· U C ·

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Filipe Manuel Simões Caldeira

Trust and Reputation for Critical Infrastructure Protection

PhD Thesis in Informatics Engineering,
submitted to the Faculty of Science and Technology of the
University of Coimbra

Advisor: Professor Edmundo Heitor da Silva Monteiro
Co-Advisor: Professor Paulo Alexandre Ferreira Simões

Coimbra, July 2013

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Filipe Manuel Simões Caldeira

Confiança e Reputação em Infraestruturas Críticas

Tese de Doutoramento em Engenharia Informática,
apresentada à Faculdade de Ciências e Tecnologia da
Universidade de Coimbra

Orientador: Professor Edmundo Heitor da Silva Monteiro
Co-Orientador: Professor Paulo Alexandre Ferreira Simões

Coimbra, Julho 2013

Acknowledgments (Agradecimentos)

Muitos foram os que contribuíram, de forma direta ou indireta, para a elaboração desta dissertação. Pretende-se, deste modo, expressar o reconhecimento e gratidão às pessoas e instituições cuja colaboração foi determinante para a sua concretização.

Ao Professor Doutor Edmundo Monteiro, orientador desta dissertação, um agradecimento destacado pelas valiosas sugestões e críticas pertinentes, bem como pelo constante incentivo, cordialidade e amizade.

Ao Professor Doutor Paulo Simões, por todo o apoio nos momentos mais complicados, pelo continuo incentivo e pela constante partilha de conhecimento. Não posso deixar de agradecer a sua amizade e companhia, em particular nas várias viagens que fizemos, nas quais sempre teve a paciência de ouvir.

À Fundação para a Ciência e Tecnologia pela bolsa de doutoramento (SFRH BD / 35772 / 2007).

Aos familiares, amigos e colegas, pela amizade, apoio e colaboração.

À Guida pelo apoio permanente e incondicional.

Ao Luís e à Filipa agradeço a paciência e compreensão por não ter estado mais tempo disponível e pelo mau feitio mostrado nos momentos mais complicados!

Acknowledgments

I would like to thank the FP7 - MICIE Project, that partially supported this work and also to the MICIE participants for all the productive discussions and support.

I would also like to show my gratitude to Thomas Schaberreiter for his collaboration and for the long discussions that we had during this period. I would like to thank also Sebastien Varrette for providing useful data for this work.

Foreword

The work described in this thesis was conducted at the Laboratory of Communication and Telematics (LCT) of the Centre for Informatics and Systems of the University of Coimbra (CISUC) within the context of the MICIE project – *"Tool for systemic risk analysis and secure mediation of data exchanged across linked CI information infrastructures"* - http://www.micie.eu, – The MICIE project was carried out between September/2008 and February/2011 and was financed by FP7 grant agreement no. 225353. The MICIE project, being in line with the European Union initiative to establish a Critical Infrastructure Warning Information Network (CIWIN), aimed to design and implement a so-called *"MICIE alerting system"* able to identify, in real time, the level of possible threats induced on a given CI by "undesired" events happened in such CI and/or other interdependent CIs. In particular, whenever such events occur, the MICIE alerting system supports the CI Operators providing them with a real time risk level.

The work done during this thesis resulted in the following publications:

- Caldeira, F., Castrucci, M., Aubigny, M., Aubert, J., Macone, D., Monteiro, E., Rente, F., Simões, P., and Suraci, V. (2010a). Secure mediation gateway architecture enabling the communication among critical infrastructures. In Cunningham, P. and Cunningham, M., editors, *Proceedings of the Future Network and MobileSummit 2010 Conference*, Florence, Italy, 16–18 June 2010. IIMC International Information Management Corporation.

- Caldeira, F., Monteiro, E., and Simões, P. (2010b). Trust and Reputation for Information Exchange in Critical Infrastructures. In Xenakis, C. and Wolthusen, S., editors, *Proceedings of the 5th International Workshop on Critical Information Infrastructures Security (CRITIS 2010)*, volume 6712 of *Lecture Notes in Computer Science*, pages 140–152. Springer Berlin Heidelberg, Athens, Greece, 23–24 September 2010.

- Caldeira, F., Monteiro, E., and Simões, P. (2010c). Trust and reputation management for critical infrastructure protection. In Jahankhani, H. and Tenreiro de Magalhães, S., editors, *Special Issue on Global Security, Safety and Sustainability*, volume 3(3) of *International Journal of Electronic Security and Digital Forensics*, pages 187–203. Inderscience Publishers.

- Caldeira, F., Monteiro, E., and Simões, P. (2010d). Trust and reputation management for critical infrastructure protection. In Tenreiro de Magalhães, S., Jahankhani, H., and Hessami, A. G., editors, *Proceedings of the 6th International Conference on Global Security, Safety, and Sustainability (ICGS3 2010)*, volume 92 of *Communications in Computer and Information Science*, pages 39–47. Springer Berlin Heidelberg, Braga, Portugal, 1–3 September 2010.

- Caldeira, F., Schaberreiter, T., Monteiro, E., Aubert, J., Simões, P., and Khadraoui, D. (2011). Trust based interdependency weighting for on-line risk monitoring in interdependent critical infrastructures. In Cuppens, F., Foley, S., Groza, B., and Minea, M., editors, *Proceedings of the Sixth International Conference on Risks and Security of Internet and Systems (CRiSIS 2011)*, pages 1–7, Timisoara, Romania, September 26-28, 2011. IEEE Computer Society.

- Caldeira, F., Schaberreiter, T., Varrette, S., Monteiro, E., Simões, P., Bouvry, P., and Khadraoui, D. (2013). Trust based interdependency weighting for on-line risk monitoring in interdependent critical infrastructures. In Khan, K. M., editor, *International Journal of Secure Software Engineering (IJSSE)*, volume 4(4). IGI Global.

- Capodieci, P., Diblasi, S., Ciancamerla, E., Minichino, M., Foglietta, C., Lefevre, D., Oliva, G., Panzieri, S., Setola, R., De Porcellinis, S., Priscoli, F., Castrucci, M., Suraci, V., Lev, L., Shneck, Y., Khadraoui, D., Aubert, J., Iassinovski, S., Jiang, J., Simões, P., Caldeira, F., Spronska, A., Harpes, C., and Aubigny, M. (2010). Improving Resilience of Interdependent Critical Infrastructures via an On-Line Alerting System. In Rizzo, A., editor, *Proceedings of the first International Conference COMPENG 2010, "Complexity in Engineering"*, pages 88–90, Rome, Italy, 22–24 February 2010. IEEE Computer Society.

- Castrucci, M., Neri, A., Caldeira, F., Aubert, J., Khadraoui, D., Aubigny, M., Harpes, C., Simões, P., Suraci, V., and Capodieci, P. (2012). Design and implementation of a mediation system enabling secure communication among critical infrastructures. In Shenoi, S., editor, *International Journal of*

Critical Infrastructure Protection, volume 5(2), pages 86–97. Elsevier Science Publishers.

- Schaberreiter, T., Caldeira, F., Aubert, J., Monteiro, E., Khadraoui, D., and Simões, P. (2011b). Assurance and trust indicators to evaluate accuracy of on-line risk in critical infrastructures. In Bologna, S. and Wolthusen, S., editors, *Proceedings of the 6th International Workshop on Critical Information Infrastructures Security (CRITIS 2011)*, Lecture Notes in Computer Science, Lucerne, Switzerland, 8–9 September 2011. Springer Berlin Heidelberg.

In the scope of MICIE project, the following Project Deliverables were co-authored by the candidate:

- Bertoni, A., Ciancamerla, E., di Prospero, F., Lefevre, D., Minichino, M., Lev, L., Iassinovski, S., Foglietta, C., Oliva, G., Panzieri, S., di Giorgio, A., Liberati, F., Aubert, J., Caldeira, F., Simões, P., Harpes, C., and Pauplin, O. (2010a). *Interdependency modelling framework, indicators and models – Final Report.* Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.2.3. European Commission FP7.

- Bertoni, A., Ciancamerla, E., Foglietta, C., Lefevre, D., Minichino, M., Cohen, A., Lev, L., Hunovich, T., Ohana, R., Holzer, R., Tanenbaum, D., Adar, A., Iassinovski, S., Oliva, G., Panzieri, S., Castrucci, M., Priscoli, F., di Giorgio, A., Liberati, F., Aubert, J., Incoul, C., Caldeira, F., and Simões, P. (2010b). *Interdependency modelling framework, indicators and models – Second Interim Report.* Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.2.2. European Commission FP7.

- Castrucci, M., Macone, D., Harpes, C., Pascoli, Aubigny, M., Aubert, J., Incoul, C., Gateau, B., Khadraoui, D., Panzieri, S., Oliva, G., Silvestri, Caldeira, F., Rente, F., Simões, P., Lev, L., and Tanenbaum, D. (2009). *MICIE ICT System Requirements – Preliminary Version.* Castrucci, M., editor. MICIE Project Deliverable D4.1.1. European Commission FP7.

- Castrucci, M., Macone, D., Harpes, C., Pascoli, Aubigny, M., Aubert, J., Incoul, C., Khadraoui, D., Panzieri, S., Oliva, G., Caldeira, F., Rente, F., Simões, P., Lev, L., Tanenbaum, D., Minichino, M., and Ciancamerla, E. (2010a). *MICIE ICT System Requirements – Final Version.* Castrucci, M., editor. MICIE Project Deliverable D4.1.2. European Commission FP7.

- Castrucci, M., Macone, D., Suraci, V., Inzerilli, T., Neri, A., Panzieri, S., Foglietta, C., Oliva, G., Aubert, J., Incoul, C., Caldeira, F., Aubigny, M., Harpes, C., and Kloda (2010b). *Secure Mediation Gateway Architecture – Final Version.* Castrucci, M., editor. MICIE Project Deliverable D4.2.2. European Commission FP7.

- Ciancamerla, E., di Blasi, S., Fioriti, V., Foglietta, C., Minichino, M., Lefevre, D., Cohen, A., Lev, L., Hunovich, T., Ohana, R., Holzer, R., Tanenbaum, D., Adar, A., Iassinovski, S., Menezes, N., de Porcellinis, S., Oliva, G., Panzieri, S., di Giorgio, A., Liberati, F., Aubert, J., Incoul, C., Caldeira, F., Rente, F., and Jiang, J. (2009). *Interdependency modelling framework, interdependency indicators and models – First Interim Report.* Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.2.1. European Commission FP7.

- Inzerilli, T., Castrucci, M., Macone, D., Suraci, V., Aubert, J., Incoul, C., Caldeira, F., Rente, F., Simões, P., Aubigny, M., Pascoli, Harpes, C., Oliva, G., Kloda, and Szewczyk (2009). *Secure Mediation Gateway Architecture – Preliminary Version.* Inzerilli, T. and Castrucci, M., editors. MICIE Project Deliverable D4.2.1. European Commission FP7.

- Lev, L., Hunovich, T., Ohana, R., Holzer, R., Tanenbaum, D., Adar, A., Ciancamerla, E., di Blasi, S., Fioriti, V., Foglietta, C., Minichino, M., de Porcellinis, S., Panzieri, S., Menezes, N., Caldeira, F., Rente, F., Simões, P., Castrucci, M., di Giorgio, A., Aubert, J., Incoul, C., Gateau, B., and Khadraoui, D. (2009). *Reference Scenario and service oriented approach – Interim Report.* Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.1.1. European Commission FP7.

- Lev, L., Tanenbaum, D., Ohana, R., Holzer, R., Hunovich, T., Adar, A., Cohen, A., Capodieci, P., Minichino, M., Ciancamerla, E., Foglietta, C., Oliva, G., Simões, P., Caldeira, F., Castrucci, M., Bojar, K., Jager, Pascoli, Aubigny, M., and Harpes, C. (2011). *Validation Activities.* Lev, L. and Baruch, Y., editors. MICIE Project Deliverable D6.3. European Commission FP7.

- Lev, L., Tanenbaum, D., Ohana, R., Holzer, R., Hunovich, T., Adar, A., Cohen, A., Capodieci, P., Neri, A., Minichino, M., Panzieri, S., Caldeira, F., Simões, P., Castrucci, M., Kloda, R., and Szewczyk, R. (2010b). *Integration Process Report.* Lev, L. and Adar, A., editors. MICIE Project Deliverable D6.2. European Commission FP7.

- Neri, A. (2010). *Secure Mediation Gateway SW Beta Release.* Neri, A., editor. MICIE Project Deliverable D5.2. European Commission FP7.

- Panzieri, S., Oliva, G., Foglietta, C., Minichino, M., Ciancamerla, E., Macone, D., Castrucci, M., Suraci, V., Pauplin, O., Aubert, J., Caldeira, F., Simões, P., and Curado, M. (2010). *Common Ontology and Risk Prediction Algorithms – Final Version.* Panzieri, S., editor. MICIE Project Deliverable D3.2.2. European Commission FP7.

Abstract

Currently, our society has at its disposal an uncountable number of services able to support the global economy and also our current way of life. Services such as power distribution, water, gas, transport networks, telecommunications, the Internet, among others, are now an integral part of the citizens' lives and businesses. These services play such a big role in our lives that their importance is only appreciated when they are unavailable. These types of services, that our lives so heavily depend on, are provided by Critical Infrastructures. They are referred to as "Critical" due to the fact that in case of failure or breakdown in providing quality of service, the impact on society and the economy of a country can be enormous. Beyond the phenomena of nature and risks inherent to the infrastructure operation, the risks faced by these infrastructures have continuously increasing, by attracting interest from groups of hackers and terrorist groups. Primarily due to the strong visibility and consequences that may result even from a small successful attack.

Among the problems inherent to the operation of Critical Infrastructures, it is possible to emphasise the existence of dependencies and interdependencies among infrastructures. For example, a telecommunications service is inherently dependent on the electricity supply or, for instance, banking services are dependent on both telecommunications and energy supply services. However, is it not the service that provides power supply actually dependent on telecommunications services and also on information systems? Based on these examples it becomes apparent that in addition to the (inter)dependence that may exist, it is also necessary to examine the cascading effects that may arise after the failure of a Critical Infrastructure.

Critical Infrastructures security has been the subject of discussion by numerous governments with the support of the academia by promoting research efforts in these areas, in particular in areas such as power distribution and telecommunications. Furthermore, within the European Union, there is determination to promote projects in these areas, in particular the promotion of projects that foster the exchange of

information, in the form of warnings, among infrastructures. These warnings allow the Critical Infrastructure to be informed and aware of the increasing risk of loss or reduction in quality of the service received. This exchange allows the infrastructure to timely implement their contingency and recovery plans to minimise any service breaks and consequently minimise the unwanted effect of a cascading failure. The motivation for the work presented in this thesis arose from the identification of the main open issues relating to the exchange and management of risk warnings among Critical Infrastructures.

Many of the existing approaches to security in Critical Infrastructures are focused on obtaining risk levels through the use of models based on the infrastructure. Although these models allow a solid foundation for risk monitoring, they do not have mechanisms for exchange, management and assessment of its quality. This work addresses the problem related to trust, reputation and risk alerts management within Critical Infrastructures. Accordingly, it is proposed to introduce mechanisms to manage and measure at each instant, the degree of confidence assigned to each of the alerts received or computed internally. Allowing improvement of their accuracy and consequently improving the resilience of Critical Infrastructures when faced with inaccurate or inconsistent risk alerts.

This thesis addresses the problem of interdependent Critical Infrastructure security and identifies the main problems related to risk information sharing. In particular, how to allow information sharing in a secure manner, the management of that sharing and how to assess the reliability of such information.

This thesis proposes the application of Policy Based Management mechanisms for the management of the risk alert information shared among Critical Infrastructures. In order to improve the information sharing management and the further interpretation of the risk alerts, it is proposed to evaluate Trust and Reputation in order to assess the shared information and also to consider the behaviour of the entities involved.

The proposals presented in this thesis are discussed and applied in the context of the European Project MICIE (Tool for systemic risk analysis and secure mediation of data exchanged across linked CI information infrastructures). In particular with regard to the proposed solution for the management of shared risk alerts, which uses the Policy Based Management paradigm. By incorporating the proposed Trust and Reputation indicators it allows to improve the Critical Infrastructure protection considering the use of untrustworthy or inconsistent information. It is also proposed the adaptation of the presented concepts to the *CI Security Model*, a model for real

time risk analysis evaluation, in which the identified shortcomings are addressed with the integration of the Trust and Reputation approach proposed in this thesis. The results of the proposals evaluation are discussed based on simulation scenarios as well as through real data of a Critical Infrastructure.

The achieved results indicate that the proposed mechanisms meet the objectives such as, by contributing to the increase in confidence that a Critical Infrastructure has on the information received about the services on which it depends. To allow improvement in management of such information as well as contribution to increased reliability of results obtained from the risk models applied to the infrastructure.

Resumo

Atualmente a sociedade contemporânea tem ao seu dispor um sem numero de serviços que suportam toda a economia globalizada em que vivemos bem como o nosso modo de vida. Serviços como distribuição de energia, água, gás, redes de transportes, telecomunicações, a Internet, entre outros, são atualmente parte integrante da vida dos cidadãos e das empresas. Estes serviços estão de tal forma presentes nas nossas vidas que a sua relevância e o grau de dependência aos serviços, apenas é sentido aquando da sua indisponibilidade. Este tipo de serviço dos quais depende o nosso modo de vida, são fornecidos por infraestruturas críticas, assim referidas pois a sua falha ou quebra da qualidade do serviço prestado pode ter um grande impacto na sociedade ou economia de um País. Para além dos fenómenos da natureza e dos riscos inerentes à sua própria exploração, os riscos que estas infraestruturas correm têm vindo a aumentar ao atrair cada vez mais o interesse de grupos de *hackers* e terroristas, principalmente pela forte visibilidade e consequências que mesmo um pequeno ataque pode acarretar.

De entre os problemas inerentes ao funcionamento das infraestruturas críticas destaca-se o fato da existência de dependências ou interdependências entre infraestruturas. Veja-se o exemplo do serviço de telecomunicações que está por natureza dependente do fornecimento de energia elétrica ou dos serviços bancários que estão dependentes de ambos. Mas não está atualmente o fornecimento de energia dependente dos serviços de telecomunicações e dos seus sistemas de informação? Destes exemplos torna-se visível que, para além da (inter)dependência que possa existir, é necessário analisar também os efeitos em cascata que podem surgir após a falha de uma infraestrutura.

Com o objetivo de promover a segurança em infraestruturas críticas, vários governos, em conjunto com a comunidade científica, promovem esforços de investigação nesta área. Em particular, nas áreas da distribuição de energia e das telecomunicações. Ao nível da União Europeia, existe grande determinação para promover

projetos nesta área, em particular, projetos que promovem a troca de informação entre infraestruturas, na forma de alertas de risco, prevenindo os Operadores das infraestruturas relativamente a um aumento de risco de perda ou quebra de qualidade do serviço fornecido. Esta troca permite que as infraestruturas possam aplicar atempadamente os seus planos de contingência ou recuperação, minimizando eventuais quebras de serviço e consequentemente reduzindo o indesejado efeito de falha em cascata. A motivação para o trabalho apresentado nesta tese, surgiu da identificação dos principais aspectos em aberto relativos à troca e gestão de alertas de risco entre infraestruturas críticas.

Muitas das abordagens existentes relativas à segurança em infraestruturas críticas focam-se na obtenção de níveis de risco através do uso de modelos mais ou menos complexos das infraestruturas. Apesar de estes modelos permitirem uma base sólida para a monitorização do risco, não apresentam mecanismos para a sua troca, gestão e avaliação de qualidade. Este trabalho aborda o problema relacionado com a confiança, reputação e gestão de alertas de risco no seio das infraestruturas críticas. Nesse sentido é proposta a introdução de mecanismos que permitam gerir e aferir em cada instante, o grau de confiança atribuído a cada um dos alertas de risco recebidos ou calculados internamente, permitindo melhorar a sua precisão e consequentemente melhorar também a resiliência da infraestrutura critica quando confrontada com alertas de riscos imprecisos ou inconsistentes.

Na tese é abordado o problema da segurança em infraestruturas críticas interdependentes e identificados os principais problemas inerentes à troca de informação de risco, em particular, a forma de efetuar a partilha de informação de uma forma segura, a gestão dessa mesma partilha e a avaliação da fiabilidade da informação envolvida na partilha.

Propõe-se nesta tese, a aplicação de mecanismos de gestão baseados no paradigma de gestão por politicas para a gestão da partilha de alertas de risco entre infraestruturas críticas. Com o objetivo de melhorar a gestão da partilha e posterior interpretação dos alertas de risco, é proposta a introdução da análise de confiança e reputação na avaliação da fiabilidade da informação envolvida na partilha e na avaliação do comportamento das entidades envolvidas.

As propostas apresentadas nesta tese são discutidas e aplicadas no âmbito do projeto Europeu MICIE (Tool for systemic risk analysis and secure mediation of data exchanged across linked CI information infrastructures), em particular, no que se refere à solução proposta para a gestão da partilha de alertas de risco, que em con-

junto com os indicadores de confiança e reputação propostos, permitem melhorar a proteção de cada infraestrutura relativamente ao uso de informação menos confiável ou inconsistente. Apresenta-se também a adaptação dos conceitos propostos ao *CI Security Model*, um modelo de análise de risco em tempo real, no qual as falhas identificadas são atenuadas com a introdução da análise de confiança e reputação proposta nesta tese. Os resultados da avaliação das propostas apresentadas são discutidos com base em cenários de simulação bem como através de dados reais de uma infraestrutura crítica.

Os resultados obtidos indicam que as propostas apresentadas satisfazem os objectivos definidos, nomeadamente, ao contribuir para o aumento da confiança que uma infraestrutura crítica tem relativamente à informação recebida em tempo real acerca dos serviços dos quais depende, ao permitir uma melhor gestão dessa mesma informação e também ao contribuir para o aumento da fiabilidade dos resultados provenientes dos modelos de risco em uso na infraestrutura.

Table of Contents

List of Figures

List of Tables

List of Acronyms

AAA Authentication, Authorisation and Accounting

AlertParMO Alert Parameters Managed Object

API Application Programming Interface

BGP Border Gateway Protocol

BN Bayesian Networks

CAS Complex Adaptive System

CI Critical Infrastructure

CIP Critical Infrastructure Protection

CIWIN Critical Infrastructure Warning Information Network

COPS Common Open Policy Service

CPT Conditional Probability Table

CRUTIAL CRitical UTility InfrastructurAL Resilience

DAML Darpa Agent Markup Language

DCS Distributed Control System

DMTF Distributed Management Task Force

ECI European Critical Infrastructures

EPCIP European Program for the Protection of Critical Infrastructure

FISR Fault Isolation and System Restoration Service

GUI Graphical User Interface

ICS Industrial Control System

ICT-MC ICT Monitoring Component

ICT Information and Communication Technology

IDS Intrusion Detection System

IEC Israel Electric Corporation

IETF Internet Engineering Task Force

INSPIRE INcreasing Security and Protection through Infrastructure REsilience

IRRIIS Integrated Risk Reduction of Information-based Infrastructure Systems

LAN Local Area Network

LDAP Lightweight Directory Access Protocol

MANET Mobile Ad-hoc Network

MHR Mixed Holistic Reductionist

MICIE Tool for systemic risk analysis and secure mediation of data exchanged across linked CI information infrastructures

MIT Middleware Improved Technology

NET-SCIP Innovation Network on Security and Critical Infrastructure Protection

OIL Ontology Interface Layer

OrBAC Organization-Based Access Control

OWL Web Ontology Language

PBMS Policy Based Management System

PBNM Policy Based Network Management

PDP Policy Decision Point

PEP Policy Enforcement Point

PLC Programmable Logic Controller

PT Prediction Tool

QoS Quality-of-Service

RDF Resource Description Framework

RSMGW Remote SMGW

RsmgwMO Remote SMGW Managed Object

RTU Remote Terminal Unit

SCADA Supervisory Control and Data Acquisition

SimCIP Simulation for Critical Infrastructure Protection

SLA Service Level Agreement

SLS Service Level Specification

SMC Self Managed Cell

SMGW Secure Mediation Gateway

SNMP Simple Network Management Protocol

STB Simulation Test Bench

TISN Trusted Information Sharing Network

TPM Trusted Platform Module

TRS Trust and Reputation System

URI Unique Resource Identifier

VANET Vehicular Ad-hoc Network

W3C World Wide Web Consortium

WAN Wide Area Network

WSN Wireless Sensor Network

WWW World Wide Web

XACML Extensible Access Control Markup Language

XML eXtensible Markup Language

Chapter 1

Introduction

This thesis aims to address problems that might arise within Critical Infrastructures, while evaluating real time risk levels that will allow to visualise the probability of a future service loss or service degradation. This information is intended to be shared and used among multiple (inter)dependent Critical Infrastructures (CIs) in order to reduce the possibility of service failure on dependent services thus minimising the expansion of cascading effects that might take place. In this first chapter the motivation behind this work is presented in Section 1.1 and the objectives and contributions of the work are described in Section 1.2. Section 1.3 concludes the chapter by presenting the structure of this thesis.

1.1 Motivation

Critical Infrastructures provide services that support our society and economy. Telecommunication infrastructures allow us to communicate with people and businesses at local or remote locations. Transport and air traffic infrastructures allow us to travel for leisure or business activities and make the global commerce flow. Internet access is widely available even in places where we think it would be impossible to be connected to the rest of the world. The electricity infrastructure enables a variety of services and applications that we take for granted. Can we take it for granted?

Natural disasters as, for example, hurricane Katrina in 2005, the earthquake followed by the tsunami that affected Fukushima nuclear reactor in Japan that took place in March 2011, and more recently, in 2012, the hurricane Sandy show us that some essential services can become unavailable causing chaos and difficulties for citizens and the economy. Those examples show that Critical Infrastructures are one of the most important technical or industrial systems that have a strong impact on peoples' lives and the operation of economy worldwide. Those types of infrastructures, such as facilities/utilities, provide services that are essential to the actual society as the services they provide are usually basic inputs to other simple or complex systems. This dependency on services provided by CIs can, in case of an improper operation of the CI, lead to the disruption of other dependent services.

Nowadays, the media is paying special attention to this type of Infrastructures while the risk of "traditional" or cyberterrorism attacks increases. Citizens are also becoming aware and concerned about those risks due to, e.g., to a recent television series, '24 – season seven', where the fictional character Jack Bauer fights a terrorist group intending to destroy some Critical Infrastructures in the United States of America. Putting the fiction aside, this TV series, one of the first of this kind, clearly helped to demonstrate how important those infrastructures are and how weak they can be.

Governments from many countries around the world are already aware of the importance of their Critical Infrastructures not only for the well-being of their Citizens but also for the survivability of their nations in terms of economy and defence.

The relevance of the area was first defined by the Administration of the United States of America as,

> *"Critical infrastructures are those physical and cyber-based systems essential to the minimum operations of the economy and government. These systems are so vital, that their incapacity or destruction would have a debilitating impact on the defence or economic security" (Clinton, 1996).*

More recently, on the 13th of February 2013, the President of the United States of America, Barack Obama, issued the Executive Order 13636 "Improving Critical Infrastructure Cybersecurity" (Obama, 2013). In this document, the most relevant policy is highlighted in the first section as:

> *"Repeated cyber intrusions into critical infrastructure demonstrate the need for improved cybersecurity. The cyber threat to critical infrastructure continues to grow and represents one of the most serious national security challenges we must confront. The national and economic security of the United States depends on the reliable functioning of the*

> *Nation's critical infrastructure in the face of such threats. It is the policy of the United States to enhance the security and resilience of the Nation's critical infrastructure and to maintain a cyber environment that encourages efficiency, innovation, and economic prosperity while promoting safety, security, business confidentiality, privacy, and civil liberties. We can achieve these goals through a partnership with the Owners and Operators of critical infrastructure to improve cybersecurity information sharing and collaboratively develop and implement risk-based standards."* (Obama, 2013)

One important aspect of President Obama's Executive Order on "Improving Critical Infrastructure Cybersecurity" (Obama, 2013) is identified in Section 4 of that document. Where it is proposed that the Policy of the United States Government should help improve the cyber threat information sharing among private sector entities that control CIs, so that those entities can improve the weapons available in the fight against cyber threats.

Legislation or encouraging policies that aim to improve the information sharing among Critical Infrastructure owners has already been issued in other regions such as the European Union and Australia. In the United States of America, these type of policies have special relevance as most of the CIs are privately owned. Apart from this "novelty", the Executive Order (Obama, 2013) proposes the development of a "Baseline Framework to Reduce Cyber Risk to Critical Infrastructure" that can help to improve, for example, the regulations ISO 27001 (widely used in industry) (ISO/IEC, 2005), 800-53 (used within the USA government) (NIST, 2009) or NERC CIP (used in energy sectors) (NERC, 2009). An aspect that can call a drawback is stated in Section 8 of the Executive Order (Obama, 2013), defining that the adoption of the framework can be done on a voluntary basis. For instance, the information sharing among CIs has higher relevance with the increase in the number of CIs that are willing to exchange threats and risks.

Although, the U.S.A. Executive Order 13636 (Obama, 2013) can be seen as a step to improve the Critical Infrastructure Protection area, it is the author's opinion that the foreseen deadlines for implementation and the voluntary adoption of the framework, can quickly transform this action in just one more attempt to achieve regulation among heterogeneous and private managed CIs. Apart from this opinion, those initiatives are needed and encourage information sharing initiatives among Critical Infrastructures.

Australian Federal Government also considers Critical Infrastructures as essential for contemporary society existence and defines it as *“...those physical facilities, supply chains, information technologies and communication networks which, if destroyed, degraded or rendered unavailable for an extended period, would significantly impact on the social or economic well-being of the nation, or affect Australia’s ability to conduct national defence and ensure national security.”* McClelland (2010).

As presented by McClelland (McClelland, 2010), former Attorney-General of Australia (from 2007 to 2011), the Australian Government has created a National strategy in order to enhance the security and resilience of the Critical Infrastructures in the Country. Critical Infrastructures relevance to modern society can also be highlighted quoting McClelland where he highlightes possible risks that can damage CIs and then states *“... Those risks - from natural disasters, to equipment failure and crime - can damage or destroy critical infrastructure as well as disrupt the essential services that are provided by these assets, networks and supply chains.”*. The same document also states similar concerns as other Nations, for instance he refers to *“ ... Such an incident could significantly affect all Australians because of our reliance on critical infrastructure, which is of major importance to businesses, governments and communities. A resilience based approach to critical infrastructure is vital so we can better adapt to change, reduce our exposure to risk and learn from incidents when they occur. The responsibility for the continuity of critical infrastructure is shared by all governments and by owners and operators.”*. In the same document, McClelland continues to emphasize CIs relevance in Australia and the Government’s support to initiatives aiming to improve the security and resilience of Australians CIs.

Australia has created an agency named Trusted Information Sharing Network (TISN) for Critical Infrastructure Resilience aiming to provide an environment *“where business and government can share vital information on security issues relevant to the protection of our critical infrastructure and the continuity of essential services in the face of all hazards”* (TISN, 2011). This agency aims to bring together CI Owners and Operators from multiple sectors including also two Expert Advisory Groups able to provide advice on aspects of Critical Infrastructures requiring expert knowledge.

The European Commission is also committed to enhancing security on Critical Infrastructures. The Directorate-General of the European Commission in charge of the policy area known as “Home Affairs” define CIs as - *“Critical infrastructure is an essential asset for the maintenance of vital societal functions. Damage to the critical infrastructure, its destruction or disruption by natural disasters, terrorism,* *criminal activity or malicious behaviour, may have negative consequences for the security of the EU and the well-being of its citizens"* (European Commission, 2012). This Directorate-general also states that *"Reducing the vulnerabilities of critical infrastructure is one of the major objectives of the EU. An adequate level of protection must be ensured and the detrimental effects of disruptions on the society and citizens must be limited as far as possible."* (European Commission, 2012).

The European Council that took place on the 17th and 18th of June 2004, asked the European Commission to prepare a strategy in order to improve the protection of Critical Infrastructures. In response, on the 20th of October 2004, the Commission published the communication "Critical Infrastructure protection in the fight against terrorism" (European Commission, 2004).

In the scope of the European Program for the Protection of Critical Infrastructure (EPCIP), eleven sectors were identified as CIs in the publication of a green paper (European Commission, 2005) and reinforced by the communication from the European Commission in the year 2006 (European Commission, 2006). Those sectors are further described in Section 2.1.

An important element of the EPCIP program is a Directive on European Critical Infrastructures (European Commission, 2008) as it establishes a procedure for identifying and designating European Critical Infrastructures (ECI) and a common approach to improve the protection of such infrastructures. Currently, the Directive's scope is limited to the energy and transport sectors. In EPCIP the initial responsibility for the protection of CI is a National responsibility, however, a distinction is made among National CIs and European CIs (an infrastructure that is considered critical for more than one Member State of the Union). The dependency that exists among Critical Infrastructures is also under the European Commission's attention.

Understanding the importance of the role, that the exchange of information about threats and vulnerabilities plays in protecting CIs, an information network has been created for that role - Critical Infrastructure Warning Information Network (CIWIN) (European Commission, 2008). CIWIN has two main objectives: the establishment of an electronic forum for the purpose of exchanging information on the protection of CIs; and the development of a rapid alert system for the delivery of early warnings for Member States to inform the Commission regarding risks and threats.

Beyond the measures adopted by the European Union, Portugal also demonstrates a strong interest in the field of Critical Infrastructures protection. One example of this interests is the development of the Innovation Network on Security and Critical Infrastructure Protection (NET-SCIP) by the Portuguese Fundation for Science and Technology under the Carnegie Mellon – Portugal Program. The main goal being to develop comparative advantages for Portugal in new security technologies and services for the protection of Critical Infrastructures (NET-SCIP, 2011). Sharing strong similarities with other International initiatives, NET-SCIP *"is currently gathering the scientific community, the private sector and the main government agencies with the goal of developing comparative advantages for Portugal in new security technologies and services for the protection of critical infrastructures."* (NET-SCIP, 2011).

Considering the above, it is clear that CIs are one of the most Information and Communication Technology (ICT) dependent areas of contemporary societies where we should ensure the highest security levels.

It is also important to mention that, with time, Critical Infrastructures have grown not only in importance but also in complexity. One key aspect of that growth is that CIs have now become dependent on each others outputs (interdependency). From the field of fault tolerant computing we know that a complex system that depends on multiple interacting components is exposed to a high risk of failure mainly due to the risk of failure of each individual and to some other side effects that a single failure can cause.

The efforts made in the Critical Infrastructure Protection in ICT area increased, especially after "Stuxnet" attack. "Stuxnet" is a computer worm and, as reported in 2010, it was the first malware specifically targeting control systems as the ones used in many existing Critical Infrastructures. In a similar way as viruses and worms,"Stuxnet" exploited some vulnerabilities that were unknown at the time of the attack in order to replicate and spread itself among the exploitable equipments. However, the main goal of this worm was to attack the industrial control systems, by introducing changes (not visible to the system operator) in the Programmable Logic Controllers (PLCs), modifying their normal behaviour, to make them work as the attacker intended (Falliere et al., 2011).

"Stuxnet" is harmless when it is spreading and when it is in the latent state in the infected equipment. At the time of the attack "Stuxnet" was not detectable by any anti-virus software since no one can hardly see abnormal behaviours on the infected equipment. The actual impact of the worm started when it targeted the control system and started to make changes in the PLCs. The media and some published work (Farwell and Rohozinski, 2011), (Miyachi et al., 2011), (Sheng et al., 2011) reported attacks in the Islamic Republic of Iran where it is said that more than sixty thousand computers were infected. It is also mentioned that this work damaged centrifuges used to enrich uranium at Iran's nuclear plant. It is also mentioned that the attack had affected the centrifuges causing them to switch back and forth between high and low speeds at intervals, for which the machines were not designed (Falliere et al., 2011). None of these events were confirmed by the Iranian government who only admitted to the presence of infection but if the media can be considered reliable, "Stuxnet" has managed to achieve a successful attack.

Attacks like "Stuxnet" are causing the need to different approaches to protect CIs as some have thought that those infrastructures were not vulnerable. Several works are being developed to improve CIs security.

The work described in this thesis originated during the development of the MICIE FP7 project (MICIE Consortium, 2008). This project aimed at the development of an alerting system that identifies, in real time, the risk level induced on a given Critical Infrastructure caused by undesired events or malicious attacks happening in the reference CI or in other dependent CIs. The author of this thesis worked actively on several work packages within the MICIE project and in particular, in the analysis and development of the MICIE Secure Mediation Gateway (SMGW). This component of the MICIE architecture is mainly responsible for the information exchange among CIs, and among other aspects, for the prevention of the occurrence of cascading effects.

The MICIE project was able to gather risk information from CIs and also adequately integrate risk information received from dependent CIs. As an improvement to the SMGW, the author has proposed the integration of a Policy Based Management System (PBMS) allowing the SMGW Operator to manage the system in a simple way by the use of policies. Those policies, apart from the common conditions used in access control systems (IP Address, user, date, time, etc.) that are able to help take system management decisions, are also able to use risk information in order to improve those same decisions. With this approach it is possible to use risk information allowing system management. From this point it started to become clear, that in addition to the risk exchange information from dependent CIs and integration of this information in the risk evaluation, it should also be possible to infer on the exchanged information in order to try to minimise the use of incorrect information.

As it was not foreseen by the MICIE system, to infer on the exchanged information, this gap was identified and a framework for Trust and Reputation Management has been proposed. This framework, not planned at the beginning of the project, has been implemented and tested as an add-on to the project - the Trust and Reputation System (TRS).

Since the results of the application of the Trust and Reputation System on the MICIE project were promising, an independent architecture has been defined and validated by using also different CI risk models and risk evaluation tools.

The scope of the work of this thesis was motivated by the identification of the main issues concerning risk gathering and risk information exchanged among Critical Infrastructures and its applicability to improve interdependency models that are able to help protecting those Infrastructures.

1.2 Objectives and Contributions

The main objective of this thesis is to contribute to the improvement of Critical Infrastructures security by addressing, in an integrated manner, problems that result from a scenario were it is planned to evaluate risk levels of Critical Infrastructure and share those risk levels among multiple (inter)dependent CIs in order to help them in their own risk evaluation. The main contributions of this thesis can be summarised as the following:

Analysis of problems resulting from CI risk exchange

The first contribution was a Policy Management Architecture to manage the MICIE Secure Mediation Gateway, responsible for implementing the risk exchange mechanisms. This contribution was integrated in the MICIE project architecture, namely, into the SMGW Manager for which it was developed a Policy Management Tool, integrated and delivered by the MICIE project. The developed tool is also able to integrate information gathered by the Trust and Reputation System (Caldeira et al., 2010a,c,d; Castrucci et al., 2010a,b, 2012; Ciancamerla et al., 2009; Inzerilli et al., 2009; Lev et al., 2009, 2011, 2010b; Neri, 2010; Panzieri et al., 2010).

Introduction of Trust and Reputation approaches in CI risk exchange

The second contribution was the conception and development of a Trust and Reputation framework able to infer trust information from the exchanged information in the MICIE system. The proposed framework is able to help the CI Operator to reason about the exchanged information and also to dynamically include the risk assessment in the defined management policies. This allows to improve the security of the existent MICIE Secure Mediation Gateway through, for instance, denying sending or receiving information from untrusted peers. A prototype was developed as an add-on to the MICIE project, allowing to be integrated within the Secure Mediation Gateway from where the necessary data for the evaluation is gathered. It is also able to receive stored data for simulation purposes (Bertoni et al., 2010a,b; Caldeira et al., 2010b,c,d; Castrucci et al., 2010b; Inzerilli et al., 2009; Lev et al., 2011, 2010b).

Integrate the trust model approach in CI risk models

The third contribution resulted from a joint work with Thomas Schaberreiter and consisted in the development of models aiming to build a trust relationship among CIs. This relationship is based on the common abstract information that CIs share, describing how trust can be used in the model to dynamically re-evaluate the impact a risk level received, from a dependency, has on the modelled risk in a CI.

Specifically, the trust model is now part of the CI Security Model (proposed by Thomas Schaberreiter) and can be used to reason on the exchanged information and internally in one CI to reason about the information gathered from the field.

A modelling tool able to model CIs as represented by the CI Security Model existent entities, as been developed. This tool allows evaluating risk and trust indicators in real time, with data coming from the CI or by receiving previously prepared data in order to simulate a specific scenario (Caldeira et al., 2011, 2013; Schaberreiter et al., 2011b).

The methodology used during this thesis was the study of the state of the art in the Critical Infrastructure security, CI modelling and Trust and Reputation System. Then, the Policy Based Management architecture and the Trust and Reputation framework were studied and developed taking into consideration the analysis of related work. The evaluation of the proposed schemes and the validation of the relevance of their contribution were done in part within the MICIE project (the Policy Based Management) and by simulation with data generated from simple statistical distributions and also with data gathered from a CI during approximately one year.

1.3 Structure of the Thesis

This thesis is organised in six chapters, including this one, according to the structure bellow:

Chapter 2 presents a contextualized characterisation of the current practice in CI Protection, identifying the main CIs characteristics, risk assessment and modelling methodologies applied to CIs. Three selected European Projects aiming to improve Critical Infrastructure Protection (CIP) are also presented. The use of ontologies and Policy Based Management are also discussed throughout the chapter in the context of Critical Infrastructures. Furthermore, the main aspects related to Trust and Reputation are presented, including the description and comparison of some existent Trust and Reputation frameworks.

Chapter 3 describes the MICIE project, including the MICIE Alerting System, the MICIE Secure Mediation Gateway, and the solutions to specifically incorporate CIs interdependencies in the online risk assessment framework. In this chapter the author's contributions within the MICIE project are highlighted.

Chapter 4 identifies the main issues related to the risk information exchange and describes the proposed approach to evaluate Trust and Reputation indicators. Simple validation scenarios are also shown in this chapter.

The integration of the Trust Model in the CI Security Model is presented in Chapter 5 by describing three application scenarios, presenting each of them, the used approach for the integration of both models, the validation results and discussing the obtained results.

Chapter 6 completes this thesis by presenting the conclusions drawn from this work, including also a summary of the main achieved results and contributions. As the experience gained from this work allowed to identify some open issues, they are presented in the chapter in order to be addressed in future work.

Chapter 2

Trust and Reputation Management for Critical Infrastructure Protection

The research problems associated with Critical Infrastructure security and protection are described in this chapter by first describing Critical Infrastructures, the problems that may arise from the existent CI (inter)dependencies among them and by describing how these problems are being currently addressed by the research community. This chapter briefly presents a review of some of the most representative research work done in Critical Infrastructure modelling, simulation techniques and risk assessment, discussing their main characteristics and limitations. Open issues and challenges associated with them will also be covered. Selected European Projects that deal with CI protection are described. This chapter also addresses issues related to Policy Based Management. An overview is made of some existent Trust and Reputation Models focusing on their applicability to the context of Critical Infrastructures protection and information exchange among Critical Infrastructures.

This chapter is structured as follows: Critical Infrastructures are defined and presented in Section 2.1 which also includes dependency and interdependency aspects, modelling techniques and CI risk assessment. Section 2.2 describes three representative European projects in the area of CI protection. The CI Security Model (proposed by Thomas Schaberreiter) is presented with more detail in Section 2.3 as this model has been used to validate one of the contributions proposed in this thesis. Sections 2.4 and 2.5 discuss, respectively, the use of ontologies and Policy Based Management tools in the context of Critical Infrastructures.

Section 2.6 discusses the main aspects related to Trust and Reputation including the description and comparison of some available Trust and Reputation frameworks. The chapter's last section presents a summary of the issues discussed throughout the chapter.

2.1 Critical Infrastructures

As already defined, CIs provide vital services for the normal functioning of a community or a country. As the academics become more aware of the impact that a disruption in those services can have, Critical Infrastructure Protection has become an important research topic in the last years.

In the previous chapter, some definitions of Critical Infrastructure were presented. One question that can still arise is "How can one know if an infrastructure is in fact critical?". Despite the fact that an answer to this question appears rather easy, there is no definite answer. For example, Moteff et al. reported to the U.S. Congress (Moteff et al., 2003) that they should not consider the definitions that were given over the years of what makes a Critical Infrastructure rigorous, while referring to the following definition - *"Critical Infrastructures were originally considered to be those whose prolonged disruptions could cause significant military and economic dislocation"*.

Moteff et al. also states that all definitions leave space for new interpretations regarding whether infrastructure can meet the definitions. Definitions and list of sectors considered critical should be considered as examples and not as exhaustive and closed lists. For instance, Moteff et al. in addition to the infrastructures whose failure can have impact to the national defence, economy, public health and safety, also refers assets that, if destroyed, can have impact to the national morale in a country (Moteff et al., 2003).

Although one can realise that there are no closed lists and rigid means of defining Critical Infrastructures, it is important to define and use some criteria as guidelines in order to classify infrastructures as critical.

Within the European Union, experts from multiple European Countries have defined sectors and organisations in order to classify them as critical. According to the European Commission, Critical Infrastructure encompasses the sectors and related sub sectors described in Table 2.1 (European Commission, 2005).

Table 2.1: Critical Sectors (European Commission, 2005)

Sector	**Sub-sector**
Energy	- Oil and gas production, refining, treatment, storage and distribution by pipelines - Electricity generation and transmission
Nuclear industry	- Production and storage/processing of nuclear substances
ICT	- Information system and network protection - Instrumentation automation and control systems (SCADA etc.) - Internet - Provision of fixed telecommunications - Provision of mobile telecommunications - Radio communication and navigation - Satellite communication - Broadcasting
Water	- Provision of drinking water - Control of water quality - Stemming and control of water quantity
Food	- Provision of food and safeguarding food safety and security
Health	- Medical and hospital care - Medicines, serums, vaccines and pharmaceuticals - Bio-laboratories and bio-agents
Financial	- Payment and securities clearing and settlement infrastructures and systems - Regulated markets
Transport	- Road transport - Rail transport - Air transport - Inland waterways transport - Ocean and short-sea shipping
Chemical industry	- Production and storage/processing of chemical substances - Pipelines of dangerous goods (chemical substances)
Space	- Space
Research facilities	- Research facilities

Among the different sectors of activity that encompasses Critical Infrastructures, the electricity facilities and telecommunications providers are the sectors that are getting more attention from researchers. Nowadays, the power grids are vital to the society causing major losses when they are not available. Power grids usually cover a wide geographical area producing energy from large central power generation stations to small home generation systems and deliver it to consumers via transmission grids.

Power grids are large complex systems composed by multiple components, such as *power substation* using transformers to convert electricity from high voltage to low voltage, *transmission networks* used to transport the energy from the generation plants to the substations, *distribution networks* that distribute energy from the substations to the consumers and *control systems* able to manage all the power grid components. Figure 2.1 represents a simple Power Grid CI, including some of the main components that usually are in use in those systems. This system encompasses a control room, transmission networks and control devices.

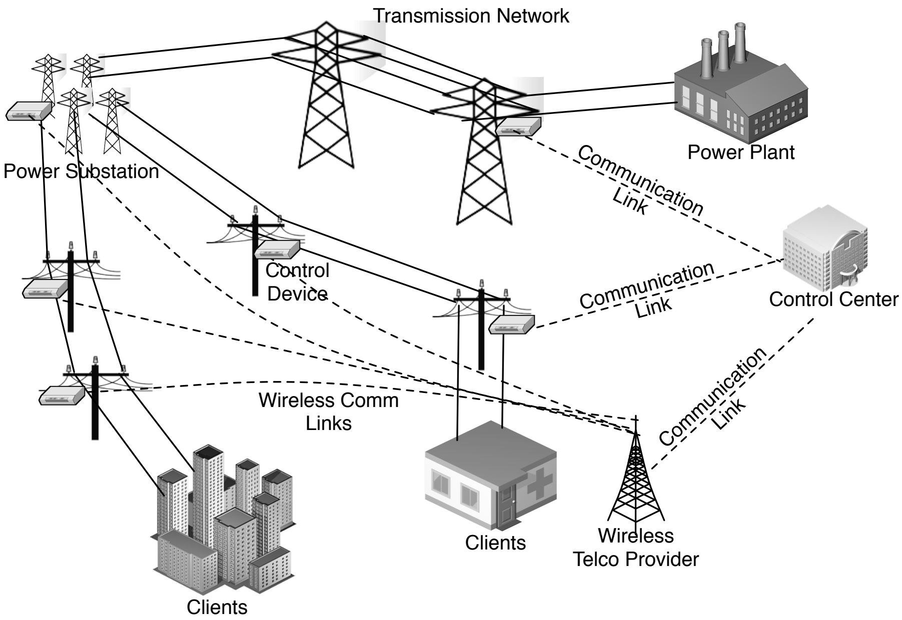


Figure 2.1: Power Grid example

Just like the other Critical Infrastructures, Power Grids are generally managed by Industrial Control Systems (ICSs) that usually include Supervisory Control and Data Acquisition (SCADA) systems, Distributed Control System (DCS), PLC, among other systems typically used in industrial contexts such as utilities (electricity distribution and production, water supply, etc.).

Using a simplified view, it is possible to say that a SCADA system comprises a set of Remote Terminal Units (RTUs) able to collect data from the field equipment and to send this data to a master station using an available communication system. The master station can display individual or aggregated data allowing an overview of the controlled system. Also, it is possible for the CI Operator to remotely control some equipment and perform control tasks. In such a system as SCADA, the data accuracy and is timely reception (real-time) allow the CI Operator to have more reliable information thus allowing a more efficient and safer operation.

From the above, it is clear that also telecommunication networks, apart from the fact that they provide vital services for final consumers and on which actual society depends for business or leisure, most of the actual CIs also depend on the services they provide. For instance, Internet access, mobile phone and data networks, private data links, among others, provide essential services that are vital for the ICSs objectives within a CI.

SCADA systems are frequently used to control dispersed assets using centralised data acquisition and supervisory control. In order to achieve their goals, these systems now rely, for instance, on proper communication availability, in the quality of the information gathered by the field instrumentation and control devices and also on the quality and reliability of the ICT systems that support those activities. For years, ICS and SCADA systems were generally isolated inside the CI ICT systems mostly relying on internal services thus ensuring security by obscurity and systems isolation.

The evolution of the Industrial Control Systems used in CIs is occurring by passing from proprietary and closed architectures to open standard based solutions. These new solutions are designed to simplify the interoperability with other platforms, systems and different devices. As usual while dealing with security, it is important to balance Confidentiality, Integrity and Availability. In this case, ICS systems are becoming more available within the CI and even among partner CIs. While improving availability it is important to focus also on maintaining ICS data confidential and reliable. Currently, ICSs are considered a critical and strategic asset that produces information about all the CI. The failure or just the existence of false or inadequate information in the ICSs has a huge potential for catastrophic consequences within the CI.

As ICS systems as SCADA move from an isolated environment to an open and interconnected environment, the identified vulnerabilities grow in number and also in the danger they represent to the CIs. For instance, Ryu et al. presents a list of vulnerabilities that can affect SCADA systems and suggests some ideas to minimise those vulnerabilities (Ryu et al., 2009). The vulnerability list contains among others, references to problems that may arise due to the multiplicity of vendors, the size and complexity of the networks, the equipment age, the use of widely available operating systems, the insider attacks, etc. Although it is not intended in this thesis to directly indorse this problem, it is important to create mechanisms that can help to verify the information quality inside such a potential vulnerable environment.

As already stated, CIs can suffer damages or could even potentially be destroyed due to numerous threats. Those threats include natural disasters, negligent comportments, terrorist acts, hacking, robbery and criminal behaviours, among others. From this small list it is clear that there are several threats hanging over this type of infrastructures. Problems that may arise from existing threats should be reduced. According to several documents from the European Union, for instance (European Commission, 2005, 2006, 2008) it is important that any Critical Infrastructure disruptions or manipulations should, within possibility, be brief, infrequent, manageable, geographically isolated and minimally harmful to the welfare of the affected country and their citizens.

Using the European Union as an example, the European Commission also highlights the fact that a damage or loss of critical services in one Member State may cause adverse effects on several others and on the European economy as a whole. This risk is gradually increasing due to the globalisation (at European or Worldwide level) supported by the introduction of communication technologies (e.g. the Internet) allowing that some of the infrastructures control systems to become part of a larger interconnected network. It needs to be emphasised that actual infrastructure sectors do not exist isolated but interact among each other as represented in Figure 2.2.

Such a situation means that CI protection must be addressed across sectors and across borders. Interdependencies between CIs imply that an impact (e.g. an undesired event) occurring on one infrastructure results also in an impact on one or more interdependent infrastructures (Figure 2.2 shows a typical CI interdependency tree).

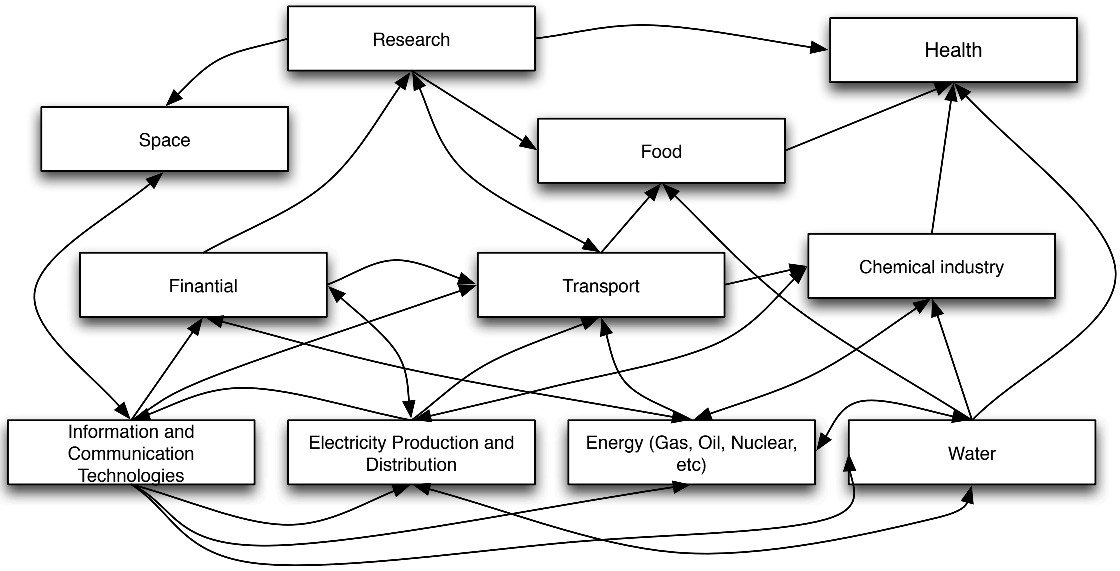

Research
Health
Space
Food
Chemical industry
Finantial
Transport
Information and Communication Technologies
Electricity Production and Distribution
Energy (Gas, Oil, Nuclear, etc)
Water

Figure 2.2: Critical Infrastructure interdependency example (Rinaldi et al., 2001)

2.1.1 Critical infrastructure (inter)dependency

One major study in the area of Critical Infrastructure (inter)dependency was carried by Rinaldi et al.. In this work (Rinaldi et al., 2001) a Critical Infrastructure is defined as an agent in a Complex Adaptive System (CAS) due to the complex and continuously changing set of interacting components. Rinaldi et al. intends to clarify and discuss the existing dependencies or interdependencies among CIs, in particular analysing those (inter)dependencies from six dimensions, namely, the infrastructure characteristics, the types of interdependencies, the environment where the CI operates, the state of operation, the type of possible existing failures and the coupling and response behaviour (Rinaldi et al., 2001).

According to Rinaldi et al. each CI can be either dependent or interdependent to other CI determined by whether it is a supporting or supported CI. Rinaldi et al. defines dependency and interdependency as follows:

- Interdependency: *"A bidirectional relationship between two infrastructures through which the state of each infrastructure influences or is correlated to the state of the other. More generally, two infrastructures are interdependent when each is dependent on the other."* (Rinaldi et al., 2001)

- Dependency: *"A linkage or connection between two infrastructures, through which the state of one infrastructure influences or is correlated to the state of the other."* (Rinaldi et al., 2001)

Rinaldi et al. also defined four categories allowing to group the existing different interdependency types (Rinaldi et al., 2001). The defined classes are: physical (the CI state is dependent on the physical output of another CI), cyber (the state of one infrastructure depends on information conveyed using an ICT infrastructure), geographic (when a local environmental change could origin a state change in one or more geographically interdependent infrastructures) and logical (dependency that is not physical, cyber or geographic, for instance human decisions that impact on one infrastructure can cause a state change in a dependent infrastructure).

According to Rinaldi et al., the current state of operation of an infrastructure has a direct impact on the state of operation of interdependent infrastructure. The states of operation can range from normal to stressed/disrupted and repair/restoration (Rinaldi et al., 2001). In this thesis, the cyber interdependency type will be addressed with the objective of minimising the effects that this type of dependency can have on the state change of an infrastructure.

An additional key characteristic of interdependencies is the failure that can occur in an infrastructure and how it may spread to other interdependent infrastructures. As CIs become more interconnected, it is possible to compare them to different components of a single network where a failure in a node of this complex network of interdependent infrastructures, can results in disastrous failures including some that are not yet foreseen. These type of failures result from the propagation of some small failure through other interdependent systems. This propagation effect is known as the *cascading effect* and has been the subject of multiple research works (Rinaldi, 2004).

