System-of-Systems Security for Crisis Management

Kashif Kifayat, Abdullahi Arabo, Oliver Drew, David Llewellyn-Jones, Madjid Merabti, Qi Shi

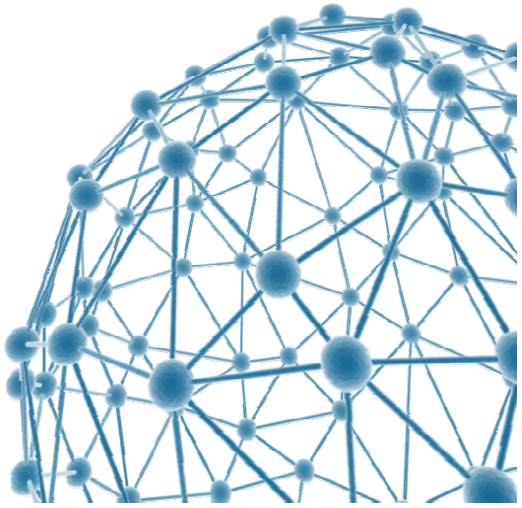
Liverpool John Moores University

Adrian Waller, Rachel Craddock, Glyn Jones

Thales Research and Technology (UK) Ltd.

Outline

- Background
- Challenges
- Our Approach
- Requirements
- Implementation
- Scenario
- Future Work



Network image by gerard79: http://www.sxc.hu/photo/1008231

THALES

- A crisis or disaster is a natural or man-made disruptive event
  - 2005 Hurricane Katrina: over 1604 people died with estimated financial loss of $25-$100 billion
  - 2005 Buncefield Oil Depot Explosion: 0 deaths and $1billion
- ICT requirements in crisis management
  - Dynamic ad-hoc communication network between different agencies using System-of-Systems approach
    - Each agency will have its own networks and systems
  - Secure information sharing without delay

Image courtesy of Chilton Air Support Unit and Hampshire Constabulary

- System-of-Systems Security and Assurance Issues
  - Interaction between component systems may affect the security and assurance of the overall system
    - Assurance in components is important and can be built on, but is not enough in itself
    - Crisis situations are highly dynamic and unpredictable, and it is not possible to engineer components to be suitable for all situations
  - Assurance takes time to establish
  - High security may have an adverse effect on information flow
    - Need balance between security and operational effectiveness

Image http://www.istockphoto.com

- Secure System Composition Modelling and Evaluation
  - Analyse different pre-deployment System-of-System scenarios
  - Highlight the post deployment security issues in real operations

- Secure System Composition Dynamic Analysis
  - Tools and techniques for dynamic analysis (component analysis and composition analysis)

- Provide users with composition assurance layer tools to help them compose systems rapidly and intelligently
  - Want to avoid putting together whatever is available and “hoping for the best”
  - Build on known and assured properties of individual components
  - Establish known and assured properties of potential composed systems
    - Enable different combinations of components to be investigated, and selection of the most appropriate
- Although our focus is System-of-Systems, the tools and techniques proposed here should be applicable more widely

Requirements

- User-friendly Interface
- Clear Message and Indication
- Identify and prompt users of possible security problems
- Highlight risk areas dynamically
- Suggest potential solutions to mitigate risks
- Dynamic real time analysis
- Automated analysis
- Applicable in dynamic, mobile networks

