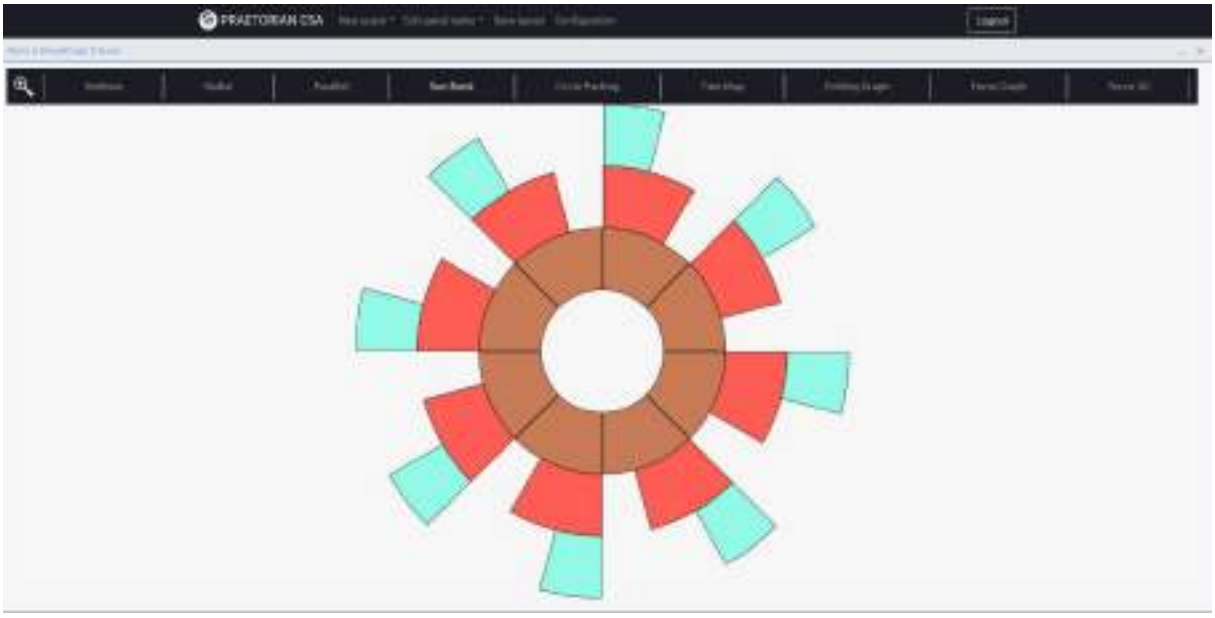
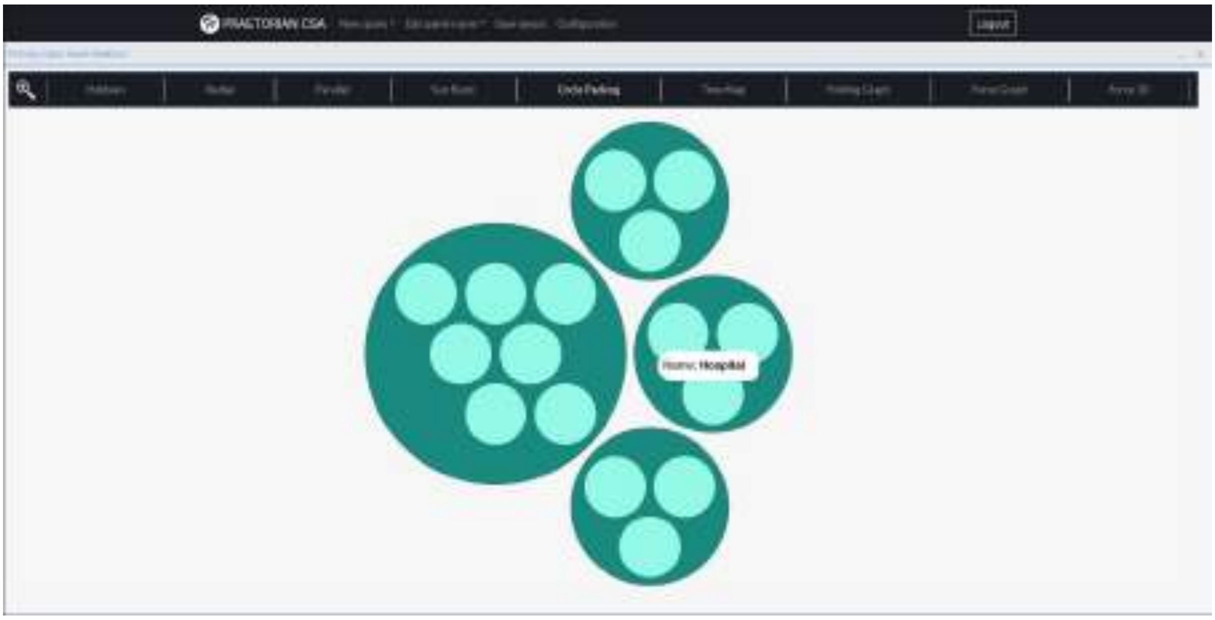


Figure 21 – Sunburst view II

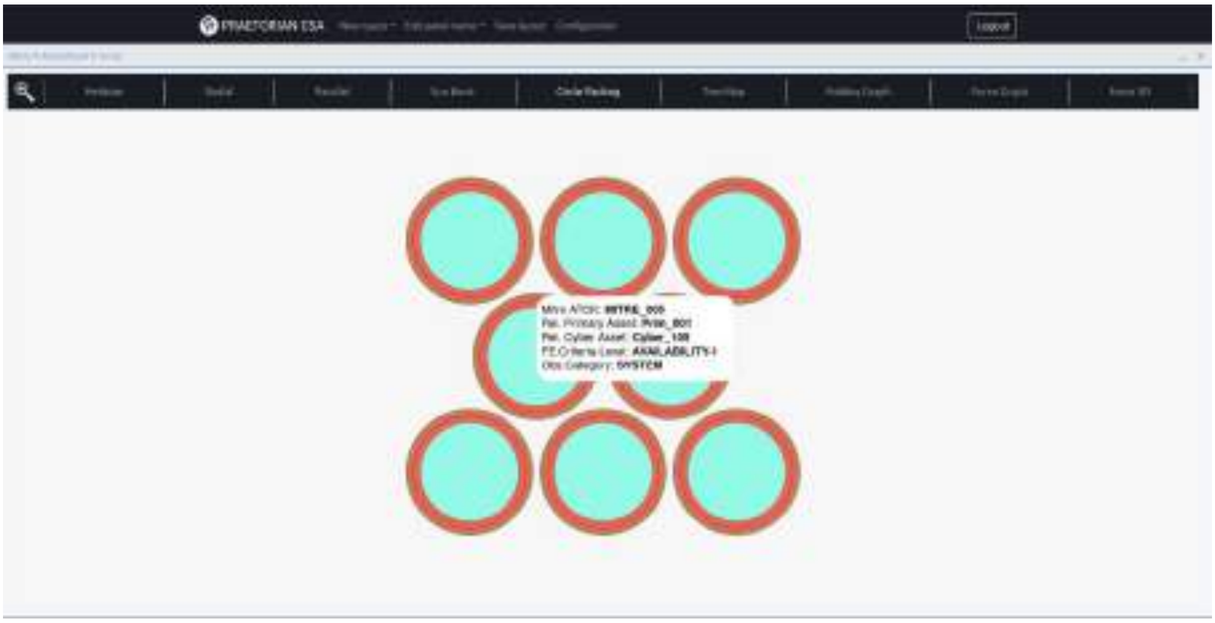
For the circle packing visualization, the approach is quite similar to the previous one, grouping the data in circular sets where the user can get into or out of, just by clicking on the parent element.



Name: Hospital

Figure 22 – Circle Packing I

Depending on the number and distribution of the inner grouped elements, we can see them as circles or even as outer/inner rings as shown in the next figure.



Mitre ATCK: **MITRE_005**
Rel. Primary Asset: **Prim_001**
Rel. Cyber Asset: **Cyber_108**
F.E.Criteria Level: **AVAILABILITY-I**
Obs Category: **SYSTEM**

Figure 23 – Circle Packing II

In the following view, we can see the grouping of several dimensions.

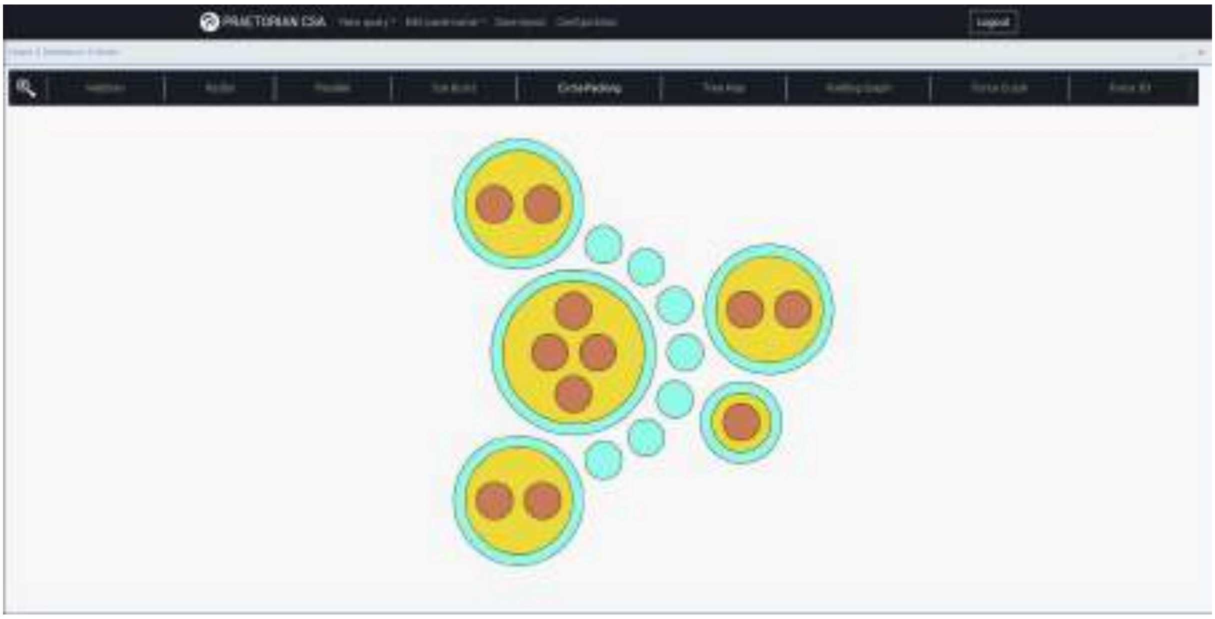


Figure 24 – Circle Packing III

The same grouping of the past figure, applied to a tree map can be seen in the next capture where the colour codes and their representation are properly shown.

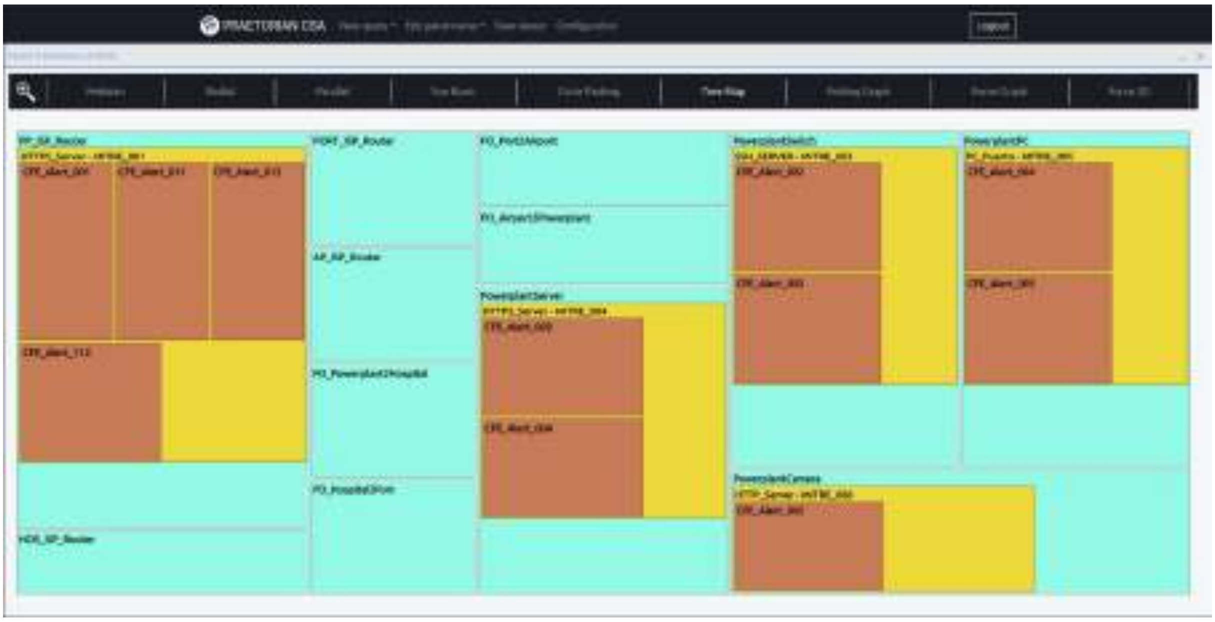


Figure 25 – Tree Map I

It can be seen in the previous representation that the groupings cluster information in a hierarchical way, and that the system tries to represent everything in a stacked manner. Colour codes enhance the differences between categories, so users can see at a glimpse which items include more elements.

On the other hand, a simpler view that only corresponds to the relation between primary and cyber assets (thus colour codes do not change) are shown in a tree map, in the following snapshot.

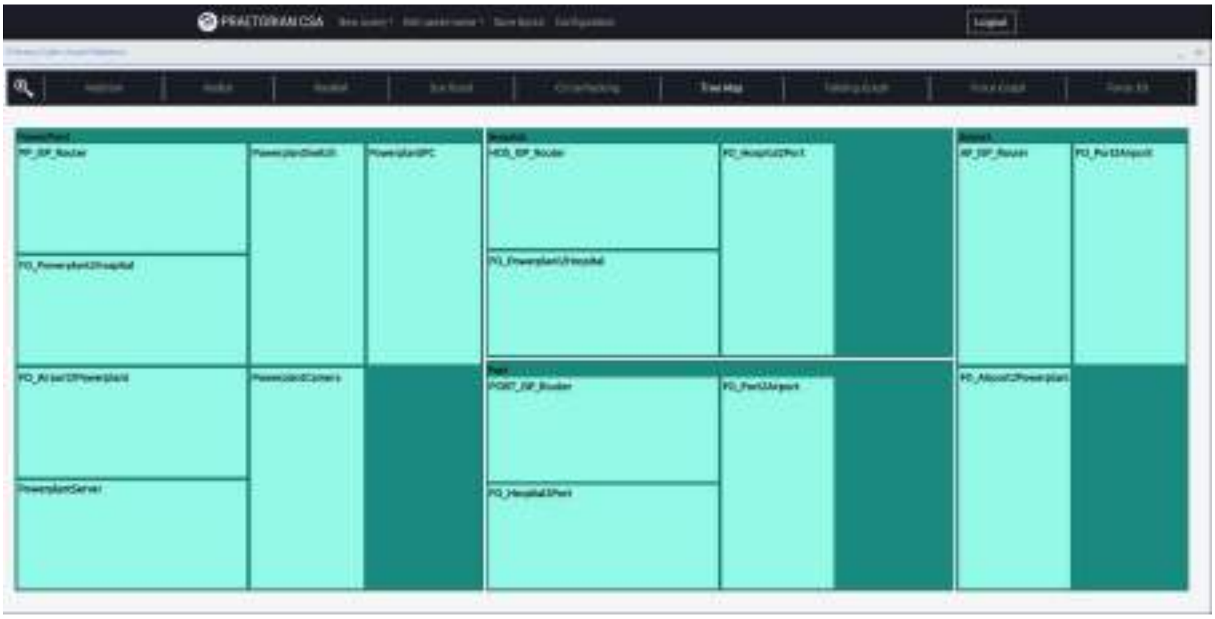
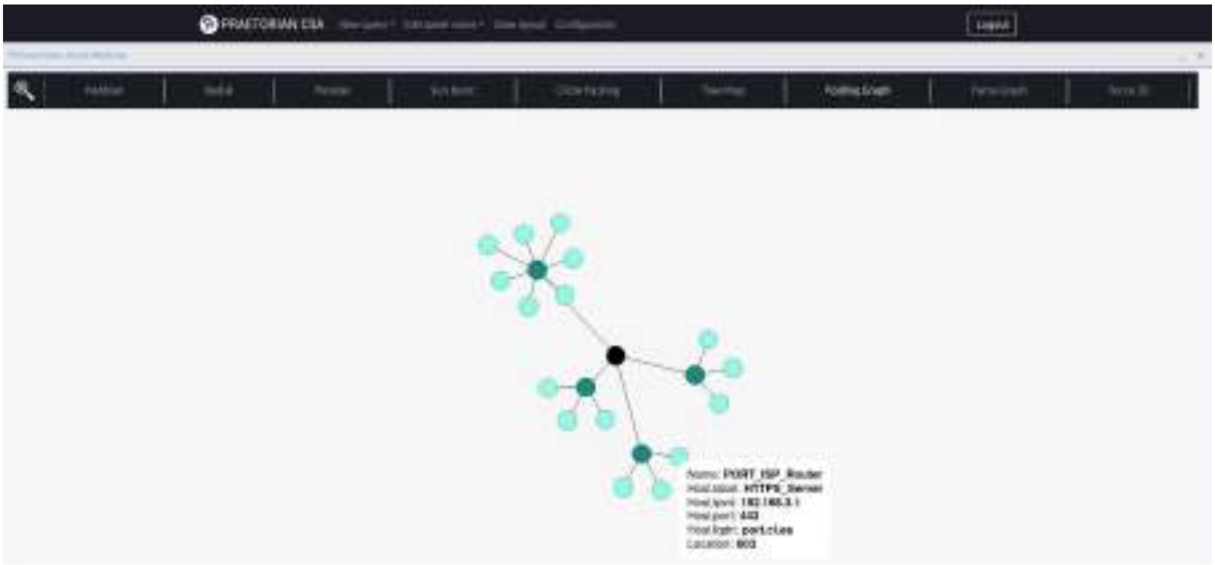


Figure 26 – Tree Map II

The next representation belongs to a new group of views, which is more graph oriented. In those, selected elements for a given query will be shown in a planar graph. On the previously shown views, the data is shown based on a regular approach, in the following cases, it is shown in a more dynamic graph perspective, with automatic arrangement of elements based on force dynamics.

