Empowered by Innovation NEC

Trust and Reputation Management in Distributed Systems

Máster en Investigación en Informática

Facultad de Informática

Universidad Complutense de Madrid

Félix Gómez Mármol

NEC Laboratories Europe, Alemania

(felix.gomez-marmol@neclab.eu)

Madrid

29 de abril de 2013

Trust and Reputation Management in Distributed Systems

Agenda

- Introduction & General Overview
- Application Scenarios
- Generic Steps
- Security Threats
- Models Comparison
- TRMSim-WSN
- Conclusions

Trust and Reputation Management in Distributed Systems

INTRODUCTION & GENERAL OVERVIEW

Trust and Reputation Management in Distributed Systems

Introduction & General Overview

- Internet and WWW have changed our lives
- Despite their several advantages, there are also many security risks
- Traditional security solutions are very effective but not always applicable
- Trust and reputation management has been proposed as an accurate alternative
- Oneself can make his/her own opinion about how trustworthy or reputable another member of the community is
- Increases the probability of a successful transaction while reducing the opportunities of being defrauded

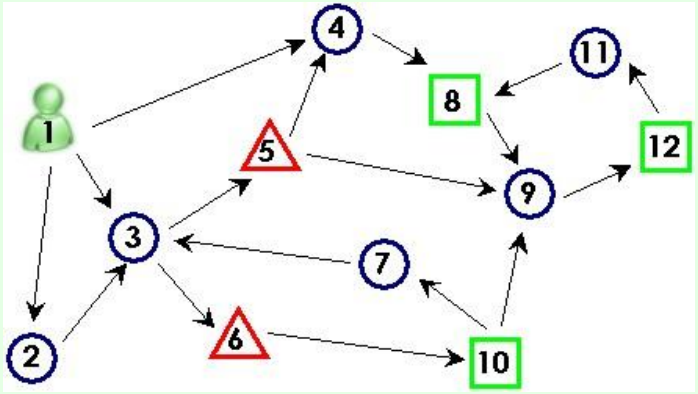
Trust and Reputation Management in Distributed Systems

APPLICATION SCENARIOS

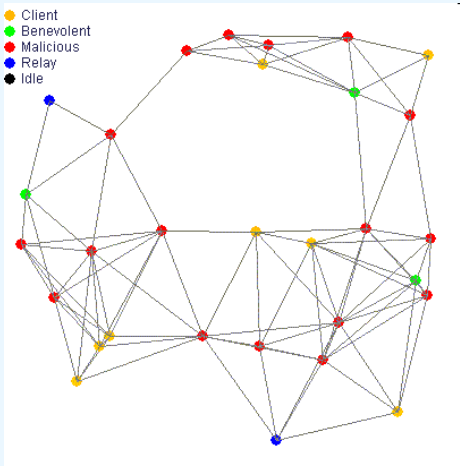
Trust and Reputation Management in Distributed Systems

Application Scenarios (I)

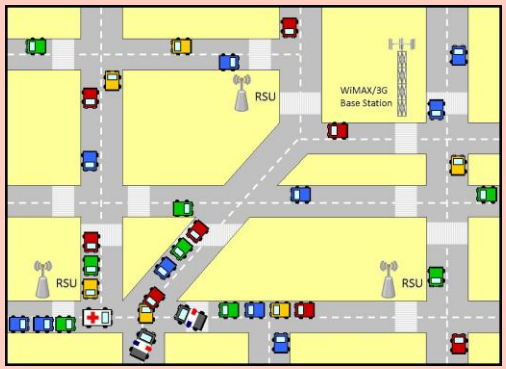
- P2P networks
  - Searching a generic service
  - Sharing a file
  - …



- Wireless sensor networks (WSN)
  - Measuring temperature
  - Measuring humidity
  - Measuring pressure
  - Detecting presence
  - …



- Vehicular Ad-hoc Networks (VANETs)
  - Emergency messages transmission
  - Traffic conditions
  - Weather conditions
  - Advertisements
  - …



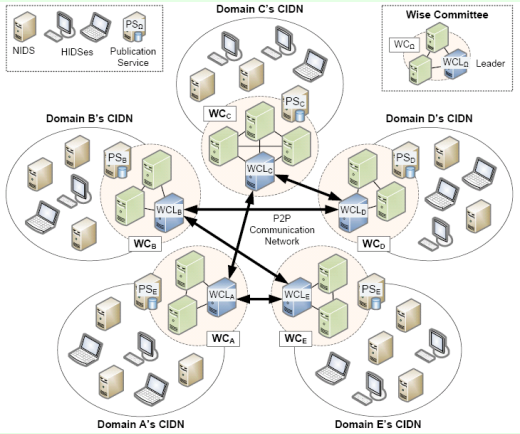
- Identity Management Systems
  - Sharing users' attributes
  - Identity federation management
  - …