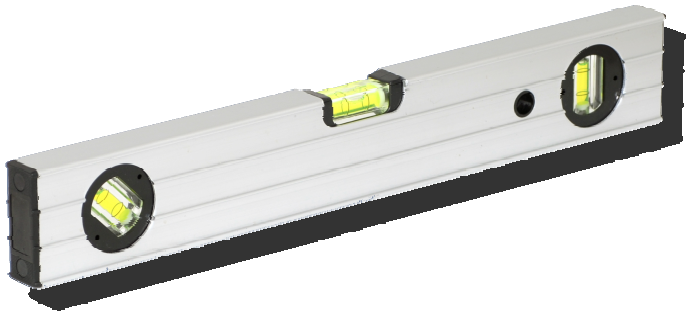
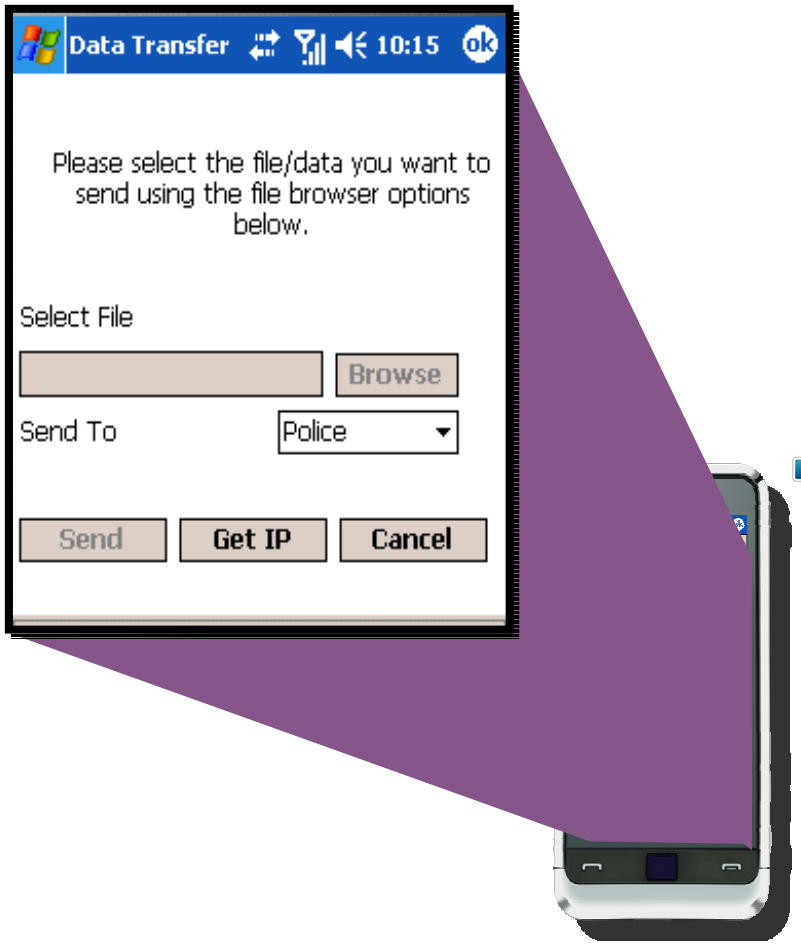


Image by topfer: http://www.sxc.hu/photo/871496

MATTS (the Mobile Agent Topology Test System)