The research that has been carried out for studying the structure and behaviour of Critical Infrastructures is frequently divided in two main distinct but interrelated groups. The first one is involved with the study and analysis of the infrastructures while the second focuses on the understanding of the CIs dynamic behaviour

Actual research in these areas aims to identify and to develop techniques and tools in order to describe the current status of an infrastructure. This is an important aspect, as the cascading effect can be minimised if one knows the actual or predicted state of an infrastructure on which one depends. In order to gather detailed information from CIs, researchers exploit multiple CI vulnerability, risks and threats. These processes aim to reveal the potential points of failure and to describe the expected consequences of each failure. One of the breakdowns of this approach, that is commonly cited, is that although the main points of failure can be detected, it is not easy to extrapolate all the possible consequences maintaining a situation where not all consequences are visibly perceived and understood.

Some publications address the problem of how to identify the various kinds of dependencies that can occur among CIs. Rinaldi et al. specifies an overview on the multiple dimensions in which (inter)dependencies can occur (Rinaldi et al., 2001). Rinaldi (Rinaldi, 2004), discusses and analyses different modelling techniques (discussed in Section 2.1.3) applied to CI (inter)dependencies. Another important publication, among others, that proposes CI models based on various different modelling techniques was presented by Sokolowski et al.. This publication (Sokolowski et al., 2008) describes a conceptual modeling method for CIs that aims to create an abstract and simplified view of a CI.

It is noticeable that most of the published work dealing with CI models and CI risk models is very heterogeneous with respect to their purpose and the extent to which they might be implemented. The majority of the models usually allow modelling to be too high level and consequently difficult to apply. In some cases, the proposed frameworks are focused on one specific CI or CI type and therefore it is difficult to use them in different CIs.

A significant work is also been carried out by Panzieri et al.. In this work (Panzieri et al., 2005), the Complex Adaptive System approach is used in order to allow CI modelling. In this work it is proposed to model interdependencies by studying a set of mutually dependent systems (agents), each of them able to represent a macro-component of a given infrastructure, in order to achieve the global CI model. In particular, this approach is focused on analysing performance degradation and fault propagation that can occur after one or more failures, not including recovery or repair procedures that may take place after one failure (Panzieri et al., 2005).

De Porcellinis et al. described the Mixed Holistic Reductionist (MHR) approach as a methodology able to model Critical Infrastructures including the existent interdependencies, considering the predefined level of Quality-of-Service that should be provided to customers or other dependent CIs (De Porcellinis et al., 2009). A reductionist approach aims to model systems into small parts while the holistic method looks globally at systems including the existing interactions. In the MHR perspective, the high level interdependencies are modelled following a holistic paradigm, while interdependencies between components are modelled with reductionistic techniques. This aspect is considered by De Porcellinis et al. as positive as it combines the advantages of both approaches (De Porcellinis et al., 2009).

One interesting aspect of the MHR approach is, as stated (De Porcellinis et al., 2009), that holistic blocks are able to represent the holistic perspective of the infrastructures, and are able to interact with other existing holistic blocks in order to inform of their status (De Porcellinis et al., 2009). For example (Simões et al., 2009), the failure block allows modelling social events (e.g. strike, panic) that are difficult to model on a more focused abstract level. Holistic blocks have the possibility to impact the operative conditions of a service, based on feedback received from reductionistic elements (Simões et al., 2009).

Also, major work, able to use CI models to infer risk information in CIs, was also proposed by some authors. For instance, Haslum and Arnes describe the use of continuous-time hidden Markov models for real-time risk calculation and estimation

(Haslum and Arnes, 2006). Baiardi et al. presents a risk management strategy based on a hyper-graph model in order to detect complex attacks while also supporting risk mitigation (Baiardi et al., 2009).

The recently described CI security model, presented by Schaberreiter et al. is, as stated by the author, based on a different approach that differs greatly from the models previously published (Schaberreiter et al., 2011a). It tries to establish abstract models of CIs that can be compared with each other, while maintaining generality by enabling it to be applied to all kinds of CI sectors. As some of the validation work carried out by the author of this thesis was based on this model, the presentation of this methodology is detailed in Section 2.3.

2.1.2 Assessing Risk in Critical Infrastructures

It is possible to find many definition for risk, for example, the Stanford Encyclopedia of Philosophy (Hansson, 2012) states that the word "risk" refers to situations in which it is possible for some undesirable event to occur. Apart from the fact that the word "risk" can have more specific uses and meanings in different contexts, the Encyclopedia defines five definitions stated to be widely used across disciplines, as the following:

- *risk = an unwanted event which may or may not occur.*
- *risk = the cause of an unwanted event which may or may not occur.*
- *risk = the probability of an unwanted event which may or may not occur.*
- *risk = the statistical expectation value of an unwanted event which may or may not occur. The expectation value of a possible negative event is the product of its probability and some measure of its severity.*
- *risk = the fact that a decision is made under conditions of known probabilities ("decision under risk" as opposed to "decision under uncertainty")*

Risk management is becoming more commonly used when dealing with ICT security. As the security threats increase, companies are adopting risk management strategies in order to better defend their assets. ICT infrastructures are becoming more complex thus more difficult to manage and secure. Risk management consists of a process aiming to identify a set of security measurements in order to achieve the required security level.

As stated by Adar and Wuchner, *"A risk can be defined as any event that may result in a missed business objective"* (Adar and Wuchner, 2005). When dealing with risk management, one relevant assumption is that not every existing risk can be prevented but prior knowledge about the possible risk helps the manager in taking informed decisions.

Being part of the company security program, risk management should be evaluated in multiple steps in order to identify and classify the existing risks (risk assessment). The results should then give origin to a set of balanced security measures able to reduce or minimise those risks. Figure 2.3 describes the risk management life cycle proposed by Adar and Wuchner.

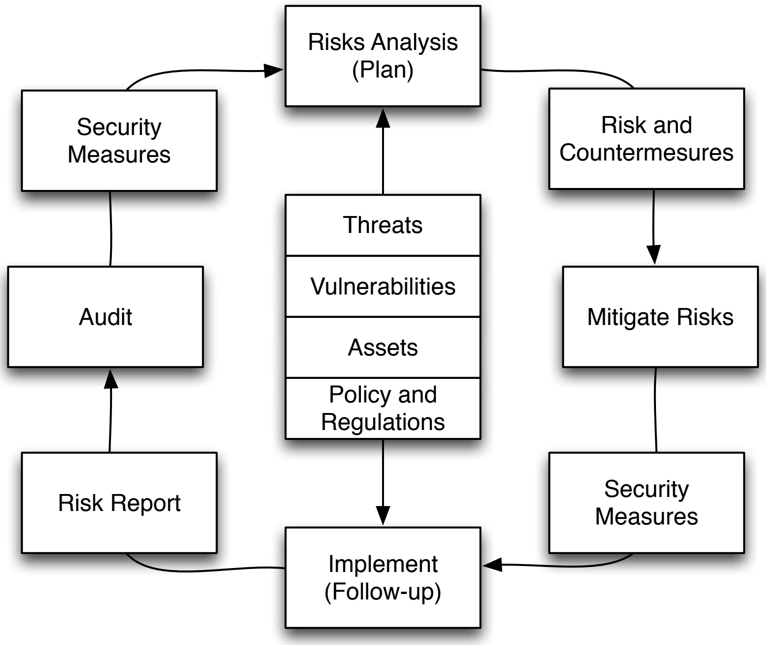


Figure 2.3: Risk management life cycle (Adar and Wuchner, 2005)

Risk management plays an important role in terms of compliance to National or International regulations and it is also relevant to justify the investment in ICT security allowing to facilitate and to explain the potential loses that we can avoid.

Critical Infrastructures pose new challenges to risk management mainly due to the following factors (Adar and Wuchner, 2005):

- The Nature of Critical Infrastructure Protection - It is fundamental to identify the key processes involved in each Critical Infrastructure and also their unique vulnerabilities.

- Organisational Complexity - Usually, Critical Infrastructures are of a big dimension and complexity usually with multiple interdependencies to external services. Risk management should be able to focus and follow the entire production process.

- Dynamic Aspects of Risk - Actual business is continuously changing and this is particularly true in Critical Infrastructures that keep evolving and creating new services for custumers.

- Need for Compliance - Usually, Critical Infrastructures are under supervision of governmental agencies and need to comply with National or International rules. Those rules can have a relevant impact on the way risk should be managed.

- Efficiency and Cost Effectiveness - In the real life economic scenario, efficiency and cost control are of highest priority for management decisions. Securing an infrastructure should be efficient but also cost effective.

- Human Factors - One of the major aspects of security relates to Human Factors. It is well know that human resources related to security need highly skilled specialisation in their work area. Also sensitive processes should not be kept in the hands of just a few individuals as this fact can, solely, become a new risk.

Risk assessment in Critical Infrastructures usually has a tight relation with the model used to gather CI information along with multiple methods, tools and frameworks. As several methodologies are described in the literature, some representative work will be discussed.

Haslum and Arnes describe a methodology for real-time risk assessment considering the measurements received from system's sensors (as for example Intrusion Detection System (IDS)). Both Hidden Markov models and weighted sums are used in considering the sensors input by assuming that some sensors are statistically more reliable and significant than others. The state of each represented asset is modeled using a Markov model with three different states: Good, Under Attack and Compromised. Hidden Markov model exist for each sensor in order to describe the state transition of the asset (probability that the asset passes from one state to another). In order to derive the risk for the complete system, the risks of each asset/sensor are summed and weighed according to their assumed reliability (Haslum and Arnes, 2006).

The risk management strategy described by Baiardi et al. is based on multiple models each one describing dependencies among infrastructure components and representing one CI specific level of detail. This set of models composes a hyper-graph model of infrastructures. The main objectives of this work are the detection of complex attacks as well as a support for risk mitigation. In this work, Baiardi et al. states that a dependency exists when a security-related attribute of a component depends on the attributes of other components. Each of the considered components has identified three security attributes (confidentiality, integrity and availability) evaluated from the components internal information and from the existent dependencies. One interesting aspect that is considered in this model is that each of the component's parameters can have influence to the same or different parameter in another component. For instance, low integrity in one component can affect the availability of another component (Baiardi et al., 2009).

Baiardi et al. perceives a complex attack as being carried out by a set of successive simple attacks affecting different components. It is proposed to build an evolution graph representing the states that can exist in a CI as a consequence of those simple attacks. The evolution graphs can then represent complex attacks describing a path from the first simple attack to the last simple attack. Risk analysis is then achieved by assigning probabilities, supported by historical data, to complex attacks. The proposal presented by Baiardi et al. aims to mitigate risks. The author assumes that a set of countermeasures can stop or mitigate simple attacks reducing the possible existent paths to form a complex attack. To support the described work tools were implemented to compute strategies to stop evolutions through a graph by eliminating a subgroup of simple attacks (Baiardi et al., 2009).

A framework for large-scale systems, able to identify, prioritise, assess and manage risk scenarios, was proposed by Haimes et al.. This framework is structured in eight steps. The first step, scenario identification, uses a hierarchical holographic model composed by multiple, complementary decompositions in order to better describe the system scenario. The second step, scenario filtering, aims to identify risks in the defined scenarios. Step three, bi-criteria filtering and ranking, is based on the risk filtering process considering different types of information: Likelihood and consequences. The fourth step, multi-criteria evaluation, the scenario is evaluated against it's ability to improve the resilience, robustness and redundancy of the underlying system. In the quantitative ranking process, fifth step, the probability of each scenario is computed using Bayes theorem. In the sixth step, risk management, an evaluation is made to the scenarios aiming to reduce the identified risks. In the safeguarding against missing critical items step the proposed risk mitigation strategy is evaluated against the previously filtered scenarios in order to be sure that the final implementation will hold against the unfiltered risk scenarios. The operational feedback step considers the fact that risk is not static and will evolve over time. Therefore, feedback and re-evaluation of the proposed method is applied to refine the scenario filtering process (Haimes et al., 2002).

2.1.3 Critical Infrastructure Modelling and Simulation

In order to better understand Critical Infrastructures and their dependencies, modelling and simulation techniques are seen as a key element for sucess. The main benefits that can be obtained using this techniques are described in detail by Rinaldi and can be summarised as being able to (Rinaldi, 2004):

- Help to identify the potential national and economic security implications following a catastrophic infrastructure failure;
- Improve the ability to perceive clearly the infrastructure operation when affected by extreme and rare events (for example natural disasters or terrorist attacks);
- Provide valuable information that helps to understand the recovery process after rare or extreme events that have led to catastrophic failures. Considering the rarity of these events, it is stated that modelling and simulation may provide the only guidance available as the historical records may be inexistent or in such a small number that they are useless in understanding the failure;
- Improve the process of infrastructure risk analysis (vulnerability assessment, consequence analysis, threat assessment);
- Use the obtained results in order to improve the development, testing, implementation and validation of the infrastructure protection policies. It is possible to incorporate policies into certain types of models allowing to simulate the effect that those policies have on the infrastructure behaviour and operation;
- Be used along with decision support tools that enable situational awareness (for example monitoring and visualisation). Modelling and simulation can be used for what-if analysis and for example simulating the consequences of a decision;

- Help deploying exercises and training related to CI protection in a similar way as is done by military personal using simulations in its war-games and training. In this context, simulations could be used to develop realistic instructional scenarios that better represent the effects of disruptions. These simulations are particularly relevant if encompassing extreme or rare events for which there is little information available.

In his work, Rinaldi defines different types of Critical Infrastructure interdependency models categorised in the following six different types (Rinaldi, 2004):

- Aggregate Supply and Demand Tools - These types of tools would be able to evaluate the demand of Critical Infrastructure services in a region and the ability to provide those services. The capacity an infrastructure has to meet the actual or future demands on its services can provide accurate references of its actual health and on potential problems acting as a early warning system;

- Dynamic Simulations - Allows to analyse the flow of commodities and services provided by Critical Infrastructures in order to model it's dynamics including also the interdependency among multiple infrastructures. Dynamic simulations can also be used to inspect the effects of policies, regulations and laws related to the Critical Infrastructures;

- Agent-Based Models - These types of models use agents in order to model the behaviour of Critical Infrastructures and the interdependencies among them. In order to better understand the CI, it is possible to model the CI's physical elements as agents allowing a better perception of its operational characteristics. Policy and decision makers can also be modelled as agents that allow us to examine its applicability and correctness;

- Physics-Based Models - Use engineering techniques (for example power flow and stability analysis on electric power grids) in order to provide a detailed model representing the physical behaviour of a Critical Infrastructure;

- Population Mobility Models - Aims to model the movements of entities (generating and consuming CI commodities) across urban regions. Modelling those movements or routines allows a better development of strategies for optimised planning. These models are often used to model transportation infrastructure scenarios;

- Leontief Input-Output Models - These models of economic flows, used to represent the interdependencies between economic sectors, represent a simplified view of an economy and are able to predict the proper level of production for each of the several types of goods or services. These models can also be applied to Critical Infrastructures using a linear, aggregated and time-independent analysis of the flow of commodities among CIs.

At the moment, a growing number of modelling and simulation approaches already exists or is under development. The majority of the available approaches are aiming to address interdependencies among CIs and to offer a considerable insight into the operational and behavioural characteristics of CIs. The main drawback identified among the existent approaches is the lack of comprehensive models, able to be easily adopted, according to the particularities of each CI.

2.2 Critical Infrastructure Protection Projects

As already stated, the European Union is highly concerned with the security level of European CIs. In order to address those concerns, in the past years the European Commission, in the scope of community research and development strategies, promoted some research projects related to CI Protection. The next section briefly describes three of those projects presenting their main achievements.

2.2.1 IRRIIS (Integrated Risk Reduction of Information-based Infrastructure Systems)

According to Klein et al., the Integrated Risk Reduction of Information-based Infrastructure Systems (IRRIIS) project aimed to enhance the dependability of large complex Critical Infrastructures. This was achieved by developing and applying appropriate modelling and simulation techniques and developing proper middleware based communication technologies (MIT) among CIs (Klein et al., 2009).

Authors have defined two application scenarios for the project. Those scenarios represent, respectively, one electrical power infrastructure and one telecommunication infrastructure that supports the first. For each of those scenarios the authors analyse the way CIs are connected to the exterior by the use of networks such as Internet (IRRIIS, 2008).

One main challenge addressed by IRRIIS is the different type and comportment of different Critical Infrastructures. IRRIIS approach to this was to build information models able to model physical aspects, as well as information and control aspects of CIs. Simultaneously, the project addressed the variety of existing CIs. Including the dependencies that may exist to other CIs while trying to maintain the model able to be generally used by multiple CI types.

As described by Klein et al., IRRIIS information model introduces three generalization models: (1) The generic information model intended to be used at the top level of the abstraction level, described as assuming that all Critical Infrastructures have a common core information model that can be used to instantiate more specific models. (2) The domain specific model that allows to extend the generic model integrating domain specific information related to specific components in each area (e.g. electricity network, or transportation network). (3) The instance level models, the top-down layer, that is intended to a specific CI, as instances of the domain specific model (e.g. electricity network of Provider A and transportation network of area A) (Klein, 2010; Klein et al., 2009).

The generic model has special relevance, as is used to allow generalisation and also is the place where interdependencies are considered. The IRRIIS generic information model is organised in three layers, namely, the static model, the behaviour model and the problem solving model.

The static model allows modelling the infrastructure according to services they provide and receive. In this model, components and systems are used to describe the CI structure and topology. Each system or component is represented by the services provided and by the existing connections among services. It also describes the services provided to another system and the services consumed and provided by other system. The static model provides the foundations that allow building the behaviour model on which ,states and their transitions are the key elements. As this model deals with system or service states and state transitions. The state of a single entity (e.g., system, connection, or service) is described. In the following step, the described states can be propagated to other interdependent systems, components or services. The problem solving model encompasses a mechanism not only to trigger existing state changes in the behaviour model (events) but also to react to state changes in the behaviour model (actions).

Project IRRIIS has developed a set of applications called Middleware Improved Technology (MIT). Those applications allow the communication among different heterogeneous CIs that usually have incompatible applications. MIT's main objective is to permit a simple, fast and reliable information exchange among CIs, thus reducing response time to incidents that may occur within the CIs, by maintaining network managers well informed about the CI state (IRRIIS, 2008).

Each MIT is composed of two main components: the communication and add-on components. Communication components are used to exchange information related to events occurring in the CI and are also used to send negotiation messages related to QoS in the CIs. The add-on components were developed with multiples objectives in mind. The main add-ons are namely: a tool to extract the CI functional status; a risk estimator based on expert systems that is able to make predictions regarding expected risks according to the information gathered; an incident knowledge analyser tool used to send information about accidents that occurred in the CI to a database; a knowledge based tool that allows for the knowledge management, considering the existing rules and lastly a tool able to reason on the functional state of the system and recommend actions to be applied to it (Schembri, 2008).

According to Klein et al., the IRRIIS information model provides an important framework for CI interdependencies simulation, risk estimation and decision support within CIs. The validation approach for the project is based on federated simulation. A special purpose simulation tool - Simulation for Critical Infrastructure Protection (SimCIP) – is used to simulate different CIs while the IRRIIS information model is used to deal with interdependencies among the infrastructures on top of the special purpose simulations (IRRIIS, 2008; Klein et al., 2009).

2.2.2 CRUTIAL (CRitical UTility InfrastructurAL Resilience)

CRitical UTility InfrastructurAL Resilience (CRUTIAL) project (CRUTIAL, 2008) aimed to improve the resilience of Critical Infrastructures in order to avoid or minimise problems that may occur due to the large and complex CI ICT systems and also to the increasing number of interconnections among CIs (CRUTIAL, 2008). According to Verissimo et al., the CRUTIAL project provides an architecture including multiple tools and algorithms aimed at improving resilience on global critical information infrastructures, taking into account computer-borne attacks and faults (Verissimo et al., 2008b). Although the CRUTIAL architecture is focused on the computer systems that support an electrical utility infrastructure, Verissimo et al. describes it as a useful reference for all types of Critical Infrastructures. In actual fact, the CRUTIAL architecture is used as a reference to various studies and in particular to multiple European projects.

The existing interdependencies among CIs are analysed and the need of an architecture that considers the global view on those interconnected infrastructures is identified by the CRUTIAL approach which provides a global view on the interconnected infrastructures. It is also discussed that conventional security mechanisms are not able to be directly applied to CI protection (Bessani et al., 2007; Verissimo et al., 2008a,b).

According to Verissimo et al., CRUTIAL architecture is composed of four main areas: (1) Architectural configurations - placing trusted components in key places; (2) Middleware devices - supply trusted services out of non-trustworthy components; (3) Trustworthiness monitoring - detection of non predicted situations, providing adaptation mechanisms to survive those situations; (4) Definition and application of organization-level security policies and access control models for securing information existent in the CIs (Verissimo et al., 2008b).

CRUTIAL architecture is established with an intrusion tolerant design in mind on which the resilience of infrastructures is achieved by deploying a trustworthy operation supported by secure and trusted hardware. It uses a similar approach to the Trusted Platform Module (TPM) (TCG, 2013), being able to support intrusion tolerance for the rest of the system and by transparent intrusion tolerance achieved without changing the legacy structure of the existing components (for instance, existent SCADA systems and technologies used in the corporate network are basically unchanged) (Verissimo et al., 2008b).

Despite the possible existence of faults and intrusions, the non-stop CI operation is enabled by the developed methodologies (proactive-resilience). The components behaviour is monitored over time in order to detect deviations (components that have a behaviour different from the expected). This information is analysed by a state diagnosis component (that guesses the internal state of a component based on the deviation detection) in order to reason whether some individual components may or may not affect the rest of the system. This detection is particularly useful to lower the risk of spreading cascading events (Verissimo et al., 2008b).

CRUTIAL infrastructure is described as a Wide Area Network (WAN) composed by several Local Area Networks allowing a secure communication among components and also within infrastructures. This WAN-of-Local Area Networks (LANs) interconnects the local communication components (LAN) within a part of the CI as, for instance, a substation or a corporate office. The LAN collection is interconnected via a global connection network WAN using, for instance, dedicated lines or the Internet. All the application level firewalls and IDSs are connected using this approach (Bessani et al., 2007).

The communication security is assured by employing communication devices that are built as trusted components projected to be also intrusion tolerant. This last aspect is achieved by introducing redundant components and by employing a voting system (majority voting) on the components outputs. The fact that local components are built in order to be trustworthy helps to assure secure communications on a global level (WAN) (Bessani et al., 2007).

CRUTIAL architecture also includes solutions for access control allowing to enforce global-level security policies defined in each Critical Infrastructure. Global communication devices check the local security policies according to the global defined policies. The access control model Organization-Based Access Control (OrBAC) is used to achieve the proposed approach allowing to model the organization as a regulated group of entities in which each participant can have a specific role (Verissimo et al., 2008a).

In order to improve the security in a context of collaboration among multiple Critical Infrastructures, CRUTIAL uses PolyOrBAC in order to publish and negotiate access control rules in a intra-organizational context. In adittion, allowing it to support an Authentication, Authorisation and Accounting (AAA) architecture to deploy control access for runtime access to remote services (Verissimo et al., 2008a).

The main purpose of the CRUTIAL project was to enable the collection of improved knowledge about Critical Infrastructures, permitting the development of more resilient infrastructures. The validated results obtained within the CRUTIAL project are considered a major contribution to the development of the state of the art in CI protection (Bessani et al., 2007; CRUTIAL, 2008; Dondossola et al., 2008; Verissimo et al., 2008a,b). The CRUTIAL developed methodologies and tools are highly focused on ICT and the Electricity Power Systems thus the applicability in different areas becomes more difficult. As the main goal of CRUTIAL is to develop more resilient infrastructures, it does not considers the existence of alerting systems or risk information exchange among CIs.

2.2.3 INSPIRE (INcreasing Security and Protection through Infrastructure REsilience)

As described by D'Antonio et al., the INcreasing Security and Protection through Infrastructure REsilience (INSPIRE) project aimed to enhance the European potential in the field of security by ensuring the protection of Critical Infrastructures through the identification of their vulnerabilities and the development of innovative techniques for securing networked process control systems (D'Antonio et al., 2009).

The core idea of the INSPIRE project is to protect Critical Infrastructures by appropriately configuring, managing, and securing the communication network which interconnects the distributed control systems. A working prototype has been implemented as a final demonstrator of selected scenarios.

The main research objectives addressed by the INSPIRE project were: the design and implementation of traffic engineering algorithms to provide SCADA traffic with quantitative guarantees, thus increasing SCADA resilience to attacks or malfunctions; the use of peer-to-peer overlay routing mechanisms for improving the resilience of SCADA systems; the design of an architectural framework for SCADA systems monitoring, diagnosis and reconfiguration and also the development of diagnosis and recovery techniques for SCADA systems (INSPIRE, 2010).

2.3 The CI Security Model

Considering that the operation of complex systems such as Critical Infrastructures is a challenging task, CI providers place substantial efforts into maintaining CIs running while trying to reduce risks of any kind, particularly, the risk of failure, the risk of intrusion or the risk of incorrect operation. With these problems in mind, a novel approach for security modelling and CI risk evaluation and monitoring was presented by Schaberreiter et al. – the CI Security Model (Aubert et al., 2010a,b; Schaberreiter et al., 2011a). Aubert et al. states that current risk analysis methodologies do not provide a simple framework to share risk information among Critical Infrastructures. Essentially, the main identified problem that leads to the development of this model is the fact that CIs owners are not enthusiastic about exchanging risk information, mainly due to confidentiality reasons and also because they do not intend to publicise the details regarding their systems.

In a more recent work, Schaberreiter et al. defends that the main idea behind this work is to be able to estimate, in real-time, the existent risk in CI services. The CI security modelling approach addresses the challenge of on-line monitoring of the state of CI services taking into account the services they depend on. In this work, the types of risks considered are, respectively, a breach of confidentiality, a breach of integrity and a degradation of availability (Confidentiality, Integrity and Availability) of a service. To be able to estimate these risks, the necessary evidence is gathered from measurements retrieved from the existent CI components (base measurements). One important aspect included in the CI Security Model is the ability to consider risk information available from dependent CI services in the risk calculation of a CI service (Schaberreiter et al., 2011a).

By applying the CI Security Model, Schaberreiter et al. aims to simplify the infrastructure complexity in the model through the use of security properties in order to create an abstraction layer over the physical implementation of the services. Aubert et al. suggests that this abstraction can be applied to a wider range of systems (e.g. energy, telecommunication, air traffic) as they share the same security objectives. Another major benefit of this model is that the information exchanged among CIs is specifically related to the security belonging to shared services, thus keeping confidential information inside each CI. As providers usually hesitate to share the information that would enhance the security of their CIs, it is assumed that the abstraction to a small set of common parameters will encourage service providers to share them with (inter)dependent providers (Aubert et al., 2010b).

As illustrated in Figure 2.4 the aim of the approach is to transform real-world CI information to abstract risk related information (in this case CIA - confidentiality, integrity and availability). Aubert et al. denotes that this approach fills a gap in Critical Infrastructure modelling by presenting a more complete view on CIs, whereas the existing previous models are more focused on modelling system failures in order to deal with the consequences of those failures. In the CI Security Model, system failure is only one of the three aspects the model pretends to represent as it is directly related to availability (Aubert et al., 2010b).

The information obtained by applying the CI Security Model can then be used to (1) monitor the state of the Critical Infrastructure and (2) share it with (inter)dependent CIs in order to be able to evaluate the current infrastructure risk by incorporating the risk related to the existent dependencies.

The CI Security Model's underlying methodology approach is composed of three

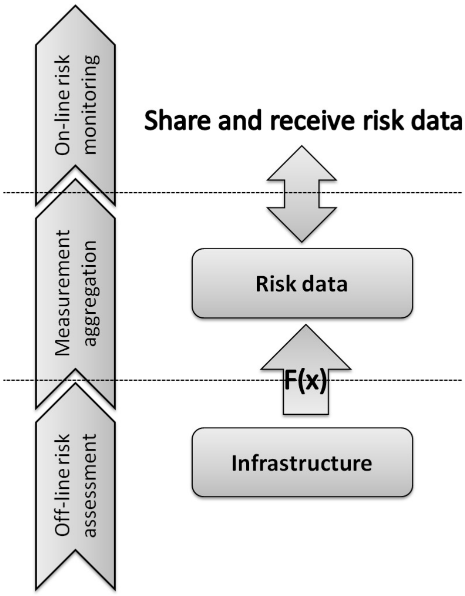


Figure 2.4: CI Security Model approach (Aubert et al., 2010a)

steps, namely, the off-line risk assessment, the measurement aggregation and an on-line monitoring step as represented in Figure 2.4.

Off-line risk assessment

The off-line risk assessment step is of special importance in the CI Security Model since risk estimation and monitoring can only be precise if the structure of the systems is amply understood and captured. The off-line risk assessment, the first step in the CI Security Model, is intended to adequately analyse CIs in order to recognise the entities that must be represented in the model, namely, the critical services, critical service (inter)dependencies and the base measurements.

To successfully complete the first step, proper information should be gathered from multiple sources, for example from various social (e.g. management, technical personal) and technical (e.g. documentation, manuals, vulnerability feeds) sources. According to these information, the critical services that compose a CI can be properly identified and described. To deal with the expected complexity in CIs, the CI Security Model allows to decompose each identified service into a set of more fine-grained sub-services.

In order to have an expressive representation of the risk in a complex CI, it is proposed to decompose the CI into smaller and simpler parts. For example in

Figure 2.5, CI A is decomposed into a set of services provided by the CI. Depending on the size and complexity of a CI, each of the identified services can be further decomposed into sub-services used to provide the higher level service (super service), which results in a hierarchical, tree like representation.

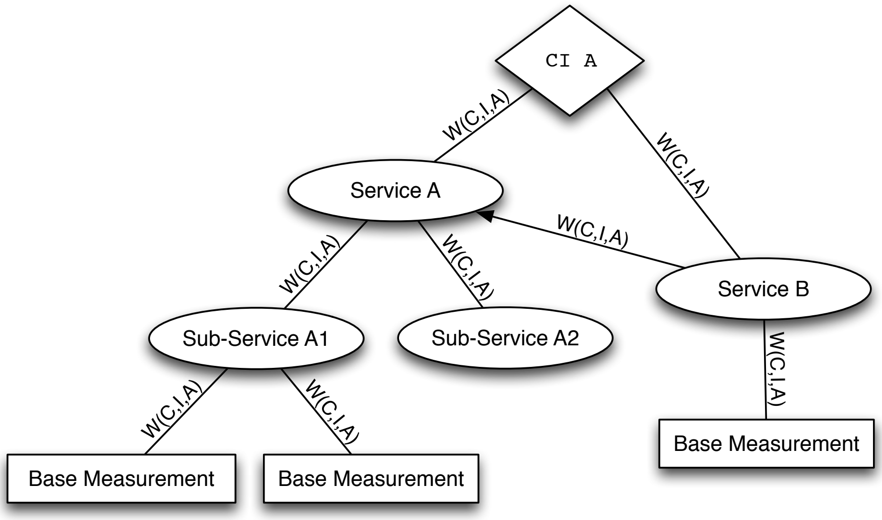


Figure 2.5: Weighted infrastructure decomposition graph

In general, each identified service can be divided into components that provide base measurements for risk aggregation or sub-services. The lowest level of each path through the decomposition tree should contain components that are used in order to provide the service. Logically, each sub-service can be seen as a dependency of its super service. Service dependencies are also taken into account. Dependencies can exist either to one of the other services identified during the decomposition (like Service A and Service B in Figure 2.5) or to a service provided by another CI. The arrow direction in the example represented in Figure 2.5 highlights that Service A depends on Service B.

With a simple representation, each service can then easily be investigated separately in order to identify specific base measurements within CI components that define the state of the CI service and to identify dependencies to other internal or external CI services. Logically, sub-services can be treated as dependencies of their super-service (Schaberreiter et al., 2011a).

Following the risk assessment step, CIs are represented using a directed graph including only three entities: the critical services, the dependencies among critical services and all the base measurements associated with each critical service.

One important aspect in the CI Security Model is the normalisation of the base measurements. This normalisation is necessary due to the variety of existent CIs, each one with multiple heterogeneous components providing measurements in different formats, ranges or quantities. In order to allow a continuous and accurate measurement from the components, the model needs to be able to deal with, for instance, different (physical) quantities and different value ranges.

The normalisation is achieved by estimating the measurement output during normal operation and by defining ranges for the allowed deviation from normal operation. For the CI Security Model, it is necessary for the base measurement outputs to be changed to a discrete scale of 5 levels (1 meaning normal output and 5 indicating the maximum deviation from normal output). As stated by Schaberreiter et al., a 5 levels scale was chosen as a compromise between granularity of risk representation and an easiest understanding of the information by an operator within a stress situation (Schaberreiter et al., 2011a).

Base measurements informing only boolean values can only take the discrete values 1 (reached) and 5 (not reached) in order to comply with the defined discrete scale. The estimation of normal operation and the categories of allowed deviation from normal operation is again done by taking into account all available social and technical sources that can provide evidence for the identification of the correct values.

A key aspect of the off-line risk assessment step is to weigh the significance of each dependency and each base measurement according to their relevance to, respectively, the confidentiality, the integrity and the availability $W(C, I, A)$ of the service. This enables quantifying the influence that each base measurement or dependency has to a service (Schaberreiter et al., 2011a).

In a scenario where a base measurement or dependency does not contribute to the Confidentiality, Integrity or Availability (C,I,A) of a service, a weight of 0 is assigned. Again, all concerned social and technical sources are employed to estimate accurate values for the weights. Schaberreiter et al. refers that these weights are assigned by CI experts and are only modified in case of a new iteration of the off-line risk assessment if some inaccuracy is detected (Schaberreiter et al., 2011a).

Measurement aggregation

As a result of the previous step, all information needed to evaluate the service risk (R_S) is identified and properly normalised. The measurement aggregation step evaluates each service risk (R_S) using an averaged weighted sum of the normalised base measurements (μ) and dependent service risk (R_{Dep}) using the corresponding weights (ω) assigned in the off-line risk assessment step. For each risk indicator (CIA), the evaluation is achieved as presented in Equation 2.1 (Schaberreiter et al., 2011a).:

$$R_S = \left\lfloor \frac{\sum_{i=1}^{n} \mu_i * \omega_{\mu_i} + \sum_{i=1}^{m} R_{Dep_i} * \omega_{Dep_i}}{\sum_{i=1}^{n} \omega_{\mu_i} + \sum_{i=1}^{m} \omega_{Dep_i}} \right\rfloor . \tag{2.1}$$

Service risk (R_S) indicators are evaluated for each identified service according to the CI Security Model perspectives for risk indicators (CIA), each one characterising a risk level between [1..5] (Schaberreiter et al., 2011a).

On-line risk monitoring

The on-line risk monitoring step is responsible for the risk information distribution to the existent dependent services and also to provide aggregated risk indicators to the CI Operator. A further particular aspect highlighted by Schaberreiter et al. regarding the planned model simplicity and improved simplicity, refers to the importance of representing the risk in an easy and comprehensible format. In this way an Operator can react quickly to changing risk and be able to determine the source of the increasing risk (Schaberreiter et al., 2011a).

Risk in this context can be seen as a situation where the CI behaviour is different from the expected behaviour. This definition can be applied to virtually all situations where a CI service behaves differently from normal operation. In this approach, the changes in CI behaviour can be expressed numerically with the CIA indicators.

It is assumed by Schaberreiter et al. that by following the presented CI decomposition process, it is possible to represent a complex CI using the security model. Since this approach allows to freely define logical entities as services, it is easy to aggregate risk only for parts of the CI that an interdependent CI or CI services are interested in. This provides a more accurate measure of interdependent service risk while still hiding the complexity of a CI service behind risk parameters.

Recently, Schaberreiter et al. proposed an improvement to the CI Security Model through the modification of the methodology employed to evaluate risks. The idea is to represent the model as a Bayesian Networks (BN) where the nodes are used to represent CI services and system measurements, and the edges represent dependencies between nodes. The BN Conditional Probability Table (CPT) have probabilities that can be learned from data records and also assigned by CI experts based on their experience (Schaberreiter et al., 2013).

According to Schaberreiter et al., using a Bayesian approach for the CI Security Model allows for some improvements to the previous approach. Among others, it is stated that the graphical representation of the model is easier to interpret and more importantly, that risk prediction is improved as risk is estimated both immediately after an incident and also estimated for the mid and long-term consequences of an event (Schaberreiter et al., 2013).

2.4 Common ontologies for CIs management

New generation networks are becoming more independent from the services they support, while the information that is needed for the management of those networks is becoming more dispersed and located in heterogeneous sources. The use of ontologies to support event management in CIs is relevant, as it is possible to take the management out of the physical infrastructures and focus on the functional network components where the network manager or CI Operator can have a better CI overview (Hochstatter et al., 2008). There are some projects with already good results aiming to develop frameworks able to improve the way this type of information is represented, including representation about events occurring in multiple contexts.

Ontology is a formal representation for concepts regarding a domain that includes the relations among these concepts. It can be used to gain knowledge of the properties of that domain, and also to define the domain. In order to model Critical Infrastructures, the use of ontologies is commonly used.

The way information needed to manage Critical Infrastructures is represented has become a very important question that has been studied in recent years. Due to the development of new information systems and communication technologies, actual systems are becoming more heterogeneous. This is particularly true in the Critical Infrastructures context (Panzieri et al., 2010). Actually, an adequate integration of all information produced inside an organisation is an important and complex issue as interoperability among systems is becoming more complex. Existent scenarios have a vast diversity of systems and technologies with each one using different techniques of sharing information. In some cases, for each scenario, a different modelling approach is used in order to manage vital system information. These different modelling approaches create difficulties when one intends to share information with users or external systems (Serrano et al., 2007).

One actual challenge to develop best interoperability practices among Critical Infrastructures is the development or adoption of an information model with the capacity to define all the existing concepts using a method that is technologically neutral. By using such a method it is possible to develop information models that can be shared and integrated with others. These models will be the "knowledge database" of the system (Strassner, 2003, 2004).

In the actual state of the art, the use of Ontologies is proposed to represent critical information. Ontology can be defined, in this context, as being "an explicit specification of a conceptualization" (Gruber, 1993). In this area, the use of ontologies to manage knowledge must focus on building models that can represent CIs. The adopted ontology must provide a vocabulary used to establish the model. To develop and use an ontology is vital to have the consensus of almost all entities involved in the infrastructure that is planned to model.

In this context, a well-defined Ontology for CI management must allow (Serrano et al., 2007):

- the representation of knowledge without ambiguous interpretations or inconsistencies;
- the sharing of knowledge between heterogeneous sources of information;
- the provision of an accurate description of knowledge without the existence of ambiguities;
- the expansion from generic domains to specific areas using hierarchical structures;
- the creation of a formal semantics easily understood by all participants in the process that can be easily implemented in systems.

In Critical Infrastructure modelling context, it is important to identify and define the common cross-domain semantic elements that can be used to describe the services and the data of the heterogeneous CI information systems and their interdependencies. The use of ontologies simplifies the definition not only regarding tangible objects, but also abstract concepts and paradigms, relations, references, indicators and dependencies. A semantic language defines an ontology, which is an extensible conceptual reference infrastructure that could be used by all the CIs to define themselves and the interfaces they expose in a comprehensible way (Panzieri et al., 2010).

Critical Infrastructures semantic must be described using standard ontology languages to minimise the overcoming of the heterogeneity and the ambiguity of existing syntaxes and also to obtain a common semantic. A common semantic is a key aspect for a successful interoperation. It can be automatically processed by a distributed system allowing different autonomous systems to communicate with each other without ambiguity (MICIE Consortium, 2008).

As several languages and tools are available to represent and use ontologies, it was chosen to describe among the ones analysed, two that are considered the most representative.

Resource Description Framework (RDF)

The Resource Description Framework is a recommendation from the World Wide Web Consortium (W3C) for a graphical language originally defined for the representation of information related to existing resources on the World Wide Web (WWW) that can also be considered as a basic language for ontology representation. Resources are declared in the language through the properties and values associated with those properties. Resource Description Framework (RDF) is supported on XML and by the use of Unique Resource Identifiers (URIs) to identify each feature (W3C, 2009). On the RDF's definition it uses only binary properties, which is a disadvantage when it is intended to use predicates with more than two arguments. Another limitation of this language is related to the treatment of classes and properties management. The semantic extension of RDF - RDF Schema language allows the representation of ontologies using basic classes, properties and fields, bringing the possibility of building ontologies in a simple and formal way (Serrano et al., 2006).

Web Ontology Language (OWL)

Web Ontology Language (OWL) is defined by W3C as one of the standards applicable to ontologies. Initially this language was designed to describe classes and was derived from DAML+OIL (DAML, 2001) – Darpa Agent Markup Language (DAML)/ Ontology Interface Layer (OIL) – this language is based also on RDF. Classes are the basic elements used to construct ontologies with OWL. This language also contains a large number of constructors ready to use. OWL has three versions (Lite, DL and Full) – OWL Lite supports those users primarily in need of a classification hierarchy and simple constraints; OWL DL supports those users who want the maximum expressiveness while retaining computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). OWL DL includes all OWL language constructs, but they can be used only under certain restrictions (for example, while a class may be a subclass of many classes, a class cannot be an instance of another class). OWL DL is so named due to its correspondence with description logics; OWL Full is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. OWL Full allows an ontology to augment the meaning of the pre-defined (RDF or OWL) vocabulary. Existing versions permit this language to be used in various areas with different needs. Each OWL version is an expansion of the previous version. An ontology represented in one version is always valid using the next version. The opposite may not happen (W3C, 2004). The use of OWL has left the ambit of its initial specification (represent Semantic Web information) beginning to have a wider use in other applications where the use of ontologies is needed. Considering the complexity of the Critical Infrastructures that one pretends to model, OWL language has proven to be a strong candidate for the task (Uszok et al., 2008).

In order to use ontologies, several software tools are available. For this work two of them were evaluated, namely Protégé and Swoop.

Protégé

Protégé is developed and maintained by the Stanford Center for Biomedical Informatics Research at the Stanford University School of Medicine as a free open-source platform. It is composed of a group of tools able to construct domain models and knowledge-based applications supported by ontologies. This platform implements a set of knowledge modelling structures and actions and supports the creation, visualisation and manipulation of ontologies in various formats. Protégé is highly adaptable and supports various application domains. In the actual version, this tool can be extended by the use of a Java-based Application Programming Interface (API) for building knowledge-based tools and applications that can take advantage of the created ontologies. One of the advantages of using Protégé was the development of Protégé-OWL. This is an extension to Protégé that supplies support to the use of OWL language. Protégé was developed under a flexible architecture based on plug-ins development which resulted in an adaptable and easily expandable tool (Protégé, 2011).

Swoop

Specifically designed to support OWL language, Swoop tool has been initially built to work as an ontology browser and editor. Developed by the MIND Laboratory – Maryland University, it permits the creation, edition and ontology debugging using OWL. This project has recently become an open-source project (Swoop, 2009). Swoop is based on the actual web browsers paradigm, having as its principal characteristic the use of URIs to allow construction and understanding of OWL based ontologies. The application was designed to have an interface similar to web browsers including, for example, navigation buttons and address toolbar (Uszok et al., 2008).

According to an enquiry made in the beginning of 2007 to ontology users in the field of semantic web (Cardoso, 2007), the presented languages (OWL and RDF) are the ones chosen by the majority of the users – 76% of the users have chosen OWL and 65% are using RDF. The same study has concluded that Protégé is the tool most used by the users with a total of 68.2% against only 13.6% that use Swoop.

Taking into account the needs observed mainly for the MICIE project, the use of OWL and Protégé was considered to construct the ontology and also to use these tools to create a security ontology containing policies that will allow the implementation of security policies within the MICIE system.

2.5 Policy Based Management

Considering the important role that ICT plays in a Critical Infrastructure and in particular when it is planned to share information among CIs, the problem of how to manage such a system needs to be addressed. In this case, policies can be used to regulate the content and amount of information that should be shared among peers. More specific policies allow supporting dynamic decisions based on the information available at each moment regarding the participation of each peer in the communication. For instance, it should be possible to integrate indicators such as Trust and Reputation while taking decisions. Also, as they are frequently used, policies must be able to be used to enforce confidentiality, accountability, identity management and access control among other aspects.

In a scenario where CIs are willing to exchange information, it is important to deploy management strategies that address security aspects while permitting an easy definition of security rules by the CI Operator or System Administrator.

Currently, a great effort is made to diminish the increasing complexity of networks management, through the use of multiple paradigms. The Policy-Based Management Model aims to be the result of the change from the actual configuration mechanisms to an integrated management system.

One interesting work has been published by Li et al. where authors describe a framework that makes use of an holistic approach, based on the concepts of situation awareness for monitoring the state of a CI employing policies to manage the proposed system. Although the authors refer to the system as being able to be used in Critical Infrastructures security management, the application of the proposed system is illustrated by using it to protect the traditional Internet backbone by automatically configuring Border Gateway Protocol (BGP) on router systems and the application to help secure devices and information in mobile networks. This is an interesting work in terms of Policy Based Network Management (PBNM) not only for using the PBNM approach but also by incorporating some reasoning regarding information trust on the decisions taken by policies (Li et al., 2012a).

Li et al. has recently presented another work that makes use of policies and trust in order to improve security in Mobile Ad-hoc Networks (Li et al., 2013). In this work, Li et al. describes a framework able to use multiple existent contextual information such as battery status, and weather condition, in order to determine whether some misbehaviour is a result of malicious activity or not. The author also uses trust information gathered from the devices in order to improve the Policy Management.

Network and systems management solutions based on the use of policies are not new. The PBNM model has been adopted by the Internet Engineering Task Force (IETF) and the Distributed Management Task Force (DMTF) for networks and distributed systems (Yavatkar et al., 2000) with the following main objectives:

- Centralised network management;
- Support for abstract definition of rules and policies;
- Use of the same rules for different types of equipment and automation of network management tasks.

In this concept, the system administrator must describe what the network or system should do, instead of worrying about the way in which this will be implemented. PBNM architecture is commonly composed of four main entities (Figure 2.6):

- Management console - allows the network manager interaction with the system in order, for instance, to define, change or remove policies;
- Policy repository - the place or places where the defined policies are stored;
- Policy Decision Point (PDP)- responsible for taking decisions reasoning according to the defined policies and the information available at each moment;
- Policy Enforcement Point (PEP) - the entity that enforces the actions decided by the PDP.

The communication among these entities can be achieved by using different protocol, namely,

- one protocol to access the repository (for example Lightweight Directory Access Protocol (LDAP));
- a protocol for policy exchange (for example Common Open Policy Service (COPS) or Simple Network Management Protocol (SNMP)).

The use of PBNM architecture permits reducing the volume of administrative tasks and the amount of errors originated from user intervention when configuring devices or applications and also an easier deployment of fine-grained policies.

It is possible to model and implement the proper behaviour of a network or a system with the use of policies. Using policies, the desired behaviour of a system can be formally modelled, providing support for distribution of tasks, automation and adaptation of a controlled system, with scalability, flexibility and consistency (Damianou, 2002).

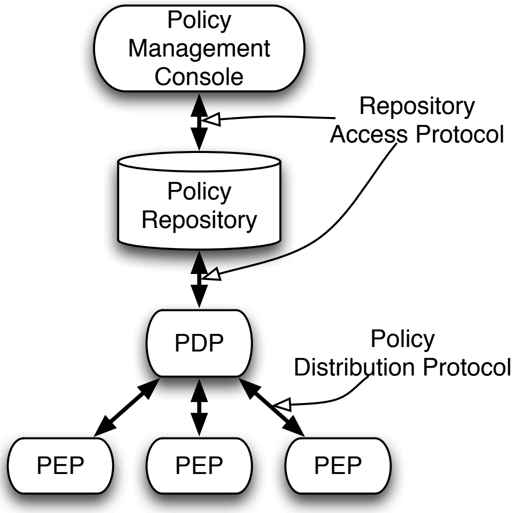

Policy Management Console
Repository Access Protocol
Policy Repository
PDP
Policy Distribution Protocol
PEP
PEP
PEP

Figure 2.6: PBNM architecture components.

In the context of this thesis, the use of policies can support writing, verification and deployment of security policies able to protect the information gathered by one Critical Infrastructure. For example, defined policies can allow one to:

- define how and to whom each particular piece of information can be sent;
- define trust relations between different CIs;
- enforce different communications protocols/technologies in each particular context;
- enforce Service Level Agreements (SLAs) or Service Level Specifications (SLSs) between CIs;
- decide how received events will be managed by the CI.

In order to represent policies used in PBNM architecture, a language is needed through which the network manager can describe system behaviour/configuration. There are several proposals, each one concerning an area of application. The language must supply a unique platform with support to concepts existent in PBNM architecture. In this scenario, it is imperative to identify requirements that a policy specification language should support.

A policy specification language must be simple and easily understandable by the users. It must support multiple management activities, like security policies and access control. The policies must be grouped and not treated individually in order to facilitate the policy specification related to complex networks. It also has to permit the relation between multiple areas of network management.

The language must allow some type of policy composition, with the possibility to analyse in terms of conflicts with other policies, as well as verifying the consistency of global specification. The language must be expandable in order to allow new policy types that will appear in the future. With the existence of diverse language specification, each one with its proper syntax and semantics, one must choose a language that is sufficiently extensible and with a great degree of scalability, permitting it to add new functionalities without much additional effort. Examples of existent languages are the Extensible Access Control Markup Language (XACML) (OASIS, 2013), the Rei policy language (Kagal, 2005) and PONDER (Ponder, 2010).

XACML is supported by XML allowing it to express access policies. XACML allows the user to manage actions and supports the resolution of possible existent conflicts among policies (OASIS, 2013). Rei is a policy language designed for pervasive computing applications and it is intended to deal with deontic concepts (obligation, permission, and related concepts) while being based in a semantic language (Kagal et al., 2003). PONDER, and his updated version PONDER2, focus on simplicity, flexibility and extensibility in order to provide users with the ability to interact easily with the managed system. PONDER2 is easily able to interact with multiple software and hardware components and is being used in environments ranging from single devices, to personal area networks, ad-hoc networks and distributed systems (Twidle et al., 2009).

Regarding the flexibility allowed by PONDER, it was proposed within the MICIE consortium to evaluate the PONDER and PONDER2 languages (Ponder, 2010) developed at the Imperial College – London.

PONDER consists of a set of tools and services that were developed for the specification, analysis and enforcement of policies - the name PONDER became associated with the entire toolkit. PONDER2 has since been developed as a significant redesign and re-implementation of PONDER.

PONDER2 is an extensible framework that can be used at different levels of scale from small-embedded devices to complex services and Virtual Organisations. This framework provides a means of specifying security policies onto access control implementation mechanisms for firewalls, databases, shared ontologies, among others (Twidle et al., 2009).

In PONDER2 context, policies are seen as rules governing choices in the system behaviour. Two types of policies are supported - authorisation and obligation policies. Authorisation policies define which actions are permitted under given circumstances and obligation policies define which actions should be performed in response to an event occurring if specific conditions are satisfied. Policies can be dynamically changed, loaded, enabled, and disabled without interrupting the system.

PONDER2 provides positive authorisation (auth+) and also negative authorisation (auth-). Only one type of obligation policy is specified, stating that a subject is obliged to perform certain action on that target. An obligation policy can be enforced only if the corresponding authorisation policy has been specified in the system. An event field specifies the trigger of the obligation. Optional constraints may apply, and in this case, they are evaluated against the state of the system.

In the actual development stage, PONDER2 comprises a self-contained, stand-alone, general-purpose object management system with messages passing between objects. It incorporates an awareness of events and policies and implements a policy execution framework. PONDER2 has a high-level configuration and control language – PonderTalk - a high-level, object orientated language. It is also possible to expand PONDER2 system with simple Java programming (Ponder, 2010).

As stated by the developers (Ponder, 2010), PONDER2 has been designed to achieve the following goals:

- Simplicity: The design of the system must be as simple as possible;
- Extensibility: It is possible to dynamically extend the policy environment with new functionalities;
- Self-containment: The policy environment does not depend on any-infrastructure and contains everything necessary to apply policies to managed resources;
- Ease-of-use: Facilitates use of policies in new environments and in different applications;
- Interactivity: Managers and developers can simply interact with the policy environment, issue commands to the Managed Objects and create new policies;
- Scalability: Policy environment must be executable in almost any type of resource.

One important concept in PONDER2 is that of Managed Object. A Managed Object is an entity capable of receiving and replying to PonderTalk messages. A Managed Object is written in Java and uses Java annotations to create the links between PonderTalk message keywords and Java methods. A Managed Object may communicate with other Managed Objects, it may use Swing as part of a GUI, it may act as an Adaptor and communicate with external entities, among other possibilities of use (Ponder, 2010).

In PONDER2, three pre-defined types of Managed Objects exist: domains, event templates and policies. Objects are addressed by name, (e.g. root/policy/temppolicy) and their basic value types which include Strings, Numbers, Arrays, Hashes, XML. New Managed Objects are written in Java with simple annotations to manage messages (Ponder, 2010).

One other important concept is the Self Managed Cell (SMC) that is defined as a set of hardware and software components forming an administrative domain that is able to function autonomously and is thus capable of self-management. This concept is extremely important, as each SMC will be an autonomous system with regards to management. Inside the SMC, existing management services interact with each other through asynchronous events propagated through a content-based event bus. Each SMC is able to interact with other SMCs and thus able to compose in larger scale SMCs (Ponder, 2010).

PONDER2 concept of Domain is used to refer to a collection of objects explicitly grouped for management purposes, for example to apply a common policy. Domains can be nested and can overlap with others. It is possible to specify policies that will apply to domains instead of single Managed Object. The use of domains can be very useful to apply the same policies to the same type of Managed Objects (Ponder, 2010).

As already stated, PONDER2 uses PonderTalk language to specify Managed Objects and the messages to be sent to them. PONDER2 version 1 uses XML for the configuration and control language. PonderTalk is based on Smalltalk. It uses a simple syntax in a sequence of statements. Statements are like sentences and they are separated with a full-stop (period). A statement specifies a receiver (object) and a message (command) to be sent to the receiver. The receiver then returns another object (or itself) in response to a message (Ponder, 2010).

2.6 Trust and Reputation

Along with the growing number of publications in the area, it is possible to find many definitions of trust. The main concept appears in different areas such as sociology, economics, law and computer science. The concept itself depends on the application area, yet it is commonly used to assist decision processes such as, assisting a customer when buying from an online store where he can find different opinions from previous clients allowing to mentally create a notion of trust or distrust. One interesting work was published by Miller et al. in which, authors begin by trying to understand the concept of trust. One thought-provoking sentence allows one to understand how ambiguous the term trust can be and also to realise how it should be used - *"Trust is less confident than know, but also more confident than hope."* (Miller et al., 2010). Miller et al. also makes one think that trust also includes an element of risk. That is, if someone knows some fact then to trust someone else about that fact is unnecessary, but when someone does not have the means to know then it is important to trust.

The difficulties to define and conceptualise trust are also realised by the essay written by McLeod in the The Stanford Encyclopedia of Philosophy McLeod. Until a complete consensus is reached (if this is possible in such a widely applied concept), different authors that make use of trust use different definitions leading to some misunderstanding about their work. For example, Gambetta rationalises trust as *"if we trust someone, we implicitly mean that the probability that he will perform an action that is beneficial or at least not detrimental to us is high enough for us to consider engaging in some form of cooperation with him."* (Gambetta, 2000). Another interesting definition has been given by Ruohomaa and Kutvonen, where trust is *"the extent to which one party is willing to participate in a given action with a given partner, considering the risks and incentives involved"* (Ruohomaa and Kutvonen, 2005).

For reference, in this thesis, the trust definition presented by Mui and Mohtashemi is used - *"Trust: a subjective expectation an agent has about another's future behaviour based on the history of their encounters."* (Mui and Mohtashemi, 2002). It is also possible to state that trust is the opinion of one entity about another single entity, while reputation is the community opinion about one entity. In this thesis, the concept of reputation is slightly different. It is considered that the reputation of an entity is an indication of how much one trusts that entity, taking into consideration the trust one has in a collection of multiple aspects regarding that entity. The trust concept is an ancient one in human relations, however it is relatively new in the computer science areas. Although in recent years it has become widely adopted to help solve problems in multiple ICT areas, particularly in ICT security.

According to Artz and Gil, trust research is organised in four main areas (Artz and Gil, 2007):

- *Policy-based trust* where trust is used within the use of policies in order to manage and exchange credentials mostly used to enforce access policies. In this scenario, a third-party trusted entity is usually used to help others in creating trust relations;
- *Reputation-based trust* where trust based on past interactions with an entity is used to assess future interactions. In this case, the history of an entity regarding past actions and behaviour is used to compute trust. It is also possible in this type of scenario to use third-party information (for instance recommendations or the trust others may have in an entity) to compute trust;
- *General models of trust* where researchers pretend to model all the environment of trust, for instance, defining trust, the existent prerequisites, conditions and consequences;
- *Trust in information resources* that is mostly focused in Web related research in particular in areas such as Semantic Web.

