ΚΕΝΤΡΟ ΜΕΛΕΤΩΝ ΑΣΦΑΛΕΙΑΣ

CENTER FOR SECURITY STUDIES

CICSYN2013

5th International Conference

Computational Intelligence, Communications Systems and Networks

Madrid Spain, 5-7 June 2013.

A BLEND OF SEMANTIC MONITORING AND INTRUSION DETECTION SYSTEMS FOR THE PROTECTION OF CRITICAL INFRASTRUCTURES: RESEARCH EFFORTS WITHIN THE GREEK CYBERCRIME CENTER *

Team

Presenter: Vasilis Tsoulkas, PhD,

Co-Authors:

- Dimitris Kostopoulos, MSc, Software Engineer/Analyst
- George Leventakis, PhD, Security Risk Analyst
- Prokopis Drogkaris, PhD, Security Policy Analyst
- Vicky Politopoulou, MSc, Analyst-Law Enforcement Agent

*
The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement No. HOME/2011/ISEC/AG/INT/4000002166.

Presentation

- Motivation and Objectives
- Critical Infrastructure Description
- Semantic System Modeling Aspects (Brief Overview)
- Monitoring and Stream Reasoning Process
- Decision Support Tool Interface & Risk Analytics
- Research Directions and Conclusions within the Greek Cybercrime Center

Motivation and Objectives

- **Critical Infrastructures are characterized by**: Increased Connectivity of their Information and Data Processing Networks
- **Information sharing** provides better Resource Optimization and Effectiveness.
- **Also substantial Cost Reduction** for Management and Systems Maintenance
- **Unfortunately Increased Connectivity** and Data Sharing introduces new challenges on Cyber – Risks and Vulnerabilities.

Motivation and Objectives

Some Critical Infrastructures vulnerabilities

1. **Cyber-Attacks** against *interconnected* Information & Communication channels disrupt Exchanged Data flows and Integrity

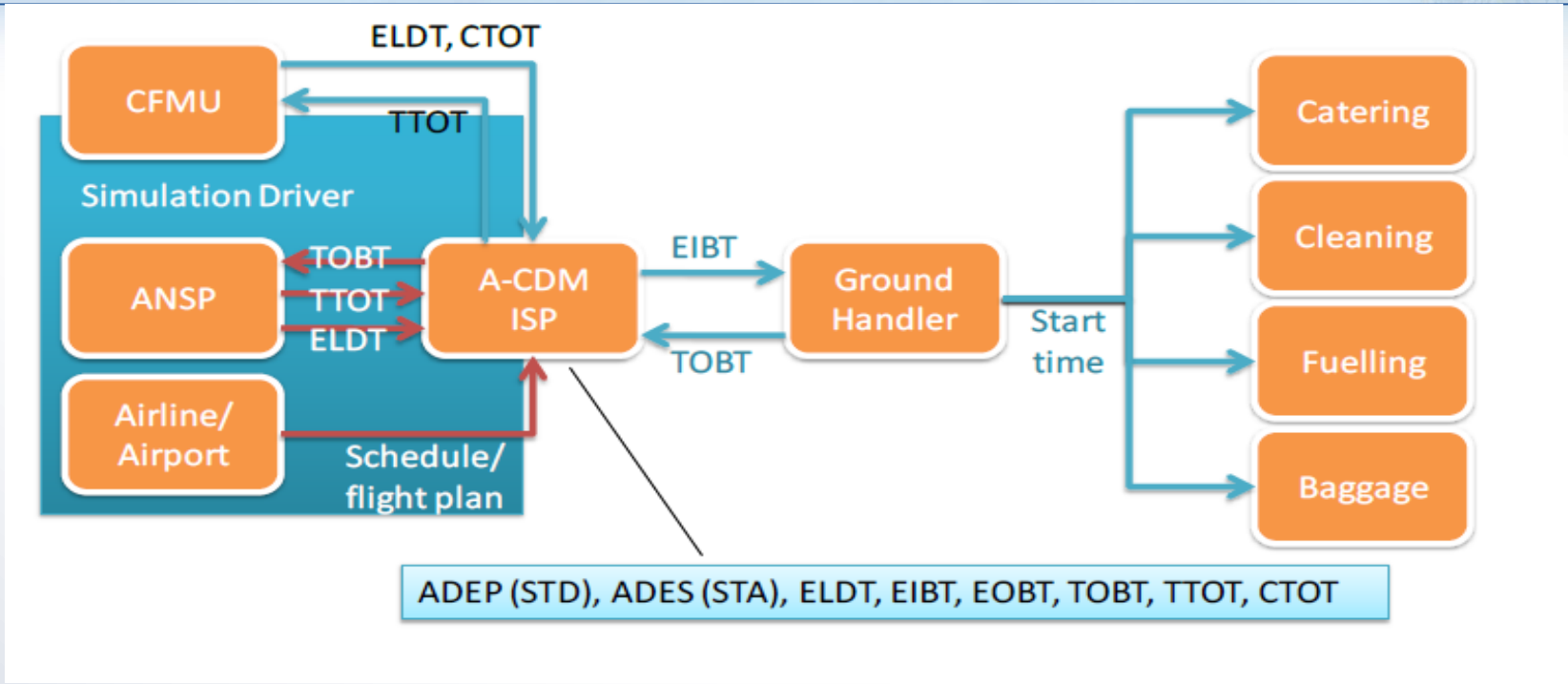
2. **Local Disruptions** in one System can be distributed to other Systems due to coupling and inter-dependencies

3. **Reduced Resilience** against cyber-disruptions due to reduced *excess capacity* arising from the exchanged data.

Research Objectives

- Implementation of Agile Service Oriented Technologies for Multi-Stake Holder Systems in order to handle:
  - **Dynamic composition of** ICT connections of the Critical Infrastructure at Run-Time and NOT at Design Time.
  - **Dynamic monitoring of** ICT components against well-defined Assets dependability criteria
  - **Development and Integration of** : Stream Reasoning and Intrusion Detection schemes for Real Time Operator Assistance

CI: An Airport-Collaborative Decision Making – European Air Traffic Management System (Configuration with Emulated Services)



Services are accessible by a consumer (**aircraft operator**) through **SLA templates**.

The Ground Handler is responsible for coordination of Ramp Services (catering, fuelling, cleaning, baggage handling)

The Ground Handler: Is an Orchestrator of Ramp Services to have an aircraft ready for its next flight

Some Air-Traffic Critical Parameters

EET	Estimated Elapsed Time	The estimated time required to proceed from one significant point to another (ICAO)
EEZT	Estimated End of De-icing Time	The estimated time when de-icing operations on an aircraft are expected to end
EIBT	Estimated In-Block Time	The estimated time that an aircraft will arrive in blocks. (Equivalent to Airline/Handler ETA – Estimated Time of Arrival).
ELDT	Estimated Landing Time	The estimated time that an aircraft will touchdown on the runway. (Equivalent to ATC ETA –Estimated Time of Arrival = landing).
EOBT	Estimated Off-Block Time	The estimated time at which the aircraft will commence movement associated with departure (ICAO).
ERZT	Estimated Ready for De-icing Time	The estimated time when the aircraft is expected to be ready for de-icing operations
ETFMS	Enhanced Tactical Flow Management System	
ETO	Estimated Time Over	
ETOT	Estimated Take Off Time	The estimated take off time taking into account the EOBT plus EXOT. (Equivalent to ATC ETD– Estimated Time of Departure).
ETTT	Estimated Turn-round Time	The time estimated by the AO/GH on the day of operation to turn-round a flight taking into account the operational constraints
EXIT	Estimated Taxi-In Time	The estimated time between landing and in-block
EXOT	Estimated Taxi-Out Time	The estimated time between off-block and take off

Data quality and Key Performance Indicators (KPIs)

- **Data**: Confidentiality, Integrity, Alarms, Data Display
- **KPI**s: Reflect the Quality of Service Delivery
- **KPIs properties**: Is the Quality of Time Estimates

  Accuracy
  Predictability
  Stability
- An SLA Architecture was developed with the following KPIs & Parameters in the **A**irport **C**ollaborative **D**ecision **M**aking (A-CDM) context:
  - **System Availability**
  - **Data Quality**
  - **Data timeliness, delivery deadlines**
  - **Confidentiality**

The Critical Infrastructure Modeling Challenge for Automated Machine Reasoning

The Dynamic Multi-Stakeholder system consists of **4-levels of abstraction**

1. **Core ontology structure:** to model the System and its assets subject to threats and protected by Counter-measures (**controls**).
2. **Dependability model:** describing system independent: assets, threats, controls. **Only OWL classes** and relationships are used. Security expertise is encoded in the Critical Infrastructure (**CI**) model.
3. **Abstract system model**: describes system-specific threats and counter-actions. Extends dependability model classes with imported security knowledge.
4. A **concrete system model**: provides snapshots of the running system and instances of the participating assets + contextualised threats & controls.

