Deliverable D3.8
5G-PPP Security Enablers Documentation (v2.0)
Enabler Trust Builder

Project name	5G Enablers for Network and System Security and Resilience	
Short name	5G-ENSURE	
Grant agreement	671562	
Call	H2020-ICT-2014-2	
Delivery date	31.08.2017	
Dissemination Level:	Public	
Lead beneficiary	NEC	Felix Klaedtke, felix.klaedtke@neclab.eu
Authors	IT INNOV: Juri Papay, Oliver Hayes, Toby Wilkinson	

Document Version	Date	Change(s)	Author(s)
0.1	02.06.2017	Created template	Felix Klaedtke
0.2	31.07.2017	User guide section	Juri Papay
1.0	02.08.2017	Reformatted final version	Juri Papay
1.1	08.08.2017	Formatting and typo fixes	Tommi Pernilä, Aleksi Dahl
2.0	08.08.2017	Typos corrected comments addressed	Juri Papay
3.0	24.08.2017	Updated installation instructions	Oliver Hayes, Toby Wilkinson

Foreword

5G-ENSURE belongs to the first group of EU-funded projects which collaboratively develop 5G under the umbrella of the 5G Infrastructure Public Private Partnership (5G-PPP) in the Horizon 2020 Program. The overall goal of 5G-ENSURE is to deliver strategic impact across technology and business enablement, standardization and vision for a secure, resilient and viable 5G network. The project covers research and innovation - from technical solutions (5G security architecture and testbed with 5G security enablers) to market validation and stakeholder's engagement - spanning various application domains.

The presented document is part of D3.8 and describes the updated version of Trust Builder.

Disclaimer

Copyright notice

Contents

1 Introduction

The notion of trust in 5G mobile networks if one of the hot topics of research. In this respect, we may consider trust between the end users and network operators or trust between the network operators only. Over the years numerous trust models have been suggested, however they must be updated so that these models reflect the requirements of 5G. One of the motivations of our work was to deliver a tool that enables to construct trust networks, to reason about these networks and perform "*what if studies*". The presented Trust Builder uses ontology-based reasoning for analyzing dynamic complex systems. This tool enables the user to identify security threats and take mitigation actions.

Trust Builder is a graphical tool that enables to construct trust networks for representing 5G networks, generate potential threats and validate the model. The output of validation is a modified trust network enriched by features that were not captured by the initial design [1] [2].

For describing the Trust Builder, we use a set of terms for explaining various features if the tool:

- *Core Model* – the core ontology, defining common vocabulary and relationships used in all higher-level models.
- *Generic Model* – an ontology defining the typology of Assets, Threats and Controls (security measures) for a given domain (e.g. 5G networks).
- *Design-Time System Model* – an abstract model of a particular system, described in terms of relationships between system specific Asset classes. The design time model can be enriched by specifying which Security Controls. These controls allow to protect the assets, and generate a set of system-specific Threat Classes for describing potential threats to the system.
- *Runtime System Model* – a model using instances of Assets, Threats and Controls for describing what is known about the current state of the system.
- *Domain Modeler* - a software tool for defining a generic domain model.
- *System Modeler* - a software tool for defining a design-time system model in terms of assets and other elements from a suitable generic model.

The presented document is structured as follows:

- Section 2 is the Installation and Administration Guide that describes the system requirements, configuration, installation and troubleshooting.
- Section 3 is the User and the Programmer Guide for Trust Builder. The User Guide represents the bulk of this document and provides a detailed account of the system's functionality.
- Section 4 describes a case study that illustrates a typical usage of Trust Builder.

The Trust Builder is released as a confidential project output to partners under the terms of the 5G-ENSURE consortium agreement.

2 Installation and Administration Guide

This section describes the system requirements, configuration, installation and administration of "*Trust Builder*" software. Trust Builder is released as a confidential project output to partners under the terms of the 5G-ENSURE consortium agreement.

2.1 System Requirements

This enabler runs on any host (preferably Linux) with Docker installed.

2.2 Enabler Configuration

The default configuration provides a deployment onto a single machine. Account configuration must be done using the default admin account, or an account provided in advance.

2.3 Enabler Installation

Prerequisites:

- Docker

Unzip provided archive and change into the created folder:

```
unzip trust_builder_dist.zip
cd trust_builder_dist
```

Then create 5g-ensure docker network (only once):

```
docker network create -d bridge 5g-ensure-trust-builder
```

Start mongo container:

```
docker run \
--name 5g-ensure-mongo \
--network 5g-ensure-trust-builder \
--rm -ti -p 27017:27017 mongo
```

Confirm that Mongo database server started successfully by locating the following line in the output:

```
2017-08-22T08:46:23.998+0000 I NETWORK  [thread1] waiting for connections
on port 27017
```

Start Trust Builder with the following command:

```
docker run \
--name 5g-trust-builder \
--network 5g-ensure-trust-builder \
--link 5g-ensure-mongo:mongo \
-v "$PWD"/trust-builder.war:/usr/local/tomcat/webapps/trust-
builder.war \
--rm -ti \
-p 8080:8080 \
-e spring.data.mongodb.host=mongo \
```

```
tomcat:7-jre8
```

Confirm that Trust Builder started successfully by locating the following line in the output:

```
08:46:51.432  INFO u.a.s.i.s.s.SystemModellerApplication:57: Started
SystemModellerApplication in 10.075 seconds (JVM running for 14.595)
```

You should be able to access the following endpoint in your browser at this stage:

Trust Builder: http://localhost:8080/trust-builder/

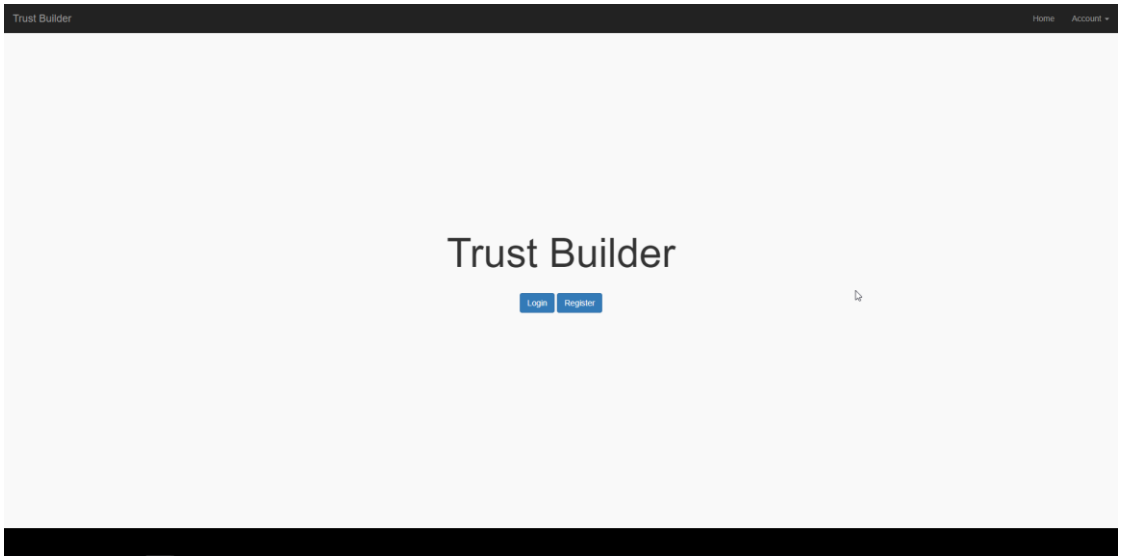


Figure 1 - Trust Builder landing page

2.4 Troubleshooting

The installation procedure declares explicitly the dependencies of the software as standard packages and libraries. The application is composed of a standard MongoDB installation and a web application running on a Tomcat server. Any errors are captured in the standard log files (e.g. /var/log/tomcat7/catalina.out) after the application war file has been deployed.

3 User and Programmer Guide

The presented User Manual described the functionality of the Trust Builder developed in the 5G-Ensure project. The presented Trust Builder uses ontology-based reasoning that allows to analyze complex systems, identify security threats and measures to counter the threats.

3.1 User Guide

In this section, we provide definitions of the main concepts, such as:

a) *Asset* - is an element of the network,
b) *Stakeholder* - can be a person or an organization, i.e. an entity that can carry out actions
c) *Process* - usually represented by a software

d) *Network asset* - an element of the infrastructure or environment).

The term *Misbehavior* represents different ways in which assets may be compromised as a consequence of an active threat. An "involved" asset is an asset which forms part of a pattern and the presence of which is necessary for a threat to occur or to be managed. A *Control Strategy* is a set of controls located at different assets that block or mitigate a threat.

The following sections provide details of the system's functionality covering:

- User management
- Model management
- Model editing
    a. Stage 1: defining assets and relationships which provide the initial model of a network
    b. Stage 2: validation and auto-generation of threats
    c. Stage 3: defining threat management strategy (selecting controls for assets or control strategies for threats)
- Model outputs

The modelling process has three stages that can be repeated several times. First the user constructs the model by putting assets into the modelling panel and sets the links between the assets. The user defined assets are called "*asserted asset*s". The validation process in Stage 2 automatically generates inferred assets, threats and security controls to counteract the threats. The validation process also determines whether the information provided about the assets and relationships is consistent and complete. If the validation fails i.e. the model gets marked as 'invalid' then the user should go back to Stage 1 and update the model so that it contains sufficient information for a successful validation. In Stage 3 the user addresses threats by selecting or modifying the set of security controls that protect the assets in the system. The aim is to eliminate or at least mitigate the threats.