Although Artz and Gil has organised the actual research in these four areas, it is difficult to establish boundaries among them, as most of the work aiming to use trust to help solve a specific problem usually requires contributions from more than one area.

As stated, ICT areas have a growing interest in Trust and Reputation approaches, in particular, current research is focusing on the development or refinement of trust models. Trust models are usually developed for a specific application area like commerce web sites (e.g. Ebay, Amazon) or more generally for the use in distributed environments where transactions occur between persons or computer systems. In actual fact, the existent trust models tend to be very heterogeneous, possibly due to the use of different definitions of trust that support the models making them context-dependent. Latest research efforts are starting to focus on developing general models to overcome the shortcoming of context-dependent approaches. For instance, Moyano et al. presents a framework intended to help developers to implement applications that might require information from Trust and Reputation Models (Moyano et al., 2012).

From the past couple years, most of the research work in this area is focused on P2P systems (Chen et al., 2009a,b; Spitz and Tuchelmann, 2009), Mobile Ad-hoc Networks (MANETs) (Li et al., 2013, 2012b), Vehicular Ad-hoc Networks (VANETs) (Gómez Mármol and Martínez Pérez, 2012), Wireless Sensor Networks (WSNs) (Ganeriwal et al., 2008; Momani et al., 2008; Zahariadis et al., 2008), on-line personal interactions, software agents and in evaluating generic models and formalisms for Trust and Reputation systems (Jøsang et al., 2007; Malik and Bouguettaya, 2009; Ray et al., 2009; Sabater and Sierra, 2005). Each of the models proposed has its own way of evaluating trust but the majority makes use of statistical approaches.

The major differences that one may encounter among the available trust models are related to the type of information and the information sources used to evaluate trust. The most common information sources are *direct experience* that reflects the experience that one entity has in the relation with another, thus reflecting this experience in a trust indicator. *Witness observation* which uses one or many third-party opinions to evaluate trust. *Certified reputation* that consists of the use of certified references disclosed by third-party entities. *Role-based trust* that analyses predefined role-based relationships between two entities to infer trust. This seems to be the major issue in developing a kind of standard model for trust evaluation. One aspect that can be inferred from the literature review is that it is commonly agreed that, for now, existent Trust and Reputation Models are still context sensitive and thus difficult to apply directly in multiple scenarios (Noorian and Ulieru, 2010).

While most of the previous work integrates observed trust with reputation information received from a third-party entity, this thesis will focus on building trust information based on observed comportments and deriving trust from evidences directly related to the entity whose trustworthiness is being evaluated. In this context, most of the work reviewed evaluates trust using basically the amount of positive or negative transaction experiences (Hussain et al., 2007; Jøsang et al., 2007; Ray et al., 2009). Although it is possible to adapt these models in order to enable that observed events use a value in a defined range (e.g. [0..100]) for each transaction.

Some existing models provide only a single value for trust. This value can be binary (trustee or non trustee) or can also be represented by more than two discrete values using either discrete or continuous numbers or labels. For this thesis it is considered that a trust model should at least give the user a value of trust in a defined discrete range. Another aspect related to trust models of particular importance when it is necessary to take decisions, is that they should provide measures to express uncertainty, reliability or confidence associated with a trust value. Some authors propose models that are able to express uncertainty (Huynh et al., 2006; Jøsang and Ismail, 2002; Teacy et al., 2006).

Trust may be quantified and computed in numerous ways. In particular, several methods are proposed to derive trust from the collected evidence (Sabater and Sierra, 2005). Some authors propose simple probability use (Aime and Lioy, 2005), Fuzzy approaches (Ludwig et al., 2009) or the use of Bayesian Networks (Momani et al., 2008). There are substantial differences among proposed methods. These differences are related to the information used to evaluate trust, use of reputation information, the use of ageing parameters on observed values, use of inactivity periods (Spitz and Tuchelmann, 2009), among others.

In this thesis the approaches described by both Aime and Lioy and Spitz and Tuchelmann are followed regarding the model used to evaluate trust from past experience and the use of a statistical approach to evaluate trust values (Aime and Lioy, 2005; Spitz and Tuchelmann, 2009). Since trust is evaluated as a simple probability and according to the trust definition presented above, it is possible to infer that a trust value expresses the probability that an entity will behave as expected.

2.7 Summary

The need to protect Critical Infrastructures has motivated the development or the adjustment of several frameworks, techniques and mechanisms in order to increase the resilience of such important infrastructure on which actual society depends. This section first introduced the concept of Critical Infrastructure and the problems that might arise in their operation. As the interdependencies existent among CIs pose a serious risk in the CIs operation, this problem was discussed while describing some relevant approaches that deal with this subject. Critical Infrastructure modelling, simulation techniques and risk assessment frameworks have been addressed in order to understand their main characteristics and the major problems they pretend to solve. The CI Security Model is detailed in this chapter as an introduction to Chapter 5 in which this model is used to validate the work.

The relevance of the area is also highlighted by describing three selected European projects that had major contributions in the area of CI security and protection.

More specific aspects are discussed in this chapter in order to locate the contributions presented in this thesis in the state of the art. In particular, some existent work on Policy Based Management and methodologies to deal with Ontologies were discussed.

An overview on Trust and Reputation Models focusing on their applicability to the context of Critical Infrastructures protection and on the information exchange among Critical Infrastructures concludes the chapter.

The following chapter describes the MICIE FP7 project with particular attention to contributions achieved by the author of this thesis.

Chapter 3

MICIE FP7 Project

In this chapter, the Tool for systemic risk analysis and secure mediation of data exchanged across linked CI information infrastructures (MICIE) FP7-ICT project (MICIE Consortium, 2008) is described, detailing the developed alerting system, the MICIE Secure Mediation Gateway (SMGW) and the adopted solutions to specifically incorporate CI interdependencies into the online risk assessment framework.

Most of the research challenges that led to this thesis had their genesis in the scope of the MICIE project. Although the author of this thesis has been involved in the most part of the project, it is possible to highlight his involvement in the work related to the MICIE work package 4000 (WP4000) - Mediation System Design - in which he participated actively on the publication of all the deliverables issued as a result of the work package.

In particular, during the execution of the WP4000, the author of this thesis played an active role during the drafting and development of the Security Mediation Gateway (SMGW). Besides the global participation, the author of this thesis proposed and implemented a Policy Based Management framework for the SMGW, contributed to the definition of the Ontology used in the MICIE data repository and proposed a framework for Trust and Reputation Management, implemented as an add-on security mechanism allowing to improve the security of the information exchange among Critical Infrastructures, namely, the Trust and Reputation System (as described in Chapter 4) to be applied to the MICIE Security Mediation Gateway.

The major achievements that resulted from this work are described in the following publications: (Castrucci et al., 2009), (Castrucci et al., 2010a), (Inzerilli et al., 2009), (Castrucci et al., 2010b), (Caldeira et al., 2010d), (Caldeira et al., 2010a), (Caldeira et al., 2010c) and also in (Castrucci et al., 2012), among others.

This chapter is structured as follows: Section 3.1 presents an overview about the MICIE project while Section 3.2 describes the MICIE system's overall architecture. Sections 3.3 and 3.4 respectively describe how the MICIE project handled CI modelling and risk prediction activities.

Section 3.5 presents the MICIE Secure Mediation Gateway architecture from which the SMGW Management is described in detail in Section 3.6. The validation activities carried out within the MICIE project are briefly described in Section 3.7. To conclude, Section 3.8 presents a summary of the main issues discussed throughout the chapter.

3.1 MICIE Project overview

In accordance with the discussion found in the previous chapters, it is now commonly agreed that Critical Infrastructures are one of the areas of contemporary societies where it is vital to ensure the application of the highest security levels. In this context, the European Commission launched the EPCIP whose main goal is to improve the protection of CIs in the European Union (European Commission, 2006) and where one primary objective is also defined, the implementation of the Critical Infrastructure Warning Information Network (CIWIN). CIWIN main objective is to provide a platform for the exchange of rapid alerts among CIs in order to help European Member States and CI Operators to share information on common threats and vulnerabilities. The idea is quite ambitious, requiring the support of players in the market, which need to have confidence in the proposed system in order to participate.

As seen, the recent efforts to improve security and protection in Critical Infrastructures, are mostly focusing on each CI individually, launching the foundations for more secure CIs with enhanced robustness, security and resilience. Introducing, for instance, fault-tolerant architectures, redundant components and more resilient ICT systems. An important aspect that needed to be addressed relates to the interdependency existent among CIs. This interdependency can lead, in an extreme situation, to a global failure in an undefined number of CIs, started by a single trivial incident in one CI (cascading effect).

Although the large resources that are being allocated on CI modelling, CI (inter)dependency modelling studies and on evaluating CI risk information from those models, most of this valuable information gathered from the CI models is still kept and only used inside each CI, being not regularly shared among interdependent CIs.

One problem that has been identified within the presented context is the lack of sharing mechanisms able to exchange risk information among interconnected CIs. This sort of mechanism aims to allow CI Operators to have a real time view on the risk level associated with services on which the modern society depends, such as power, water supply, or communication lines. Sharing this type of information is also important to increase the accuracy of the CI risk models, by introducing risk information related to external failures on these models (Simões et al., 2010). Another problem that was identified is to what extent, CI Owners, are willing to exchange this sensitive information without disclosing CI details that could endanger their activities.

MICIE project suggests that the use of mechanisms able to share CIs risk information can allow, besides more resilient CIs, increase the security level of multiple (inter)dependent CIs. To achieve improved service levels, a robust, resilient and inter-dependencies-aware alerting system was designed and implemented. This was the main goal of MICIE (Tool for systemic risk analysis and secure mediation of data exchanged across linked CI information infrastructurees) FP7-ICT project, aiming the design and implementation of a real-time risk level dissemination and alerting system (MICIE Consortium, 2008).

Figure 3.1 represents a top level view of the MICIE system. In this Figure, four CIs operating the system are represented. It is notable the existence of CIs that depend on services provided by partner CI (dependency - one CI depends on one or more services provided by another CI) and also CIs that are interdependent (CIs dependent on services provided by each other). For instance, CI A depends on services provided by CI C and provide s services to CI B. CI B depends on services provided by CI A and CI B. CI C only depends on services provided by CI B. One independent CI using the MICIE system (CI D) is also represented in Figure 3.1, highlighting the fact that this system can also be used in a independent manner, meaning that only local risk prediction is evaluated and used without information being exchanged.

A distributed on-line Prediction Tool (PT) supported by the defined abstract CI model is continuously evaluating the risk level indicators. The CI models are kept

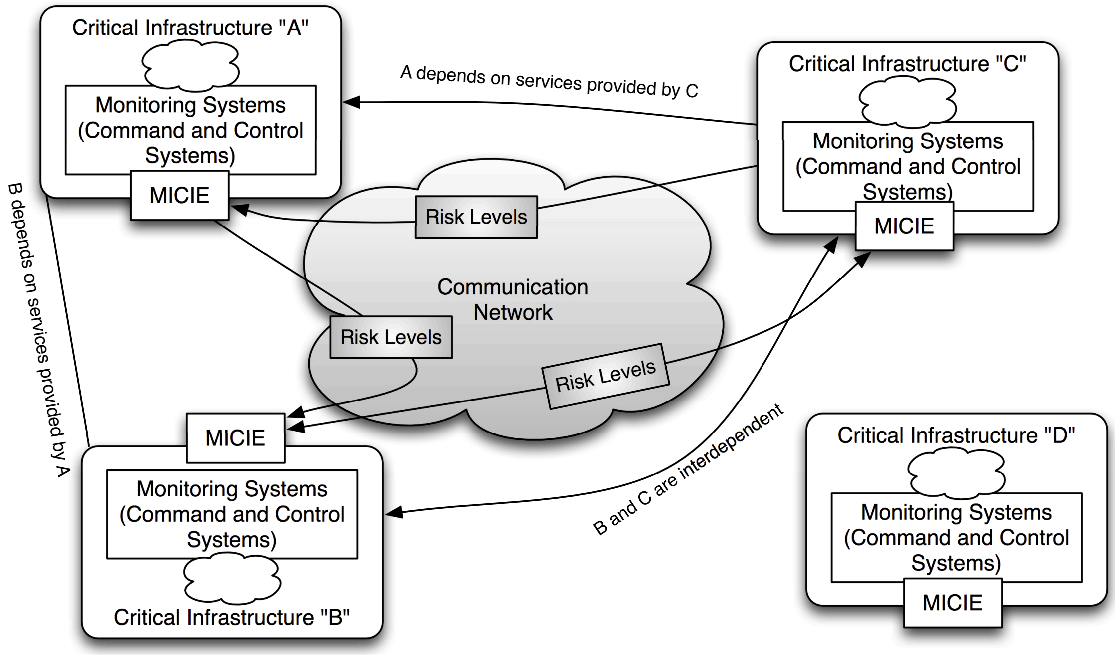

Critical Infrastructure "A"
Monitoring Systems (Command and Control Systems)
MICIE
A depends on services provided by C
Critical Infrastructure "C"
Monitoring Systems (Command and Control Systems)
MICIE
Risk Levels
Communication Network
Risk Levels
Risk Levels
B depends on services provided by A
MICIE
Monitoring Systems (Command and Control Systems)
Critical Infrastructure "B"
B and C are interdependent
Critical Infrastructure "D"
Monitoring Systems (Command and Control Systems)
MICIE

Figure 3.1: MICIE system overview.

constantly updated by receiving information coming from the field of each CI (aggregated metadata that proper describe the CI status). As represented in Figure 3.1 by the existing independent CI, the defined models are also able to be used *off-line* allowing the evaluation of risk levels considering only the information gathered inside the CI and also, to comprehend the level of (inter)dependency existent among CIs. From this information it would be possible to improve the characterisation of the most vulnerable elements existent in the (inter)dependent systems.

MICIE alerting system includes an appropriate communication infrastructure, namely the Secure Mediation Gateway, which provides secure communication across the MICIE system. The SMGW is designed to retrieve, from each CI, all the information required for the real-time risk prediction. Additionally, the system implements the information sharing mechanisms according to a highly available and secure framework (Figure 3.2) (Capodieci et al., 2010).

The MICIE project was able to test the achieved results in the field, with the contribution of the Israel Electric Corporation (IEC), that, among other contributions, provided all the knowledge gathered from a portion of the electrical and telecommunication infrastructures of Israel, both managed by the Israel Electric Corporation. This knowledge allowed the MICIE consortium to use a part of two Critical Infrastructures (energy and communications) as well as the IEC ICT infrastructure as a test-bed for the MICIE on-line alerting system.

3.2 MICIE overall system architecture

The MICIE system is a distributed environment composed by multiple heterogeneous CIs that might depend on one or more services provided externally by other CIs. Considering that the CIs are willing to cooperate in order to improve the provided quality-of-service, the MICIE system introduced mechanisms allowing CIs to be able to predict and exchange risk information across trusted or untrusted network infrastructures (e.g. Internet).

The information exchange among CIs is critical for the MICIE system as it supports the risk analysis and prediction performed inside each CI by taking into account the existent CI (inter)dependencies. It is clear that this sensitive information must be kept within the system in a secure manner. It is commonly known that solely disclosing this type of information is considered a high security risk for all involved CIs. The information exchange is achieved through a MICIE component, the local Secure Mediation Gateways (SMGW), able to provide secure communication channels across the MICIE system.

Figure 3.2 represents the MICIE's system overall architecture. The Figure allows to distinguish the main, following entities that compose the MICIE system (Caldeira et al., 2010a; Castrucci et al., 2012):

- Critical Infrastructure: the infrastructure from which the MICIE system predicts the risk. Multiple CIs might participate in the MICIE system even if they are heterogeneous and even if they are situated in remote locations.
- Prediction Tool (PT): the entity responsible for undertaking risk prediction within a CIs. Each CI has at least one local Prediction Tool. However, in order to achieve all the project benefits, it requires, in addition to local information, information related to remote (inter)dependent CIs.
- CI Monitoring System: the local framework able to perform monitoring activities within an infrastructure. This system is able to detect failures, degradation of QoS, among others. As it is assumed that the participant CIs are or can be heterogeneous, each CI can have its own specific monitoring system. Due to the fact that monitoring systems are closely related to the CI physical

components, it is assumed that this component is a legacy system completely decoupled from the MICIE system.

- Adaptor: the entity employed for interconnecting each CI's particular monitoring system with the MICIE system. It is able to connect to the CI's monitoring system collecting information from it and providing the necessary translation to a common data representation format. It also performs operations such as filtering, aggregation, translation of information with the main goal to provide the MICIE system with all the information needed to accomplish risk prediction.
- Secure Mediation Gateway (SMGW): the entity that provides to the PT, the information needed for risk prediction. It is able to gather and compile local information retrieved by the adaptor(s) and to receive remote information from peer SMGWs. It has also the role of providing the necessary information to remote peer SMGWs in order to assist them with performing their functionalities with proper knowledge. Since the information treated by the SMGW is sensitive, it fulfils a number of security requirements including also the secure communication with remote peer SMGWs.

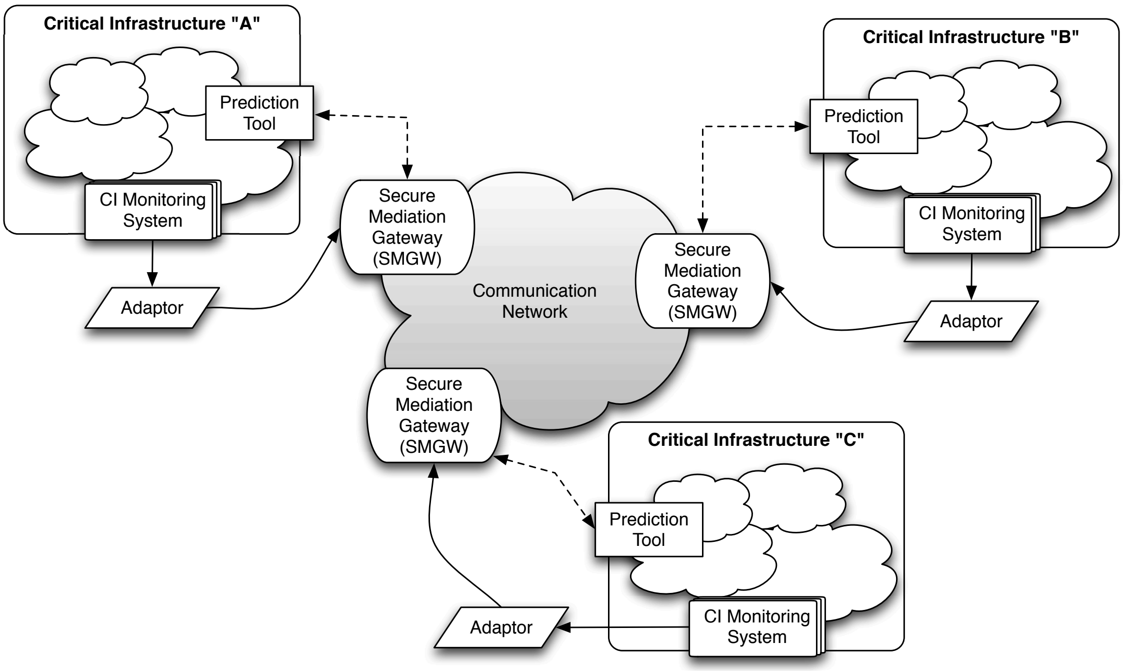

Critical Infrastructure "A"
Prediction Tool
CI Monitoring System
Adaptor
Secure Mediation Gateway (SMGW)
Communication Network
Critical Infrastructure "B"
Prediction Tool
CI Monitoring System
Secure Mediation Gateway (SMGW)
Adaptor
Secure Mediation Gateway (SMGW)
Critical Infrastructure "C"
Prediction Tool
CI Monitoring System
Adaptor

Figure 3.2: MICIE overall system architecture.

In order to design and implement the complete MICIE system a reference scenario has been defined, with the stakeholder accordance, and modelled including the discovered (inter)dependencies. Three new functional modules were fully designed and implemented, namely the Adaptor, the SMGW and the PT. This section will focus more on the SMGW as it was the area to which the author of this thesis most contributed. Detailed information on the adaptors design and implementation can be encountered in (Castrucci et al., 2010b). The MICIE reference scenario, the applied interdependency modelling framework, indicators and models were part of MICIE Work package 2000 (Bertoni et al., 2010a,b; Ciancamerla et al., 2009, 2010b; Lev et al., 2009). The Prediction Tool is detailed in deliverables produced within MICIE Work Package 3000 (Panzieri et al., 2009, 2010; Simões et al., 2010, 2009), where also the risk prediction algorithm is detailed.

3.3 MICIE - Critical Infrastructure Modelling

One of the main problems that the MICIE project had to overcome was related to Critical Infrastructures modelling. Despite the existence of several proposals for CI modelling this appeared to be one of the most difficult tasks to complete. The difficulty was mainly due to the usual complexity and size that such an infrastructure entails. In order to build a representative and realistic model for the MICIE project, a reference scenario has been defined within the project consortium supported by the expertise of the stakeholder, the Israel Electric Corporation. The use of a reference scenario was a fundamental decision thus narrowing down what should be in the model, and providing a concrete context of operation, focused on CI (inter)dependencies.

A reference scenario should identify the services provided by the CI: the events or sequences of events that are adverse and that could cause a relevant impact in the quality of the identified services (for instance in terms of continuity or performances); the interconnected networks that support the services (e.g. topologies, essential systems, etc.); the interconnections existent among networks and systems; the procedures defined by the Operator regarding the implementation and maintenance of each service (Ciancamerla et al., 2010a).

The reference scenario is composed of the ICT systems and two distinct CIs (energy and telecommunications). It has been established from one set of the Israel Electric Company infrastructures, systems and their interconnections. The main components of the reference scenario are (Ciancamerla et al., 2009, 2010b; Simões et al., 2010):

- a section of an electricity distribution network, including a Medium Voltage (MV) power grid at 22 KV and a High Voltage (HV) power grid at 160 KV;
- a supervisory control and data acquisition (SCADA) system that remotely monitors and controls the power distribution grid and Remote Terminal Units for remote operations;
- a section of the IEC telecommunications network with fiber optics and radio links used mainly to control the electricity distribution network;
- the interconnection of SCADA with the portion of IEC telecommunications transmission network;
- the ICT infrastructure.

A top level view of the reference scenario is represented in Figure 3.3. On the left side of the Figure, some examples of possible adverse events, that can eventually occur and cause some impact in the normal operation of the services provided by the CIs, are represented. The represented adverse events can be triggered due to multiple causes, namely, natural causes, malicious attacks or simply due the malfunction of some equipment. A simple event can affect just one of the represented CIs or services, for instance the electrical CI or the telecommunications CI, and, as both of the represented CIs are interdependent, the effect triggered by such a small event may propagate and reach the customers of the medium voltage grid causing, for instance, the interruption of power supply.

MICIE models were evaluated and tested based on multiple heterogeneous models (stochastic versus deterministic, agent based, dynamic simulation, etc.) with the main objective of evaluating a short-term estimate for the Quality-of-Service supplied by the different Critical Infrastructures. The models were created based on the underlying interconnected networks that collaborate for service delivery and, as represented in Figure 3.3, according to multiple possible adverse events (for example, attacks to critical elements, sequences of characteristic failures and congestions or failure in communication networks).

One major aspect that has been addressed is the impact that the potential degradation of the QoS on the SCADA system (for instance, service connectivity, reliability, rerouting, response time, operability level) might have on the quality of the power supply provided by the power grid Operator to power grid customers (measured in terms of duration and number of interruptions). In this scenario, the Power Grid Fault Isolation and System Restoration Service (FISR), performed by SCADA

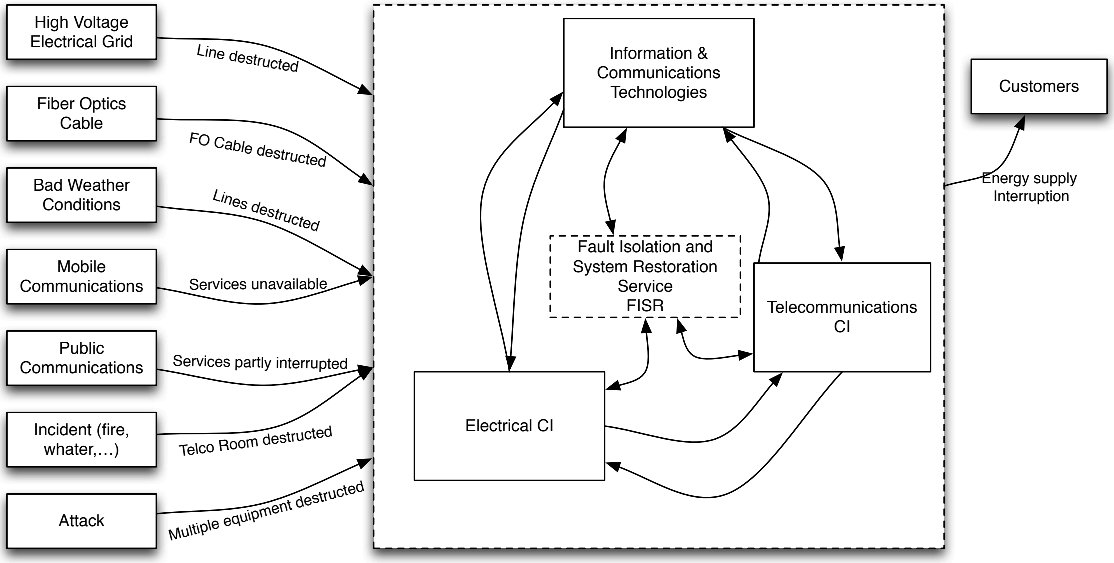

High Voltage Electrical Grid
Fiber Optics Cable
Bad Weather Conditions
Mobile Communications
Public Communications
Incident (fire, whater,...)
Attack
Line destructed
FO Cable destructed
Lines destructed
Services unavailable
Services partly interrupted
Telco Room destructed
Multiple equipment destructed
Information & Communications Technologies
Fault Isolation and System Restoration Service FISR
Telecommunications CI
Electrical CI
Customers
Energy supply Interruption

Figure 3.3: High level view of the reference scenario (adapted from Ciancamerla et al. (2010a)).

through its Operator, is also considered a particularly critical service (represented in Figure 3.3) as FISR aims to detect and isolate grid outages, and to restore the grid in order to provide service continuity to the customers (Capodieci et al., 2010). For example, the failure of the FISR service caused, for instance, by a telecommunications failure, incapacitate the Operator to take rapid remote actions while trying to restructure the grid.

Capodieci et al. describes that the MICIE project was able to identify the interconnected networks that support the FISR service, namely, the SCADA system, the telecommunication network and the power distribution grid. This identification was made in terms of topologies, functionalities, performances, rerouting and failure behaviours, interconnections at physical, geographical and logical layers (Capodieci et al., 2010). Different techniques and tools are used to represent the reference scenario. Bertoni et al. describes in detail the application of such techniques and tools within the MICIE project (Bertoni et al., 2010a).

In this context, some QoS indicators depend upon failure and behaviour in case of necessary repair of network elements. Examples of such QoS indicators are connectivity and availability. According to (Capodieci, 2011), these indicators are computed using analytical methods and from the integration of the different topologies using the Mixed Holistic Reductionist (MHR) approach.

Detailed information presenting the results achieved by MICIE, regarding CI modelling activities integrated in the work package "Interdependency Analysis and Modelling", is available in the project documentation (Bertoni et al., 2010a). It is described how the modelling efforts were able to, completely or partially, represent the reference scenario defined for the project.

The MICIE CI modelling approach also considered the ICT security analysis for each CI. The core critical assets were identified and their level of potential vulnerability has been evaluated also considering the information they are associated with. The security risk is estimated by judging the impact the core critical assets may have in a single CI and also the impact that they may cause on (inter)dependent CIs. In such a complex system as a Critical Infrastructure, it is also essential to establish a tolerable level of risk and to select those risks on which it will be necessary to act upon and control.

The results from the ICT security analysis that are integrated into the MICIE Alerting System, can be summarily enumerated as a set of security requisites and respective related risks, as first described in (Lev et al., 2009) and further detailed in (Bertoni et al., 2010b) and (Bertoni et al., 2010a):

- Reliability and integrity of information which feeds the MICIE alerting system;
- Security risks introduced by the CI ICT network and impact of those risks on the MICIE alerting System;
- Interdependency model resilience;
- Security risk analysis, for the risks related with interdependency modelling;
- Non-repudiation of each peer.

The author of this thesis contributed to the work package responsible for modelling activities tasks (WP2000 - Interdependency Analysis and Modelling), by proposing solutions for modelling ICT security attacks. In particular the main idea is to evaluate and use the CI ICT security risks as one of the information sources of the general CI model. In this context, some ICT Monitoring Components (ICT-MCs) were proposed to be added to the MICIE framework (Lev et al., 2009). ICT-MC includes a Monitoring System and a set of Intrusion Detection Systems covering the corporate ICT network and, when possible, the SCADA network. It is mostly focused on CI ICT network monitoring (especially for intrusion detection, but also for detection of other anomalies). This component was the author's first contribution to the project and has evolved to the Managing System and to the Trust and Reputation System that are described in further detail throughout this thesis.

Modelling ICT security inside a Critical Infrastructure is a demanding and complex task mainly due to the size and complexity of the ICT networks and computer systems implemented. Due to the complexity of the model that can be gathered from a CI, the author of this thesis proposed a sort of “black box” model supported by Bayesian Networks on which the main model is divided into several sub-models, each one regarding an area of operation. This approach aims to simplify the task of gathering the information required to build the Bayesian Network (in particular the quantitative information).

This system – designated as Information and Communication Technology Monitoring Component allows to gather information from multiple Bayesian Networks, called “Experts”, used to feed a top level Bayesian Network able, to infer using multiple “Experts” information. In this scenario, each “Expert” can also make use of Bayesian Learning. Figure 3.4 represents an example of a Bayesian Network for the ICT-MC.

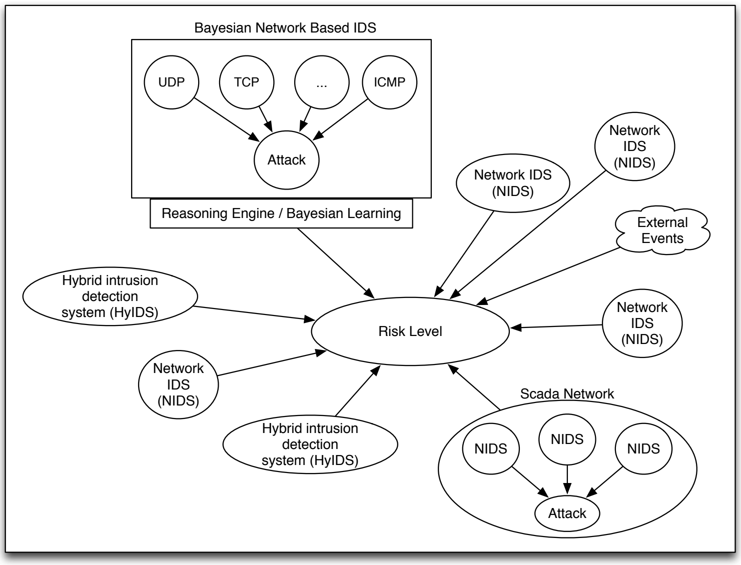


Figure 3.4: ICT-MC Bayesian network example (Lev et al., 2009)

According to the proposed approach it is possible to have a set of IDSs in the corporate network and also in the SCADA network. The ICT-MC system is meant to work in an autonomous way while detecting intrusion anomalies and also, for instance, detecting abuse on the CI defined security policy (for instance an access control policy).

The information generated on the ICT-MC system can be aggregated into the MICIE system along with other types of CI monitoring information (e.g. SCADA systems and legacy systems) thus helping to integrate ICT security into the MICIE CI model (Figure 3.5). Furthermore it is possible to use the ICT-MC as an independent system inside each CI in order to provide the network administrator with valuable information concerning ICT security.

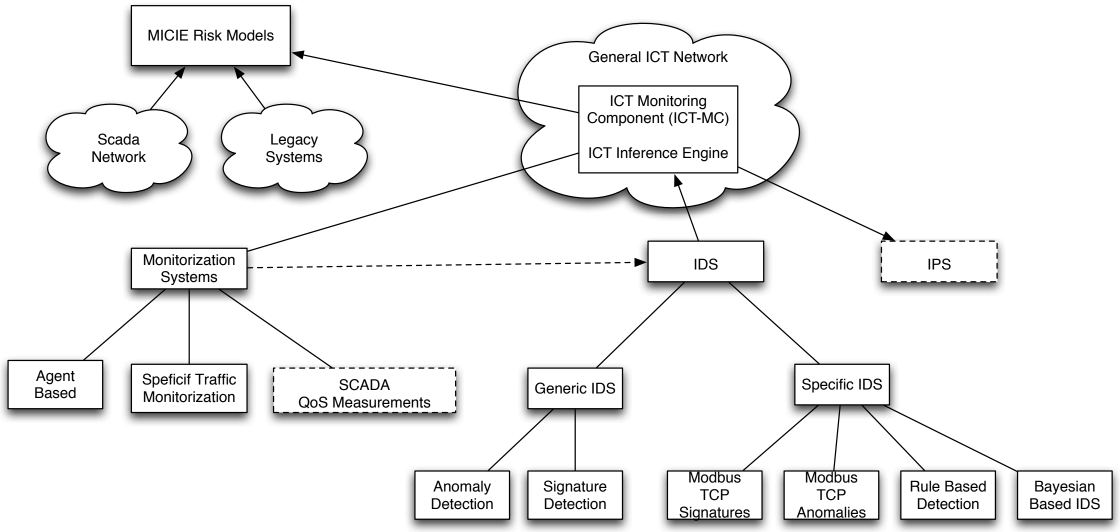


Figure 3.5: ICT Monitoring Component (ICT-MC) overview.

Even though the ICT-MC component hasn't been integrated in the final implementation of the MICIE project, this contribution led to several important discussions within the MICIE Consortium about security and CI modelling. Among others, these fruitful discussions raised multiple concerns that had not been covered by the MICIE project. Some of these concerns are now being studied and developed under a new European project - Project CockpitCI (Cybersecurity on SCADA: risk prediction, analysis and reaction tools for Critical Infrastructures), which aims to improve the resilience and dependability of Critical Infrastructures through the automatic detection of cyber-threats and the sharing of real-time information about attacks among CI Owners (CockpitCI, 2013). This project is also making use of some MICIE results and, as stated, is focusing in the detection of cyber-threats and attacks that can affect CIs.

3.4 Online Risk Prediction Tool

The design of the Prediction Tool, and the inference rules used to calculate the risk prediction from the information about the status of each CI, including the interdependent CIs, are just briefly presented in this section. More information regarding the results achieved in this area is available from the MICIE work package 3000 – Risk prediction system design documentation and from several publications describing in detail this subject (Ciancamerla et al., 2010a; De Porcellinis et al., 2009; Gasparri et al., 2009; Oliva et al., 2010; Panzieri et al., 2010; Simões et al., 2010).

One problems one must solve while defining a centralised state estimator is related to the fact that complete knowledge about the status of every infrastructure and their components must be achieved. From the discussion above, it is clear that such a prerequisite is not easily satisfied, mainly due to the vast amount of data that has to be considered and because of the security aspects that can arise in case of disclosure of this critical information (Capodieci et al., 2010).

Capodieci et al. refers to the application of a more realistic approach based in a decentralised scenario able to synchronise with external scenarios. In this approach, each control centre should include a global model representing all existent systems (Figure 3.6) (Capodieci et al., 2010) . Each infrastructure has a tool that receives information originated from inside the infrastructure. As the different tools must be interconnected in order to achieve a global prediction, it is important to maintain them synchronised. An example of how to achieve model synchronisation has been presented by Gasparri et al. for the case of linear distributed interdependency estimators with complete information sharing (Gasparri et al., 2009).

According to Capodieci et al., the easiest method that allows to maintain the consistency of the overall state, estimated by the independent dispersed tool, is to use a common general model in every system, although each specific system just needs to receive a specific subset of the inputs available (Capodieci et al., 2010).

On the proposed framework for the MICIE project, all of the existent prediction tools have the same overall model. The Mixed Holistic Reductionist approach has been adopted along with the CISIA (De Porcellinis et al., 2008) simulation framework. CISIA allows to manage multiple heterogeneous models into a single framework, with the desired level of granularity (Capodieci et al., 2010).

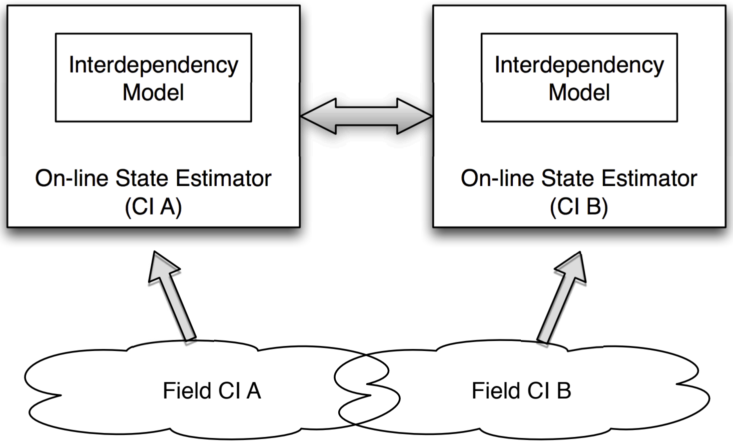


Figure 3.6: Decentralized risk prediction tool (Capodieci et al., 2010).

The Prediction Tool design and the inference rules, used to evaluate the risk prediction based on the gathered information about the status of interdependent CIs, is out of the scope of this thesis. Nevertheless the author of this thesis contributed to this work package focusing on the use of ontologies to represent the model's metadata considering also the planned exchange of this metadata among multiple CIs.

In order to allow metadata exchange among CIs, in a simple approach, it is possible to identify two main entities that must be described using ontologies. These entities are Remote SMGWs and MICIE Risk Alerts. Remote SMGWs must be represented due to the fact that it is mandatory to maintain information about the system partners, mainly for security reasons. MICIE Risk Alerts represent the information that is to be exchanged among CIs as the result of the risk prediction algorithms. Figure 3.7 presents a simplified view for the proposed ontology (Panzieri et al., 2010; Simões et al., 2009).

The presented approach for data representation has been applied in the MICIE project supporting the existent MICIE data and metadata database (Panzieri et al., 2010; Simões et al., 2009).

3.5 MICIE Secure Mediation Gateway

The identification and modelling of interdependencies can be very useful in order to limit the effects of a failure in a CI and even to prevent cascading effects. In particular, if a CI Operator has the opportunity to be informed of the status of the existing interdependencies, he can then evaluate predictions on the status of the

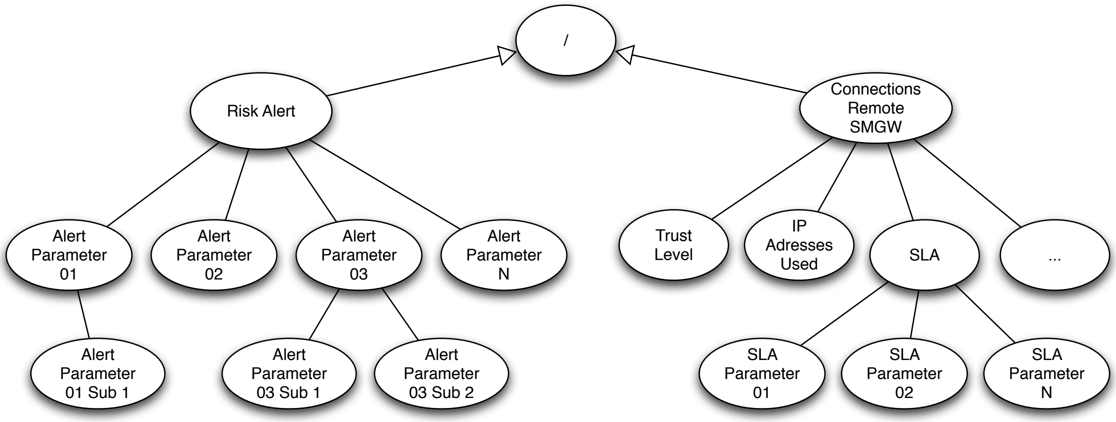


Figure 3.7: Basic ontology for the MICIE data repository (Simões et al., 2009)

delivered QoS level of its services. He also can undertake specific actions in order to prevent the failure of the CI if failures occur in interdependent CIs. To reach this goal, the need of a communication system interconnecting different CIs, was identified and defined as the Secure Mediation Gateway (SMGW).

The MICIE SMGW is the key element of the existent communication infrastructure that is composed by a set of SMGWs (one for each CI in the system).

The MICIE SMGW architecture, represented in Figure 3.8, is able to interact with four main entities through the following interfaces (Caldeira et al., 2010a; Castrucci et al., 2010b, 2012):

- to the local CI monitoring system through the SMGW-Adaptor Standard interface;
- to the local Prediction Tool through the SMGW-PT interface;
- to other remote SMGWs through the SMGW-SMGW interface;
- to the system administrator through the SMGW control interface.

Within the MICIE system, the main tasks performed by the SMGW are briefly described as: (i) collecting information about the local CI (i.e. the CI where the SMGW is located); (ii) retrieving information about the other interdependent CIs in the system; (iii) sending information about the local CI to remote CIs; (iv) providing all the collected information to the Prediction Tool.

Figure 3.9 illustrates how the MICIE system can be interfaced with the CI where the SMGW is located.

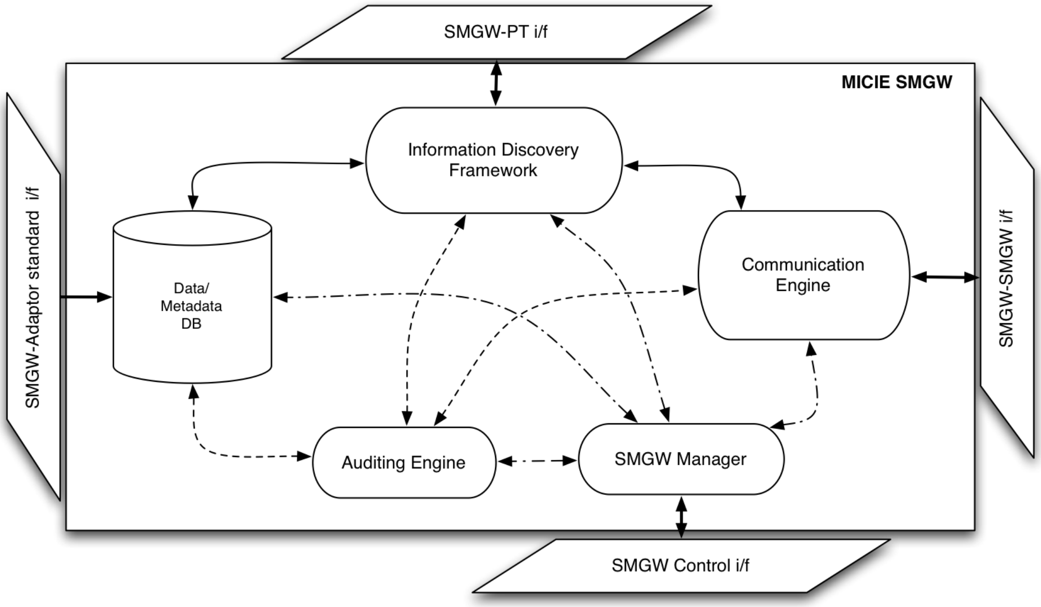


Figure 3.8: SMGW Architecture (Castrucci et al., 2012).

In order to implement the main SMGW functionalities proposed in the architecture, five independent entities were developed, namely, the *Data/Metadata Database*, the *Information Discovery Framework*, the *Communication Engine*, the *SMGW Manager* and the *Auditing Engine* (Caldeira et al., 2010a; Castrucci et al., 2010b, 2012). In the following, each existent entity is described. The *SMGW Manager* will be described in a independent section (Section 3.6) in order to highlight the contributions proposed and implemented by the author of this thesis.

Data/Metadata DB

This entity acts as the overall MICIE information database and an instance must exist in each participant CI. The information stored in the database includes local information aggregated by the Prediction Tool and raw information retrieved by the local adaptors from the field. The stored metadata also includes information able to determine which information aggregated by the Prediction Tool can be exchanged or made available for other remote Prediction Tools. This metadata should include confidentiality as well usability requirements in order to allow the peer PTs to work properly.

Existing PTs are able to exchange the information they need to use in order to perform the risk prediction. With access granted to all available information (with

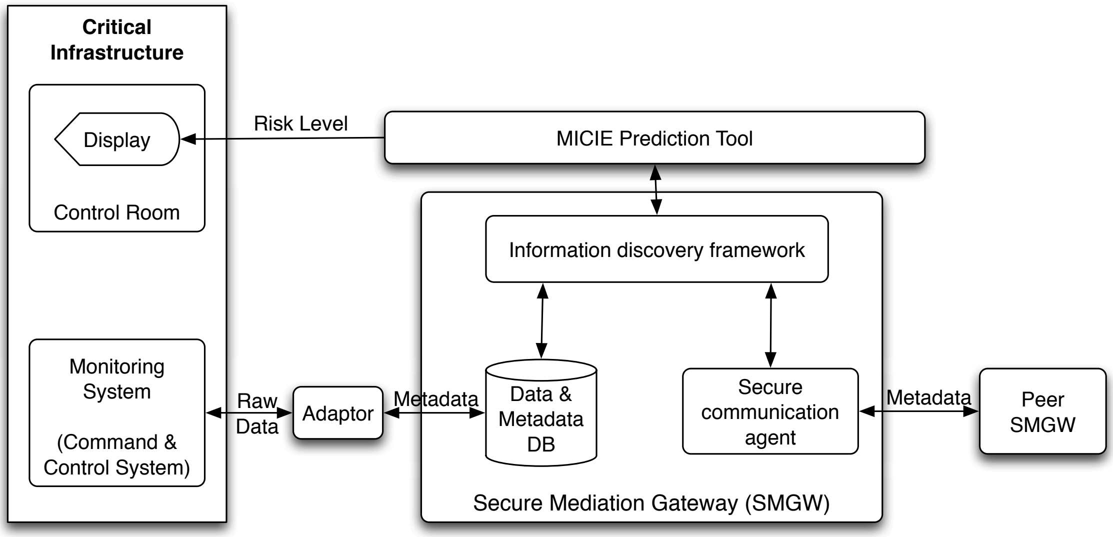


Figure 3.9: MICIE system in use on a CI (Capodieci et al., 2010)

restrictions ensured by defined confidentiality aspects), each PT is able to evaluate the future state of the system and provide such information to its peers CIs, to the CI Operators and to the Stakeholders.

One problem that arouse is the fact that, typically, the participant CIs are heterogeneous. The existence of multiple types of monitoring systems collecting a wide range of heterogeneous information implies the application of an interoperation strategy, that allows the system to work properly thus allowing the on-line PTs to perform risk prediction.

The implementation of the interoperation strategy was the introduction of a common semantic for the shared information. A common semantic is fundamental in order to guarantee a successful interoperation among different CIs. It allows the information to be automatically processed by a distributed system formed by independent autonomous systems that are able to communicate with each other without ambiguity.

Heterogeneous raw data collected by the different monitoring systems is described using the same ontology that represents the MICIE information format (using a standard format described by using ontology language (OWL) (Castrucci et al., 2010b). The component responsible for making the information adaptation was implemented, named the Adaptor. Basically, the adaptation is achieved by a translation from the CI particular raw data format, to the specified data/metadata ontological format.

In order to enforce privacy constraints on data stored in the database, the disclosure level for data exchange is defined depending on the information contained in that data. The information stored in the SMGW is organised and partitioned into subsets. Each subset manually associated to a specific disclosure level, defined on the basis of rules and policies decided by the CI Operator, that determines which SMGW can have access to specific data (Caldeira et al., 2010a; Castrucci et al., 2012).

Information sharing framework

The Prediction Tool component requires actual information gathered from the CI. The gathering process if achieved with the discovery process that performs all functionalities related to the discovery and composition of information. The discovery process works together with the composition process in order to combine information arriving from interdependent CIs. The composition allow to enhance the monitoring capabilities of the existent PT deployed in each local CI. During the process, the available local information is combined with remote information provided by interdependent CIs using the defined ontology. It is also possible to perform semantic inference on the newly created ontology. Local ontologies in the SMGWs follow a composition process in order to discover new relationships among entities linked in the ontology.

The SMGW is able to provide the PT with all the information for which the PT subscribed. To achieve this goal the SMGW needs to be able to discover it within the MICIE system. This process includes searching in the internal resources (local storage system and databases) and also in remote resources (interdependent CI).

The SMGW must request permission from the system manager in order to search internally to see if the information is available. In order to search for information across the MICIE system, the local SMGW cooperates with external SMGWs allowing it to discover needed information. Once discovered, a pointer linking to the information is sent to the requester. The prediction capabilities of a single PT are enhanced by the use of combined information that had been discovered across multiple CIs. The discovery process can occur on-demand when requests are launched from a PT to collect information both from local and remote CIs, for local processing. The process can also occur in trigger mode, during which, the occurrence of specific events and changes on status variables within one CI are automatically propagated to both local and remote PTs across the SMGW protected communication infrastructure (Caldeira et al., 2010a; Castrucci et al., 2012).

Auditing engine

The Auditing Engine performs log management to collect information in order to support forensic analysis regarding the SMGW. A set of operational modes are defined in the SMGW Manager in order to achieve this operation. Actions triggered by specific management policies initiate these operational modes.

When a specific security event is detected, a management policy, able to deal with that particular event, is triggered leading to the execution by the SMGW Manager of relevant policy related to that particular event. The triggered policy can adjust a dedicated auditing engine to work on a specific mode, which is more appropriate to the type of the detected event.

The SMGW Manager has a specific interface that is used to control the auditing engine and to retrieve information stored in the auditing log.

The auditing engine has the following four distinct operational modes that allow forensic analysis activities. (1) The *Normal usage* mode is the default mode and is applied during the regular system usage, providing basic log collection functionalities such as monitoring of processor, memory and network events logging. (2) The *Authentication Process* triggered when an authentication process takes place (e.g. inter-CI communication or CI Operator interventions). It supports forensic analysis in case of attacks focused on the authentication processes (e.g. impersonalization techniques, authentication exploitation attempts, etc.). (3) The *Tampering/Injection Attempts* mode executes in case of detection of any kind of evidence of tampering or injection attacks existent in the communication network. The *Network services subversion* executes upon the detection of anomalies in the network traffic, or when the system resources usage levels reach a predefined value. In this case, forensic analysis regarding network services exploitation is triggered (Castrucci et al., 2010b, 2012).

Communication engine

Information security is one important aspect considered in MICIE. Being a distributed system, the communication security support is one of the main security aspects that need proper attention. The Communication engine should enforce security policies in order to control all the communications from and to the MICIE system.

The information shared among MICIE participants comprises specific data related to each of them (for example status, quality of service, etc.). It is fundamental to consider this data as being extremely sensitive in terms of security, as it's eventual disclosure could affect the operative status of one or multiple CIs. In the case of problems in the communication link between two or more SMGWs, for instance, if the conveyed information is disclosed or modified, the status of one or various CIs could become unreachable or even modified in order to cause damage to the peers. In such a scenario, the alerting system becomes unusable and transforms itself to a new point of failure in the CI security protection system.

The communication security problem is also critical, as MICIE foresees, the communications might also be based on an untrusted network such as the Internet. In such an environment, the MICIE consortium proposed a security framework for the communication system. In this context, a data exchange policy was defined. This policy allows defining rules, considering the type of data flow and data exchange modality used in the MICIE framework. Two methodologies for data flow were defined. The first method allows all MICIE information to be shared among all existent SMGW. This method allows for redundancy while raising serious problems concerning data and communication management, as all participants will receive the same information. The second method only shares relevant information for each neighbour or dependent CI, avoiding transmitting unnecessary information and simplifying the management.

The data exchange is achieved either by broadcasting or by sending information on demand. Due to safety or commercial reasons and the existence of different data non-disclosure policies, the second solution is the better approach. The main security features of data exchange was defined to ensure data availability, integrity and confidentiality, and also non-repudiation and accountability.

To ensure the defined security specifications, a security risk assessment has been made including the dedicated interfaces of two linked SMGWs and the untrusted network link. The risk assessment consisted of listing the threats that can occur in the system and in proposing the security objectives to reach a secure communication (Castrucci et al., 2010b), (Castrucci et al., 2012).

3.6 SMGW Management

In the MICIE project, the SMGW is one of the key element present in the MICIE overall system architecture, as it is the network element that allows the exchange of information among different and heterogeneous CIs.

Information exchanged among different SMGWs is extremely sensitive, as it is related to the Critical Infrastructures, their status, and their services. It is clear that non-authorised third parties should not be able to acquire the information exchanged among CIs and, at the same time, it should not be possible for non-authorized third parties to send information to the SMGWs.

Considering the important role that SMGWs play in the system, the problem of how to manage the system has been addressed. Management strategies were developed that address security aspects while permitting an easy definition of security rules by the system administrators.

Currently, several efforts are made to mitigate the increasing complexity in network management, by using different management paradigms and approaches. The Policy Based Management approach aims to be the result of the change from the actual configuration mechanisms, to an integrated management system. In this context, the development of a Policy Based Management Architecture was proposed, allowing an easy and flexible manner to manage all security and operation aspects related to the SMGW (Caldeira et al., 2010a,c,d; Castrucci et al., 2010a,b, 2012; Ciancamerla et al., 2009; Inzerilli et al., 2009; Lev et al., 2009, 2011, 2010b; Neri, 2010; Panzieri et al., 2010)

SMGW management is able to handle all SMGW administration, including testing and alarming. Also included are the functions of intrusion prevention and intrusion detection. These functions involve the online monitoring of the SMGW operation as well, as the online configuration of the security policies applicable to the communication engine. The SMGW management process supports the use of policies implemented in the form of a Policy Based Management Tool. This tool handles authorisation, authentication and accounting functions. Specific policies are able to define all aspects of the existent relations among SMGWs, including the definition of how each particular CI can connect, access control to alert information and the enforcement of intrusion detection policies.

Another important element of the MICIE system architecture is the Adaptor, which is used to interface the MICIE system with the CI monitoring systems located in the field. Management strategies were also developed in order to allow an easy management of the Adaptors hence allowing the Operator to configure them. For instance, configuring settings in the communication protocol, data filtering rules and output data format.

3.6.1 SMGW Policy Based Management

The proposed approach offers the CI Operator a management tool where it is possible to define, in a high level manner, the intended behaviour of the system. Traditional approaches are mainly oriented to the management of individual components, not completely considering the system structure as a whole. In this proposal the concepts of Policy Decision Point - the SMGW Manager - and Policy Enforcement Point - the entities that must enforce policies, are applied. For instance the Communication Engine and the Subscription Filter. Figure 3.10 represents the SMGW Manager within the SMGW architecture. In this Figure the proposed PDP and the existing PEP are represented. In particular, the one intended to enforce policies related to data subscription and the other enforcing policies related to the remote connections in the Communication Engine. It is also possible to deploy PEP acting outside the SMGW, able to manage, for instance, communication aspects that are not visible within the SMGW. The SMGW Manager includes the PDP, all defined policies, the Managed Objects and also a Trust and Reputation System (described in Chapter 4).

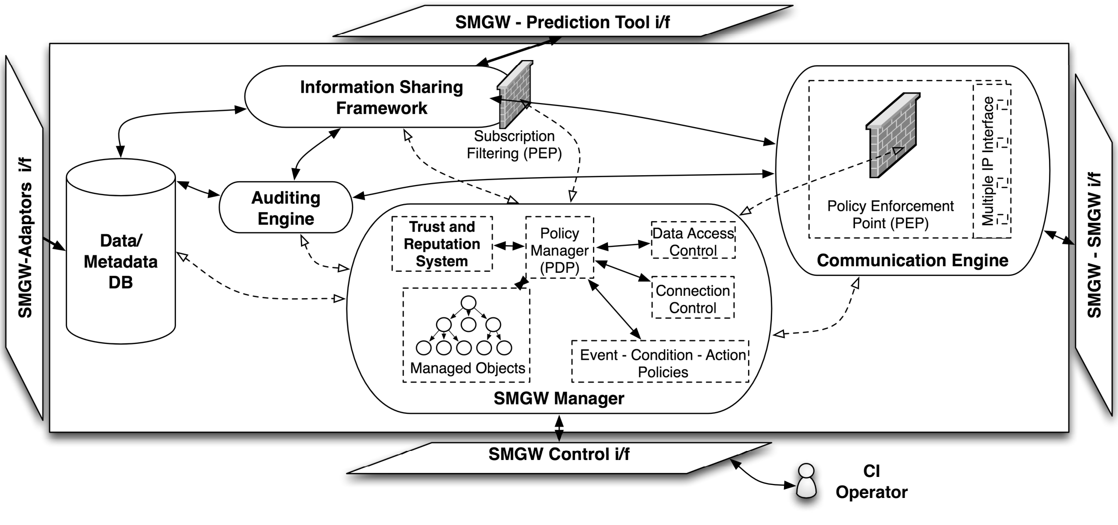


Figure 3.10: MICIE SMGW detailed architecture (Caldeira et al., 2010b).

