ΚΕΝΤΡΟ ΜΕΛΕΤΩΝ ΑΣΦΑΛΕΙΑΣ
CENTER FOR SECURITY STUDIES

CRITIS 2013

8th International Conference on Critical Information Infrastructures Security

SEPTEMBER 16-18, 2013, AMSTERDAM

Real Time Threat Prediction, Identification and Mitigation for Critical Infrastructure Protection using Semantics, Event Processing and Sequential Analysis

R&D Team

Presenter: Vasilis Tsoulkas.

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.

2

Presentation

- **Motivation and Objectives**
- **Critical Infrastructure Description**
- **Semantic System Modeling Aspects** (The CI Modeling Challenge)
- **Monitoring and Stream Reasoning Process** (Behavior Analyzer and NP-CUSUM)
- **Decision Support Tool View**
- **Future Directions - Conclusions**

3

Motivation and Objectives

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

4

Motivation and Objectives

Critical Infrastructures vulnerabilities

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

2. **Local Disruptions** in one System is distributed to other coupled sub-Systems

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

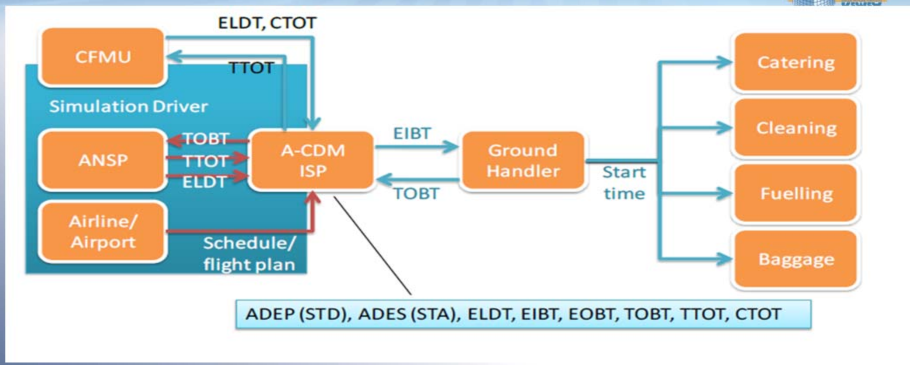
5

R&D Objectives

- Implementation of **Agile Service Oriented Technologies** for Multi-Stake Holder Systems for:
  - **Dynamic composition of** ICT connections of the CI 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 for Real Time Operator Assistance.

6

CI Desciption: An Airport-Collaborative Decision Making – European Air Traffic Management System



Services are accessible by a consumer (**aircraft operator**) through **SLA templates**.
The Ground Handler coordinates the 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

7

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**

8

Semantics Systems Modeling Aspects: The CI Modeling Challenge

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

1. **Core ontology structure:** to model 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.
3. **Abstract system model:** describes system-specific threats and counter-actions.
4. **Concrete system model:** provides snapshots of the running system and instances of the participating assets + contextualised threats & controls.

9

Brief Analysis of adopted System Ontology & CI Modeling

1. **The Semantic Ontology** is constructed so 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.

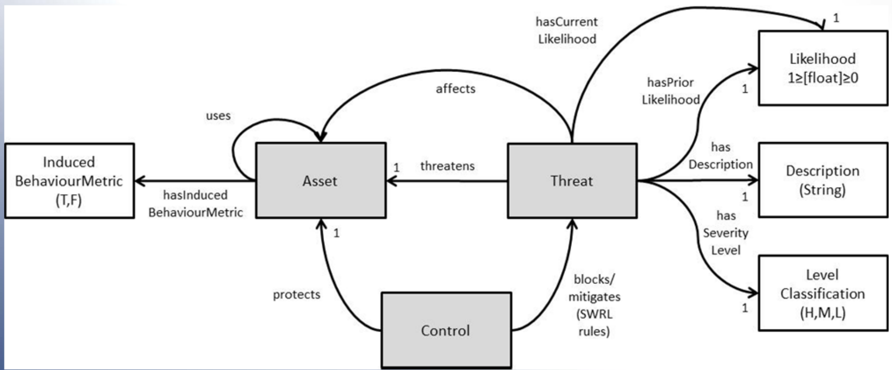
10

Innovation of the Approach

- The Modelling approach is constructed using Semantics Modelling for Intelligent 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".