Trust and Reputation Management in Distributed Systems

Application Scenarios (II)

Collaborative Intrusion Detection Networks (CIDN)

- Trust level on generated alarms
- Bootstrapping reputation for newcomers
- …



Cloud Computing

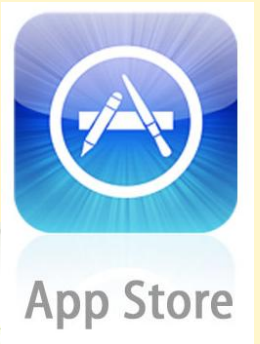
- Most trustworthy service selection
- Trust-based cloud services orchestration
- Tenants trustworthiness
- …

Internet of Things (IoT)

- Similar to wireless sensor networks
- Trustworthy information
- Trustworthy services
- …

Application Stores

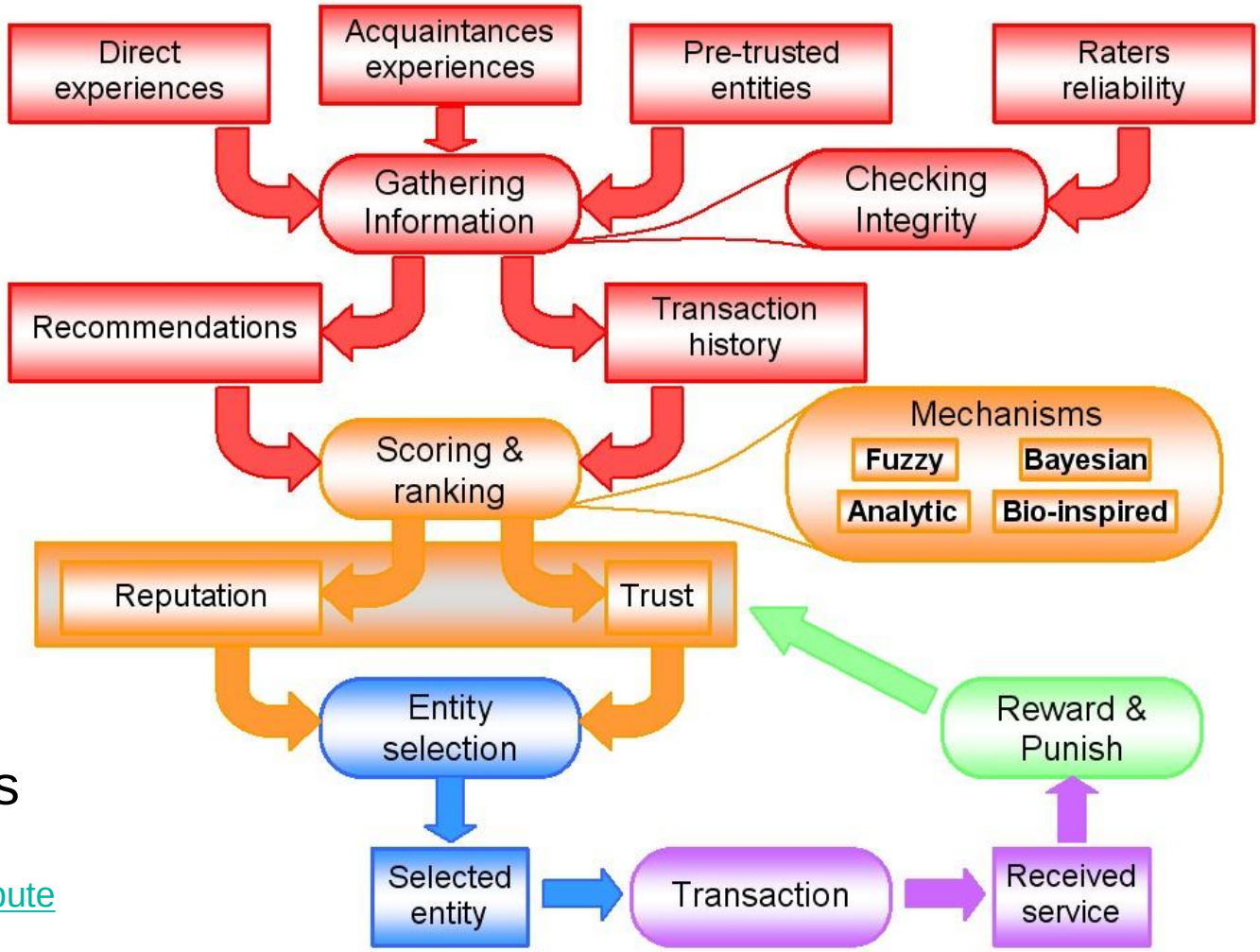
- Trustworthy applications
- Trustworthy developers
- …