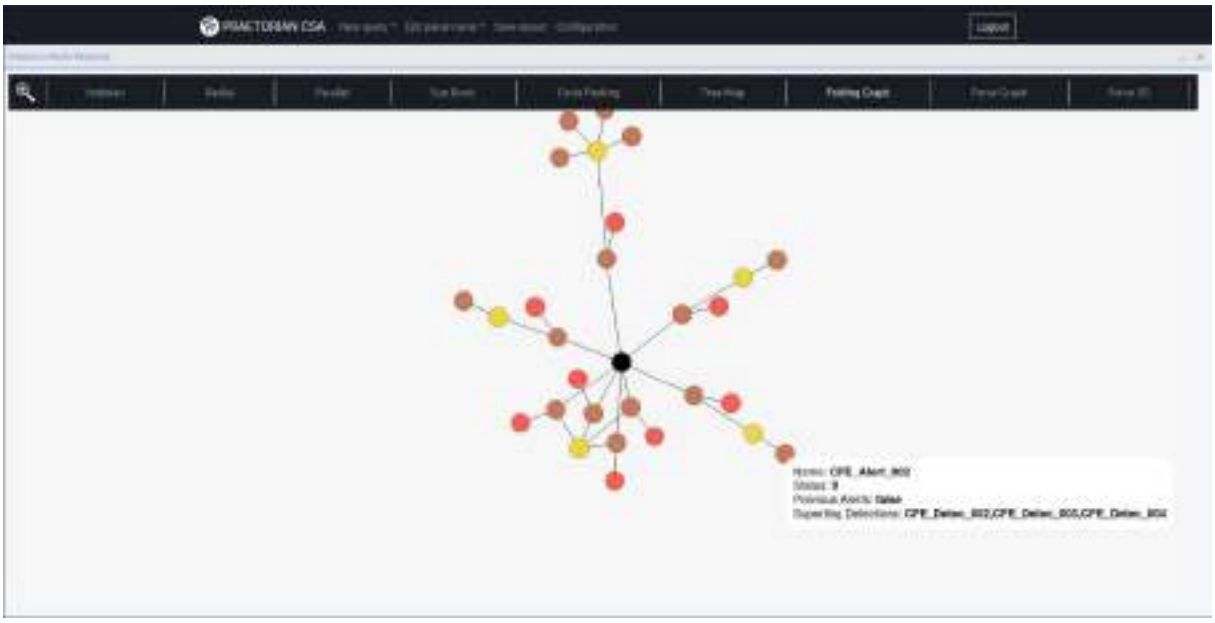
The next figure shows a folding graph.



Name: PORT_ISP_Router
Host.issue: HTTPS_Server
Host.ipv4: 192.168.3.1
Host.port: 443
Host.login: port.cl.es
Location: 803

Figure 27 – Folding Graph I

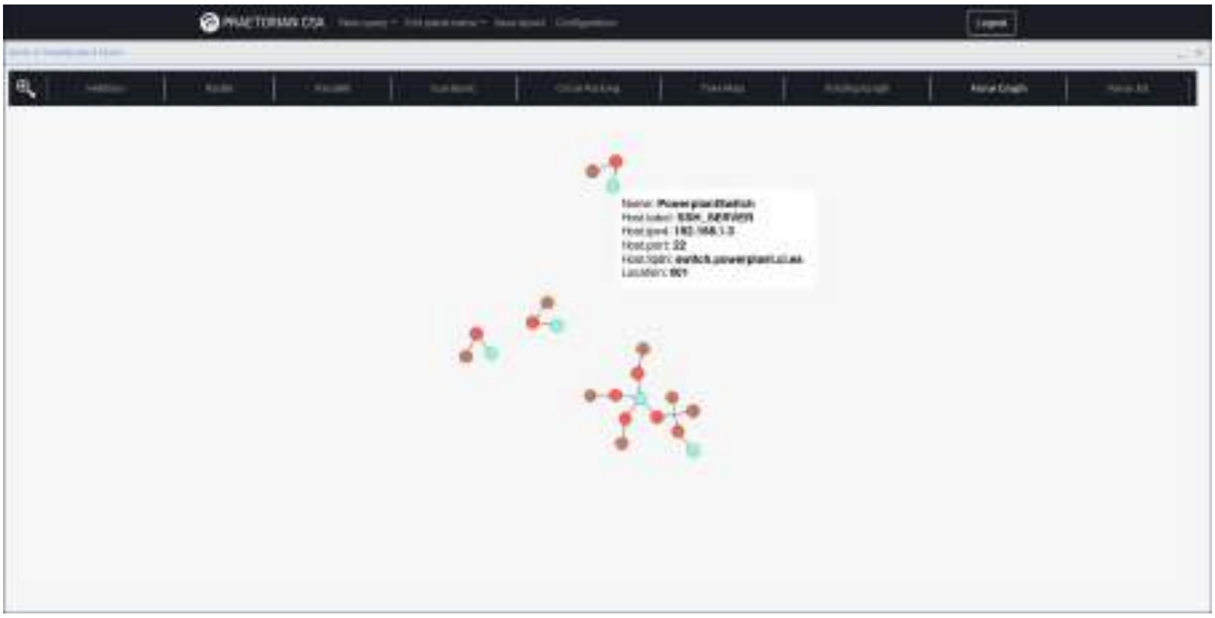
Again, we are seeing information complemented by colour codes. The hierarchy is captured. Central nodes are at level 1 (darker green). They connect to nodes of the same kind (lighter green at level 2). Finally, those are connected to lighter green nodes of the same type.



Name: CPE_Alert_802
Status: 9
Previous Alerts: false
Supporting Detections: CPE_Detec_802,CPE_Detec_805,CPE_Detec_804

Figure 28 – Folding Graph II

A force graph is a more dynamic view, as shown below.



Name: PowerplantSwitch
Host label: SSH_SERVER
Host ipv4: 192.168.1.3
Host port: 22
Host fqdn: switch.powerplant.cl.es
Location: 001

Figure 29 – Force Graph I

A force graph is based on a vector-field dynamics of forces among the different nodes in order to reallocate them and occupy as much on-screen space as possible. Therefore, it is highly interactive. Every node can be dragged and dropped by the user. The view and setup will be relocated. This can be seen in the following figure.

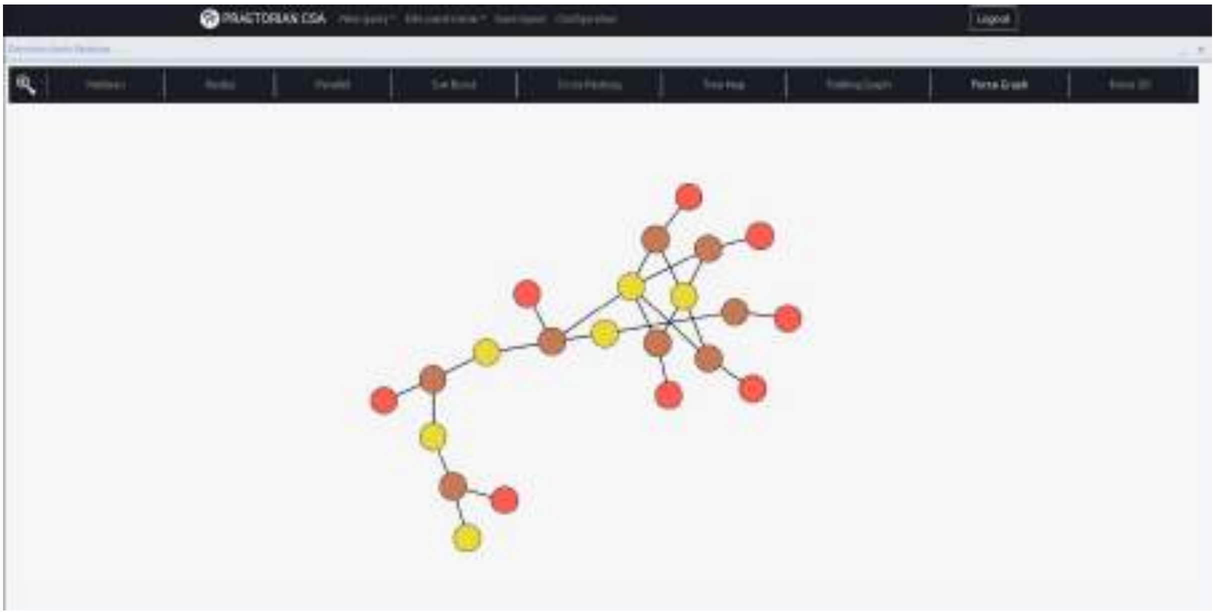


Figure 30 – Force Graph II

If we move the image, the graph can be reallocated as follows.

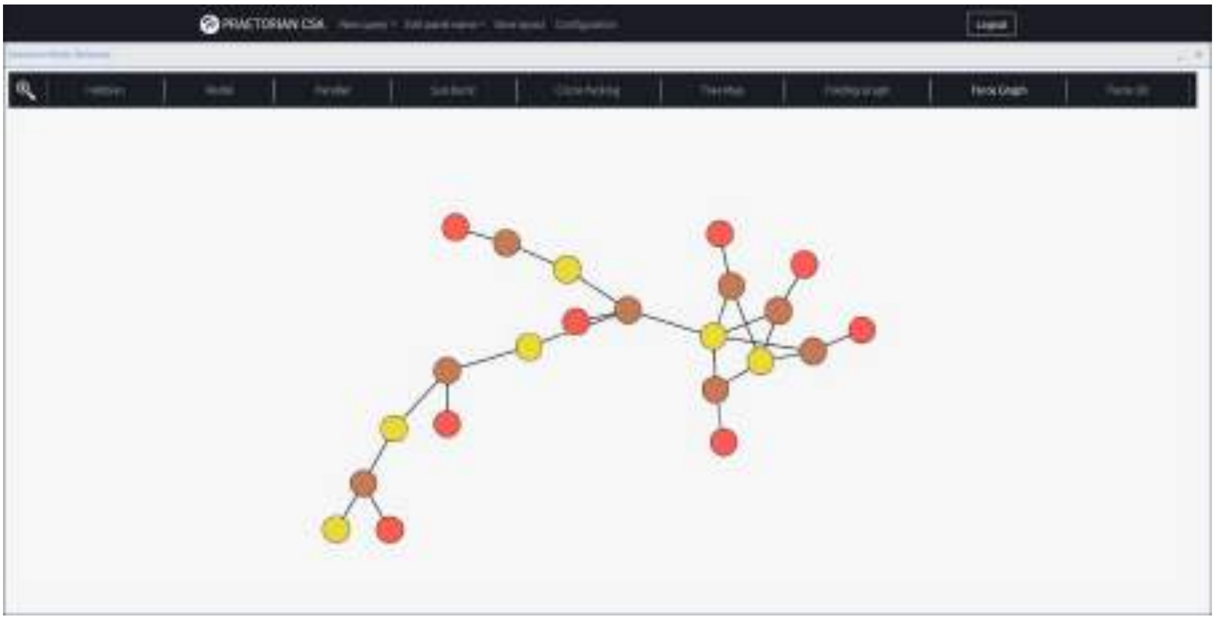


Figure 31 – Force Graph III

It is the same data representation. Force graphs allow dragging some nodes around, and letting the system recalculate the location of the rest.

3.2.4 3D on-screen interactive visualizations

3D on-screen interactive views allow the user to visualize data sets resulting from given queries in a 3D perspective, allowing rotating, reversing and zooming in/out of a given perspective. This can be especially useful when the user is confronted with data sets with huge amounts of elements. We will show some of its features. Being 3D, things can be rotated on any axis (x, y, z).

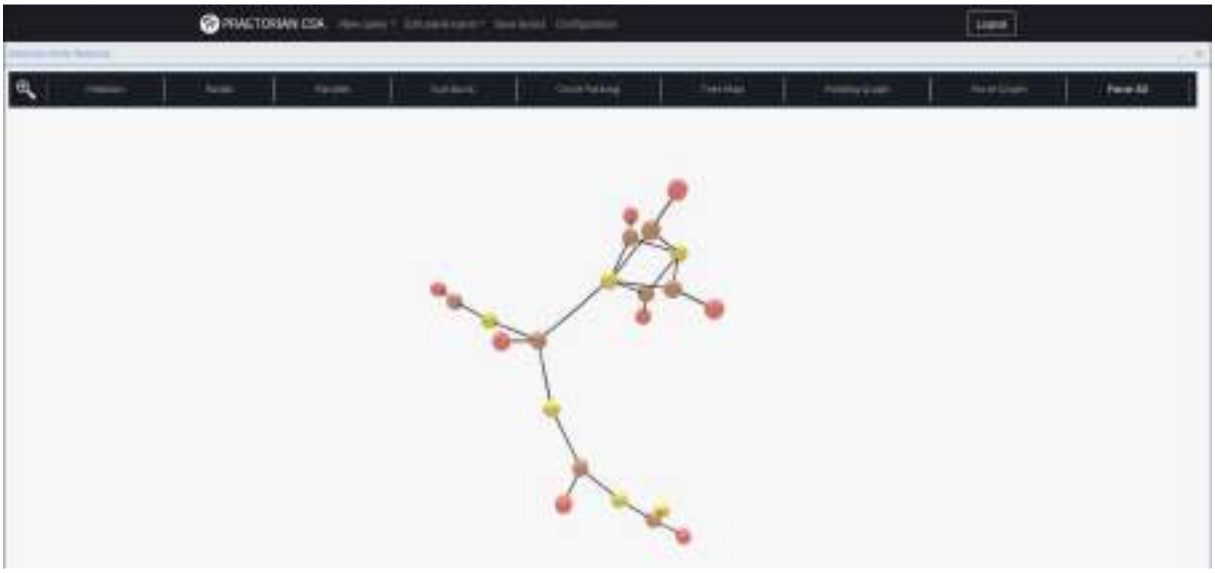


Figure 32 – Force 3D I

So, starting from the previous view, here is an example of a rotation with regards to the y axis:

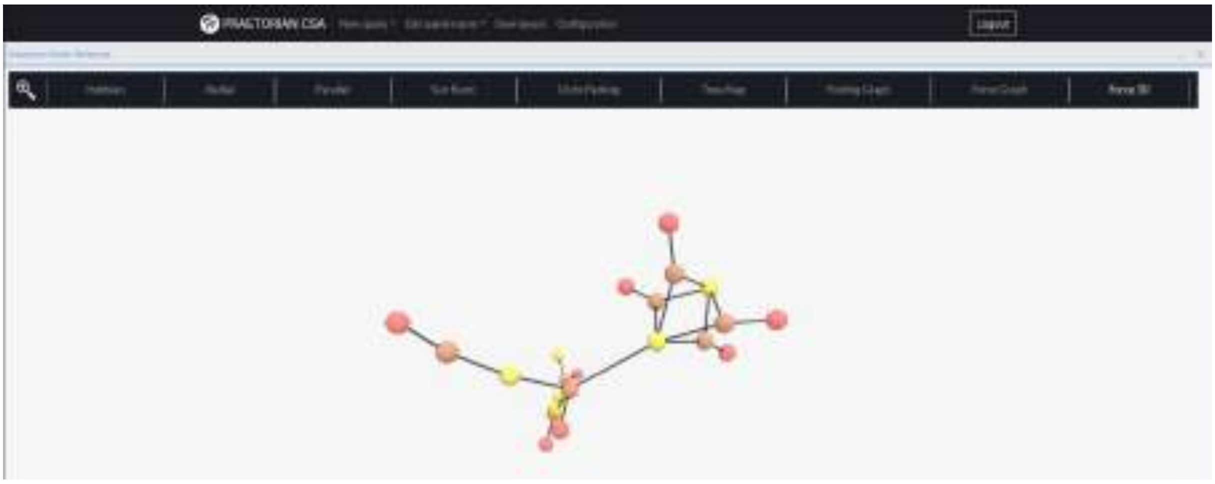


Figure 33 – Force 3D – y axis rotation

Or if we rotate with regards to the x axis:

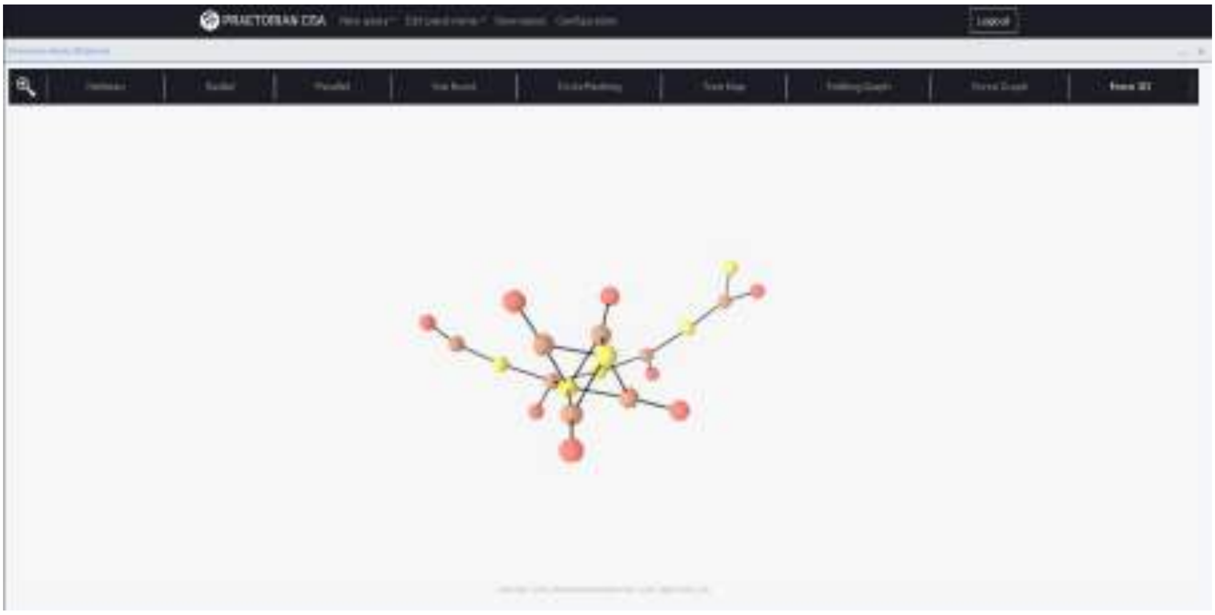


Figure 34 – Force 3D – x axis rotation

We can zoom in:

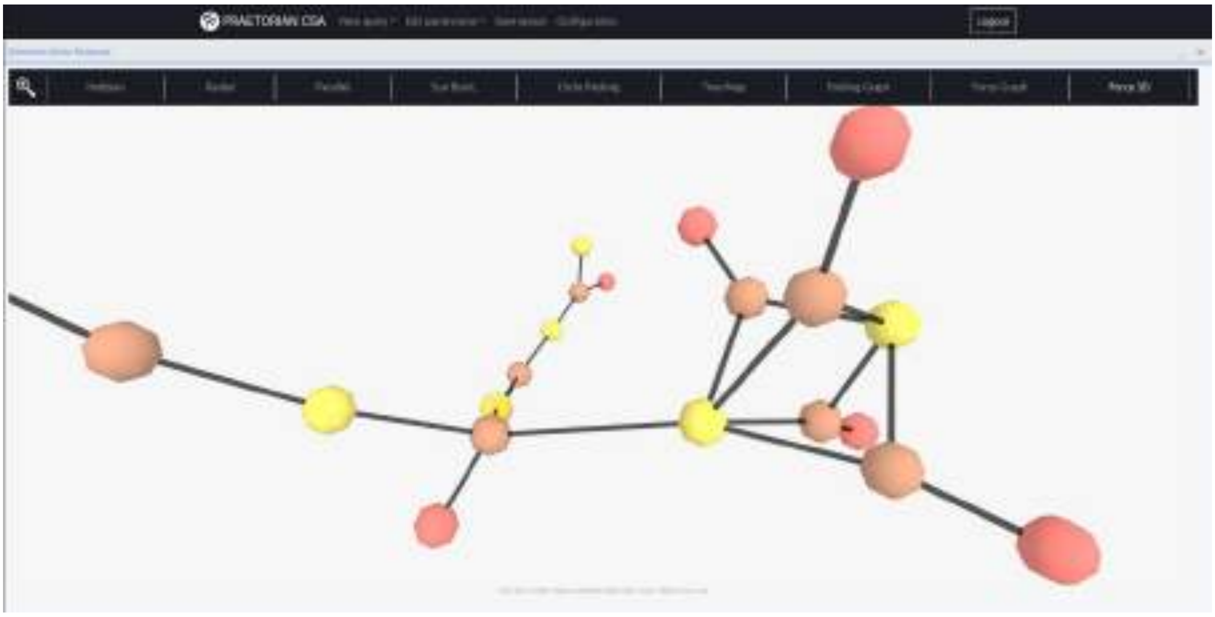


Figure 35 – Force 3D – zoom in

Or even zoom out:

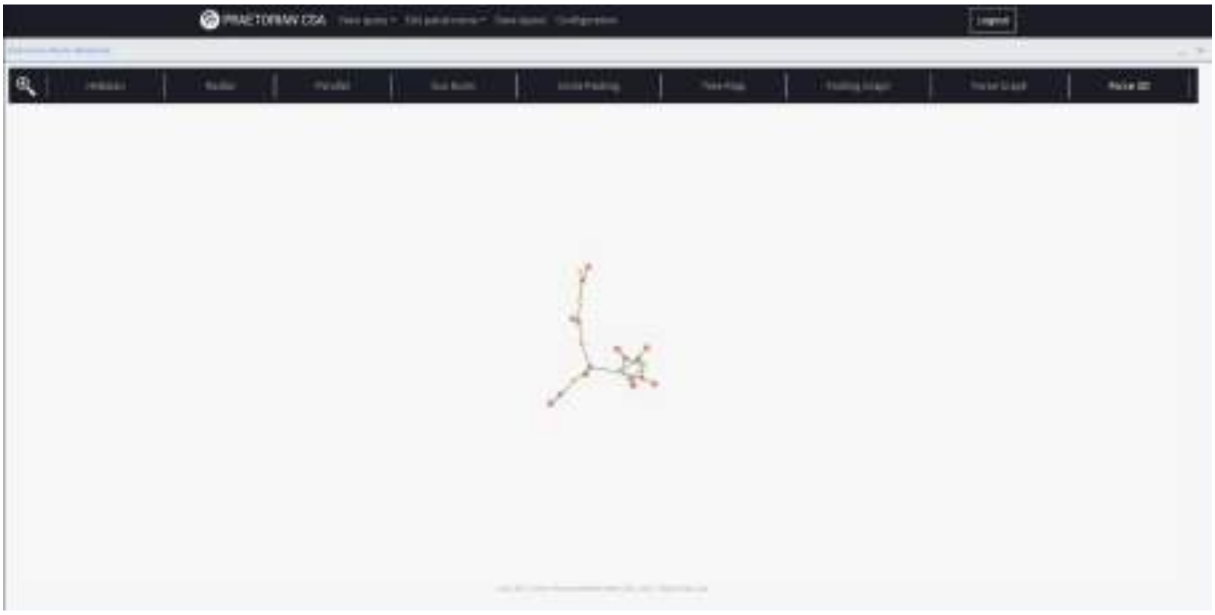


Figure 36 – Force 3D – zoom out

As will be shown in the “Additional CSA HMI functionalities” section, several customizations can be shown at the same time for the CSA HMI, depending on the criteria the user has, and benefiting from the system’s docking panel capabilities. Therefore, any combination of views can be set up for a given CSA instance, dynamically adding or removing any kind of views and arranging them in the layout at will.

Despite being explained in sections to come, in the following one, some snapshots of this are shown.

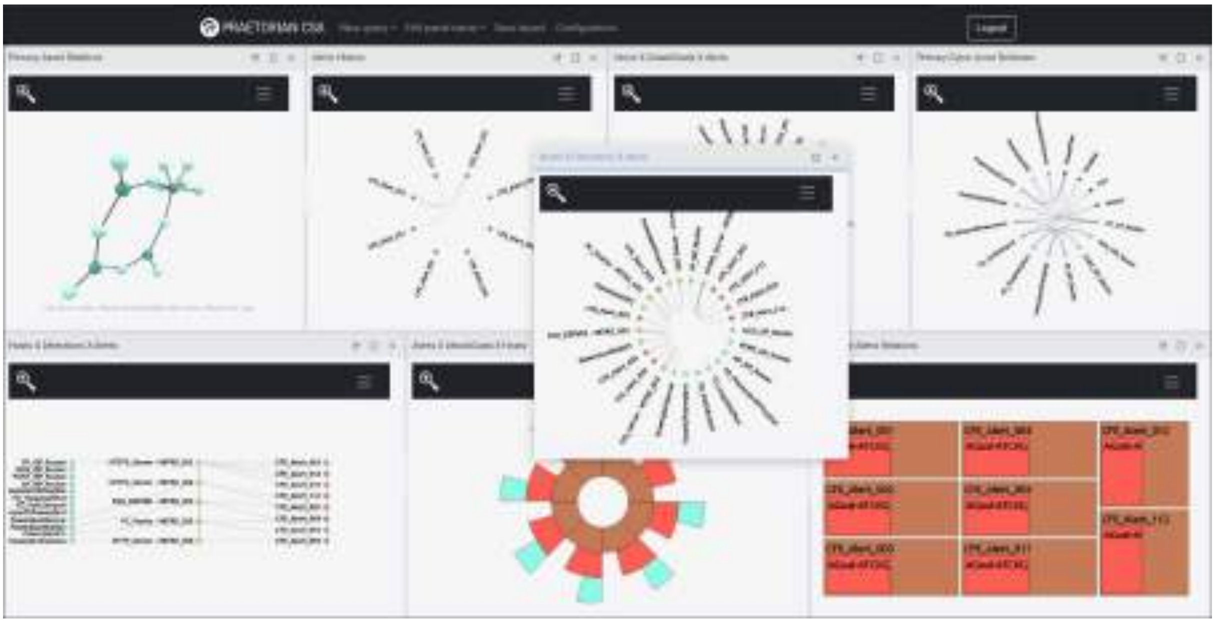


Figure 37 – Overall view I

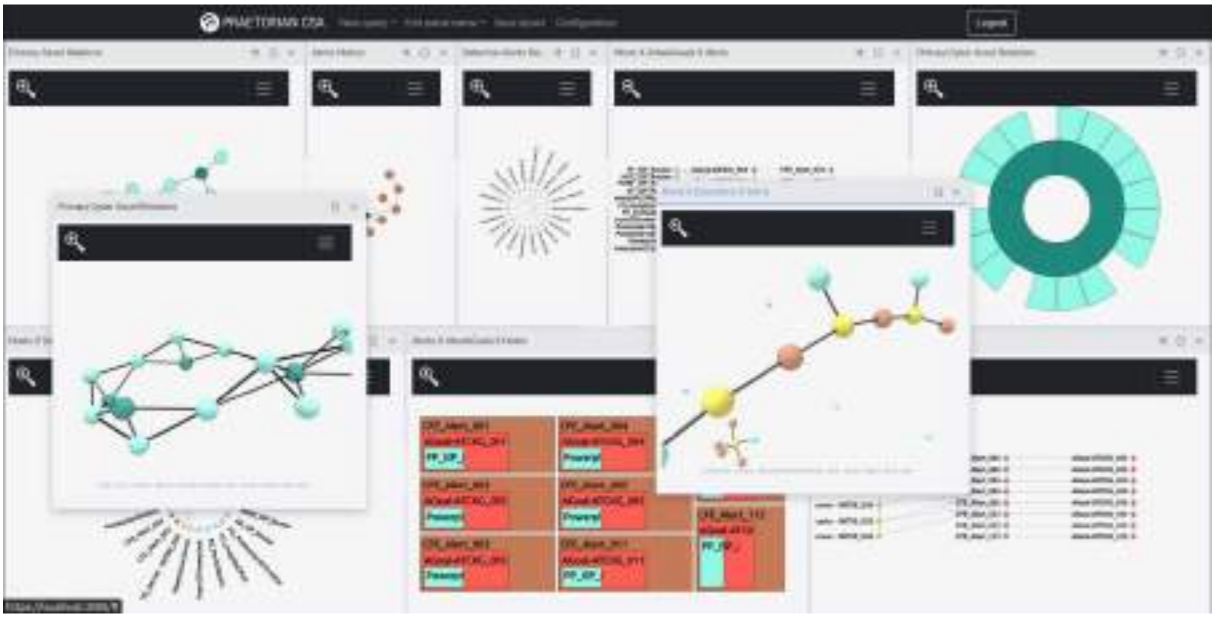
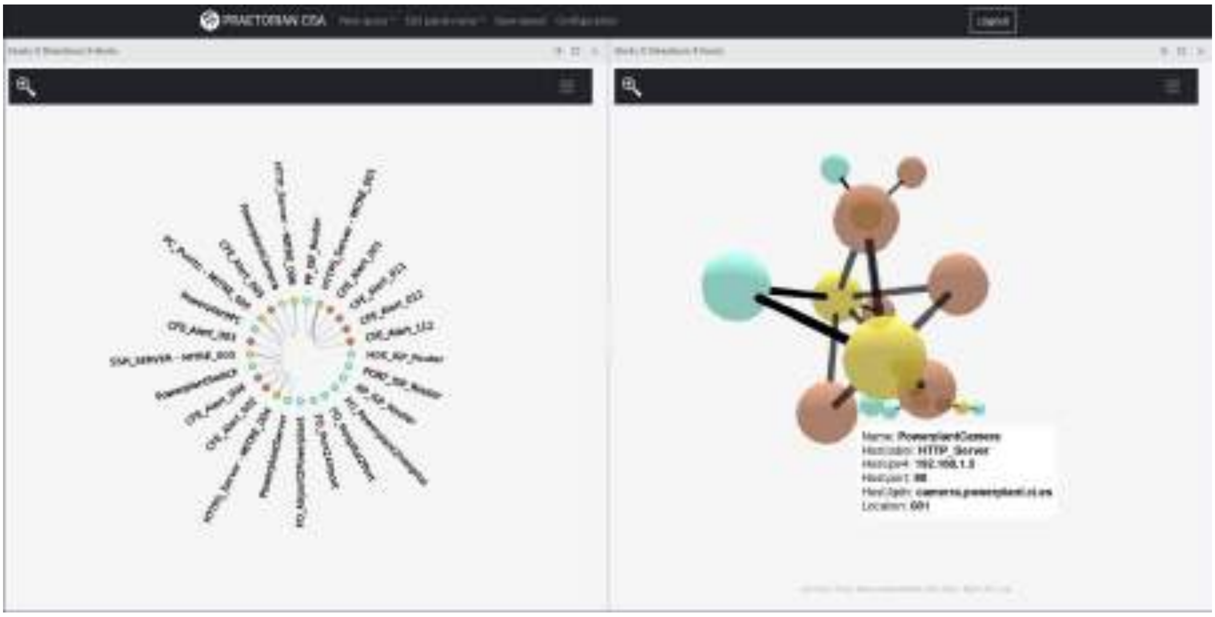


Figure 38 – Overall view II



Name: PowerplantCamera
Hostcode: HTTP_Server
HostipV4: 192.168.1.5
HostPort: 80
HostDptn: camera.powerplant.ci.es
Location: 001

Figure 39 – Overall view III

3.2.5 Approach to immersive technologies

The PRAETORIAN CSA HMI provides the capability of showing visualizations of data in an immersive way, making use of Virtual Reality (VR) glasses. Currently, the glasses used in the project are Oculus rift CV1.

Figure 40 – Oculus Rift CV1 headset

This device is a well-known and mature product, superseded by other more recent products, yet with still proper features, established in the AR and VR research community. That is one of the reasons for its selection.

Display & Resolution	2160×1200 (1080×1200 per eye), Pentile AMOLED, 90Hz.
Field of View	110º
Optics/lenses	First-gen Hybrid Fresnel Lenses. Adjustable IPD (58mm to 71mm)
Sensors	Accelerometer Gyroscope Magnetometer
Tracking	6DoF external camera tracking. 360-degree IR LED head tracking.
Head mounting	Velcro head straps.
Weight	470 g.
Connectivity	HDMI 1.3 , USB 3.0, USB 2.0 (4 meter connection)

Table 1 - Oculus Rift CV1 specifications

The overall setup used is the following.

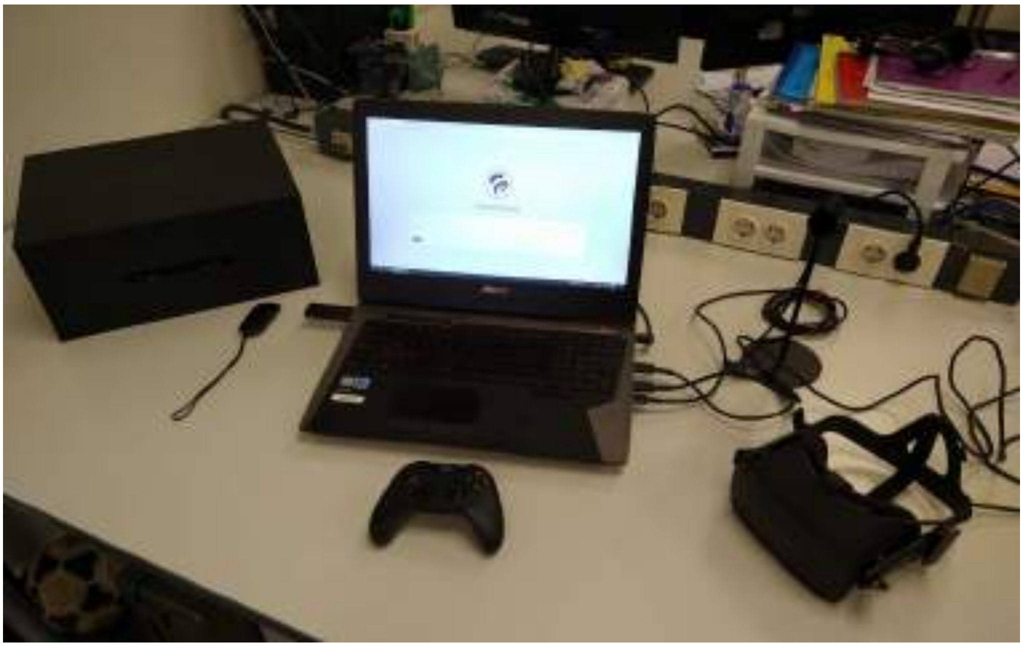


Figure 41 – Oculus Rift setup used

And all the components, up and running and showing the PRAETORIAN CSA visualizations can be seen in the following figure.

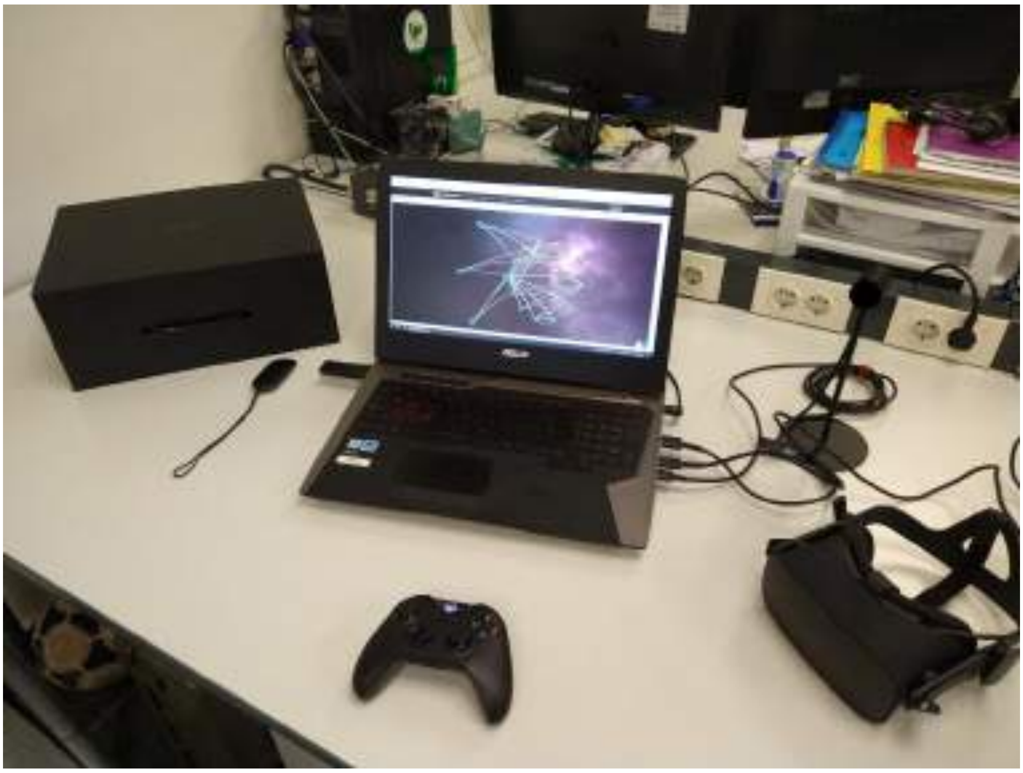


Figure 42 – Oculus Rift CSA HMI infrastructure up and running

To activate the visualizations, the user must select the specific view for Immersive VR, ‘Immersive Global view’, selected from the drop down for new queries, ‘New query’.

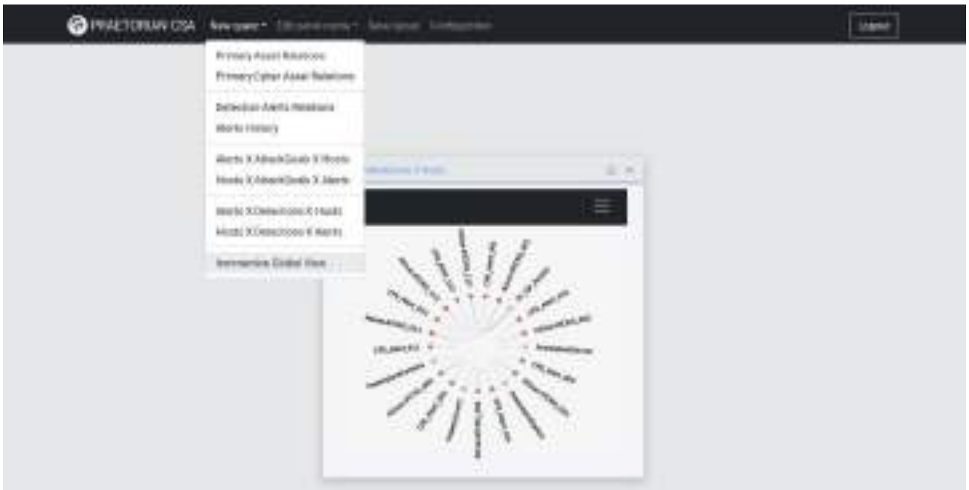

PRAETORIAN CSA
New query
Logout
Primary Asset Relations
Primary Cyber Asset Relations
Detection Alerts Relations
Alerts History
Alerts X AttackGoals X Hosts
Hosts X AttackGoals X Alerts
Alerts X Detections X Hosts
Hosts X Detections X Alerts
Immersive Global View

Figure 43 – Immersive capabilities selection

Once selected, the user will be presented withwith the following interface.