   It is generated by populating the abstract system model classes with instances, based on run-time monitoring data from the system

   Each level inherits properties from its predecessor. The **final concrete model** has simple structure and integrates knowledge from: Abstract system model and Dependability model.

Brief Analysis of Ontology & Models

1. **The Semantic Ontology** is constructed such that:
   - Only OWL Classes are used for design-time modelling
   - OWL Instances are used for modelling the Run – Time System Composition
   - Security expertise is added at design time in the OWL classes

2. **The Dependability model** provides the first step to develop the Abstract System Model which is a *Design – Time Model* of the system that will be composed dynamically "On the Fly"

3. **The Concrete Model Generator** is connected to the monitoring subsystem to create a model of the Running System.

   The Concrete Model is Automatically Generated from System Monitoring Data for Machine Reasoning.

Main Innovation of the Approach

- The Modelling approach is constructed using Semantics Modelling for Machine Reasoning automated threat analysis and risk estimation when the system is composed at "Run-Time".

- The design – time Service Oriented Dynamic models are abstract: They describe the structure but NOT the composition of the system which is NOT KNOWN until "Run-Time".

Some Model Explanations

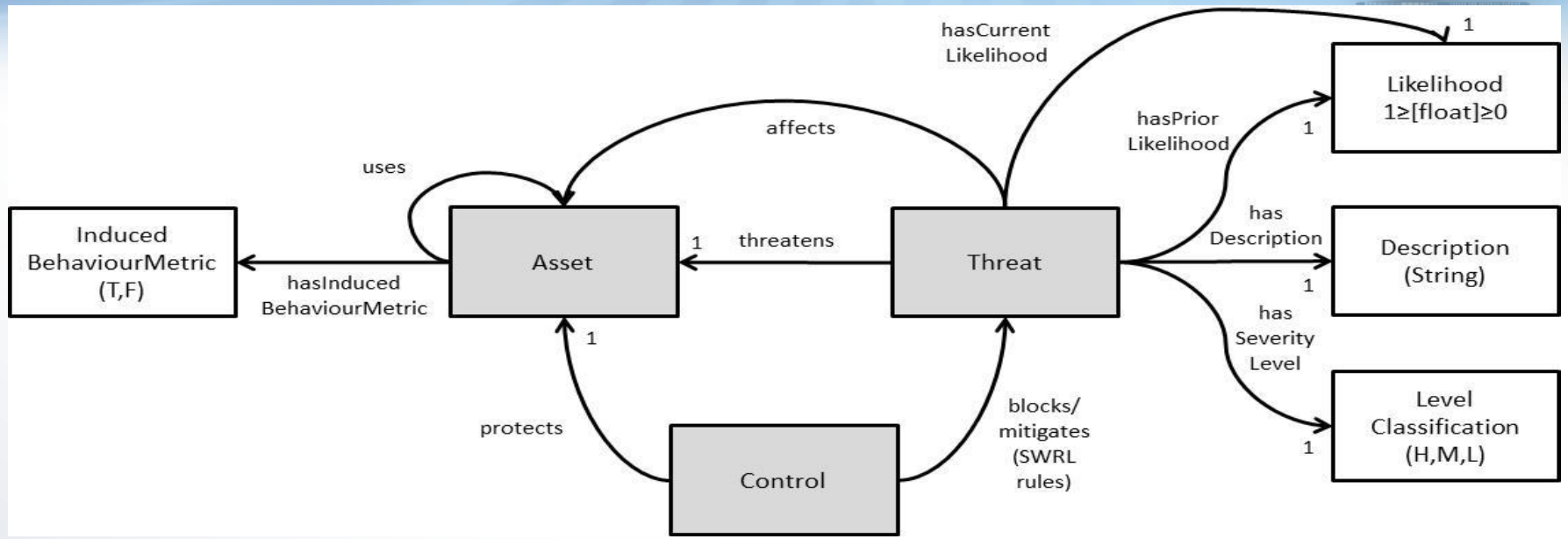
- Assets represent the entities that provide the functional system interactions.
- They are classified into three types: services, clients, resources.

Services: Are system components that provide services.

Clients: Are system components that access these services.

- It is possible for an asset to be both a service and a client.
- **Threat types** are defined only for services.