The CI Operator is able to define policies that will address the relations among the local, and foreign SMGWs. Policies include forms of defining how each particular CI must connect and also include data access policies. The SMGW Manager Graphical User Interface (GUI) allows to browse existent information and also define actions that the remote SMGWs can perform (e.g. write or read risk information). All data access controls are implemented with a high level of granularity thus maintaining simplicity.

The use of Policies will supports writing, verification and deployment of security policies related to the information gathered by the MICIE system. For example, these policies allow to (Caldeira et al., 2010c):

- define how and to whom each particular piece of information can be sent;
- define trust relations between different CIs through the use of the Trust and Reputation System (detailed in Chapter 4);
- enforce different communications protocols/technologies in each particular context;
- enforce Service Level Agreement or Service Level Specification between CIs acceptable at the Communication Engine;
- decide how received events will be managed by the SMGW.

The CI Operator defines policies by means of the defined policy specification language by using the provided GUI. This GUI displays the representation of all managed entities allowing the CI Operator to easily define relations among them (policies).

The defined policies are represented using a policy specification language and stored in a policy repository. The SMGW Manager interacts with other entities on the SMGW through a dedicated API implemented on a Web Service. The SMGW Manager is based on the PONDER2 Toolkit (Ponder, 2010; Twidle et al., 2009) on which each SMGW entity is represented through the PONDER2 concept of Managed Object. The complete set of SMGW entities will form a PONDER2 Self Managed Cell. The policy makes use of PONDER2 Authorisation Policies and Event-Condition-Action concepts.

Apart from the existing PONDER2 communication modes, a dedicated API was developed in order to manage all communication aspects between the SMGW components and the SMGW Manager. This API provides the possibility to, for example:

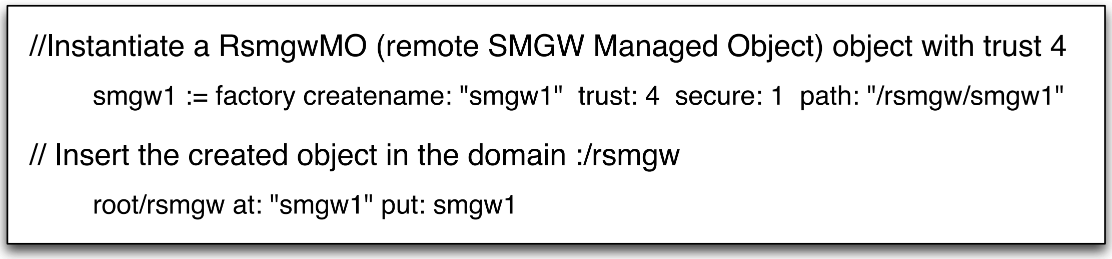
- Change state (attributes) of the Managed Objects (e.g. change the connection type allowed for a remote SMGW): Handled as an Event-Condition-Action in PONDER2 context;
- Authorisation requests: Handled as authorisation Policies in PONDER2 context (e.g. enforce new authorisation policies on the PEPs).

The SMGW SMC comprises the representation of all system's Managed Objects. In a simplified approach it is possible to identify two main types of Managed Objects: Connections to the SMGW are represented by Remote SMGW Managed Objects (RsmgwMOs) and the Alerts information is represented by the defined Alert Parameters Managed Objects (AlertParMOs).

RsmgwMOs represent and maintain information related to the remote SMGWs that are participating in the MICIE system. Each Remote SMGW (RSMGW) includes all the necessary attributes that allow to describe itself along with its actual connection attributes. For instance, it is possible to have among the RSMGW attributes: the RSMGW name, the SLA parameters defined to allow connections, the IP addresses used to connect, the trust level, encryption type, etc.

The AlertParMOs maintains information regarding all Alert parameters that then SMGW is able to exchange with peers RSMGWs. Among the AlertParMO attributes are, for example, the attribute name and value as well all information needed to interpret that parameters.

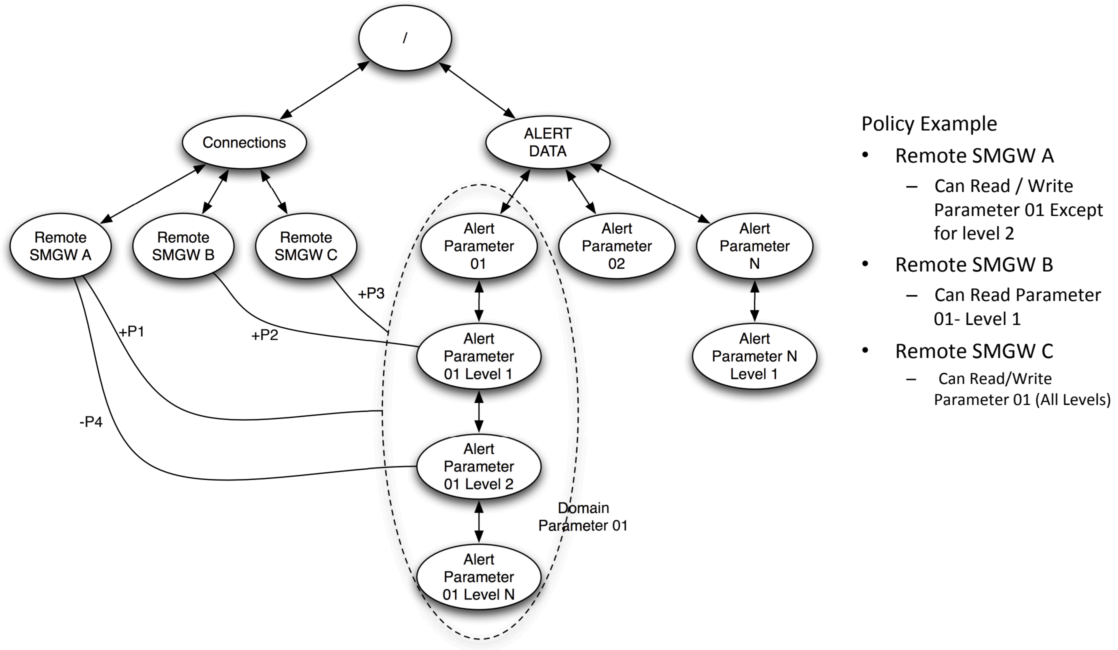
The Managed Objects are created and instantiated by directly using the PonderTalk language (Ponder, 2010) or through the Java API developed on top of PONDER2. An example of an object creation using PonderTalk is presented in Figure 3.11. In this example, a new RsmgwMO named *smgw*1 is created under the *rsmgw* object, with the attributes trust and secure respectively assigned with the values 4 and 1. Upon creation, the newly created object is inserted in the object domain *rsmgw*.



```
//Instantiate a RsmgwMO (remote SMGW Managed Object) object with trust 4
    smgw1 := factory createname: "smgw1"  trust: 4  secure: 1  path: "/rsmgw/smgw1"
// Insert the created object in the domain :/rsmgw
    root/rsmgw at: "smgw1" put: smgw1
```

Figure 3.11: RsmgwMO manipulation example.

In order to allow access control, multiple policies might be applied to objects or object domains. For example, it is possible to define a policy allowing the remote SMGW X to read Alert parameter Y. Also, a connection request event can enforce changes on the firewall configuration regarding the access of one particular RSMGW. Figure 3.12 presents a simplified approach to the existent MICIE SMGW Self Managed Cell.



Policy Example

- Remote SMGW A
  - Can Read / Write Parameter 01 Except for level 2
- Remote SMGW B
  - Can Read Parameter 01- Level 1
- Remote SMGW C
  - Can Read/Write Parameter 01 (All Levels)

Figure 3.12: SMGW Manager SMC simplified approach.

The SMGW Manager Server Application (PDP) accepts connections from one or more policy enforcement points and speaks to Ponder2 directly. The server application has the following main functionalities:

- Enables the Operator to have an overview of all the objects in the domain;
- Store changes made from the SMGW Manager GUI: save/retrieve Managed Objects and their attributes; save/retrieve authorisation policies with conditions; save/retrieve event policies conditions;
- Authenticate CIs that request a Managed Object through the web service;
- Provide attributes from Managed Objects requested from the web service by authorised CIs;
- Define new attributes when the current number of attributes is no longer sufficient;

- Creating policies that can use attributes from both source and target objects.

The Managed Objects are stored as XML files. Every time an object changes (e.g. it is created, deleted, updated) the appropriate XML file is updated. When the server initiates, it looks for these XML files, analyses them and creates the objects in both Java and PONDER2. This type of persistence has others advantages, for instance, the XML files can easily be copied to another location as a matter of backup. In case of a hardware crash (or any other failure) the server can be booted from another system with the backup XML files. Another advantage of the use of XML files, to save the objects is that the communication among the PEPs and the PDP, is performed using the exact same XML structure. The architecture for the described framework is represented in Figure 3.13.

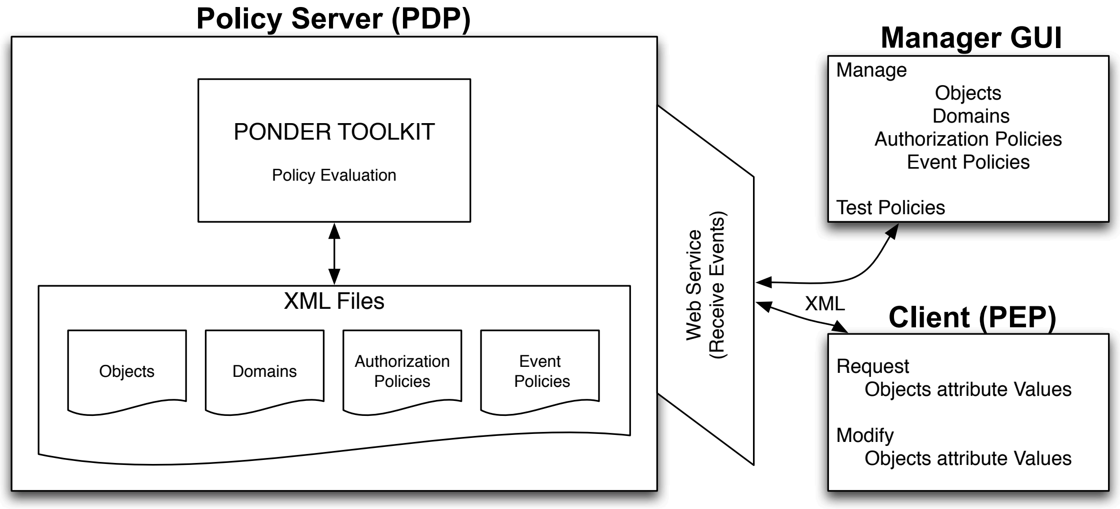


Figure 3.13: SMGW Manager simplified architecture.

An example of how a Managed Object is represented in XML is visible in Figure 3.14. It is a simple and common XML representation containing the *path* (location) of the object inside the domain and, in this example, the following attribute-value pairs: $level = 10$, $trust = 5$ and $value = 2$.

The communication among the PDP and the PEPs is carried through XML messages. In all cases, the root-node of these XML messages is $< transaction >$ and each transaction might possess one or more sub nodes. These sub nodes are then considered as *actions*. With this approach, by exchanging just one XML message it is possible for the server to execute more than one action. Figure 3.15 presents an example of a transaction representation in XML. In this example, the first action requests to perform the action *editObject* for the object in $path = /domain1/object1$.

```
<object>
  <path>/domain1/object1</path>
  <attributes>
    <attribute>
      <name>level</name>
      <value>10</value>
    </attribute>
    <attribute>
      <name>trust</name>
      <value>5</value>
    </attribute>
    <attribute>
      <name>value</name>
      <value>2</value>
    </attribute>
  </attributes>
</object>
```

Figure 3.14: Example of a Managed Object represented in XML

In this case new attribute-value information is inserted in this object. The second action included in this example, is *getAll* that instructs the PDP to send, all existing objects available to that PEP, to the requester in order to enable it to refresh it's domain view.

The use of XML messages gives further advantages to communication. Firstly, it is easily read by humans, so if someone desires to create a different and more user friendly GUI, to manage the system, all one has to do is to respect the defined XLM structure. A second advantage, is that the data, or only the entire string can be easily secured by using encryption mechanisms. Table 3.1 presents a list with all supported actions.

The SMGW Manager architecture has been proposed by the author of this thesis and has been integrated in the existent MICIE SMGW as explained by Castrucci et al. while describing the design and implementation approach used for the SMGW development (Castrucci et al., 2010b, 2012).

The author of this thesis has developed the SMGW Manager comprising of three main modules, namely, the Server GUI, the Manager GUI and the Migration Tool.

The Policy Server is an application with a simple GUI on which the status of the system and possible errors are displayed. It is the system's main component as it is responsible for making the bridge among the XML files, the Java API and the PONDER2, toolkit, where policies are evaluated. Implemented using Java language,

```
<transaction>
  <action type=”editObject”>
    <object>
      <path>/domain1/object1</path>
        <attributes>
          <attribute>
            <name>risk</name>
            <value>2</value>
          </attribute>
          <attribute>
            <name>level</name>
            <value>10</value>
          </attribute>
          <attribute>
            <name>value</name>
            <value>2</value>
          </attribute>
        </attributes>
    </object>
  </action>
  <action type=”getAll”></action>
</transaction>
```

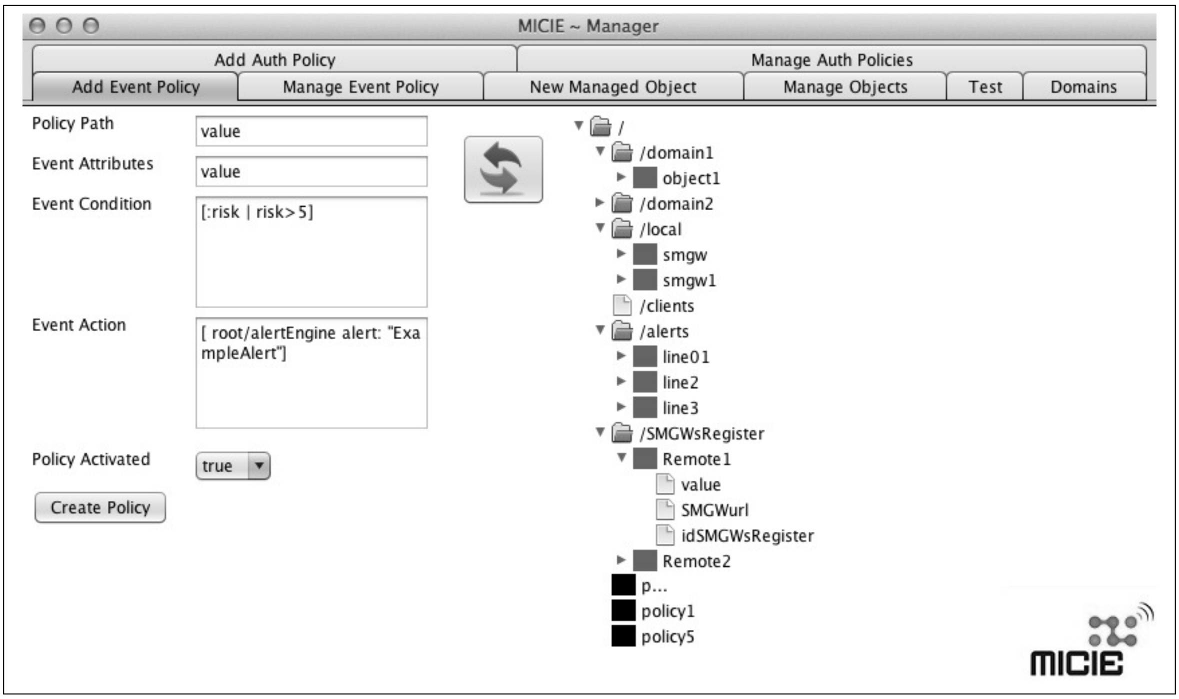
Figure 3.15: Example of a transaction represented in XML

Table 3.1: Actions supported on the PDP

Action Type	**Actions**
Policy Manipulation	createPolicy getPolicies deleteAuthPolicy changePolicy activateDeactivePolicy
Object Manipulation	createObject getObjects editObject deleteObject
Domain Manipulation	createDomain
Event Policy Manipulation	createEventPolicy changeEventPolicy deleteEventPolicy getEventPolicies
Other	getAll getAttribute

it incorporates the main web service that allows communication from the Operator GUI and also from clients (PEPs).

The SMGW Manager GUI allows the CI Operator to define the behaviour it planned for the system. It allows creating, modifying and deleting the Domains, the Managed Objects and Policies. It has drag and drop functionalities, allowing a simple use of the existent attributes. It is also possible, in this tool, to have an overview of all Managed Objects. An important aspect that was implemented refers to policy testing. The Operator is able to specify and simulate particular requests made by a PEP and verify the results, according to the defined policies. With this functionality, the Operator is able to verify with detail if the defined policies are able to enforce the defined security requirements. Figure 3.16 displays an overview on the existent SMGW Manager GUI.



Policy Path: value

Event Attributes: value

Event Condition: [:risk | risk>5]

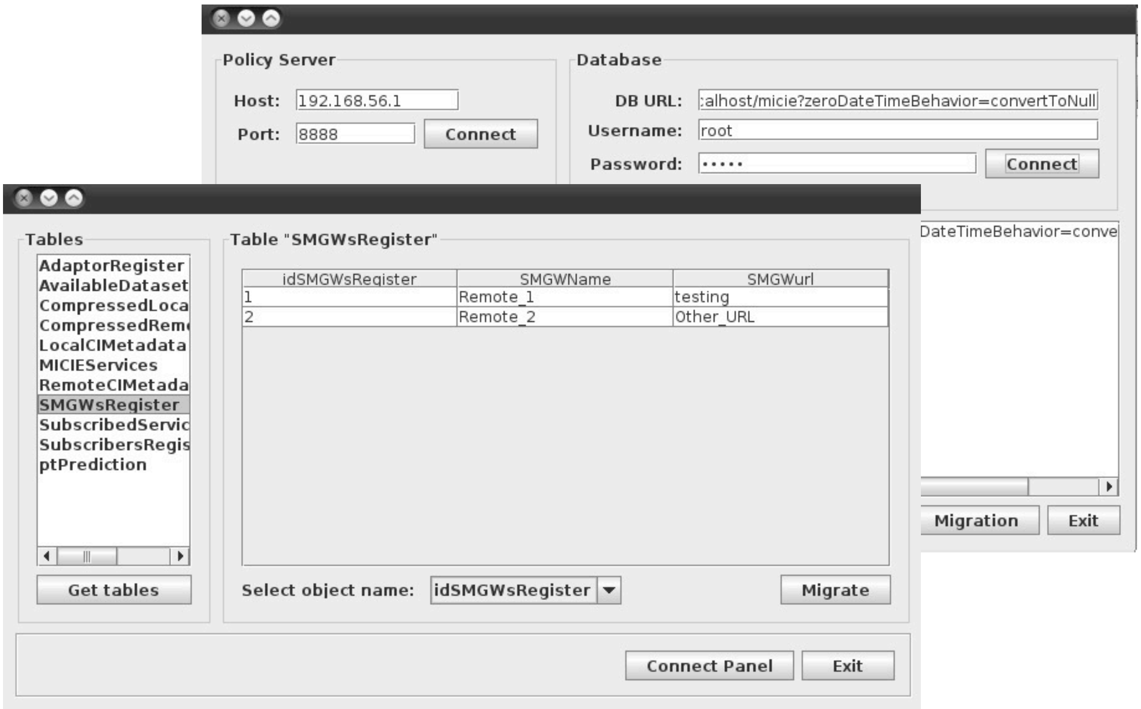
Event Action: [root/alertEngine alert: "ExampleAlert"]

Policy Activated: true

Figure 3.16: MICIE SMGW Manager GUI.

The developed Migration Tool allows the migration of the existent data in the SMGW Database to the Policy Server (for example, remote CI names, risk data names). Basically, it is possible to migrate and map to the SMGW Manager, all the existent database tables with their respective attributes. An overview of the Migration tools application is displayed in Figure 3.17. This application was of great importance during the testing phase of the system. A test client was also developed in order to simulate the requests arriving from the PEPs.

Regarding the sensitive nature of the exchanged information, the MICIE project has dedicated special attention to the security requirements, such as confidentiality, integrity, availability, non repudiation and auditability/traceability. In order to contribute to the security requirements, the author of this thesis proposed the evaluation and usage of Trust and Reputation indicators in the SMGW Manager,



Policy Server

Host: 192.168.56.1

Port: 8888

Database

DB URL: :alhost/micie?zeroDateTimeBehavior=convertToNull

Username: root

Password: •••••

DateTimeBehavior=conve

Tables

AdaptorRegister
AvailableDataset
CompressedLoca
CompressedRem
LocalCIMetadata
MICIEServices
RemoteCIMetada
SMGWsRegister
SubscribedServic
SubscribersRegis
ptPrediction

Table "SMGWsRegister"

idSMGWsRegister	SMGWName	SMGWurl
1	Remote_1	testing
2	Remote_2	Other_URL

Select object name: idSMGWsRegister

Figure 3.17: MICIE SMGW Manager Migration Tool.

allowing also these indicators to become available to the Prediction Tool. The main goal of these indicators is to contribute on the accuracy improvement of the existent information, to help the SMGW Manager to protect each CI from receiving and using inconsistent information and to gather Trust and Reputation information regarding the behaviour of each involved CI.

The TRS evaluates information exchanged among CIs in order to infer a trust level for each transaction. This service incorporates a level of trust in the data received from each partner, allowing those trust levels to be incorporated in risk assessments, as a mean of improving its accuracy and its resilience to inconsistent information. It is possible, for instance, to give more weight to highly trusted data or to ignore data provided by low-trust partners. The proposed framework employed will be described in Chapter 4.

3.7 Validation Activities

MICIE project carried out a work package dedicated to the validation activities foreseen for the project – WP6000 - Validation. In this context, several validation activities were accomplished with the contribution of the Israel Electric Corporation (Lev et al., 2010a).

Most of the validation activities took place in the IEC labs in Haifa, Israel, and considered the IEC premises based on: the Simulation Test Bench (STB) that aggregates real CIs equipment, simulated equipment, historical real data and working procedures (Lev et al., 2011, 2010a,b).

The demo system (represented in Figure 3.18) aimed to validate the MICIE tool by running the validation scenarios. It includes a STB which simulates a "real life" operational CI with a few, partially built and partially simulated pseudo CIs (Lev et al., 2011). All the MICIE components, in particular the SMGW (including the SMGW Manager), the Prediction Tool and the multiple Adaptors), were tested in the scope of the corresponding development groups, before integrating into the demo system (Lev et al., 2010b).

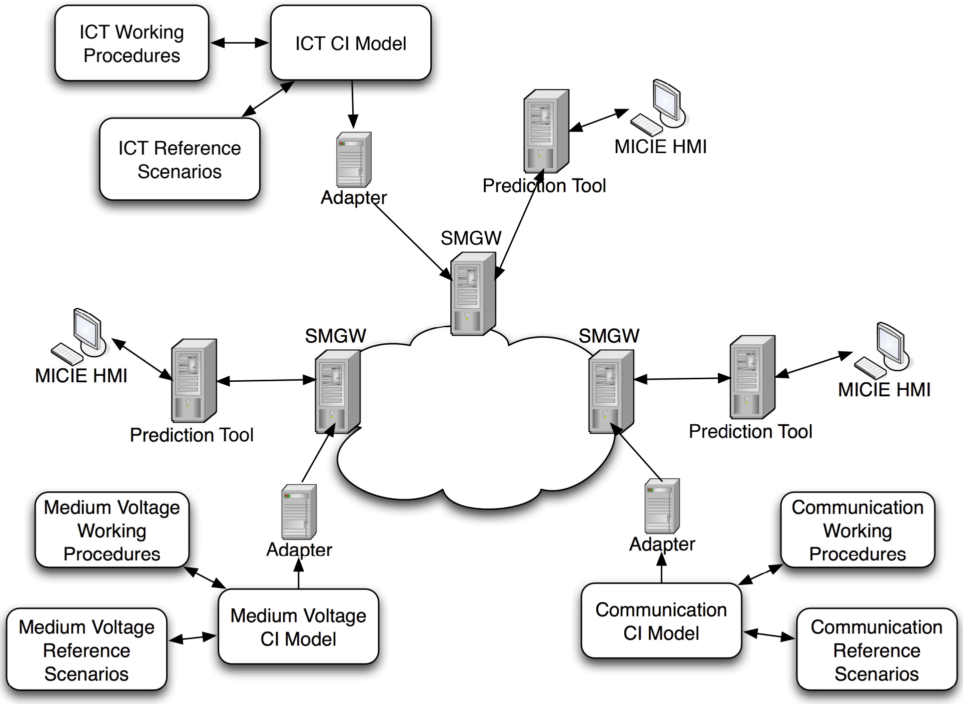


Figure 3.18: MICIE Demo System Configuration (Lev et al., 2011).

Figure 3.18 represents the demo system implemented in a IEC test lab in Haifa, Israel. It simulates the usage of the MICIE system within a Communication CI, Medium Voltage CI and an ICT CI and includes all of MICIE's main components. Some real field equipment was also included in the validation scenario (Lev et al., 2010b).

The MICIE validation and demonstration plan has been achieved following three phases (Lev et al., 2011). First of all the involved CIs were operated according to the real working procedures without introducing any external events. This step allowed to gather the information produced during normal operation. In the second phase, external events were induced within the scenarios and again the information from this operation was gathered. The third phase was carried out with the MICIE system already installed in order to validate and demonstrate the possibilities of the MICIE project for supporting the decision making process of the CI Operator while acting within situations of uncertainty (Lev et al., 2011). The SMGW Manager, incorporating the Policy Based Management approach, contributed by the author of this thesis, was validated within the SMGW validation activities. The results from these activities revealed that the system was properly managed within the project expectations.

Based on the test and validation activities performed at the IEC labs it is possible to state, that the MICIE system is a powerful tool. It is able to assist the CI Operator in reducing the risk of failure in a network suffering from events induced by interconnected CIs. It was also inferred from the tests that the MICIE system is able to help reduce the outage time in the electric distribution network (in case of fault events) by about 2%. The MICIE system also provides the Operators with a wider perspective on the status of the overall system. It takes advantage of a prediction of the availability of most relevant services in short, medium and long term. This last fact is important as the CI Operator rapidly becomes aware of the status of the services allowing more efficient response (Lev et al., 2011).

According to the MICIE validation and demonstration results the MICIE system can be usefull, among other applications, to: protect energy and communication CIs; decrease uncertainty while operating the CI; reduce time of service restoration; support on-line decision making to predict cascade failure; SLA improvement based on analysis of highly risk potential outages of the CI (Lev et al., 2011).

3.8 Summary

The MICIE system acts on a distributed environment composed of multiple heterogeneous CIs, that might depend on one or more services provided externally by others CIs. Considering that the CIs are willing to cooperate, in order to improve the provided quality-of-service, the MICIE system introduces mechanisms allowing

CIs to be able to predict and exchange risk information across trusted or untrusted network infrastructures (e.g. Internet).

MICIE project was discussed in depth due to the fact that most of the research challenges that led to this thesis were first discussed and applied within this project. This chapter allows the reader to see an overview of the project with particular emphasis on the components in which the author of this thesis has been actively involved. In particular it is possible to highlight the author's involvement in the work related to the MICIE work package 4000 (WP4000) - Mediation System Design.

The MICIE project's main goals were discussed, as well as the proposals that were presented to achieve them. The modelling activities that took place within the project were discussed including the author's contribution to them – ICT-MC. The risk prediction work package was also addressed in the chapter. The architecture of the system was described along with their main components.

The author's proposal for the SMGW management was also addressed and discussed in this chapter. The validation activities that took place in the IEC facilities in Israel were presented, in order to demonstrate the project's success while dealing with such a complex problem. In this chapter the use of Trust and Reputation indicators was also addressed, in order to improve the SMGW management and also the results from the Prediction Tool. A detailed description of such a framework will be presented in the following chapter.

Chapter 4

A Framework for Trust and Reputation Management in Critical Infrastructures

As discussed in previous chapters, there are several models that provide different approaches in order to understand the (inter)dependencies occurring among heterogeneous CIs. The MICIE project highlighted the relevance of a system, able to use these models, to provide specific instruments to CI owners, in order to reduce the risk of service unprovisioning. For instance, in the MICIE project, risks evaluated in each member CI, are possible and intended to be shared among other CIs in order for them to better evaluate their own risk while considering the existent risks in each (inter)dependent service. The design and implementation of such a system - a real-time risk level dissemination and alerting system – has been successfully tested within the MICIE project.

Current research in CI Protection is mostly focused on understanding and modelling (inter)dependencies among CIs and the use of these models in allowing the development of risk prediction tools. In order to properly evaluate CI risks, these prediction tools receive inputs from several sources, such as, monitoring and control equipment, Operator information and risk information provided by (inter)dependent CIs. Considering the MICIE project as an example, a secure communication system, allowing the secure exchange of risk information, has been deployed allowing the participant CIs to share relevant information that can feed their own risk prediction tools.

Although the existing risk evaluation methodologies or prediction tools are able to safely merge risk information arriving from multiple sources (sources available in the local CI or external sources). The lack of mechanisms allowing to observe and reason about the confidence one can have in the information collected from these sources was identified. Also, it is important to understand the behaviour of the information sources in order for it to become possible to infer trust information regarding that behaviour.

In short, it is intended to answer, at least, one main question that remains open - “How can information used for risk calculation be evaluated for correctness?” This chapter describes the proposed Trust and Reputation framework aiming to allow the incorporation of Trust and Reputation indicators on the information exchanged among Critical Infrastructures and also on information coming from monitoring equipment. The application scenario within the MICIE project (Bertoni et al., 2010a) and the approach used for the Trust and Reputation framework are described before presenting validation work.

Although the Trust and Reputation framework was initially focused on the MICIE project, the proposed framework can be considered a general framework and thus can be applicable within different models and scenarios (discussed in Chapter 5). Contributing to the improvement of risk estimate and sharing mechanisms within a Critical Infrastructure.

This chapter is organised as follows. Section 4.1 describes the proposed general Trust and Reputation Model while Section 4.2 describes the methodology employed for its application. Section 4.3 presents some examples of the resulting work. The tools developed to proof the concept are described in Section 4.4. Section 4.5 summarises the issues addressed throughout the chapter.

4.1 Trust and Reputation Model

Usually, a scenario on which multiple CIs are willing to exchange risk information, in order to improve their risk prediction accuracy, is considered as being a closed and protected system. That is, all the participants assume that the information is securely shared among them and, it is assumed that all participants trust each other, assuming that they are integrating the system with good will. In fact, this should be the correct assumption for such a scenario. However, and not denying the fact that all participant CIs should have honest intentions, this assumption might not be exactly accurate as several problems can occur.

Although a system able to exchange CI risks must enforce multiple security mechanisms that allow for information security, typically those mechanisms are focused on the communication and not on the exchanged data itself. It is always possible for a participant CI to provide inaccurate information thus affecting the dependent CIs. This can happen either, maliciously (e.g. if the system is somehow compromised) or due to the existence of faulty components in the CI monitoring frameworks.

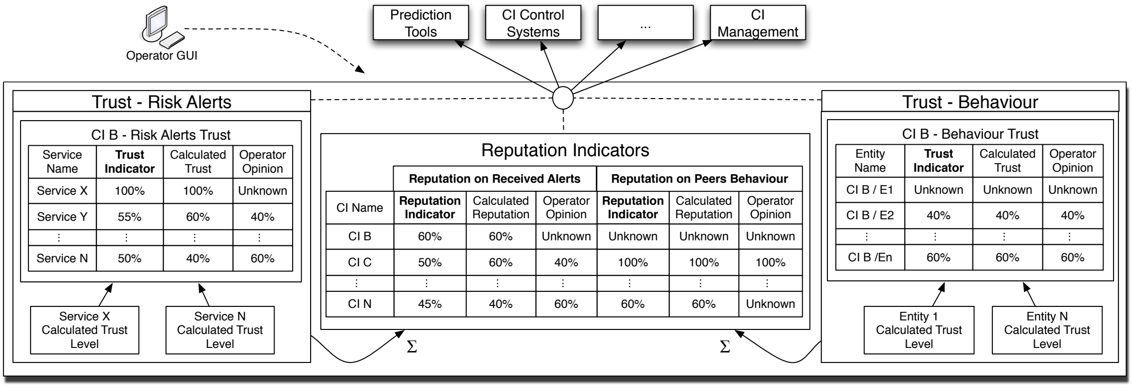
In such a context, it is important to introduce mechanisms able to allow reasoning on the exchanged information quality and also about the context on which the information is being exchanged.

From the above, the need for a Trust and Reputation System, employed in each CI, able to maintain real-time trust information concerning (inter)dependent CIs and CI services, was identified. This system is able to monitor information exchanged among CIs or among CI services and also to monitor their behaviour in order to gather a trust level for each CI service and to infer CI reputation (Bertoni et al., 2010a; Caldeira et al., 2010b,c,d; Castrucci et al., 2012).

The proposed framework aims to evaluate the information received from a dependency based on the previous observations made on that dependency and also to understand the behaviour of the participant CIs within the partnership. Depending on the outcome of such an evaluation, a CI Operator can decide to what extent the information received from the dependency will be incorporated in the CI risk evaluation. The proposed evaluation can be achieved by building a trust relationship between CIs or CI services through a TRS and by the use of gathered trust level indicators to evaluate the correctness of the received CI information. A shortcoming of applying Trust and Reputation Systems to the domain of CIs is related to the variety of CIs that may exist. Each infrastructure can have different information being compared and evaluated. Building a Trust and Reputation System taking into account several dependencies within different contexts, can be a fairly complex task to which the proposed framework makes a significant contribution, as it allows a methodical and simple approach to the process.

It is important to clarify that, in this context, dependency or interdependency, does not only refer to relations among different CIs but may also refer to relations among services existent within one CI. From the TRS point of view, such a distinction is not relevant for its usage. Due the fact that the presented framework was primarily intended to be applied in scenarios on which CIs share risk information related to the interdependencies existing among them, this chapter will describe the TRS within such a scenario.

There are two main areas where the proposed Trust and Reputation Model is able to be applied (see Figure 4.1). First, to a trust indicator concerning the information received from (inter)dependent CIs (*risk alerts*). It is possible to evaluate this indicator from two distinctive perspectives: for each available service, evaluating each service provided by a remote CI, thus reflecting the trust on the risk alerts received from each dependent service (Risk Alerts Trust); for each CI, evaluating an indicator for each interconnected CI, representing the reputation of that particular CI. Second, the Trust and Reputation System is also capable of understanding the (inter)dependent CIs' behaviour, for instance, in terms of ICT security (Behaviour Trust). In this case, the evaluation is made on multiple entities, each one representing one particular aspect of the CI or CI service. The aggregation of the behaviour evaluation, from multiple entities belonging to the same CI or CI service, represent the Reputation of that CI or CI service.


Operator GUI
Prediction Tools
CI Control Systems
...
CI Management
Trust - Risk Alerts
Service X Calculated Trust Level
Service N Calculated Trust Level
Σ
Σ
Trust - Behaviour
Entity 1 Calculated Trust Level
Entity N Calculated Trust Level

CI B - Risk Alerts Trust

Service Name	**Trust Indicator**	Calculated Trust	Operator Opinion
Service X	100%	100%	Unknown
Service Y	55%	60%	40%
⋮	⋮	⋮	⋮
Service N	50%	40%	60%

Reputation Indicators

	Reputation on Received Alerts			**Reputation on Peers Behaviour**		
CI Name	**Reputation Indicator**	Calculated Reputation	Operator Opinion	**Reputation Indicator**	Calculated Reputation	Operator Opinion
CI B	60%	60%	Unknown	Unknown	Unknown	Unknown
CI C	50%	60%	40%	100%	100%	100%
⋮	⋮	⋮	⋮	⋮	⋮	⋮
CI N	45%	40%	60%	60%	60%	Unknown

CI B - Behaviour Trust

Entity Name	**Trust Indicator**	Calculated Trust	Operator Opinion
CI B / E1	Unknown	Unknown	Unknown
CI B / E2	40%	40%	40%
⋮	⋮	⋮	⋮
CI B /En	60%	60%	60%

Figure 4.1: Trust and reputation model

In a situation where CIs are sharing sensitive information - risk information, it is assumed that a monitoring system is employed. For instance, the interactions among participant CIs should be observed and must obey the defined security policies. Also, the CI Operator must know the normal behaviour of the managed system, not only regarding the security aspects, but also aspects related to the proper operation of the system. Aspects, such as, acceptable value ranges, time intervals on which information must be gathered, unexpected comportments of sensor, are some examples of information the CI Operator should know. All this types of information must be collected in order to gather intelligence about the partnership. Thus if one CI behaves incorrectly according to the defined security policies and expected behaviour, for example, trying to repeatedly retrieve private information, this can be seen as an ICT incident. This type of information is included in the evaluation of the Behaviour Trust indicator. Figure 4.1 represents the main indicators able to be gathered within the proposed model.

The proposed model exemplified in Figure 4.1, evaluates information exchanged among CIs, in order to infer and associate a trust level for each transaction (risk information received from a peer CI or CI service). This allows to incorporate trust indicators in CI risk assessments as a means of improving its accuracy and its resilience to inconsistent information. It will be possible, for instance, to give more weight to highly trusted risk alerts or to ignore risk alerts provided by low-trust CIs or services (Caldeira et al., 2010b).

Within the proposed approach, it is possible to evaluate Trust and Reputation indicators gathering and evaluating information received from multiple and heterogeneous sources. The fact that the information sources are heterogeneous is particularly relevant while evaluating a CI or a service behaviour. On the described Trust and Reputation Model, it is possible to identify three main information sources: The past data provided by the (inter)dependent CI services, the information gathered regarding the behaviour of a CI or CI service and also the CI Operator trust on each CI or CI service (Caldeira et al., 2010b,c,d).

The historical data provided by the (inter)dependent CI services is one of the major information sources used in the model. This data is analysed in order to compare, for each service, the service risk alerts received over time, against the actual QoS level of each service. To achieve this analysis, it is mandatory to have available, at each moment, the QoS level measurement for each service based on which it is planned to evaluate trust.

In the cases on which the received risk alerts and the QoS level are represented using different value ranges, they must be normalised in order to allow to compare them. For instance, if the received risk alerts are defined within the discrete interval [1..5] and the QoS is measured as a percentage of the service availability, then it is easily converted to the same range as the risk alerts. For different types of QoS measurements one must find a function or normalisation table that allows to normalise the information.

The results from the analysis of the historical data against the service QoS is then used to infer the degree of trust of actual and future received risk alerts. For example, if a CI service keeps informing the highest risk alert level and the measurements of the service QoS never indicated a service failure, it is natural to infer that this particular risk alert has low credibility. This source of information is represented in Figure 4.1 on the Trust – Received Alerts block. A representation of a possible integration of the Trust Model, within a CI risk sharing system, is presented in Figure 4.2.

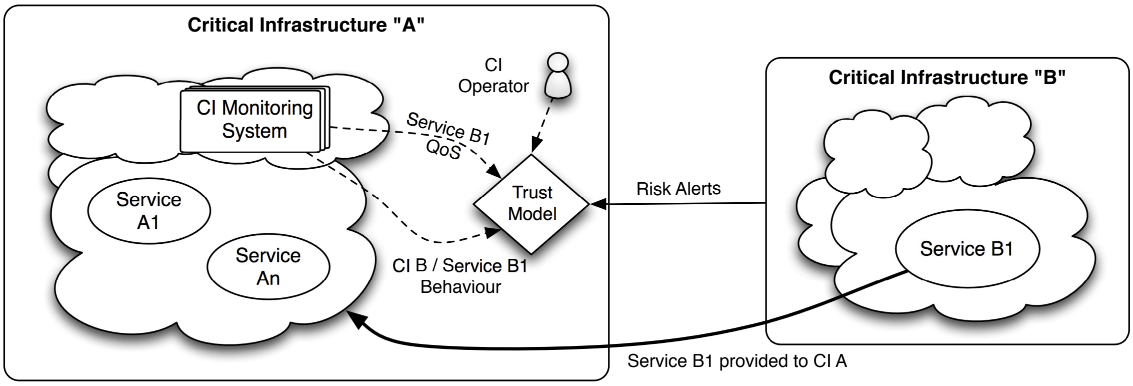

Critical Infrastructure "A"
CI Operator
CI Monitoring System
Service B1 QoS
Trust Model
Service A1
Service An
CI B / Service B1 Behaviour
Risk Alerts
Critical Infrastructure "B"
Service B1
Service B1 provided to CI A

Figure 4.2: Trust model application example.

The analysis of (inter)dependent CI or CI service behaviour is supported by the knowledge gathered from the security entities existent in the CI and on the knowledge gathered by analysing the existing deviations from normal behaviour that may exist, for instance, among CIs, services or CI components. As an example, it is possible to analyse the CI behaviour in terms of ICT security. For instance, if the (inter)dependent CI behaves abnormally (for example, it is not sending information within the expected time frame) the level of trust associated with that CI should be downgraded, as this fact could indicate that the CI control centre is faulty or does not have good intentions. Figure 4.1 illustrates this source of information on the Trust – Behaviour block.

A human factor is also included in the model. Although one can rightly argue, that the information provided by humans can be subjective, the human factor intends to reflect the perception of the CI Operator of each (inter)dependent CI / service on the evaluated trust indicators. Among other aspects, it is relevant to integrate this factor in the assessment as the CI Operator or Operators have significant expertise on some highly specialised area.

Furthermore, it is possible for the CI Operator to have access to information regarding each (inter)dependent CI / service and may desire to incorporate it in the

Trust and Reputation assessments. For instance, it is possible to conceive the realistic situation of which the CI Operator is aware (he could had been personally informed) that some CI has operated with faulty equipment during a determined period of time. In this situation, it is likely that, during that period of time, the calculated trust indicator decreases. In this case, the CI Operator must be able to act by incorporating his own information. For instance, by raising the CI Operator's confidence parameter in one particular CI and consequently preventing a decrease of the global trust value.

The human factor or the Operator opinion (as represented in Figure 4.1 is applied in all the evaluated indicators. That is, the CI Operator can include his opinion on different levels, for instance, reflecting his trust on one particular service received by a peer CI or reflecting his trust in a peer CI.

In the following section, the methodology used for implementing the proposed model, is described in detail.

4.2 Trust and Reputation System

The following presented Trust and Reputation System, implements the described trust model. It is presented also focusing on its application within the MICIE system.

Figure 4.3 represents the TRS architecture which is composed of four main components: Two Agents (the Risk Alerts Trust Agent and the Behaviour Trust Agent) aiming to gather all the necessary information; the TRS Discovery Tool that computes the information received from the Agents and the Queries Service/Operator GUI that allows the TRS to interact with the CI Operator and also to publish the obtained results.

The information required to evaluate the proposed indicators is gathered from the system using two types of Agents.

The first type, the Risk Alerts Trust Agents, are continuously observing the QoS of each service and are kept informed of all risk alerts received from peer CIs or CI services. According to this real-time information, these Agents are able to detect and evaluate an accuracy value for each risk alert event. The concept of event will be detailed later in this section. In a simplified perspective, a risk alert event is a

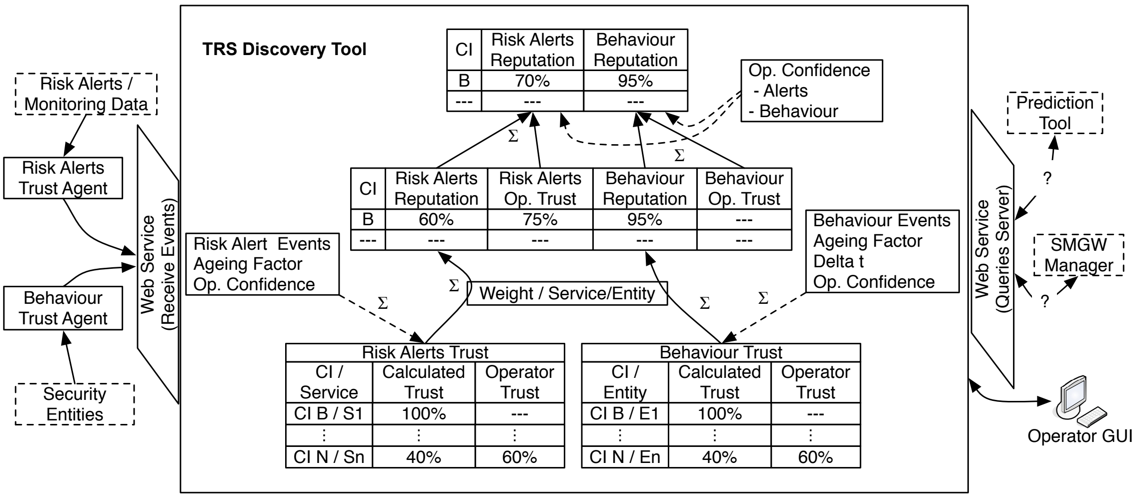


Figure 4.3: Trust and Reputation System (Caldeira et al., 2010b)

situation in which the received risk is different from normal or the monitored service QoS decreased or both.

The second type, the Behaviour Trust Agents, are intended to receive and normalise all the behaviour events. A behaviour event can be any kind of abnormal situation as observed by the Security Entities. Each of the agents send the discovered events to the TRS Discovery Tool, aiming to compute in real-time all the Trust and Reputation indicators. The computed indicators are provided to external entities, for example, in the MICIE system, both CI SMGW Manager and the CI Prediction Tool are able to make use of these indicators. A graphical interface provides the CI Operator with an overall view of Trust and Reputation indicators while allowing the CI Operator to also update his opinion.

4.2.1 TRS Agents

As stated, the Trust and Reputation System (TRS) architecture employs two agents for gathering the required information for trust evaluation on both risk alerts and CI/service behaviour. In the following, each of the agents are described.

Risk Alerts Trust Agents

Within a scenario of interconnected CIs willing to cooperate by exchanging proper risk information, each CI is able to subscribe risk alert information regarding dependent services and use it, in particular, to compute its own risk level.

In order to be able to evaluate trust aspects related to the received risk alerts, the first goal is to define an accuracy value for each received risk alert. This is the goal of the Risk Alerts Trust Agent. Figure 4.4 represents an overview of the process handled by this agent. As it is noted, each observed service has its own Risk Alerts Trust Agent. This happens, as the information gathered from the services is usually heterogeneous, thus each one needs a proper information normalisation process.

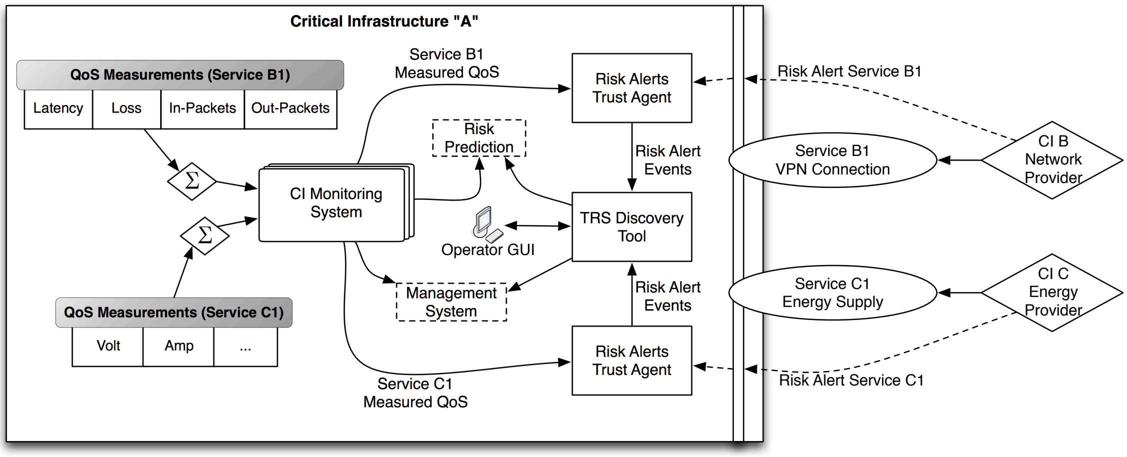


Figure 4.4: Risk Alerts Trust Agents example.

The Risk Alerts Trust Agent receives from the CI Monitoring System, the QoS measurement from the service which the agent is observing. It also receives, from the responsible component (e.g. in the MICIE system the SMGW is responsible to receive the risk information and to provide it internally), the risk alert information as received from the dependent service. It then allows to compare the risk alert information received for the dependent services with the measured QoS of the service. For this purpose, the concept of Risk Alert Event has been introduced.

Within such a complex systems as CIs, it is possible to have different modes of receiving risk alerts and also different approaches to the service QoS measurement. In order to be able to compare both information, a simple normalisation process is required. For instance, if the received risk alert is within the discrete range [1..5], the measured QoS indicator should be mapped to the same value range. This can be achieved directly within the Monitoring Systems or processed by each Risk Alerts Trust Agent. In both cases, in this example, if the service QoS is represented using the range [0..100], a mapping table, equal or similar to the one presented in Table 4.1, can be used. In this normalisation it is necessary to have a good definition for the obtained values in order to be able to compare and interpret them. In this case, the received risk alert of 1 means no risk and 5 means high risk. For the measured service QoS, value 5 means that the service is within the lowest QoS range admitted for the service. A value of 1 means that the service QoS is within the optimum values. It should be highlighted that this is just an example and that these normalisation tables, if existent, need to be defined by a CI expert.

Table 4.1: Example of a service QoS normalisation table.

Measured service QoS	Normalised measured QoS
[0-20]	5
]21-40]	4
]41-60]	3
]61-80]	2
]81-100]	1

In Figure 4.5, the Risk Alerts Trust Agent is monitoring, in real-time, the received risk alerts levels (Rl_t) and the current service levels (service QoS (Sl_t) in order to detect events. In this example, both indicators, Rl_t and Sl_t, belong to the [0..100] range. In this case, $Rl_t = 0$ means no risk and a value of $Sl_t = 0$ means that the service QoS reached is minimum level or even that the service is not being provided anymore.

A Risk Alert Event is detected upon the occurrence of one of the following situations (Figure 4.5):

- The Quality of Service of a dependent service decreases bellow the defined threshold (in this case, the event ends when the QoS exceeds the threshold or, if in the meantime, an alert different from 0 is received, the event ends when the alert goes back to 0);
- After the reception of a risk alert message indicating a risk alert greater than 0 (this event ends when both indicators return to normal values).

Depending of the underlying system, the alerts may just be received when a change occurs in the indicator. In this case, the last received alert is considered actual and active until the next value arrives. In this scenario, a value of risk alert is always relevant within the Risk Alerts Trust Agent. The same can happen with the measured service level. If it is updated on a regular basis (a defined time interval), the last measured value is used for the evaluation. These assumptions can be seen as dangerous while evaluating trust for the received risk alerts by using out-dated information. Per se, this is certainly a drawback, as the following evaluated indicators

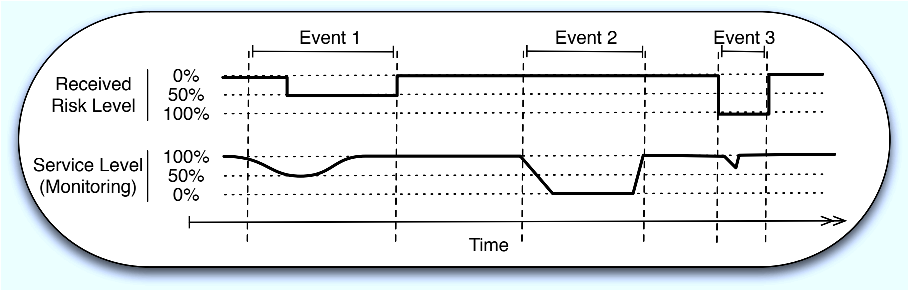


Figure 4.5: Event characterisation example

will not reflect the actual situation. This identified consequence is minimised with the use of the Behaviour Trust indicators as it will become apparent throughout this section.

For each event $A(Event_n)$, the accuracy is defined as the average of all comparisons made during the event (value T), between the observed service level (Sl_t) and the received risk level (Rl_t), as defined in Equation 4.1. As the function $f(Sl_t, Rl_t)$ is a discrete function, a sample rate, regarding the time factor, needs to be used. This sample rate can be different on each service and depends on the information available on the system. It is natural that a smaller sample rate yields more realistc observations.

$$A(Event_n) = \frac{\sum_{t=1}^{T} (f(Sl_t, Rl_t))}{T} \quad , \tag{4.1}$$

where $f(Sl_t, Rl_t) = |Sl_t - Rl_t|^{\kappa} \;\; , \kappa \in R^{+}$. The value k was introduced with the intent to penalise the larger differences or the small differences and should be assigned considering the degree of importance of each service. For instance, during an event, if the measured QoS of a service is always above the defined threshold (normal situation), it will make more sense to penalise more the risk alerts as $Rl_t = 100\%$ than the ones that refer a $Rl_t = 5\%$. By defining a value $k < 1$, it means that the TRS is willing to trust, even in the cases where large differences ($|Sl_t - Rl_t|$) are observed. Applying a value $k > 1$ the biggest differences will suffer a higher penalisation. In this approach, the duration of an event is not considered as the agent is only focusing on the accuracy of each received risk alert.

The satisfaction degree for each event is expressed by $A(Event_n)$ which results in a value within the [0..1] range. It is possible to interpret this value and, for example, to say that one particular alert was *very satisfactory* (1.0), *satisfactory* (0.6), *unsatisfactory* (0.2), or *very unsatisfactory* (0). Each $A(Event_n)$ value is sent by the agent to the TRS Discovery Tool in order to be incorporated within the CI/service Trust and Reputation indicators, as explained in Section 4.2.2.

Behaviour Trust Agents

An aspect that was particularly focused on throughout this thesis is the security issues that arise from a system with multiple CIs aiming to collaborate. Considering the MICIE project as an example of such a system, it is possible to denote the existence of several security entities composing the system. For instance, as presented throughout Chapter 3, the MICIE SMGW through the SMGW Manager is aware of possible security faults. Within the MICIE system, the SMGW is able to provide the collection and analysis of data related to security aspects, that can be useful in order to infer a trust indicator for each peer/service behaviour.

In this context, the possibility to use, among other, the information gathered from several security entities, was identified, in order to better understand the behaviour of the partnership. Also, an important aspect, to infer a more complete indicator aiming to improve the trust indicator related to the received risk alerts.

The information available in order to evaluate Trust and Reputation indicators regarding the behaviour of an external CI or dependent service, is firstly evaluated, within the Behaviour Trust Agents. As represented in Figure 4.6 it is possible to have one or more Behaviour Trust Agent within the TRS. The Behaviour Trust Agent's main goal, is to gather all types of information available, that might characterise the behaviour of a CI/service. In the TRS approach, all kind of information can be used to characterise the behaviour. This is achieved by designing a flexible approach for the existent agents. Indeed, the TRS is, in this case, focused on receiving behaviour events. Behaviour events are defined as being a type of abnormal event that is occurring in the system and is able to help characterise a service or a CI. Each behaviour event sent to the TRS, is composed of its origin and the respective trust level.

As mentioned, a behaviour event can be almost anything that is able to characterise the behaviour of a CI or a service, in one or multiple particular aspects. Figure 4.6 highlights some representative entities from which it is possible to gather behaviour information. One particular entity represented in the Figure 4.6 is the Behaviour Security Model. This virtual entity, that in fact, is integrated into each Behaviour Trust Agent, contains a representation of how the normal behaviour of the system

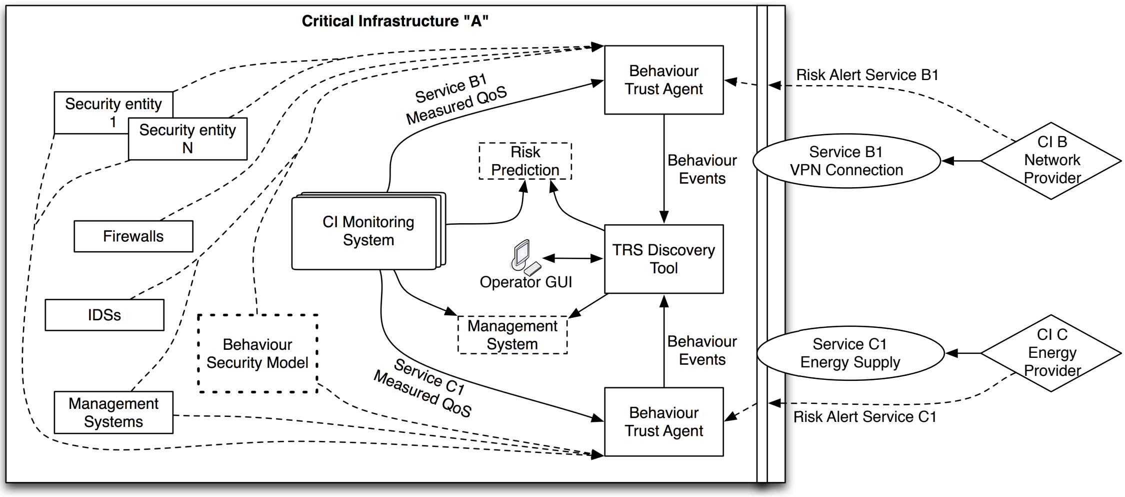


Figure 4.6: Behaviour Trust Agents example.

should be and also, a set of identified abnormal behaviours. Furthermore, it contains quantitative information, defined by an expert. This information defines how much one should trust a CI or a service, in case of the presence of a particular, previously identified, abnormal behaviour.

