STAR-TRANS Modeling Language (STML) Modeling Risk in the STAR-TRANS Risk Assessment Framework for Interconnected Transportation Systems

Dimitris Zisiadis
Centre for Research & Technology Hellas
dzisiadis@iti.gr
Spyros Kopsidas
Centre for Research & Technology Hellas
kopsidas@iti.gr
Vassilis Grizis
Center for Security Studies (KE.ME.A.)
vgrizis@kemea.gr

George Thanos
Centre for Research & Technology Hellas
gthanos.iti@gmail.com
George Leventakis
University of Aegean
george.leventakis@gmail.com
Leandros Tassiulas
Centre for Research & Technology Hellas
leandros@iti.gr

ABSTRACT

The present paper introduces a high level modeling language, capable of expressing the concepts and processes of the Strategic Risk Assessment and Contingency Planning in Interconnected Transportation Networks (STAR-TRANS) framework. STAR-TRANS is a comprehensive transportation security risk assessment framework for assessing related risks that provides cohered contingency management procedures for interconnected, interdependent and heterogeneous transport networks. STAR-TRANS modeling Language (STML) is a domain specific language combining language simplicity with a very clear syntax, providing all the necessary elements (assets, threats, incidents, consequences etc.) to model the STAR-TRANS risk assessment framework.

Keywords

Modeling language, risk assessment, heterogeneous transport networks, STAR-TRANS, STML

INTRODUCTION

The Strategic Risk Assessment and Contingency Planning in Interconnected Transportation Networks (STAR-TRANS) framework (Leventakis, Sfetsos, Moustakidis, Gkrizis, Andronopoulos, Athanasiadis, Ramfos, Tonjes, Zisiadis, Kopsidas and Nikitakos, 2011) provides a comprehensive transportation security risk assessment framework for assessing related risks and provides cohered contingency management procedures in interconnected, interdependent and heterogeneous transport networks. The objective of the current work is to develop a high level interface modeling language to implement the STAR-TRANS framework, namely the STAR-TRANS modeling language (STML), capable of representing the objects required for defining an impact assessment process, i.e., incidents, structure and assets of networks, dependencies between assets. STML is a domain specific modeling language that aims to express every single bit of operation a security manager/expert is expected to perform in order to estimate risk in a transportation network or a network of networks using the STAR-TRANS framework. STML implements all the bits and pieces of the aforementioned process, without automating any business logic. The end-user (security manager/security expert) is responsible to make the appropriate scenario decisions in order to successfully execute a complete threat scenario.

STAR-TRANS FRAMEWORK FUNDAMENTALS

Transportation networks are open and accessible, by design, and thus contain vulnerabilities that may be exploited for malicious purposes. Furthermore, any damage or operational failure on an asset of a transportation network can lead to cascading failures or even damages on interoperable assets of the same network. Since transportation networks are tightly coupled with each other, a failure in one network can spread failures or functional disorders in interconnected transportation networks, leading to widespread operational disturbance. The wider set of interconnected networks can be viewed as the “network-of-networks”. The STAR-TRANS risk assessment framework models those kinds of cascading failures and aims to estimate the overall risk along a network or the network-of-networks, given that an initial triggering event takes place. The framework consolidates the concept that a failure into one point of the network can lead to further failures to interconnected assets, using on a Markovian chain approach. A thorough presentation of the STAR-TRANS framework can be found in (Leventakis et al., 2011) and in (Thanos and Zisiadis, 2011).

RELATED WORK

Risk assessment for transportation networks is an important issue both for transport operators as well as national security authorities. The tools and methodologies that have been developed for risk assessment, like CRAMM (Yazar, 2002), CORAS (Lund et al, 2011; OMG document formal/06-05-02, 2006; Dahl et al., 2007), are either generic and are not considered suitable for modeling transportation networks, or are more or less focused on a specific application areas, irrelevant to transportation. Tools and methodologies to address risk assessment in the field of information technology and software design have been proposed by recent research (Mayer and Heymans, 2007; Strecker, Heise and Frank, 2011). A semi-formal, UML based modeling language that enhances collaboration between process and security managers, within business organizations is proposed in (Sienou, Lamine, Karduck and Pingaud, 2008). A framework for risk assessment in dangerous good transportation is proposed in (Fabiano, Currò, Palazzi and Pastorino, 2002), failing to suggest any tool or language to assist the risk assessment process. Risk assessment methodologies for maritime transportation are presented in (Soares and Texeira, 2001).