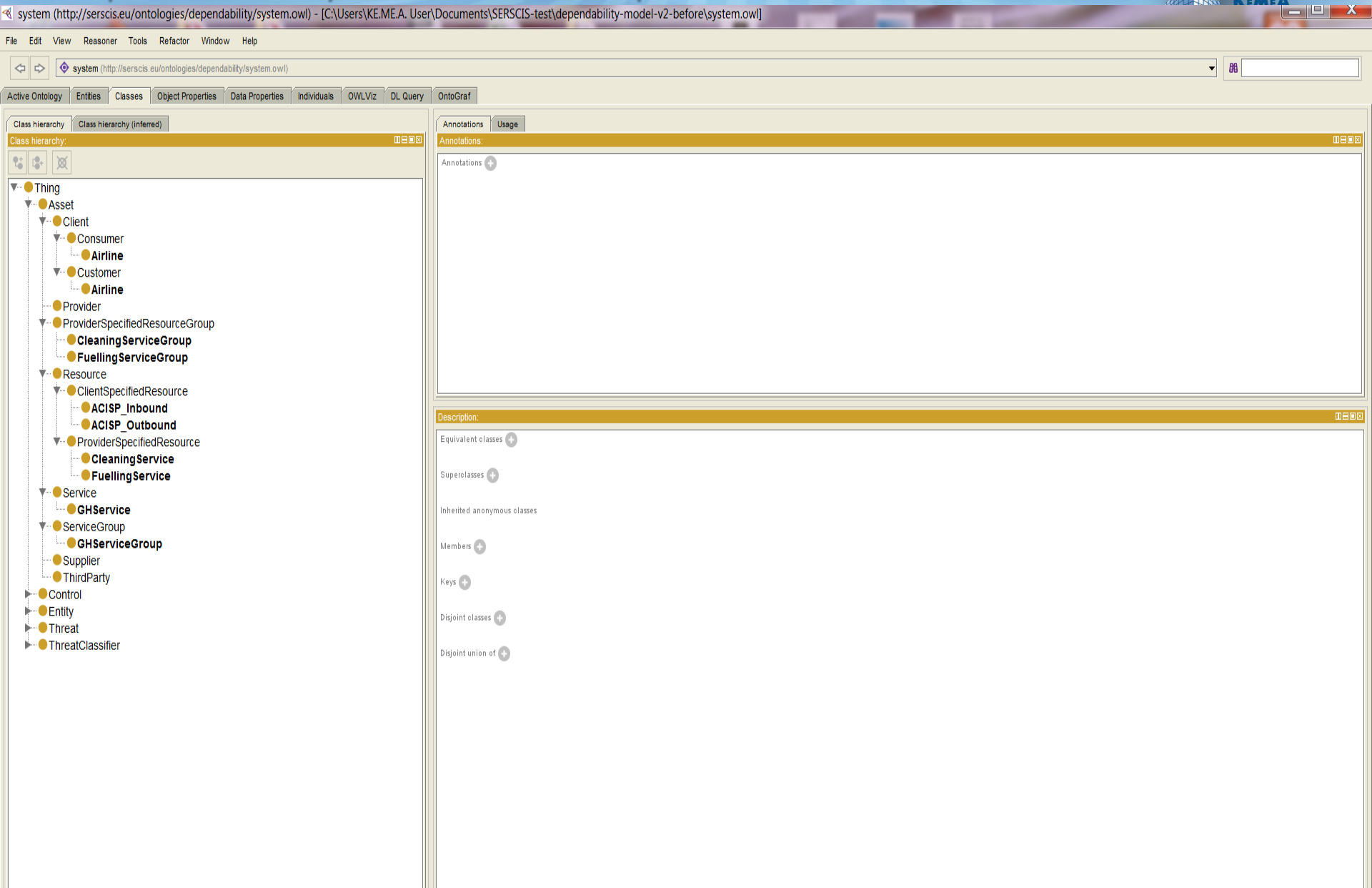
Core System Domain Ontology



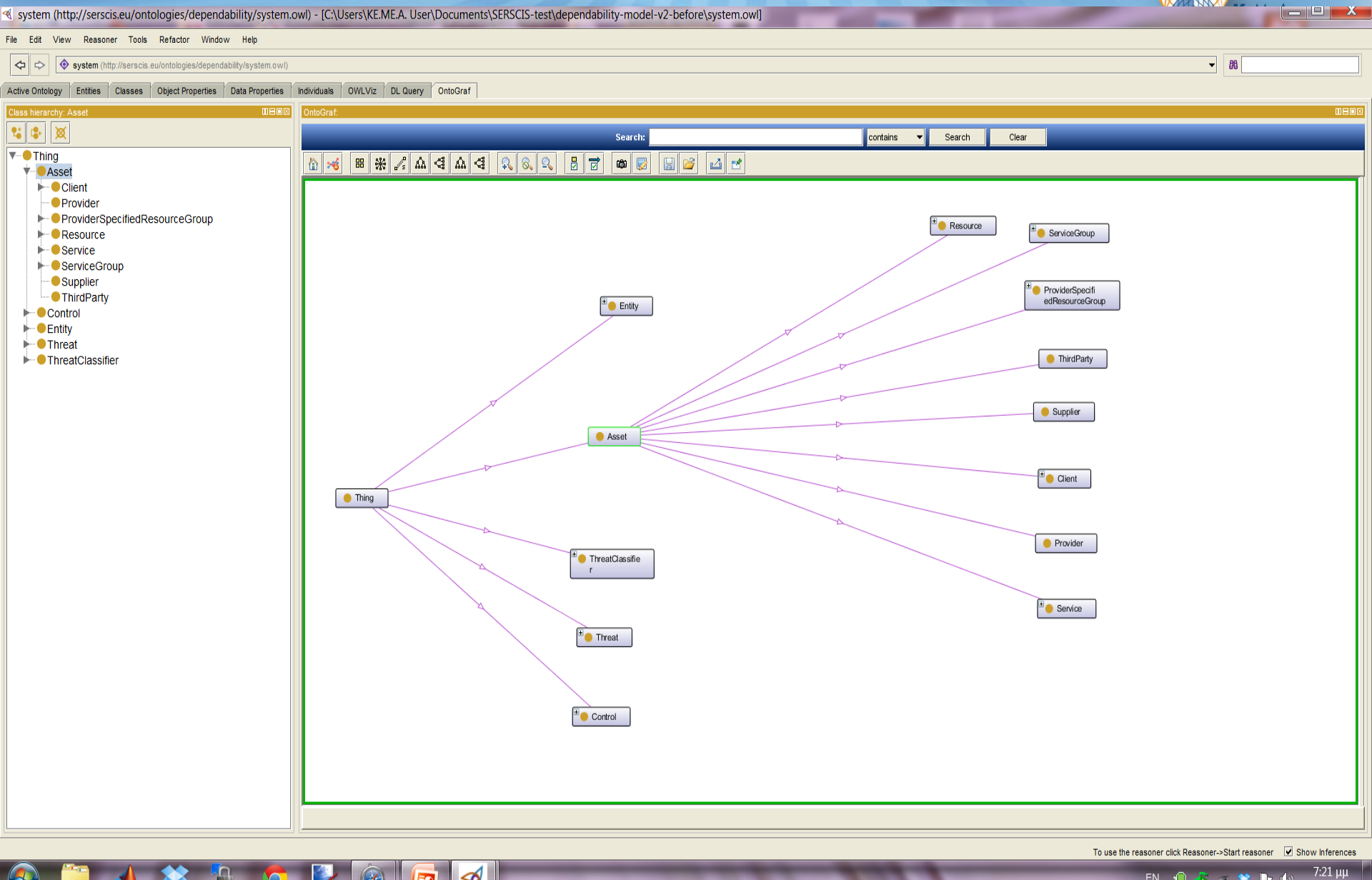
- This basic system structure, determines what reasoning is used

Threat class	Description	Controls needed
Unauthorized access	The service processes an unauthorised request from an attacker.	Client AuthN + Client AuthZ
Unaccountable access	Type of unauthorized access, designed to get the service without paying for it.	Client AuthN + Client AuthZ

Critical Infrastructure System Domain OWL Classes Dependability Model – Sample Screen



Assets Dependability Graph Visualization



SWRL (Logical axioms) encoded in the Dependability Model

dependability Protégé 3.4.6

File Edit Project OWL Reasoning Code Tools Window BioPortal Collaboration Help

protégé

Metadata(dependability.owl) | OWLClasses | Properties | Individuals | Forms | SWRL Rules