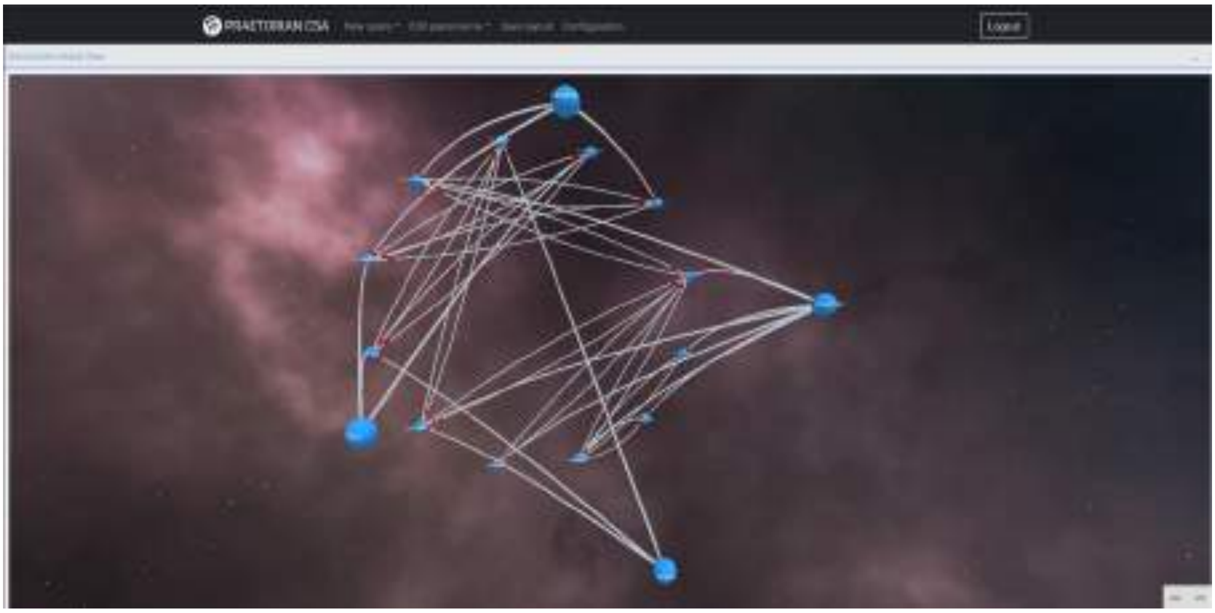


Figure 44 – Immersive view on a 2D screen

The information shown here is mainly related to the network topology. Basically, it depicts the assets that are present in a given operation (shown as nodes), the connections among them (net flows, shown as links with animated packets traversing from one to another), as well as some extended information for the given assets.

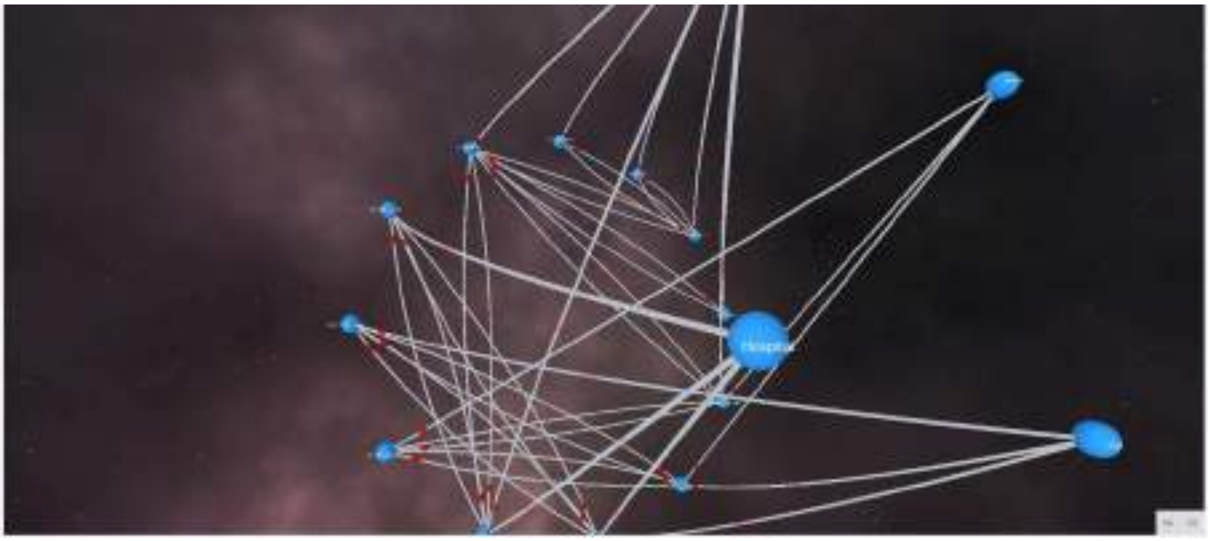


Figure 45 – Data representation in 3D

The system is designed to show this information in an on-screen 3D view or in the glasses. To send the data and show it on the glasses, the user must click on the button VR and then the visualization will be displayed in the 3D view on screen or in the glasses.

In the following snapshot, we can see an approximation of what is seen at the glasses, provided by the debugging tool of the Oculus Rift infrastructure.

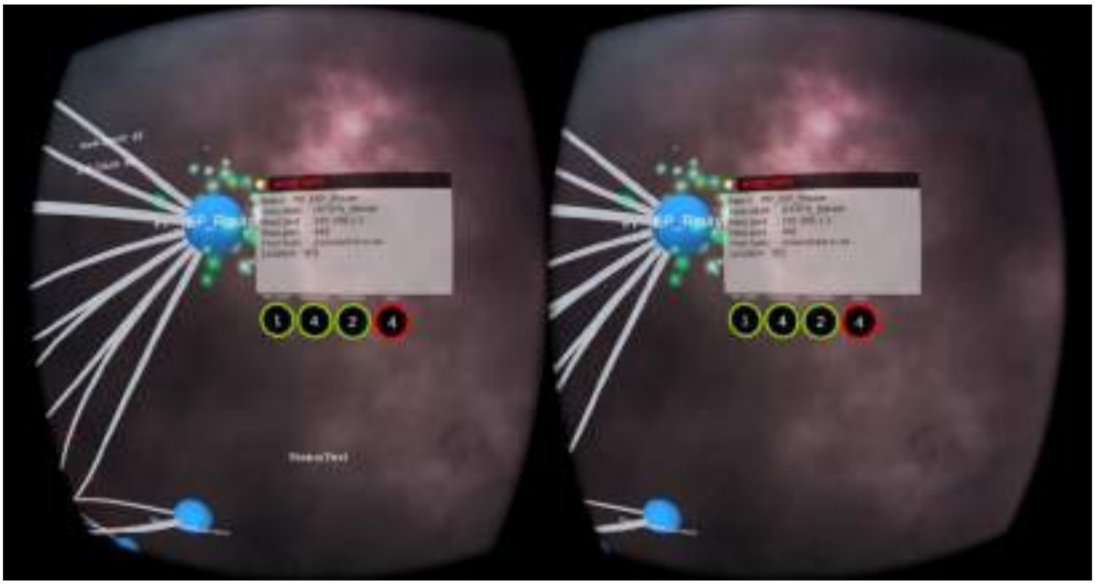


Figure 46 – Information as seen in the immersive glasses

It can also be seen at a given screen, in parallel with the regular usage. As shown in the following figure, the operator is seeing the visualization in an immersive experience and, at the same time, the visualization is projected in a 3D view over a 2D laptop screen. Notice how the user is making usage of the haptic devices to interact with the visualization (as well as moving 360º his head).

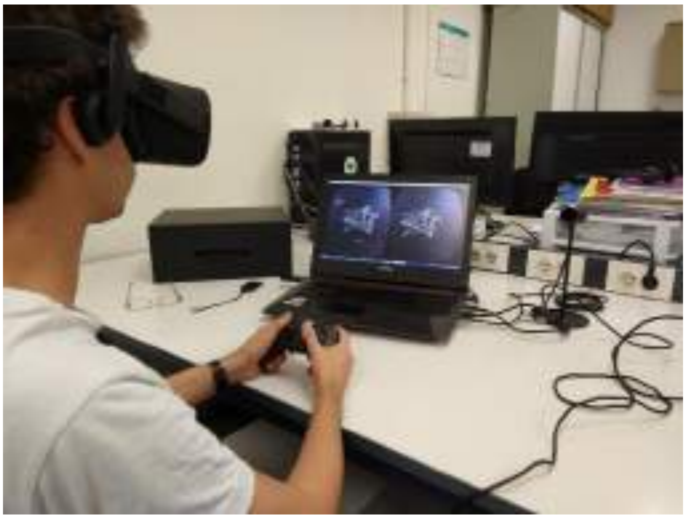


Figure 47 – Immersive experience

To access extra information, both on the 3D screen and the glasses, the user must point to the node of interest and then click on it. This will be different depending on the kind of interface used (mouse + keyboard, or the haptic device provided with the glasses). Once clicked, the node will be selected, as shown in the following figure.

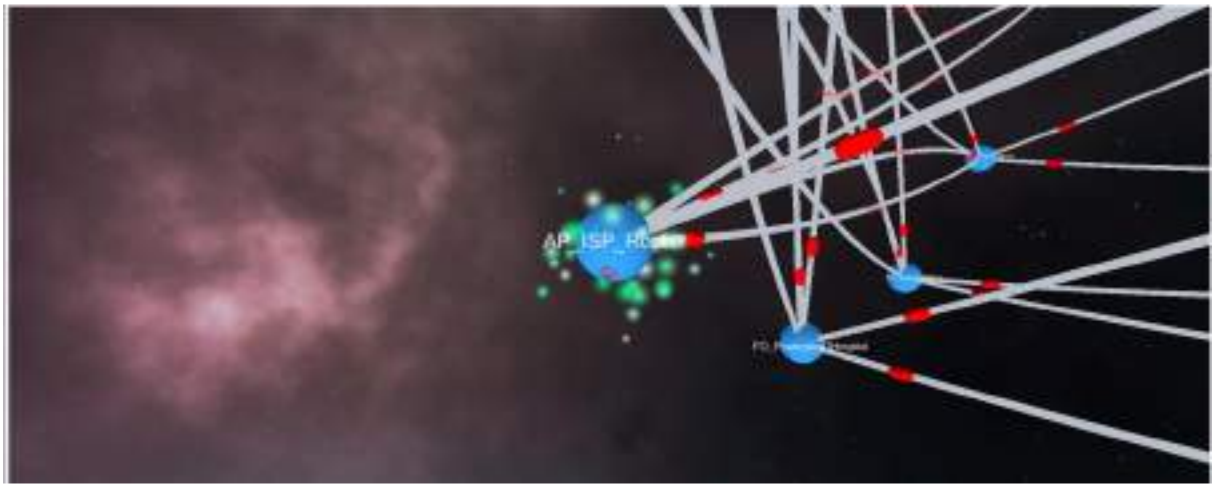


Figure 48 – Item selection

If we click now the right button (equivalently on the haptic), we will see extra information about the given element, for instance the name, IP address, ports on usage, FQDN, physical assets associated, cyber assets, detections and alerts.

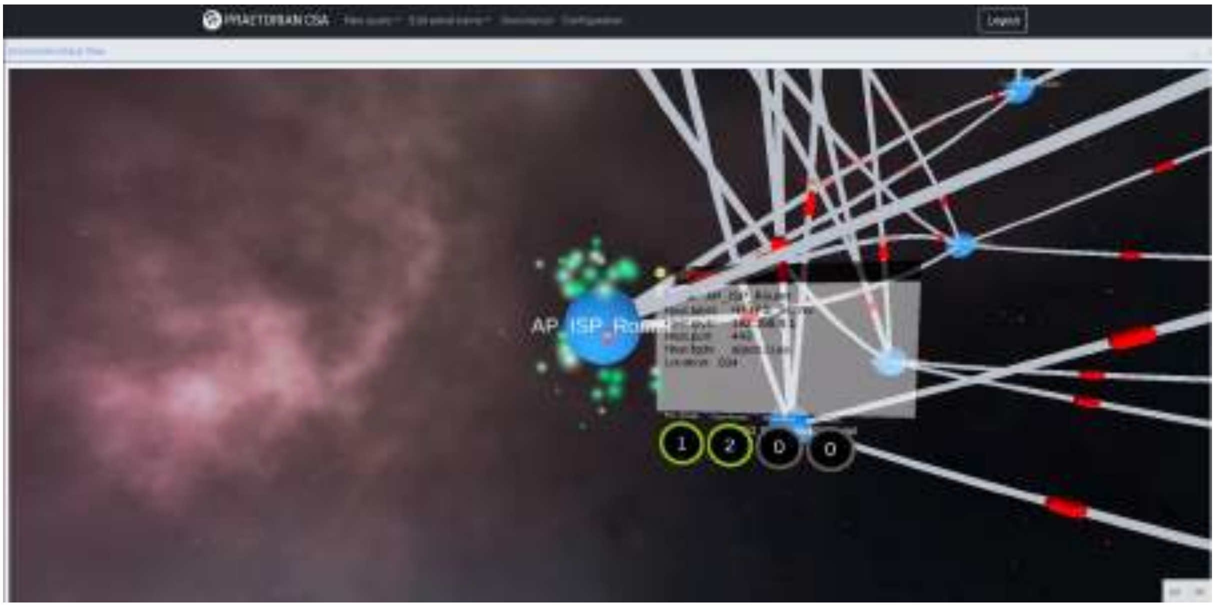
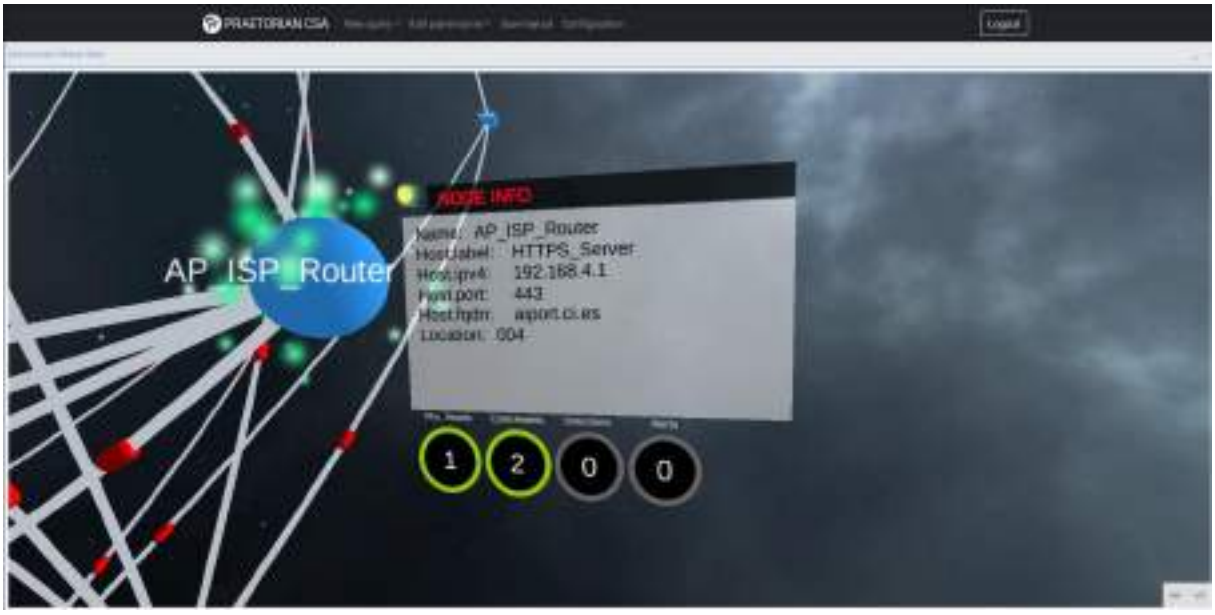


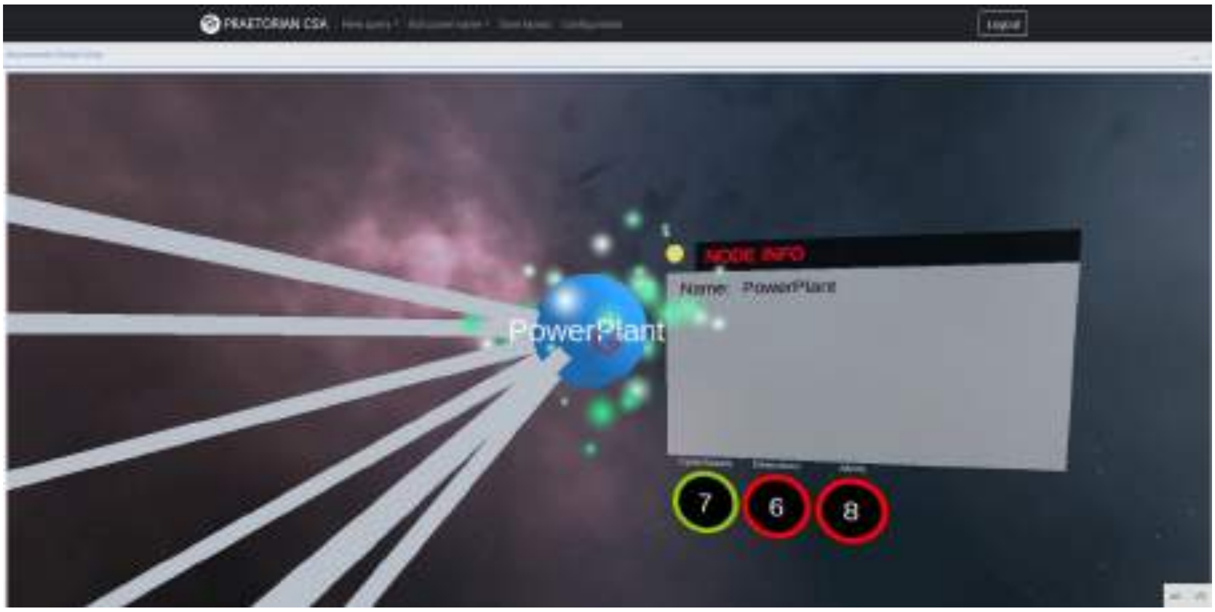
Figure 49 – Extra information for a given node I



Name: AP_ISP_Router
Host.label: HTTPS_Server
Host.ipv4: 192.168.4.1
Host.port: 443
Host.fqdn: aiport.ci.es
Location: 004

Figure 50 – Extra information for a given node II

If the CFE detection engine detects something or triggers an alarm, then it will be shown in red colour.