All of the above proposals describe specific methodologies and/or provide tools to assist risk assessment either in a generic way not applicable to transportation (Yazar, 2002; Lund et al., 2011) or they target specific domains irrelevant to transportation (Mayer, 2007; Strecker, 2011; Sienou et al., 2008; Fabiano et al., 2002; Soares, 2001). The concepts and procedures presented being interesting, nevertheless none of them introduces a modeling language per se, that is powerful, dynamic and able to express the risk assessment process for the specific domain. In the STAR-TRANS framework, a domain specific modeling language has been introduced, as it is described in the present work.

STAR-TRANS MODELING LANGUAGE

STAR-TRANS Modeling Language (STML) is a domain specific language that can fully model the STAR-TRANS framework. STML is a high level, request-response, command-line interface (CLI) language that represents all the concepts and ideas introduced by STAR-TRANS. In particular, STML provides an interface to a software component, the STML Engine Core (STML-EC). STML-EC models the STAR-TRANS framework and maintains the state of the risk assessment process extracting the risk for particular assets, the network and the network-of-networks. STML-EC is part of a larger system that is assumed to be a GUI-enabled tool that aids the transportation network security managers and risk analysts in the process of risk assessment for a predefined network. STAR-TRANS implements the Impact Assessment Tool (IAT) as a GUI-enabled, full-featured platform for risk assessment, as depicted in Figure 1.

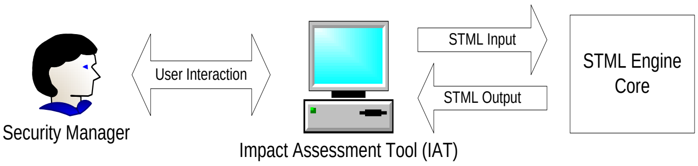

Figure 1: Graphical representation of the interaction between the end-user and the STAR-TRANS tool

STML Object Types

STML defines the following object types:

Threats: Threats are the blueprints (abstract data types) of Incidents. A threat is potential hazard that can be instantiated in a network asset. STML enables the creation, deletion and modification of threats.

Consequences: according to STAR-TRANS analysis, consequences apply to any security incident of the transportation network. STML supports the creation deletion and modification of consequence types.

Threat Consequence Matrix: Threat Consequence Matrix is a two dimensional matrix where inputs are a) defined threats of possible instantiated incidents and b) the available consequence types. The matrix values are Boolean values of true/false (1/0) indicating whether a consequence is valid for an instantiated threat or not. Upon a new incident, any consequence type should have a severity value, only if threat consequence matrix defines that consequence is valid for the target threat.

Incident Propagation Matrix: Incident Propagation Matrix is a three dimensional matrix where inputs are a) threats of instantiated incidents, b) threats of propagated incidents on the immediately interconnected assets and c) the interdependency type under investigation. The matrix shows in a consolidated form the likelihood of threats that may be triggered in interdependent assets based on the threat of originating security incidents.

Networks and Network-Of-Networks: The STAR-TRANS framework is built on the concept of the transportation network as an integral part of a wider set of transportation networks called the "network-of-networks". STML follows this approach by defining a hierarchical set of objects based on already predefined object types. In particular STML defines the following hierarchy: *a) the "network-of-networks", b) transportation networks that belong to the "network-of-networks" and c) assets that belong to a predefined network of the "network-of-networks"*.

Assets and Asset Interdependencies: Assets are the primary elements of the transportation network. Assets can be *direct assets (vehicles, stations, transportation load etc.)* or *indirect assets (communication, signalling and power supply)***.** Assets interoperate with each other according to a set of predefined interdependency types. The following four interdependency types are defined in the STAR-TRANS framework: *Physical, Systematic, Geographical* and *Logical*.

Incident: Incident is an instantiation of a threat of a specific asset. Incidents can be introduced by security experts as triggering security bleach events on the transportation network or can be introduced as side effects or previously created incidents.

Scenario: Threat Scenario is a collection of incidents that form an autonomous set of incidents.

STML Statements

As a high-level command-line interface language STML defines the following types of statements:

- **create:** Create statements define the creation of new language objects. Language objects can be any of the following: *a) networks, b) network-of-networks, c) assets, d) threats, e) consequences and f) threat scenarios.*
- **delete:** Delete statements define the deletion of previously created language objects.
- **set/unset:** Statements that set or unset specific parameters of language objects.
- **show:** Show information regarding the state of the transportation network, such as *network or network-of-networks topology information, asset parameters or asset dependencies information, consequence values, likelihood values, threat scenario state, risk assessment on specific assets, networks, or network-of-networks*.
- **attach/detach:** Attach/detach an asset to a network or a network to a network-of-networks.

The complete set of STML statements and detailed analysis is given in (Thanos and Zisiadis, 2011).

STML IMPLEMENTATION DESIGN

STML is based on a request-response messaging scheme between the end user and the software component that maintains the state of the risk assessment process named as the STML Engine Core (STML-EC), as depicted in Figure 2 below.