As an example of information that might be used to trigger behaviour events, it is relevant to explain why the service measured QoS and the received risk alerts are sent to the Behaviour Trust Agents. One should remember that this information is also gathered and evaluated by the Risk Alerts Trust Agents in order to infer the accuracy of each risk alert event. In the behaviour analysis, this information is analysed from a different perspective. In this case, aspects such as the quality or quantity of data may be analysed.

For example, suppose that it is established that one must receive a new risk alert message within every five minutes. If a risk alert is received within this time interval there is no abnormal behaviour. However, if no risk alert is received within the specified interval, the expert can, for instance, state that this fact indicates a decrease of 10% in the trust of behaviour of that alert. If no alert is received within one hour, the decrease can be greater.

The same approach can be applied to the service measured QoS from which it is also possible to apply behaviour observations. For example, a measurement of the service QoS must remain within the range of possible values previously defined. Receiving a value that is out off-range is seen an as abnormal event. These are just some examples among the multiple abnormal situations that can be defined by the CI expert, for the behaviour of the received alerts and for the service measured QoS.

Examples of other entities from which data can be collected are represented in Figure 4.6. Among these entities, the IDS can detect, for instance, an intrusion originated on a peer CI. This is clearly a behaviour event that must be evaluated in the trust one has on that peer behaviour. The firewall is also able to detect behaviour events, for instance, by identifying attempted connections to internal services. In particular, and from the example of MICIE project, the management system employed to manage the information sharing among participants, should be the primary source of information for the Behaviour Trust Agent. Due to the fact that it is assumed that this entity should aggregate, among others, all the system's security information. An example of such an entity is the MICIE SMGW, on which it is possible to evaluate trust on each CI behaviour, by considering, among other possibilities, all the interactions among peer CIs in terms of ICT security (internal or external). For instance, the events can be Intrusion Detection Systems (IDS) alerts, failed connection attempts, attempts to read/write information without permission.

As expected, the available data is gathered from heterogeneous sources, thus, it is anticipated that this data is received in multiple different formats and representations. Normalising and evaluating the received information, according to the behaviour security model that identifies relevant behaviour patterns, resolves this problem. The behaviour security model implements a simple but powerful and scalable approach. It consists of a set of tables, each one mapping the possible received values (or value ranges) and the correspondent Behaviour Trust, assigned by the expert, to that received value. One or multiple mapping tables are implemented in each specialised Behaviour Trust Agent.

For instance, according to a behaviour security model, it is possible to define that the existence of four failed authentication attempts within a system, occurred within a minute, will produce a confidence level of twenty on the behaviour security model (as exemplified in Table 4.2). Apart from being able to represent and to quantify foreseen behaviours, this model also acts as an adaptor between heterogeneous sources and the trust estimator algorithm. By employing these adaptors it is possible to infer trust indicators, as all security events are possible to be quantified and then used in a common calculation.

Each Behaviour Trust Agent collects proper behaviour information from the defined entities. Upon applying the relevant adaptor table, as defined in the Behaviour

Table 4.2: Behaviour Trust Agent - Adaptor Table Example

Failed Authentication Attempts/Minute		
Trust Indicator Level	Description	Received Values
100	No Failures	0
80	One/Three Failures	1-3
20	Four/Ten Failures	> 3 and < 10
0	More that 10 Failures	>= 10

security model, the resulting normalised trust value (behaviour event) is sent to the TRS Discovery Tool in order to be incorporated in the global Trust and Reputation indicators.

4.2.2 TRS Discovery Tool

The TRS Discovery Tool is responsible for the calculation, in real-time, of the Trust and Reputation indicators, as represented in Figure 4.3. For each main type of indicator (Risk Alerts and Behaviour Trust), the TRS Discovery Tool maintains and evaluates all the information received from the Agents. In particular, all current and past calculated indicators are stored in a database in order to provide them to the CI Operator. These indicators are also provided to the proper entities in order to enable accurate risk prediction. For instance, if applied to the MICIE project, the TRS Discovery Tool would provide this data to both the SMGW Manager and the Prediction Tool. In the following, the methodology used for Trust and Reputation evaluation, within the TRS Discovery Tool, is presented.

Trust and reputation indicators on received risk alerts

As represented in Figure 4.3, the TRS is able to evaluate two main Trust and Reputation indicators allowing one to reason about the confidence on the received risk alerts. The first indicator (Risk Alerts Trust) represents the confidence one has in the received risks related to one particular dependent service. This is, excluding the event accuracy, the most specific indicator evaluated. Second, an indicator is evaluated in order to describe the confidence one may have on the risk alerts received from one particular CI (Risk Alerts Reputation). This indicator incorporates the trust, related to all the services provided by each particular CI. Both indicators are able to incorporate the CI Operator's opinion as mentioned in Section 4.1.

In a simple approach, the trust that CI A has on the risk alerts received for the dependent service X, provided by CI B, is represented by $T_{(A,B,X)}$ and can be calculated by the average of the accuracy of each past event between those two CIs, regarding that particular service (Equation 4.2).

$$T_{(A,B,X)} = \frac{\sum_{i=1}^{N} A(Event_i)}{N} \quad . \tag{4.2}$$

As stated in previous work (Aime and Lioy, 2005; Spitz and Tuchelmann, 2009), this solution has already some identified weaknesses. For instance, it is possible for a situation to occur during which, one peer can behave correctly during a series of events and then capitalise the gained trust in order to send false alarms. These problems occur mainly due to the fact that the trust value will change slowly as it depends equally on all the past transactions. This weakness must be minimised. One approach to minimise it, is the introduction of the ageing concept. Essentially, the ageing concept, allows the evaluation to give different weights to older and recent events (Spitz and Tuchelmann, 2009). In this context, the TRS employs a discount factor D, allowing it to give more weight to the recent received events. The ageing factor should always depend on the context and should be assigned by an expert while considering the specific characteristics of each service. Within the TRS, it is required to define the ageing factor on a per peer/service basis.

In this context, the Equation 4.2 can be improved and the trust that CI A has on service X provided by CI B, $T'_{(A,B,X)}$, is computed for the Nth event as presented in Equation 4.3:

$$T'_{(A,B,X)} = \frac{(D * (N-1) * T_{(A,B,X)}) + A(Event_N)}{D * (N-1) + 1} \quad . \tag{4.3}$$

The ageing factor D is defined with a value belonging to the [0..1] interval. In this evaluation, a small value of D causes the importance of the recent events to increase, while a value of D close to one, provides less ageing to the oldest events and consequently increases their contribution to the evaluation. By increasing the ageing factor, the previous identified problems that led to the introduction of this factor, are reintroduced. There are several approaches for selecting the adequate ageing factor D. For instance, it is possible to use a fixed value defined by the CI expert. It is also possible to make the factor D decay exponentially, for instance by using a $D = f(t) = x^t$ $(0 < x < 1, t = 1..N)$. Another approach could be, for instance, to use a methodology similar to the one presented by Aime and Lioy, and use the observation on the partner behaviour instability and focus on more recent alerts, when observed behaviour reveals strong time correlation (Aime and Lioy, 2005). In the actual implementations of the presented TRS, it is possible to define a fixed value for D.

In Figure 4.7 it is possible to observe the influence value D has on the calculation. In the plot represented in the Figure, a set of events are evaluated using three different D values in order to calculate trust. The previously stated comportment induced to the trust indicator by the ageing value D, is clearly visible while comparing the results obtained with $D = 1$ and $D = 0.1$. It is noticeable that the indicator that uses the smallest D value, has a quicker reaction to changes within the risk alert events values.

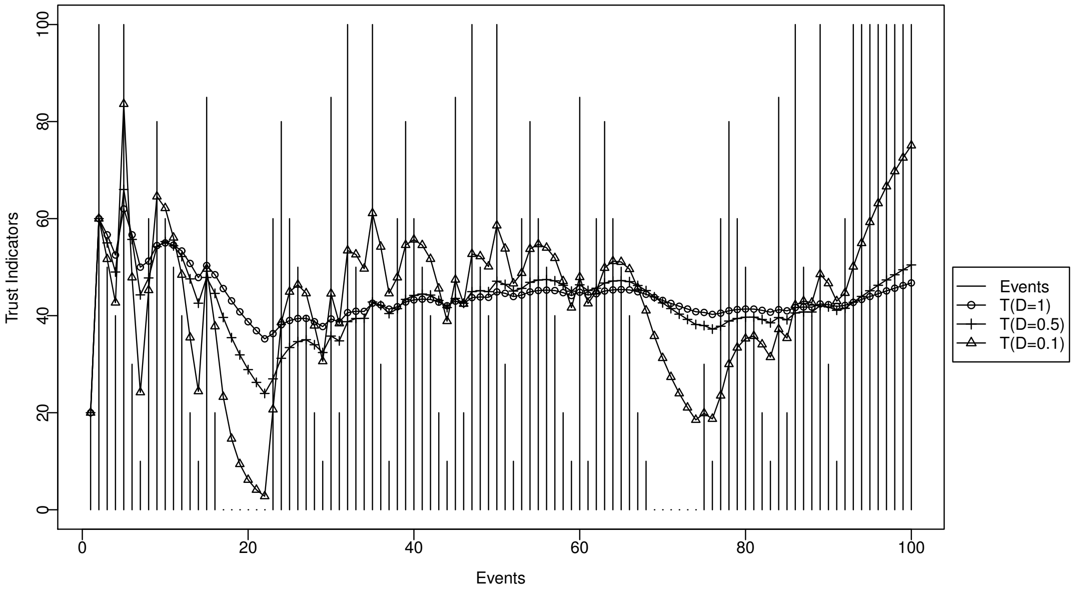


Figure 4.7: Example of the ageing factor (D) influence

As discussed in Section 4.1, the TRS is able to incorporate a human factor in the trust evaluation. In the context of this thesis it is expected that this human factor or opinion relates to the CI Operator opinion. The CI Operator's opinion can be introduced in two situations: to initialise the trust indicator when there are no past observations available or at any moment, reflecting the CI Operator opinion and contribution to the trust calculations. In the second case the weight this contribution has on the calculation, needs to be specified. Considering the human factor, the final trust value for a specific CI service is defined in the Equation 4.4.

$$T(final)_{(A,B,X)} = (1-\alpha)(T_{(A,B,X)}) + \alpha(TO_{(A,B,X)}) \ \ , (0 < \alpha < 1) \ . \tag{4.4}$$

For the Equation 4.4, the α factor is assigned by the CI Operator depending on the confidence he has in the value $(TO_{(A,B,X)})$ that represents his trust on service X. The resultant value, $T(final)_{(A,B,X)}$, represents the TRS confidence in the received alerts for each service individually, taking into account also the CI Operator perspective. In order to better understand how all evaluated indicators evolve over time, and also to allow defining a relation among them, a time value is associated with each evaluated $T(final)$.

After evaluating the alert trust for each service, the reputation of each involved CI can be computed. To allow the evaluation of this indicator, it is necessary to assign a weight to each service. This weight is assigned by the CI Operator and should represent the relevance each service has within the set of services provided by each CI. This information should be defined on the existent CI models on which, each (inter)dependent service should have been analysed and weighted.

Upon the services weighing, the reputation of each CI is evaluated by applying Equation 4.5. In this evaluation, $GT'_{(A,B,t)}$ represents the reputation that CI A has about CI B at time t. $GT_{(A,B)}$ represents the last evaluated indicator. S represents the number of services that A receives from B and W_i is the weight associated to each i service provided by CI B. $T(final)_{(A,B,i)}$ represents the last service risk alert indicator, available for service i. N is the number of evaluations already accomplished. D is the ageing factor defined for each individual CI reputation indicator. The reputation indicator, as defined in Equation 4.5 is evaluated every time a service risk alert indicator changes.

$$GT'_{(A,B,t)} = \frac{(D*(N-1)*GT_{(A,B)}) + \frac{\sum_{i=1}^{S}(T(final)_{(A,B,i)}*W_i)}{\sum_{i=1}^{S} W_i})}{D*(N-1)+1} \ . \tag{4.5}$$

As represented in Figure 4.3, the CI Operator is also able to contribute to the reputation indicator evaluation, by integrating a value representing his opinion. Equation 4.6 evaluates the final indicator $(GT(final)_{(A,B,t)})$ with the inclusion of the CI Operator's opinion regarding each specific CI. In Equation 4.6, θ is assigned by the CI Operator and denotes the confidence he has regarding the subjective reputation value $TO_{A,B}$ that he includes in the indicator.

$$GT(final)_{(A,B,t)} = \theta(TO_{A,B}) + (1-\theta)(GT_{(A,B)}) \ \ , (0 < \theta < 1) \ . \qquad (4.6)$$

It is important to note that it is not mandatory for the CI Operator to incorporate his opinion within the Trust and Reputation indicators evaluation. If this opinion is not intended to be incorporated, the Equations 4.4 and 4.6 must be configured with a value of 0 for the parameters that represent the weight assigned to the CI Operator's opinion.

Trust and reputation indicators on peers behaviour

According to the representation in Figure 4.3, the TRS is also capable of evaluating two main Trust and Reputation indicators, expressing the confidence one may have on CI's or service's behaviour . The first indicator (Behaviour Trust) represents the trust one CI has on the behaviour of one particular dependent service. Second, an indicator is evaluated in order to describe the Behaviour Trust one may have on the behaviour observed from one particular CI (Behaviour Reputation). This indicator incorporates the perceived behaviour of all the services provided by each particular CI and also other possible observations, related to the CI, that are independent from the services (e.g. ICT security observations). Both indicators are able to incorporate the CI Operator's opinion as mentioned in Section 4.1.

Although the evaluation method used to evaluate Risk Alerts Trust and Behaviour Trust is very similar, a major difference exists in the type of data used to evaluate trust. As presented in the previous section, the behaviour of a CI or dependent service can be evaluated using all information existent in the defined Behaviour Security Model. As described, adaptor tables exist in order to help the agents to translate one specific situation to a behaviour event, with a trust value associated. In this context, behaviour events and risk alert events are similar, as the respective agents, detect or are informed of an event, evaluate the event accuracy and send this value to the TRS, in order to aggregate it in the Trust and Reputation indicators. Indeed, the major encountered difference arises from the fact that, if something or someone is behaving as expected, it is not probable that someone will complain about that circumstance. In fact, most of the entities that are able to help understand the system behaviour, might just raise events when they detect some uncharacteristic behaviour.

Considering the possible existent entities used to gather behaviour information (Figure 4.6) and in particular, the security management and monitoring systems, it is expected to receive behaviour events, only when misbehaviour is detected. For instance, the existent network IDS usually just triggers an alert/event when an intrusion or tentative of intrusion is detected. The same happens within the management systems. It is unusual for an authentication request, accepted on the first, to be logged as a security issue.

The fact that, it is anticipated to receive behaviour information mostly when misbehaviour is detected, leads to a situation in which the received events are almost all adverse to the trust in the behaviour. By considering only these events on a simple statistical evaluation, it is anticipated that the results of such an evaluation, in the majority of the cases, will indicate a low Behaviour Trust value, thus not representing the complete peer behaviour. In this case, the atypical behaviour is always considered while the normal behaviour is ignored if the entity does not notify it.

It is possible to reconfigure all entities present in the system, enabling them to notify all types of behaviour, including also the normal behaviour. However this would imply a major change within the CI and thus, leading to a more difficult implementation. To overcome the problem while considering also the normal behaviour, the concept of Inactivity was introduced.

Within the concept of Inactivity, it is assumed that a scenario on which, no behaviour events were received during a certain period of time - Inactivity - indicates that, during this period of time, the behaviour of the observed entity was appropriate. In a simple way, it is assumed that if the TRS does not receive any behaviour information during a specified period of time, it will assume that the behaviour was normal and considers, for the trust evaluation, the trust value defined for the normal behaviour in the respective adaptor table.

In order to consider the existence of inactivity periods, the time is divided into a set of time slots (Spitz and Tuchelmann, 2009), each slot with Δt duration. If inactivity exists during one slot, it is assigned the proper normal behaviour value to that slot. If some event is received during one slot, the slot value is assigned with the average of all events received during that slot. The Behaviour Trust value for each time slot, $Event_{(Slot\ s)}$, is calculated from Equation 4.7 on which $NEvents_{(Slot\ s)}$ and N both represent the number of events observed on a particular entity, within the duration of the $Slot\ s$.

$$Event_{(Slot\ s)} = \begin{cases} 100\texttt{, if } NEvents_{(Slot\ s)} = 0 \\ \\ \frac{\sum_{i=1}^{N} Event_i}{N}\texttt{, if } N = NEvents_{(Slot\ s)} > 0 \end{cases} . \qquad (4.7)$$

Figure 4.8 represents seven time slots with each respective event value as evaluated using Equation 4.7. On each slot are also represented each individual events that were received within the slot period.

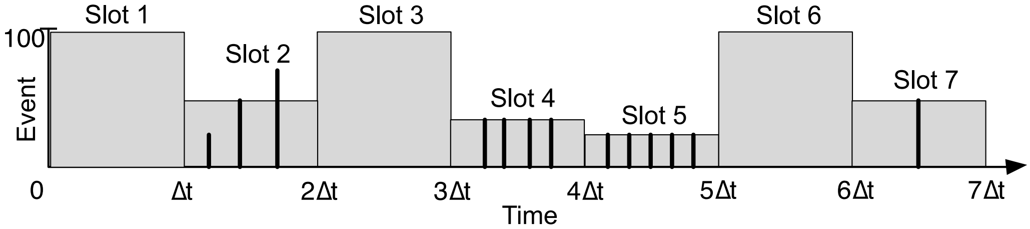


Figure 4.8: Time slots example

The duration of each time slot, Δt, needs to be defined by a CI expert, according to the specificities of each security entity (e.g. firewall, IDS, specified period during which it is expected to receive risk alerts, etc.) and it is possible to allow it to represent a period of only a few seconds or even hours. A larger Δt value, implies slow changes to the trust indicator, this is more evident when only a few events are received over time. In that case it will be better to choose a small value for Δt. Examples of how the chosen Δt value can affect the obtained results can be observed in Figure 4.9. In this Figure, the two graphs on top display one observation with more values over time than the one observed on both graphs at the bottom.

For the time slot s, the trust on the behaviour of the entity E related to CI B or to a particular service provided from CI B, ($T'_{(E,B,s)}$), is calculated using Equation 4.8 on which, D is the ageing factor (similarly employed as described for the risk alerts trust), $T_{(E,B)}$ is the indicator evaluated for the previous slot ($s-1$) and $Event_{(Slot\ s)}$ is the event trust value for the slot s. All the evaluated indicators are stored in the TRS database on which the time they were evaluated is also associated to each one.

$$T'_{(E,B,s)} = \frac{(D * (s-1) * T_{(E,B)}) + Event_{(Slot\ s)})}{D * (s-1) + 1} . \qquad (4.8)$$

As described, it is also possible, although not mandatory, to include the CI Operator's trust on the behaviour observed at each entity. This human factor is important while evaluating the behaviour of the system as it allows the CI Operator to, at each

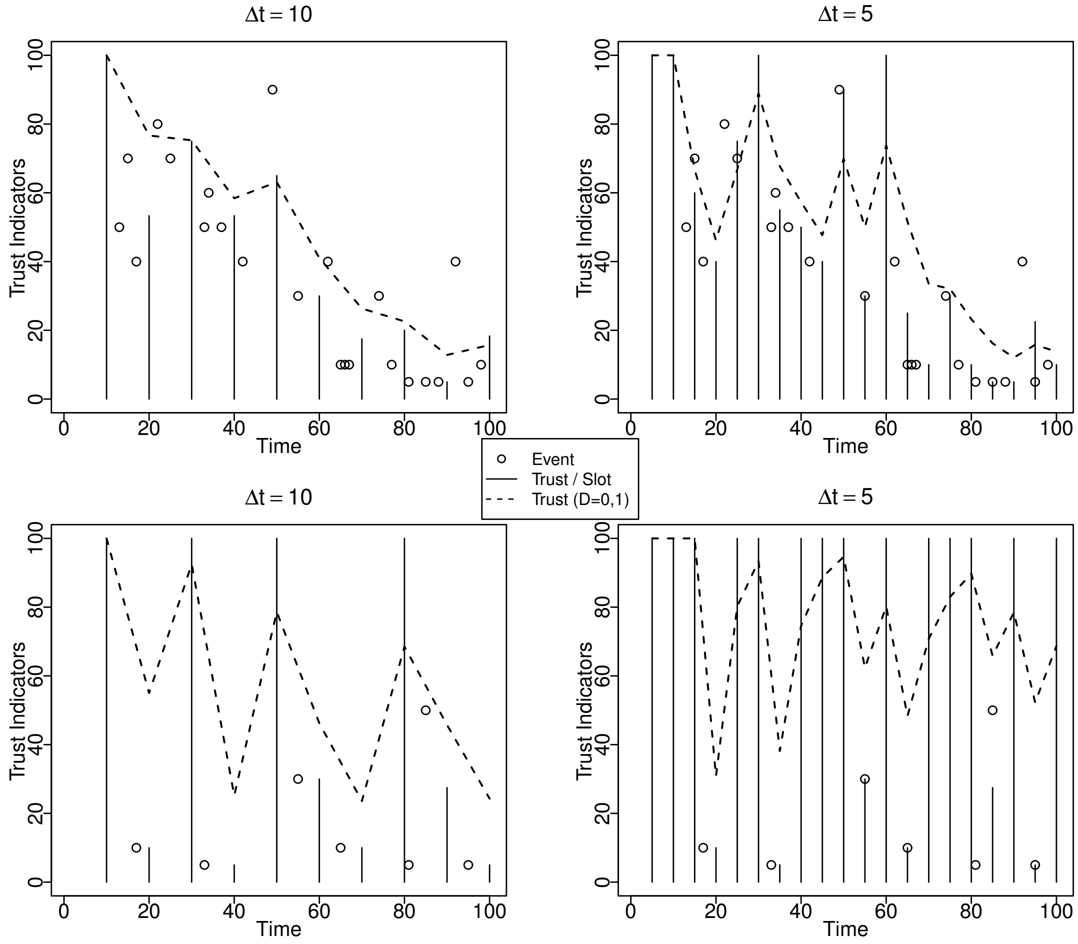


Figure 4.9: Example of the influence of the time interval duration (Δt)

moment, fine tune the final Behaviour Trust indicator. This fine tuning might be needed, for instance, if the CI Operator is aware of the occurrence of some particular behaviour that lowers the trust indicator, however due to his knowledge his aware of the fact that it should not be used to lower the confidence in the CI or service.

Equation 4.9 allows to include the CI Operator's trust in CI B / service behaviour concerning security entity E. The α factor is assigned by the CI Operator representing the confidence in his subjective trust ($TO_{(E,B)}$).

$$T(Final)_{(E,B)} = \alpha(TO_{(E,B)}) + (1-\alpha)(T_{(E,B)}) \ \ ,(0 < \alpha < 1) \ . \tag{4.9}$$

As the event values are normalised (according to the description in Section 4.2.1), it is also possible to evaluate an indicator encompassing all types of behaviour events related to one particular CI or service. This indicator characterises the Behaviour Reputation.

A weight must be assigned for each entity from which the behaviour is inferred. Each weight should represent the relevance each entity has to the global behaviour of the analysed CI or service. For simplicity reasons, in Figure 4.3, the reputation is just represented for each CI, although the behaviour reputation can be evaluated for each CI or / and for each dependent service.

The behaviour reputation is evaluated by applying Equation 4.10. In this Equation, $TBehaviour'_{(B,t)}$ represents the reputation of CI B (or one particular service) behaviour at time t. W_i is the weight associated to each security entity i. $TBehaviour'_{(B)}$ represents the last evaluated reputation indicator for the evaluated CI or service. The assigned weight should be defined along with the definition of the Behaviour Security Model, representing the relevance of each entity in maintaining security in the three considered main aspects, confidentiality, integrity and availability. An ageing factor D is also included in the evaluation.

$$TBehaviour'_{(B,t)} = \frac{(D*(t-1)*TBehaviour_{(B)}) + \frac{\sum_{i=1}^{E}(T(Final)(i)*W_i)}{\sum_{i=1}^{E} W_i})}{D*(t-1)+1} \quad . \tag{4.10}$$

The CI Operator is also able, if necessary, to contribute to the indicator with his knowledge about the global behaviour of one particular CI or service. In a similar manner as the previously described indicators, this is achieved by using Equation 4.11, in which the θ factor is assigned by the CI Operator, representing the confidence on his subjective trust indication ($TO_{(B,t)}$) at time t.

$$TBehaviour_{(Final)(B,t)} = \theta(TO_{(B,t)}) + (1-\theta)(TBehaviour_{(B,t)}) \ \ , (0 < \theta < 1); \ . \tag{4.11}$$

Both indicators, the Risk Alerts Trust and the Behaviour Trust, can be composed in a global indicator, representing the trust one has on the received risk alerts and in the behaviour of a service. This can also be achieved for the Reputation indicators. Combining both indicators is achieved by applying Equation 4.12. The CI expert needs to assign the weight each indicator has in the global indicator $((0 < \theta < 1), (0 < \alpha < 1))$.

$$TGlobal_{(B,t)} = \theta(TBehaviour_{(Final)(B,t)}) + \alpha(GT(final)_{(A,B,t)}) \quad . \tag{4.12}$$

The described Trust and Reputation System has been initially defined for application within the MICIE system. In order to verify the TRS's contribution to the SMGW Manager and to the MICIE Prediction Tool, several simulations intended to validate the approach were carried out. In the following, some validation results are described.

4.3 Validation

The validation work presented in this section was carried out within the scope of the MICIE project. As stated, the author of this thesis has actively contributed to the development of MICIE's management system, namely, the SMGW Manager. In this context, the TRS was introduced in order to improve the system's management with the introduction of trust and reputation indicators. These indicators allow gathering a deeper knowledge of the data exchanged among MICIE's participants and also of the behaviour of the peers during the communication process.

The TRS was not initially foreseen in the MICIE project and it was proposed as an add-on within the SMGW Manager. In order to validate the TRS's applicability to the MICIE system, several validations were carried out by simulating possible scenarios supported by the architecture presented in Figure 4.10, in which is distinguished the TRS. In the following examples, the interactions among two Critical Infrastructures with one dependent service among them are simulated.

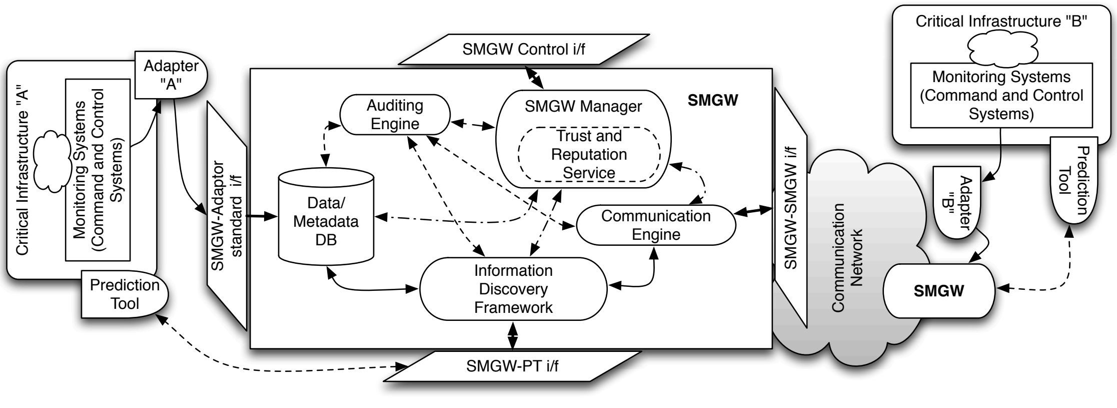

Critical Infrastructure "A"
Monitoring Systems (Command and Control Systems)
Adapter "A"
Prediction Tool
SMGW-Adaptor standard i/f
SMGW Control i/f
SMGW
Auditing Engine
SMGW Manager
Trust and Reputation Service
Data/ Metadata DB
Communication Engine
Information Discovery Framework
SMGW-PT i/f
SMGW-SMGW i/f
Communication Network
Critical Infrastructure "B"
Monitoring Systems (Command and Control Systems)
Adapter "B"
Prediction Tool
SMGW

Figure 4.10: MICIE overall system and SMGW architecture

The simulations were firstly conducted using the statistical simulation tool R (R Development Core Team, 2009). The results obtained from the simulations were also helpful while validating the Java based applications that implement the TRS.

Several simulation scenarios were developed and tested (Bertoni et al., 2010a; Caldeira et al., 2010b,d). One subset of those scenarios is described in Table 4.3, representing the following situations:

- (S1) The system behaves as expected with only small discrepancies observed between the received risk alert and the service measured level. In this situation, the accuracy of each event is always above 60% and mainly between 90% and 100%;

- (S2) The system is not accurate but can still be trustworthy, as evaluated event accuracy is always above 40%;

- (S3) According to the measured service level, the received alerts are not as expected. In this case, 60% of the events have an accuracy lower than 20%, while in the remaining events, an accuracy value higher than 60% is never observed;

- (S4) The system is not trustworthy as 90% of the events have an accuracy lower than 20%.

Table 4.3: Simulation Scenario (average of the differences between measured service level and received risk alerts)

Event Interval	Scenarios			
	S1	S2	S3	S4
[0-10]	0	0	40	80
]10-20]	0	0	20	10
]20-30]	0	0	10	5
]30-40]	0	0	10	5
]40-50]	0	10	10	0
]50-60]	0	10	10	0
]60-70]	5	10	0	0
]70-80]	5	10	0	0
]80-90]	10	20	0	0
]90-100]	80	40	0	0

% of occurence

Trust in Received Risk Alerts

In order to simulate the Risk Alerts Trust indicator, the event's accuracy was generated from random numbers produced in R (R Development Core Team, 2009), in accordance to the conditions defined for each scenario, as described in Table 4.3.

According to the presented TRS framework, the following parameters are used for the simulation: penalisation factor $k = 2$; ageing factor $D = 0.3$; a threshold of 10% meaning that the observed differences between received risk alerts and measured service level are considered correct when within the threshold.

Figures 4.11 present the Risk Alerts Trust indicators obtained from simulating the existence of 1000 events generated respectively according to the defined scenarios 1 to 4. It is clear that, for each scenario, the Risk Alerts Trust indicator will tend to the value corresponding to the average of the accuracy of the generated events. It is important to note that in the considered worst scenarios (S3 and S4), the trust indicator drops below the average of the events accuracy. This fact highlights the relevance of the chosen value for the penalisation factor ($k = 2$). In these scenarios (S3 and S4), as the difference between measured service level and received risk alerts is higher, these events are heavily penalised due to the chosen k.

The first presented simulation exemplifies a simple scenario, on which a possible fault within the CI providing the service, exists. In this scenario, the accuracy of each event is high (S1) for most of the simulation period, with exception of one small period on which the accuracy of the received risk alerts is the lowest defined within the proposed situations on Table 4.3 (S4).

For this first simulation, two slightly different scenarios are described in order to validate that the framework acts as expected, independently of the amount of events received. In this context, the one simulation considers 5000 events, while the other considers just 50 events. It is relevant to test the framework for such a different number of events, because in a real life situation with two CIs cooperating in order to provide better services to its clients, it is not expected to receive a large number of different risk alerts, except within exceptional conditions.

Figure 4.12(A), represents the results obtained for the Risk Alerts Trust indicator for the first presented simulation. In this case, the first observed 2000 events (Figure 4.12(A)) were generated according to scenario (S1). In these first events, the indicator, as denoted also by Figure 4.11, tend to approximate to average value of the used scenario. Following this, the next 1000 events represent a substantial

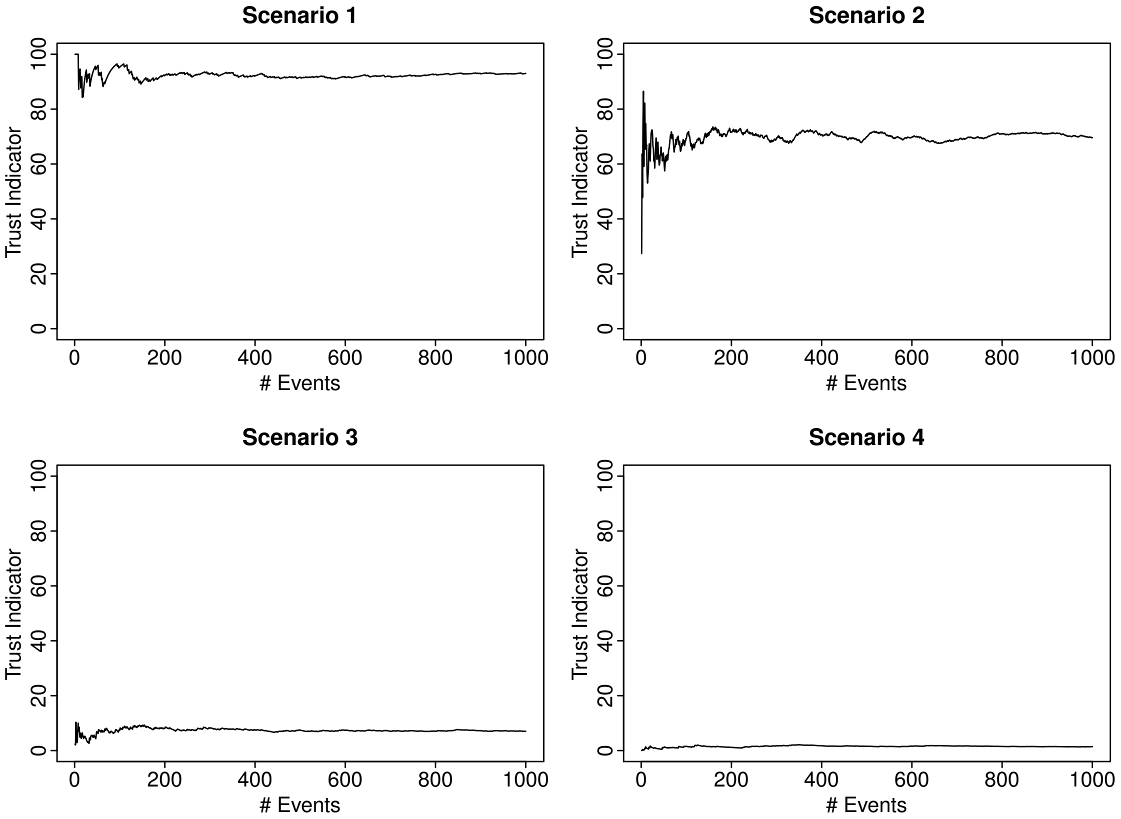


Figure 4.11: Risk Alerts Trust indicator for the scenarios 1 to 4

degradation of the event's accuracy (S4). During this period of 1000 events, it is perceptible that the indicator rapidly decreases due to the influence of the newer situation. In this case, the trust one has on the received alerts is below the mean value and would allow the CI Operator to assign less weight to those alerts in its own risk evaluation. After the event 3000, the indicator starts to grow gradually as, from this point to the end of the simulation, the event's accuracy is again within scenario (S1).

The simulation observed in Figure 4.12(B) intends to demonstrate the applicability of the indicator even with a small number of observed events. In this case it is possible to observe that with just 50 events the observed indicator has basically the same results as with 5000 events. In this example, the incorporation of the CI Operator is illustrated. Assuming that the CI Operator received reliable information that the fault affecting the supplier CI has been solved, he acts by assigning a value for his trust as being 90% and by defining the weight of his contribution to the final indicator, equal to 0.8. With this contribution (human factor), the CI Operator is

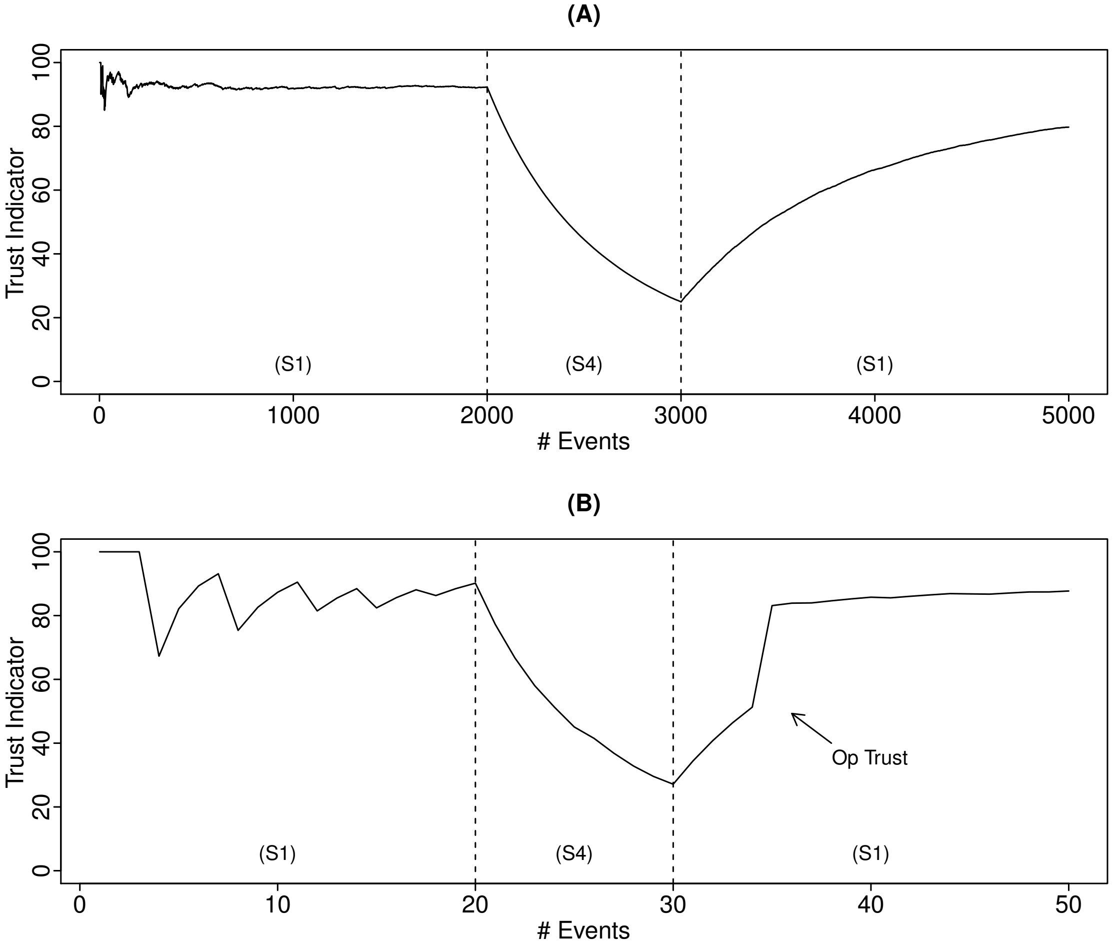


Figure 4.12: Simulation 1 - Trust on Received Alerts. (A) 5000 events / (B) 50 events

able to rapidly regain the trust in a CI or service, as represented in Figure 4.12(B), by the existent indicator after the event indicated by the existent arrow.

A second simulation is presented (Figure 4.13) in order to allow the observation of the evaluation for the Risk Alerts Trust indicator during a situation on which the accuracy on the received events is changing frequently.

The second simulation (Figure 4.13(A)) represents a situation on which the events generated according to Table 4.3 change after each 100 events. In this simulation, on the first 100 events, the accuracy of the events is defined according the scenario (S1) and then abruptly changes to the worst scenario (S4) during the succeeding 100 events. It is visible that this change is almost immediately incorporated in the trust indicator as expected. After the 100 events obtained from scenario (S4),

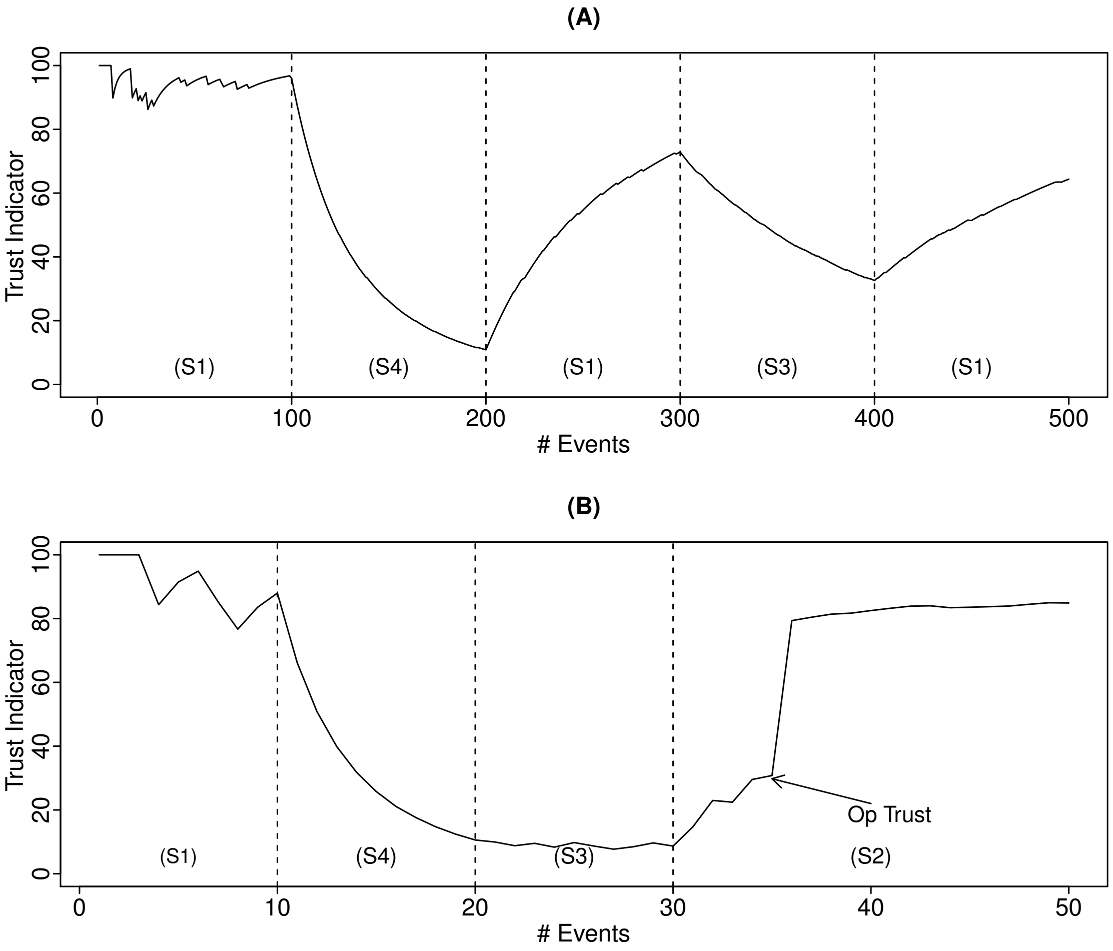


Figure 4.13: Simulation 2 - Trust on Received Alerts. (A) 500 events / (B) 50 events

the trust indicator clearly indicates that one should not trust the received alerts for this particular service. After 200 events the received risk alerts become, again, reliable. As the ageing factor is used, the trust indicator rapidly incorporates the new scenario. From the 300th to the 400th event, the scenario changes to (S3), thus decreasing the trust indicator (in this case the indicator decreases more slowly that in the last case). This simulation highlights the ability that the trust indicator has, to rapidly react when the scenario changes. It is also clear that even in case of an abrupt change of the event's accuracy, the indicator changes gradually due to the ageing factor.

In the simulation presented in Figure 4.13(B)), less events are used than in the previous simulation (Figure 4.13(A)), showing that the TRS is still accurate even with a small number of received events. In this simulation, the received alerts are unreliable between the 20th and the 30th event. This leads to a very low trust value that gradually starts to grow after the 30th event (grows gradually as the received events are based on scenario (S2)). As previously explained, the CI Operator can incorporate his opinion on the indicator. In this case, the CI Operator assigned a value for his trust equal to 90% and defined a contribution of 0.8 to the trust indicator. Although the indicator continues to incorporate changes from the events, the final value will be higher. It should be noted that it is important for the CI Operator to know the consequences of this action as his contribution to the indicator continues to be applied until it is updated or removed.

As discussed, the Risk Alerts Reputation indicator incorporates the trust evaluated for each service provided by one particular CI. This enable us to understand the confidence one may have in the alerts received from one CI. Although the Risk Alerts Trust indicator should be more useful for improving risk prediction, as this is a more detailed indicator, the reputation on the received alerts is of more use within the system management. For instance, in the MICIE SMGW Manager, one can define security policies that are triggered in case of a decrease in the Risk Alerts Reputation of a CI.

Figure 4.14, presents the results obtained in the third simulation, on which two services provided by the same CI are evaluated, as well as the reputation indicator. The simulation was implemented using information observed from two separate services, each service receiving an average of 5 events per hour. As the main goal of this simulation is to highlight the Risk Alerts Reputation indicator, the events generated to simulate each service, were obtained from a combination of the scenarios presented in Table 4.3. In order to evaluate reputation, the CI Operator assigned a weight of 0.7 to service 2 and 0.3 to service 1. A value of $D = 1$ was used for the ageing parameter. From the simulation results, it is perceptible, as expected, that when the most weighted service becomes unreliable, the CI reputation decays, even when the other service is trustworthy.

Trust on Peers Behaviour

In order to formulate scenarios which are able to simulate a CI or service behaviour, it is important to know the distribution over time of the simulated behaviour events. For the following described simulation, the arrival time for the events is generated from random numbers produced in R (R Development Core Team, 2009). The function developed in R, generates random values representing an average of x events

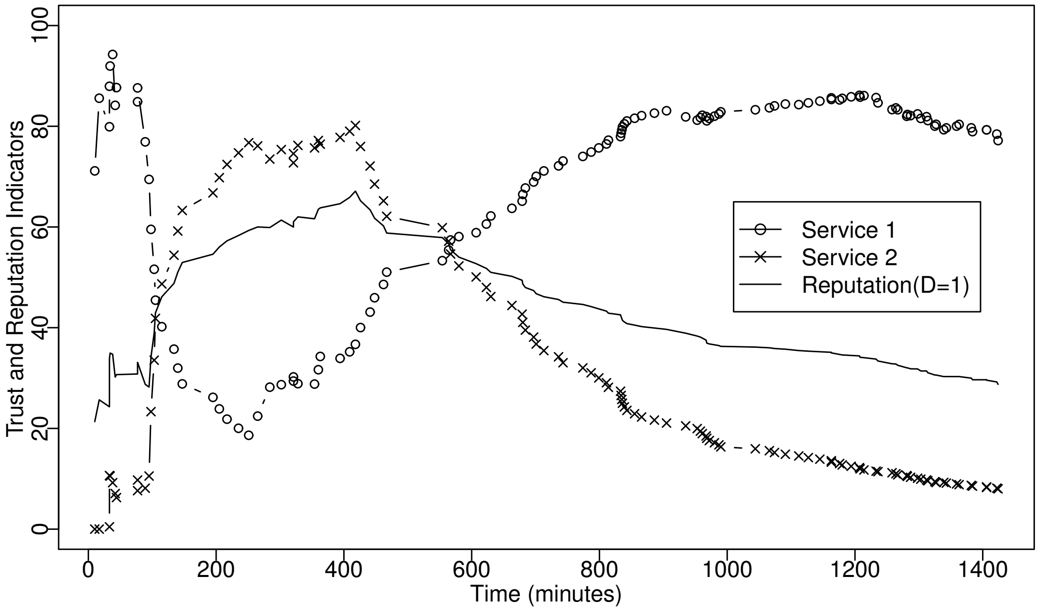


Figure 4.14: Simulation 3 - Risk Alerts Reputation

per hour. The values assigned to each simulated event are generated based on the scenarios expressed in Table 4.3.

The following described simulations are intended to evaluate the Behaviour Trust indicator for a single provided CI/service. The TRS parameters for the Behaviour Trust are common to all the simulations and are as follows: time slot size $\Delta t = 10$, ageing factor $D = 0.05$ and a simulation total time of 24 hours (1440 minutes). As it is not supposed, in normal circumstances, to receive behaviour events, it is important to choose a small ageing factor as the one chosen, in order to allow the indicator to rapidly incorporate the situations.

In Figure 4.15(A), the behaviour events were generated according to the scenario (S3) defined in Table 4.3. These events occur with an average frequency of 5 events each hour. As the events occur at a relatively low frequency, the trust indicator begins with no defined tendency. In this scenario, as the events arrive at an averaged constant frequency, each one with a value from (S3), the value will tend, gradually, to around 50%. It is possible to denote that even with some detected incorrect behaviour, the indicator raises in the periods without events due to the introduction of the inactivity concept.

In the simulation presented in Figure4.15(B), the first half (first 12 hours) of the

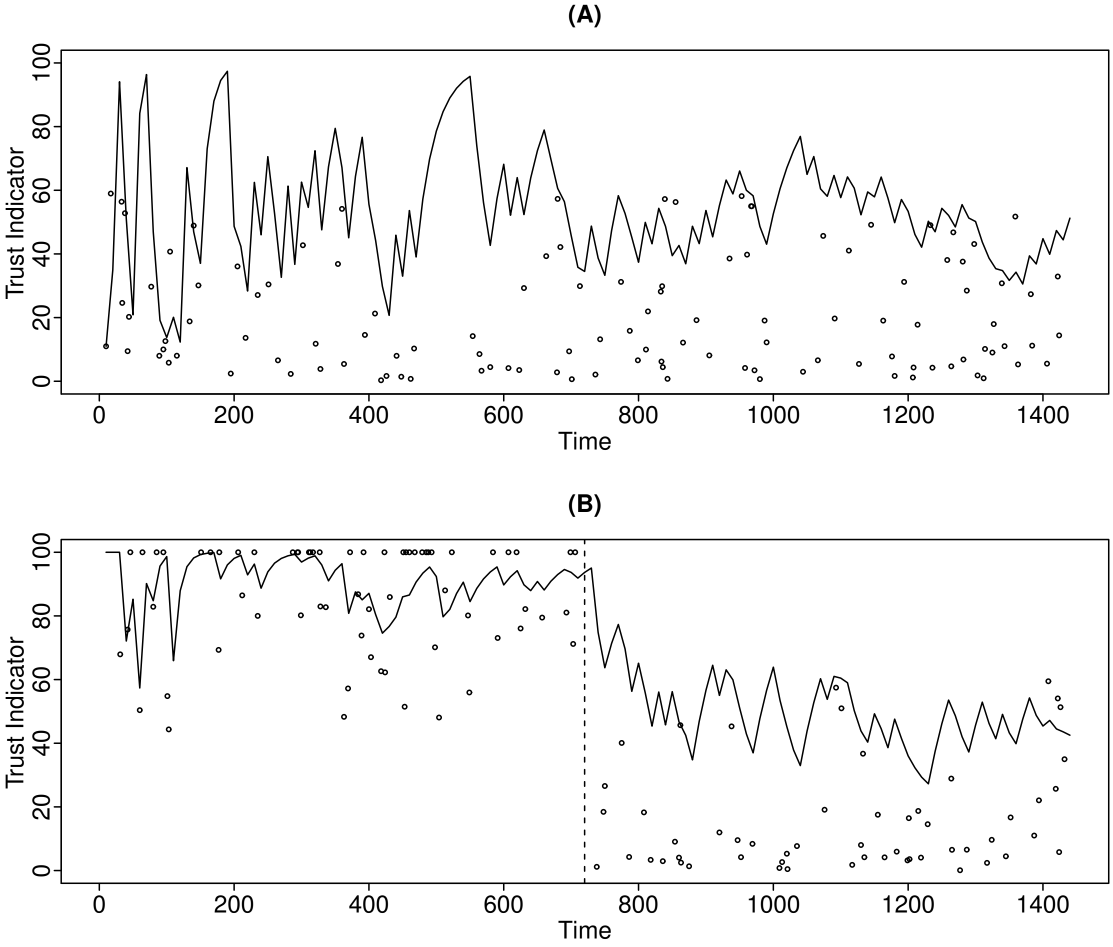


Figure 4.15: Behaviour Trust - Simulation 1

events are valued according to the scenario (S2) and the last half (last 12 hours) according to scenario (S3) (from scenarios presented in Table 4.3). The arrival rate for the generated events is 5 events/hour. The results achieved with this simulation allow us to observe that, due to the ageing value, each half of the simulation seems independent from the other, demonstrating that the Behaviour Trust indicator is able to rapidly incorporate the changes happening in the CI/service behaviour.

In the following simulation (Figure 4.16(A)) , the events are generated at a rate of 1 event per 60 minutes. The value assigned to each of the events is computed from within multiple scenarios available in Table 4.3, namely, scenarios (S1), (S3) and (S4). In this case, as just a few events are simulated, the Behaviour Trust indicator does not drop below 60%. This is due to the influence of the slots on which the system is behaving normally. This simulation aims to demonstrate the importance of the value assigned for the time slot. In this case, a wider time slot would lead to a lower Behaviour Trust indicator. It is also important for the CI Operator to know how the defined parameter influence the TRS, in order to allow an improved understanding of the received indicators and in order to properly configure the system.

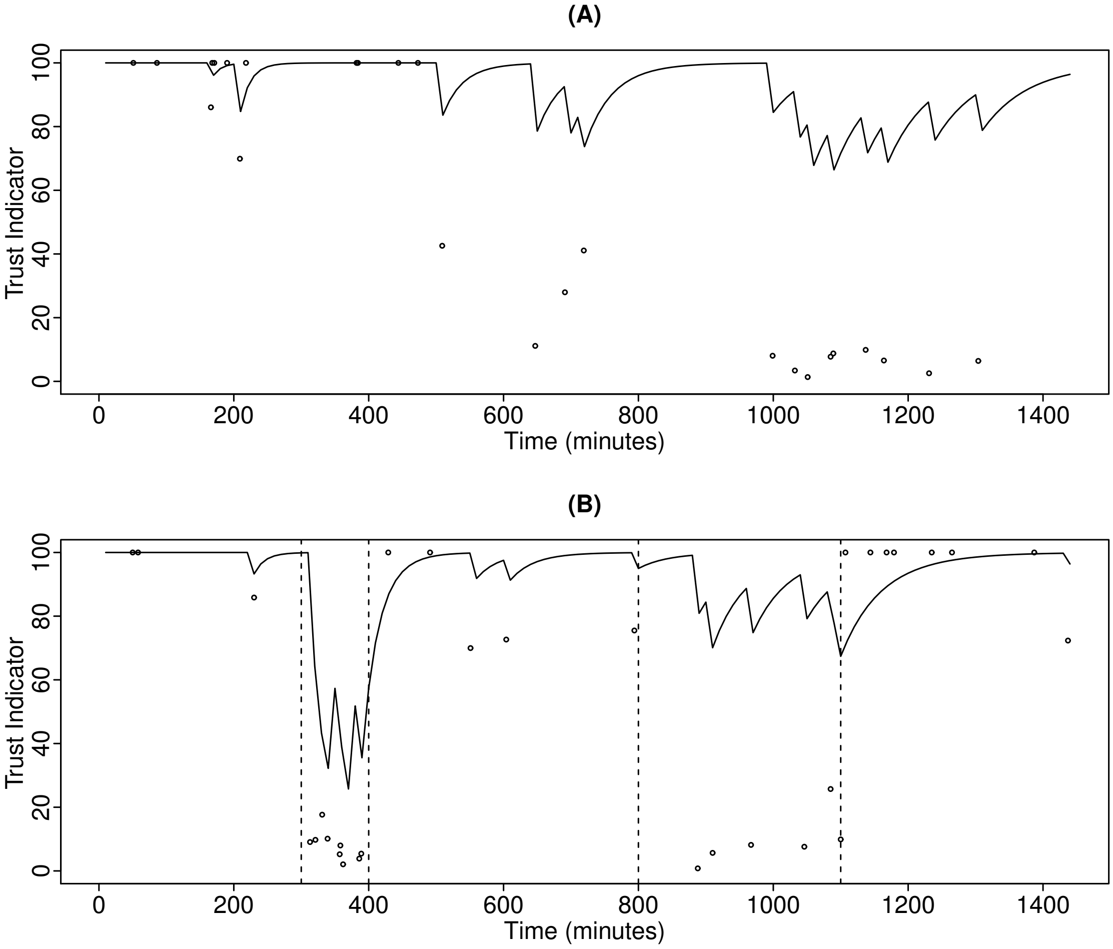


Figure 4.16: Behaviour Trust - Simulation 2

The results obtained when aiming to simulate a situation with two possible induced attacks or misbehaviours is presented in Figure 4.16(B). In this simulation, on the first 300 simulated minutes, events generated according to scenario (1) occur at a rate of 1 each 60 minutes. Following this, during a period of 100 minutes, the scenario changes to (S1) (the worst scenario), and the events occur more frequently with an event rate of 5 events/60 minutes. When the scenario changes it is noted that the Behaviour Trust indicator rapidly decays below 50%, thus clearly indicating that something is wrong. Next, the behaviour events are simulated according to (S2) and with a lower frequency allowing the indicator to increase. Between the 800th and 1100th minutes, the scenario changes to (S4). During this period, the events occur at a rate of 1/60 minutes. It is noticeable that, even with the occurrence of just a few events, the CI Operator would be able to infer that the CI/service behaviour is not normal. For instance, according to this indicator, a management system (e.g. the MICIE SMGW Manager) could act by, for instance, blocking system access for that particular CI. The final simulated period represents events generated according to the scenario (S1), occurring at a rate of 1 event/60 minutes. In this scenario, with a lower event rate, the Behaviour Trust indicator clearly indicates the resolution of the past situations.