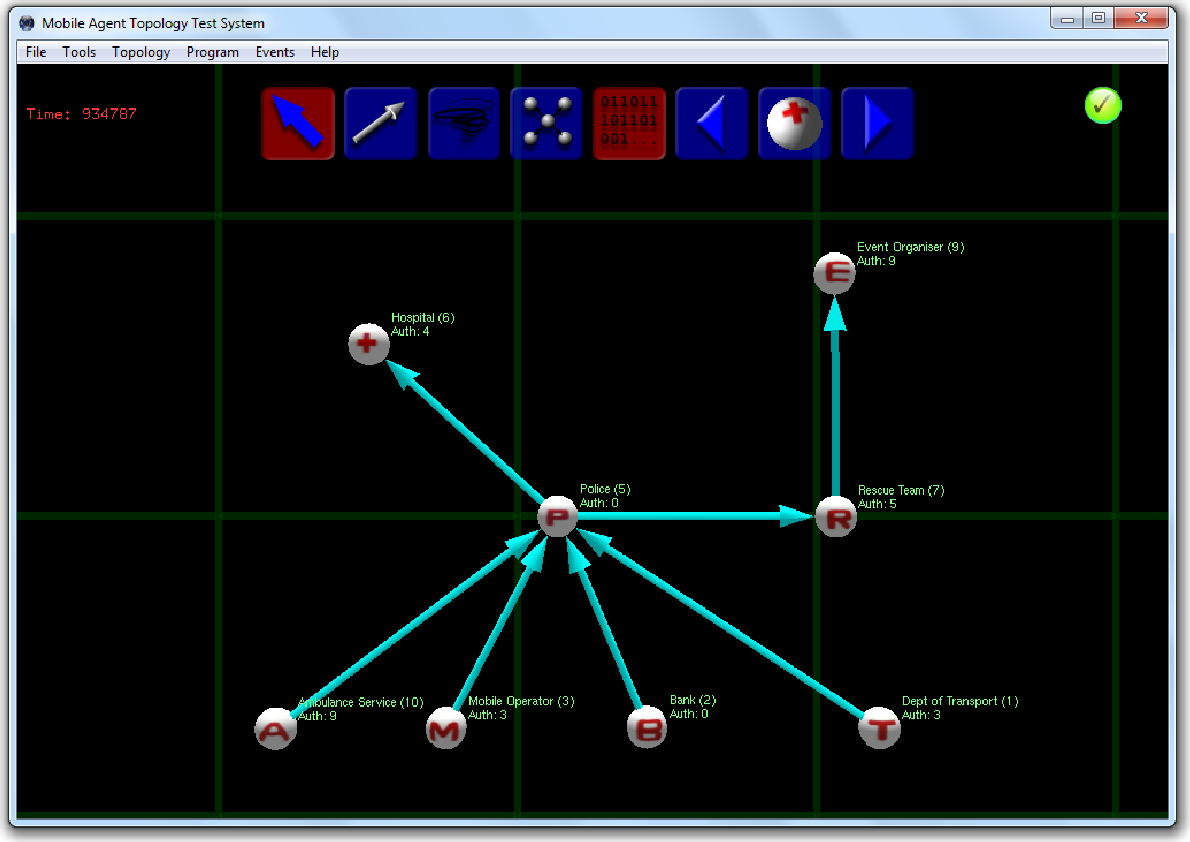


- Consists of two applications
  - Composition client
    - Represents any organisation (police, fire service, paramedical, etc.) which could participate in crisis management
    - Included with communication devices (PDAs, smart phones, laptops, etc.)