GENERIC STEPS

Trust and Reputation Management in Distributed Systems

Generic Steps (I)



- Generic steps
- Generic interfaces
  - IETF Repute

https://tools.ietf.org/wg/repute

- Generic Data Structures
  - OASIS Open Reputation Management Systems (ORMS)

https://www.oasis-open.org/committees/orms

Trust and Reputation Management in Distributed Systems

Generic Steps (II)

10 design advices

1) Anonymous recommendations
2) Higher weight to more recent transactions
3) Recommendations subjectivity
4) Redemption of past malicious entities
5) Opportunity to participate for benevolent newcomers
6) Avoid abuse of a high achieved reputation
7) Benevolent nodes should have more opportunities than newcomers
8) Different trust/reputation scores for different services
9) Take into account bandwidth, energy consumption, scalability...
10) Consider the importance or associated risk of a transaction

Trust and Reputation Management in Distributed Systems

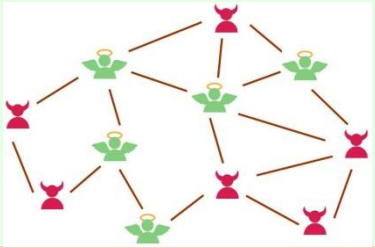
SECURITY THREATS

Trust and Reputation Management in Distributed Systems

Security Threats (I)

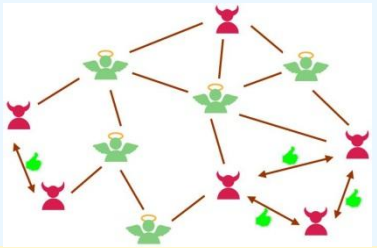
Individual malicious nodes

- Malicious nodes always provide a bad service
- Their reputation decreases and hence are not selected



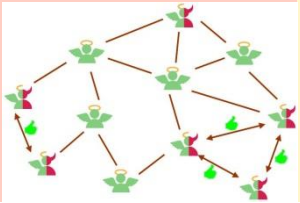
Malicious collectives

- Malicious nodes always provide a bad service
- Malicious nodes collude to unfairly provide high ratings about each other
- Their reputation decreases and hence are not selected
- ➢ Recommendations reliability should be handled



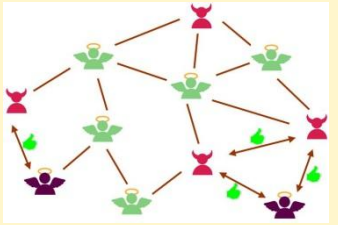
Malicious collectives with camouflage

- Malicious nodes provide a bad service p% of the times
- Malicious nodes collude to unfairly provide high ratings about each other
- Their reputation decreases and hence are not selected
- ➢ Recommendations reliability should be handled
- ➢ Store transactions history
- ➢ Not always considered as a threat
- ➢ Depends on behavioral pattern



Malicious spies

- Malicious nodes always provide a bad service
- Malicious nodes collude to unfairly provide high ratings about each other
- Malicious spies provide good services but positive recommendations about malicious nodes too
- Their reputation decreases and hence are not selected
- ➢ Recommendations reliability should be handled

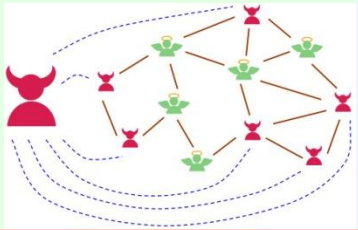


Trust and Reputation Management in Distributed Systems

Security Threats (II)

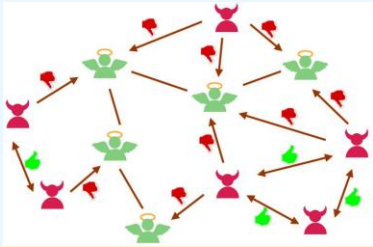
Sybil attack

- Attacker creates a disproportionate number of malicious nodes
- Malicious nodes always provide a bad service
- When reputation decreases, node leaves and enters again the network with a different identity
  - ➢ Associate some cost to new identities generation