According to the presented results, it is clear that the indicators gathered within the TRSs are within the expectations, allowing to enhance CIs risk prediction by incorporating a trust value on each received risk alert. Also, the behaviour of a peer CI or service is able to be monitored allowing to contribute to the evaluation of trust in received risk alerts and also to evaluate an indicator regarding multiple types of observations on CIs/services.

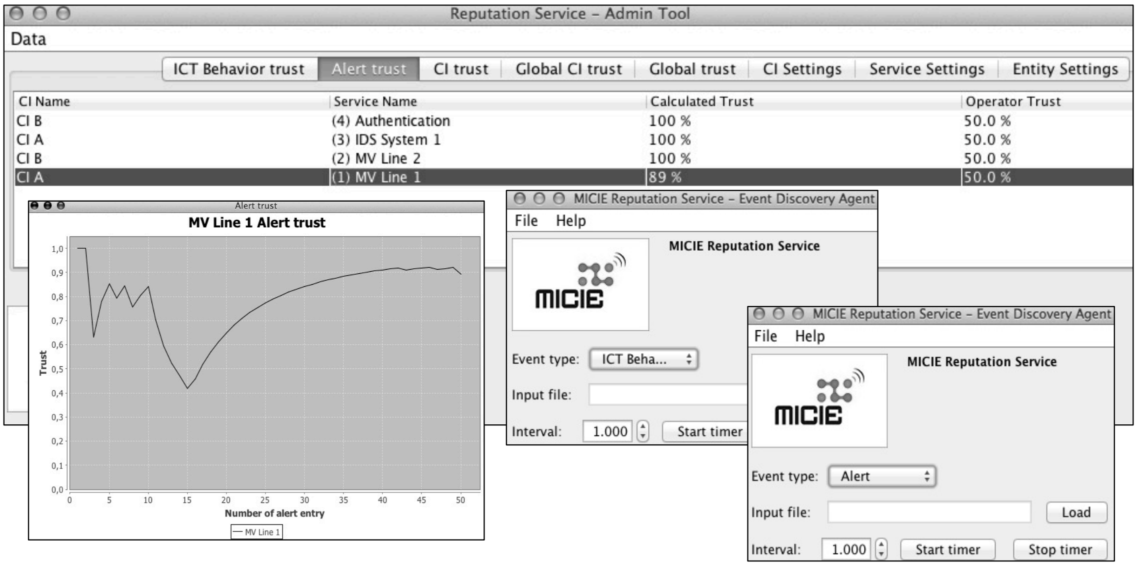
4.4 TRS proof of concept applications

The described TRS is implemented within two different approaches. First, a proof of concept application was developed within the MICIE project and the second implements the TRS within with the CI Security Model as described in Section 2.3.

The first TRS's implementation aimed to allow integration within the MICIE SMGW Manager. It implements the TRS as defined in Figure 4.3. All application modules are written in Java and communicate by exchanging XML messages through Web Services. Both the Risk Alerts Trust Agent and the Behaviour Trust Agent, are able to receive XML messages containing the needed data to evaluate events. They are able to work in push or pull mode, by also retrieving information from a Web Service when they need it.

In this implementation, the TRS Discovery Tool is implemented as a service, receiving events from the agents and evaluating the TRS indicators in real-time. For instance, the time slots in the Behaviour Trust are automatically evaluated within the TRS. The Discovery Tool implements a Web Service able to provide current or past indicators to a client. In this case, possible clients are the MICIE Prediction

Tool, the MICIE SMGW Manager or the CI Operator GUI. Figure 4.17 represents several screenshots of the TRS application. In the picture an overview of the Operator GUI, two agents and a generated graph are visible.



CI Name	Service Name	Calculated Trust	Operator Trust
CI B	(4) Authentication	100 %	50.0 %
CI A	(3) IDS System 1	100 %	50.0 %
CI B	(2) MV Line 2	100 %	50.0 %
CI A	(1) MV Line 1	89 %	50.0 %

Event type: ICT Beha...
Input file:
Interval: 1.000

Event type: Alert
Input file:
Interval: 1.000

Figure 4.17: MICIE TRS implemented tools.

The Operator GUI retrieves data from the TRS Discovery Tool (refreshed manually or in defined intervals) and displays all information within a simple interface. It allows the CI Operator to: configure all the TRS parameters, insert, update or delete CIs, services or entities, among other operations. Although it is possible to have a global vision of all the indicators, the developed GUI is also able to draw graphs of the available indicators (Figure 4.17).

One interesting aspect of this implementation is that it was also designed with the aim to function as a simulation tool. It is possible to load values (with timestamps) from XML files into the agents in order to simulate the existence of events. The developed tool is then able to integrate a system like MICIE and also to serve as a powerful simulation tool in order to evaluate Trust and Reputation among CIs.

The second TRS tool is a Java application developed as a proof of concept, implementing the framework described throughout this section within different purposes. In particular, this tool is intended to allow the use of the TRS along with the CI Security Model described in Section 2.3. The tool is able to represent a scenario using the CI Security Model as illustrated in Figure 4.18 and to evaluate the Risk Alerts Trust and the Behaviour Trust indicators. The tool is also able to receive real time data from CIs through the use of Web Services or to act as a simulator by reading data records from XML files.

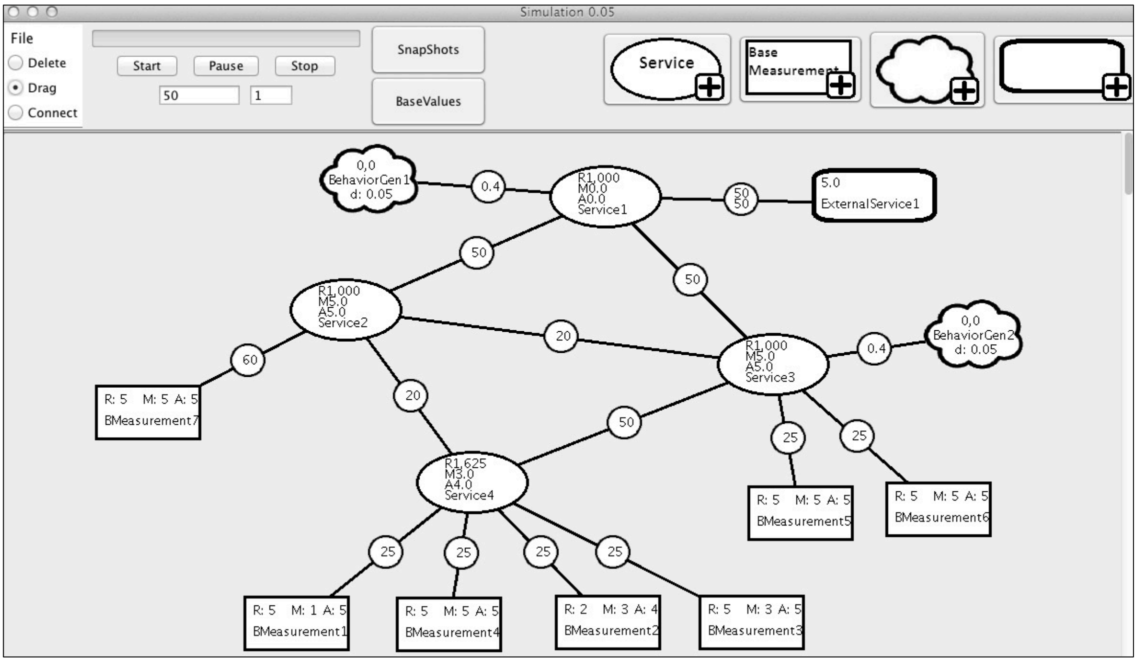

Simulation 0.05
File
○ Delete
● Drag
○ Connect
Start
Pause
Stop
50
1
SnapShots
BaseValues
Service
Base Measurement
0,0 BehaviorGen1 d: 0.05
0.4
R1,000 M0.0 A0.0 Service1
50 50
5.0 ExternalService1
50
50
R1,000 M5.0 A5.0 Service2
20
R1,000 M5.0 A5.0 Service3
0.4
0,0 BehaviorGen2 d: 0.05
60
R: 5 M: 5 A: 5 BMeasurement7
20
50
25
25
R1,625 M3.0 A4.0 Service4
R: 5 M: 5 A: 5 BMeasurement5
R: 5 M: 5 A: 5 BMeasurement6
25
25
25
25
R: 5 M: 1 A: 5 BMeasurement1
R: 5 M: 5 A: 5 BMeasurement4
R: 2 M: 3 A: 4 BMeasurement2
R: 5 M: 3 A: 5 BMeasurement3

Figure 4.18: TRS implementation tool for the CI Security Model.

This last tool supports the application of the TRS to the CI Security Model as it is described in the following Chapter 5. Figure 4.18 presents a screenshot of the developed tools on which are visible the main CI Security Model components, namely, the service, base measurements, behaviour entities and external services. The application GUI has been developed aiming an improved usability, incorporating drag and drop possibilities and an easy way of updating all the TRS's and Model's parameters. It is also possible from this tool, to export the evaluated results in XML format and also in graphical format.

4.5 Summary

In a context in which Critical Infrastructures are combining efforts in order to achieve improved risk estimations, by exchanging risk information to their dependent CIs, allowing them to incorporate the received risks within each local risk evaluation tool, several aspects need to be addressed. Among these aspects, the information exchange security needs to be assured. Assuming that each CI employs proper security mechanisms, service QoS measurement equipment and also proper risk prediction and evaluation tools, the CIs through the CI Operators face a new challenge. This challenge is how to answer questions such as: “How can information used for risk calculation be evaluated for correctness?”, “How are the existing measurement tools behaving?”, “How is one peer CI acting in terms of ICT security?”.

In this chapter, a Trust and Reputation framework was introduced aiming to allow the incorporation of Trust and Reputation indicators on the information exchanged among Critical Infrastructures and also on information coming from heterogeneous monitoring equipment. Although the presented framework is independent from the MICIE project, as it can be applied in multiple contexts, the MICIE application scenario for the framework was described.

The framework implementation (Trust and Reputation System) was described, including the description of the existent components, namely, the Risk Alerts Trust Agent, the Behaviour Trust Agent and the Discovery Tool. An example of integration of this system within CIs has been shown. The indicators obtained from the introduction of the TRS are presented. In particular, the indicators gathered according to the received risk alerts and the measured QoS (Risk Alerts Trust and Risk Alerts Reputation). In order to complement the risk alerts indicators and to enable the estimation of the system behaviour (related to one particular CI or CI service), the TRS is also able to infer trust regarding the behaviour of multiple entities present on the system, including the behaviour of the received risk alerts. These indicators (Behaviour Trust and Behaviour Reputation) were introduced and discussed throughout the chapter.

Upon the TRS’s description, the evaluation of the TRS by simulation was presented, in different possible scenarios. It showed that the TRS produced correct indicators for Trust and Reputation, within the simulated situations. The developed tools that implemented the TRS were also presented in this chapter.

The presented TRS showed the ability to improve a system’s and theCI Operator’s capacity to deal with uncertainty, and to fulfil its mission, in a timely manner, for instance, in the presence of attacks, failures, or accidents. The TRS might be applied in multiple scenarios within Critical Infrastructures. Examples of such scenarios, for which the TRS was adapted in order to contribute to the risk estimate mechanisms improvement within a Critical Infrastructure, are detailed in the following chapter.

Chapter 5

Application Scenarios and Validation

The Trust and Reputation System (TRS) presented in Chapter 4 has been proposed and implemented within the MICIE projects and validated using simulation. Although this system was out of the MICIE scope, the TRS has been proposed as an add-on to the project validation activities (Lev et al., 2011).

Throughout this work, one of the faced drawbacks was the lack of information to properly validate the TRS. In the MICIE project it was not possible to have access to information exchanged over time, as the validation activities were focused on the accuracy of the models, the Prediction Tool and on the proper operation of the SMGW and its components. Another aspect that is intended to be validated is the applicability of the TRS within different application scenarios and supported by different CI modelling approaches.

To better validate the TRS, a joint work has been carried out with Thomas Schaberreiter, author of the CI Security Model presented in Section 2.3. The main goal was to be able to improve both works, validating the TRS while enhancing and extending the CI Security Model. This was achieved by adding Trust and Reputation components as a means to improving its accuracy and its resilience to inconsistent information provided by dependent CIs and allowing to evaluate the correctness of information received from those dependencies.

In this chapter three application and validation scenarios are presented for the Trust and Reputation System, supported by the CI Security Model proposed by Thomas Schaberreiter.

The first described scenario, incorporates a new information source (trust) in the original CI Security Model in order to allow a dynamic adjustment of the weight that the risk levels received from dependencies have on the evaluated risk within the CI Security Model.

The second scenario, combines the CI Security Model assurance levels with trust indicators. Originally, the CI Security Model assurance levels were defined by an expert and are not intended to dynamically change. In this second scenario, the trust indicators are used to evaluate the behaviour of the entities on which the assurance levels are assigned in order to adjust the global service assurance level indicator.

The third scenario, incorporates the concepts introduced in the first two scenarios and validates the proposed approaches within a more realistic test-bed application scenario - The Grid 5000 project.

The first two application scenarios were validated using simulation. The last application scenario uses real data and intends to prove the applicability of both contributions (TRS and the CI Security Model) to a real test-bed application scenario – The Grid 5000 project.

The following contributions were published respectively in (Caldeira et al., 2011), (Schaberreiter et al., 2011b) and (Caldeira et al., 2013).

For the following application scenarios and validation, the author of this thesis contributed with the work on Trust and Reputation Systems while Thomas Schaberreiter contributed with his work on the CI Security Model.

The adaptation of the Risk Alerts Trust indicator and the Behaviour Trust indicator from the TRS (proposed by the author of this thesis) as well as the adaptation of the CI Security Model (proposed by Thomas Schaberreiter) both need to be seen as equal contributions to the validation work. They have resulted from many discussions enabling the combining of the CI Security Model and the Trust and Reputation System.

Also, the evaluation of how CI service risk alerts, received from dependencies, can be evaluated for correctness and how Risk Alerts Trust and Behaviour Trust can be adapted for this purpose, are seen as an equal contribution by the author of this thesis and Thomas Schaberreiter.

The TRS was adapted by the author of this thesis, for each application scenario as well as to be included in the CI Security Model. Another contribution of the author was the definition of the evaluation and validation possibilities within the Grid'5000 project. The author of this thesis carried out all the experiments for the case studies including simulation, data adaptation and the development/implementation of the tools used in this work.

5.1 Trust based dependency weighting

The work on CI security modelling, presented in Section 2.3, establishes a CI model based on the risk that enables on-line risk monitoring in interdependent CIs. As mentioned, the motivation of this model is to decompose the complexity of CIs into smaller and abstract entities, to be able to compare dependent CIs and incorporate them in the risk estimation. In this model, special attention is given to information sharing between dependent CIs that can belong to different providers.

Modelling techniques used in Critical Infrastructures in order to infer risks, usually rely on information received from sensors in the field and on information shared among dependent services. Thus, a scenario where that information is missing or incorrect, leads to wrong assumptions about risk. In this context, it is important to use mechanisms able to evaluate the correctness of the information used for risk calculation. As the originally proposed CI Security Model has no reasoning mechanism about the exchanged information, the following question remains unanswered "How can shared information be evaluated for correctness?". In order to answer this question, the Trust and Reputation System was introduced allowing to improve the CI Security Model and also validate the applicability of the TRS.

In this first scenario, it is proposed to evaluate the information received from a dependency, based on observations about that dependency. Depending on the outcome of such an evaluation, a CI Operator can decide to what extent the information received from the dependency will be incorporated into the CI risk evaluation. A way to carry out evaluation is to build a trust relationship between CI services through the Trust and Reputation System and use the trust level to evaluate the correctness of received CI information. A shortcoming of applying Trust and Reputation Systems to the domain of CIs is the variety of CIs. Each infrastructure can have different information to compare and evaluate. Building a Trust and Reputation system, taking into account several dependencies can be a quite complex task.

Merging both CI Security Model and Trust and Reputation System, allows the introduction of a way of building a trust relationship among CIs, based on the common abstract information they share. In the next section, the method of gathering information used in the CI Security Model is presented. The method of calculating trust that is used in the CI Security Model to dynamically re-evaluate the impact of a risk level received from a dependency has on the modelled risk in a CI, is also presented.

The following describes how the Risk Alerts Trust indicator (presented in Section 4.2) can be integrated into the CI Security Model. The information sources that can be used by the Risk Alerts Trust Agents to collect information are different for each CI sector and therefore, each CI has to be evaluated separately to gather this information. In the CI Security Model such information is represented in an abstract and uniform way (C,I,A) that is the same for each CI. In this context, it is presented how abstract and uniform information about CI risks can be gathered using the CI Security Model and how Risk Alerts Trust can be calculated using this information.

5.1.1 Risk alert events

In order to be able to evaluate the trust that each CI has on received risk alerts (Risk Alerts Trust), an entity has to be found that can be compared with each risk alert for the evaluation of its correctness. In the example seen in Figure 5.1, a service of a telecommunication provider ("server room") receives risk alerts from a service of a dependent CI ("Low Voltage distribution service"). If a high risk alert is received regarding the energy supply, the telecommunication provider needs to evaluate the correctness of this value. From the infrastructure decomposition in the risk assessment step of the CI Security Model, a service can be defined by the telecommunication provider ("Main power supply level") that allows to monitor the current energy level using the telecommunication provider equipment (e.g. voltage meter). Gathering this information, local risk indicators can be aggregated, which can be compared with the received risk alert. It is important to note that the example in Figure 5.1 indicates that the same approach of risk information gathering, can also be applied between two dependent services of the same CI ("server room air conditioning" and "server room", "room temperature" used to evaluate correctness of risk alerts) and between a service and a sub-service ("UPS" and "server room", "UPS energy supply level" used to evaluate correctness of risk alerts).

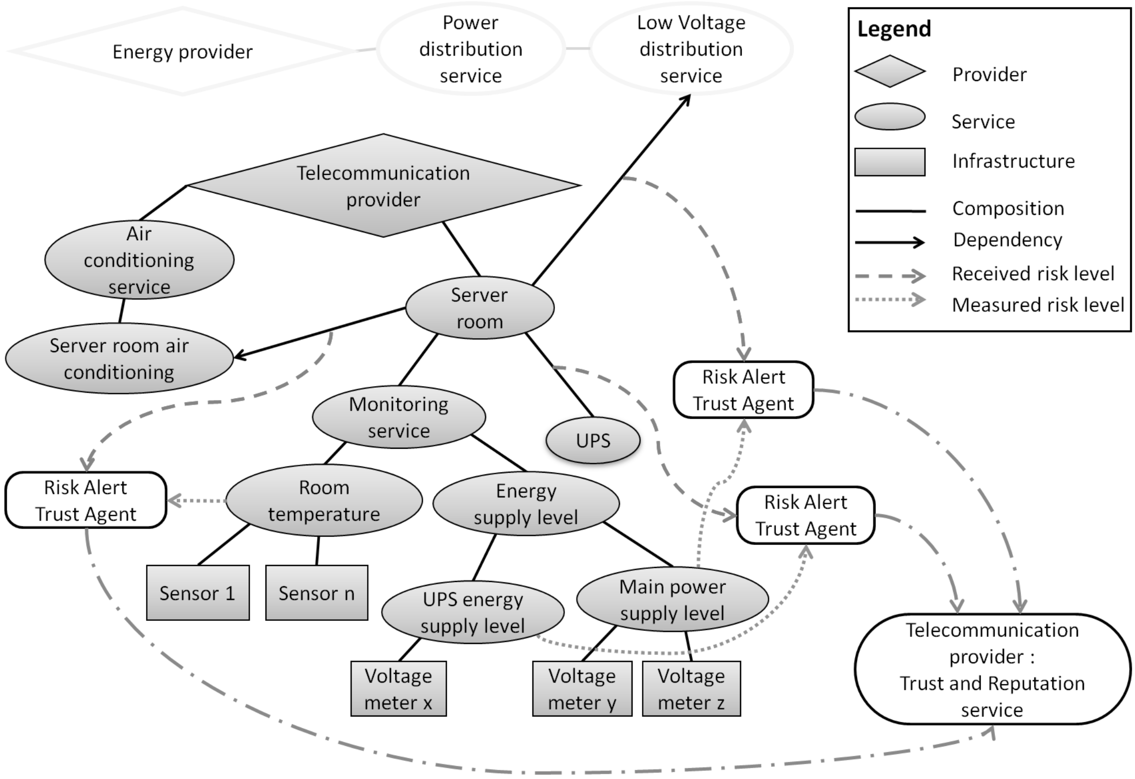


Figure 5.1: Risk alert indicators (Caldeira et al., 2011)

In order to evaluate trust aspects correlated to the received risk alerts, the primary goal is to define an accuracy value for each received risk alert. For this purpose, the concept of Risk Alert Event introduced in Section 4.2.1 is used - An event starts when one or both risk indicators (received risk alert and measured risk levels) are different from one (no risk). The event ends when both indicators drop to one.

The Risk Alerts Trust Agent is monitoring the risk alert levels (Rl_t) and the current measured risk levels (Ml_t) in order to detect events. Rl_t and Ml_t belong to the [1..5] range. The accuracy of each event $A(Event_n)$ is defined as the average of all comparisons made during the event (value T), between the measured risk level and the received risk alert level (Equation 5.1). As described in Section 4.2.1, function $f(Ml_t, Rl_t)$ is a discrete function so a sample rate for the time factor is needed. This sample rate can be different for each service and will depend on the information available on the system.

$$A(Event_n) = 100 - \left(\frac{\sum_{t=1}^{T} f(Ml_t, Rl_t)}{T} * 100 \right) \quad , \tag{5.1}$$

where $f(Ml_t, Rl_t) =\mid \frac{Ml_t - Rl_t}{4} \mid^k, k \in R^+$. The value k allows to penalise the larger differences or the small differences. The duration of an event is not considered, only focusing on the received risk alert accuracy is necessary.

As detailed in Section 4.2.2, the trust that CI_A has in risk alerts received for service X provided by CI_B is represented by $T_{(A,B,X)}$ and is calculated by the average of the accuracy of each past event between those two CIs for that particular service (Equation 5.2). The concept of ageing is used, applying a discount factor D, to give more weight onto recent events. The ageing factor should always depend on the context. In this scenario, the ageing factor needs to be defined on a per CI peer/service basis. In this example, $T'_{(A,B,X)}$ is computed for the Nth event as:

$$T'_{(A,B,X)} = \frac{(D * (N-1) * T_{(A,B,X)}) + A(Event_N)}{D * (N-1) + 1} \quad . \tag{5.2}$$

D is a value in the [0..1] interval and a small value of D will raise the importance of the last events while a value of D near 1 will provide less ageing for the oldest events.

A human factor reflecting the CI Operator opinion and contribution to the trust calculation is also considered in trust evaluation (Equation 5.3).

$$T(final)_{(A,B,X,t)} = \alpha(T_{(A,B,X)}) + (1-\alpha)(TO_{(A,B,X)}) \tag{5.3}$$

The factor α is in the range [0..1] and assigned by the CI Operator depending on the confidence he has in $(TO_{(A,B,X)})$. $T(final)_{(A,B,X,t)}$ represents the TRS confidence in risk alerts while also taking into account the CI Operator perspective. In order to understand how the Risk Alerts Trust indicators evolve over time, and to define a relation among them, a time value is associated with each $T(final)$.

5.1.2 Incorporating Trust in the Security model

As stated, the introduction of the TRS allows the overtaking of one shortcoming of the CI Security Model where, in is first definition, the weights for dependencies and sub-services are manually assigned by CI experts and are thus prone to human errors. The approach of trust based weighting, allows to calculate trust for the risk alerts received from each (inter)dependent CI or CI service and to combine the calculated trust with the initial weights assigned by experts. This result is a more precise estimate of the influence one service has to another. In this application example, the Risk Alerts Trust derived for a received dependent service risk is utilised ($T(final)$) as the trust indicator for a dependent service x ($T(x)$) (Caldeira et al., 2011).

In order to associate $T(x)$ with the dependency weight, the meaning of the weights had to be changed when compared with the original CI Security Model. In the original approach a weight assigned by an expert, represented the influence of a dependency to a service. This value has to be as accurate as possible. In the context of the TRS this weight now has to represent the maximum assumed influence a dependent service can have to a service. According to the current Risk Alerts Trust provided by the dependent service, this initial weight can be lowered accordingly. This is represented by Equation 5.4, where $\omega(x)$ is the newly calculated weight for the dependency x, ω_E is the original weight assigned by an expert and T is the Risk Alerts Trust for the dependency or sub-service x. The expected result of this calculation is a dependency weight ω in the range]0..100]. Note that 0 was excluded from the range. The presumably rare events where no trust in the correctness of the received risk levels ($T(x) = 0$) would produce a weight $\omega(x) = 0$ have to be treated separately, for example by excluding the risk alerts received from this particular dependent service from risk level aggregation. Allowing the existence of a dependency weight of 0 could cause problems in the risk level aggregation of the CI Security Model and possibly result in a division through 0 (Aubert et al., 2010a).

$$\omega(x) = \frac{\omega_E(x) * T(x)}{100} \quad , \omega_E(x), T(x) \in [0..100] \ . \tag{5.4}$$

5.1.3 Validation Results

As already described, the Trust and Reputation System presented in this work has already been submitted to some evaluation tests using simulation (Caldeira et al., 2010b). In this example, the same simulation tools are used in order to validate the applicability of the TRS to the CI Security Model.

The presented simulation focuses on the two scenarios described in Table 5.1. The simulated scenarios pretend to represent the following situations: (S1) The system behaves as expected with only small errors with the average of the differences between measured risks and received risk alerts mainly between 0 and 1; (S2) The system is inaccurate as the received risk alerts and the measured risk differences are mainly between 3 and 4.

Table 5.1: Simulation Scenario (average of the differences between measured and received risks)

Scenarios	$(Ml_t - Rl_t)$/Event			
	[0..1]	]1..2]	]2..3]	]3..4]
S1	90	5	5	0
S2	0	5	5	90

For the following simulation, the events represented in Figure 5.2 were generated as random numbers in R (R Development Core Team, 2009). The following parameters are used for the TRS: penalisation factor $k = 1.25$; ageing factor $D = 0.3$.

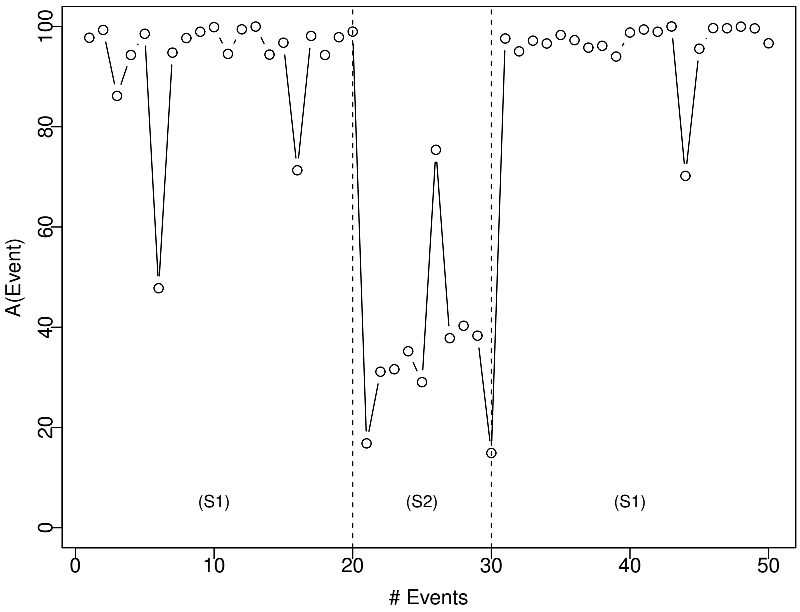


Figure 5.2: $A(Event_n)$ values

The simulated events presented in Figure 5.3, aim to represent a faulty component situation. As observable in Figure 5.3, the first 20 events belong to S1. Next, the received risk alerts become inaccurate (S2) during 10 events returning to its normal behaviour after that (S1). It is visible that the Risk Alerts Trust indicator decreases rapidly and the next starts to grow gradually. Figure 5.4 describes how the Risk Alerts Trust indicator contributes to the weight that the received risk alert will have in the CI Security Model. In this case, the expert has given a maximum weight of 80% to this risk. With the application of Equation 5.4, the final weight (expert*trust) value will change gradually depending on the Risk Alerts Trust indicator.

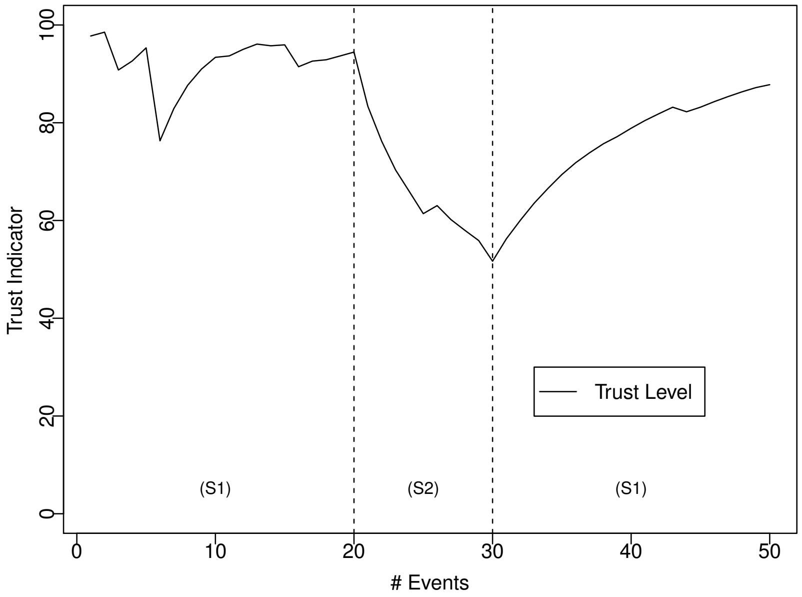


Figure 5.3: Risk Alerts Trust indicator

In Figure 5.5 is represented the contribution that the Risk Alerts Trust indicator and the expert weight have on the final weighted risk level. For the simulation the received risk alert was fixed to 5 and the measured risk level was varying leading to the Risk Alerts Trust indicator presented in Figure 5.3. In this scenario, receiving only a risk alert of 5, the weighted risk changes according to the expert opinion. In this case never reaching or exceeding 80% of the received risk alert, depending on the initial expert weight and the current trust. When the Risk Alerts Trust lowers, less importance to the received risk alert is given, maintaining a low risk level.

It might not seem perceptive that the weighted risk level in Figure 5.5 decreases when the Risk Alerts Trust decreases. The weight represents the impact a risk alert received from a dependency has to the aggregated risk of a service. A low trust in the correctness of risk alert values received from a dependency, means that its importance to the service should be lower. Therefore, high-risk alerts received from this dependency represent only a low risk for the service.

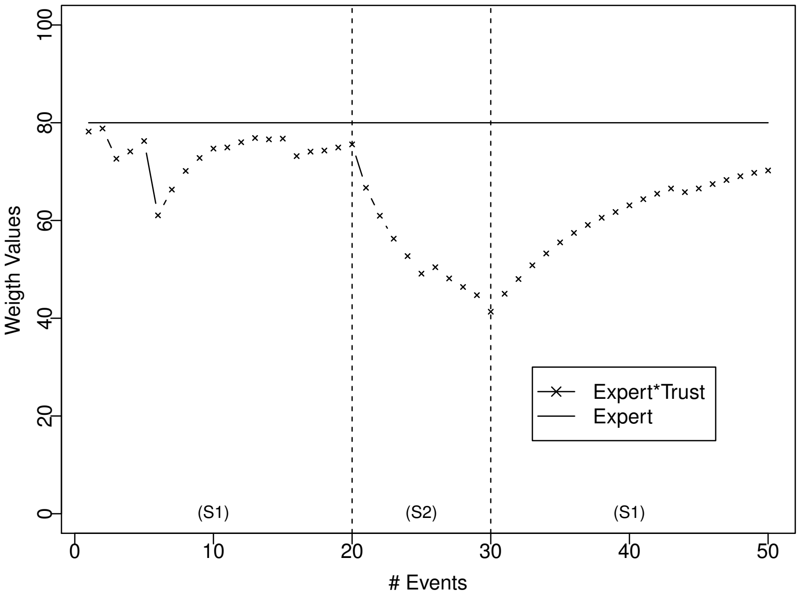


Figure 5.4: Weighted expert trust

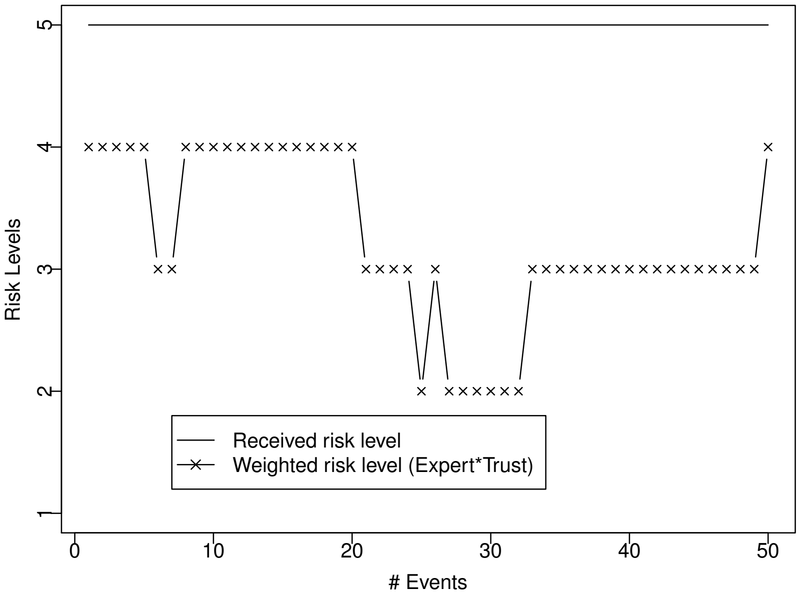


Figure 5.5: Weighted risk levels

5.1.4 Discussion

Within this application scenario, an enhancement to the original CI Security Model was added, allowing to validate the application of the TRS to a different scenario. In order to be able to evaluate the correctness of information received from dependencies, a trust based approach has been introduced. It evaluates the trust in risk alerts received from (inter)dependent CIs or CI services, based on the differences between observations about the dependency and the information received from the dependency.

One of the advantages of integrating a TRS in the CI Security Model, is that trust can be calculated from aggregated risk parameters, not accessing actual infrastructure information. Hence, the Trust and Reputation calculation is simplified and can be applied without modification to any CI that is using the CI Security Model.

The presented approach allows a more accurate evaluation of service risk, since the influence of dependent service risk is dynamically re-evaluated and the impact of incorrect information received from dependencies is reduced based on the trust indicator for the dependency.

One drawback of the proposed trust based dependency weighting approach, is the case on which the trust in one service dependencies and sub-services drops to a considerably low value, the aggregated service risk will be 1. Although, this behaviour is intended, the Operator needs to be aware of why a low risk is shown. The solution found within the CI Security Model is based on the concept of assurance levels originally presented in Aubert et al. (2010b) and is used in the next presented application scenario.

In the next section, a new validation scenario is described on which assurance levels and trust indicators are combined in order to evaluate accuracy of on-line risks in Critical Infrastructures.

5.2 Combining assurance levels and trust indicators

In the previous application scenario, trust indicators were used in order to answer one key question that was not answered in the originally proposed CI Security Model: “How can estimated service risk be validated?”. In order to be able to evaluate the trust that each CI has on a received or locally evaluated risk alert, an entity able to measure the actual service level, has been introduced in the model in order to be compared with each calculated or received risk alert for the evaluation of its correctness.

In the CI Security Model, the accuracy of each service risk relies on the correctness of the subjacent base measurements, as well on their dynamic behaviour during operation. For example, due to some change in environmental conditions, the accuracy of the base measurements can be affected and consequently affect the accuracy of the estimated CI service risk. Assurance levels are part of the CI Security Model as presented in (Aubert et al., 2010b) and (Schaberreiter et al., 2011a).

In the CI Security Model, the correctness of the calculated risk is evaluated by means of assurance indicators, aiming to gather evidence from the underlying systems allowing to categorise each system's assurance into 5 classes (class 1 meaning low confidence in the system, class 5 meaning high confidence). For the next presented scenario, the concept of risk based security assurance and trust-based indicators were combined and adapted, in order to derive assurance indicators that can be used to reason about the accuracy of each calculated CI service risk.

5.2.1 CI service risk assurance indicators

The scenario defined for the application scenario is presented in Figure 5.6. In this Figure two main indicators are represented, namely, the Service Risk and the Service Assurance Level. These two indicators denote the calculated service risk (based on the base measurement information) and our confidence in the correctness of that service risk (Service Assurance Level). In order to estimate the correctness of the CI service risk, the following assurance indicators were defined: the accuracy of each base measurement (*base measurement assurance*); the evaluation of the dynamic behaviour of the base measurements by employing a trust-based approach to capture the dynamically changing accuracies (Risk Alerts Trust) and by evaluating the dynamically changing behaviour of the system and base measurements (Behaviour Trust).

As the CI security modelling and the Trust and Reputation System are already described in this document, respectively in Section 2.3 and Chapter 4, this section will focus on the contributions and improvements that were introduced.

In order to better understand the proposed model, three main indicators are described: the Base Measurement Assurance, the Risk Alerts Trust and the Behaviour Trust, as illustrated in Figure 5.6.

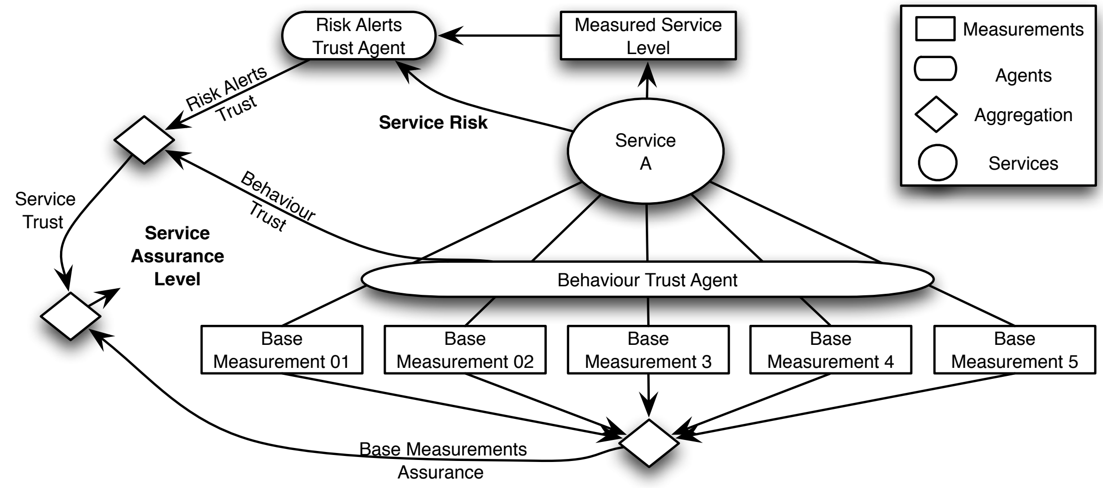


Figure 5.6: System overview (Schaberreiter et al., 2011b)

Base measurement assurance

Based on the original CI Security Model, each service uses base measurements in order to evaluate his aggregated risk level. In this context, the notion of assurance can be defined as the confidence one has in those aggregated risk levels of a service. It is possible to say that a service assurance level represents the accuracy of an aggregated risk level (Aubert et al., 2010b).

Each base measurement has is own associated assurance level. The information to define this assurance level is collected for each base measurement by CI domain experts. This task should be completed during the model definition, in particular, during the off-line risk assessment step of the CI Security Model. The expert should gather evidence about the correctness of each base measurement. For instance, the expert can have an opinion regarding the capabilities of one particular equipment (that provides base measurements) he understands. Also, it is common to classify the measuring equipment reflecting its accuracy. For example, a thermometer usually has its accuracy defined by the manufacturer. In this particular illustration, the manufacturer can say that the error margin is 1°C. This information, when available, can also be used as evidence and help define the accuracy of the base measurements. The assurance level for each base measurement becomes the combination of the expert subjective opinion about that base measurement and the evidences collected for the base measurement (Schaberreiter et al., 2011b).

The assurance level is represented by an integer number in the range [1..5]. The reasons that lead to the adoption of this representation are described by Schaberreiter et al. as a trade-off between accuracy and interpretability by an Operator in a stress situation (Schaberreiter et al., 2011b). Also, the assignment of assurance levels to base measurements by a domain expert is manageable in this way. Schaberreiter et al. also claims that the expert needs to have a sufficient amount of different choices, but at the same time the choice needs to be limited in order to have a meaningful comparison between the values (Schaberreiter et al., 2011b).

Once each base measurement has is assurance level defined, those assurance levels are combined using an average weighted sum, in order to represent the confidence in the accuracy of the corresponding service level. An aggregation process is also used to obtain the service risk level, where each weight represents the relevance a base measurement has to the service risk. In this context, the same weights can be used to aggregate service risk and service assurance levels.

In the original CI Security Model, service assurance levels are not supposed to change often, as the expert is the only one that can alter them. An expert revaluation on the base measurements could happen, for instance after an equipment change or after detecting erroneous assumptions made during the off-line risk assessment phase.

As an example to calculate service assurance level, let us suppose the existence of a service measured using four base measurements (μ). If the expert confidence is high in one base measurement, medium in one base measurement and low in two base measurements ($AL_{\mu} = \{5, 3, 1, 1\}$) and the importance that the base measurements have to the service are respectively ($W_{\mu} = \{0.9, 0.3, 0.1, 0.1\}$) from where is noted that the service in which the expert has more confidence is the one with major importance to the service.

Using the CI Security Model methodology, the aggregated service assurance level (AL_S) is calculated as described in Equation 5.5.

$$AL_S = \left\lfloor \frac{\sum_{i=1}^{n} (AL_{\mu_i} * W_{\mu_i})}{\sum_{i=1}^{n} W_{\mu_i}} \right\rfloor = \left\lfloor \frac{5 * 0.9 + 3 * 0.3 + 1 * 0.1 + 1 * 0.1}{0.9 + 0.3 + 0.1 + 0.1} \right\rfloor = 4 \ . \quad (5.5)$$

It can be seen that the aggregated service assurance level is relatively high, due to the fact that the base measurement with the highest confidence is assumed to be the one most relevant to the service.

Risk Alerts Trust

As already presented, Risk Alerts Trust is seen as the trust in the correctness of the calculated service risk. The idea behind the concept of Risk Alerts Trust is to compare the service risk Rl_t to the actually measured service level (Ml_t) as a measure of the Quality-of-Service. For example, if a power generation service has a high risk of availability degradation and the measured service level does not indicate that degradation, the trust in the accuracy of that service risk level should be lowered. The measured service level must be collected using measurement equipment that must be independent from the service itself. After collecting the accuracy value for each calculated service risk, the Trust and Reputation System is able to determine the trust as detailed in Section 4.2.

Behaviour Trust

As stated, in the original CI Security Model, each base measurement assurance level is basically static if not manually changed by the expert. In the Trust and Reputation System, Behaviour Trust refers to the trust in the correct behaviour of an entity (for example a service or a single component). As presented in Figure 5.6, the behaviour of the base measurements is examined in order to incorporate that behaviour and use it to reason about the assurance one may have in the service risk calculated based on the information those base measurements provided.

The main goal is to understand and quantify the behaviour of each monitored entity considering what should be its normal behaviour. When a deviation from normal behaviour is detected, an event is triggered in order to incorporate this event in the Behaviour Trust indicator. The events used to evaluate trust in service behaviour can include all the monitored interactions among services (internal or external). For instance, the events can be Intrusion Detection System's alerts, failed connection attempts, attempts to read/write information without permission or the fact that some entity does not update risk information for a long period of time.

In the presented scenario, the main sources of information used to evaluate the Behaviour Trust are the base measurement entities. As it can be simple to describe the normal behaviour of those entities (usually simple measurement equipment) it is possible to generate a security event when an abnormal behaviour is detected. For instance, although the normal temperature of an equipment can range from -10°C to 70°C, in some cases, it can be considered abnormal if the sensor reads 20°C and one second later reads 70°C and continues in this cycle. This fact demonstrates an abnormal behaviour of the sensor. Another and rather common case of abnormal behaviour is the when the sensor does not reports values. If a normal situation as defined by an expert, is one in which the temperature sensor should inform the temperature at least every 60 seconds, it is possible to say that the behaviour is abnormal if that has not been accomplished.

In order to evaluate and measure behaviour events, it is mandatory to quantify the defined abnormal events. Essentially, it is important to know and quantify "how much" the behaviour differs from the expected.

As presented in Section 4.2, behaviour information is normalised based on a security model that identifies relevant behaviour patterns. Tables representing possible observed values and correspondent Behaviour Trust event values compose this security model. This model acts as an adaptor between multiple heterogeneous sources and the TRS allowing it to evaluate the Behaviour Trust indicators, as all entities are quantified and can be used in a common evaluation.

For this application scenario, the security model includes Table 5.2. This Table, defines the trust indicator level associated to the time on which each information is received from the sensor. For instance, as represented in Table 5.2, if a reading from the sensor is made 50 seconds after the last reading, a trust indicator level of 2 is used for Behaviour Trust evaluation. The range [1..5] used in Table 5.2 to represent the trust indicator level as been chosen in order to allow a better integration with the CI Security Model.

Table 5.2: Normalisation Table Example

Received information from sensor X	
Trust Indicator Level	Seconds since last value
1	<= 30
2	> 30 and < 60
3	>= 60 and < 120
4	>= 120 and < 180
5	>= 180

As presented in Chapter 4 one may expect to receive behaviour alerts only when misbehaviour is detected, leading to a situation where almost only "bad behaviour" events are received and used in the evaluation. If not treated, this situation would generate low Behaviour Trust over time. In order to evaluate an accurate indicator, the time factor and the management of inactivity periods were added. Time is divided into a set of time slots and if there is inactivity in one slot, it means that the entity behaviour indicators should have the maximum value for that period (normal expected behaviour). If information is received during one slot, the value for that slot becomes the average of all values received during that slot. Besides this, it is also possible to include "normal behaviour" in the tables representing possible observed values and correspondent Behaviour Trust event values. For example, in Table 5.2, the trust indicator level is 1 (best value = normal behaviour) when no failures are detected.

Using the methodology described in Section 4.2, for the time slot s, the trust in entity E ($T'_{(B,s)}$) is calculated using Equation 5.6, where D is the ageing factor, $T_{(E)}$ is the indicator evaluated for the slot $(s-1)$ and $Event_{(Slot\ s)}$ is the event value of the slot s.

$$T'_{(E,s)} = \frac{(D * (s-1) * T_{(E)}) + Event_{(Slot\ s)})}{D * (s-1) + 1} \quad . \tag{5.6}$$

5.2.2 Validation Results

The use of a combination of assurance levels and trust indicators, has enable the evaluation of a more precise indicator (service assurance level), allowing inferring on the service risk accuracy. The uses of these indicators were validated by the use of simulation and the outcome is promising as the simulation results are in-line with the main goals. Also it became clear that the trust model is flexible and adaptable to multiple scenarios where it is necessary to reason about the reliability of some indicators.

This section presents an example in order to demonstrate the proposed approach and to help understand the influence of trust indicators in the service assurance level and the contribution of the TRS to the CI risk evaluation. The simulations were achieved using R (R Development Core Team, 2009) and also using the developed simulators presented in Section 4.4.

The scenario used for this example is represented in Figure 5.6. A simple scenario is presented as the CI Security Model allows the simplification of a CI model by representing each of the services that compose the CI. This simple scenario is composed of one single service collecting information from five base measurements (derived from sensors).

For simplicity reasons it is assumed that the service present in the scenario does not depend on other services. However, as already understood, if a dependet service exists, the risk information received from the dependent service would be used in the service risk evaluation and also in the Risk Alerts Trust evaluation, as demonstrated by the previous validation scenario. For the simulation, the following assumptions are made:

- The base measurements are retrieved and evaluated once per minute;
- The simulation total time is 50 minutes;
- In order to evaluate service risk and base measurement assurance, the contribution that each base measurement has to the service has been defined as follows: $S1 - 10\%$; $S2 - 10\%$; $S3 - 30\%$; $S4 - 20\%$; $S5 - 10\%$;
- The service risk R_S is aggregated using the previously defined average weighted sum method[1];
- The measured service level M_S is aggregated using a similar setup of 5 independent sensors;
- The confidence in the correctness of all the base measurements is high and results in a base measurement assurance level of 5;
- The service trust is derived from the Risk Alerts Trust and from the Behaviour Trust using the following weights: 0.5 for the Risk Alerts Trust and 0.5 for the Behaviour Trust;
- The service assurance level is derived from the service trust and the base measurement assurance using the following weights: 0.4 for the service assurance level and 0.6 for the service trust;
- All indicators are defined using a scale of 1 to 5. In the case of the base measurements, for the trust indicators and for the assurance levels, 5 represent the best situation and 1 represents the worst. For the service risk level, 5 represents the highest risk and 1 represents the lowest risk;
- When a base measurement sensor does not update its status, the last received value for that sensor is used to evaluate the service risk.

[1] For simplicity reasons, only one risk indicator is taken into account for simulation. Whenever R_S is mentioned, it represents either C,I or A risk indicator.

In order to implement a credible scenario for the simulation, the following situations were generate: for the first 20 minutes all sensors that support base measurements are reporting the value 5 (maximum value) leading to a low risk level $R_S = 1$. In the same period, the independent sensors used to aggregate the measured service level are also equal to 5 producing a service level $M_S = 5$. Also, during the first 20 minutes, it was possible to gather information from the base measurements once a minute. This means that the Behaviour Trust in base measurements has the maximum value. As the Risk Alerts Trust and the Behaviour Trust have both the maximum value, the composed indicator service trust will also have the maximum value.

After the first 20 minutes, and for a duration of 10 minutes, sensors 1 and 3 of R_S (sensors that support the base measurements) become unreliable but continue to report a value as presented in Figure 5.7. During this period those sensors always report value 5, while the information arriving from sensors 1 and 3 of M_S (measured service level) are generated using the following criteria: The difference between the sensor outputs of R_S and M_S is 1,2,3,4 respectively in 0%, 5%, 5%, 90% of the cases. After t = 30 minutes, the difference between the sensor outputs returns to 0. In Figure 5.8 the Risk Alerts Trust indicator displays this comportment. The indicator drops when the values become unreliable and gradually starts to grow when the situation reverts to normal.

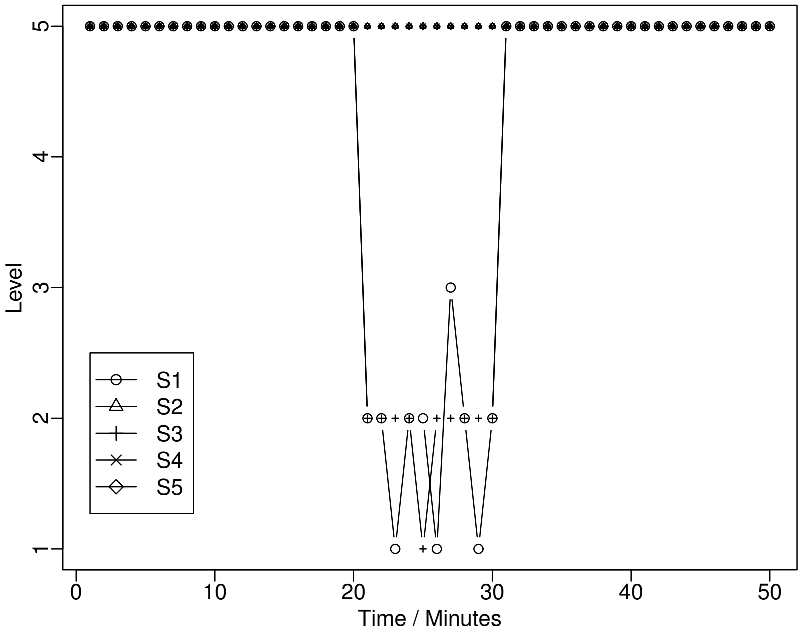


Figure 5.7: Sensor Output for measured service level

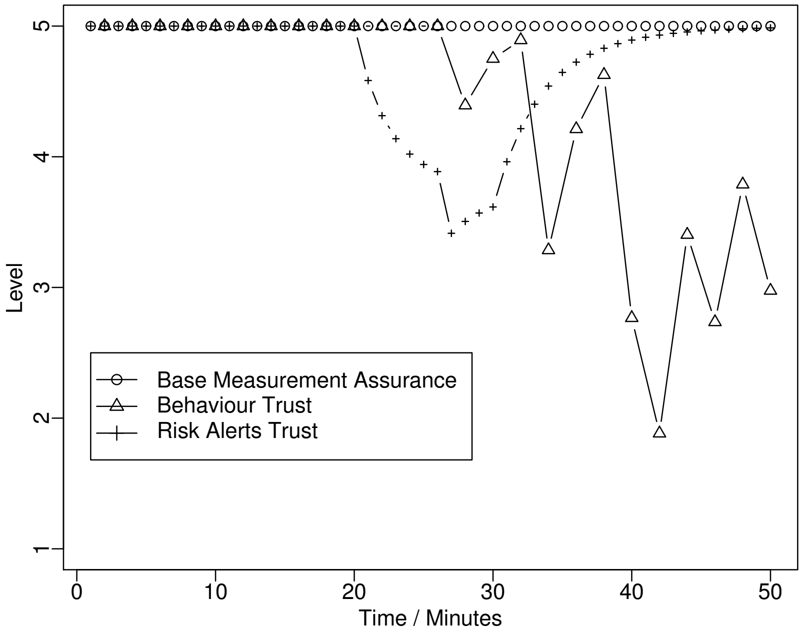


Figure 5.8: Assurance Indicators

As explained, the Behaviour Trust indicator can be evaluated based on information gathered from multiple sources available on the CI that should represent the behaviour of the system. In this example, the behaviour of the sensors is simulated regarding their ability to send periodic information. For this simulation, the normalised values presented in Table 5.2 are used in order to represent a situation where the sensors are not able to send well-timed information as expected.

In order to show the impact that Behaviour Trust has in adapting the service assurance level according to what is observed in the system, it is implemented in the simulation the fact that after 25 minutes of the simulation time, two of the base measurement sensors stop sending periodic information.

As described, for the service risk aggregation, the last information received from the sensors will be used. By looking at Figure 5.7 it is not possible to detect that information is not being received as it should. It becomes clear that only the observation of the system behaviour through the Behaviour Trust reveals that the comportment of the system is not as it should be. This fact can be observed in Figure 5.8 where it is shown that the Behaviour Trust indicator changes rapidly, as a result of the inconstant updates by the sensors.

During all the simulation, the calculated service risk had always the value 1, meaning that there is no risk in the service. This fact is explained by the fact that the base measurement sensors always gives the value 5 (maximum value). By observing Figure 5.9 it can be seen that, although the service risk always indicates no risk (value 1), the service assurance level indicates that our confidence in the risk estimation changed based on the dynamic behaviour observed by the Risk Alerts Trust indicator and the Behaviour Trust indicator. The static base measurement assurance indicator presented in the original CI Security Model would not have captured this behaviour.