3.1.1 User Management

3.1.1.1 Main page

On the main page of Trust Builder (Figure 2) there are several links *Trust Builder*, *Home* and a drop-down menu under *Account*. The *Trust Builder* and *Home* links take the user to the main page of "*Trust Builder*". The options under the *Account* dropdown is *Sign In, Register and Forgot Password.*

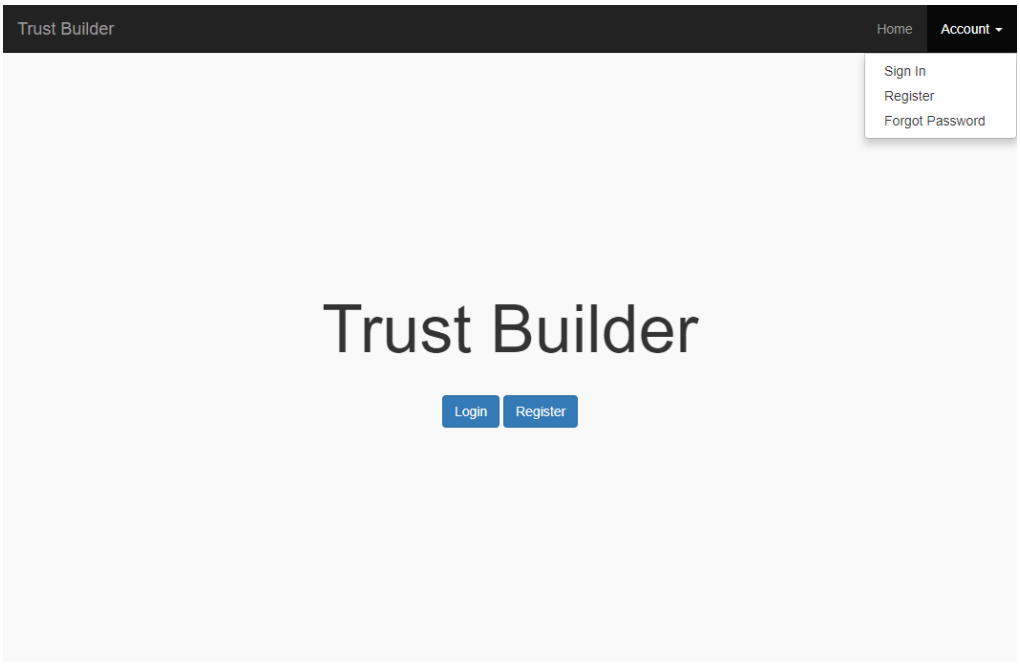


Figure 2 - Trust Builder main page

3.1.1.2 User login

The login page of Trust Builder is activated either by clicking on *Sign In* link in the dropdown menu or by clicking on the *Login* button (see Figure 3). The user must enter their username and password. These are case-sensitive.

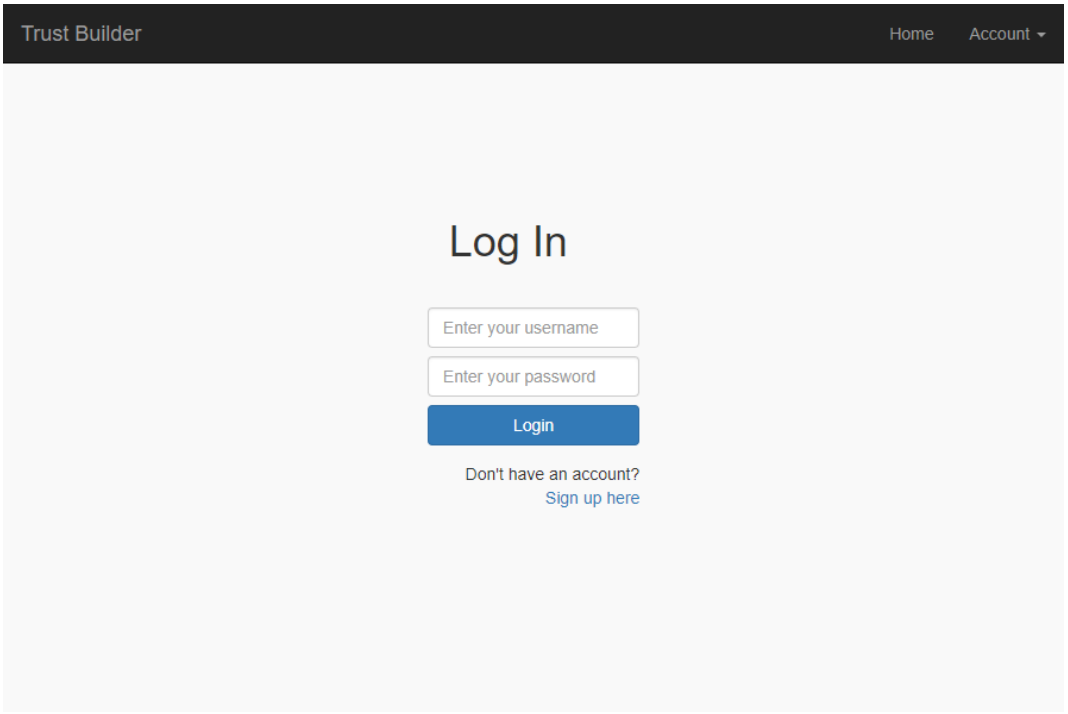


Figure 3 - User login page

3.1.1.3 User registration

The user can register by providing email address and password with their full name (see Figure 4). On registration, the user's account is still inactive. The system administrator needs to activate the account before the user is able to login. The user is notified via email about activating the account.

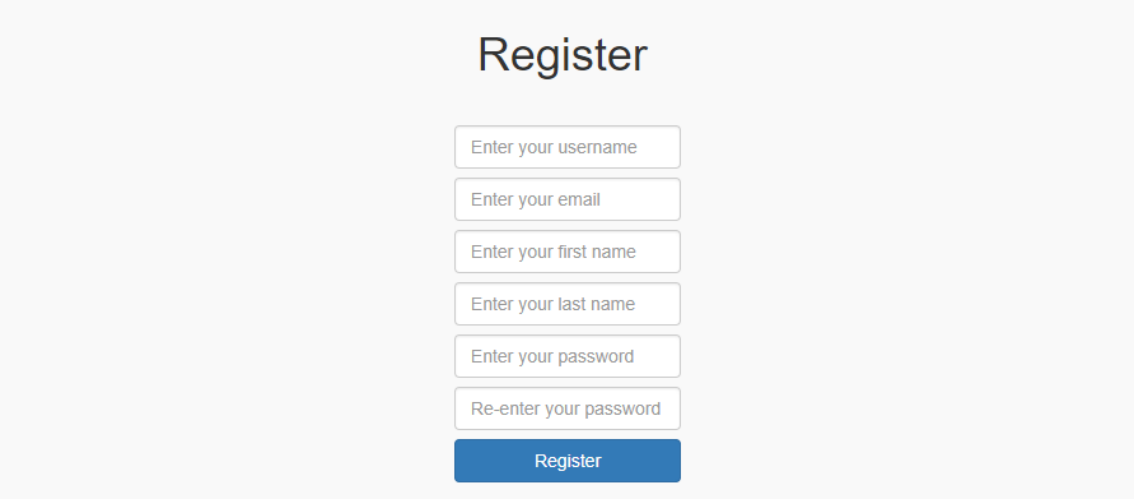


Figure 4 - User registration

3.1.1.4 Password management

The user can reset the password by clicking on the *Forgot Password* link in the dropdown menu (see Figure 5). The username must be a valid email address.

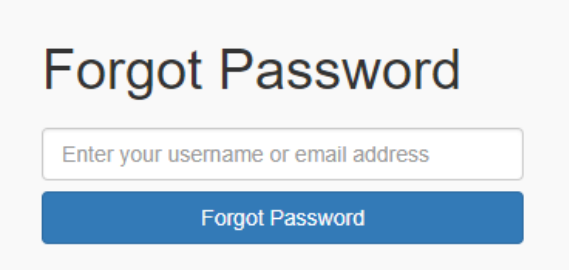


Figure 5 – Forgotten password

After typing in the user id and clicking on the "*Forgot Password*" button the user is presented with the "*Reset Password*" page (see Figure 6).

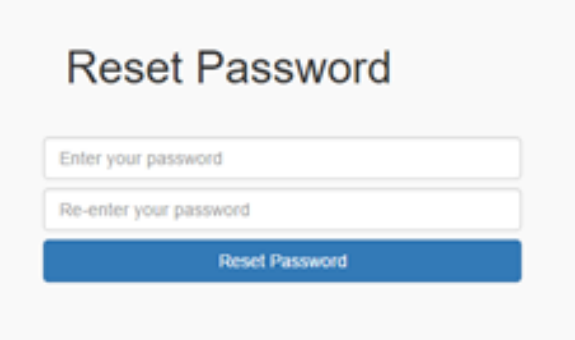


Figure 6 - Password reset

3.1.1.5 Logout

Logout is activated by clicking on the *Sign Out* link in the dropdown menu on the main page under the currently logged in user.

3.1.2 Model management

The term Model management incorporates several functions such as:

a) Listing models created by the user or shared by others
b) Creating models
c) Importing/Exporting Models
d) Managing access to models
e) Deleting models
f) Checking in/out models

As of the second release of Trust Builder, not all of the features listed above are fully implemented. Checking in/out models is not currently implemented.

3.1.2.1 List models

After a successful login, the user can view the models that the user either owns or has read/write access to (see Figure 7).

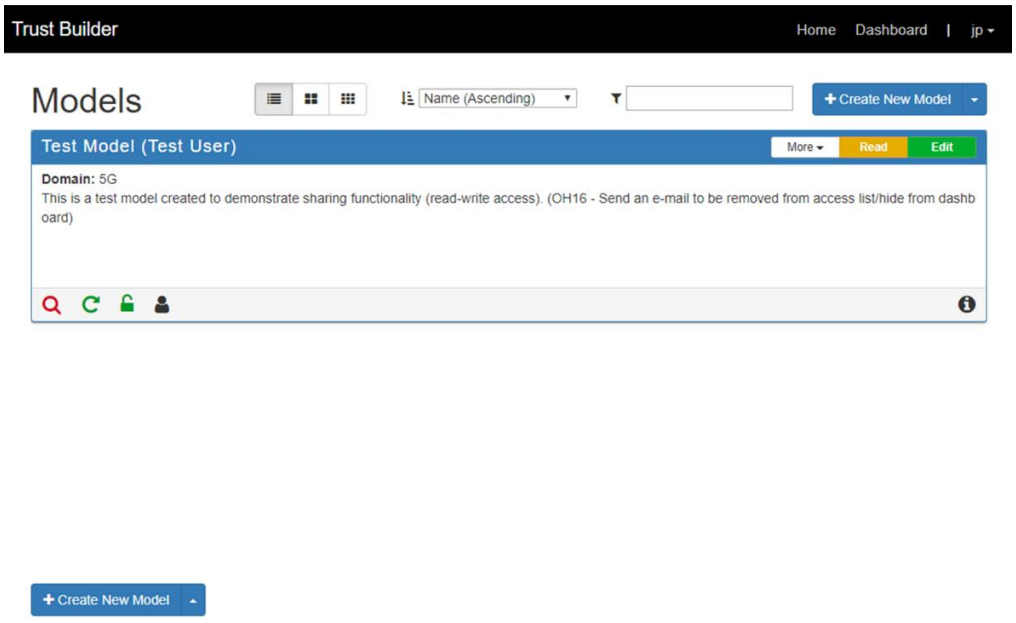


Figure 7 - Viewing the models

In Figure 7 we can see the *Test Model* that is used for demonstrating the sharing functionality when several users can edit the same model. In the center of the model, the domain used by the model is labeled. There is also a description of the model, which can be edited using the functions described in Figure 8. At the bottom of the model window there are five icons that reflect the status of the model. For example, if the "magnifying glass" icon is red it means that for the given model no report was generated (see Table 1).