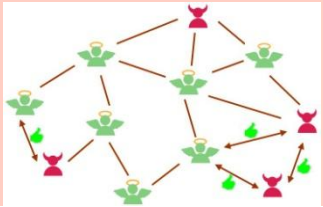
Driving down benevolent nodes reputation

- Malicious nodes always provide a bad service
- Malicious nodes collude to unfairly provide high ratings about each other
- They also provide bad recommendations about benevolent nodes
  - ➢ Recommendations reliability should be handled



Partially malicious collusion

- Malicious nodes always provide a bad service
- A node can be malicious for a given service but, benevolent for a different one
- Malicious nodes collude and rate positively each other
  - ➢ Different reputation values for different services



Malicious pre-trusted nodes

- Malicious nodes always provide a bad service
- Pre-trusted nodes provide positive recommendations about malicious nodes and negative ones about benevolent nodes
  - ➢ Dynamic selection of pre-trusted nodes

Trust and Reputation Management in Distributed Systems

Security Threats (III)

- Security threats taxonomy
  - Attack intent
  - Targets
  - Required knowledge
  - Cost
  - Algorithm dependence
  - Detectability

Security threats taxonomy

Security threats	Attacks dimensions					
	Attack intent	Target	Required knowledge	Cost	Algorithm dependence	Detectability
Individual malicious peers	Whole	Individual	Low	Low	Generic	High
Malicious collectives	Praise	Subset	Medium	Medium	Generic	Medium
Malicious collectives with camouflage	Praise	Subset	Medium	Medium	Generic	Low
Malicious spies	Praise	Subset	High	High	Generic	Low
Sybil attack	Whole	Subset	Low	Medium	Generic	Low
Man in the middle attack	Whole	Individual	Medium	Medium	Generic	Medium
Driving down the reputation of a reliable peer	Whole	All	High	High	Generic	Low
Partially malicious collectives	Whole	Subset	High	High	Generic	Low
Malicious pre-trusted peers	Whole	Subset	High	High	Specific	Low

Trust and Reputation Management in Distributed Systems

MODELS COMPARISON

Trust and Reputation Management in Distributed Systems

Models Comparison (I)

- Lack of mature bio-inspired and fuzzy approaches
- Lack of standard APIs and data structures
- Lack of security threats analysis
- Lack of generic testing tools

	Fuzzy	Bayesian	Bio-inspired	Analytic
Agent	PATROL-F AFRAS	MTrust BNBTM	AntRep	ATRM ATSN Sporas Regret
P2P	PATROL-F PTM	BNBTM PTM RRS	AntRep TDTM	DWTrust TPOD GroupRep EigenTrust
Ad-hoc		PTM RRS		ATRM
WSN		RFSN		ATRM DRBTS ATSN

Trust and Reputation Management in Distributed Systems

Models Comparison (II)