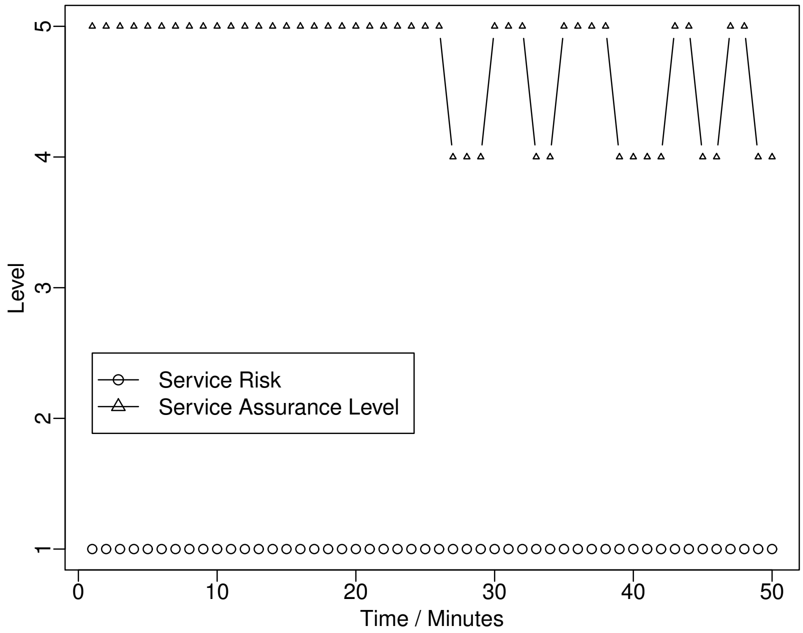


Figure 5.9: Service Risk / Service Assurance Level

5.2.3 Discussion

With this application scenario it was possible to introduce and discuss indicators that can be used to evaluate the correctness of aggregated CI service risk (Schaberreiter et al., 2011b).

The scenario presented in this section uses the CI Security Model, that represents risk on the level of provided CI services and the risk of the services they depend on. Three assurance indicators were identified and presented – the service assurance level, representing the confidence in the correctness of the measurements that are used to evaluate the CI service risk; the Risk Alerts Trust indicator, evaluating the inconsistency that may happen between calculated or received service risk and actual observed/measured service level – the Behaviour Trust indicator able to evaluate the dynamic behaviour of the base measurements.

This application scenario allows the demonstration of the applicability of the Trust and Reputation System in different scenarios where its contribution can clearly improve the risk estimate mechanisms within a Critical Infrastructure (Schaberreiter et al., 2011b).

The next section describes the work and the results achieved while exploring a new application scenario. During most of this work, it always felt necessary to test the main achievements on a realistic CI scenario allowing to build a CI Security Model as well as access to dynamic data of CI behaviour (in normal operation as well as during security incidents). One drawback was the lack of CIs willing to share data in order to test the proposals. Fortunately, with the support of Sebastien Varrette from the University of Luxembourg, it was possible to obtain data from the Grid'5000 project (Grid5000, 2013) and implement a different scenario (supported by real data). The next section briefly describes the Grid'5000 project, the implemented scenario and highlights the obtained results.

5.3 Trust based interdependency weighting - The Grig'5000 case study

Previous application scenarios already introduced CI security modelling to enable on-line risk monitoring in, for instance, CIs that depend on each other by exchanging risk alerts expressed in terms of a breach of Confidentiality, a breach of Integrity and degrading Availability (C,I,A). While generally providing a solid basis for risk monitoring, there was no way of evaluating if a risk alert received from an external CI is accurate.

In this application scenario the applicability of the proposed solution to this problem is demonstrated by adding a trust based component to the CI Security Model in order to improve its accuracy and resilience to inconsistent or inaccurate risk alerts provided by (inter)dependent CIs. Hence allowing to evaluate the correctness of the received alerts. Although the approach has been already presented and validated using simulations, the need of further testing, that should be performed in real scenario with the use of real data, was identified.

In this section the applicability of the proposed approach is validated by simulating a use case scenario taking advantage of information from a real-world infrastructure, namely the *Grid'5000* (Bolze et al., 2006; Grid5000, 2013) platform.

The Grid'5000 project supports an academic computing grid with clusters distributed at numerous locations, such as France and Luxembourg, with the objective to help performing large-scale experiments that involve considerable amount of processing power, storage or both.

This scenario focuses on evaluating a dependency between the computing grid and the telecommunication infrastructure used to interconnect each site of the infrastructure. The trust is evaluated based on a dataset of measurements gathered by the available monitoring tools.

5.3.1 Trust and the CI Security Model

In this example the focus is on a dependency between the computing grid and the telecommunication infrastructure used to interconnect each sites of the infrastructure. In this case, services belong to different Critical Infrastructures as presented in Figure 5.10.

This section describes how the Risk Alerts Trust and Behaviour Trust are integrated with the CI Security Model allowing the evaluation of the correctness of the CI service risk as received from dependencies. The methodology used in this scenario is illustrated in Figure 5.10.

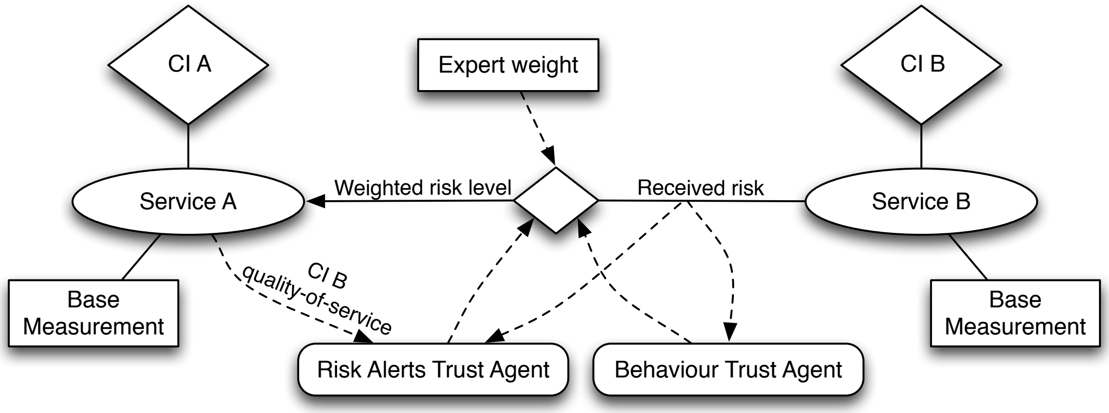


Figure 5.10: Trust based dependency weighting (Caldeira et al., 2013).

Risk Alerts Trust

From the CI Security Model, the received CI service risk is intended to be abstract information and it is supposed that the receiving CI usually neither knows how the risk received from a dependent service was derived, nor if the risk value is correct. The introduction of mechanisms allowing the CI Operator to reason on the received information can help increase the confidence among CI providers.

In this context, the use of Risk Alerts Trust is a way of reasoning about the received risk values by comparing it to the experienced Quality-of-Service. For example, in Figure 5.10, *Service A* depends on the services provided by *Service B*. If *Service B* continuously sends a low risk value, however *Service A* is suffering a degradation in the agreed service level, it becomes clear that the trust that CI A has on the accuracy of the risk level received from *Service B* in CI B, should be reduced. As described in Chapter 4, in order to evaluate trust aspects related to the received risks, it is required to define an accuracy value for each received risk alert. Also, in this scenario, a Risk Alerts Event is defined as one of the following situations: An event starts when one or both risk indicators (received risk from *Service B* and *Service B* Quality-of-Service measured by CI A) are different from one (no risk). The event ends when both indicators assume again the value one.

As represented in Figure 5.10, in order to identify events, the Risk Alerts Trust Agent is constantly observing the received risks (Rl_t) and the current quality of the service (Ml_t) provided by CI B. According to the information used in the CI Security Model, both Rl_t and Ml_t are expressed using the [1..5] range. From the Trust and Reputation System described in Chapter 4, each event $A(Event_n)$ is defined as the average of all comparisons made during the event (value T), between the measured risk level and the Quality-of-Service level (see Equation 5.1).

$$A(Event_n) = 100 - \left(\frac{\sum_{t=1}^{T} f(Ml_t, Rl_t)}{T} * 100 \right) \quad , \tag{5.7}$$

where $f(Ml_t, Rl_t) = \mid \frac{Ml_t - Rl_t}{4} \mid^k, k \in R^+$. The factor k allows to penalise the larger differences or the small differences. In this approach, the duration of an event is not reflected, as the main objective is to discover the accuracy of each received risk alert.

According to the model described in Chapter 4, the trust that CI_A has in the received risk alerts related to *Service X* provided by CI_B is represented by $T_{(A,B,X)}$

and is calculated by the average of the accuracy of each past events between those two CIs for that particular service (see Equation 4.5). In order to better interpret the $T_{(A,B,X)}$, the time at which each indicator was evaluated is also included. By including time, the indicator $T_{(A,B,X,t)}$, represents the trust that CI_A has in *Service X* provided by CI_B at time t.

Behaviour Trust

As explained, Behaviour Trust allows to evaluate and understand an indicator, reflecting the trust one can have in the comportment associated with the exchange of risk information with a dependency. For example, it is expected that *Service B* in Figure 5.10 must update and send the service risk in a fixed time interval. If this expected behaviour fails for a determined period of time it is possible to assume that something is wrong with the service or with a component involved in the communication. In this case, during that unusual behaviour, the trust one may have in the received service risk should decrease, at least due to the fact that the last received value is outdated.

As explained in previous examples, the information sources that are suitable to estimate the trust regarding the behaviour of an external CI or dependent service, are in fact the interactions among CI/services in terms of security, either internal or external. For instance, the events can be IDS's alerts, failed connection attempts, attempts to read/write information without permission or the lack of information when expected, etc.

By applying the methodology presented in Chapter 4 for the Behaviour Trust evaluation, it is possible to evaluate an indicator encompassing all types of defined security/behaviour events. Using a weight factor for each entity, the behaviour reputation for each CI (or group of services) can be computed, considering also the Operator information. This indicator, $TBehaviour'_{(B,t)}$, represents the reputation of the behaviour of CI_B including all its services at time t and is computed using Equations 4.7, 4.8, 4.9 and 4.10 described in detail in Chapter 4.

Improved Trust Based Dependency Weighting

As already stated, one identified weakness, of the original CI Security Model (described in Section 2.3), is the dependency weights used to integrate received service risks into the risk evaluation are manually assigned by experts and consequently prone to human errors or inaccuracies. Also, as already stated, those weights are not supposed to change dynamically during system operation. In this scenario, the author of this thesis and Thomas Schaberreiter, author of the CI Security Model, improved the methodology presented in Section 5.1 for the trust based dependency weighting, as illustrated in Figure 5.10, by using all available indicators. Namely, the Risk Alerts Trust, the Behaviour Trust and the initial expert, all gave weight to evaluate the influence a service risk received from a dependency has on the dependent service risk estimation.

For example, in the CI A, represented in Figure 5.10, the use of the trust based dependency weighting, will allow to obtain a more precise service risk estimate for *Service A* since the weight that *Service B* has for *Service A* risk evaluation can be lowered, when it is considered that the support information or behaviour cannot be trusted.

The global trust indicator used by *Service A* can be acquired by the aggregating of both defined trust indicators (Risk Alerts Trust and Behaviour Trust), this indicator is described in Chapter 4 and obtaining from the use of Equation 5.8, where $T_{RiskAlert}(A, B, X, t)$ represents the trust that CI_A has in $ServiceX$ of CI_B at time t, $T_{Behaviour}(E, B, t)$ expresses the trust that CI_A has on entity E from CIB on the same time t and β denotes the weight that each indicator has on the service trust indicator.

$$T(A, B, X, t) = \beta * T_{RiskAlert}(A, B, X, t) + (1 - \beta) * T_{Behaviour}(E, B, t), (0 < \beta < 1) \quad . \tag{5.8}$$

In a similar manner as described in Section 5.1.2, in order to allow association of the indicator $T(A, B, X, t)$ with the dependency weight for the services, the meaning of the weights has been changed in comparison with the original CI Security Model. In the original approach, an expert weight represents the influence of a dependency on a service. In the actual context, with the use of the TRS, the expert weight describes the maximum influence that a dependent service can have on the service risk evaluation. Those weights are now continuously adjusted according to $T(A, B, X, t)$ as exemplified by Equation 5.9. In this Equation, $\omega(A, B, X, t)$ is the trust based dependency weight used by CI_A when including information from *Service X* of CI_B on its own service risk evaluation. $\omega_E(A, B, X)$ is the originally defined expert weight and $T(A, B, X, t)$ corresponds to the service trust indicator for the dependency X. ω is computed in the range]0..100] and 0 is not a possible value as in the case that the trust on one dependency is 0, one should act accordingly and remove this dependency from the service risk evaluation. This problem is inherent to the CI Security Model and has been identified in (Aubert et al., 2010a).

$$\omega(A, B, X, t) = \frac{\omega_E(A, B, X) * T(A, B, X, t)}{100} \quad , \omega_E(A, B, X), T(A, B, X, t) \in [0..100] \ . \tag{5.9}$$

5.3.2 Case study: the Grid'5000 project

In order to confirm the applicability of the presented approach, the validation is supported by a case study based on a realistic scenario. More specifically, the validation work described in the following section was based on the CI Security Model and data collected from the Grid'5000 grid platform (Bolze et al., 2006; Grid5000, 2013).

Grid'5000 aims to provide the users with a fully customisable testbed able to perform advanced experiments in all areas of computer science related to parallel, large-scale or distributed computing and networking. The project web site defines Grid'5000 as "a research effort developing a large-scale nation wide infrastructure for large-scale parallel and distributed computing research" (Grid5000, 2013).

The first prototype of what Grid'5000 is now has began developing in France in 2003 and has been open for users since 2005. Actually, the infrastructure covers a set of eleven geographical sites composing Grid'5000 – ten in France (Bordeaux, Grenoble, Lille, Lyon, Paris/Orsay, Nancy, Reims, Rennes, Sophia, Toulouse) and one in Luxembourg. Furthermore, the infrastructure has been expanded beyond France and Luxembourg, with the deployment of extra international connections to Brazil, Japan and the Netherlands. Those international connections are provided via the site of Grenoble (Grid5000, 2013).

According to the project web site (Grid5000, 2013), the support backbone for the Grid'5000 sites in France is based on 10Gbit/s dark fibres providing also IP connectivity to all overseas sites. The backbone is also interconnected with the GEANT high bandwidth pan-European research and education network (Géant, 2013), the SFINX global Internet exchange point (SFinx, 2013), the DAS-3 - The Next Generation Grid Infrastructure in The Netherlands and with the NAREGI Grid in Japan (NAREGI, 2013). According to (Grid5000, 2013), the introduction of a dark fibre infrastructure now allows to allocate dedicated 10Gbit/s links for specific research projects. The overall IP network of the Grid'5000 is presented in Figure 5.11 available on the project Web Site.

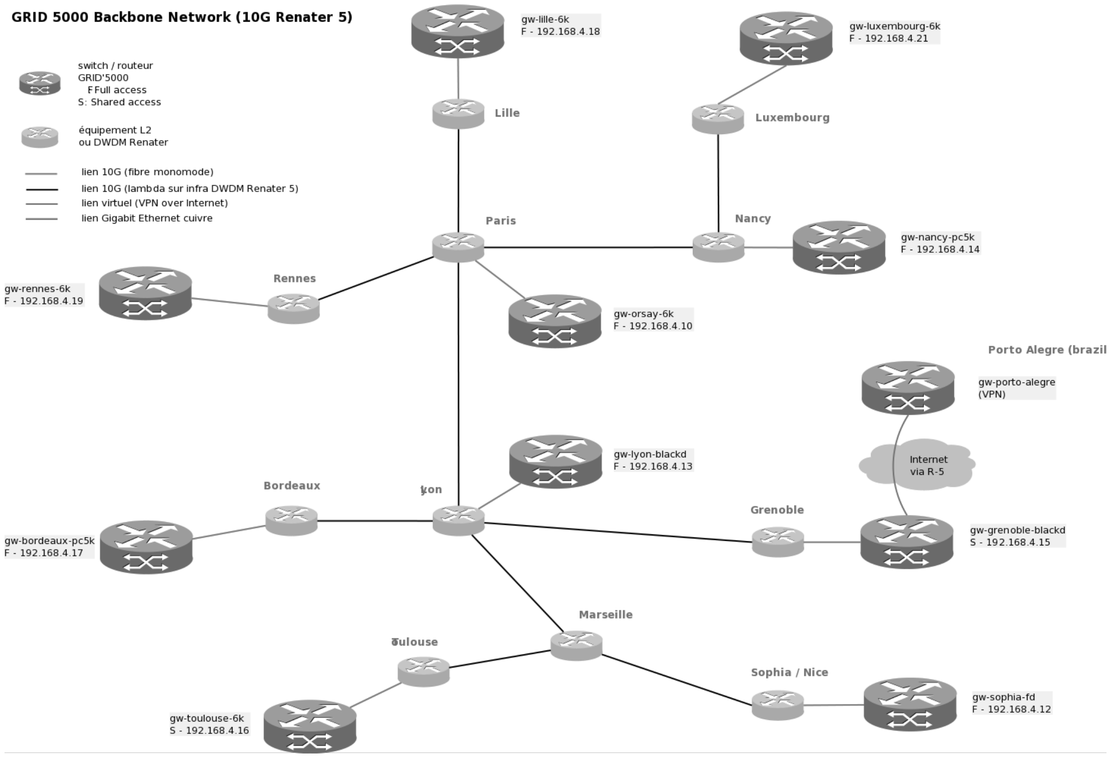


Figure 5.11: Grid5000 IP Network (Grid5000, 2013).

Although one can argue that this type of infrastructure is not critical due, for instance, to the fact that no human lives depend on Grid'5000 services (at least in the short and medium term), as a research facility, Grid'5000 fits the critical sectors defined by the European Commission presented in Table 2.1. Grid'5000 is already a large infrastructure shared among multiple users and supported by internal services (sites) and by external services (telecommunication services). Apart from the definitions it is the author's opinion that Grid'5000 can be considered a Critical Infrastructure involving several crucial security components, for instance, as described by Sébastien Varrette (Caldeira et al., 2013):

- the *Puppet infrastructure* (Puppet, 2013), responsible for the configuration of all grid services within Grid'5000;

- the *Chef and Kadeploy infrastructure*, which pilots the deployment of the computing nodes of the platform;

- *OAR* (Capit et al., 2005), the resource manager of Grid'5000;

- the *network backbone*, operated by independent providers, namely Renater (Renater, 2013) in France and Restena (Restena, 2013) in Luxembourg.

As mentioned above, these components, considered critical for the Grid operation, are dispersed among all the sites (mainly in France and in Luxembourg - international connections to Brazil, Japan and the Netherlands are operated via the site of Grenoble) that compose the Grid'5000 infrastructures.

According to Sébastien Varrette (Caldeira et al., 2013), the Grid'5000 platform is managed and monitored by a technical committee formed by two engineering work groups, specifically: 1) the support staff: whose main functions are to coordinate the platform administration; the development of specific administration tools and to provide user support. 2) the development team: mainly responsible for the design and development of the fundamental tools used during the platform operation. These groups also develop and maintain the Grid'5000 API that, as described in (Grid5000, 2013), is divided into six main areas, namely: the *Metrology API* – provides multiple metrics of the existent nodes, for example, memory, CPU usage, byte in, bytes out, among others; the *Monitoring API* – provides the status of the nodes; *Jobs API* – allow user to submit jobs on the grid sites; *Deployments API* – allows do deploy specific environment configurations on a grid node; *Users API* – offers mechanisms for user account management; *Reference API* – provides general information about the grid such as, list of sites, nodes, installed environments, etc. An example from an application developed on top of the Grid'5000 API is shown in Figure 5.12. This Figure is a snapshot of the Grid'5000 dashboard, available for the Grid users from where they can visualise, in real time, the status of all sites composing the Grid.

Among the multiple critical components present in the described infrastructure, this case study focuses on the existent dependencies among services. In particular, focuses on the dependency between the Grid'5000 sites and the network (telecommunications) infrastructure. This example provides a scenario where it is possible to highlight a dependency between two independently operated and managed CIs.

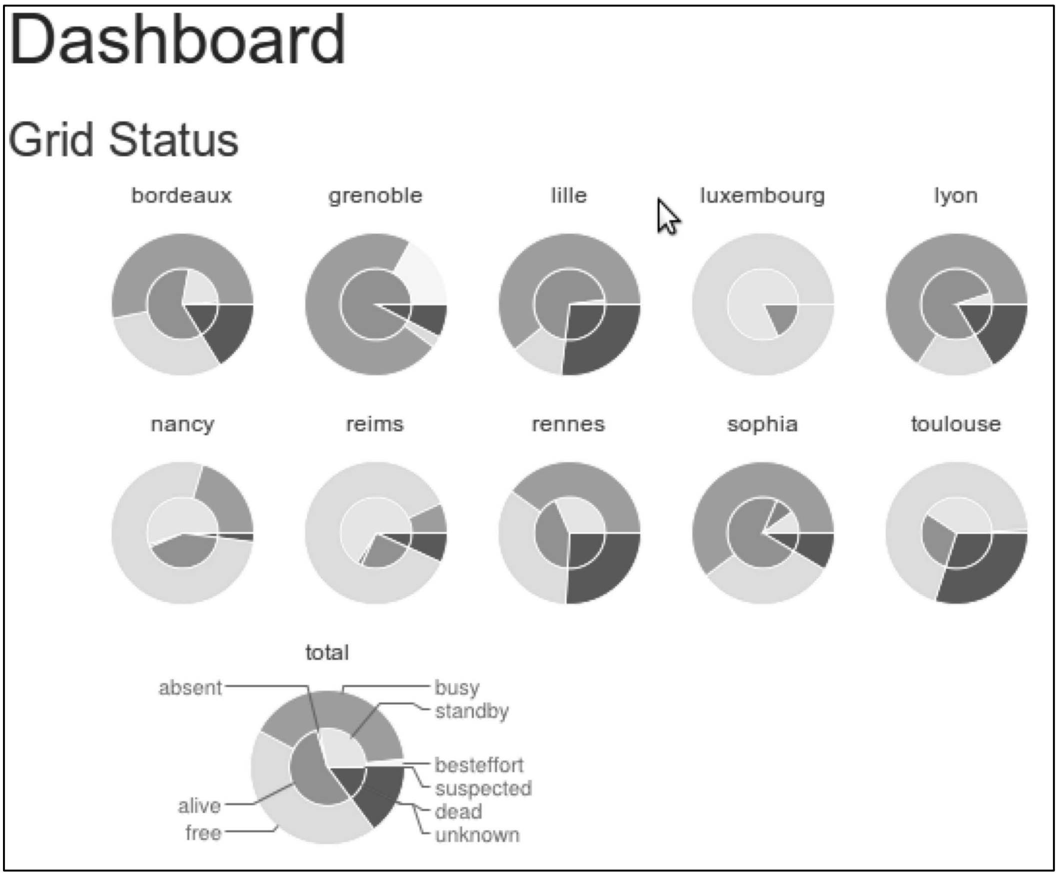


Figure 5.12: Grid5000 Dashboard (from (Grid5000, 2013)).

Application scenario

The general representation of the case study is presented in Figure 5.13. The network infrastructure provided by Renater is visible in the Figure, that comprises 13 network segments, each one indicating a connection between two GRID'5000 sites. As indicated in Figure 5.11, most of these segments are implemented as dedicated 10 Gbits/s Ethernet lines, with an exception to the international connections from Grenoble to Porto Alegre (Brazil), Naregi (Japan) and DAS3 (Netherlands) that are served using VPN (virtual private network) connections.

In order to apply the CI Security Model to the presented scenario it is necessary to locate the available information for each service (base measurements). This information is then used to evaluate service risk. In this case, the state of each network segment can be described by available base measurement information and used, in this particular example, to evaluate the risk of degrading Availability.

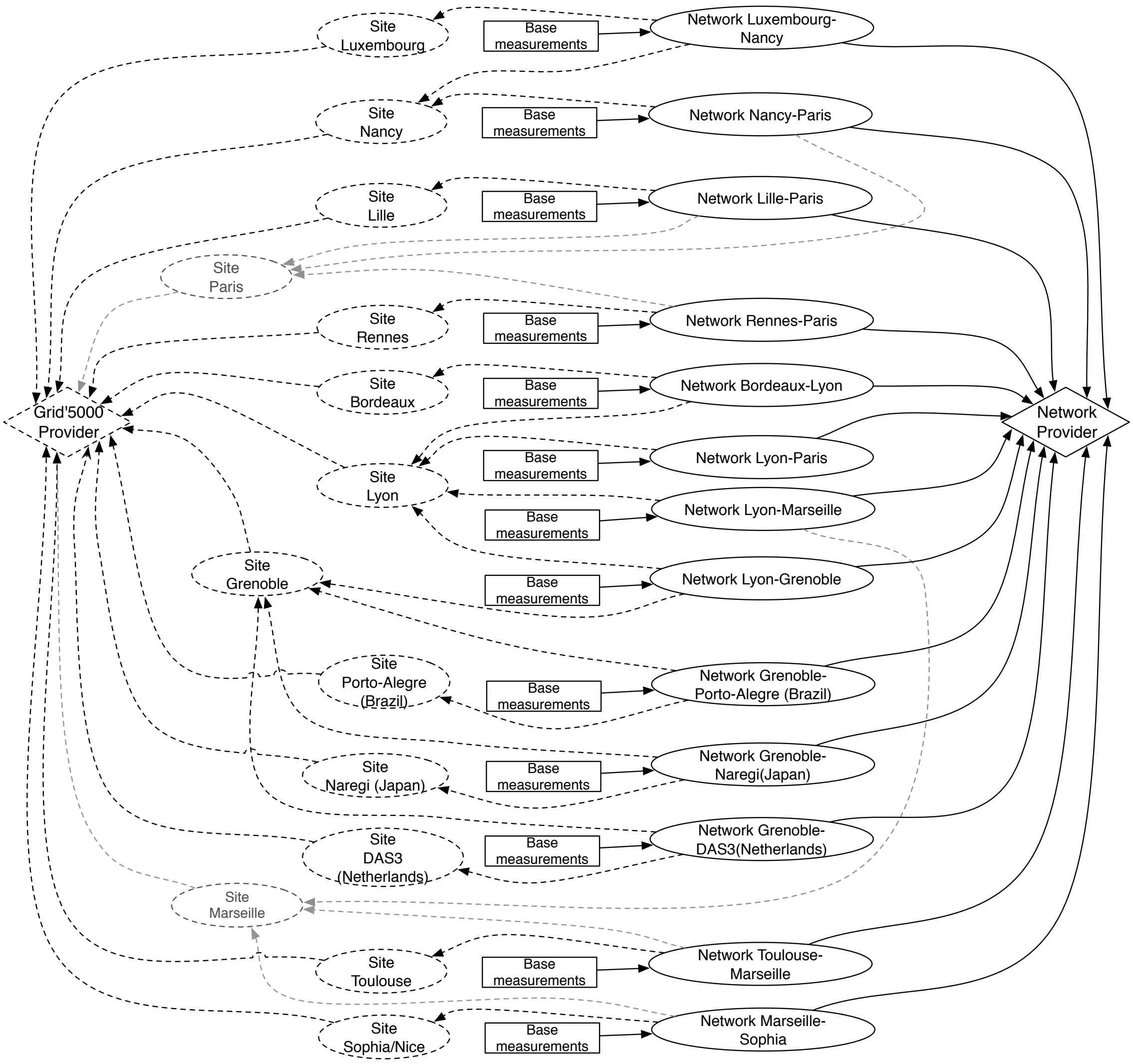


Figure 5.13: Case study overview: illustration of the dependency between Grid'5000 and the network infrastructure (Caldeira et al., 2013).

It is clear that each of the existing sites in the Grid infrastructure depend on the services provided by the network provider, which allow communications to be established and maintained across all sites. In order to estimate the risk for each site, the site must receive risk information regarding the network segment on which it depends and include this information in that risk estimation. Note that all sites can communicate to each other via the shown network connections. For instance, the site Luxembourg can communicate with the site Lyon via the network segments Luxembourg-Nancy, Nancy-Paris and Paris-Lyon as represented in Figure 5.11. In this case, the network connexion (Luxembourg-Lyon) should be characterised by the three network segments involved.

In order to reduce the complexity while presenting the case study and also due to constrains in the data that was available from the infrastructure, this scenario is focused on the site Luxembourg that depends on the network segment Luxembourg-Nancy as illustrated in Figure 5.14.

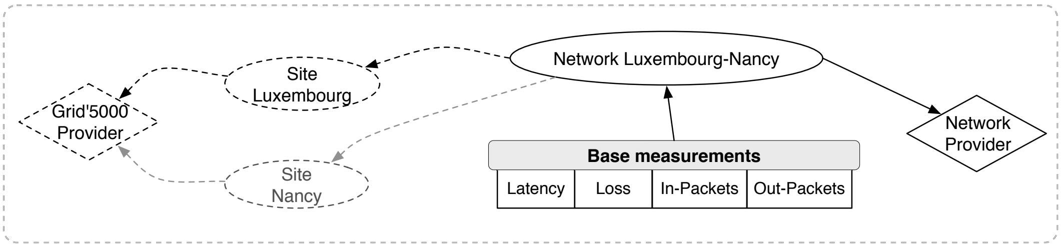


Figure 5.14: Detailed view of the Luxembourg-Nancy network segment (Caldeira et al., 2013).

Among the available raw data gathered from the measurement equipment, four base measurements were identified in order to allow the representation of each segment and in particular the segment Luxembourg-Nancy. The available base measurements are described in Figure 5.14 and include network performance and traffic measurements. In particular the observed base measurements are:

1. **Latency**: Measures how much time a packet requires to travel from source to destination and then returned to its sender (measured in seconds).

2. **Loss**: The number of packets that are lost while measuring the latency.

3. **In-Packets**: The number of packets arriving into the segment (in packets/second).

4. **Out-Packets**: The number of packets leaving the segment (in packets/second).

The available base measurements can be gathered and used by the CI Security Model in order to compute *Availability risk*. To accomplish risk evaluation, a set of weights, expressing the relevance each base measurement has to the service, must be defined. After specifying the weights, the average weighted method described for the CI Security Model is applied. Although information was obtained from the presented four indicators, the quality of the data collected for each indicator differs making it difficult to use them in a simulation. Also the integration of multiple base measurements would not help explaining and interpreting the results. After analysing all data available, it was decided to use only the *latency* base measurement to characterise the service risk of the network segment Luxembourg-Nancy.

Grid'5000 uses the Smokeping (Smokeping, 2012) tool in order to collect the data from the infrastructure and RDDTool (RDDTool, 2012) for data storage in a database. The RDDTool stores data using a round-robin database, due to this fact, the recent data is kept stored in a shorter time interval than older data, allowing the system storage footprint to remain constant over time. The provided database contains latency measurements for a time period of about one year (from June 2011 until June 2012).

Figure 5.15 shows a graph representing the available data for the latency base measurement gathered from the network segment Luxembourg-Nancy. The graph has been planned in a way to allow the observation of the latency over time (x-axes). Thus it becomes clear that, due to the fact that data is stored using a round-robin database, the first periods of time have fewer information and occupy more space (more spread over time) than the newer records that have more information for an equal period of time.

In order to allow a proper visualisation of the simulation results, to reduce the differences existing in the amount of data over time and also to show the data evenly distributed, it was decided to use the provided information as (#Events) and to remove the time/date information. In a real application of the proposed approach, as the evaluation is to be made in real-time, it is expected to receive each Event (latency value) on a regular time interval. Therefore, using the available information as Events, allows simulating a real-time environment where the reporting interval is not supposed to change. One problem that arises from the use of real latency data gathered from a system as the Grid'5000, is that it was not possible to identify

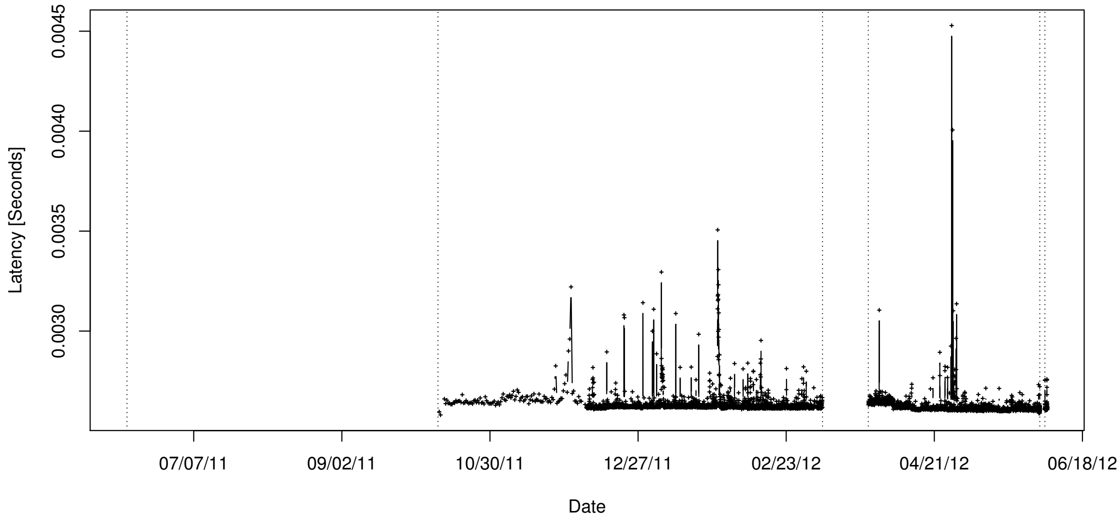


Figure 5.15: Latency dataset for network segment Luxembourg-Nancy.

any period in which the system could be with high availability risk. In particular, during the measured period, the maximum observed latency is below $5ms$. Through a simple analysis of the latency values represented in Figure 5.15, it becomes obvious that this latency should not pose any availability risk for a high-speed network connection and that the availability risk should always be set to 1 (no risk).

Due to the described constraint related to the observed latency values, it is necessary to change the way the measured latency values are interpreted in order for it to become possible to design a meaningful example. In order to create a scenario with circumstances that may lead to a risk situation, the boundary for what is considered a risk situation is artificially lowered according to the intervals expressed in Table 5.3.

Table 5.3: Risk level / Latency measurement.

Risk level	Latency in Seconds
1	>= 0 and < 0.00265
2	>= 0.00265 and < 0.0027
3	>= 0.0027 and < 0.0033
4	>= 0.0033 and < 0.005
5	>= 0.005

Based on the available latency information and applying the intervals defined in Table 5.3, the service risk for the network segment Luxembourg-Nancy is calculated. It is now assumed that this service risk is received from the network Operator. The evaluated service risk is described by the graph represented in Figure 5.16. From observing Figure 5.15 it was already noticed that some values are missing. This fact is now clearly noticeable by the analysis of Figure 5.16. The periods on which no measured values are available for the latency, are highlighted with vertical dotted lines in both Figure 5.15 and Figure 5.16. As it will be described, those missing observations will be employed to evaluate the Behaviour Trust.

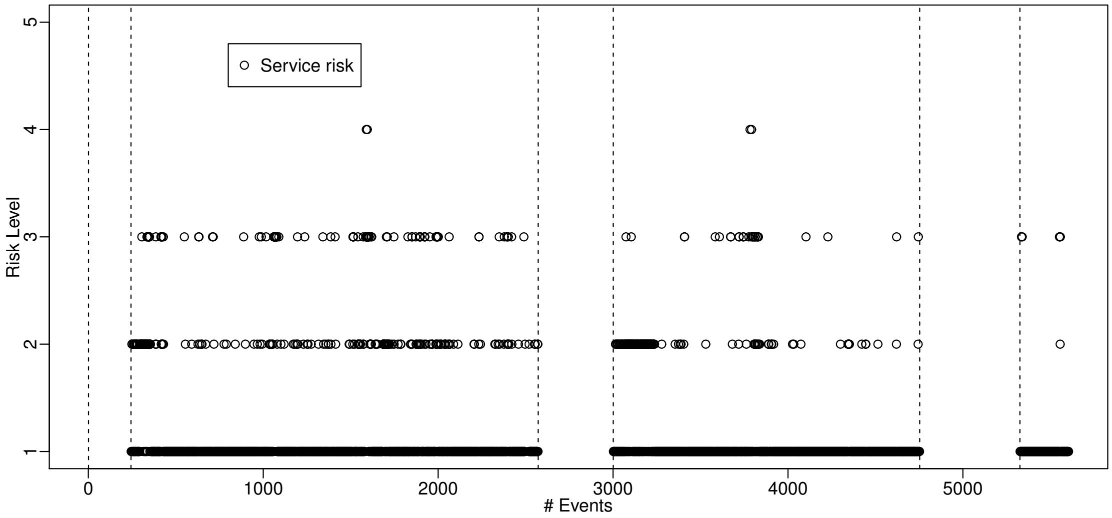


Figure 5.16: Service risk for network segment Luxembourg-Nancy.

Experimental set-up

As the objective of this case study is to prove the relevance of the Trust and Reputation indicators in permitting to improve the confidence a CI has on each received service risk, the CI Security Model and the TRS are put together in this case study. The prepared experimental set-up is represented in Figure 5.17.

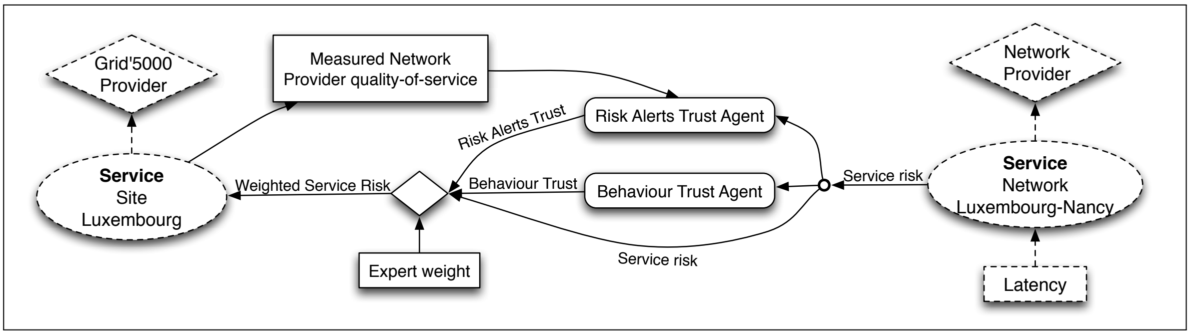


Figure 5.17: Case study experimental set-up overview.

In the presented set-up, two services from two different CIs, are defined. The service *Network segment Luxembourg-Nancy* which is part of the Network Provider CI and the service *Site Luxembourg* which is part of the Grid'5000 CI. This set-up allows the demonstration of the application on a scenario on which two Critical Infrastructures are willing to exchange risk information in order to improve security. In this scenario, it is clear that the service *Site Luxembourg* depends on the service *Network segment Luxembourg-Nancy.*

The service *Network segment Luxembourg-Nancy* evaluates the service risk for the network segment and periodically sends this information to all services that depend on this network. In this example, the service *Site Luxembourg* receives the service risk from the service *Network segment Luxembourg-Nancy* and evaluates the Risk Alerts Trust and the Behaviour Trust. The main goal is to combine these indicators, together with the initial Expert weight to calculate the Weighted Service Risk from the received service risk.

As already described, the Risk Alerts Trust is evaluated by comparing the received service risk with an independent measure of the provided service. In this case, it is mandatory to gather the Quality-of-Service measurement as experienced by the service *Site Luxembourg.* This measurement is represent in Figure 5.17 and titled *Measured Network Provider Quality-of-Service.* This measurement does not exists in the actual *Site Luxembourg* and even if some measurement in the grid site was deployed, it is expected that the measurements would not significantly differ from the measurements made by the network provider. With this fact in mind, it was necessary to emulate the measurement tools and generate values for the Measured Network Provider Quality-of-Service.

The values generated to simulate the Measured Network Provider Quality-of-Service pretends to create a scenario where the measured values normally represent a good Quality-of-Service (value 1, means no risk), and in some period, the received and measured values have some significant mismatch. To create such a data set, the values are set to 1 and only in the period between 1000 and 2000 events, a significant mismatch with the received risk level exists, on which the generated values are varying between 3 and 5. In this period, the values were generated randomly, in R (R Development Core Team, 2009), according to the following criteria: 30% of the values are set to 5, 50% are set to 4 and 20% are set to 3. The generated values representing the Measured Network Provider Quality-of-Service are shown in the graph represented in Figure 5.18.

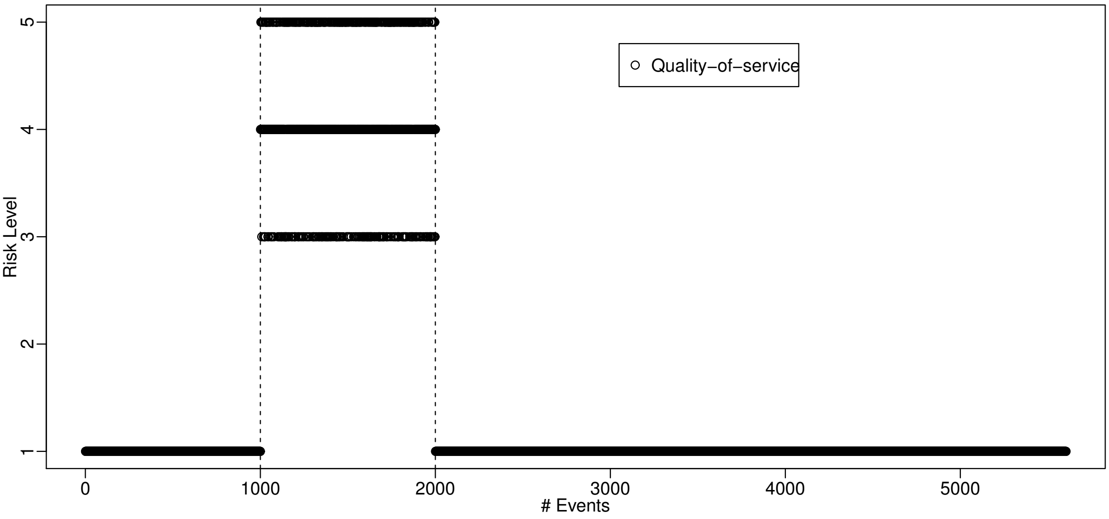


Figure 5.18: Network provider Quality-of-service for network segment Luxembourg-Nancy.

In an environment where CIs are exchanging service risk information, it is assumed that a risk level has to be sent by the network provider at fixed and previously defined time intervals. Although it is possible to define any time interval, changing its value implies that this information must be passed to the site Operator allowing him to reconfigure the TRS parameters according to the new used time interval.

As previously mentioned, the available data records for latency (Figure 5.15) has some missing measurements or there are no measurements available during some time periods. This fact is clearly visible on the data set as some records are stored with the information NaN (*Not a Number*). It was impossible to find the reason for this fact but it shows that it occurs in real systems and shall be treated appropriately. The three longer time periods, containing more that 50 NaN values in a row, are indicated in the charts using vertical dotted lines. In addition to NaN records that appear sporadically, smaller intervals containing only NaN values, are also present. These cases are not represented in Figure 5.15 as it would be difficult to do so while preserving correct interpretation.

In previous examples, in the case that a risk value is not received within the defined period, the last received value is used. This approach has shown that it is possible to minimise the impact of this assumption when adding Behaviour Trust indicators. On the other hand, if the CI Operator is just focusing on received service risk he has no means of understanding that something incorrect is happening. In this case, even if he did not receive any value, he would see a straight line on the graph that represents the risk of the service on which it depends. To avoid this situation, it is defined that when a NaN value is received (or nothing is received) during a defined time interval, the service risk assumes the value 1 (no risk) when the first missing value is detected. If more sequential omitted values are identified, the received risk is successively increased until it reaches 5 (maximum risk). Basically, when the site does not receive information from the network provider, an increasing risk is assumed, as demonstrated in Figure 5.19.

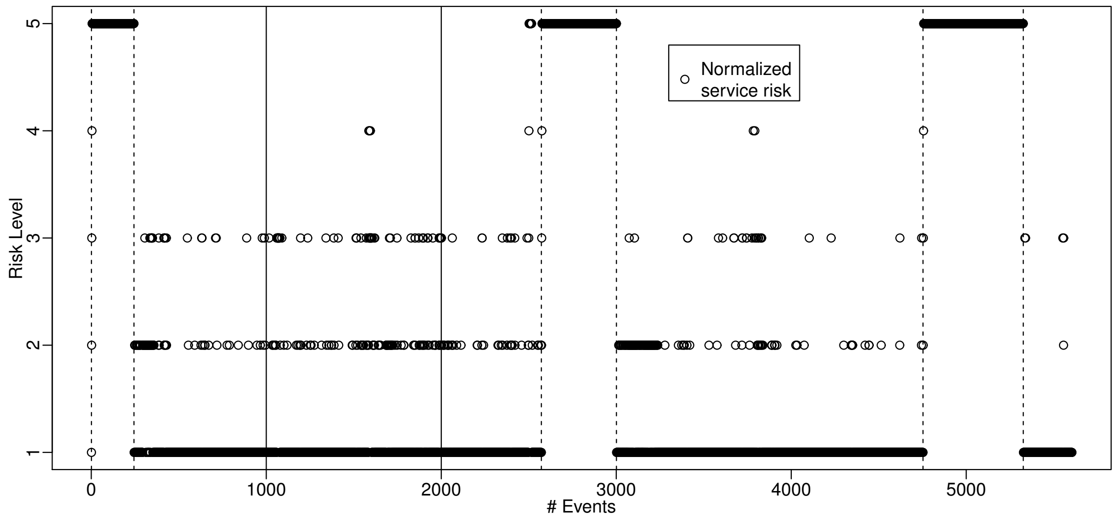


Figure 5.19: Normalized service risk for network segment Luxembourg-Nancy.

It seems dangerous to assume the worst-case when no information is received from the network provider. On first thought is seems that the site will raise its risk level and consequently the Grid'5000 risk level will also increase. All this can happen during normal operation leading to an unnecessary and unwanted risk increase. These are normal thoughts and this would be the case without the contribution of the Behaviour Trust indicator.

From the first analysis of the available data it is clear that the Behaviour Trust, in this scenario, is to be evaluated based on the absence of information or if incorrect information is received (incorrect information was not found in the data). As described in Table 5.4, if an NaN value (or nothing, unknown or incorrect) is received during the defined slot of time, this is perceived as anomalous behaviour and the Behaviour Trust is reduced. The first NaN value that appears triggers a behaviour event with a value of 80 (in a scale of 1..100). Next successive missing values trigger events with a value diminishing by 20 until it reaches 0. When a valid value is received on time, a value of 100 is triggered. Consequently, the high-risk values assumed during periods where no service risk is received, will have less influence on the service risk estimation. In order to better understand this last statement, it is important to recall that the goal of using trust indicators, is to make it possible to dynamically adjust the influence (weight) a received service risk has on the receiver's risk evaluation.

Table 5.4: Normalisation table for service risk behaviour

Behaviour event value / Service risk	
Behaviour Event Value	Description
100	Valid value received on time
80	NaN (Not a Number) or invalid value received or nothing received on time
[60..0]	Next successive missing, NaN or invalid values trigger events with a value diminishing by 20 until it reaches 0

The Grid'5000 case study scenario has been simulated using both the R tool (R Development Core Team, 2009) and the Java application described in Section 4.4. In this section, all data used to perform the simulation has been identified. Specifically, according to Figure 5.17 , following information and configuration parameters are used to evaluate the intended indicators:

- Service Risk: Normalised service risk as presented in Figure 5.19;
- Quality-of-Service: Generated values represented in Figure 5.18;
- Risk Alerts Trust: Penalisation factor $k = 1.25$; ageing factor $D = 0.3$;
- Behaviour Trust: Time slot size $\Delta t = 2$ and ageing factor $D = 0.3$;
- Service trust: Evaluated using the following weights: 60% to the Risk Alerts Trust and 40% to the Behaviour Trust;
- Expert weight: $\omega_E = 80\%$.

Results

In order to compute Risk Alerts Trust, the service risk received from the network Operator (Figure 5.19) is evaluated against the site measured QoS (Figure 5.18) as described in previous sections. The behaviour of the system is observed, measured and normalised according to the available entities and previously defined security policies. In this case, the entity used is the observation and analysis of the received service risk, according to the rules defined in Table 5.4.

For this case study, the results obtained for the Risk Alerts Trust and Behaviour Trust indicators are represented in Figure 5.20.

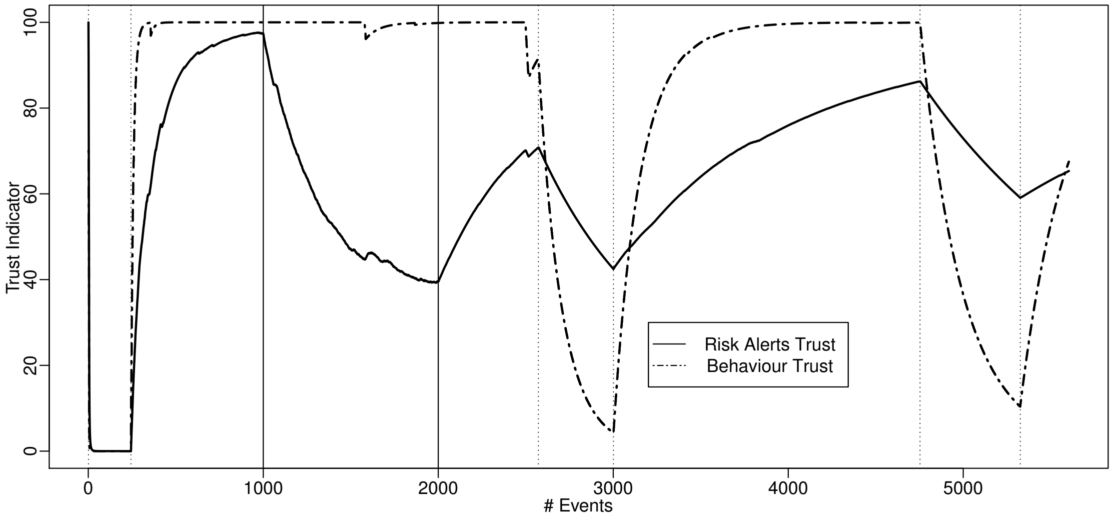


Figure 5.20: Trust indicators for network segment Luxembourg-Nancy.

From the indicators represented in Figure 5.20 the comportment of the Behaviour Trust indicator is noticeable. That, as expected, immediately starts to decay during the periods where no information is received from the service Operator and rapidly returns to an high value when the system behaves as expected (receiving service risk values on a regular basis). As described, the available data has some large blocks of NaN record (marked in the graphs) and also has some isolated or small blocks of NaN values. In these cases, a few missing values have a low influence on the Behaviour Trust indicator leading just to a small decrease with a rapid return to the original value. In this case, it can be affirmed that missing just a few values will have a low influence on the confidence one has in the received service risk alert.

Observing the Risk Alerts Trust represented in 5.20, it is visible that it decays in two different situations. During the interval from the 1000th to the 2000th event, the indicators decreases although not in a linear form. This fact is related to the defined measured risk (Quality-of-Service) observed by the site. As described, this measured risk is defined as 1 except for interval [1000..2000] to which, an important discrepancy between the measured service risk and the received service risk, was introduced. This discrepancy causes the Risk Alerts Trust indicator to decline according to the differences in the observed values before starting to progressively raise. The second situation in which the trust, in the received service risk, degrades, is when it is not possible to evaluate it as usual, due to the lack of values. It should be remembered that when a value is not received or in periods where no information is received, it is assigned an increasing risk to the service risk used in the evaluation. With this information, it becomes clear that a discrepancy between the values is also encountered leading to a decrease in the Risk Alerts Trust during the periods where no information is received.

From the original CI Security Model, each received risk information has a predefined weight specified by a CI expert. In this example, the expert fixed the contribution of the network segment Luxembourg-Nancy to the site Luxembourg as being 80%. In this case, regardless of the other indicators, this is the maximum influence the received risk can have on the risk indicator evaluated on Site Luxembourg. The main objective for this case study is to allow the weight to vary based on the dynamically evaluated confidence in the received service risk. Figure 5.21 describes how the service trust indicator (comprising Risk Alerts Trust and Behaviour Trust) contributes to the weight that the received risk alert has on the CI Security Model. With the maximum value defined by the expert as 80%, the final weight (expert*service trust), evaluated according to Equation 5.9, will change gradually in accordance with the evaluated service trust indicator.

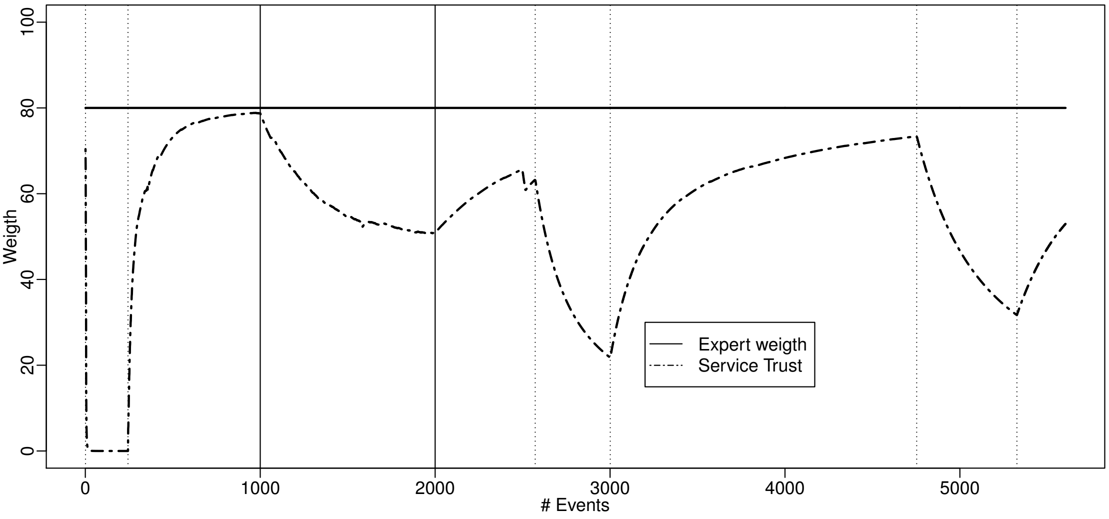


Figure 5.21: Service risk weights for network segment Luxembourg-Nancy.

The information received from the network segment Luxembourg-Nancy, considered in this case study, is shown in Figure 5.19. This information, risk level for the network segment as observed by the service provider, is incorporated into the Site Luxembourg risk assessment with a contribution defined by the weights presented in Figure 5.21 (expert*service trust). The result of this contribution is represented in Figure 5.22 that illustrate the final weighted risk level (the ingress risk indicator to Service A in Figure 5.10). It is visible that when the service trust decreases, less weight is assumed for the received risk and consequently a low risk level is maintained.

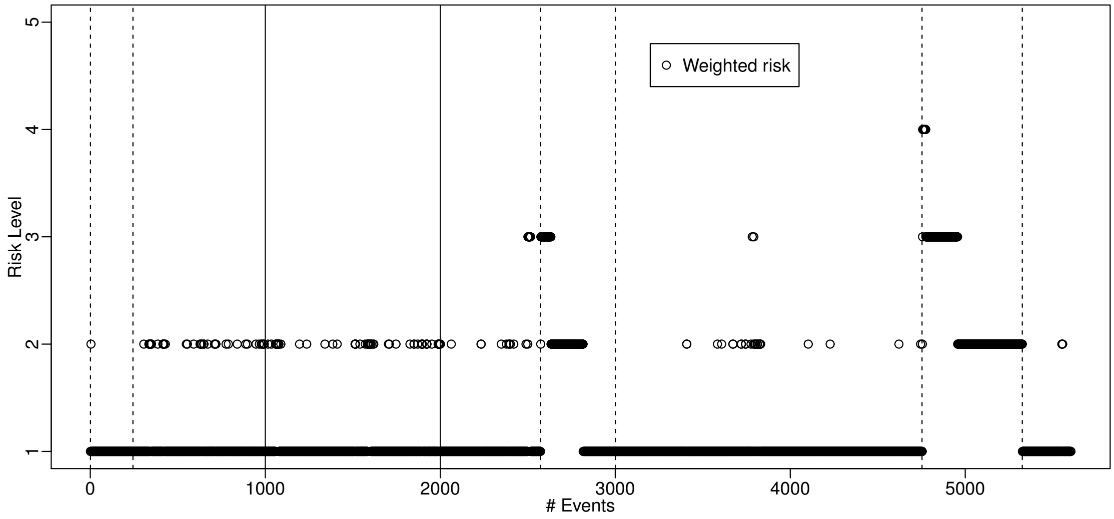


Figure 5.22: Weighted service risk for network segment Luxembourg-Nancy.

As mentioned in Section 5.1.3, in the CI Security Model, each weight denotes the impact a risk alert received from a dependency has on the aggregated risk of a service. In this context, the fact that one may have a low confidence on a received risk, the importance of this received risk has to the service should be lowered. Consequently, receiving high-risk alerts from one low trusted dependency will represent only a low risk for the service.

Simulation results from the Java application

As described in Section 4.4, a Java application implementing the TRS discussed in this thesis was developed as a proof of concept. With this tool it is possible to describe a scenario represented with the CI Security Model and compute the Risk Alerts Trust and Behaviour Trust indicators. The developed tool implements web services that allows to receive real time data from Critical Infrastructures, or to act as a simulator by reading data records from XML files. As already stated, the trust based dependency weighting presented in this section is fully implemented as well as the approach where trust is used to evaluate assurance for the correctness of CI service risks.