    - Each client has a set of security properties and policies
  - MATTS server
    - Receives client information (connectivity, security policies, etc.) in an XML file
    - Runs composition analyses to identify vulnerabilities and threats according to scripts

THALES

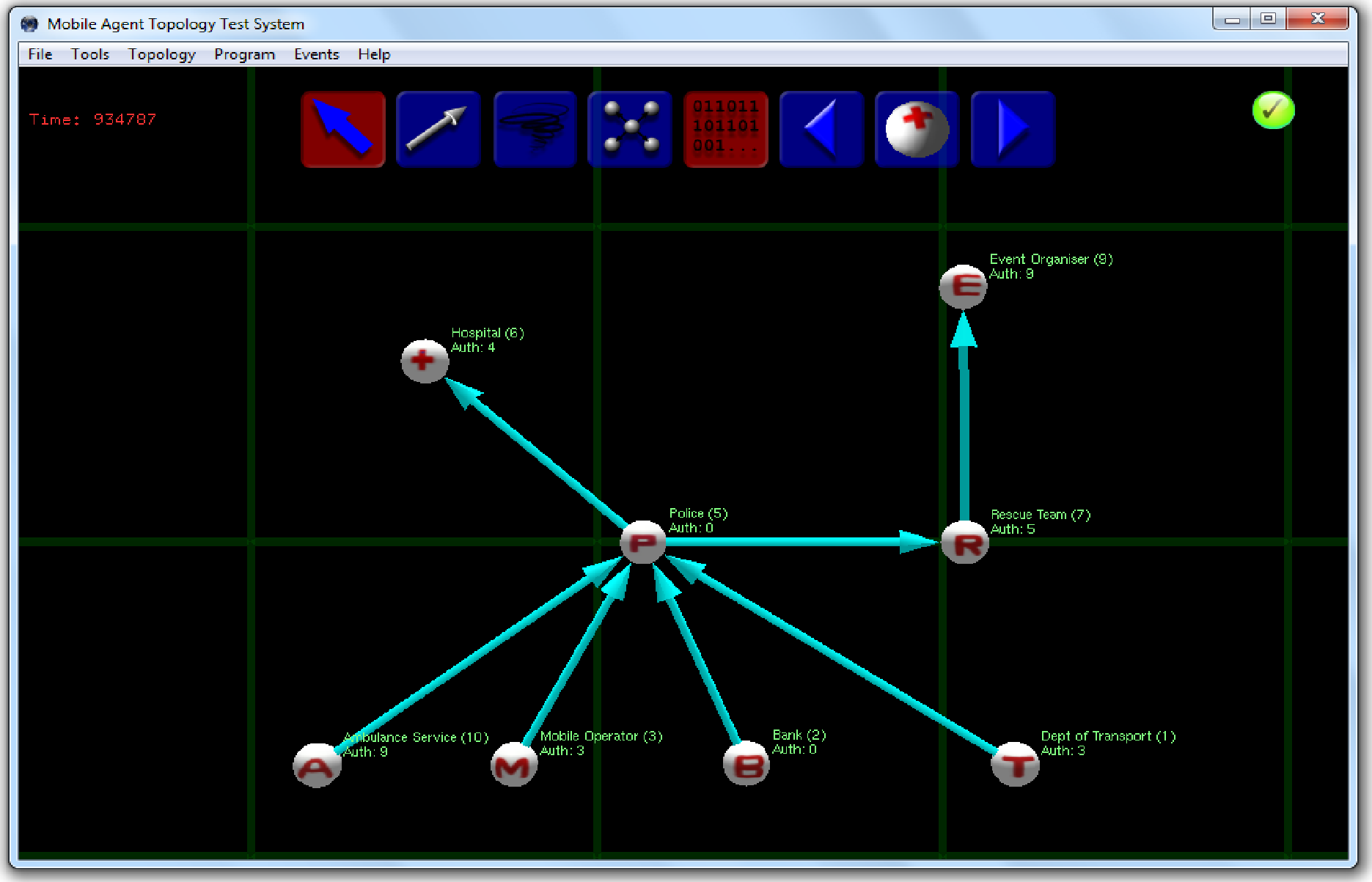
MATTS (the Mobile Agent Topology Test System)