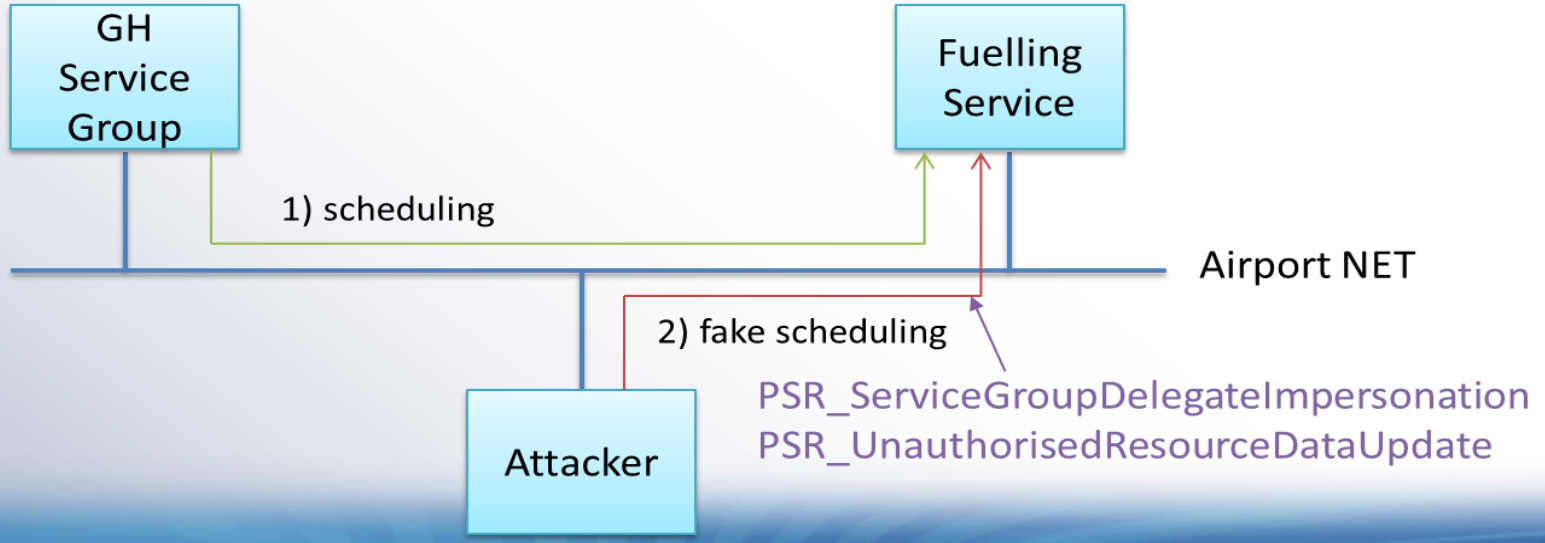
SWRL Rules

Enabled	Name	Expression
☑	@A1002_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ PersistentResourceInaccuracy(?t) ∧ ProviderSpecifiedResourceGroup(?a) ∧ ResourceBlacklisting(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A1015_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Identification(?c1) ∧ ResourceTrafficCorruption(?t) ∧ Service(?a) ∧ ServiceAuthN(?c2) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1032_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Customer(?a) ∧ Delegation(?c) ∧ MissAccountedClientSpecifiedResourceAccess(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1045_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientAuthN(?c1) ∧ ClientSpecifiedResource(?a) ∧ ClientSpecifiedResourceTrafficCorruption(?t) ∧ Identification(?c2) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1062_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ResourceImpersonation(?t) ∧ Service(?a) ∧ ServiceAuthN(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1075_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Customer(?a) ∧ CustomerStaffVetting(?c) ∧ UntrustworthyCustomerStaff(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1088_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Customer(?a) ∧ Delegation(?c) ∧ ExcessiveCustomerRequests(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1101_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c1) ∧ ClientAuthN(?c2) ∧ ClientSpecifiedResource(?a) ∧ MissAccountedClientSpecifiedResourceAccess(?t) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1118_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ResourceCapacityManagement(?c) ∧ ServiceGroup(?a) ∧ TooFewResources(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A1131_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ConsumerTrafficSnooping(?t) ∧ Customer(?a) ∧ Encryption(?c1) ∧ ServiceAuthN(?c2) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1148_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c1) ∧ AuthorisedClientSpecifiedResourceUserImpersonation(?t) ∧ ClientAuthN(?c2) ∧ ClientSpecifiedResource(?a) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1165_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ProviderSpecifiedResourceGroup(?a) ∧ ResourceCapacityManagement(?c) ∧ TooFewResources(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A1178_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Customer(?a) ∧ Delegation(?c) ∧ MissAccountedServiceAccess(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1191_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ CustomerBlacklisting(?c) ∧ ExcessiveCustomerRequests(?t) ∧ ServiceGroup(?a) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A1204_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientSpecifiedResourceTrafficCorruption(?t) ∧ Service(?a) ∧ ServiceAuthN(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1217_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c1) ∧ ClientAuthN(?c2) ∧ MissAccountedResourceAccess(?t) ∧ ProviderSpecifiedResource(?a) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1234_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientAuthN(?c) ∧ ProviderSpecifiedResource(?a) ∧ ServiceDelegateImpersonation(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1247_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientAuthN(?c) ∧ CustomerImpersonation(?t) ∧ Service(?a) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A1260_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AuthorisedConsumerImpersonation(?t) ∧ Consumer(?a) ∧ Identification(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A354_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Client(?a) ∧ ServiceAuthN(?c) ∧ ServiceImpersonation(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A367_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Customer(?a) ∧ Delegation(?c) ∧ OversizedCustomerRequests(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A380_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Service(?a) ∧ ServiceSoftwareMalfunction(?t) ∧ ServiceSoftwarePatching(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A393_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Service(?a) ∧ StaffVetting(?c) ∧ UntrustworthyProviderStaff(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A406_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Delegation(?c1) ∧ Identification(?c2) ∧ MissAccountedResourceAccess(?t) ∧ Service(?a) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A423_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ CustomerBlacklisting(?c) ∧ OversizedCustomerRequests(?t) ∧ ServiceGroup(?a) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A436_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c2) ∧ ClientAuthN(?c1) ∧ ProviderSpecifiedResource(?a) ∧ UnauthorisedResourceDataUpdate(?t) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A453_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientAuthN(?c2) ∧ Encryption(?c1) ∧ Identification(?c3) ∧ ProviderSpecifiedResource(?a) ∧ ResourceTrafficSnooping(?t)∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:protects(?c3, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A474_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c1) ∧ ClientAuthN(?c2) ∧ OversizedCustomerRequests(?t) ∧ SLAEnforcement(?c3) ∧ Service(?a) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:protects(?c3, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A495_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ PersistentResourceUnderPerformance(?t) ∧ ProviderSpecifiedResourceGroup(?a) ∧ ResourceBlacklisting(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A508_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Consumer(?a) ∧ ServiceAuthN(?c) ∧ ServiceTrafficCorruption(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A521_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientAuthN(?c2) ∧ Identification(?c1) ∧ Service(?a) ∧ ServiceTrafficCorruption(?t) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A538_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AuthorisedClientSpecifiedResourceUserImpersonation(?t) ∧ Customer(?a) ∧ Delegation(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A551_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c2) ∧ ClientAuthN(?c1) ∧ ClientSpecifiedResource(?a) ∧ UnauthorisedClientSpecifiedResourceDataAccess(?t)∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A568_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientAuthN(?c1) ∧ Identification(?c2) ∧ ProviderSpecifiedResource(?a) ∧ ResourceTrafficCorruption(?t) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A585_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Customer(?a) ∧ CustomerImpersonation(?t) ∧ Identification(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A598_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Delegation(?c) ∧ Service(?a) ∧ ServiceDelegateImpersonation(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A611_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ CriticalResourceDependency(?t) ∧ ProviderSpecifiedResourceGroup(?a) ∧ ResourceRedundancy(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A624_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientSpecifiedResourceImpersonation(?t) ∧ Service(?a) ∧ ServiceAuthN(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A637_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ ClientSpecifiedResourceSoftwareMalfunction(?t) ∧ Customer(?a) ∧ CustomerSoftwarePatching(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A650_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AuthorisedConsumerImpersonation(?t) ∧ Customer(?a) ∧ Delegation(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A663_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ DynamicFailover(?c) ∧ OneOffResourceMalfunction(?t) ∧ Service(?a) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A676_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ PersistentResourceMalfunction(?t) ∧ ProviderSpecifiedResourceGroup(?a) ∧ ResourceBlacklisting(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → MitigatedThreats(?t)
☑	@A689_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Resource(?a) ∧ ResourceSoftwareMalfunction(?t) ∧ SupplierSoftwarePatching(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A702_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AuthorisedResourceUserImpersonation(?t) ∧ ClientAuthN(?c) ∧ ProviderSpecifiedResource(?a) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A715_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Customer(?a) ∧ Delegation(?c) ∧ UnauthorisedClientSpecifiedResourceDataUpdate(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A728_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Delegation(?c2) ∧ Identification(?c1) ∧ Service(?a) ∧ UnauthorisedResourceDataUpdate(?t) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A745_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AuthorisedResourceUserImpersonation(?t) ∧ Identification(?c) ∧ Service(?a) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A758_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c1) ∧ ClientAuthN(?c2) ∧ MissAccountedServiceAccess(?t) ∧ Service(?a) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A775_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ AccessControl(?c1) ∧ ClientAuthN(?c2) ∧ ExcessiveCustomerRequests(?t) ∧ SLAEnforcement(?c2) ∧ Service(?a) ∧ core:protects(?c1, ?a) ∧ core:protects(?c2, ?a) ∧ core:protects(?c3, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A796_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Client(?a) ∧ ClientSoftwareMalfunction(?t) ∧ ClientSoftwarePatching(?c) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A809_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ SLACommitmentManagement(?c) ∧ ServiceGroup(?a) ∧ TooManyCustomers(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)
☑	@A822_8d7a28f6_c614_4272_87d9_cd3db6ceaf71	→ Identification(?c) ∧ ProviderSpecifiedResource(?a) ∧ ResourceImpersonation(?t) ∧ core:protects(?c, ?a) ∧ core:threatens(?t, ?a) → BlockedThreats(?t)

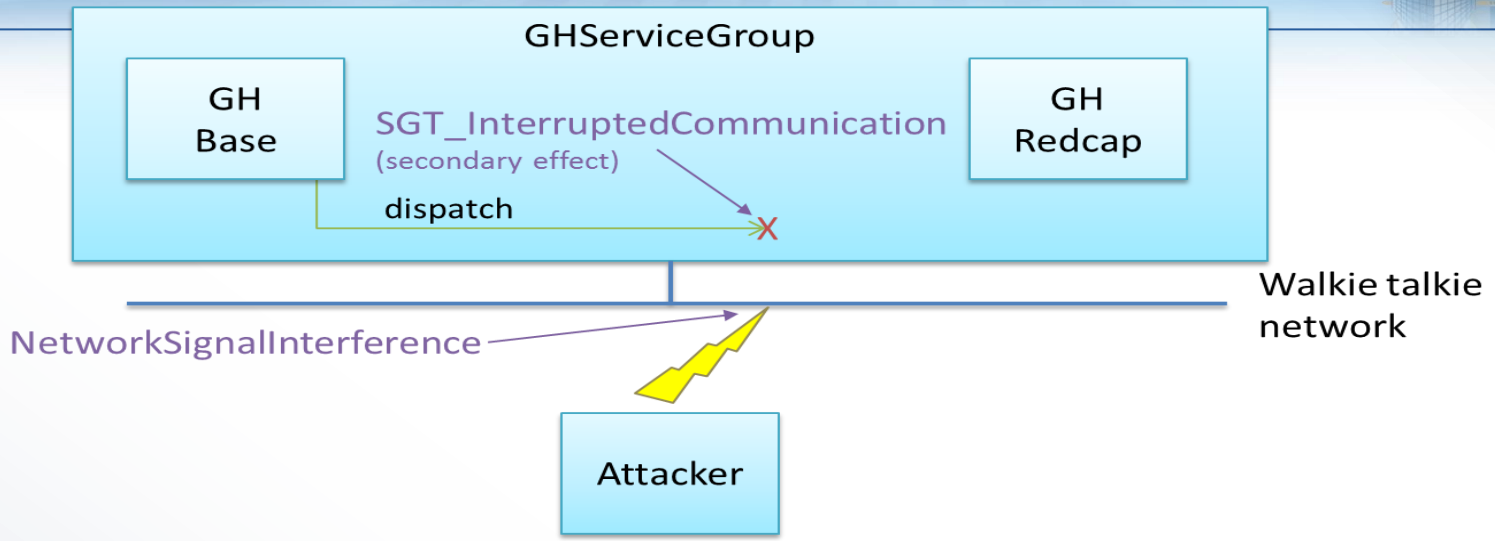
Threat Types & Threat Proof of Concept Scenario (Scenario 1: Remote Exploit on Fuelling Service)

1. Unauthorized Access (to the service)
2. Data traffic Snooping
3. Man in the Middle
4. Client Impersonation
5. Resource Failure

Unauthorized Data Update at Fuelling Service



Proof of Concept Scenario (Scenario 2: Jamming the Ground Handler's Walkie Talkie Network).