In Figure 5.23, a snapshot of the application interface simulating the Grid'5000 use case scenario, is presented. The simulation results for the Risk Alerts Trust as well as the Behaviour Trust are shown. These indicators are identical to the previously presented results with the exception that the Behaviour Trust indicator was normalised to a scale of [1..5] to be compliant with the approach described in Section 5.2.

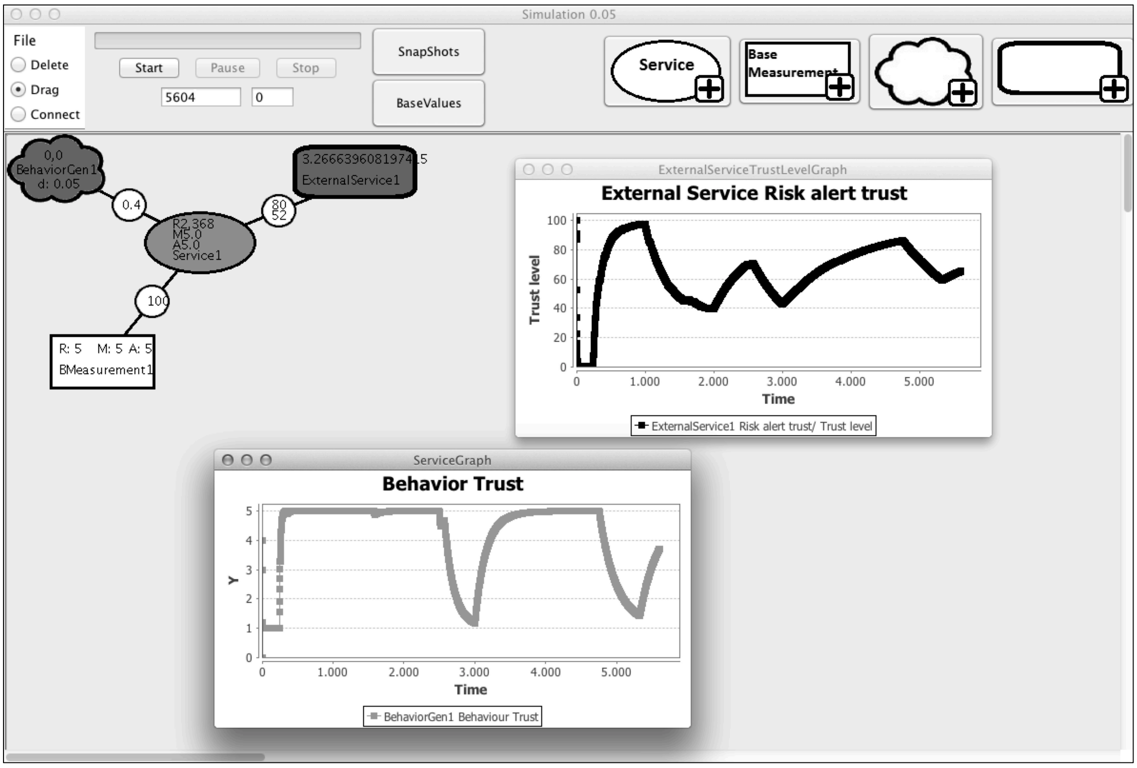


Figure 5.23: Results obtained from the developed proof of concept application.

5.3.3 Discussion

To be able to test the TRS using real data was an exciting challenge despite the fact that it was not possible to implement the presented approach (in real time) on the site Luxembourg.

For all the indicators, the set of performed simulations performed as expected and proved that, by introducing trust indicators, it is possible to improve the accuracy and the way the weighted risk levels are computed, thus allowing a more accurate risk information to be used. This expected and wanted comportment allow a CI or CI service not to raise risk level based on untrusted information as it is clear that a CI should not raise it's alert state (high risk) without a good reason (trust in the information sources).

This case-study allowed also to test and validate the simulation tools, in particular the Java application by comparing the result with the original TRS tool, developed using the statistical R Project for Statistical Computing (R Development Core Team, 2009).

5.4 Summary

This chapter presented the evaluation of the TRS when applied within the CI Security Model and evaluated it according to different aspects introduced in the three described scenarios.

The CI Security Model can be seen as a simple, flexible nevertheless a powerful CI modelling technique, allowing to enable risk evaluation on CIs that depend on services provided by other CIs (or CI services) and on which risks or failures in a dependency can cascade and cause service disruptions. The CI Security Model addresses these issues and provides a methodology for risk based dependency monitoring. However, as originally defined, the receiving CI service has no means of evaluating the correctness of the received risk information. The described validation works allowed evaluating the TRS on multiple scenarios based on the CI Security Model, confirming and highlighting the TRS high level of flexibility and adaptability. This work also contributed to the improvement of the CI Security Model by addressing some of its shortcomings, namely the introduction of risk indicators that allow the model to gather intelligence and act more independently from the CI expert thus providing better indicators to the CI Operator, in particular when used in a real-time environment.

For the scenarios described in Sections 5.1 and 5.3, the method used to weigh each CI/service dependency in the CI risk evaluation, was improved by applying the method named trust based dependency weighting. In this case, the trust indicators, Risk Alerts Trust and Behaviour Trust were introduced in order to evaluate the correctness of information received from dependencies. These trust indicators are used to define the importance of the risk information received from a dependency for risk estimation within a CI. Essentially, trusted information receives higher importance than untrusted information. The use of trust based dependency weighting can have a large impact on a CI risk estimation, as the trust one has on the information sources is now considered in the evaluation. This is the main objective of the presented approach as defined for the MICIE project, to reason about the exchanged information.

Trust based dependency weighting has been simulated using only generated data and also based on a real case study with information gathered over a period of one year. Those two scenarios are similar to each other in the main objective - to evaluate the correctness of information received from dependencies – but differ in some aspects as for instance the use of the Behaviour trust or the approach followed to employ missing values in the evaluation. The real case study describes a dependency between Grid'5000, a distributed computing grid platform, with the network infrastructure used to provide network communication among computing sites. Different providers operate the computing grid and the network infrastructure. The results from the simulations suggest that the approach works as expected and that discrepancies between announced service risk and actually experienced Quality-of-Service, as well as faulty behaviour, can be captured by the Risk Alerts Trust and Behaviour Trust indicators.

In the second scenario, described in Section 5.2, the TRS is adapted for a different use, which has not been initially foreseen. In this case, it is possible to observe and reason about the received risk and also about the behaviour of the base measurements used to evaluate service risk. This was not clearly the use that was in mind when the TRS was first planned and allows highlighting it's adaptability to new applications. In this scenario, the service assurance level, representing the confidence in the correctness of the measurements that are used to evaluate the CI service risk, has been evaluated from the Risk Alerts Trust indicator, the Behaviour Trust indicator and from the base measurements assurance level defined by an expert. Also in this simulation it was possible to demonstrate that the application of the TRS can greatly improve a CI risk model by introducing new views over the information used to reason about risks. The approach presented in this particular example can be used in combination with the approach described in Sections 5.1 and 5.3 without any further modification.

The use of the developed Java application that allows simulating all the presented scenarios was shown. This application has been validated successfully against the initial TRS tool supported by the statistical R Project for Statistical Computing tool (R Development Core Team, 2009). In addition to allowing the performance of the described simulations, the tool is also capable of performing all the evaluations in real time simply by receiving the necessary data on the available web services.

The presented application scenarios allowed to demonstrate the applicability of the Trust and Reputation System in different scenarios, where the TRS contribution can clearly improve the risk estimate mechanisms within a Critical Infrastructure.

Chapter 6

Conclusions

The main subject addressed within this thesis was the use of Trust and Reputation Management aiming to improve Critical Infrastructure Protection. The main conclusions drawn from the work conducted and the issues to be addressed in future work are presented in this chapter. Section 6.1 presents a synthesis of this thesis, while Section 6.2 describes the contributions and the most relevant conclusions that resulted from this work. Section 6.3 contains some issues that should be addressed in future work.

6.1 Synthesis of the Thesis

In current days, Critical Infrastructures are complex systems on which modern society depends. Infrastructures such as energy suppliers, telecommunications providers or water distributors are common within advanced societies. Countries and their citizens depend on services provided by CIs in order to fulfil their daily activities. One actual major concern, resides in the way CIs are protected from external environments, that might cause, accidentally or intentionally, a quality decrease in the services provided by CIs or even cause the disruption of those services. The identified necessity to protect Critical Infrastructures has driven the development of several frameworks, techniques and mechanisms in order to increase the resilience of such important infrastructures.

The concept of Critical Infrastructure was first introduced in Chapter 2 while highlighting the problems that might arise in their operation. One risk already identified within CIs protection is the existence of interdependencies among them. These interdependencies pose a serious risk in the CIs operation as a failure in one CI can affect those who depend on it. This problem was discussed while describing some relevant approaches that deal with this subject. Critical Infrastructure modelling, simulation techniques and risk assessment frameworks have been addressed in order to understand their main characteristics and the major problems they intend to solve. The importance of the CI protection is highlighted by the description of three selected European projects that have made major contributions in the area of CI security and protection.

Moreover, throughout Chapter 2, some ICT specific aspects were discussed in order to locate the contributions presented in this thesis in the state of the art. Namely, existent work on Policy Based Management and methodologies to deal with Ontologies, were discussed. Furthermore, an overview of some existent Trust and Reputation models, focusing on their applicability to the context of Critical Infrastructures protection and on the information exchange among Critical Infrastructures, were also discussed.

The MICIE project was described in this thesis, as most of the research challenges that led to this thesis, were revealed within the MICIE project. Chapter 3 presented an overview of the MICIE project with particular emphasis on the components in which the author of this thesis has been actively involved. In particular, the Mediation System Design and the modelling activities, which includes the author's contribution (ICT-MC). It is also detailed, the SMGW Manager and a first approach for the inclusion of trust indicators within the project, both were contributed by the author of this thesis.

The MICIE system acts on a distributed environment composed of multiple heterogeneous CIs, that might depend on one or more services provided externally by others CIs. MICIE's main goals are to predict and to exchange risk information among peer CIs, across trusted or untrusted network infrastructures (e.g. Internet). During the research carried within the MICIE project, the lack of mechanisms which allow to observe and reason about the confidence one can have in the information received from peer CIs, was identified. Also, the relevance of observing and of considering the behaviour of the information sources, in order for it to become possible to infer trust information regarding that behaviour, was identified. These facts led to the contribution of the Trust and Reputation framework, described in Chapter 4.

Chapter 4 introduces the Trust and Reputation framework, aiming to allow the incorporation of Trust and Reputation indicators on the information exchanged among Critical Infrastructures and also on information coming from heterogeneous monitoring equipment. Although, the presented framework is independent from the MICIE project, since it can be applied in multiple contexts, the MICIE application scenario was described.

In particular, the developed Trust and Reputation System was introduced along with the description of the main components that make up the system, namely, the Risk Alerts Trust Agent, the Behaviour Trust Agent and the Discovery Tool. The indicators evaluated by the Trust and Reputation System were described, highlighting their contribution to the improvement of CI risk prediction and to the management of a risk alert sharing system.

The contribution of the proposed indicators was validated by simulation. The results obtained within a representative set of performed simulations, was presented in Chapter 4. The obtained results, highlight the Trust and Reputation indicator's ability to improve a system and CI Operator's capacity to deal with uncertainty, and to fulfil its mission, in a timely manner, for instance, in the presence of attacks, failures, or accidents.

In order to demonstrate the flexibility and applicability of the Trust and Reputation framework within different scenarios and by using different CI modelling methodologies, the TRS was applied in the CI Security Model (described in Section 2.3). In Chapter 5, three applications and validation scenarios were presented for the Trust and Reputation System, supported by the CI Security Model. The application scenarios incorporated new information sources (trust) in the original CI Security Model. The first two application scenarios were validated using simulation. The last presented application scenario was validated using real data and allowed to prove the applicability of both contributions (TRS and the CI Security Model) to a real test-bed scenario – The Grid'5000 project.

6.2 Main contributions

This section describes the main contributions of this thesis. It is possible to identify three main contributions that were described throughout this thesis. The first, contributed to the analysis of problems resulting from CI risk exchange among Critical Infrastructure. This analysis allowed to address, in an integrated manner, problems that result from a scenario where CIs are willing to share risk information, related to services they provide, in order to improve the risk prediction within the dependent Critical Infrastructures.

In this context, a Policy Based Management Architecture was proposed to improve the management of the MICIE Secure Mediation Gateway, responsible for implementing the risk exchange mechanisms. This contribution was integrated in the MICIE project, in particular, in the SMGW Manager by the development of a Policy Management Tool.

Frameworks intended to allow and improve CI modelling, risk prediction and risk exchange among Critical Infrastructures already exist, and are proved to be accurate (e.g. MICIE project). Even though such frameworks exist, and are able to safely merge risk information arriving from multiple sources, the lack of mechanisms allowing to observe and reason about the confidence one can have in the information collected from these sources, was identified. Also, it is important to understand the behaviour of the information sources in order for it to become possible to infer trust information regarding that behaviour.

The second contribution identified in this thesis is the conception of a Trust and Reputation framework able to infer trust information on exchanged information and also on the behaviour of the subjacent system. The proposed framework is able to help the CI Operator to reason about the exchanged information and also to dynamically include the risk assessment in the defined management policies. This allows the improvement of the security of the existent secure mediation gateway through, for instance, denying sending or receiving information from untrusted peers. The validation activities carried out during this work allow the highlighting of the relevance of introducing Trust and Reputation indicators to improve CI Protection.

The third contribution is to enable the integration of the Trust and Reputation framework in CI risk models. This contribution is to introduce a way of building a trust relationship among CIs. Based on the common abstract information they share, describing how trust can be used in the model to dynamically re-evaluate the impact a risk level received, from a dependency, has on the modelled risk in a CI. Consequently, improving its accuracy and its resilience to inconsistent information provided by dependent CIs. Specifically, the Trust and Reputation System is now part of the CI Security Model and can be used to reason about the exchanged information and also internally in one CI to reason about the information gathered from the field.

Within this work, three full functional prototypes were developed in order to implement the CI Policy Management Tool and the Trust and Reputation System. First, the proposed Policy Management Tool, described in this thesis, was integrated and delivered by the MICIE project. This tool is also able to integrate and reason, using information coming from the Trust and Reputation System. Second, in order to fully implement the Trust and Reputation System, two TRS prototypes were developed. The first was developed as an add-on to the MICIE project, allowing to be integrated within the MICIE Secure Mediation Gateway from where the necessary data for the evaluation is gathered. The second, completely implements a modelling tool which is able to, by using a simple graphical interface, model a CI as represented by the CI Security Model existent entities. This tool allows evaluating risk and trust indicators in real time, with data coming from the CI or by receiving previously prepared data, in order to simulate a specific scenario. The described tools were also used for the experimental work that allowed evaluating the proposals described in this thesis.

6.3 Future Work

The problematic of Critical Infrastructure Security and Protection is still the subject of an on-going effort in the research community. Although, the evaluation of the proposals presented in this thesis has shown some interesting results on improving the accuracy of risk prediction and the manageability and security of a system able to exchange risk information among CIs, there are still open-ended issues that demand further study.

An aspect that can be addressed in more detail, is the methodology used for the Trust and Reputation evaluation. From the described state of the art in the Trust and Reputation area, it is possible to note the existence of multiple approaches, each one with its particularities. A further study on the presented Trust and Reputation framework, should include different approaches on how to derive trust from the collected evidence, for instance by using Bayesian Networks.

One particular aspect that could be addressed, is the improvement of the presented Trust and Reputation System by diminishing the CI expert influence on the definition of the TRS parameters. For instance, one should evaluate the applicability of dynamically defining the ageing factor for each CI or CI service, by incorporating the results of the Behaviour Trust indicator into a new dynamic ageing factor.

Another aspect that was kept open in this work is the possible use of shared reputation services, able to use the intelligence gathered from multiple CIs collectively, to determine the reputation of a specific CI, based on the trust each (inter)dependent CI has in each partner. This possibility was not evaluated in this work and needs special attention to maintain information source confidentiality.

Moreover, an interesting work, although difficult to achieve due to the intricate constrains, is the implementation of a test bed within a set of Critical Infrastructures, on which the presented proposals would be validated during a larger period of time, in a real scenario.

References

Adar, E. and Wuchner, A. (2005). Risk Management for Critical Infrastructure Protection (CIP) Challenges, Best Practices & Tools. In Hämmerli, B. and Wolthusen, S., editors, *Proceedings of the First IEEE International Workshop on Critical Infrastructure Protection*, IWCIP '05, pages 90–100, Darmstadt, Germany, 3–4 November 2005. IEEE Computer Society.

Aime, M. D. and Lioy, A. (2005). Incremental Trust: Building Trust from Past Experience. In Gligor, V., Martinelli, F., and Molva, R., editors, *Proceedings of the First International IEEE WoWMoM Workshop on Trust, Security and Privacy for Ubiquitous Computing - Volume 03*, WOWMOM '05, pages 603–608, Taormina - Giardini Naxos, Italy, 13–16 June 2005. IEEE Computer Society.

Artz, D. and Gil, Y. (2007). A survey of trust in computer science and the Semantic Web. In Wallace, E., editor, *Software Engineering and the Semantic Web*, volume 5 of *Web Semantics: Science, Services and Agents on the World Wide Web*, pages 58–71. Elsevier Science Publishers.

Aubert, J., Schaberreiter, T., Incoul, C., and Khadraoui, D. (2010a). Real-Time Security Monitoring of Interdependent Services in Critical Infrastructures. Case study of a Risk-Based Approach. In Ale, B., Papazuglo, I., and Zio, E., editors, *Proceedings of the European 21th Safety and Reliability Conference 2010 (ESREL 2010)*, Rhodes, Greece, 5-9 September 2010. Taylor & Francis Group.

Aubert, J., Schaberreiter, T., Incoul, C., Khadraoui, D., and Gateau, B. (2010b). Risk-Based Methodology for Real-Time Security Monitoring of Interdependent Services in Critical Infrastructures. In Takizawa, M., Tjoa, A. M., Aleksy, M., Ghernouti-Hélie, S., Quirchmayr, G., and Weippl, E., editors, *Proceedings of the Fifth International Conference on Availability, Reliability, and Security (ARES '10)*, pages 262–267, Krakow, Poland, 15-18 February 2010. IEEE Computer Society.

Baiardi, F., Telmon, C., and Sgandurra, D. (2009). Hierarchical, Model-based Risk Management of Critical Infrastructures. In Aven, T., Vinnem, J. E., and Soares, C. G., editors, *Proceedings of the 18th European Safety and Reliability Conference (ESREL 2007)*, volume 94 of *Reliability Engineering & System Safety*, pages 1403–1415, Stavanger, Norway, June 25–27, 2007. Elsevier Science Publishers.

Bertoni, A., Ciancamerla, E., di Prospero, F., Lefevre, D., Minichino, M., Lev, L., Iassinovski, S., Foglietta, C., Oliva, G., Panzieri, S., di Giorgio, A., Liberati, F., Aubert, J., Caldeira, F., Simões, P., Harpes, C., and Pauplin, O. (2010a). *Interdependency modelling framework, indicators and models – Final Report.* Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.2.3. European Commission FP7.

Bertoni, A., Ciancamerla, E., Foglietta, C., Lefevre, D., Minichino, M., Cohen, A., Lev, L., Hunovich, T., Ohana, R., Holzer, R., Tanenbaum, D., Adar, A., Iassinovski, S., Oliva, G., Panzieri, S., Castrucci, M., Priscoli, F., di Giorgio, A., Liberati, F., Aubert, J., Incoul, C., Caldeira, F., and Simões, P. (2010b). *Interdependency modelling framework, indicators and models – Second Interim Report.* Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.2.2. European Commission FP7.

Bessani, A., Sousa, P., Correia, M., and Neves, N. (2007). Intrusion-tolerant protection for critical infrastructures. Technical Report di-fcul-tr-07-8, Department of Informatics, University of Lisbon.

Bolze, R., Cappello, F., Caron, E., Daydé, M., Desprez, F., Jeannot, E., Jégou, Y., Lanteri, S., Leduc, J., Melab, N., Mornet, G., Namyst, R., Primet, P., Quetier, B., Richard, O., Talbi, E.-G., and Irena, T. (2006). Grid'5000: a large scale and highly reconfigurable experimental grid testbed. In Dongarra, J. J. and Supinski, B. R. D., editors, *International Journal of High Performance Computing Applications*, volume 20(4), pages 481–494. Sage Publications, Inc.

Caldeira, F., Castrucci, M., Aubigny, M., Aubert, J., Macone, D., Monteiro, E., Rente, F., Simões, P., and Suraci, V. (2010a). Secure mediation gateway architecture enabling the communication among critical infrastructures. In Cunningham, P. and Cunningham, M., editors, *Proceedings of the Future Network and MobileSummit 2010 Conference*, Florence, Italy, 16–18 June 2010. IIMC International Information Management Corporation.

Caldeira, F., Monteiro, E., and Simões, P. (2010b). Trust and Reputation for Information Exchange in Critical Infrastructures. In Xenakis, C. and Wolthusen, S., editors, *Proceedings of the 5th International Workshop on Critical Information Infrastructures Security (CRITIS 2010)*, volume 6712 of *Lecture Notes in Computer Science*, pages 140–152. Springer Berlin Heidelberg, Athens, Greece, 23–24 September 2010.

Caldeira, F., Monteiro, E., and Simões, P. (2010c). Trust and reputation management for critical infrastructure protection. In Jahankhani, H. and Tenreiro de Magalhães, S., editors, *Special Issue on Global Security, Safety and Sustainability*, volume 3(3) of *International Journal of Electronic Security and Digital Forensics*, pages 187–203. Inderscience Publishers.

Caldeira, F., Monteiro, E., and Simões, P. (2010d). Trust and reputation management for critical infrastructure protection. In Tenreiro de Magalhães, S., Jahankhani, H., and Hessami, A. G., editors, *Proceedings of the 6th International Conference on Global Security, Safety, and Sustainability (ICGS3 2010)*, volume 92 of *Communications in Computer and Information Science*, pages 39–47. Springer Berlin Heidelberg, Braga, Portugal, 1–3 September 2010.

Caldeira, F., Schaberreiter, T., Monteiro, E., Aubert, J., Simões, P., and Khadraoui, D. (2011). Trust based interdependency weighting for on-line risk monitoring in interdependent critical infrastructures. In Cuppens, F., Foley, S., Groza, B., and Minea, M., editors, *Proceedings of the Sixth International Conference on Risks and Security of Internet and Systems (CRiSIS 2011)*, pages 1–7, Timisoara, Romania, September 26-28, 2011. IEEE Computer Society.

Caldeira, F., Schaberreiter, T., Varrette, S., Monteiro, E., Simões, P., Bouvry, P., and Khadraoui, D. (2013). Trust based interdependency weighting for on-line risk monitoring in interdependent critical infrastructures. In Khan, K. M., editor, *International Journal of Secure Software Engineering (IJSSE)*, volume 4(4). IGI Global.

Capit, N., Da Costa, G., Georgiou, Y., Huard, G., Martin, C., Mounié, G., Neyron, P., and Richard, O. (2005). A batch scheduler with high level components. In Walker, D. W. and Kesselman, C., editors, *Proceedings of the Fifth IEEE International Symposium on Cluster Computing and the Grid (CCGrid'05)*, volume 2, pages 776–783, Cardiff, Wales, United Kingdom, May 9–12, 2005. IEEE Computer Society.

Capodieci, P. (2011). *Project Final Report.* Capodieci, P., editor. MICIE Project Final Report. European Commission FP7.

Capodieci, P., Diblasi, S., Ciancamerla, E., Minichino, M., Foglietta, C., Lefevre, D., Oliva, G., Panzieri, S., Setola, R., De Porcellinis, S., Priscoli, F., Castrucci, M., Suraci, V., Lev, L., Shneck, Y., Khadraoui, D., Aubert, J., Iassinovski, S., Jiang, J., Simões, P., Caldeira, F., Spronska, A., Harpes, C., and Aubigny, M. (2010). Improving Resilience of Interdependent Critical Infrastructures via an On-Line Alerting System. In Rizzo, A., editor, *Proceedings of the first International Conference COMPENG 2010, "Complexity in Engineering"*, pages 88–90, Rome, Italy, 22–24 February 2010. IEEE Computer Society.

Cardoso, J. (2007). The Semantic Web Vision: Where Are We? In Hendler, J., editor, *IEEE Intelligent Systems*, volume 22(5), pages 84–88. IEEE Computer Society.

Castrucci, M., Macone, D., Harpes, C., Pascoli, Aubigny, M., Aubert, J., Incoul, C., Gateau, B., Khadraoui, D., Panzieri, S., Oliva, G., Silvestri, Caldeira, F., Rente, F., Simões, P., Lev, L., and Tanenbaum, D. (2009). *MICIE ICT System Requirements – Preliminary Version.* Castrucci, M., editor. MICIE Project Deliverable D4.1.1. European Commission FP7.

Castrucci, M., Macone, D., Harpes, C., Pascoli, Aubigny, M., Aubert, J., Incoul, C., Khadraoui, D., Panzieri, S., Oliva, G., Caldeira, F., Rente, F., Simões, P., Lev, L., Tanenbaum, D., Minichino, M., and Ciancamerla, E. (2010a). *MICIE ICT System Requirements – Final Version.* Castrucci, M., editor. MICIE Project Deliverable D4.1.2. European Commission FP7.

Castrucci, M., Macone, D., Suraci, V., Inzerilli, T., Neri, A., Panzieri, S., Foglietta, C., Oliva, G., Aubert, J., Incoul, C., Caldeira, F., Aubigny, M., Harpes, C., and Kloda (2010b). *Secure Mediation Gateway Architecture – Final Version.* Castrucci, M., editor. MICIE Project Deliverable D4.2.2. European Commission FP7.

Castrucci, M., Neri, A., Caldeira, F., Aubert, J., Khadraoui, D., Aubigny, M., Harpes, C., Simões, P., Suraci, V., and Capodieci, P. (2012). Design and implementation of a mediation system enabling secure communication among critical infrastructures. In Shenoi, S., editor, *International Journal of Critical Infrastructure Protection*, volume 5(2), pages 86–97. Elsevier Science Publishers.

Chen, J., Lu, H., and Bruda, S. (2009a). Analysis of Feedbacks and Ratings on Trust Merit for Peer-to-Peer Systems. In Wang, W., editor, *Proceedings of the 2009 International Conference on E-Business and Information System Security (EBISS '09)*, pages 1–5, Wuhan, China, 23–24 May 2009. IEEE Computer Society.

Chen, S., Zhang, Y., and Yang, G. (2009b). Trust and reputation algorithms for unstructured p2p networks. In Wu, J. and Ye, Z., editors, *Proceedings of the First International Symposium on Computer Network and Multimedia Technology (CNMT 2009)*, pages 1–4, Wuhan, China, 18-20 December 2009. IEEE Computer Society.

Ciancamerla, E., di Blasi, S., Fioriti, V., Foglietta, C., Minichino, M., Lefevre, D., Cohen, A., Lev, L., Hunovich, T., Ohana, R., Holzer, R., Tanenbaum, D., Adar, A., Iassinovski, S., Menezes, N., de Porcellinis, S., Oliva, G., Panzieri, S., di Giorgio, A., Liberati, F., Aubert, J., Incoul, C., Caldeira, F., Rente, F., and Jiang, J. (2009). *Interdependency modelling framework, interdependency indicators and models – First Interim Report*. Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.2.1. European Commission FP7.

Ciancamerla, E., Foglietta, C., Lefevre, D., Minichino, M., Lev, L., and Shneck, Y. (2010a). Discrete event simulation of qos of a scada system interconnecting a power grid and a telco network. In Berleur, J., Hercheui, M., and Hilty, L., editors, *What Kind of Information Society? Governance, Virtuality, Surveillance, Sustainability, Resilience. Proceedings of the 9th IFIP TC 9 International Conference, HCC9 2010 and 1st IFIP TC 11 International Conference, CIP 2010, Held as Part of WCC 2010*, volume 328 of *IFIP Advances in Information and Communication Technology*, pages 350–362, Brisbane, Australia, September 20–23 2010. Springer Berlin Heidelberg.

Ciancamerla, E., Minichino, M., Lev, L., Simoes, P., Panzieri, S., Oliva, G., Foglietta, C., and Aubert, J. (2010b). *CI Reference Scenario and service oriented approach – Final Report*. Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.1.2. European Commission FP7.

Clinton, W. J. (1996). Executive Order 13010 - Critical Infrastructure Protection. *USA Federal Register*, 6 I(138):37347.

CockpitCI (2013). CockpitCI Project Web Site. [online] `http://www.cockpitci.eu`.

CRUTIAL (2008). CRUTIAL Project Web Site. [online] `http://crutial.rse-web.it`.

Damianou, N. (2002). *A Policy Framework for Management of Distributed Systems.* PhD thesis, March, 2002, Department of Computing, Imperial College, London.

DAML (2001). The DARPA Agent Markup Language Homepage. [online] `http://www.daml.org/2001/03/daml+oil-index.html`.

D'Antonio, S., Romano, L., Khelil, A., and Suri, N. (2009). INcreasing Security and Protection through Infrastructure REsilience: The INSPIRE Project. In Setola, R. and Geretshuber, S., editors, *Proceedings of the 3th International Workshop on Critical Information Infrastructures Security (CRITIS 2008)*, volume 5508 of *Lecture Notes in Computer Science*, pages 109–118. Springer Berlin Heidelberg, Rome, Italy, October13–15 2008.

De Porcellinis, S., Panzieri, S., and Setola, R. (2009). Modelling critical infrastructure via a mixed holistic reductionistic approach. In Chaudet, C., Grand, G. L., and Rosat, V., editors, *Special Issue on Critical Infrastructures as Complex Systems*, volume 5(1/2) of *International Journal of Critical Infrastructures*, pages 86–99. Inderscience Publishers.

De Porcellinis, S., Panzieri, S., Setola, R., and Ulivi, G. (2008). Simulation of heterogeneous and interdependent critical infrastructures. In Bologna, S., editor, *Special Issue on Complex Network and Infrastructure Protection*, volume 4(1/2) of *International Journal of Critical Infrastructures*, pages 110–128. Inderscience Publishers.

Dondossola, G., Garrone, F., Szanto, J., and Gennaro, F. (2008). A laboratory testbed for the evaluation of cyber attacks to interacting ICT infrastructures of power grid operators. In de Jaeger, E., editor, *Proceedings of the CIRED Seminar: SmartGrids for Distribution, 2008*, pages 1–4, Frankfurt, Germany, 23–24 June 2008. IEEE Computer Society.

European Commission (2004). Communication from the Commission to the Council and the European Parliament - Critical Infrastructure Protection in the fight against terrorism. *Communication from European Commission*, COM (2004) 702 Final.

European Commission (2005). Green Paper on a European programme for Critical Infrastructure Protection. *Communication from European Commission*, COM

(2005) 576 Final.

European Commission (2006). Communication from the Commission on a European Programme for Critical Infrastructure Protection. *Communication from European Commission*, COM (2006) 0786 Final.

European Commission (2008). Proposal for a council decision on a Critical Infrastructure Warning Information Network (CIWIN). *Communication from European Commission*, COM (2008) 676.

European Commission (2012). European Commision - Home Affairs. [online] `http://ec.europa.eu/home-affairs/policies/terrorism/terrorism_infrastructure_en.htm`.

Falliere, N., Murchu, L. O., and Chien, E. (2011). W32.Stuxnet Dossier. Technical report, Symantec - Security Response, [online] `http://www.symantec.com/connect/blogs/w32stuxnet-dossier`.

Farwell, J. P. and Rohozinski, R. (2011). Stuxnet and the Future of Cyber War. In Allin, D., editor, *Survival*, volume 53(1), pages 23–40. Routledge.

Gambetta, D. (2000). Can We Trust Trust? In Gambetta, D., editor, *Trust: Making and Breaking Cooperative Relations, electronic edition*, chapter 13, pages 213–237. Department of Sociology, University of Oxford.

Ganeriwal, S., Balzano, L., and Srivastava, M. (2008). Reputation-based framework for high integrity sensor networks. In Lu, C., editor, *ACM Transactions on Sensor Networks (TOSN)*, volume 4(3), pages 15:1–15:37. ACM.

Gasparri, A., Oliva, G., and Panzieri, S. (2009). On the distributed synchronization of on-line IIM Interdependency Models. In Pham, D.-T. and Colombo, A., editors, *Proceedings of the 7th IEEE International Conference on Industrial Informatics (INDIN 2009)*, pages 795–800, Cardiff, Wales, United Kingdom, 24–26 June 2009. IEEE Computer Society.

Géant (2013). Géant web site. [online] `http://www.geant.net`.

Gómez Mármol, F. and Martínez Pérez, G. (2012). TRIP, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks. In Yang, L. T. and Wang, G., editors, *Special Issue on Trusted Computing and Communications*, volume 35(3) of *Journal of Network and Computer Applications*, pages 934–941. Elsevier Science Publishers.

Grid5000 (2013). The grid 5000 project web site. [online] `http://www.grid5000.fr`.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. In *Knowledge Acquisition*, volume 5(2), pages 199–220. Academic Press Ltd.

Haimes, Y., Kaplan, S., and Lambert, J. H. (2002). Risk Filtering, Ranking, and Management Framework Using Hierarchical Holographic Modeling. In *Risk analysis : an official publication of the Society for Risk Analysis*, volume 22(2), pages 383–398. Blackwell Publishers.

Hansson, S. O. (2012). Risk. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Winter 2012 edition.

Haslum, K. and Arnes, A. (2006). Multisensor real-time risk assessment using continuous-time hidden markov models. In Cheung, Y., Wang, Y., and Liu, H., editors, *Proceedings of the 2006 International Conference on Computational Intelligence and Security*, volume 2, pages 1536–1540, Guangzhou, China, 3–6 November 2006. IEEE Computer Society.

Hochstatter, I., Dreo, G., Serrano, M., Serrat, J., Nowak, K., and Trocha, S. (2008). An architecture for context-driven self-management of services. In Merrill, D., editor, *Proceedings of the 2008 IEEE INFOCOM Workshops*, pages 1–4, Phoenix, Arizona, USA, 13–18 April 2008. IEEE Computer Society.

Hussain, F., Chang, E., and Hussain, O. (2007). State of the art review of the existing bayesian-network based approaches to trust and reputation computation. In Berthier, Y., Chen, T., Mont, M., He, L., Labbe, P., Parr, G., State, R., Reda, R., and Sundhar, S., editors, *Proceedings of the Second International Conference on Internet Monitoring and Protection (ICIMP 2007)*, pages 26–26, Silicon Valley, California, USA, 1–5 July 2007. IEEE Computer Society.

Huynh, T., Jennings, N., and Shadbolt, N. (2006). An integrated trust and reputation model for open multi-agent systems. In Rosenschein, J. S. and Stone, P., editors, *Journal of Autonomous Agents and Multi-Agent Systems*, volume 13(2), pages 119–154. Springer Berlin Heidelberg.

INSPIRE (2010). INSPIRE Project Web Site. [online] `http://www.inspire-strep.eu`.

Inzerilli, T., Castrucci, M., Macone, D., Suraci, V., Aubert, J., Incoul, C., Caldeira, F., Rente, F., Simões, P., Aubigny, M., Pascoli, Harpes, C., Oliva, G., Kloda, and Szewczyk (2009). *Secure Mediation Gateway Architecture – Preliminary*

Version. Inzerilli, T. and Castrucci, M., editors. MICIE Project Deliverable D4.2.1. European Commission FP7.

IRRIIS (2008). IRRIIS Project Web Site. [online] `http://www.irriis.org`.

ISO/IEC (2005). ISO 27001: Information Security Management System (ISMS) standard. [online] `http://www.27000.org/iso-27001.htm`.

Jøsang, A. and Ismail, R. (2002). The beta reputation system. In Gricar, J., editor, *Proceedings of the 15th Bled Electronic Commerce Conference - eReality: Constructing the eEconomy (BLED 2002)*, Bled, Slovenia, 17–19 June 2002. AIS Electronic Library (AISeL).

Jøsang, A., Ismail, R., and Boyd, C. (2007). A survey of trust and reputation systems for online service provision. In Li, E. Y. and Du, T. C., editors, *Decision Support Systems*, volume 43(2), pages 618–644. Elsevier Science Publishers.

Kagal, L. (2005). Rei : A Policy Specification Language. [online] `http://rei.umbc.edu`.

Kagal, L., Finin, T., and Joshi, A. (2003). A Policy Language for a Pervasive Computing Environment. In Lutfiyya, H., Garcia, F., and Moffett, J., editors, *Proceedings of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY 2003)*, pages 63–74, Lake Como, Italy, 4–6 June 2003. IEEE Computer Society.

Klein, R. (2010). The EU FP6 Integrated Project IRRIIS on Dependent Critical Infrastructures. In Xenakis, C. and Wolthusen, S., editors, *Proceedings of the 5th International Workshop on Critical Information Infrastructures Security (CRITIS 2010)*, volume 6712 of *Lecture Notes in Computer Science*, pages 26–42. Springer Berlin Heidelberg, Athens, Greece, 23–24 September 2010.

Klein, R., Rome, E., Beyel, C., Linnemann, R., Reinhardt, W., and Usov, A. (2009). Information Modelling and Simulation in Large Interdependent Critical Infrastructures in IRRIIS. In Setola, R. and Geretshuber, S., editors, *Proceedings of the Third International Workshop on Critical Information Infrastructures Security (CRITIS 2008)*, volume 5508 of *Lecture Notes in Computer Science*, pages 36–47. Springer Berlin Heidelberg, Rome, Italy, 13–15 October 2008.

Lev, L., Hunovich, T., Ohana, R., Holzer, R., Tanenbaum, D., Adar, A., Ciancamerla, E., di Blasi, S., Fioriti, V., Foglietta, C., Minichino, M., de Porcellinis, S., Panzieri, S., Menezes, N., Caldeira, F., Rente, F., Simões, P., Castrucci, M., di Giorgio, A., Aubert, J., Incoul, C., Gateau, B., and Khadraoui, D. (2009). *Reference Scenario and service oriented approach – Interim Report.* Ciancamerla, E. and Minichino, M., editors. MICIE Project Deliverable D2.1.1. European Commission FP7.

Lev, L., Tanenbaum, D., Ohana, R., Holzer, R., Hunovich, T., Adar, A., Cohen, A., Capodieci, P., Minichino, M., Ciancamerla, E., Foglietta, C., Oliva, G., Simões, P., Caldeira, F., Castrucci, M., Bojar, K., Jager, Pascoli, Aubigny, M., and Harpes, C. (2011). *Validation Activities.* Lev, L. and Baruch, Y., editors. MICIE Project Deliverable D6.3. European Commission FP7.

Lev, L., Tanenbaum, D., Ohana, R., Holzer, R., Hunovich, T., Adar, A., Cohen, A., Capodieci, P., Minichino, M., Panzieri, S., Simões, P., and Castrucci, M. (2010a). *Demonstration Plan.* Lev, L. and Adar, A., editors. MICIE Project Deliverable D6.1. European Commission FP7.

Lev, L., Tanenbaum, D., Ohana, R., Holzer, R., Hunovich, T., Adar, A., Cohen, A., Capodieci, P., Neri, A., Minichino, M., Panzieri, S., Caldeira, F., Simões, P., Castrucci, M., Kloda, R., and Szewczyk, R. (2010b). *Integration Process Report.* Lev, L. and Adar, A., editors. MICIE Project Deliverable D6.2. European Commission FP7.

Li, W., Joshi, A., and Finin, T. (2013). CAST: Context-Aware Security and Trust framework for Mobile Ad-hoc Networks using Policies. In Chakraborty, D., Kalogeraki, V., and Mokbel, M., editors, *Special Issue: Mobile Data Management*, volume 31(2) of *Distributed and Parallel Databases*, pages 353–376. Springer US.

Li, W., Kodeswaran, P. A., Jagtap, P., Joshi, A., and Finin, T. (2012a). Chapter 22 - managing and securing critical infrastructure – a semantic policy- and trust-driven approach. In Sajal K. Das, K. K. and Zhang, N., editors, *Handbook on Securing Cyber-Physical Critical Infrastructure*, pages 551–572. Morgan Kaufman, Boston, USA.

Li, W., Parker, J., and Joshi, A. (2012b). Security Through Collaboration and Trust in MANETs. In Joshi, J., Bertino, E., Pu, C., and Ramampiaro, H., editors, *Special Issue on "Collaborative Computing: Networking, Applications and Worksharing"*, volume 17(3) of *Mobile Networks and Applications (MONET)*, pages 342–352. Springer US.

Ludwig, S., Pulimi, V., and Hnativ, A. (2009). Fuzzy approach for the evaluation of trust and reputation of services. In Jeon, H. T., Min, K. C., and Oh, K.-W., editors, *Proceedings of the18th IEEE International Conference on Fuzzy Systems (FUZZ–IEEE 2009)*, pages 115–120, Jeju Island, Korea, 20–24 August 2009. IEEE Computer Society.

Malik, Z. and Bouguettaya, A. (2009). Reputation bootstrapping for trust establishment among web services. In Rabinovich, M., editor, *IEEE Internet Computing*, volume 13(1), pages 40–47. IEEE Computer Society.

McClelland, R. (2010). Critical infrastructure protection national strategy. Attorney General's Department, Australian Government, Commonwealth of Australia.

McLeod, C. (2011). Trust. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Spring 2011 edition.

MICIE Consortium (2008). MICIE - Tool for systemic risk analysis and secure mediation of data exchanged across linked CI information infrastructures. *FP7-ICT-SEC-2007.1.7 – 225353 – Annex I – "Description of Work"*.

Miller, W., Voas, J., and Laplante, P. (2010). In Trust We Trust. In Vetter, R., editor, *Computer*, volume 43(10), pages 85–87. IEEE Computer Society.

Miyachi, T., Narita, H., Yamada, H., and Furuta, H. (2011). Myth and reality on control system security revealed by Stuxnet. In Sugano, S. and Sampei, M., editors, *Proceedings of the SICE Annual Conference (SICE 2011)*, pages 1537–1540, Tokyo, Japan, 13–18 September 2011. IEEE Computer Society.

Momani, M., Challa, S., and Alhmouz, R. (2008). BNWSN: Bayesian network trust model for wireless sensor networks. In Mahasneh, J., editor, *Proceedings of the Mosharaka International Conference on Communications, Computers and Applications (MIC–CCA 2008)*, pages 110–115, Amman, Jordan, 8–10 August 2008. IEEE Computer Society.

Moteff, J., Copeland, C., and Fischer, J. (2003). Critical infrastructures: What makes an infrastructure critical? Report for congress rl31556, The Library of Congress, DC, USA.

Moyano, F., Fernandez-Gago, C., and Lopez, J. (2012). Implementing Trust and Reputation Systems: A Framework for Developers'Usage. In Martinelli, F. and Nielson, F., editors, *Proceedings of the International Workshop on Quantitative Aspects in Security Assurance (QASA 2012) - Affiliated work-*

shop with ESORICS 2012, Pisa, Italy, 14 September 2012. [online] https://www.nics.uma.es/system/files/main_NICS.pdf.

Mui, L. and Mohtashemi, M. (2002). A computational model of trust and reputation. In Sprague, R. H., editor, *Proceedings of the 35th Hawaii International Conference on System Science (HICSS–35)*, volume 7, Big Island, Hawaii, USA, January 7–10, 2002. Computer Society Press.

NAREGI (2013). Naregi web site. [online] http://www.naregi.org.

NERC (2009). NERC - Critical Infrastructure Protection standards. [online] http://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx.

Neri, A. (2010). *Secure Mediation Gateway SW Beta Release.* Neri, A., editor. MICIE Project Deliverable D5.2. European Commission FP7.

NET-SCIP (2011). Innovation Network on Security and Critical Infrastructure Protection (NET-SCIP). [online] http://net-scip.cmuportugal.org.

NIST (2009). NIST Special Publication 800-53, Rev. 3 – Recommended Security Controls for Federal Information Systems and Organizations. [online] http://csrc.nist.gov/publications/PubsSPs.html.

Noorian, Z. and Ulieru, M. (2010). The state of the art in trust and reputation systems: a framework for comparison. In Cerpa, N., editor, *Journal of Theoretical and Applied Electronic Commerce Research*, volume 5(2), pages 97–117. Facultad de Ingenieria, Universidad de Talca.

OASIS (2013). eXtensible Access Control Markup Language (XACML) Version 3.0. 22 January 2013. OASIS Standard. [online] http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html.

Obama, B. (2013). Executive Order 13636 - Improving Critical Infrastructure Cybersecurity. *USA Federal Register*, (2013-03915):11737–11744.

Oliva, G., Panzieri, S., and Setola, R. (2010). Agent-based input–output interdependency model. In Shenoi, S., editor, *International Journal of Critical Infrastructures*, volume 3(2), pages 76–82. Inderscience Publishers.

Panzieri, S., Oliva, G., de Porcellinis, S., Ciancamerla, E., Minichino, M., Aubert, J., Incoul, C., Aubigny, M., and Iassinovski, S. (2009). *Refined interdependency metrics and indexes for risk prediction formulation – Preliminary Ver-*

sion. Panzieri, S., editor. MICIE Project Deliverable D3.1.1. European Commission FP7.

Panzieri, S., Oliva, G., Foglietta, C., Minichino, M., Ciancamerla, E., Macone, D., Castrucci, M., Suraci, V., Pauplin, O., Aubert, J., Caldeira, F., Simões, P., and Curado, M. (2010). *Common Ontology and Risk Prediction Algorithms – Final Version.* Panzieri, S., editor. MICIE Project Deliverable D3.2.2. European Commission FP7.

Panzieri, S., Setola, R., and Ulivi, G. (2005). An approach to model complex interdependent infrastructures. In Zítek, P., editor, *Proceedings of the 16th IFAC World Congress*, pages 67–67, Prague, Czech Republic, 4–8 July 2005. International Federation of Automatic Control.

Ponder (2010). Ponder2 project web site. [online] `http://ponder2.net/`.

Protégé (2011). Protégé web site. [online] `http://protege.stanford.edu`.

Puppet (2013). Puppet labs web site. [online] `http://puppetlabs.com/`.

R Development Core Team (2009). R: A Language and Environment for Statistical Computing. ISBN 3-900051-07-0. R Foundation for Statistical Computing. Vienna, Austria.

Ray, I., Ray, I., and Chakraborty, S. (2009). An interoperable context sensitive model of trust. In Kerschberg, L. and Ras, Z., editors, *Journal of Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies (JIIS)*, volume 32(1), pages 75–104. Springer US.

RDDTool (2012). Rddtool web site. [online] `http://oss.oetiker.ch/rrdtool/`.

Renater (2013). Renater web site. [online] `http://www.renater.fr/`.

Restena (2013). Restena web site. [online] `http://www.restena.lu/`.

Rinaldi, S. (2004). Modeling and simulating critical infrastructures and their interdependencies. In Sprague, R. H., editor, *Proceedings of the 37th Hawaii International Conference on System Science (HICSS–37)*, volume 2, page 20054a, Big Island, Hawaii, USA, January 5–8, 2004. Computer Society Press.

Rinaldi, S. M., Peerenboom, J. P., and Kelly, T. K. (2001). Identifying, Understanding, and Analyzing Critical Infrastructure Interdependencies. In Braatz, R. D., editor, *IEEE Control Systems Magazine*, volume 21(6), pages 11–25. IEEE Computer Society.

Ruohomaa, S. and Kutvonen, L. (2005). Trust Management Survey. In Herrmann, P., Issarny, V., and Shiu, S., editors, *Trust Management - Proceedings of the Third International Conference, iTrust 2005*, volume 3477 of *Lecture Notes in Computer Science*, pages 77–92. Springer Berlin Heidelberg, Paris, France, 23–26 May 2005.

Ryu, D. H., Kim, H., and Um, K. (2009). Reducing security vulnerabilities for critical infrastructure. In Mannan, M. S., editor, *Papers Presented at the 2007 and 2008 International Symposium of the Mary Kay O'Connor Process Safety Center and Papers Presented at the WCOGI 2007*, volume 22(6) of *Journal of Loss Prevention in the Process Industries*, pages 1020–1024. Elsevier Science Publishers.

Sabater, J. and Sierra, C. (2005). Review on Computational Trust and Reputation Models. In Robertson, D., editor, *Artificial Intelligence Review*, volume 24(1), pages 33–60. Kluwer Academic Publishers.

Schaberreiter, T., Aubert, J., and Khadraoui, D. (2011a). Critical infrastructure security modelling and RESCI-MONITOR: A risk based critical infrastructure model. In Cunningham, P., editor, *Proceedings of the 2011 IST-Africa Conference*, pages 1–8, Gaborone, Botswana, 11–13 May 2011. IEEE Computer Society.

Schaberreiter, T., Bouvry, P., Röning, J., and Khadraoui, D. (2013). A Bayesian Network Based Critical Infrastructure Risk Model. In Schütze, O., Coelho, C., Tantar, A.-A., Tantar, E., Bouvry, P., Del Moral, P., and Legrand, P., editors, *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II*, volume 175 of *Advances in Intelligent Systems and Computing*, pages 207–218. Springer Berlin Heidelberg.

Schaberreiter, T., Caldeira, F., Aubert, J., Monteiro, E., Khadraoui, D., and Simões, P. (2011b). Assurance and trust indicators to evaluate accuracy of on-line risk in critical infrastructures. In Bologna, S. and Wolthusen, S., editors, *Proceedings of the 6th International Workshop on Critical Information Infrastructures Security (CRITIS 2011)*, Lecture Notes in Computer Science, Lucerne, Switzerland, 8–9 September 2011. Springer Berlin Heidelberg.

Schembri, M. (2008). MIT: A software tool to facilitate CIP. In Adar, E., Hämmerli, B. M., and Luiijf, E., editors, *European CIIP Newsletter*, volume 4(2). [online] `http://www.irriis.org`.

Serrano, J., Serrat, J., and Galis, A. (2006). Ontology-Based Context Information Modelling for Managing Pervasive Applications. In Dini, P., Ayed, D., Dini, C., and Berbers, Y., editors, *Proceedings of the 2006 International Conference on Autonomic and Autonomous Systems (ICAS '06)*, pages 47–52, Silicon Valley, USA, 19–21 July 2006. IEEE Computer Society.

Serrano, J., Serrat, J., and Strassner, J. (2007). Ontology-Based Reasoning for Supporting Context-Aware Services on Autonomic Networks. In Thompson, J. and Andonovic, I., editors, *Proceedings of the 2007 IEEE International Conference on Communications (ICC '07)*, pages 2097–2102, Glasgow, Scotland, 24–28 June 2007. IEEE Computer Society.

SFinx (2013). Sfinx web site. [online] `http://https://www.sfinx.fr`.

Sheng, S., Yingkun, W., Yuyi, L., Yong, L., and Yu, J. (2011). Cyber attack impact on power system blackout. In IET, editor, *Proceedings of the IET Conference on Reliability of Transmission and Distribution Networks (RTDN 2011)*, IET Conference Publications, pages 1–5, London, United Kingdom, 22–24 November 2011. The Institution of Engineering and Technology.

Simões, P., Capodieci, P., Minicino, M., Panzieri, S., Castrucci, M., and Lev, L. (2010). An Alerting System for Interdependent Critical Infrastructures. In Demergis, J., editor, *Proceedings of the 9th European Conference on Information Warfare and Security (ECIW 2010)*, pages 275–283, Thessaloniki, Greece, 1–2 July 2010. Academic Publishing Limited.

Simões, P., Curado, M., Foglietta, C., Oliva, G., Panzieri, S., Minichino, M., Ciancamerla, E., Macone, D., Castrucci, M., Suraci, V., and Pauplin, O. (2010). *Refined interdependency metrics and indexes for risk prediction formulation – Final Version*. Panzieri, S., editor. MICIE Project Deliverable D3.1.2. European Commission FP7.

Simões, P., Curado, M., Panzieri, S., Oliva, G., Minichino, M., Ciancamerla, E., Macone, D., Castrucci, M., Suraci, V., and Pauplin, O. (2009). *Common Ontology and Risk Prediction Algorithms – Preliminary Version*. Panzieri, S., editor. MICIE Project Deliverable D3.2.1. European Commission FP7.

Smokeping (2012). Smokeping web site. [online] `http://oss.oetiker.ch/smokeping/`.

Sokolowski, J., Turnitsa, C., and Diallo, S. (2008). A Conceptual Modeling Method for Critical Infrastructure Modeling. In Znati, T. F. and Karatza, H. D., editors, *Proceedings of the 41st Annual Simulation Symposium (ANSS 2008)*, pages 203–211, Ottawa, Canada, 14–16 April 2008. IEEE Computer Society.

Spitz, S. and Tuchelmann, Y. (2009). A trust model considering the aspects of time. In Jusoff, K., Mahmoud, S. S., and Sivakumar, R., editors, *Proceedings of the 2009 International Conference on Computer and Electrical Engineering (ICCEE '09)*, volume 1, pages 550–554, Dubai, United Arab Emirates, 28–30 December 2009. IEEE Computer Society.

Strassner, J. (2003). *Policy-Based Network Management: Solutions for the Next Generation*. The Morgan Kaufmann Series in Networking. Morgan Kaufmann, 1 edition.

Strassner, J. (2004). Autonomic networking theory and practice. In Boutaba, R. and Kim, S.-B., editors, *Proceedings of the 10th IEEE/IFIP Network Operations and Management Symposium - Managing Next Generation Convergence Networks and Services (NOMS 2004)*, volume 1, page 927, Seoul, South Korea, 19–23 April 2004. IEEE Computer Society.

Swoop (2009). Swoop project web site. [online] `http://code.google.com/p/swoop/`.

TCG (2013). Trusted Platform Module (TPM) Specifications. Trusted Computing Group. [online] `http://www.trustedcomputinggroup.org/resources/tpm_main_specification`.

Teacy, W., Patel, J., Jennings, N., and Luck, M. (2006). Travos: Trust and reputation in the context of inaccurate information sources. In Ghidini, C., Giorgini, P., and van der Hoek, W., editors, *Special issue - Selection of papers presented at the 2nd European Agensts and Multi-Agent Systems Conference (EUMAS'04)*, volume 12(2) of *Autonomous Agents and Multi-Agent Systems*, pages 183–198. Kluwer Academic Publishers, Barcelona, Spain, 16–17 December 2004.

TISN (2011). Trusted Information Sharing Network (TISN) for Critical Infrastructure Resilience. [online] `http://www.tisn.gov.au`.

Twidle, K., Dulay, N., Lupu, E., and Sloman, M. (2009). Ponder2: A Policy System for Autonomous Pervasive Environments. In Calinescu, R., Liberal, F., Marin, M., Herrero, L., Turro, C., and Popescu, M., editors, *Proceedings of the Fifth*

International Conference on Autonomic and Autonomous Systems (ICAS '09), pages 330–335, Valencia, Spain, 20–25 April 2009. IEEE Computer Society.

Uszok, A., Bradshaw, J. M., Lott, J., Breedy, M., Bunch, L., Feltovich, P., Johnson, M., and Jun, H. (2008). New Developments in Ontology-Based Policy Management: Increasing the Practicality and Comprehensiveness of KAoS. In Agrawal, D., Al-Shaer, E., Kagal, L., and Lobo, J., editors, *Proceedings of the 9th IEEE Workshop on Policies for Distributed Systems and Networks (POLICY 2008)*, pages 145–152, Palisades, New Tork, USA, 2–4 June 2008. IEEE Computer Society.

Verissimo, P., Neves, N., and Correia, M. (2008a). The CRUTIAL reference critical information infrastructure architecture: a blueprint. In Gheorghe, A., editor, *International Journal of System of Systems Engineering*, volume 1/2, pages 78–95. Inderscience Publishers.

Verissimo, P., Neves, N., Correia, M., Deswarte, Y., Kalam, A., Bondavalli, A., and Daidone, A. (2008b). The CRUTIAL Architecture for Critical Information Infrastructures. In Lemos, R., Giandomenico, F., Gacek, C., Muccini, H., and Vieira, M., editors, *Architecting Dependable Systems V*, volume 5135 of *Lecture Notes in Computer Science*, pages 1–27. Springer Berlin Heidelberg.

W3C (2004). W3C, Web Ontology Language (OWL). [online] `http://www.w3.org/TR/owl-features`.

W3C (2009). W3C, Resource Description Framework (RDF). [online] `http://www.w3.org/RDF`.

Yavatkar, R., Pendarakis, D., and Guerin, R. (2000). RFC 2753: A Framework for Policy-based Admission Control. Technical report, IETF, USA.

Zahariadis, T., Ladis, E., Leligou, H., Trakadas, P., Tselikis, C., and Papadopoulos, K. (2008). Trust models for sensor networks. In Grgić, M. and Grgić, S., editors, *Proceedings of the ELMAR 50th International Symposium (ELMAR 2008)*, volume 2, pages 511–514, Zadar, Croatia, 10–12 September 2008. Croatian Society Electronics in Marine - ELMAR.

Coimbra, July 2013
Author's Edition.