Icon	Description
	indicates if a report was generated (green color) or not (red color)

	indicates if the model was validated (green color) or not (red color)
	indicates if the model is not locked (green) or locked (red)
	last modification of the model
	when the model was created

Table 1 - Model status

The drop-down menu in the top right corner of the model window offers several functions, these are: View, Delete, Export, Edit Details, Share, Copy and Lock (see Figure 8).

As of the second release of Trust Builder, copying and locking models is not currently implemented.

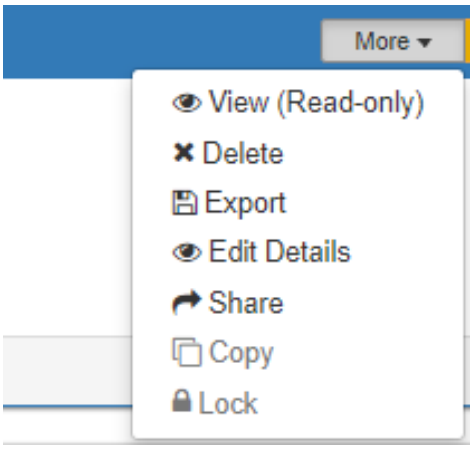


Figure 8 - Model drop-down menu

3.1.2.2 Create model

By clicking on the "*Create New Model*" the user can create an empty model (see Figure 9**Error! Reference source not found.**). The drop-down selection allows the user to choose which generic model to use. The user automatically becomes an owner of the newly created model with read/write access and with the ability of granting/revoking access rights for other users. The new model is added to the model list and it can be edited, viewed, or shared.

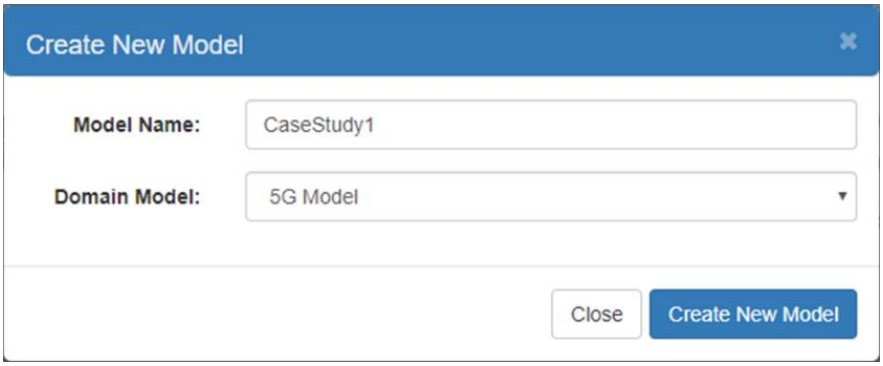


Figure 9 - Creating a new model

The "*Import Existing Model*" allows to import a model from a file. The "*Create Model from Template*" is not yet implemented.

3.1.2.3 Export/Import

Once we have constructed the model it can be exported into a file. By clicking on the "*Configure model*" control (see Table 2) a dialog opens that allows to export the model (see Figure 10**Error! Reference source not found.**).

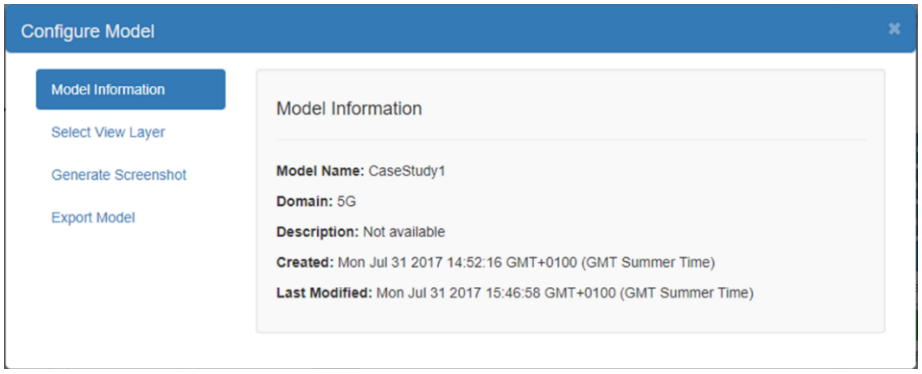


Figure 10 - Exporting a model into a file

By clicking on the "*Export Model*" link we can save the model in a file (see Figure 11**Error! Reference source not found.**), the model gets exported into "*Downloads*" directory.

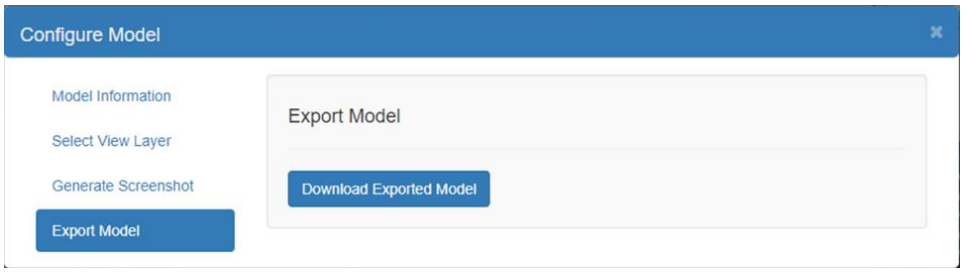


Figure 11 - Download the model into a file

The *Importing* operation allows to upload a previously saved file into the modelling canvas. For importing an existing model, we click on the "*Create New Model*" control and select the "*Import Existing Model*" option (see Figure 12**Error! Reference source not found.**). If restoring a previous version of a model, the user can check *Overwrite existing model*. Attempting to import the same model without this being checked will result in an error. A user can re-import an existing model with a different name by checking *New Name.*

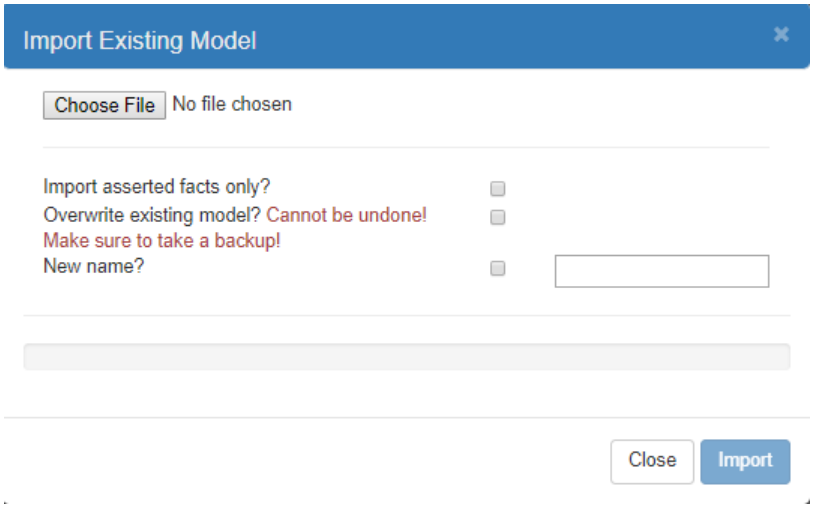


Figure 12 - Importing a model from a file

3.1.2.4 Managing access to models

By clicking share on a model tab (see Figure 8) the user can control access capabilities of users (see Figure 13).

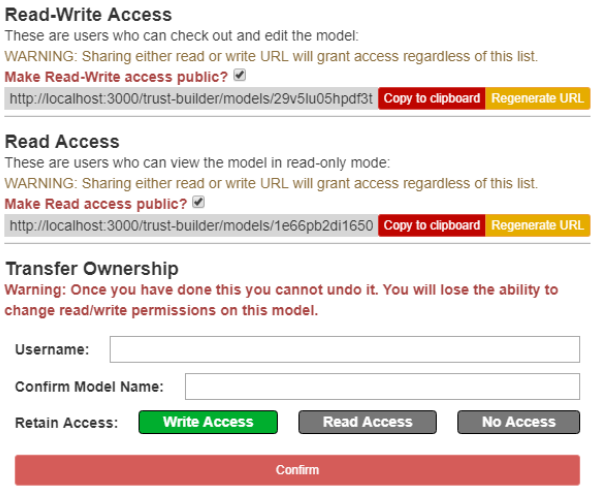


Figure 13 - Model sharing

The user can control two access lists: read-write and read; both are public by default meaning any user with the link has access. Unchecking the public flag presents the user with a username list, within which they can add the usernames to grant capability (see Figure 14). These models will then appear in that user's dashboard. A user must contact the owner of the model to be removed from the access list. Additionally, a username may only appear in one of the lists to maintain integrity.

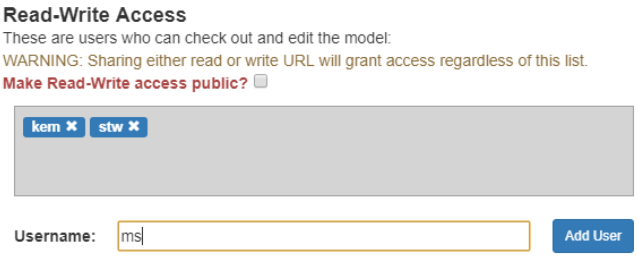


Figure 14 - Capability granting

A user can also transfer ownership of the model and retain access on either list (see Figure 15). This change cannot be reverted once confirmed. The user must type the username of the new owner, the model name in full, and what access they want to retain.

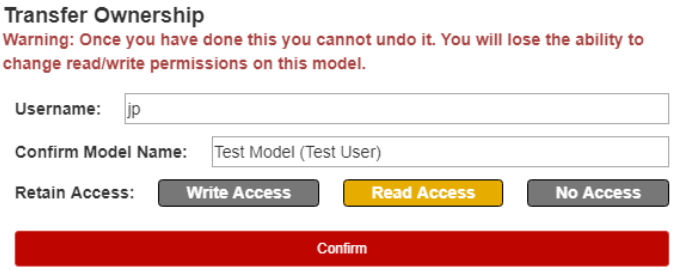


Figure 15 - Transferring model ownership

3.1.2.5 Delete model

The delete action removes the model along and also the associated resources (e.g. assets, relationships) including access rights. The precondition for the delete operation is that the model should not be checked out by another user (this feature is not yet implemented).

3.1.3 Model construction

Clicking the *Edit* button opens the editing panel that consists of three parts. On the left side, there is the "*Asset Palette*", in the middle there is the *Model Construction Canvas*, the right-side panel contains various categories related to assets, such as *Data Properties*, *Incoming Relations, Outgoing Relations*, *Control Sets*, *Threats* and *Misbehaviors* (see Figure 16).