11

Core System Domain Ontology Schematic



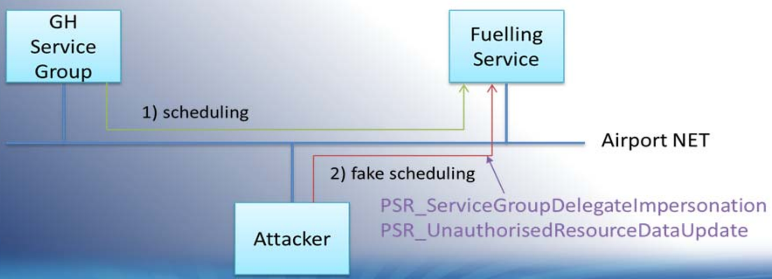
- This basic system structure, determines what reasoning is used

12

Threats & Threat Proof of Concept Scenario : 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



13

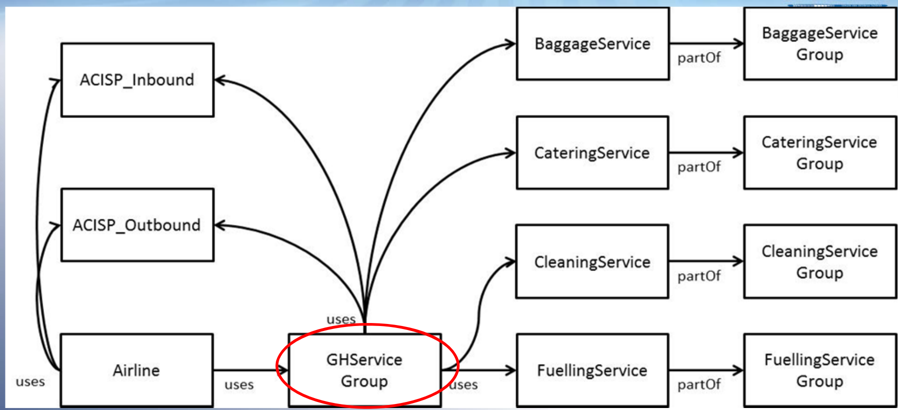
Counter – Actions (Control) Class Explanation

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;
- **description of mitigation actions**: how to operate reactive controls to protect assets when a threat is carried out against them.

14

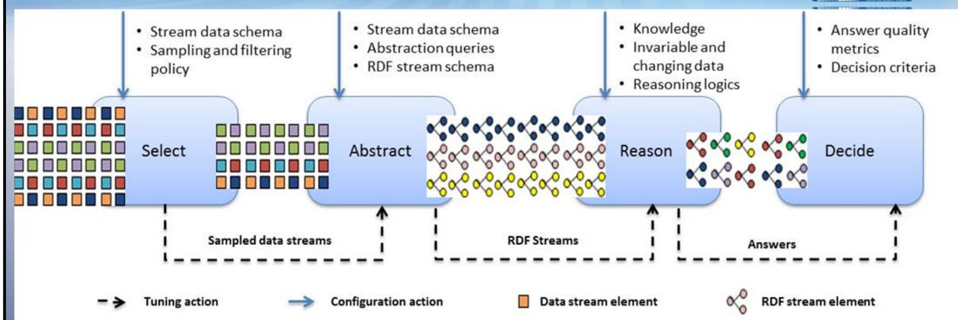
Abstract System Model of multi-stakeholder CI



- 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**.

15

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.

16

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*.

17

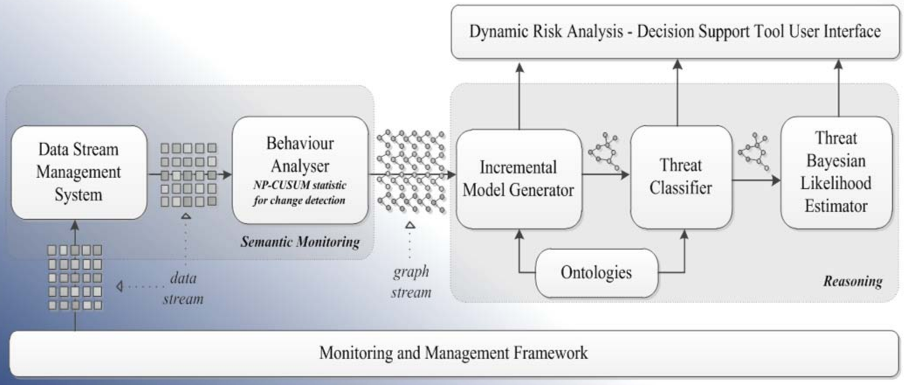
4 basic – steps in Stream Reasoning

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.