Scenario: The GH communicates with a mobile **"redcap"** via push – to – talk (**PTT**) radio units. The base station sends dispatch notifications to the redcap with details of stand and inbound flight to turnaround

An Attacker jams the communication links be emitting radio interference signals blocking 2-way message transmission.

The GH cannot deliver dispatch details to the red cap using the network. **The Turnaround** workflow **cannot be completed** for the airline customer.

2nd Scenario Logical Modeling Explanation

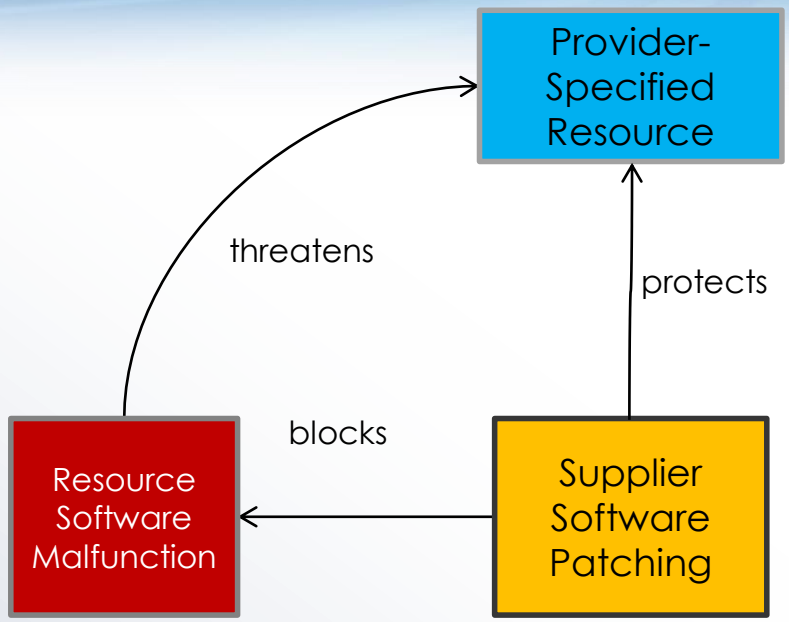
- The GH base and Redcap are logical entities encapsulated within the **GHServiceGroup** class.
- **Induced Behaviour:** When Jamming Attack is in Progress we observe from the GH Resource Manager that the metric: **http://serscis.eu/ontologies/airport/comms.owl#timeouts_WalkieTalke** is being incremented.
- It measures the number of timeouts when trying to transmit on the Walkie-Talkie networking due to jamming interference
- The **B**ehaviour **A**nalyser (**BA**) Infers from these time – outs that the Walkie-Talkie network Asset is Unavailable because of Jamming.

Counter – Actions (Control) Class Explanation

Control (counter measure) classes provide:

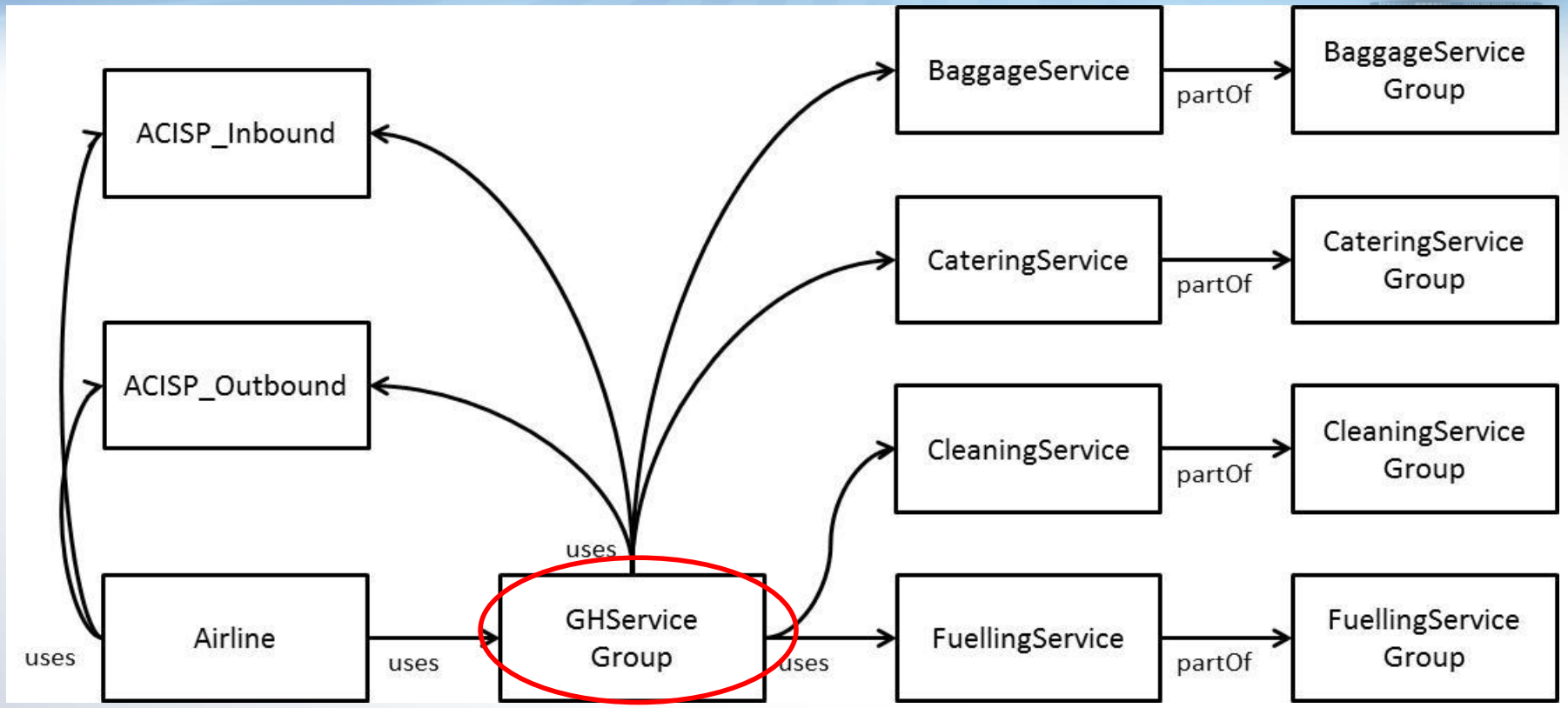
- **generic control types** that can be included directly in an abstract system model;
- **descriptions of deployment actions**: how to deploy the control into the real system;
- **descriptions of mitigation actions**: how to operate reactive controls to protect assets when a threat is carried out against them.

