

SWIN
BUR
NE

SWINBURNE
UNIVERSITY OF
TECHNOLOGY

An Ontological Approach Applied to Information Security and Trust

Artem Vorobiev and Nargiza Bekmamedova

Faculty of Information and Communication Technologies

Swinburne University of Technology

Melbourne, Australia





Outline

- Introduction to the problem
- Security ontologies
 - Introduction to ontologies
 - Security Asset-Vulnerability Ontology (SAVO)
 - Security Algorithm-Standard Ontology (SASO)
 - Security Function Ontology (SFO)
 - Security Attack Ontology (SAO)
 - Security Defence Ontology (SDO)
- Trust
- Example
- Summary
- Future research

Introduction to the Problem



- Distributed and complicated software applications
- System's security features
 - Added after functional requirements have been addressed
 - Not systematically designed into the system
 - Security "holes"
- New security attacks (e.g. multi-phased distributed attacks)
- Distributed detection and defenses
- Trusted common vocabulary comprehensible to both humans and software agents
- Attack correlation and anti-correlation

Introduction to Ontologies



- Specify many semantic relationships between various entities
- Share a common understanding of structured information among different parties such as humans or software agents which in turn, can be reasoned and analysed automatically
- Reusable and able to evolve over time
- Shared among different parties to solve interoperability problems

Security Asset-Vulnerability Ontology



- Main security ontology
- Specifies security information for different types of resources and environments; reusability and extensibility; mapping between high-level and low-level security requirements and capabilities
- Binds other security ontologies including SAO, SDO, SASO, SFO
- Attacks against peers (or hosts) affect their assets protected by defensive mechanisms
- Vulnerabilities exploited by threat agents in order to perform security attacks
- Assets evaluated by using the quantitative and qualitative analysis
- Main classes:
 - Threat, ThreatAgent, Vulnerability, Risk, Exposure, Defence, Attack, SecurityFunction, SecurityAlgorithmStandard, Supplier, Patch, Peer, etc

Security Algorithm-Standard Ontology



- Incorporates security algorithms, standards, concepts, credentials, objectives, assurance levels, etc.
- “Building blocks” for other ontologies including SFO
- Main classes:
 - SecurityAlgorithm, SecurityConcept, SecurityAssurance, SecurityCredential, SecurityObjective, etc
- Links system’s security objectives and high-level security policies with low-level technical countermeasures

Security Function Ontology



- Based on SCL (Security Characterisation Language) (Khan 2005) and Extended SCL (ESCL)
- Applied in conjunction with SASO to provide a vocabulary
- Uses SecurityAlgorithm and SecurityConcept from SASO
- Main classes:
 - **SecurityFunction** which contains [CryptoSupportFunc](#), [IdentificationAuthorisationFunc](#), [PrivacyFunc](#), [UserDataProtectionFunc](#), [TrustedChannelFunc](#), [SecurityAuditResourceUtilisationFunc](#), [Time](#), [Probability](#), etc

Security Attack Ontology



- Distributed components use SAO to interact with each other and share a common understanding of information about security attacks and defenses
- Evolves over time
- Main class is **Attack**:
 - **WSAttack** defines attacks on Web services
 - **P2PAttack** specifies security attacks against the peer-to-peer systems
 - **DoSAttack** describes various denial of service attacks
 - **SniffingAttack** characterises attacks on communication channels
 - **MultiPhasedDistributedAttack**, e.g. a Mitnick attack and its modifications, can be performed through the use of attacks from other classes

Security Defence Ontology



- Correlates with SAO
- Main class is **Defence**
- Defenses against
 - Attacks on Web services – [WSDefence](#)
 - On P2P systems - [P2PDefence](#)
 - Denial-of-service attacks – [DoSDefence](#)
 - Sniffing attacks – [SniffingDefence](#)
 - Multi-phased distributed attacks – [MultiPhasedDistributedDefence](#)

Trust



- Within the context of security ontologies, trust may include high-level of assurance, reliability and confidentiality
- Different levels: **technical level** supported by security algorithms and cryptography and **relationship level** that includes humans to interact and communicate
- **Technology trust** is supported by security technologies embedded in standardized interaction processes
 - Dimensions of technology trust may include security services which examine confidentiality, integrity, authentication, non-repudiation, and access controls
- **Relationship trust** exists among different parties that collectively assess transactions and is applied in the form of contracts, regular audit policies, quality standards, awareness training, etc
 - Dimensions of relationship trust may include competence trust, predictability trust, and goodwill trust that primarily focus on humans' behaviour

Example



- Mitnick attacks are possible to detect only by several coalition members
- Two hosts, Peer 1 (**P1**) and Peer 2 (**P2**), that are members of a coalition and which help each other to detect and mitigate various types of security attacks performed by an Attacker (**A**)
- **A** performs the Mitnick attack that exploits weakness of the TCP protocol design in making a TCP connection called the three-way handshake
- If **P1** detects the SYN/Flood attack, it will send a description of the attack to **P2**'s using security ontologies
- To ensure that new security ontology can be trusted, **P1** encrypts it using the AES algorithm that supports confidentiality. Further, **P1** signs the ontology and puts a timestamp to provide integrity, authentication, and non-repudiation using RSA and SHA-256 cryptographic algorithms

Summary



- Introduction to the problem
- Description of security ontologies: SAVO, SASO, SFO, SAO, SDO
- Trust: technical and relational level
- Example with a Mitnick attack: encrypted security ontology and trust appeal

Future Research



- Develop comprehensive security and trust ontologies that include social security and trust
- Apply ontologies to creating secure and trusted platforms (similar to IBM Trusted Virtual Domains) or to mobile environments, which change and evolve rather fast and in an unpredictable manner

References



- Khan, K 2005 'Security Characterisation and Compositional Analysis for Component-based Software Systems', PhD thesis, Monash University.
- Web Ontology Language (OWL): <<http://w3.org/TR/owl-features/>>
- Semantic Web Rule Language (SWRL):
<<http://www.w3.org/Submission/2004/03/>>



Questions?