18

Semantics Driven Architecture

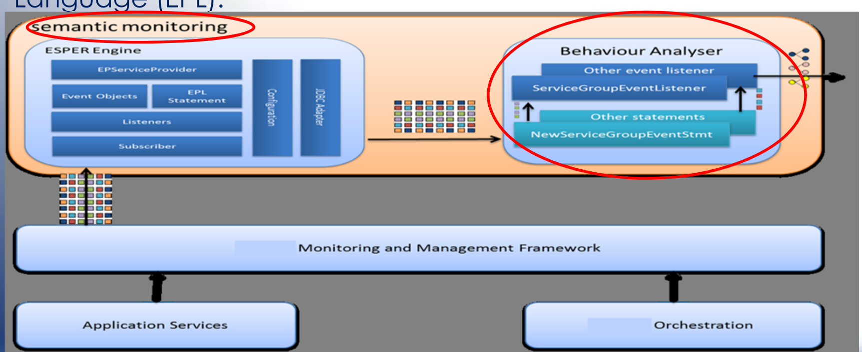


19

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).



20

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.

21

Behavior Analyser (BA)

- The **BA** is not only a Transcoder converting Monitoring Events to time stamped - RDF graphs.
- The **BA** decides about the type of Behaviors of 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).

22

Sequential Inspection

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

23

NP-CUSUM basics

Non-Parametric CUSUM test

Random Data Process Sequence (transformed)

(1). $Z_n = a + \xi_n \, I(n<m) + (h+\eta_n)\, I(n \geq m)$

$\xi = \{\xi_n\}_{n=1}^{\infty}, \eta = \{\eta_n\}_{n=1}^{\infty}$ are zero mean random sequences

$h \neq 0$ and $I(H)$ is the indicator function. Equals "1" when condition H is satisfied and "0" otherwise

Formal definition of NP-CUSUM

$$y_n = S_n - \min_{1 \leq k \leq n} S_k, \text{ where } S_k = \sum_{i=1}^{k} Z_i \text{ and } S_0 = 0$$

y_n : is the test statistic

24

NP-CUSUM basics

Recurrent version of NP-CUSUM

$$y_n = (y_{n-1} + Z_n)^+$$
$$y_0 = 0$$
$$X^+ = \max(0, x)$$
$$(X^+ = x \; if \;\; x > 0 \text{ and } 0 \text{ otherwise}).$$

Decision stopping rule of CUSUM

$$d_N(y_n) = \begin{cases} 0 & if \;\; y_n \le N \\ 1 & if \;\; y_n > N \end{cases}$$

N: Attack detection threshold

y_n represents the cumulative positive values of Z_n

A large value of y_n is a strong evidence of attack

(see 3rd graph of next slide)

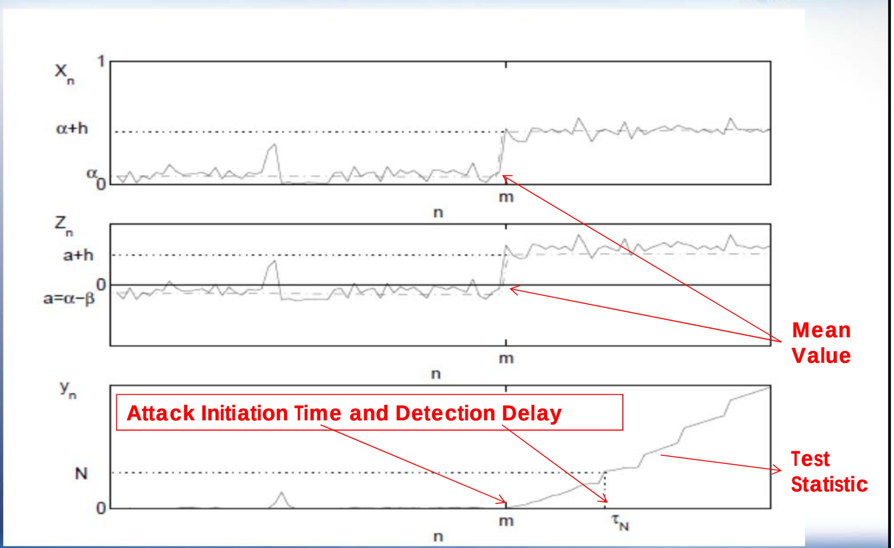
Two basic contradicting performance criteria of NP-CUSUM:

i). False Alarm Time

ii). Detection Time.