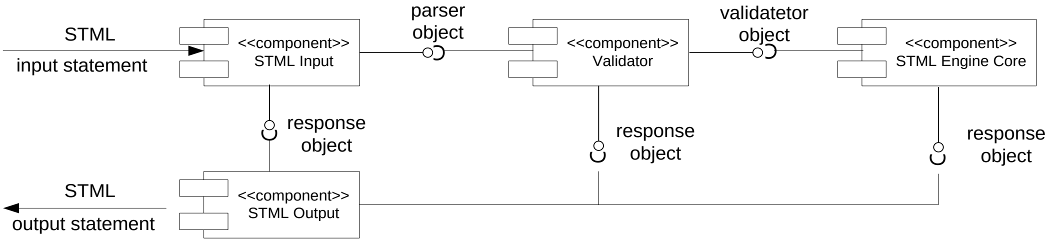

Figure 2. UML 2.0 Component Diagram for the STML Engine Core

STML Parser: Parses STML Input statements using a two level process and either generates an error object in case the syntax is incorrect or handles the result to the STML Validator.

STML Validator: Processes the output of the STML parser and either generates an error object (when validation fails) or handles its validation output to the Engine Core.

Engine Core: Receives as input the output of the Validator and performs any task related to risk assessment.

STML Response Generator: Receives response objects from the STML Parser, the STML Validator and the Engine Core and generates the required STML response message.

Analysis of STML Parser

The STML Parser can parse STML statements in a two level process. In the first level the type of the STML statement is identified. Based on this parsing information the corresponding parsing method for the actual STML statement is invoked. This second more detailed level of processing, further decomposes statements in their syntactic elements and extracts the contained information. STML statements are categorized in five separate groups, based on the first word of each statement command: *create, delete, set/unset, show* and *attach/detach*.

Each group contains a set of commands as outlined in the UML 2.0 activity diagram of the STML Parser for the "create" group of statements of Figure 3 below.

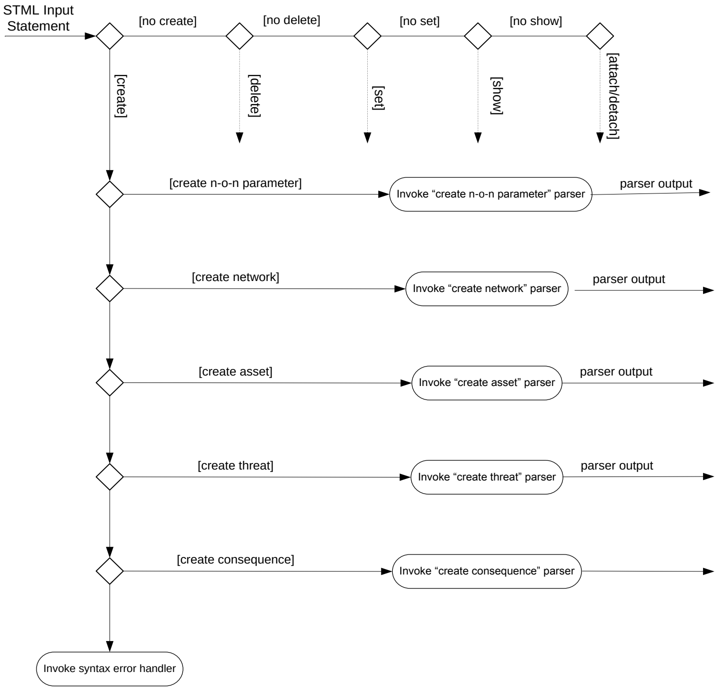

Figure 3. UML 2.0 activity diagram for the STML parser for the "create" group of statements.

CONCLUSIONS

In this paper we presented a modeling language for expressing risk in the STAR-TRANS framework for risk assessment in interconnected transportation networks, namely the STAR-TRANS modeling language (STML). STML is a high level interface modeling language, capable of representing the objects required for defining an impact assessment process, i.e. incidents, structure and assets of networks and dependencies between assets. An implementation design has also been outlined based on a request-response messaging scheme between the end user and the software component that maintains the state of the risk assessment process named as the STML Engine Core (STML-EC).