- Gives freedom to model many possible scenarios
  - Different numbers of nodes
  - Different security properties
  - Different vulnerability tests



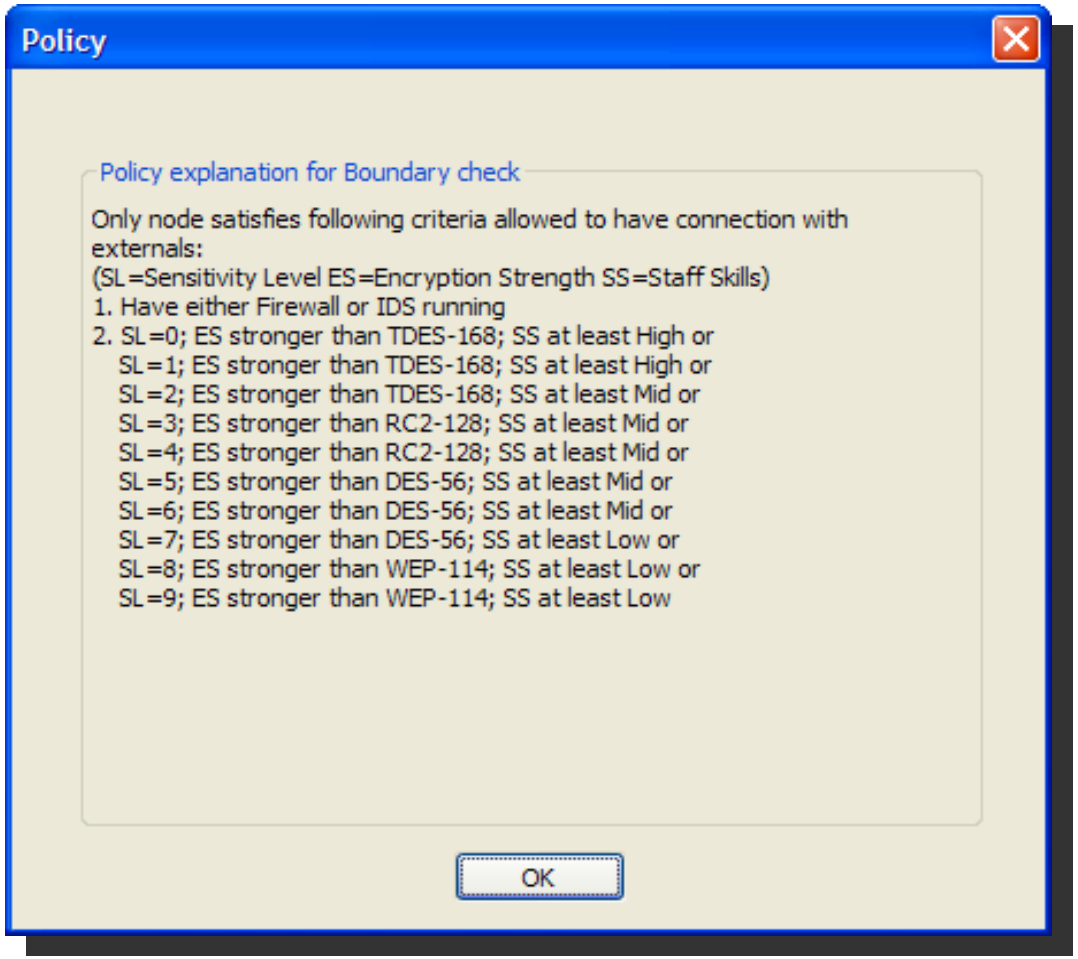
THALES

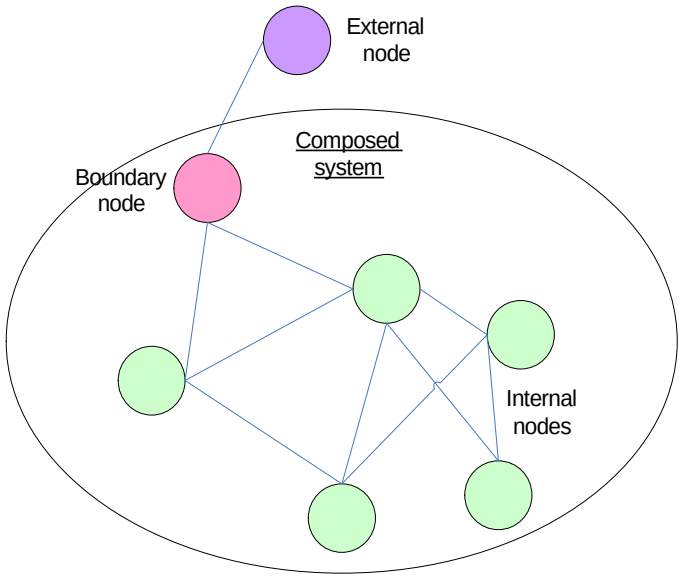
Example – Boundary Check Scenario (Initial Network)