Resource Software Malfunction (Mild Error)



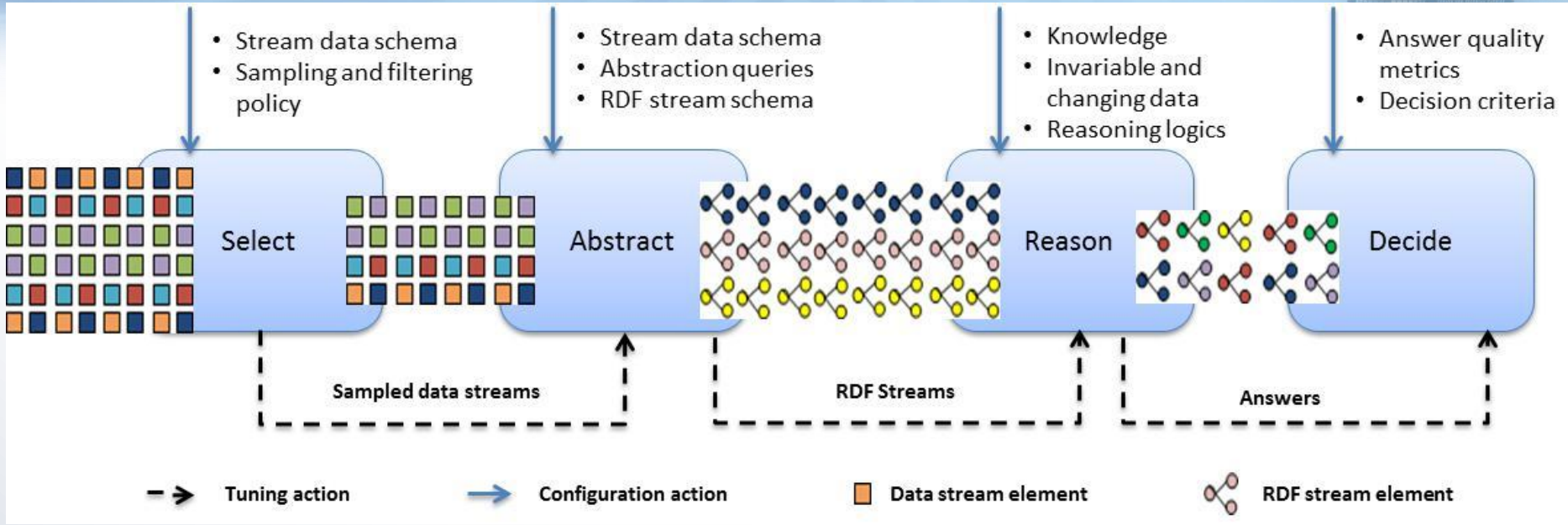
- **Threat**
  - a bug in PSResource software causes it to repeatedly produce faults
- **Controls**
  - PSResource has Suplier Software **Patching**: the Supplier has a procedure to maintain the software used by the PSResource
    - **ensures bug fixes are applied promptly**
- **System specifics**
  - **one** subclass per PSResource class
  - **one** instance per PSResource of each of the resulting classes

Abstract System Model of multi-stakeholder CI



- It is a design-time model of the structure of the dynamic, multi-stakeholder Service-Oriented system: Input for fully automated run-time model generation and analysis Tools. It is composed dynamically at Run-Time.

Monitoring and Stream Reasoning



- Information arrives as a stream of "time-stamped" graph data
- The Knowledge base is continuously updated and reasoning goals are continuously re-evaluated as new assertions arrive
- Reasoning is implemented from a Finite – Time Window and not at a Single Instant !!.
- Research Efforts on Stream Reasoning is still at its First Steps and its Infancy.

4 basic – steps in Stream Reasoning

1. **Select**: Relevant Data from Input Streams by using Sampling Policies that probabilistically drop stream elements to address bursty streams of data that may have unpredictable peaks.

2. **Abstract**: Sampled streams are input to the Abstract block to generate aggregate events by enforcing aggregate events continuously.

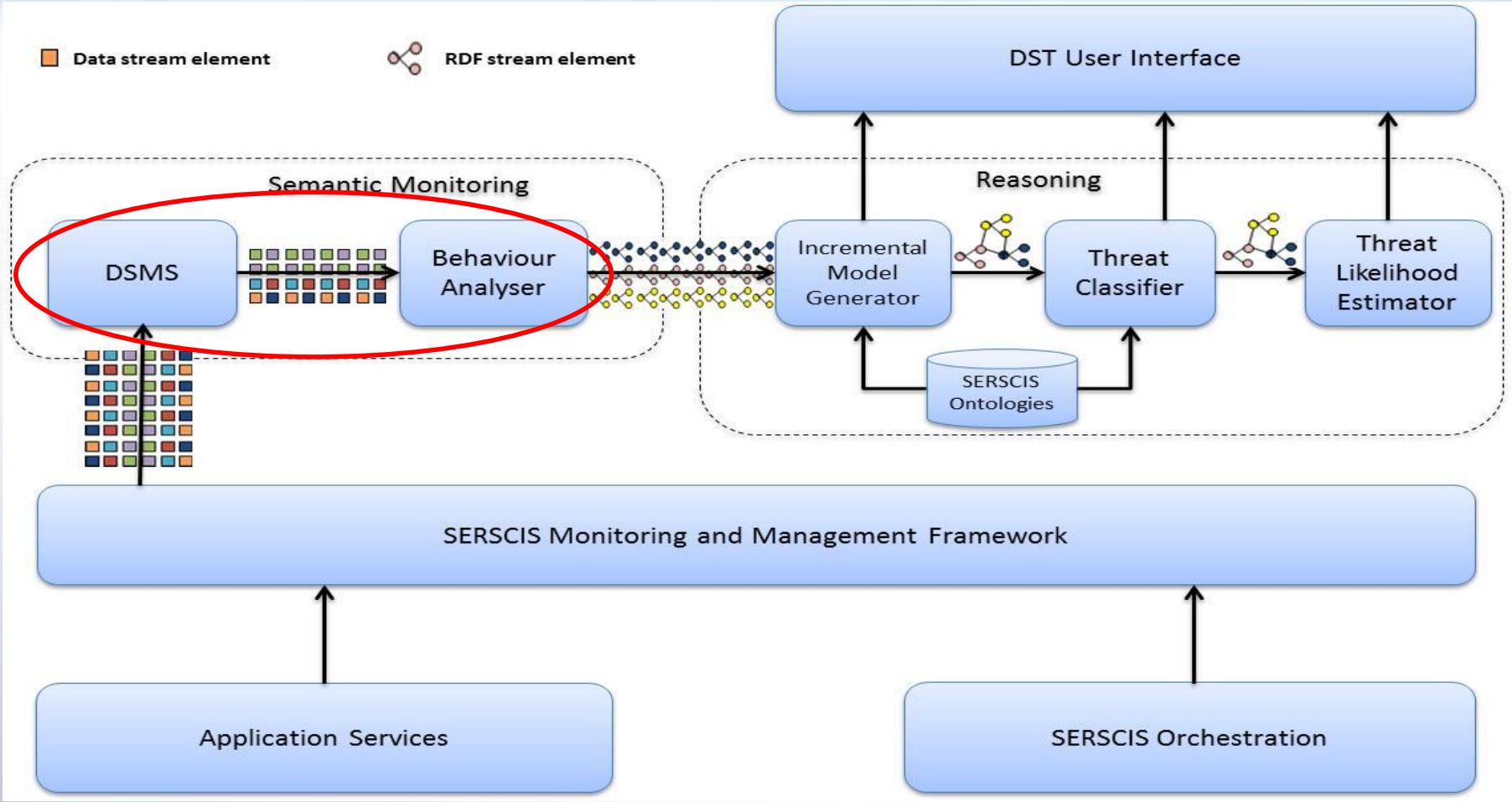
Output is RDF streams (ρ, τ) with **ρ – RDF triple and τ – time stamp** (logical arrival time of RDF statement. *Use of* ***C-SPARQL***.

Steps in Stream Reasoning

3. **Reason**: RDF (*Graph Streams*) streams are injected into background knowledge to perform reasoning tasks. Incremental implementation of RDF snapshots.

4. **Decide:** Before final answers the final answering process reaches a decision step where different experts' pre-defined metrics and criteria are used to evaluate the quality of the answer and adapt possible behaviours.