NODE INFO

Name: PowerPlant

Figure 51 – Colour codes for extra information

3.2.6 Access to extended information

The information shown to the users in the CSA HMI can be extended by interaction in the graphs. Users can select a given element shown in the advanced views and retrieve extra information on that given node. To do so, there are two main approaches: by the tool tip usage and by using the info button. Therefore, if we are accessing a visualization and we want to access extra information, we will be in a setup like the it is shown below.

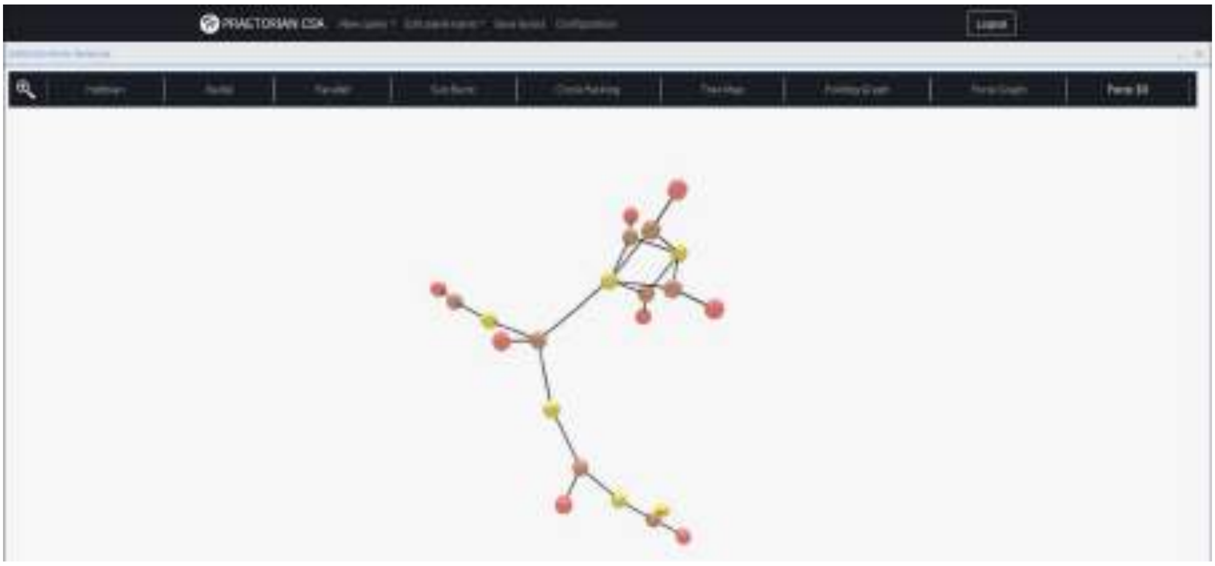
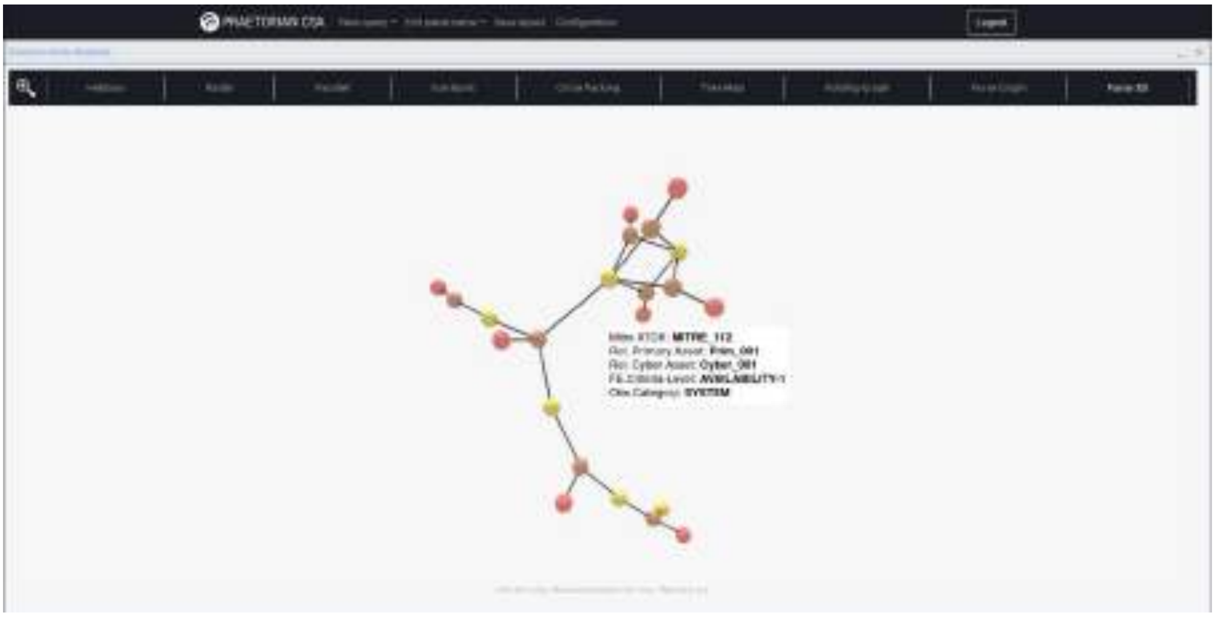


Figure 52 – tool tip I

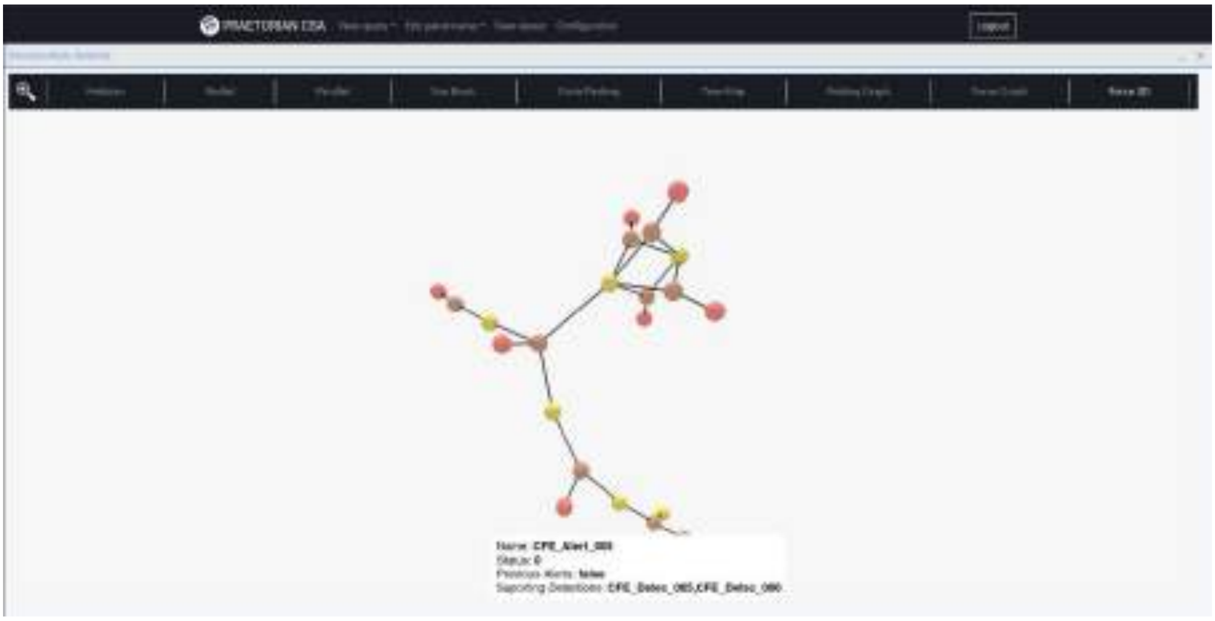
Then, if we move the cursor on top of a given node, after a second, we will see the extra information as follows.



Mitre ATCK: MITRE_112
Rel. Primary Asset: Prim_001
Rel. Cyber Asset: Cyber_001
FE Criticality Level: AVAILABILITY-1
Cita Category: SYSTEM

Figure 53 – tool tip II

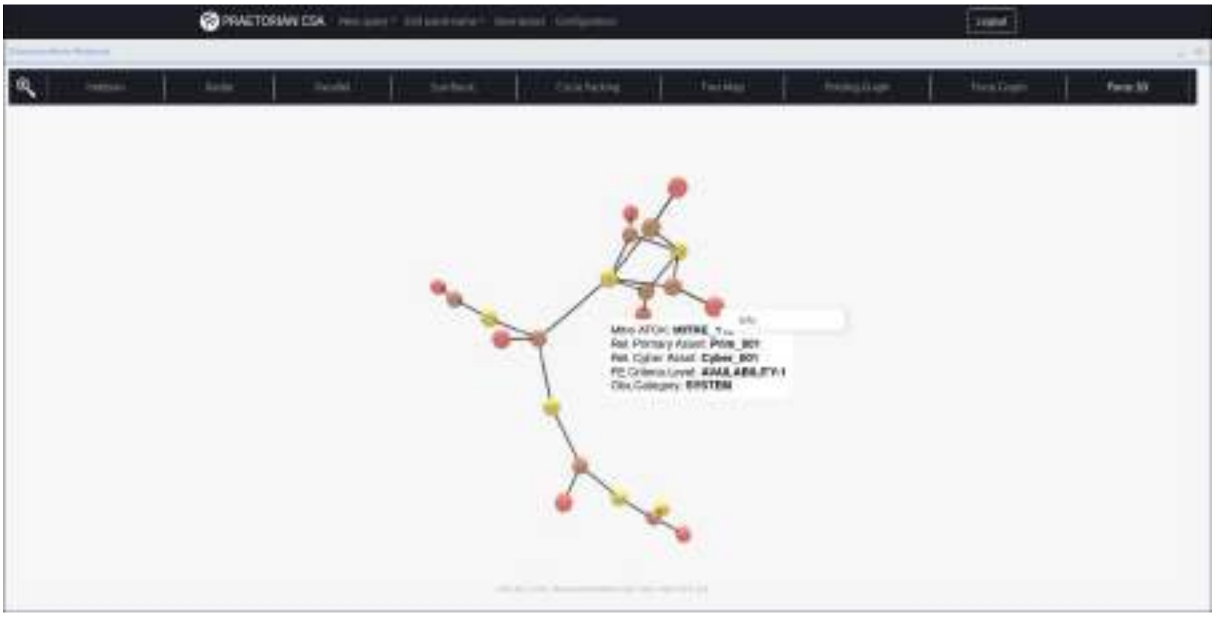
At this moment, the user can select any other node in the graph to gather extra information related to it, as can be seen as follows for a CFE Alert.



Name: CFE_Alert_008
Status: 0
Previous Alerts: false
Supporting Detections: CFE_Detec_085,CFE_Detec_086

Figure 54 – tool tip III

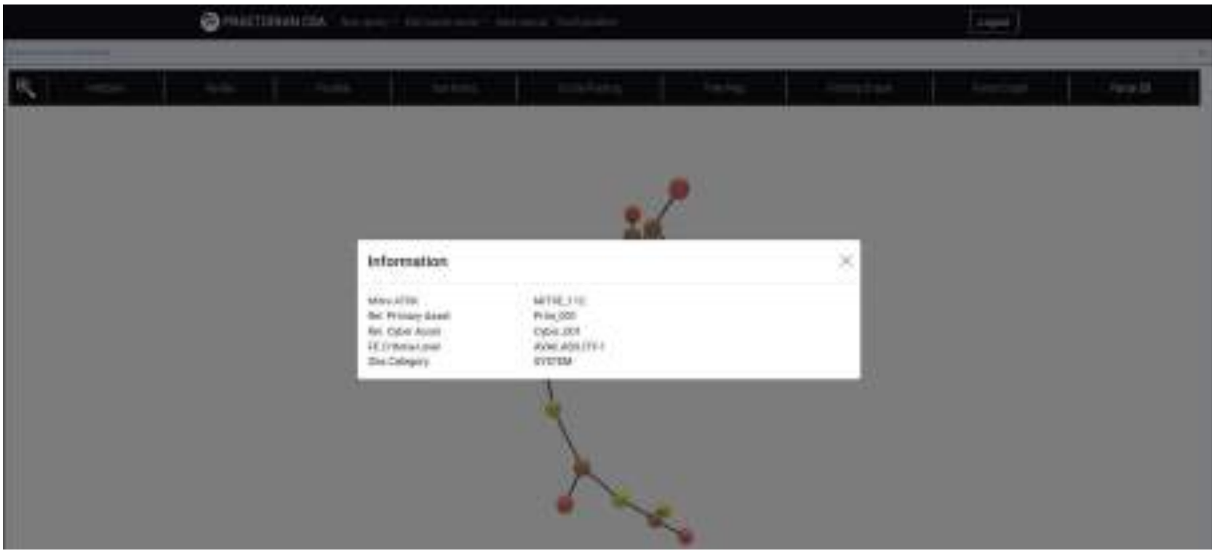
To access extra and enhanced information on the node, we have to follow the procedure described below. Now we click on the right hand button of the mouse while we have the mouse located on a given node.



Mitre ATCK: MITRE_T...
Rel. Primary Asset: Prim_001
Rel. Cyber Asset: Cyber_001
FE Criteria Level: AVAILABILITY-1
Oss Category: SYSTEM

Figure 55 – Info button I

If we click on information, we see extra information for that given element. This can be applied to any node in a given visualization.



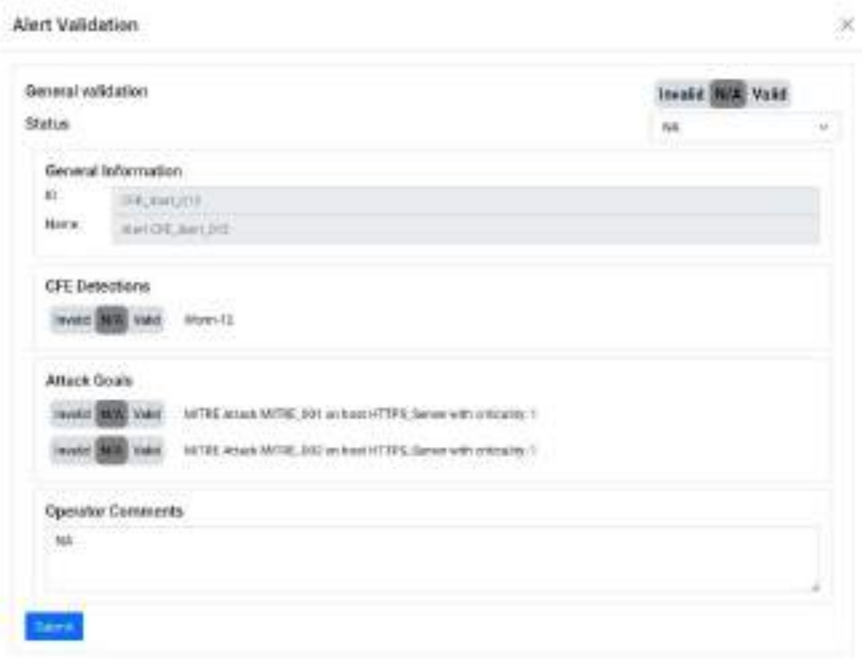
Information

Mitre ATCK	MITRE_T1[illegible]
Rel. Primary Asset	Prim_001
Rel. Cyber Asset	Cyber_001
FE Criteria Level	AVAILABILITY-1
Oss Category	SYSTEM

Figure 56 – Info button II

3.3 Information validation capabilities

The CSA HMI user interacts continuously with other elements of the PRAETORIAN CSA system. One of the responsibilities of the HMI operator, and key element of its modus operandi, is to validate CFE automatic detections and alerts, among others. Therefore, at the CSA HMI, there is an interface to validate all this machine-generated data provided in support for human decision-making. To access those validation interfaces, the operator must select an item needing validation, i.e., mainly alerts, and then right-click on it. Then, 'validation' must be left-clicked to get to the item validation main window.


Alert Validation

General validation

Status: N/A

General Information

ID: CFE_alert_013

Name: alert CFE_alert_002

CFE Detections

Worm-12

Attack Goals

MITRE attack MITRE_001 on host HTTPS_Server with criticality: 1

MITRE Attack MITRE_002 on host HTTPS_Server with criticality: 1

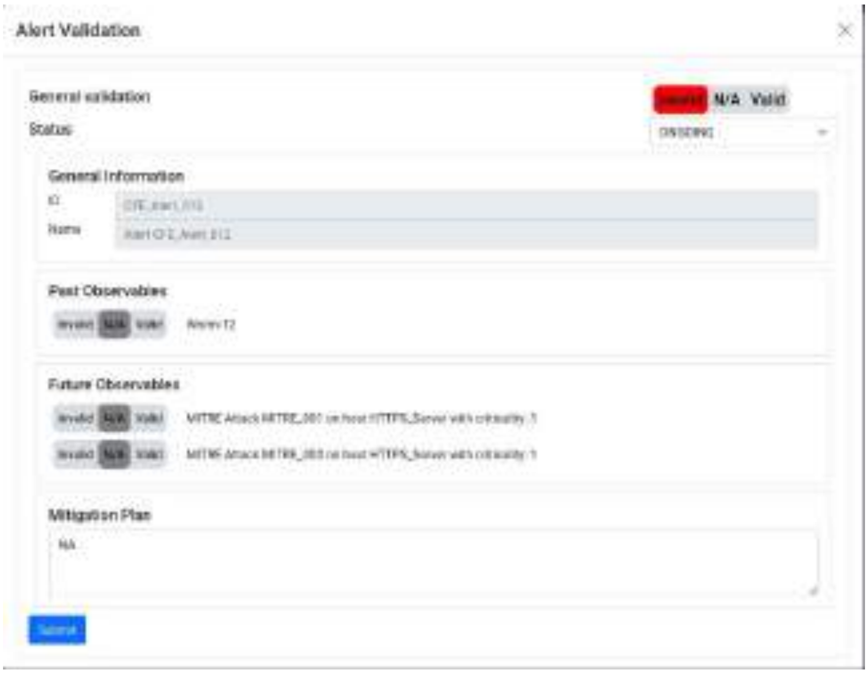
Operator Comments

N/A

Figure 57 – Main validation interface

3.3.1 CFE alert confirmation

To confirm and validate a CFE alert, the operator must select the top toggle to switch from one state to the other. Once the value is correct, the 'submit' button must be clicked to send the data of the validation process to the IOP. Doing this action, the user confirms or denies that an alert is actually an alert (referring to a potential ongoing threat).