THALES

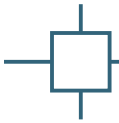
Example – Proposed Connection





THALES

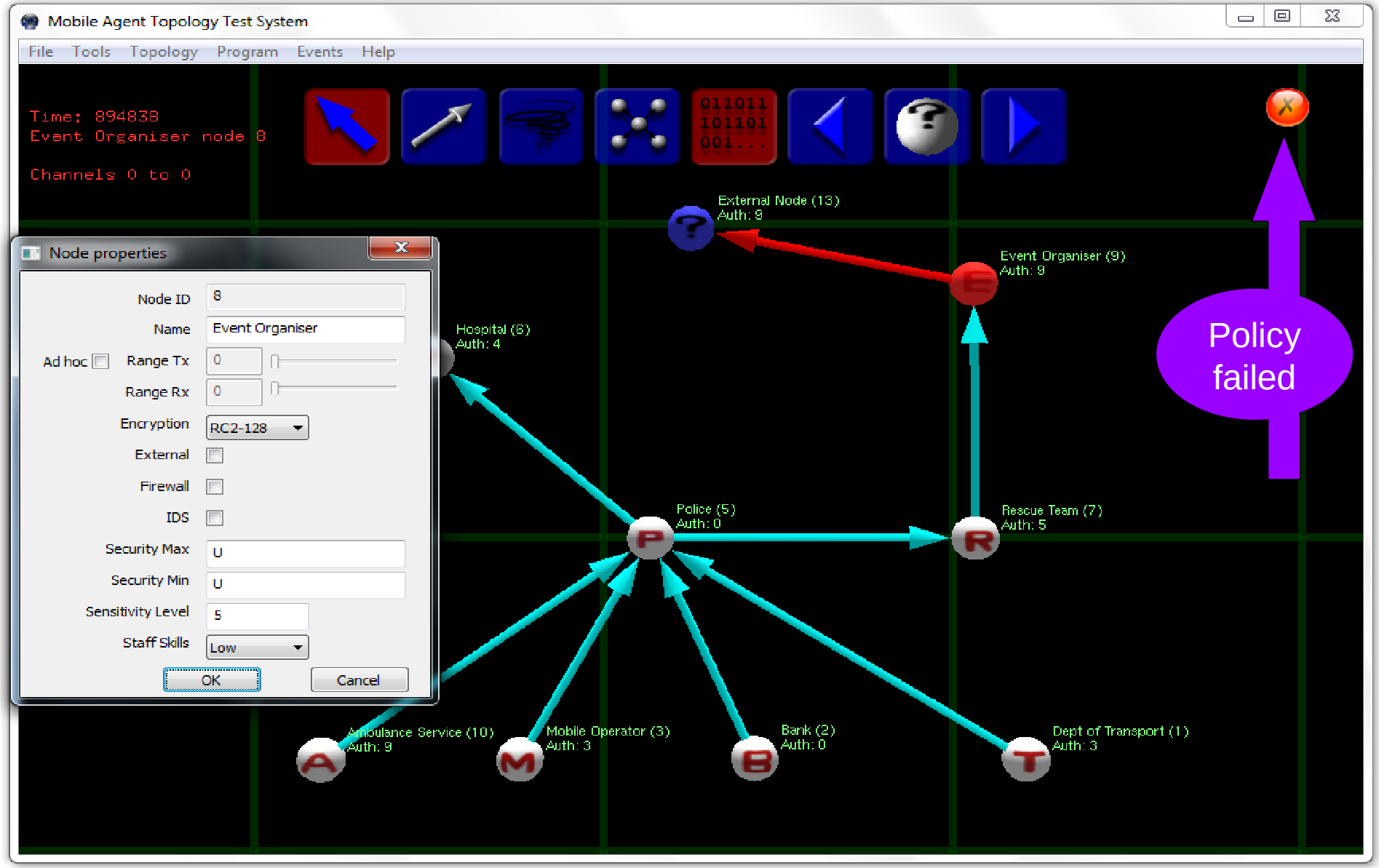
Example – Policy



"A component will only be allowed to have external connections when it has either Firewall or IDS running, and its Encryption Strength and Staff Skills satisfy the minimum requirements imposed in accordance with its Sensitivity Level."

```
<process id="check">
  <process action="link = @ilnum[@n]"/>
  <process id="link" init="0">
    <process action="link = (link-1)"/>
    <process init="0" cond="(!@a[@iln[@n][link]][External]) && (!@a[@iln[@n][link]][Firewall])
      && (!@a[@iln[@n][link]][IDS])" action="(safe=0)"/>
    <process id="ex" init="0" action="ex=@a[@iln[@n][link]][External]"/>
    <process id="sl" init="0" action="sl=@a[@iln[@n][link]][SensitivityLevel]"/>
    <process id="es" init="0" action="es=@a[@iln[@n][link]][EncryptionStrength]"/>
    <process id="ss" init="0" action="ss=@a[@iln[@n][link]][StaffSkills]"/>
    <process init="0" cond="(!ex) && (sl == 0) && ((es < 11) || (ss < 3))" action="(safe=0)"/>
    <process init="0" cond="(!ex) && (sl == 1) && ((es < 11) || (ss < 3))" action="(safe=0)"/>
    ...
    <process cond="link > 0" config="link"/>
  </process>
</process>
```

Example - Analysis Result



THALES

Implemented composition analyses

- Boundary Check
  - Ensuring the System-of Systems has a secure boundary
- Data Flow Security
  - Ensuring data cannot flow to locations with insufficient security
- Buffer Overrun
  - Detecting where buffer overrun vulnerabilities can be exploited
- Cascade Vulnerability
  - Detecting if a chain of systems can be compromised in order to access data

Conclusions and Future Work

- Current implementation
  - Extensible, automated means of detecting Systems-of-Systems security issues
  - Highlights potential problems dynamically as topology changes
  - Reasons using device properties and topological structure
  - Combines real devices and modelled nodes

- Future work
  - More automated analysis to investigate different scenarios and policies
  - Policy reconciliation
  - Correction to automatically address detected problems
  - Test in larger, real-world scenarios

Image by flaivoloka: http://www.sxc.hu/photo/994582

THALES