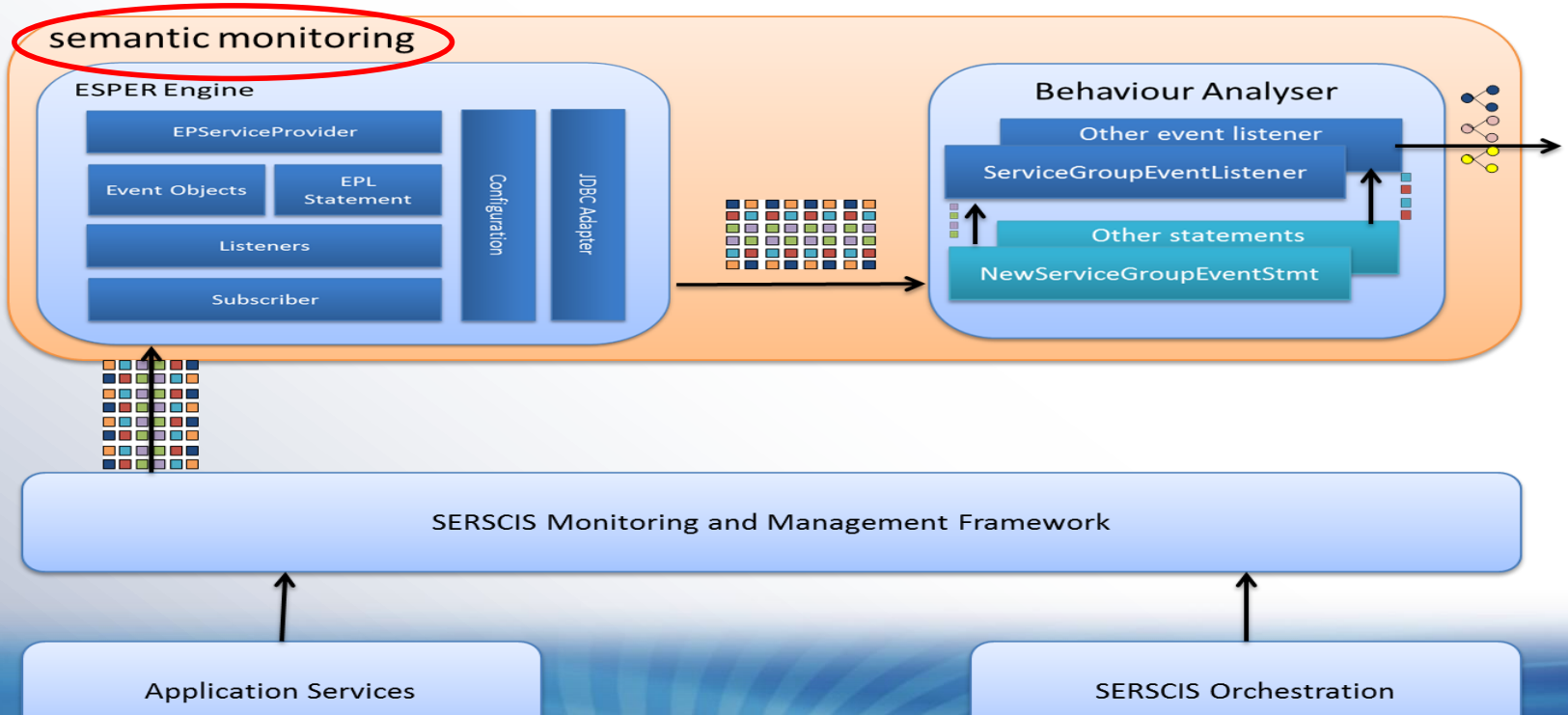
Semantic Monitoring Block



Semantic Monitoring Component : DSMS - Behavior Analyser - Sequential Detection

DSMS: Data Stream Management System : samples & filters monitoring data generated by Service Monitoring and Management Components.

- Usage of open-source CEP (Java - ESPER): Real Time engine that triggers Listeners or Subscribers using a tailored Event Processing Language (EPL).



Behavior Analyser (BA)

- Processing of multiple data streams from DSMS. Produced Output is Graph Triples (RDF).

- *Decides how to convert raw monitoring data into Semantic* Assertions related to: Presence of Assets and Behaviors.

- The monitoring framework generates 2 – types of Time stamped RDF assertions:

  (1) Presence or Absence of Assets (joining or leaving the system)

  (2) Assertions about Measurability, Presence or Absence of Adverse Behavior of these Assets.

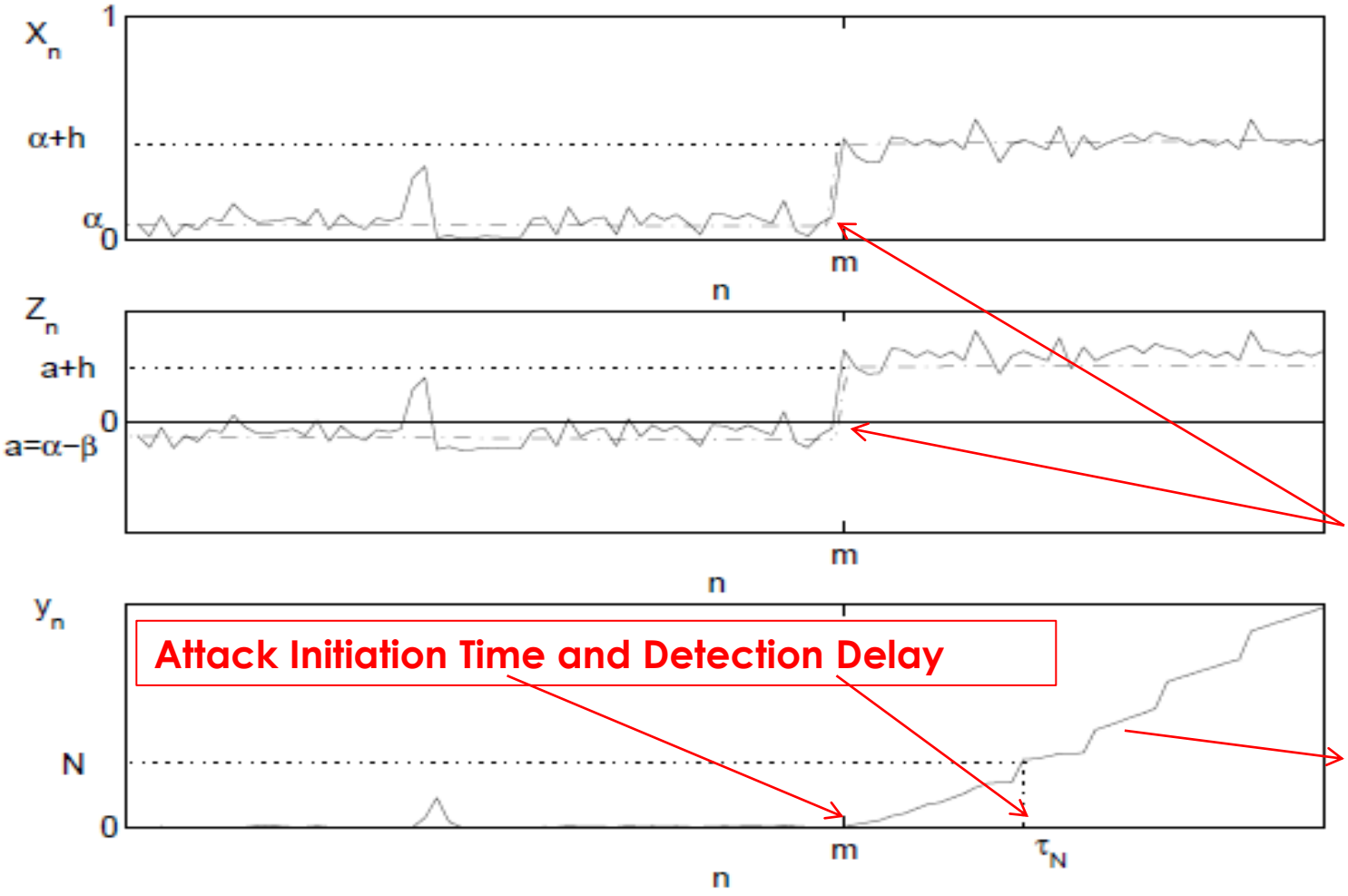
Behavior Analyser (BA)

- The **BA** is not only a Transcoder converting Monitoring Events to RDF graphs.
- The **BA** decides about the type of Behaviors (Assets and Services).
- Example: The **BA** is capable to determine if an Asset is Overloaded or Underperforming using Monitoring Data for Load and Performance (KPIs – SLA events).