Alert Validation

General validation

Status: ONGOING

General Information

ID: CFE_Alert_012

Name: Alert CFE_Alert_012

Past Observables

Worm 12

Future Observables

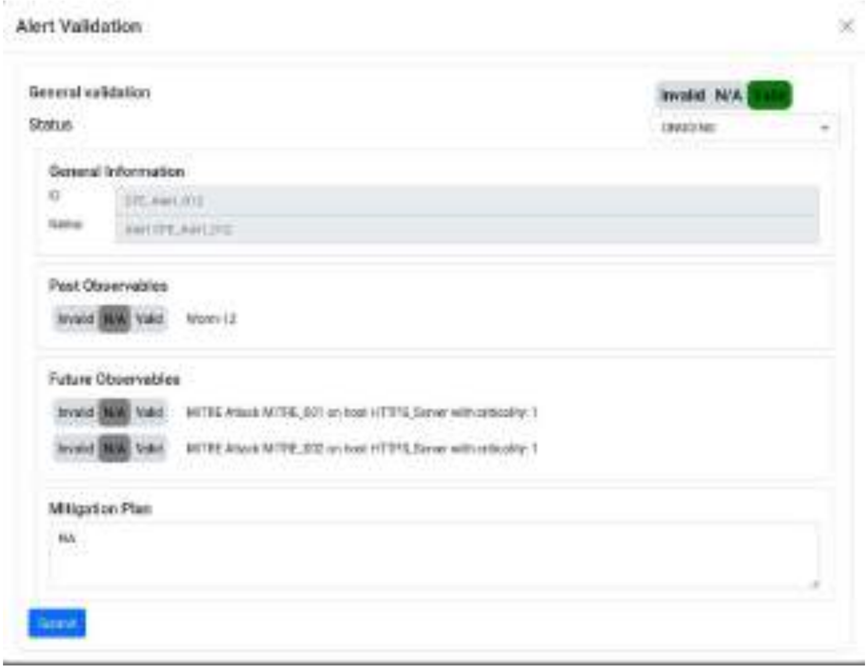
MITRE Attack MITRE_001 on host HTTPS_Server with criticality: 1

MITRE Attack MITRE_003 on host HTTPS_Server with criticality: 1

Mitigation Plan

N/A

Figure 58 – CFE alert confirmation I



Alert Validation

General validation

Status: ONGOING

General Information

ID: CFE_Alert_012

Name: Alert CFE_Alert_012

Past Observables

Worm 12

Future Observables

MITRE Attack MITRE_001 on host HTTPS_Server with criticality: 1

MITRE Attack MITRE_002 on host HTTPS_Server with criticality: 1

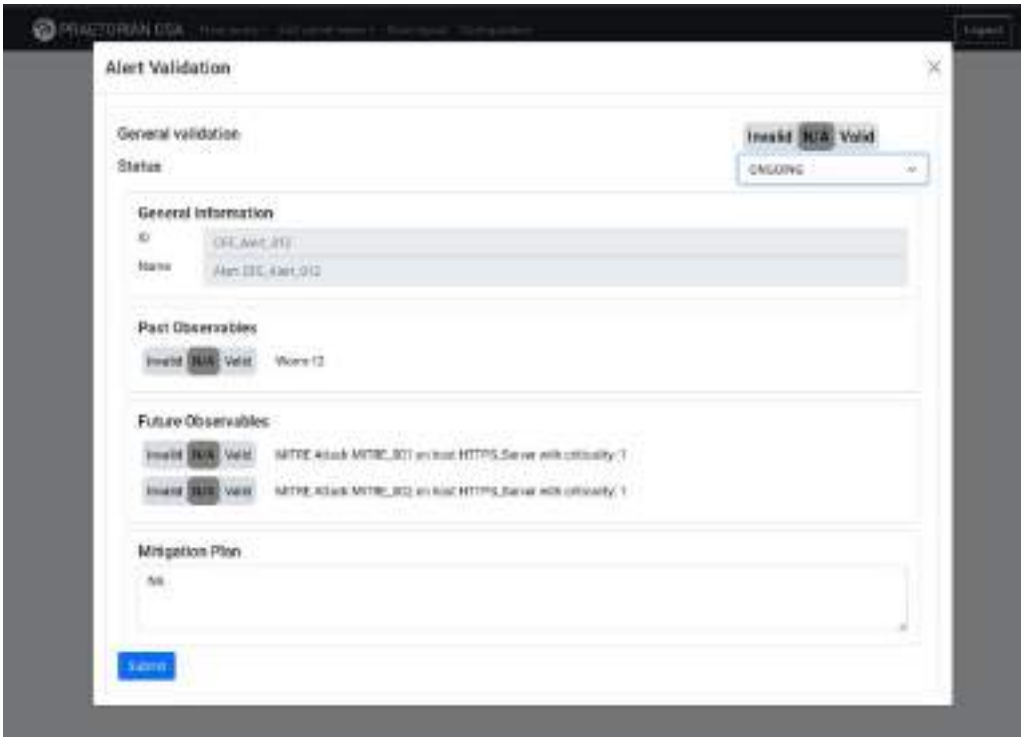
Mitigation Plan

N/A

Figure 59 – CFE alert confirmation II

3.3.2 CFE alert status update

Making use of this capability, the user can specify the current status of an alert from a defined set of possible choices. For instance, he/she can discard a non-relevant alert (by updating its status to "closed"), or to other possible statuses (if any).



Alert Validation

General validation: Invalid [✓ N/A] Valid

Status: ONGOING

General Information

ID: CFE_Alert_012

Name: Alert CFE_Alert_012

Past Observables

Invalid [✓ N/A] Valid Worm 12

Future Observables

Invalid [✓ N/A] Valid MITRE Attack MITRE_001 on host HTTPS_Server with criticality: 1

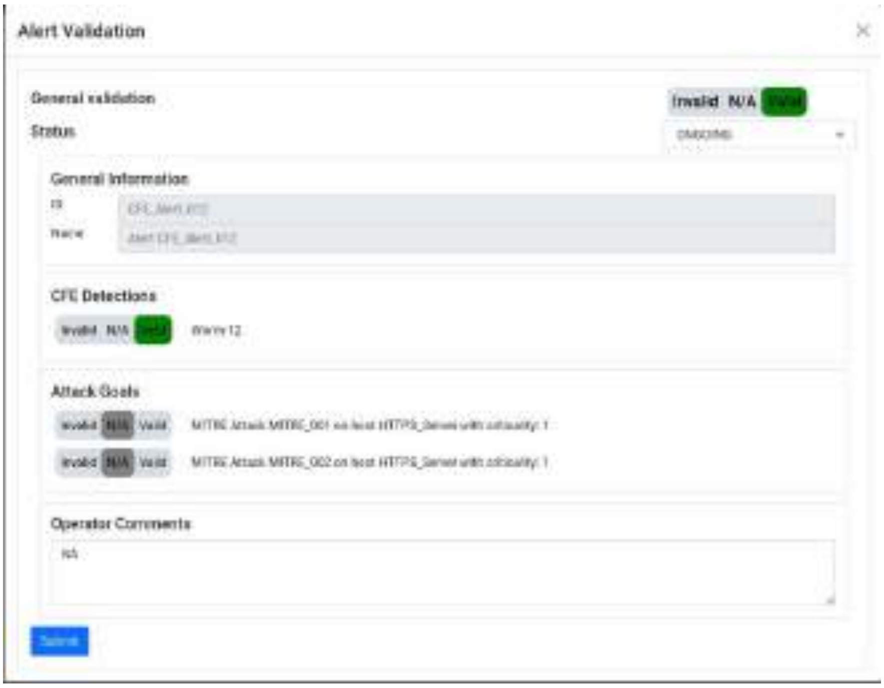
Invalid [✓ N/A] Valid MITRE Attack MITRE_002 on host HTTPS_Server with criticality: 1

Mitigation Plan

N/A

Figure 60 – Alert status update

3.3.3 CFE detections validation



Alert Validation

General validation: Invalid N/A [✓ Valid]

Status: ONGOING

General Information

ID: CFE_Alert_012

Name: Alert CFE_Alert_012

CFE Detections

Invalid N/A [✓ Valid] Worm 12

Attack Goals

Invalid [✓ N/A] Valid MITRE Attack MITRE_001 on host HTTPS_Server with criticality: 1

Invalid [✓ N/A] Valid MITRE Attack MITRE_002 on host HTTPS_Server with criticality: 1

Operator Comments

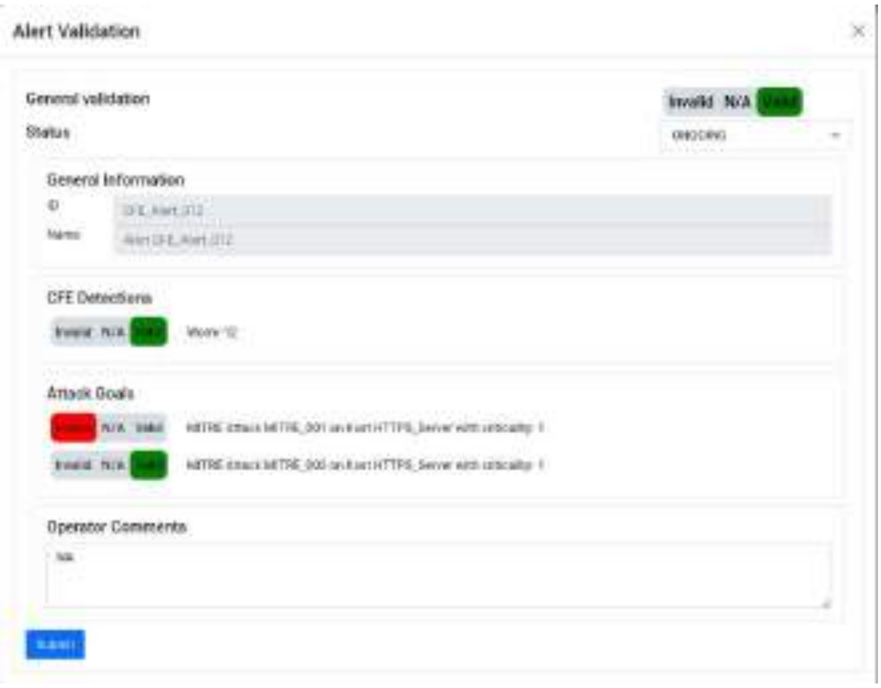
N/A

Figure 61 – CFE detections validation

By means of this interface, the user can confirm which of the detections supporting a specific alert are actually related to this alert and which are not.

3.3.4 Attack goals confirmation

This submenu allows the user to confirm which of the estimated attack goals are actually possible.



General validation

Status: ONGOING

General Information

ID: CFE_Alert_012

Name: Alert CFE_Alert_012

CFE Detections

Worm 12

Attack Goals

MITRE attack MITRE_001 on asset HTTPS_Server with criticality: 1

MITRE attack MITRE_000 on asset HTTPS_Server with criticality: 1

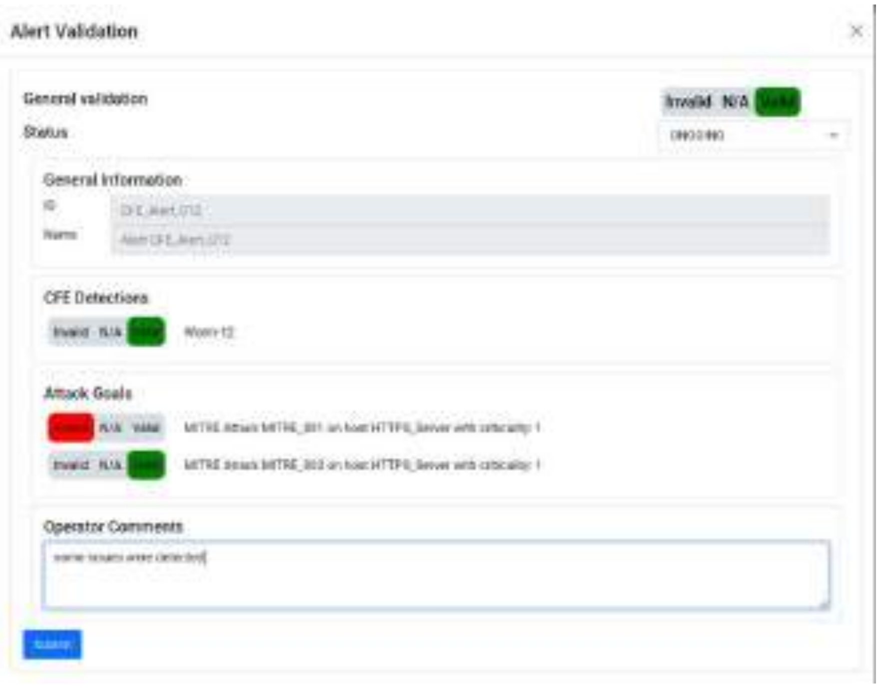
Operator Comments

NA

Figure 62 – Attack goals

3.3.5 CSA operator comments

For instance, making use of this capability, a mitigation plan can be stated as an extra set of comments. It can be seen in the following screen-shot.



General validation

Status: ONGOING

General Information

ID: CFE_Alert_012

Name: Alert CFE_Alert_012

CFE Detections

Worm-12

Attack Goals

MITRE Attack MITRE_001 on host HTTPS_Server with criticality: 1

MITRE Attack MITRE_002 on host HTTPS_Server with criticality: 1

Operator Comments

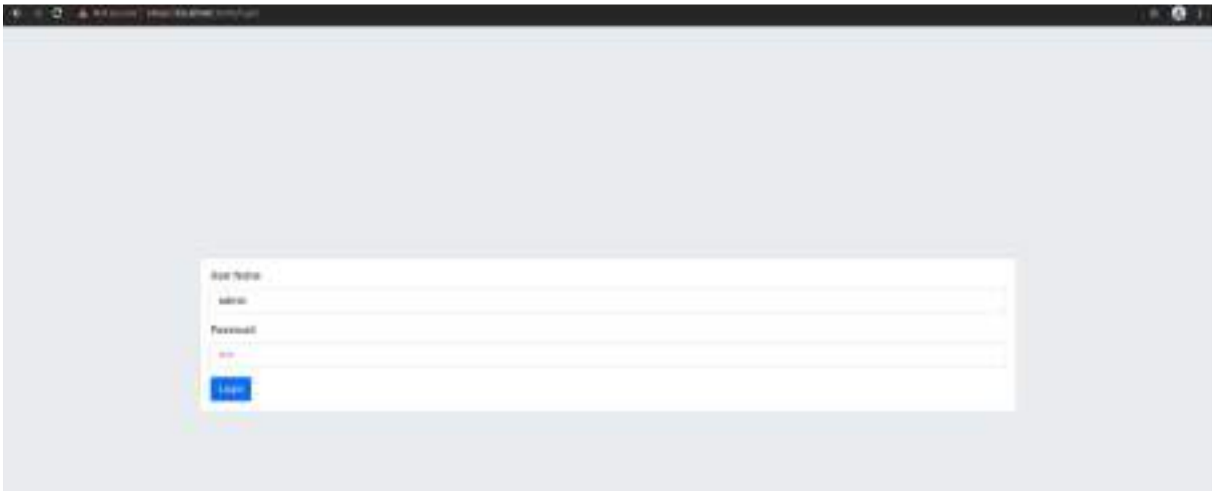
some issues were detected

Figure 63 – Operator comments

4. Additional CSA HMI functions

4.1 HMI authentication

To log into the system, the user must provide valid credentials. Therefore, each time we try to connect to the PRAETORIAN CSA, we will be faced with the following interface.



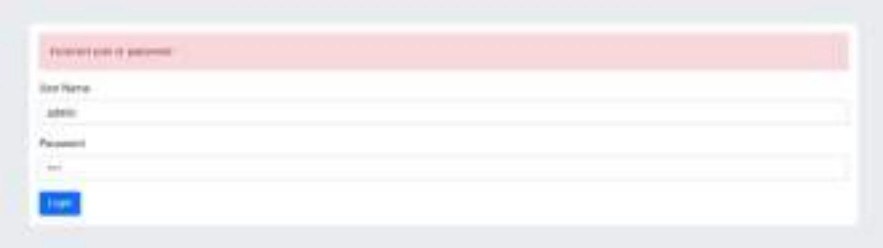
User Name

admin

Password

Figure 64 – Authentication process I

If we do provide an incorrect username and/or password, the system will provide the corresponding message, for obvious security reasons.



Incorrect user or password

User Name

admin

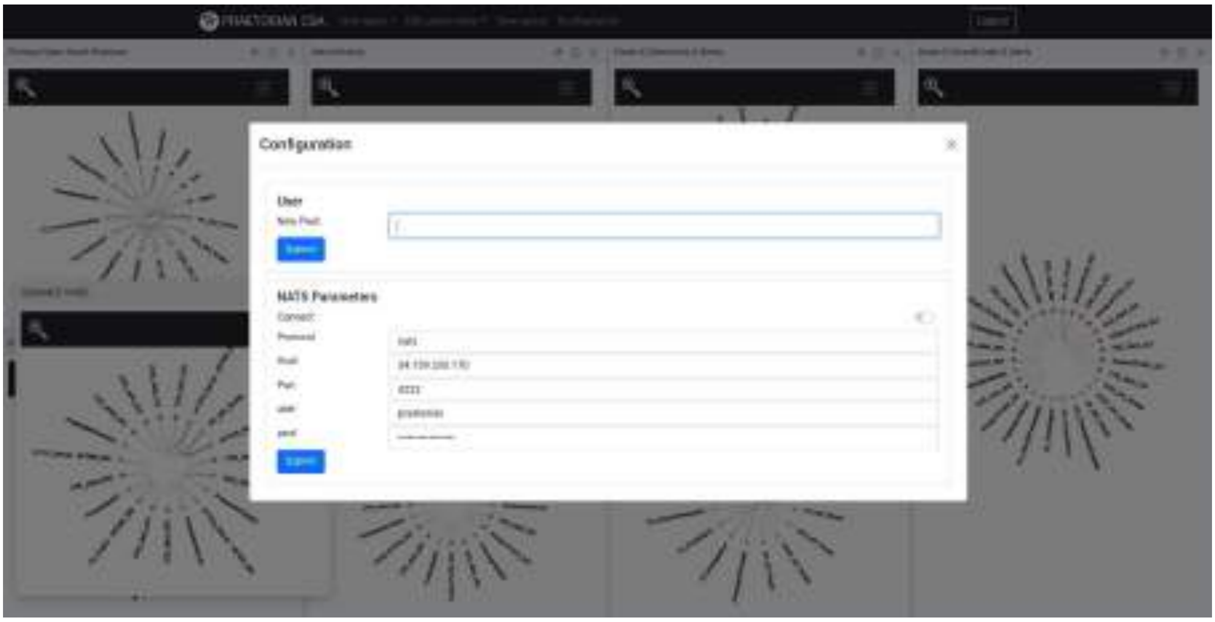
Password

Figure 65 – Authentication process II - error

If we provide proper credentials, the system will lead us to the CSA main page with our last saved layout configuration. To exit the application, click on logout.

4.2 System configuration

In this section, the user can, at any time, modify his current password and can also modify the parameters to access the CSA backend components endpoint, meaning the main PRAETORIAN data redistribution and repository, which is the IOP. To do so, the user accesses the 'Configuration' item in the corresponding drop down menu. Once done, he will be exposed to the following screen.



User
New Pwd:

NATS Parameters
Connect:
Protocol: nats
Host: 34.159.200.170
Port: 4222
user: praetorian
pwd: ••••••••••

Figure 66 – System configuration

To modify the password, just fill the corresponding text box and click on 'Submit'. To modify the backend connection parameters, the user can select:

- If it is connected or not
- Protocol
- Host
- Port
- IOP credentials

4.3 Layout customization

The first time the system starts and if no prior configuration has been defined, the user is offered with a blank screen like the following.