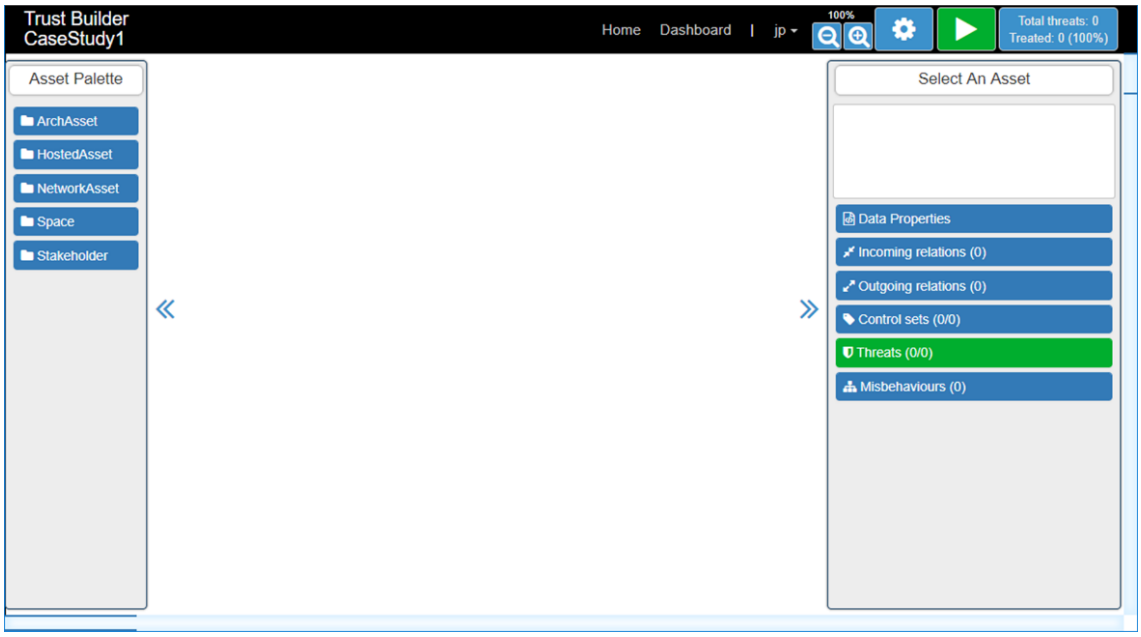


Figure 16 - Model editing

The top right corner contains four buttons, these are in Table 2

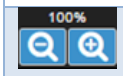
Icon	Description
	Process (validate) the model
	Configure the model
	Zoom controls

Table 2 - Model editing controls

3.1.3.1 Select and add asserted asset

The left panel of the model editor contains various assets, these fall into three categories:

a) ArchAsset (for illustration see Figure 17)
b) HostedAsset
c) NetworkAsset
d) Space
e) Stakeholders

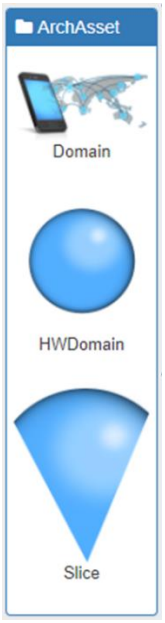


Figure 17 - Selecting items from Architecture Assets

An asset can be inserted in the model by selecting an icon in left panel and dragging it into the model canvas (see Figure 18). By clicking on the asset, the user can view/edit the properties, these are:

a) name of the asset
b) incoming relations
c) outgoing relations
d) inferred relations
e) control set
f) threats (once the model has been validated)

For illustration purposes, we analyze a typical use case representing the threats associated with accessing a web page hosted on a remote server. Constructing a security model involves placing assets on the canvas and establishing connections between them. The model itself consist of two hosts connected to the internet. The user on Host1 accesses a WebService deployed on Host2. This is a simple model represents downloading a web page from the remote web server. The reasons for selecting this simple model are as follows:

a) Frequently occurring case, typical for all web applications
b) Simplicity
c) All types of inference can be well demonstrated, these are:
    - Inferred assets
    - Inferred relationships
    - Mandatory relationships
d) The threats inherent to the model are well understood
e) The effect of controls and threats can be easily interpreted

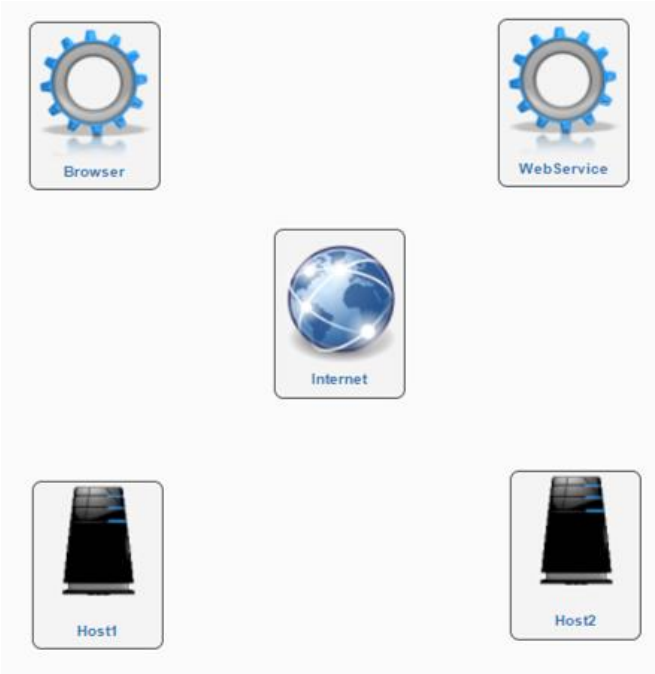


Figure 18 - Inserting assets into model canvas

3.1.3.2 Add relationship between assets

Once the assets have been put on the modelling canvas the user can connect two assets by establishing a link between them by clicking on the green cross that appears in the left corner (Figure 19).

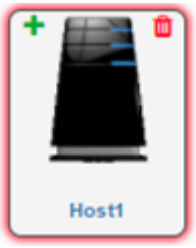


Figure 19 - Connecting assets

By clicking on the green **+** sign of the asset (in our example Host1) the blue tick sign will appear on several assets indicating that a link can be made to these assets (see Figure 20).

Figure 20 - Target assets for making connections

By clicking on the blue tick icons, we can establish connections between assets (Figure 21).

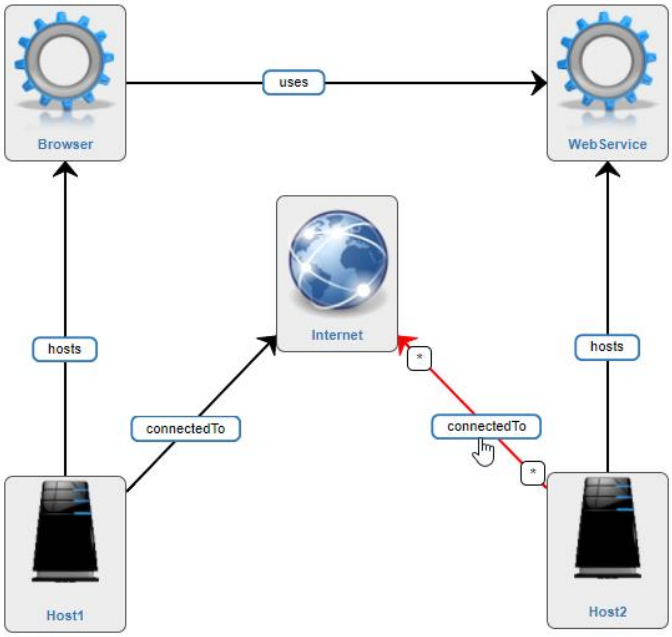


Figure 21 - Connecting assets

The cardinality of assets can be set by selecting an asset and editing the cardinality fields. The cardinality of the connection can be set by right clicking on the connection and setting the cardinality values (see Figure 22). By default, the cardinalities are infinite (marked by *).

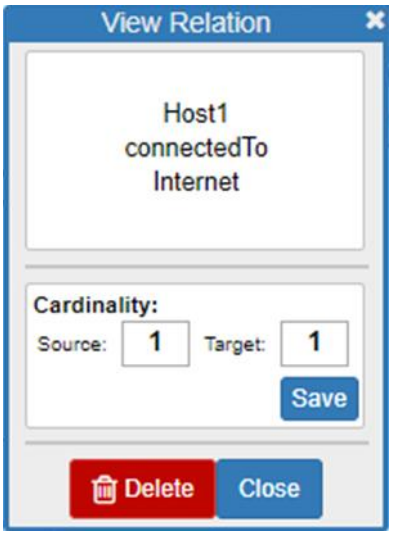


Figure 22 -Setting the cardinality of connections

3.1.3.3 Remove asset

Assets can be removed by clicking on the red trash icon of the asset in the top right corner (see Figure 19). The delete operation removes all links between the selected asset and other assets.

3.1.3.4 Remove relationship

By right clicking on the connection between two assets an option with the delete button comes up (see Figure 22). The delete applies only on the relationship the assets remains on the canvas.

3.1.3.5 Rename asset

The user can rename an existing asset by editing the asset's name under the corresponding icon. NB by changing the name the asset's connections will stay unaffected. All asset names must be unique.

3.1.4 Model Validation

Once the model is constructed it can be validated. This operation is activated by clicking on the red "play" button (see Table 2) which indicates that the model is currently invalid. The validation operation runs semantic reasoning that generates inferred assets that are added to the model and produces a list of threats that can be associated with the given model. This operation guarantees that the inferred assets are consistent with the asserted assets and relationships. On completion of the validation operation the updated model is presented to the user (see Figure 23).

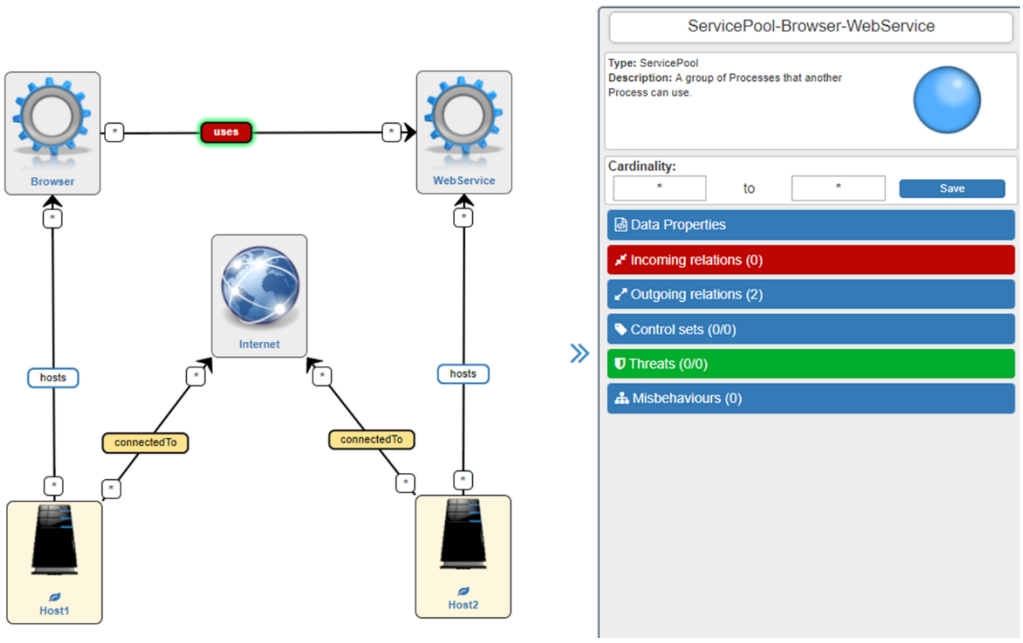


Figure 23 - Model after validation

The validation in essence generates entities that were missing from the initial model. Then the user needs to configure the new entities (e.g. inferred assets and inferred relationships). In our case there are some issues with the "*uses*" connection between the *Browser* and *Web Service*.