FUTURE WORK

Latest advances in software engineering tend to view software as a service (SaaS) (Software & Information Industry Association, 2001) rather than stand-alone, library based processing entity. Following this approach, software is built using a Service Oriented Architecture (SOA) (Erl, 2004; Erl, 2005; Bieberstein, Bose, Fiammante and Jones, 2005) enabling the utilization of software technologies of different vendors and leading to easier software decomposition, better module integration and more stable software products (Channabasavaiah, Holley and Tuggle, 2004; Stevens, 2002; Bieberstein, Bose, Walker and Lynch, 2005; Fowler, 2002). Trying to adapt to the aforementioned requirements for STML and the STML-EC, we plan to provide an alternative, more scalable interface using the technology of REST Web-Services. As an interface language STML can easily adapt to the requirements of web-services architecture by mapping web-services to STML statements and passing the required parameters and response messages using XML or JSON technologies.

ACKNOWLEDGEMENTS

This work is part of the project “Strategic Risk Assessment and Contingency Planning in Interconnected Transportation Networks” ICT-FP7-225594 STAR-TRANS, funded by the European Commission.

REFERENCES

1. Leventakis, G., Sfetsos, A., Moustakidis, N., Gkrizis, V., Andronopoulos, S., Athanasiadis, N., Ramfos, A., Tonjes, S., Zisiadis, D., Kopsidas, S. and Nikitakos, N. (2011) A security risk analysis framework for interconnected transportation systems, *Proceedings of the 8th International Conference on Information Systems for Crisis Response and Management, ISCRAM2011*, Lisbon, Portugal.
2. Thanos, G. and Zisiadis, D. (2011) STML Wiki, available online from http://startrans.iti.gr/wiki/
3. Yazar, Z. (2002) A Qualitative Risk Analysis and Management Tool – CRAMM, SANS Institute InfoSec Reading Room, available online from http://www.sans.org/reading_room/whitepapers/auditing/qualitative-risk-analysis-management-tool-cramm_83
4. Mass Soldal Lund, Bjørnar Solhaug and Ketil Stølen (2011) Risk analysis of changing and evolving systems using CORAS. *Foundations of Security Analysis and Design VI (FOSAD'11)*, number 6858 in Lecture Notes in Computer Science, pages 231-274, Springer.
5. Object Management Group (2006) UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and mechanisms. OMG document formal/06-05-02.
6. Heidi E. I. Dahl, Ida Hogganvik, Ketil Stølen (2007) Structured semantics for the CORAS security risk modelling language. *In Pre-proceedings of the 2nd International Workshop on Interoperability solutions on Trust, Security, Policies and QoS for Enhanced Enterprise Systems (IS-TSPQ'07).* Report B-2007-3, pages 79-92, Department of Computer Science, University of Helsinki, 2007.
7. Fabiano, B., Currò, F., Palazzi, E. and Pastorino, R. (2002) A framework for risk assessment and decision-making strategies in dangerous good transportation, *Journal of Hazardous Material*, Volume 93. Pages 1-15, Elsevier.
8. Mayer, N. and Heymans, P. (2007) Design of a Modelling Language for Information System Security Risk Management, *In Proceedings of the First International Conference of Research Challenges in Information Science*, Quarzazate, Morocco.
9. Strecker, S., Heise, D. and Frank, U. (2011) RiskM: A multi-perspective modeling method for IT risk assessment, *Information Systems Frontiers*, Volume 13 Issue 4, Kluwer Academic.
10. Sienou, A., Lamine, E., Karduck, A.P. and Pingaud, H. (2008) Towards a semi-formal modeling language supporting collaboration between Risk and Process Manager, *In Proceedings of the 2008 Second IEEE International Conference on Digital Ecosystems and Technologies*, Phitsanulok, Thailand.
11. Soares, C.G. and Texeira, A.P. (2001) Risk Assessment in maritime transportation, *Reliability Engineering and System Safety*, Volume 74.
12. Software & Information Industry Association (2001) Software as a Service: Strategic Backgrounder, white paper.
13. Erl, T. (2005) Service-Oriented Architecture (SOA): Concepts, Technology, and Design, Prentice Hall, ISBN:0131858580.
14. Erl, T. (2004) Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services, Prentice Hall, ISBN:0131428985.
15. Bieberstein, N., Bose, S., Fiammante, M. and Jones, K. (2005) Rawn Shah, Service-Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap, IBM Press, ISBN-10: 0131870025.
16. Channabasavaiah, K., Holley, K. and Tuggle, E. (2004) Migrating to a service-oriented architecture, IBM Software Group, White Paper, available from ftp://service.boulder.ibm.com/s390/audio/pdfs/G224-7298-00_FinalMigratetoSOA.pdf
17. Stevens, M. (2002), The Benefits of a Service-Oriented Architecture, published at Developer.com.
18. Bieberstein, N., Bose, S., Walker, L. and Lynch, A. (2005) Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals, IBM Systems Journal, Volume 44, p.691-708.
19. Fowler, M. (2002) *"Public versus Published Interfaces" IEEE Software*, Vol. 19, No. 2.