Figure 67 – System start with no setup

To start populating the tool with relevant, active, data-connected panels, the user must select them from the Queries drop down menu. To do so, the drop down must be clicked and then a specific query must be selected from the drop down menu, as can be seen in the following figures.

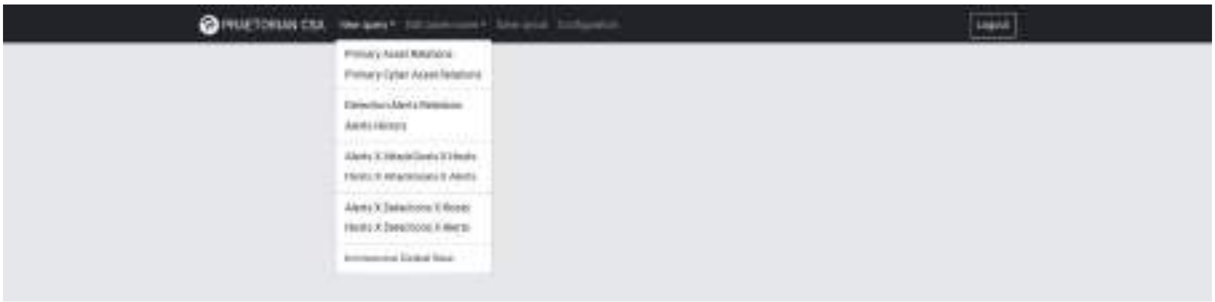

PRAETORIAN CSA
Primary Asset Relations
Primary Cyber Asset Relations
Detection Alerts Relations
Alerts History
Alerts X AttackGoals X Hosts
Hosts X AttackGoals X Alerts
Alerts X Detections X Hosts
Hosts X Detections X Alerts
Immersive Global View
Logout

Figure 68 – Selection of a given query

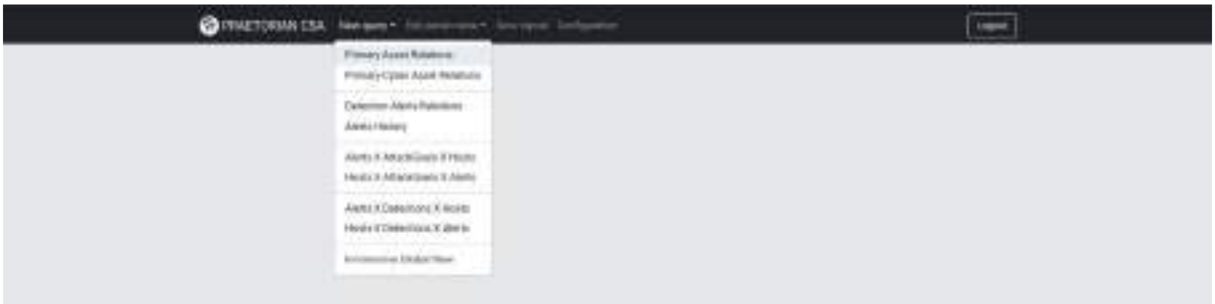

PRAETORIAN CSA
Primary Asset Relations
Primary Cyber Asset Relations
Detection Alerts Relations
Alerts History
Alerts X AttackGoals X Hosts
Hosts X AttackGoals X Alerts
Alerts X Detections X Hosts
Hosts X Detections X Alerts
Immersive Global View
Logout

Figure 69 – Selection of a given query

Once the visualization is selected, it will be shown on the application as a floating panel, as can be seen in the following figure.

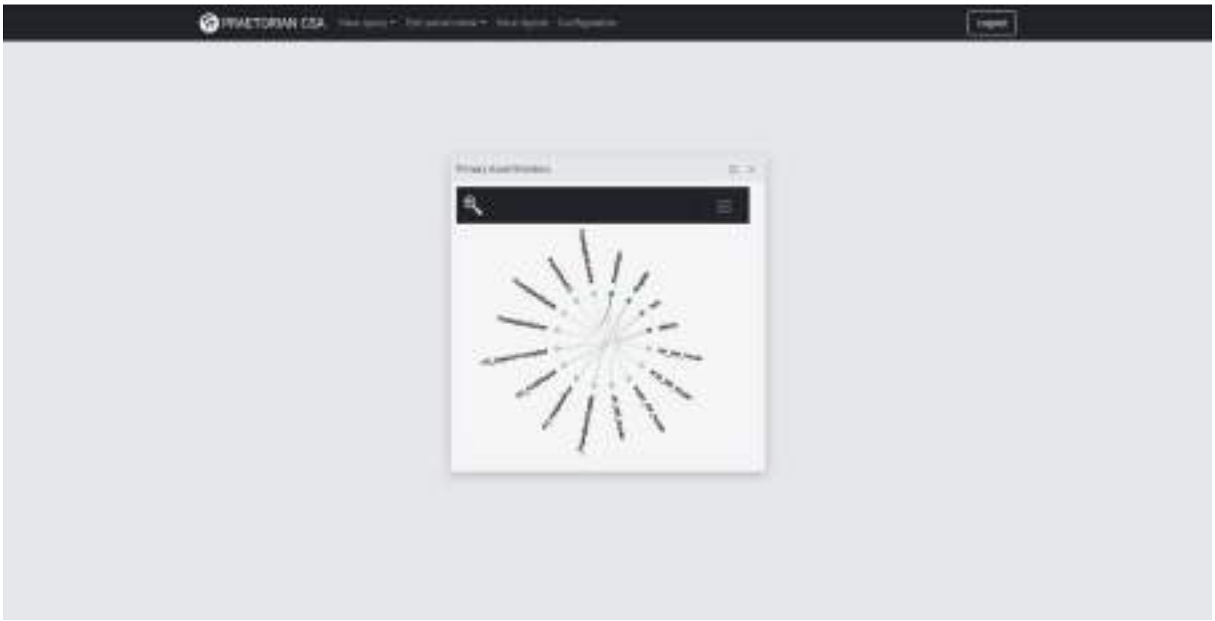


Figure 70 – Floating visualization

Panels are the key element of the configurability and flexibility of the application CSA in terms of view customization for providing the capability of having completely different visualization setups per each user. To do so, panels can be, at one time, in one of the two states: floating or docked. The floating state of a panel is shown in the previous figure, where the user can move around the panel at will. To dock a panel, the user must select it and then, while keeping the left mouse button pressed, drag it to the desired position and leave it, following the animation the HMI provides. This can be seen in the following figures.

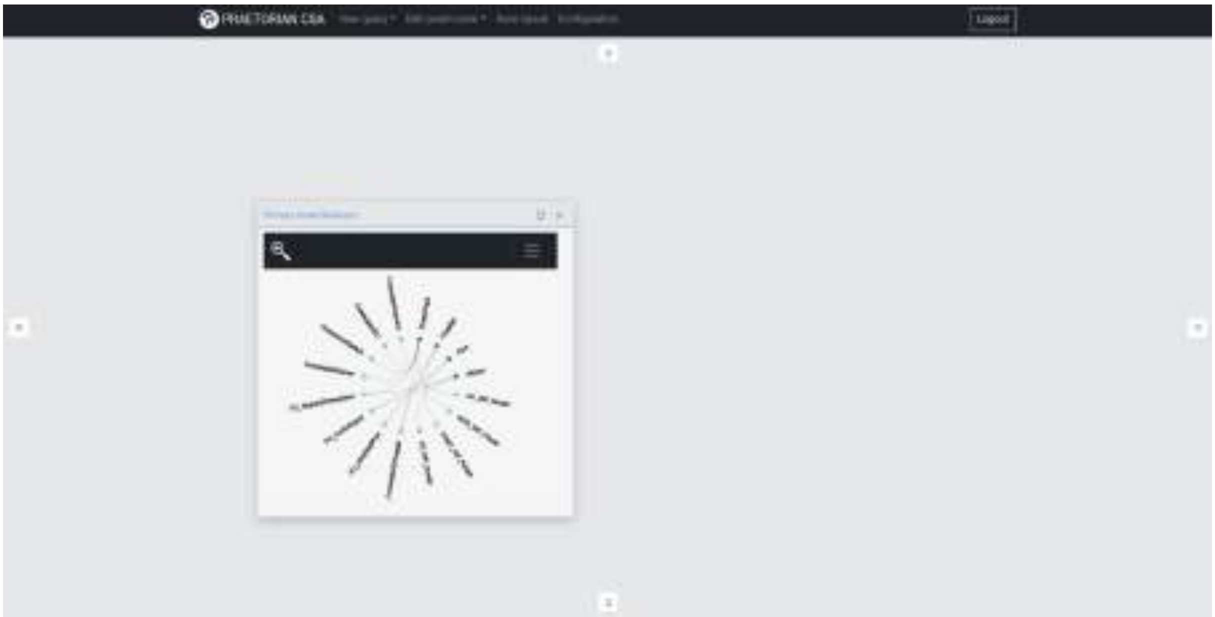


Figure 71 – layout customization I

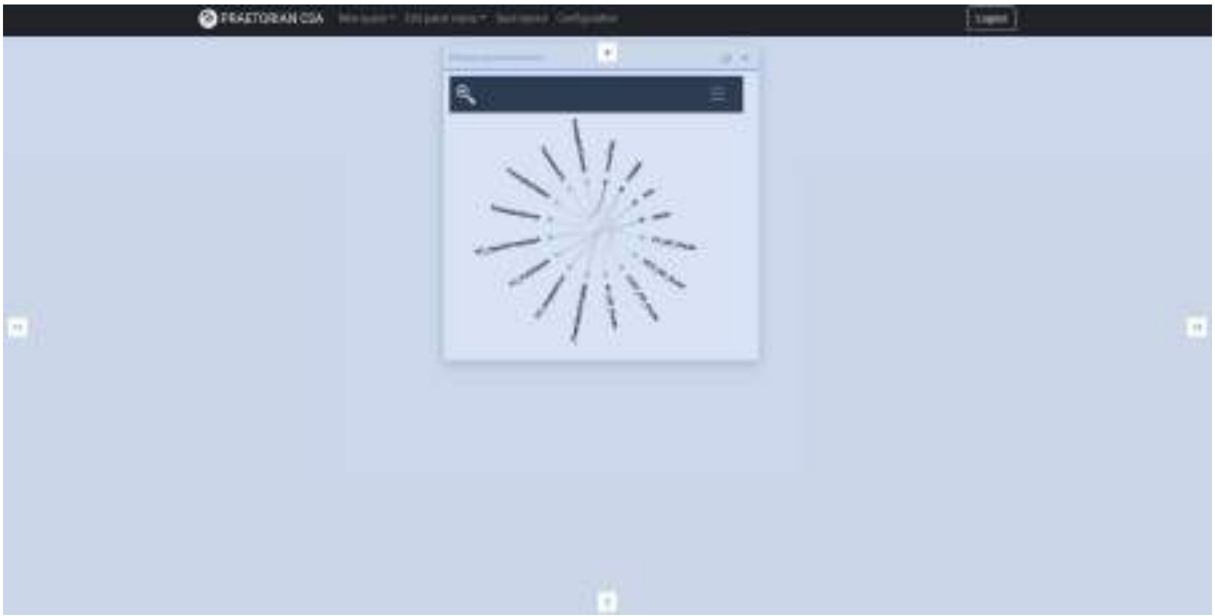


Figure 72 – layout customization II

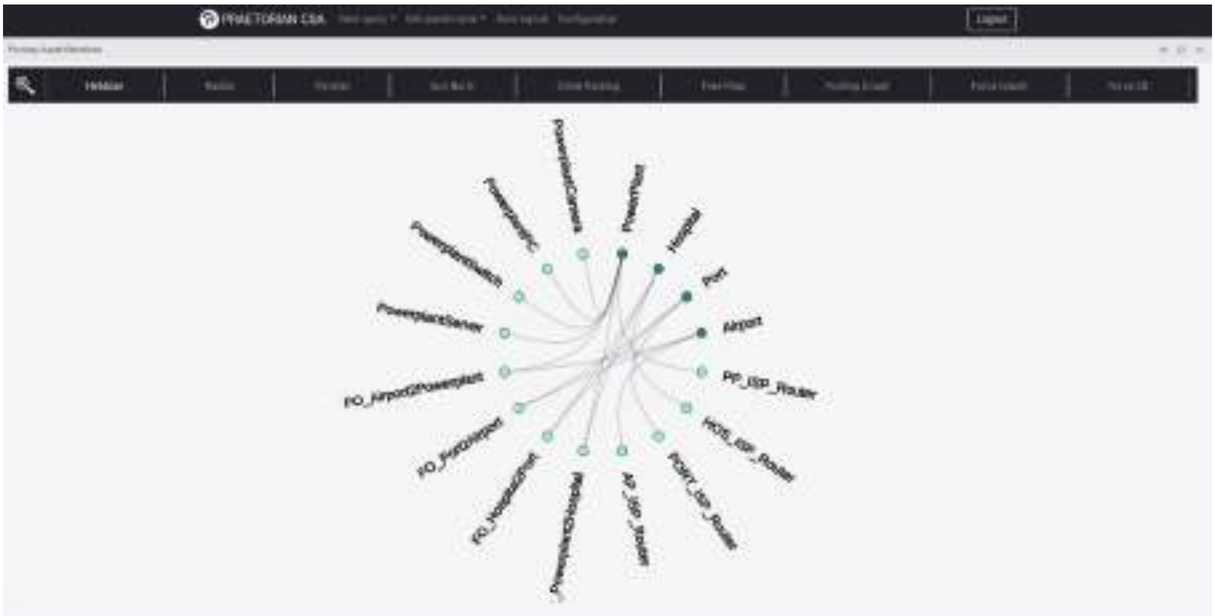


Figure 73 – layout customization III

Now, the view panel is docked in the desired position. As there were no panels already, the panel is centred regardless of the position we selected.

If we add new panels, we will see that the user is provided with several possibilities of panel docking, including panel-in-panel dockings.