The validation operation can take some time depending on the complexity of the model. The outcome of validation is indicated by changing the color of assets. A green color indicates that the validation has succeeded, otherwise the color is red. All issues identified by the validation operation must be addressed and the model validated again.

3.1.5 Addressing Incoming Relations

We select the "*uses*" connection between the *Browser* and *WebService* and click on "*Incoming Relations*". Under the "*Incoming Relations*" the user can see a list of incomplete relations that need to be specified by clicking on the "?" (see Figure 24).

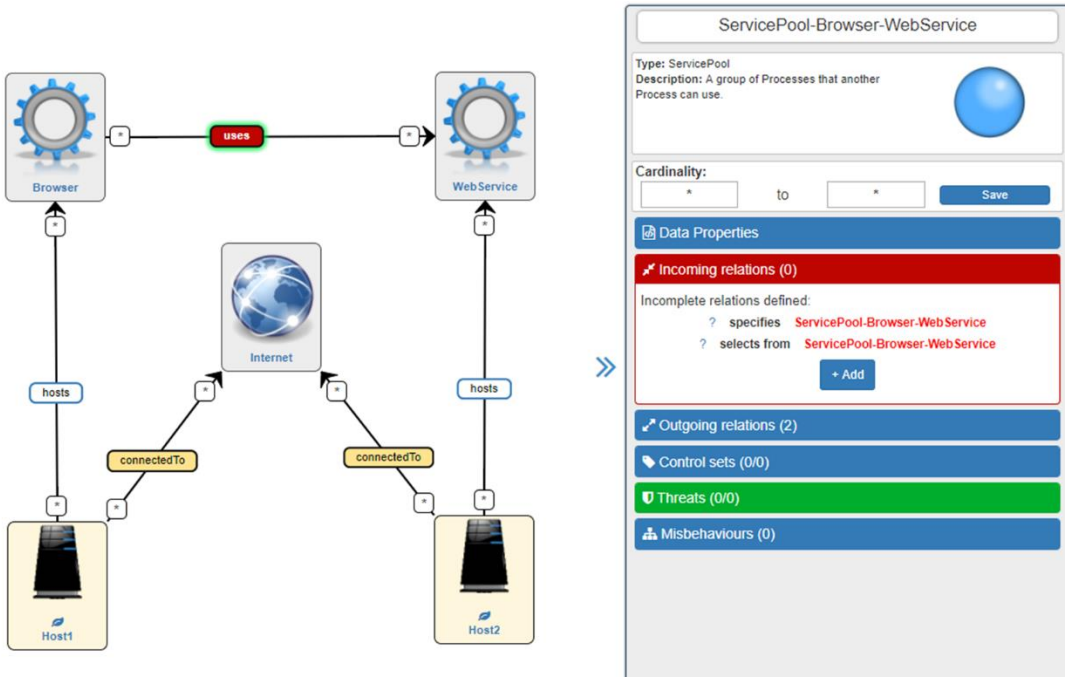


Figure 24 - Incomplete relations

The blue question marks indicate missing information. By clicking on the second question mark in front of "*selects from*" we can specify that the Browser "*selectsFrom*" the Service pool (inferred asset), see Figure 25. After the selection is made "Save changes".

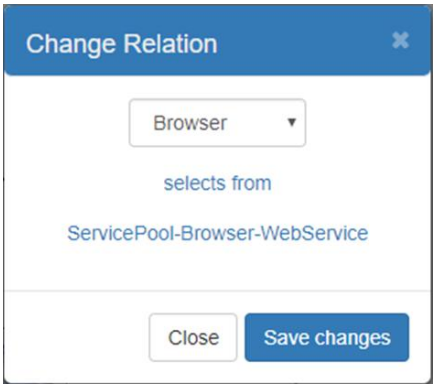


Figure 25 - Establish connection between Browser and Service Pool

By clicking on the first question mark in front of "*specifies*" we can specify the relationship between the *WebService* and *Service Pool* (inferred asset), see Figure 26. In this case the *WebService* specifies the *Service Pool*. After making the selection click on "*Save changes*".

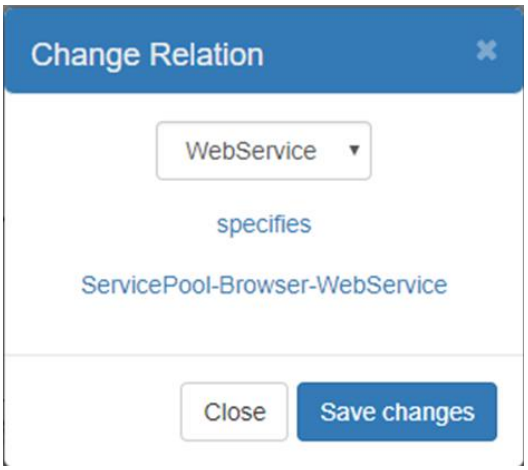


Figure 26 - WebService specifies Service Pool

After establishing connections to the *Service Pool* (inferred asset) the background color of Incoming Relations should turn blue (see Figure 27).

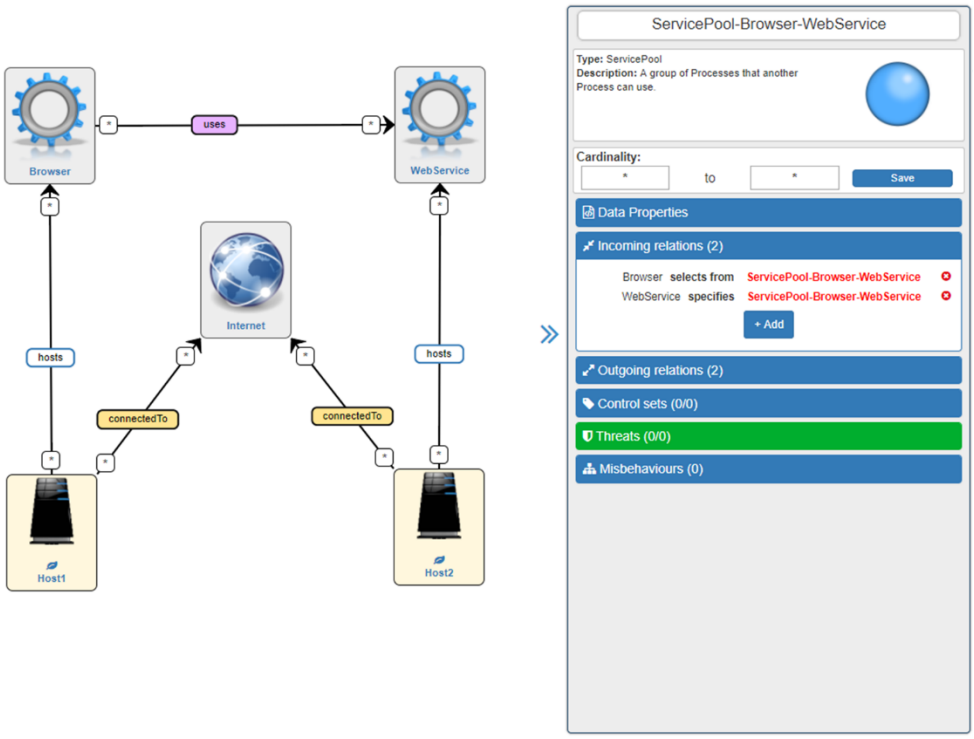


Figure 27 - Setting all Incoming Relations

In the following step, we validate the model again by clicking on the red play button in the top right corner of the screen (see Figure 28).

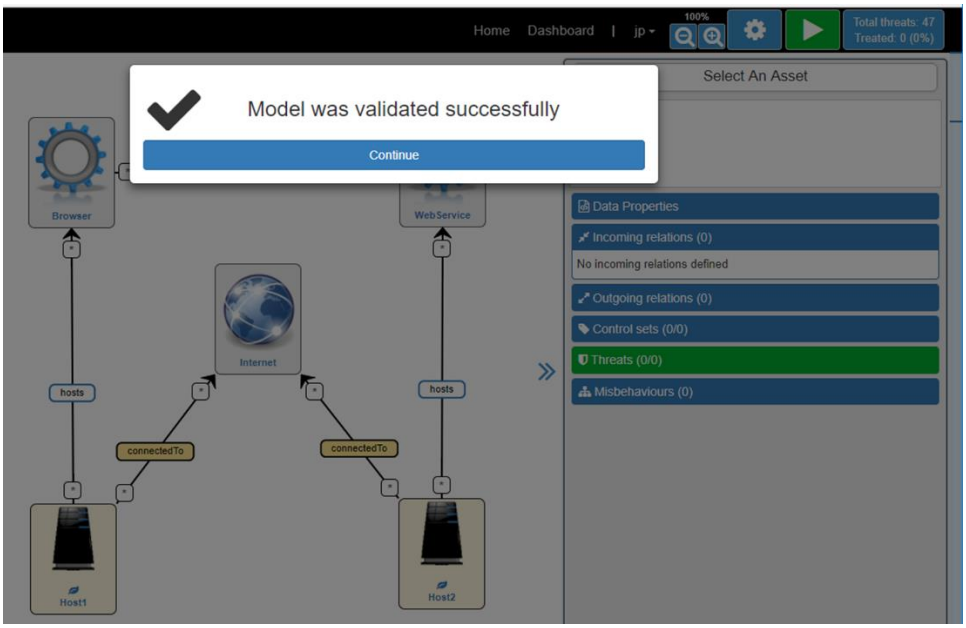


Figure 28 - Successful validation of the model

3.1.6 Threat management

3.1.6.1 Threats associated with the given asset

The user can view the threats associated with the given asset. First the user needs to select an asset then click on the "*Threats*" button on the right panel (see Figure 29). This panel also provides options for editing the incoming/outgoing connections, control strategies to address a specific threat and control sets that make up the control strategy.