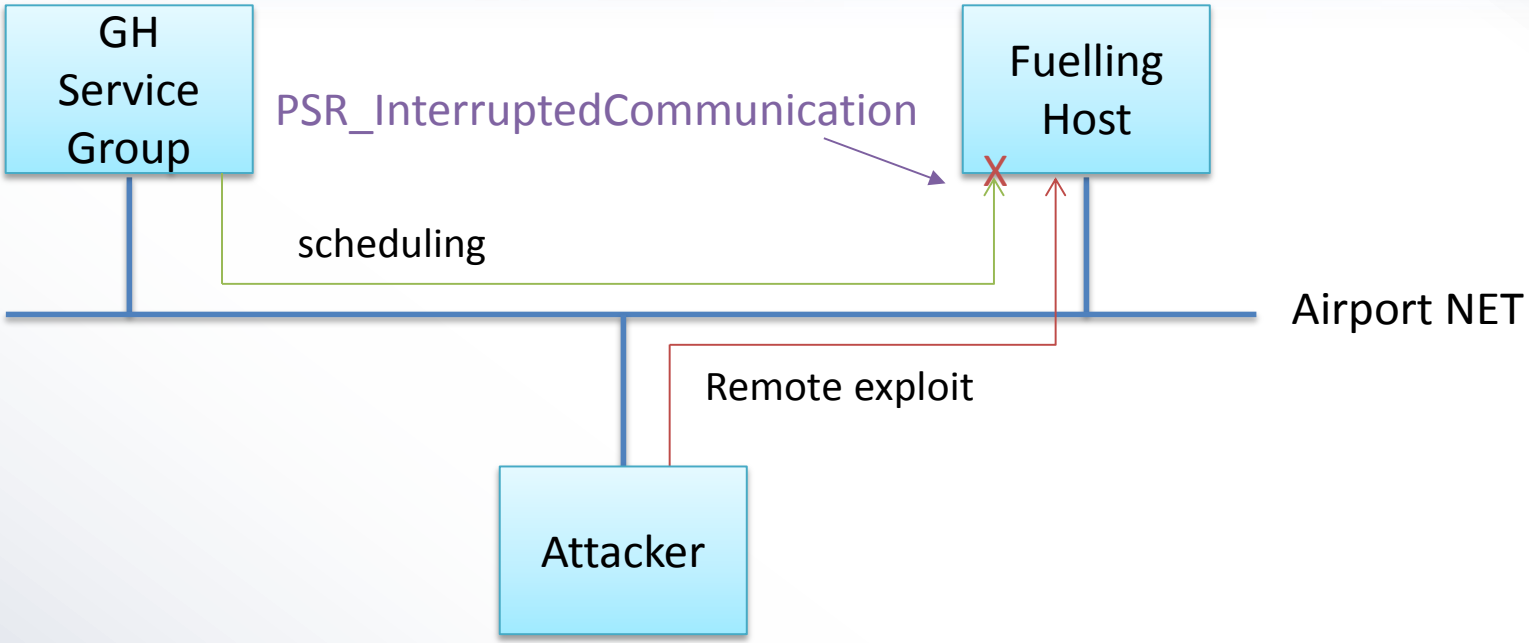
Sequential Inspection

- Cumulative Sum (CUSUM) algorithm from the sequential statistics literature.
- In general parametric models are used
- Inspection of Change in the mean of the relevant stochastic process
- We use: The non-parametric version of CUSUM

Sequential Intrusion (Behavior) Detection



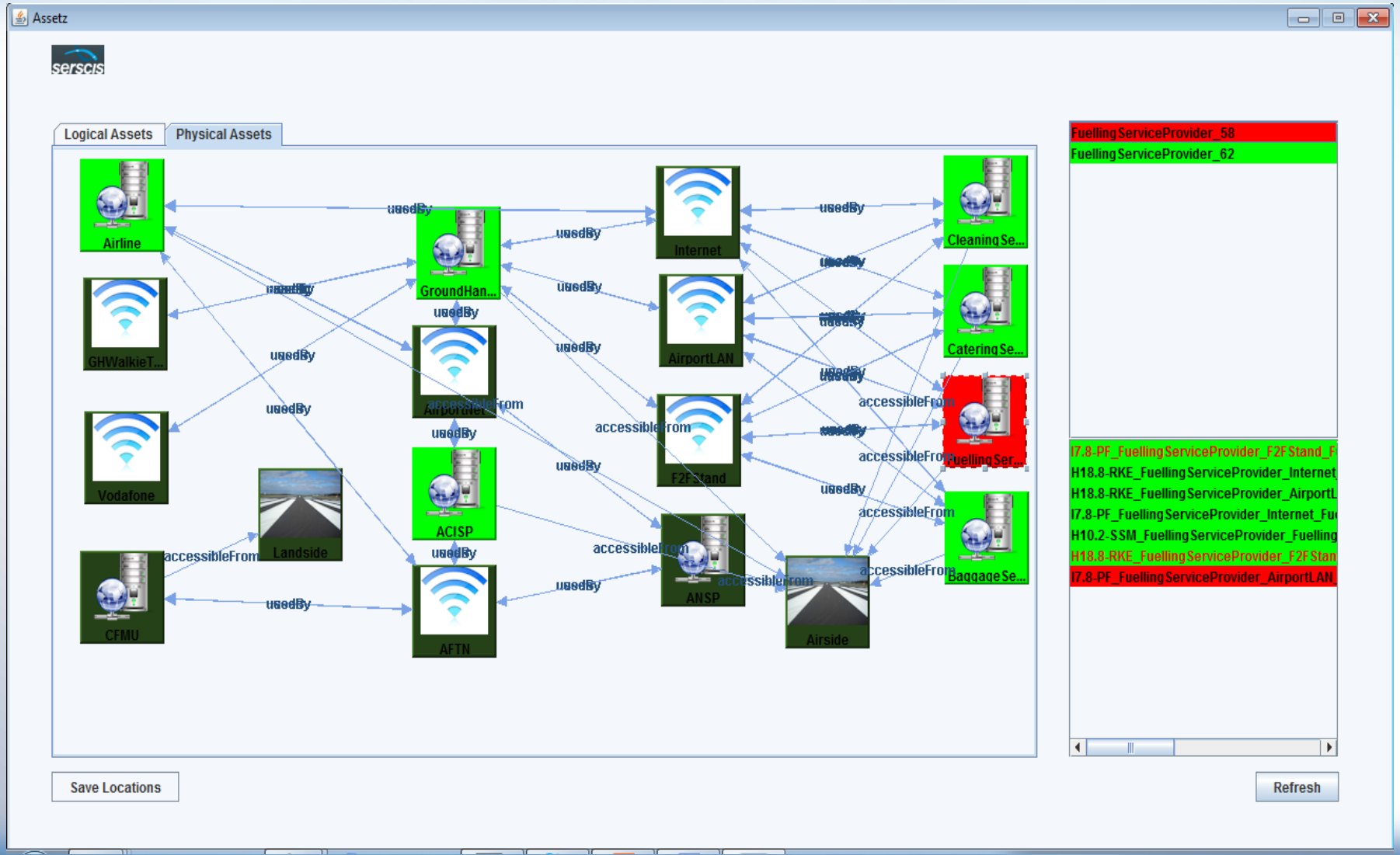
DST – Tool Dynamic Interfaces
Scenario 1: Remote exploitation on Fuelling Services



AttacK: Attacker on the AirportNet network targets the Host of the Fuelling Service.

RKE: **R**emote **K**nown **E**xploit

DST interface and Risk Analytics (Threats Involving Selected Asset)



DST interface (Threat Information and Countermeasure Suggestion)

Research Steps in the GCC framework

- Sequential detection of a change using the nonparametric CUSUM in the Behavioral Analyzer.
- Situational Awareness of the Operators using user friendly Dynamic Support Tool (DST) interfaces
- Development of additional detection approaches (**S**equential **P**robability **R**atio Test, Different Optimality Criteria such as: Lorden, Shiryaev - Roberts)
- Distributed Real Time Sequential Detection & Hypothesis Testing for Intrusion Attacks
- Incorporate Adaptive Methods for activity Monitoring with Forward – Backward Recursive Least Squares Recursions

Linear Model based Process generating data for activity monitoring

- To detect Outliers and Change Points over a stream in an "On-Line" adaptive fashion !!!!.
- Linear Models and Parameter Estimation.

$$\hat{\theta}(t) = \hat{\theta}(t-1) + L(t)[y(t) - \hat{\theta}(\tau-1)\varphi(t)]$$

$$L(t) = \frac{P(t-1)\varphi(t)}{1+\varphi^T(t)P(t-1)\varphi(t)} \qquad P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(\tau)P(t-1)}{1+\varphi^T(t)P(t-1)\varphi(t)}$$

P(0) is the Initial Condition of the Recursive algorithm for Initialization.

Conclusions

- **Implementation** of an Intelligent Prototype Tool for the Protection of Dynamic Multi Stakeholder SOA Critical Infrastructures. Air-traffic Management Systems PoC.
- **Implemented**: An Innovative core ontology model which has been reinforced with rules and classes that improve threat estimation and classification.
- **Implemented:** Advanced Stream (RDF) Reasoning – and Behavioral Analysis Algorithms.
- **Sequential data analysis** led us to Advanced Semantic Stream Reasoning for Real –Time Processing.
- **Implemented:** Dynamic User Interfaces with Risk – Threat Analytics in Real Time for A-CDM (Eurocontrol).

Questions – Discussion.

Thank you !

Contact Details:

Vasilis Tsoulkas	tsoulkas.kemea@gmail.com
Dimitris Kostopoulos	dimkostopoulos@gmail.com
George Leventakis	george.leventakis@gmail.com
Prokopis Drogkaris	prokopis.drogkaris@gmail.com
Viky Politopoulou	v.politopoulou@gmail.com