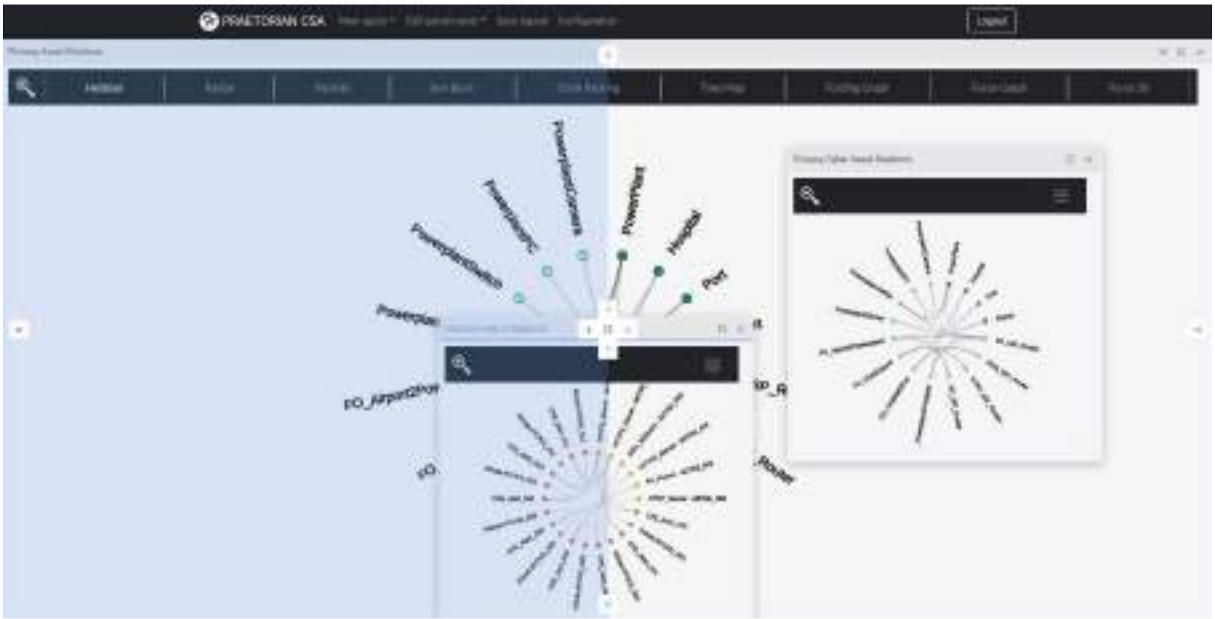


Figure 74 – layout customization IV

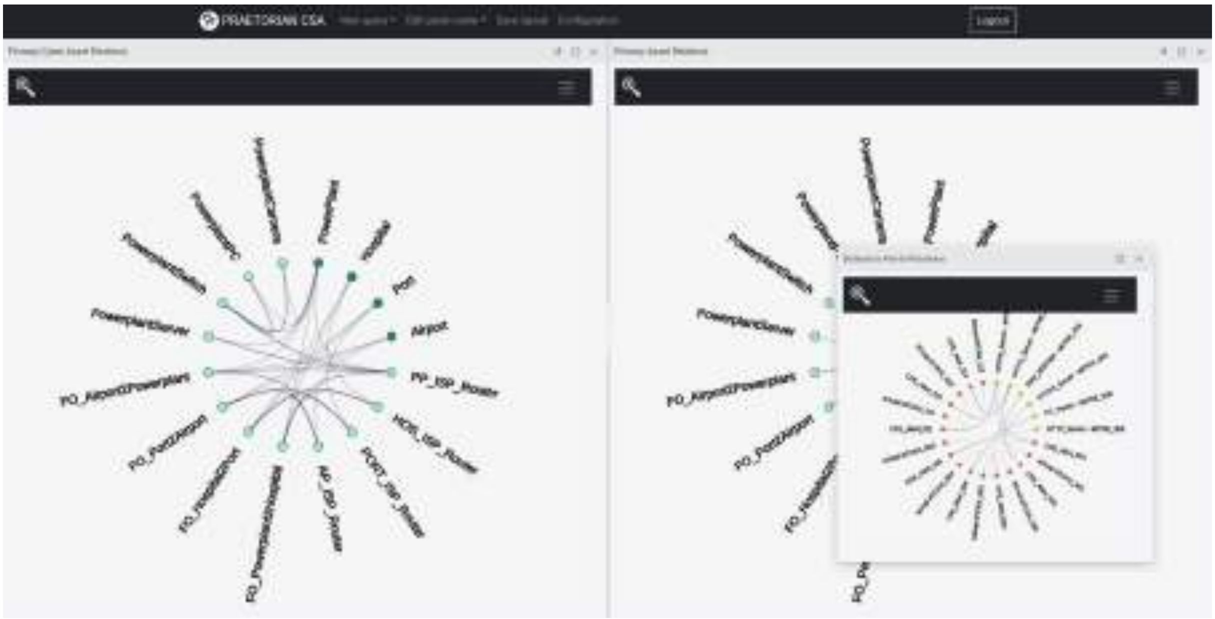


Figure 75 – layout customization V

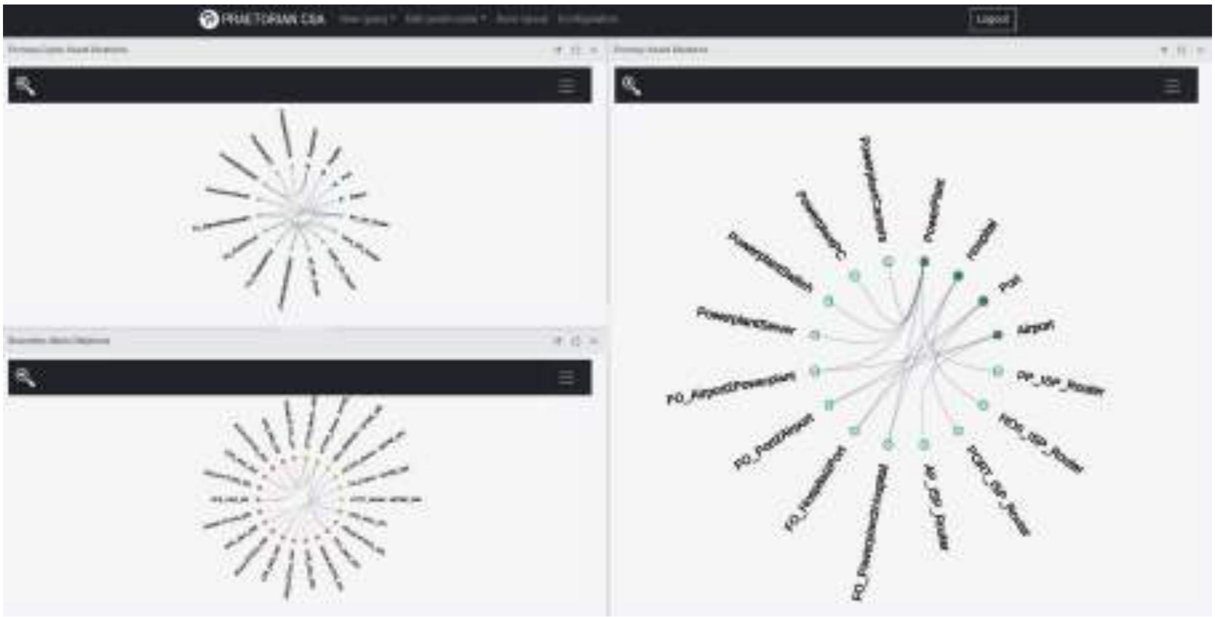


Figure 76 – layout customization VI

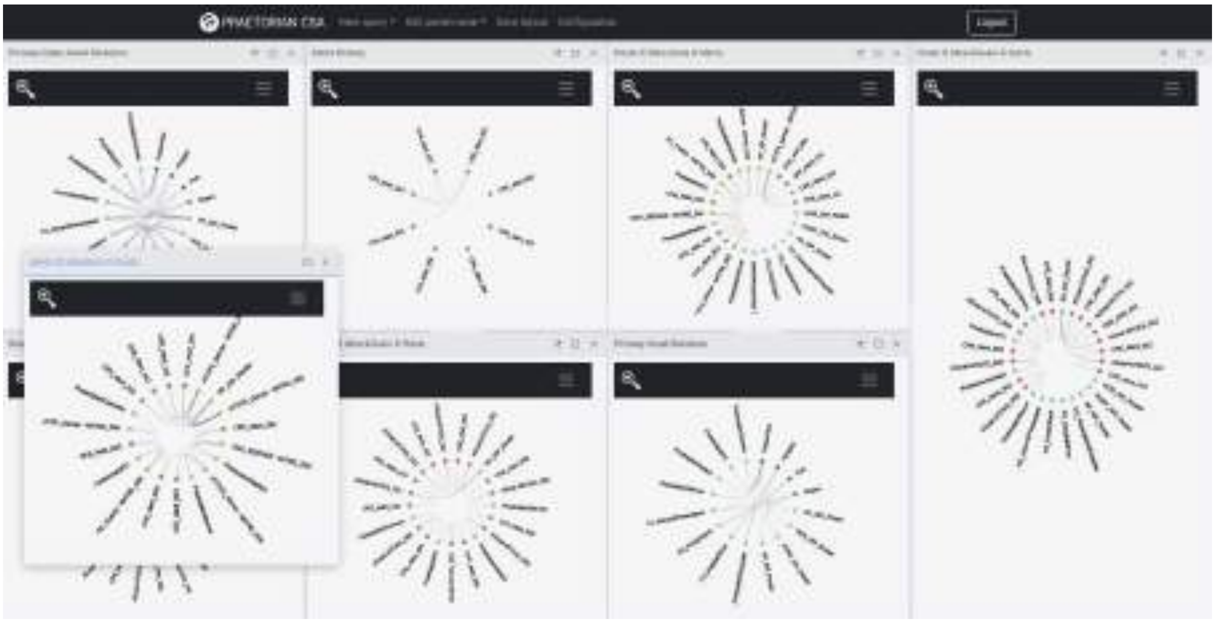


Figure 77 – layout customization VII

Another view customization capability is to rename the panel name, to provide it with a meaningful name that can be useful for the given searches, threat hunting or situation understanding activities instead of the default name the system provides. To do so, the user must click on the drop down menu 'Edit Panel name' and select from the provided list generated by the system from the existing ones.

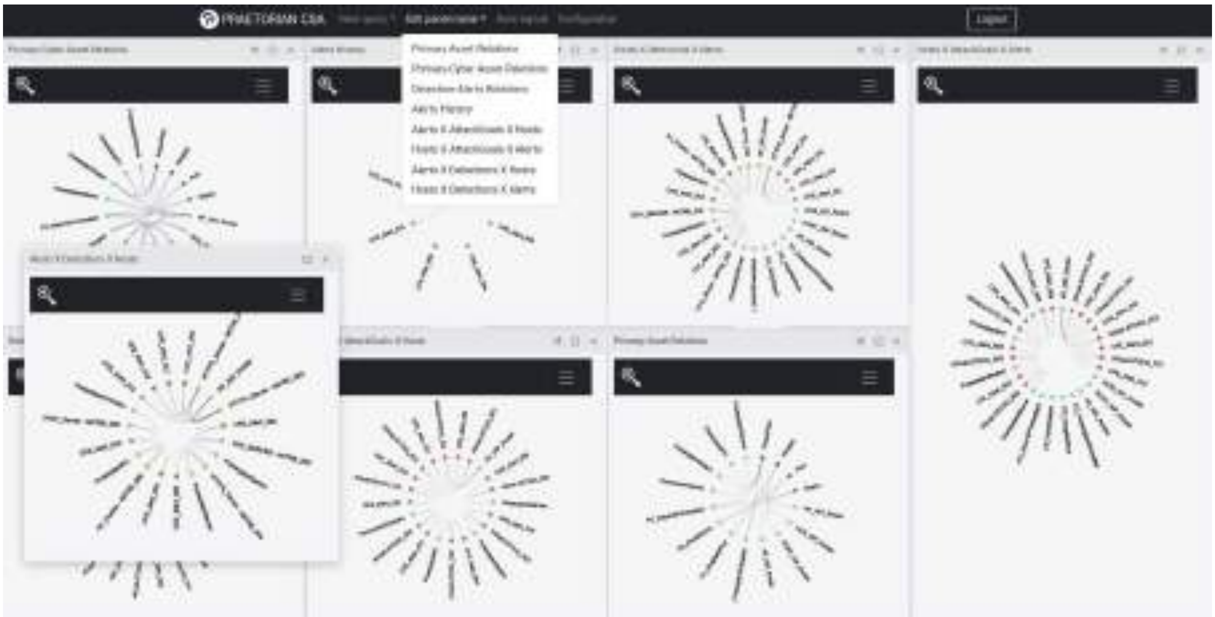
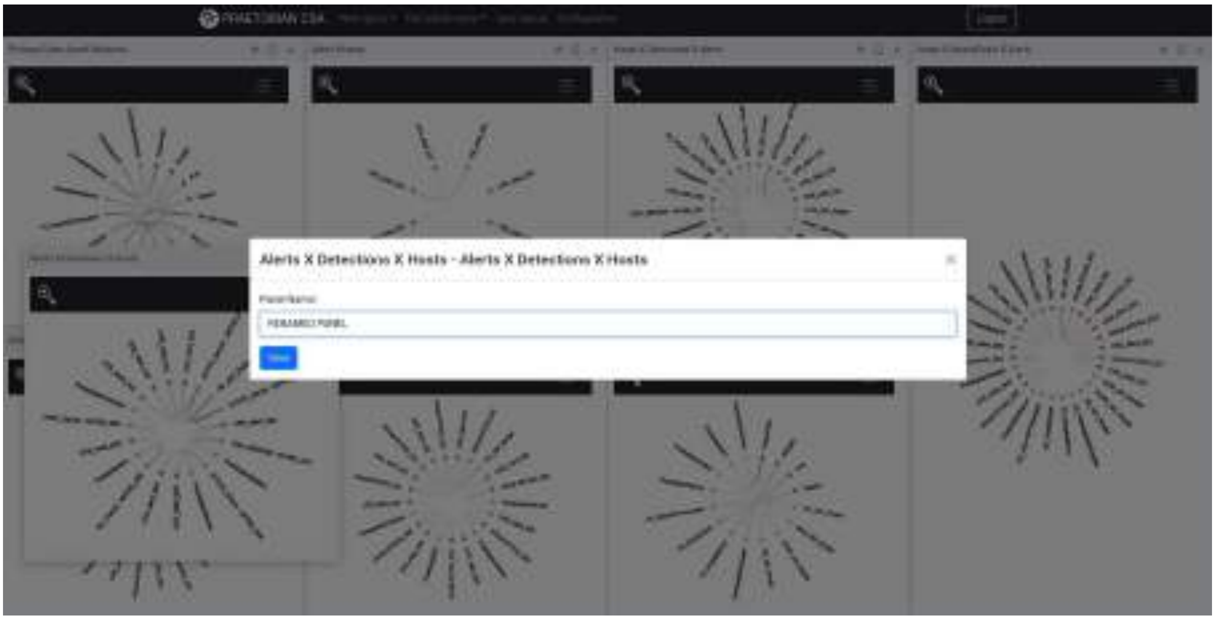

PRAETORIAN CSA
Edit panel name ▾
Primary Asset Relations
Primary Cyber Asset Detections
Detection Alerts Relations
Alerts History
Alerts X AttackGoals X Hosts
Hosts X AttackGoals X Alerts
Alerts X Detections X Hosts
Hosts X Detections X Alerts
Logout

Figure 78 – Panel edition I

This will allow for editing capabilities:


Alerts X Detections X Hosts - Alerts X Detections X Hosts

Panel Name:

RENAMED PANEL

Figure 79 – Panel edition II

Once we are done, we will see the modification.

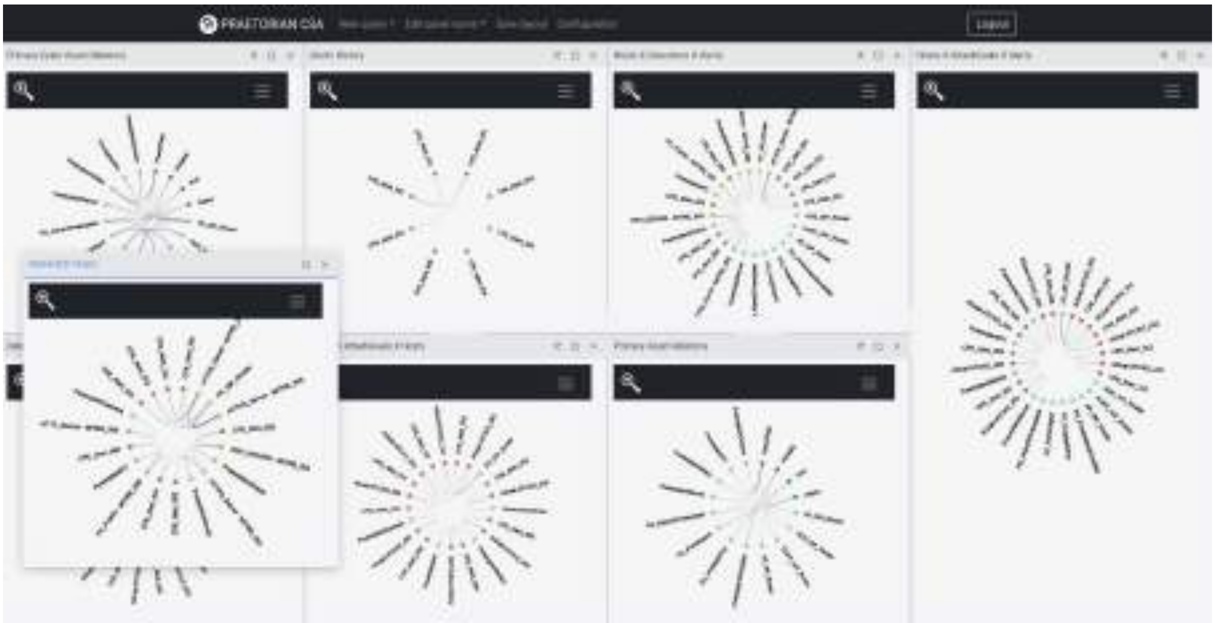


Figure 80 – Panel edition III

If the user is satisfied, he/she can click on save layout. Once done, the given layout will be stored for the given user. At any time the user logs in the system, the HMI will start with the preconfigured panel layout.

5. Conclusions

In the context of task T3.5 "Advanced visualization techniques", this deliverable has presented the HMI application developed for the CSA component of the PRAETORIAN project as part of its WP3 "Cyber Situation Awareness (CSA)". The work presented here depends on the outcomes of both T3.2 "Cyber Forecaster Engine" and T3.3 "Dedicated Cybersecurity Digital twin", as well as it constitutes one of the T3.6 "CSA framework Integration" inputs.

ItsThe main goal of enhancing the situation understanding of the CSA operator, with advanced visualization mechanisms – including 3D interactive representations and immersive reality techniques – has been studied in depth, adapted to the specific needs and use contexts of the project – and, therefore, applied to the development of the CSA HMI. The result is the implementation of an intuitive and novel application that enables the operator to acknowledge the real situation at the cyber domain of the Critical Infrastructures under protection, and validate the veracity of the CSA system's information.

In addition to describing the overall architecture of the CSA HMI tool – and each of its modules –, the document also relates in detail the operational functioning of the system and presents all of its additional functions and features.

The correctness of the development herein achieved will be verified, validated and ultimately demonstrated in the framework of WP7 "Integration and Verification", and WP8 "Demonstration Activities" respectively.

6. References

[AJS] https://angularjs.org/

[ANG] https://angular.io/

[BAB] https://www.babylonjs.com/

[BLA] https://dotnet.microsoft.com/apps/aspnet/web-apps/blazor

[Bow05] Bowden EM, Jung-Beeman M, Fleck J, et al. New approaches to demystifying insight. Trends Cognit Sci 2005; 9: 322–328.

[D2.4] ETRA, "PRAETORIAN toolset architecture implementation description (D2.4)", Protection of Critical Infrastructures from Advanced Cyber-Physical Threats (PRAETORIAN project), Nov. 2021.

[D3.2] THALES, "Cyber Forecaster Engine (D3.2)", Protection of Critical Infrastructures from Advanced Cyber-Physical Threats (PRAETORIAN project), May 2022.

[D3.3] EDF, "Dedicated Cybersecurity Digital twin (D3.3)", Protection of Critical Infrastructures from Advanced Cyber-Physical Threats (PRAETORIAN project), Jul. 2022.

[D3.4] THALES, "Digital twin Cybersecurity Assessment Lab (D3.4)", Protection of Critical Infrastructures from Advanced Cyber-Physical Threats (PRAETORIAN project), Jul. 2022.

[D3.6] THALES, "Cyber Situation Awareness system (D3.6)", Protection of Critical Infrastructures from Advanced Cyber-Physical Threats (PRAETORIAN project), Aug. 2022.

[D3j] https://d3js.org/

[DoA] SU-INFRA01-2018-2019-2020, PRAETORIAN: Protection of Critical Infrastructures from advanced combined cyber and physical threats, N 101021274, Description of actions, 2020

[ELA] https://www.elastic.co/elasticsearch/

[EMB] https://emberjs.com/

[GEP] https://gephi.org/

[GOC] https://developers.google.com/chart

[Gre11] Greitzer FL, Noonan CF and Franklin L. Cognitive Foundations for Visual Analytics, PNNL-20207. Richland, WA: Pacific Northwest National Laboratory, 2011.

[GRJ] http://www.graphicsjs.org/

[GRV] https://www.graphviz.org/

[Hee12] Heer J and Shneiderman B. Interactive dynamics for visual analysis. Commun ACM 2012; 55: 45–54.

[HIG] https://www.highcharts.com

[HIM] https://www.highcharts.com/blog/products/maps/

[Jia22] Jia et al. "Systematic literature review on cyber situational awareness visualizations", IEEE Access, 2022

[JSON] JSON-Schema, https://json-schema.org/.

[Kei08] Keim DA, Mansmann F, Schneidewind J, et al. Visual analytics: scope and challenges, visual data mining: theory, techniques and tools for visual analytics. Lecture Notes in Computer Science (LNCS), Springer, 2008.

[KIB] https://www.elastic.co/kibana/

[Kot15] A. Kott et al. "Cyber Defense and Situational Awareness", Springer 2015

[LAR] https://laravel.com/

[MET] https://www.meteor.com/

[MongoDB] MongoDB database, https://www.mongodb.com/.

[NATS] NATS Technology, https://nats.io/.

[OGL] https://www.opengl.org/

[PAN] https://github.com/De-Panther/unity-webxr-export

[Pat01] Patterson ES, Roth EM and Woods DD. Predicting vulnerabilities in computer-supported inferential analysis under data overload. Cognit Technol Work 2001; 3: 224–237.

[PLC] https://playcanvas.com

[REA] https://es.reactjs.org/

[REV] https://uber.github.io/react-vis/

[Shn96] Shneiderman B. The eyes have it: a task by data type taxonomy for information visualizations. In: proceedings of the IEEE symposium on visual languages, IEEE Computer Society Press, 1996, pp.336–343.

[SYN] https://www.syncfusion.com

[Tho05] Thomas JJ and Cook KA (eds). Illuminating the path: the research and development agenda for visual analytics. IEEE Computer Society Press, 2005. http://www.purdue.edu/discoverypark/vaccine/assets/pdfs/publications/pdf/Illuminating%20the%20Path.pdf

[Tho09] Thomas JJ. Visual analytics techniques that enable knowledge discovery: detect the expected and discover the unexpected. In: ACM SIGKDD workshop on visual analytics and knowledge discovery (VAKD '09), Paris, France, 28 June 2009

[THR] https://threejs.org

[TLS] TLS Protocol, https://datatracker.ietf.org/wg/tls/documents/.

[UNE] https://www.unrealengine.com/en-US

[UNI] https://unity.com/

[VEG] https://vega.github.io/vega-lite/

[VIS] https://visjs.org/

[VJS] https://v3.vuejs.org/

[WBR] https://webvr.info/

[Wer38] Wertheimer M. Laws of organization in perceptual forms. In:Ellis WD (ed.) A source book of gestalt psychology. London:Harcourt, 1938, pp.71–88.

[WXR] https://www.w3.org/TR/webxr/

Annexes