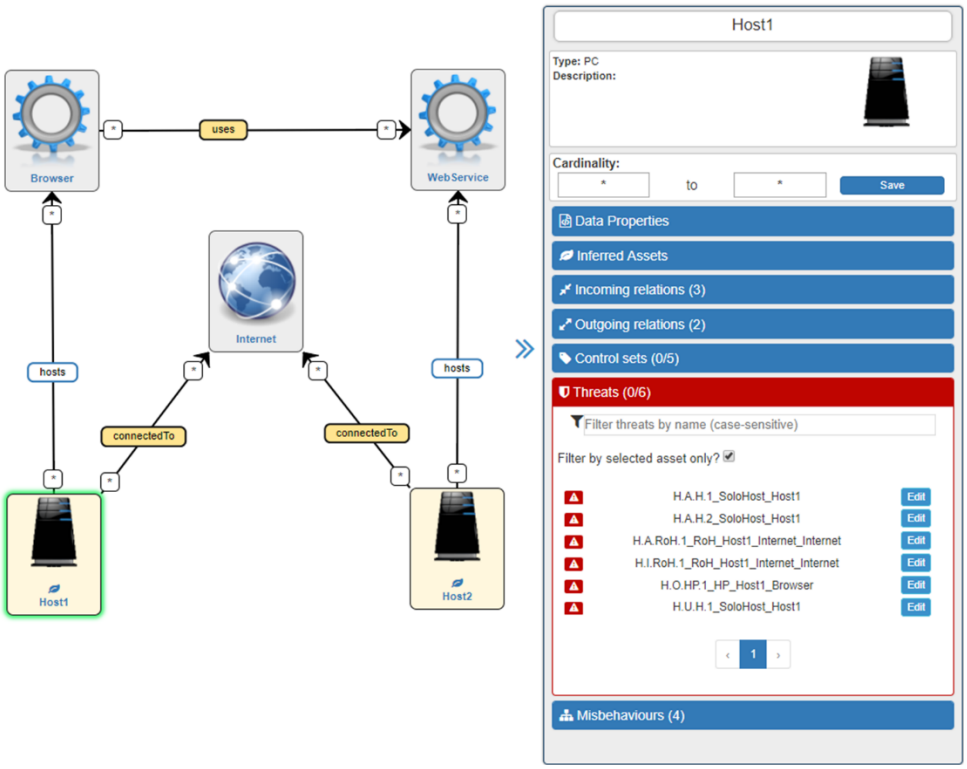


Figure 29 - List of threats associated with Host1

At this stage, none of the threats has been addressed, so the status icon is red. For example, we may select threat "*H.A.RoH.1_Host1_Internet_Internet at Host1*" (see Figure 30). Clicking the Edit button brings up the *Threat Editor* that allows to configure various parameters for the given threat.

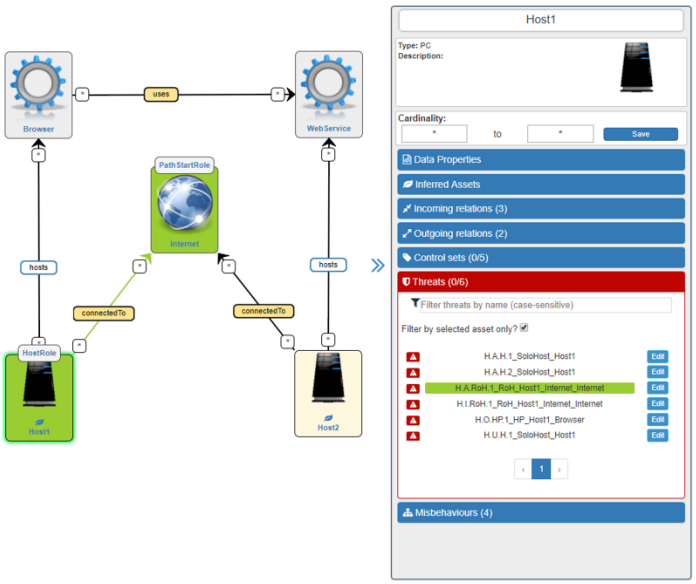


Figure 30 - Threat selection

3.1.6.2 Selecting controls in the Control set

The threats can be resolved by selecting controls under the *Control Set* tab. The controls that are available for Host 1 are given in Figure 31.

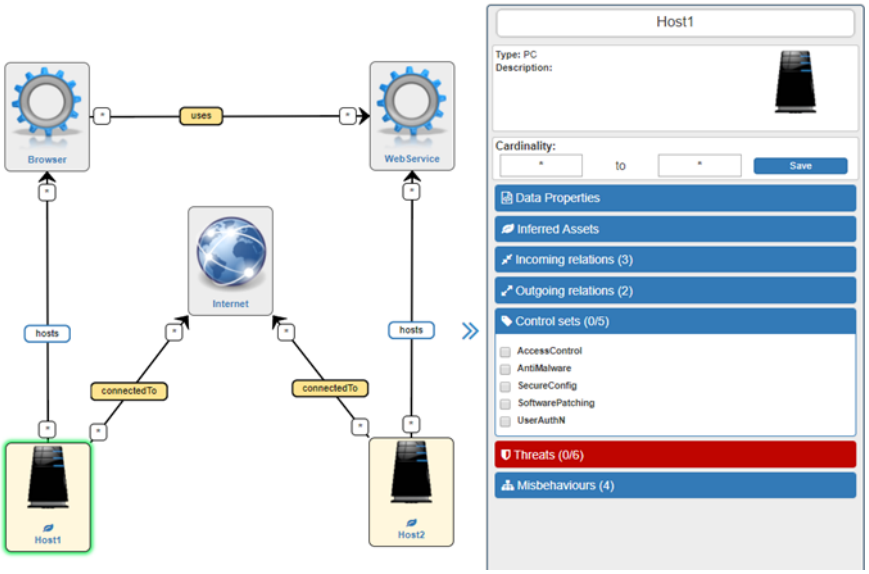


Figure 31 - Selecting Control Set properties for Host1

For the purpose of demonstration, we can select two options (*Software Patching*, *Anti Malware*) and see which threats will be resolved. These threats are indicated by green color (see Figure 32).

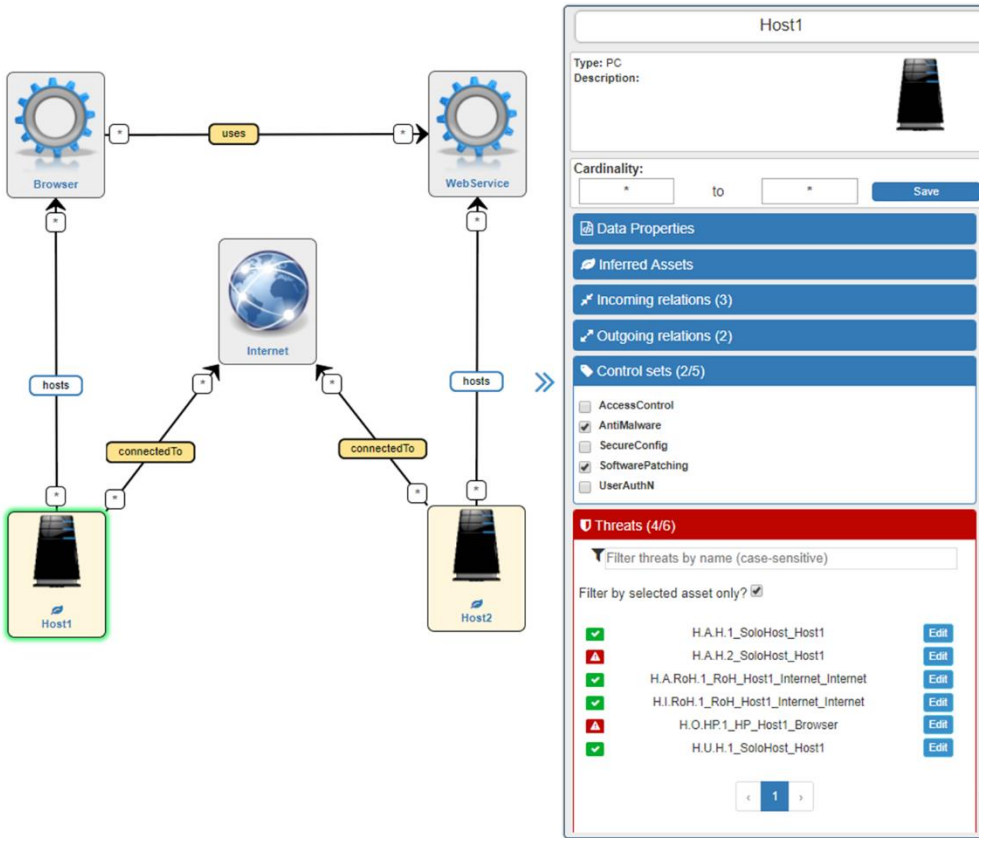


Figure 32 - Resolving threats

Figure 32 shows that five threats have been resolved as a result of selecting *Software Patching* and A*nti Malware* from the *Control set*. Resolving the threats is an iterative process, the user needs to go through the assets one by one, selecting the options from the Control Set and checking which threats have been eliminated. This User Guide describes the threat resolution steps for one asset (Host1) but the same steps are applicable to all assets. By following these steps, the user should be able to resolve or at least mitigate all threats associated with the given model.

3.1.7 Threat editor

3.1.7.1 Using the threat editor to select controls

The previous section described how to address threats by applying controls to assets directly, this is particularly useful for experienced users who know about the effects that the controls have on the assets. Unexperienced users however need some guidance on how a threat can be addressed. This is done using *Control Strategies*, which are essentially collections of *Control Sets*. An active *Control Strategy* manages a threat. While a *Control Set* describes a Control located at a particular Asset, a *Control Strategy* contains 1 to *n Control Sets*. Semantically this means that for the threat to be managed (blocked or mitigated), all *Control Sets* in the *Control Strategy* must be applied. The domain models contain mappings between Threats and Control Strategies that are made visible in the threat editor, see Figure 33.