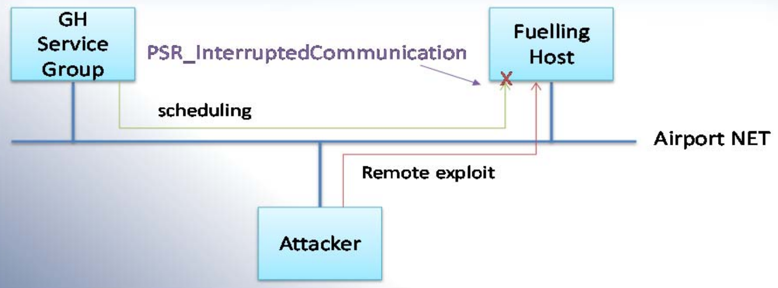
25

Sequential Intrusion (Behavior) Detection



26

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

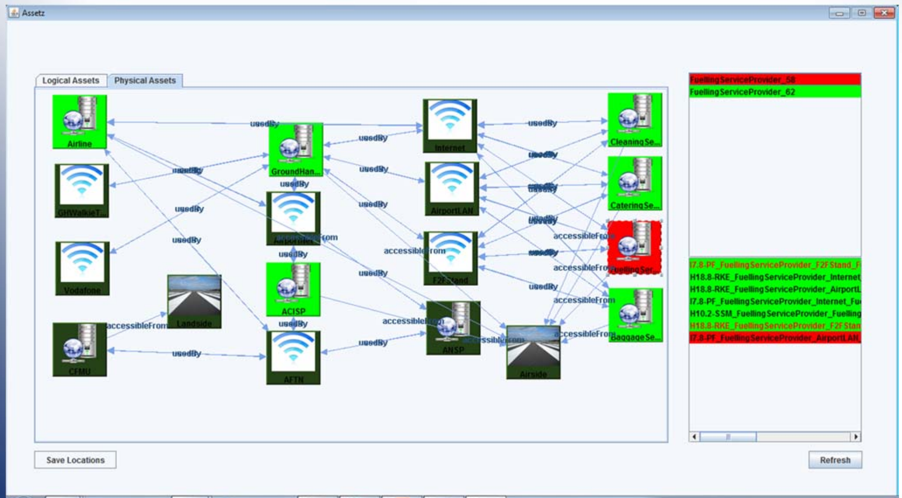


AttacK: Attacker on the AirportNet network targets the Host of the Fuelling Service.

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

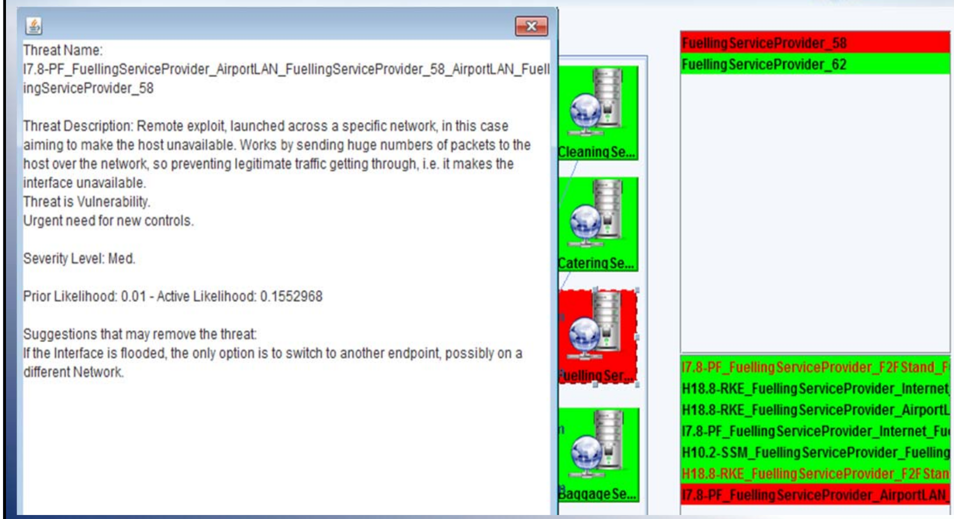
27

DST interface and Risk Analytics
(Threats Involving Selected Asset)



28

DST interface (Threat Information and Countermeasure Suggestion)



29

Future Directions

- 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

30

Linear Model based Process generating data for activity monitoring – RLS type algorithms

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

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

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

31

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).

32

Thank you
Contact Details:
Center for Security Studies (KE.ME.A)
www.kemea.gr
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
33