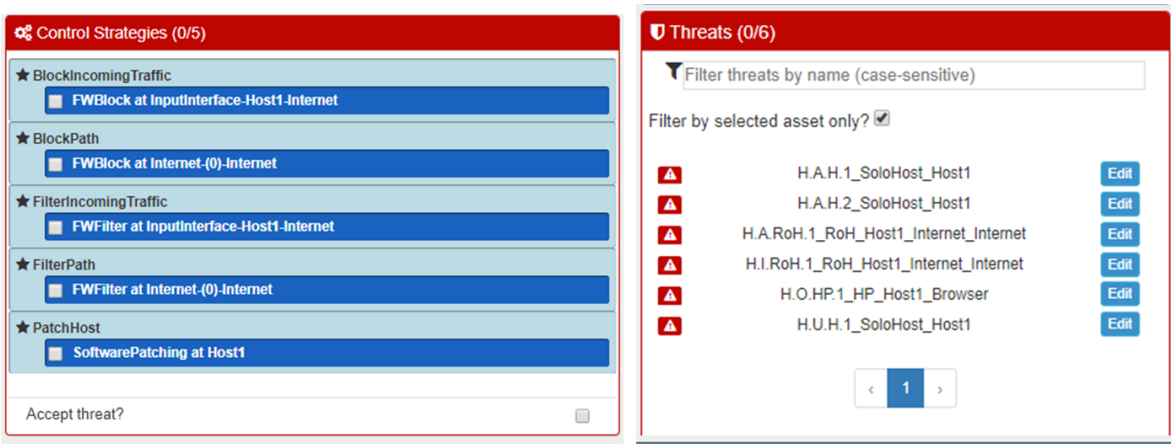


Figure 33 - Control Strategies in the Threat Editor

If any of the *Control Strategies* for a Threat are active, it means that the threat is managed. Managed threats appear in green. Hovering over the threat icon inside the panel shows the management type (e.g. “blocked”), see Figure 34.

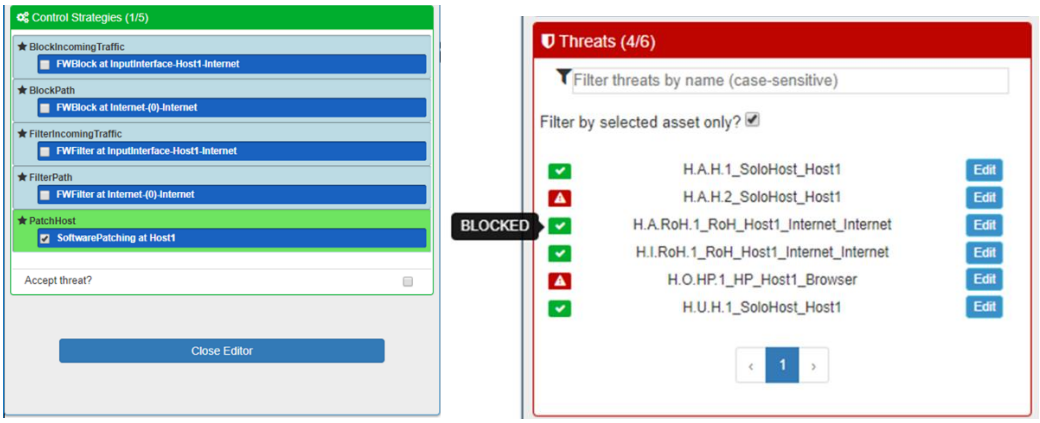


Figure 34 - Active Control Strategy

3.1.7.2 Accepting a threat

In cases where there is no control strategy exists for the given threat (see Figure 35) or the control strategy would be difficult or expensive to implement the user can accept the threat.

Figure 35 - Accepting threats

Accepting a threat means that the user accepts the risk posed by the threat (see Figure 36). The user can also type in the reason for accepting the threat.

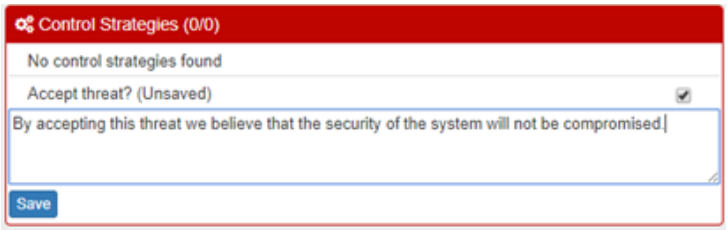


Figure 36 – Accepting a threat

Upon saving, the icon for the accepted threat will change to indicate the change of status (see Figure 37).

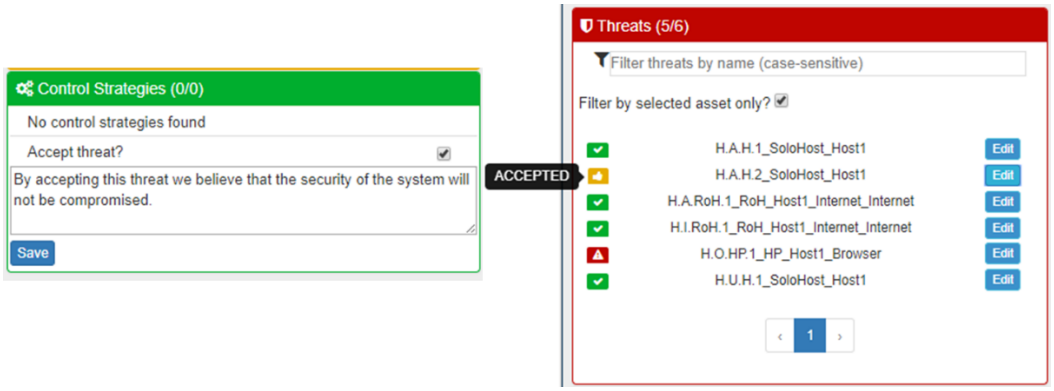


Figure 37 - The UI after accepting a threat

3.1.7.3 Breaking a pattern

In some cases, specifically for compliance threats (threats that are required for a system to be compliant with a set of regulations), accepting a threat is not an option and if there are no control strategies, the system would not be compliant. The only way to prevent a threat from appearing in the first place is to "*break*" the pattern.

The UI provides a user-friendly way of achieving this in the threat editor's "*Applies to pattern*" widget as shown in Figure 38. The user is presented with the relations that make up the pattern and can choose one from the list to delete.

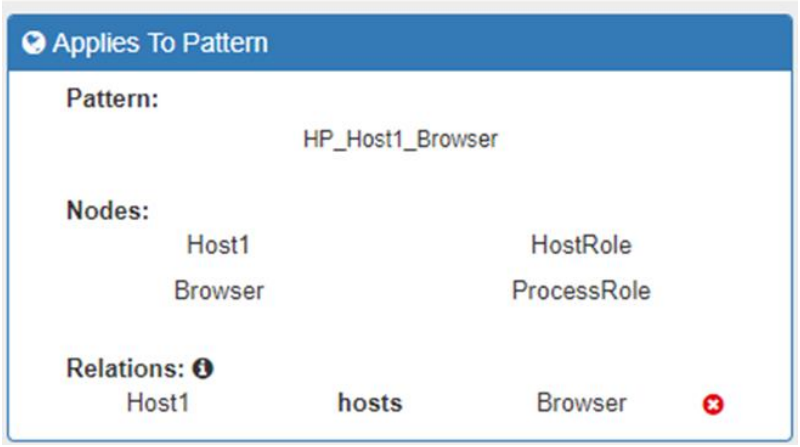


Figure 38 - Breaking a pattern

The broken pattern with the deleted relation is crossed out (see Figure 39). For updating the list of threats, the model needs to be validated again.

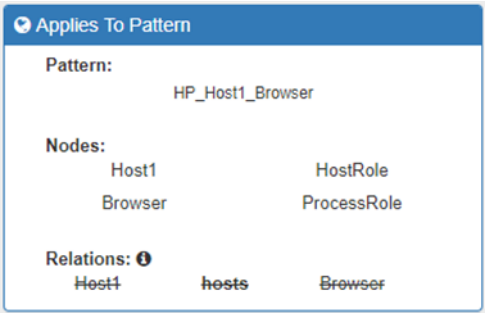


Figure 39 - A broken pattern

3.1.7.4 Causes and effects

Figure 40 shows the Cause/Effect/Secondary Effect widgets of the Threat Editor.

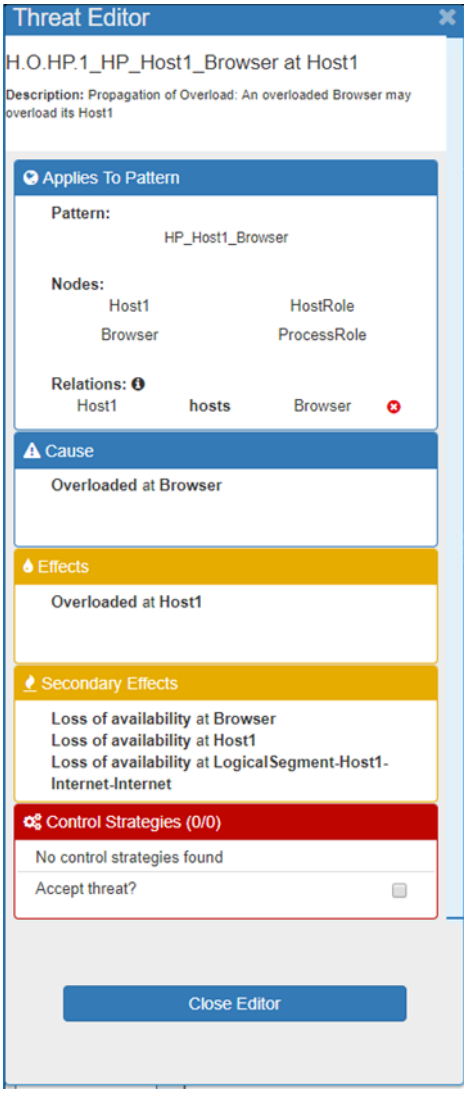


Figure 40 - Exploring secondary effects

While *Primary Threats* occur independently, *Secondary Effects* are triggered by the existing *Misbehaviors* of the System. If the *Cause* widget contains any *Misbehavior Sets*, the threat is a *Secondary effe*ct. When the validated system contains multiple *Secondary Effects,* they get linked up forming S*econdary Effect Chains*. *The Secondary Effects* widget displays all *Misbehaviors* that can be caused by *Secondary Effects* triggered by the current Threat.

3.1.7.5 Root cause analysis

Secondary effect chains can be explored in both directions. While the *Secondary Effect* widget allows a user to go “forward” in a *Secondary Effect Chain*, the *Misbehaviors* widget in the sidebar allows to look “back” to find the root cause for a threat.

Figure 41 - The Misbehaviors widget

Figure 41 shows all *Misbehaviors* for the Host1 asset. These are all possible *Misbehaviors*, caused by the threats in the system. When a *Misbehavior* is selected, the *Misbehavior Explorer* opens in a new window, see Figure 42. It contains a widget which shows *Direct Causes*, i.e. Threats that cause this *Misbehavior* (as their Effect). The other widget shows "Root Causes", i.e. any Threat on a *Secondary Effect Chain* that can cause this *Misbehavior*.

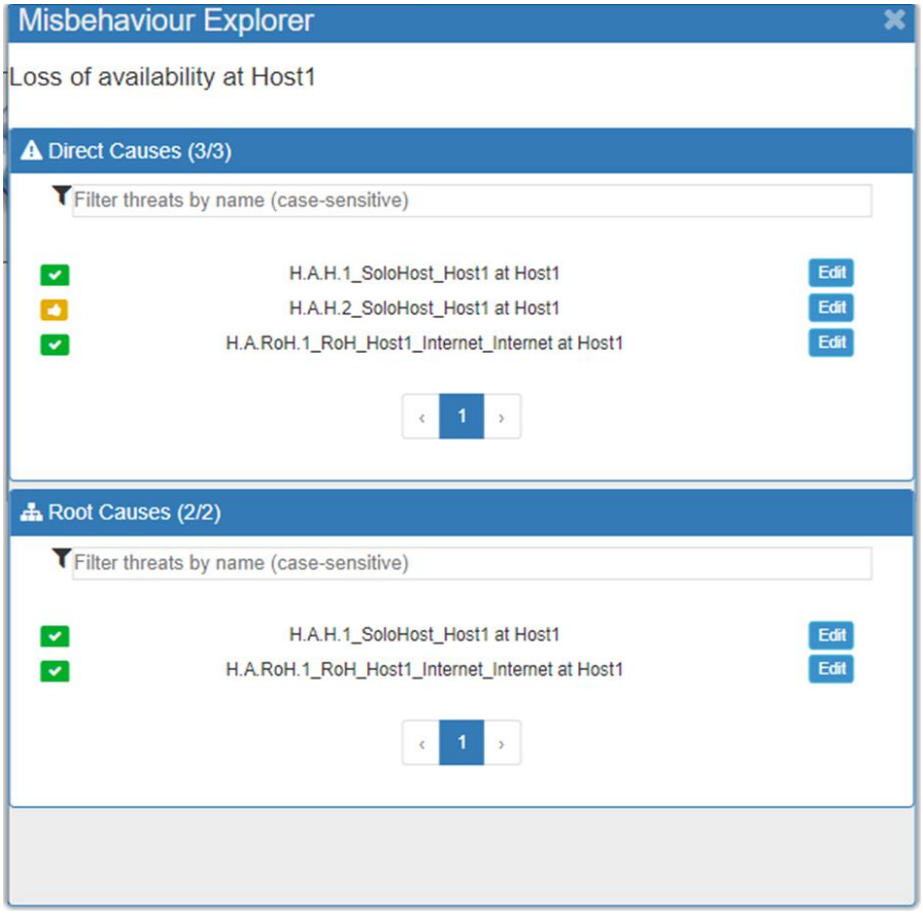


Figure 42 - The Misbehavior Explorer

3.1.8 Asset cardinality

After selecting an asset, the cardinality can be set in the top right corner of the GUI (see Figure 43).

Figure 43 - Setting asset cardinality

3.1.9 Layers

Various views of the network can be obtained by clicking on the Configure control (see cogwheel in the top right corner) and "*Select View Layer*" (see Figure 44).

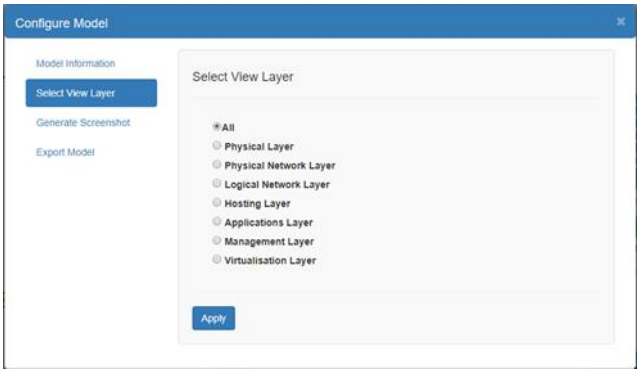


Figure 44 - Selecting a model view layer

3.1.10 Model read-only view

If required, a user can view an unchangeable version of a model by selecting the read button on a model (see Figure 7). This means the user will be unable to modify assets, relations or threats. The user will be able to browse the model in the same way as the model editor (see Figure 45).

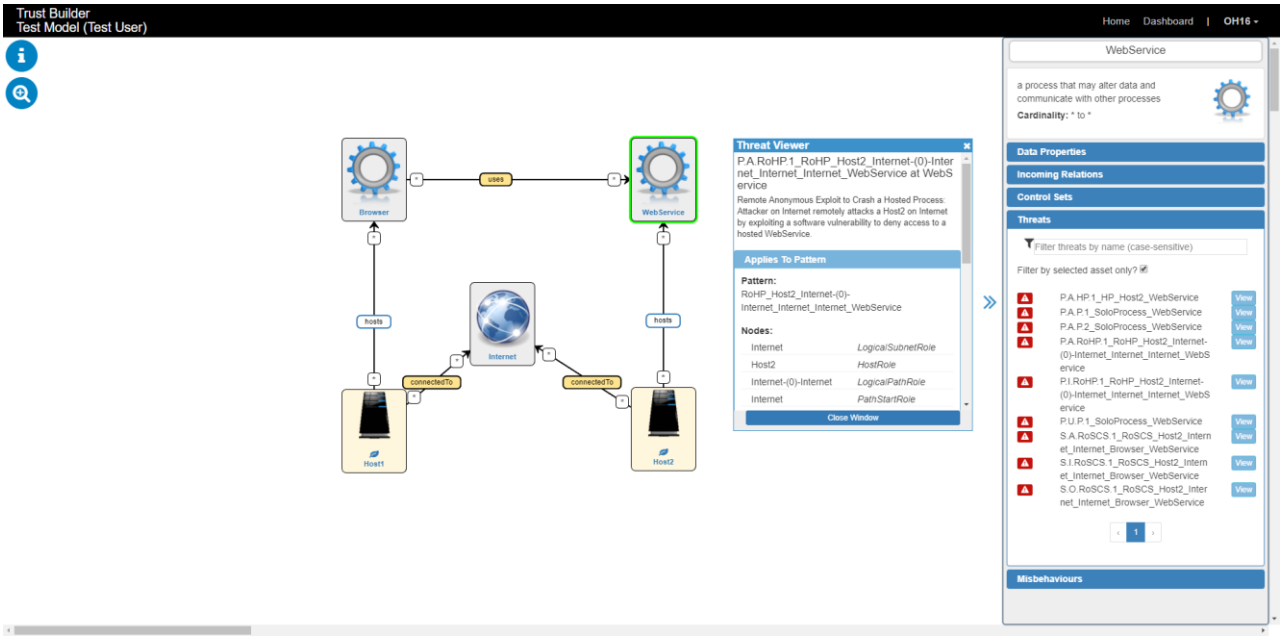


Figure 45 - Model read-only view

Assets can be selected using the asset browser in the right panel when no other asset is selected (see Figure 46). Similarly when selecting a relation which has inferred assets on it, the asset browser will offer a selection of these assets alone.

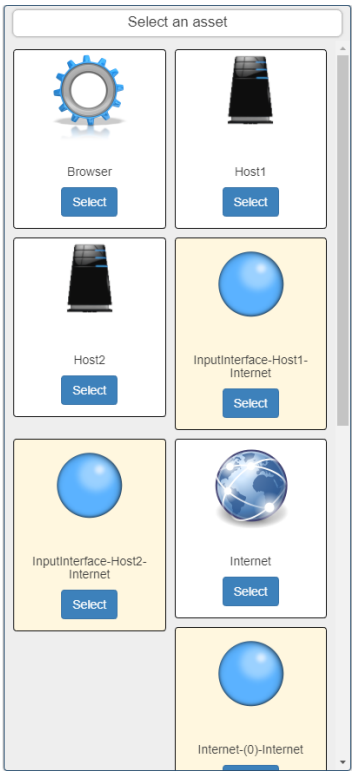


Figure 46 - Read-only asset browser

3.2 Programmer Guide

N.B. The presented *Trust Builder* is not programmable: there is no API that can drive the system programmatically.

4 Unit Tests

4.1 Information about Tests

The unit test presented in this document summarizes the sequence of steps required for model construction, validation and threat management.

4.2 Unit Test 1

The sequence of actions and the expected results are described in Table 3.

Step	Action	Expected Results	Execution
1	After starting the VM the Trust Builder can be accessed in a browser on this URL:localhost:8080/trust-builder	The main page presented to the user (see Figure 2).	manual

2	After clicking on "Login" we can log in the system by using these credentials: login: trustbuilder Password: 5fd4661f32ef9d2be4a3f794dff64cdd	Login page presented (see Figure 3).	manual
3	After successful login Click on "Create New Model".	The user can view the previously created models and create a new one (see Figure 7).	manual
4	Select the type of model from the dropdown menu.	See Figure 9.	manual
5	Drag items from the assets panel to the canvas and give meaningful names to the items.	The mode design canvas opens up (see Figure 16, Figure 18).	manual
7	Connect the assets by arrows.	By clicking on the asset, a green cross appears in the left corner, this indicates that the asset can be connected to other assets. The target assets are marked by a blue tick, showing that a connection can be made between the assets. By clicking on the blue tick icons we can establish connections between assets (see Figure 19, Figure 20, Figure 21, Figure 22).	manual
8	Validate model.	Once the model is constructed it can be validated. This operation is activated by clicking on the red "play" button (see Figure 23).	manual
9	Sort out issues with inferred assets (if there are any).	There is a "red/green" boundary for the "uses" connection. This indicates that the Incoming Relations need to be defined (see Figure 24).	manual
10	Define link between Browser and Service Pool.	See Figure 25.	manual
11	Define link between Web Service and Service Pool.	See Figure 26.	manual
12	Check that all Incoming Relations are defined.	See Figure 27	manual
13	Validate the model again.	See Figure 28	manual
14	List the threats associated with Host 1.	See Figure 29	manual
15	Selecting Control Set properties for Host1.	See Figure 31, Figure 32.	manual

Table 3 – Unit test sequence

5 Acknowledgements

The work described in this deliverable was sponsored by 5G-ENSURE Project (Grant Agreement number: 671562 — 5G-ENSURE — H2020-ICT-2014/H2020-ICT-2014-2).

6 Abbreviations

5G-PPP	5G Infrastructure Public Private Partnership
5G-ENSURE	5G Enablers for Network and System Security and Resilience

7 References

[1] 5.-E. Consortium, "Deliverable D3.5 - 5G PPP security enablers technical roadmap (Update)," 2016. [Online]. Available: http://www.5gensure.eu/sites/default/files/5G-ENSURE_D3.5 5G-PPP security enablers technical roadmap (Update).pdf, 2016.

[2] 5.-E. Consortium, "Deliverable D3.6 - 5G PPP Security Enablers Open Specifications (v2.0)," 2016. [Online]. Available: http://www.5gensure.eu/sites/default/files/5G-ENSURE_D3.6 5G-PPP security enablers open specifications (v2.0).pdf , 2016.