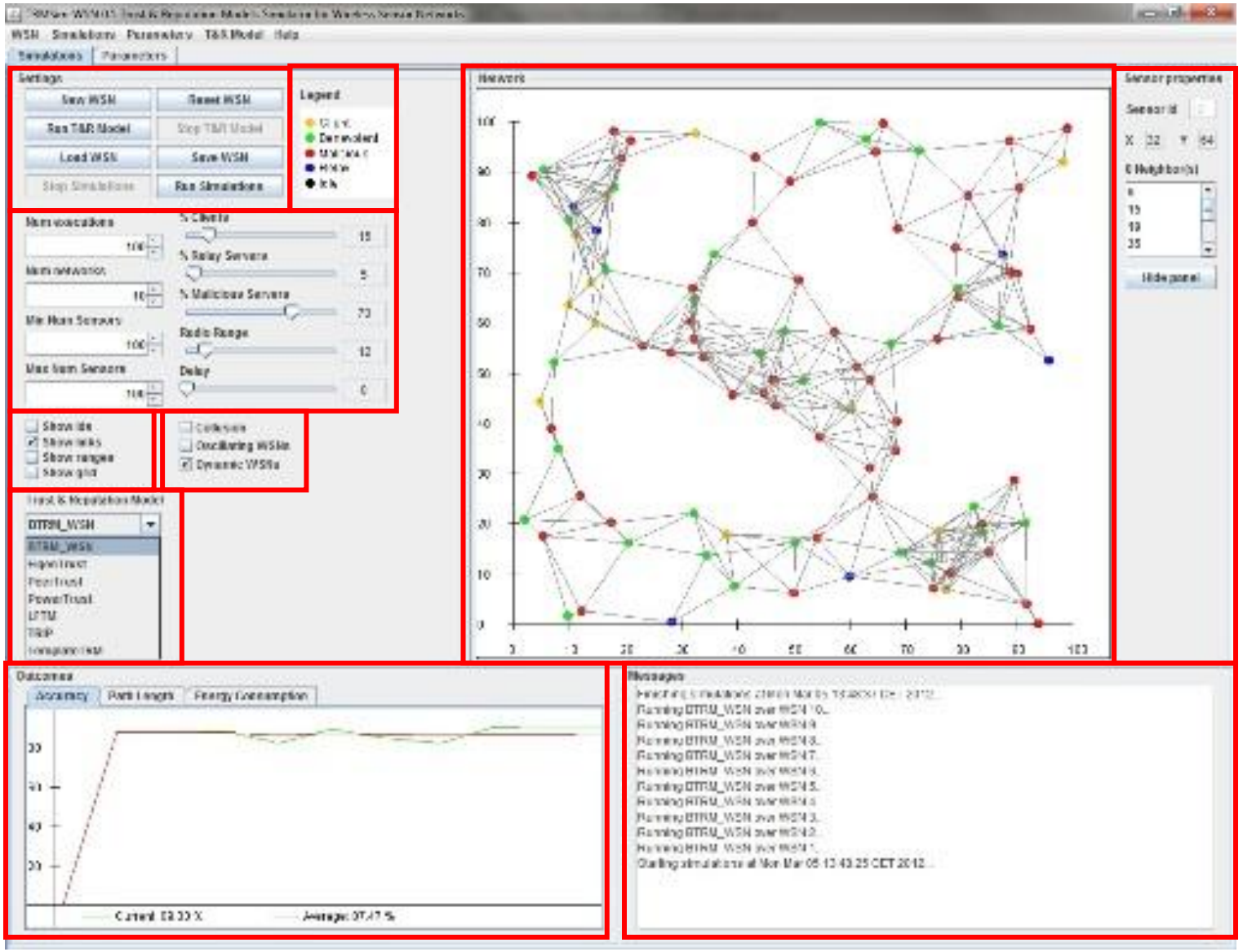
		Selected trust and/or reputation models			
		BTRM-WSN	**EigenTrust**	**PeerTrust**	**PowerTrust**
Trust and/or reputation model steps	**Gather information**	Ants explore the network, leaving pheromone traces	Each node builds the matrix $C = \{c_{ij}\}$	Each client collects other clients satisfactions to compute their credibility $Cr(v)$	Each server i collects r_{ji} and v_j from each interacted client j
	Score & Ranking	Every path is given a score $Q(S_k)$	Vector $\vec{t}_i^{(k)}$ is computed for each node i	Each client computes $T(u)$ for each reachable server u	Each server i computes v_i
	Entity selection	The path with highest quality is selected	Probabilistic selection with probability $\frac{t_i}{\sum_j t_j}$	Server u with $\max_u \{T(u)\}$ is selected	Server k with $\max_k \{v_k\}$ is selected
	Transaction	The client assesses her satisfaction with the received service	The client assesses her satisfaction with the received service	The client assesses her satisfaction with the received service	The client assesses her satisfaction with the received service
	Punish & reward	Pheromone evaporation	Not applied	Not applied	Not applied

Trust and Reputation Management in Distributed Systems

TRMSIM-WSN

Trust and Reputation Management in Distributed Systems

TRMSim-WSN



Trust and Reputation Management in Distributed Systems

DYNAMICALLY ADAPTABLE REPUTATION SYSTEMS

Trust and Reputation Management in Distributed Systems

Dynamically adaptable Reputation Systems (I)

The perfect reputation model does not exist

- There is not a computation engine suitable for all conditions
- Performance also depends on the scenario

Computation engine	Accuracy	User satisfaction	Adaptability	Behavior with malicious users	Behavior with malicious OP
Average	O	--	O	--	--
Weighted average	++	-	++	O	++
Preferences Weighted Average	++	++	+	O	++
User Weighted Average	++	++	O	++	++

-- poor - slightly poor O medium + slightly good ++ good

The reputation model performance depends on the applied scenario and current system conditions

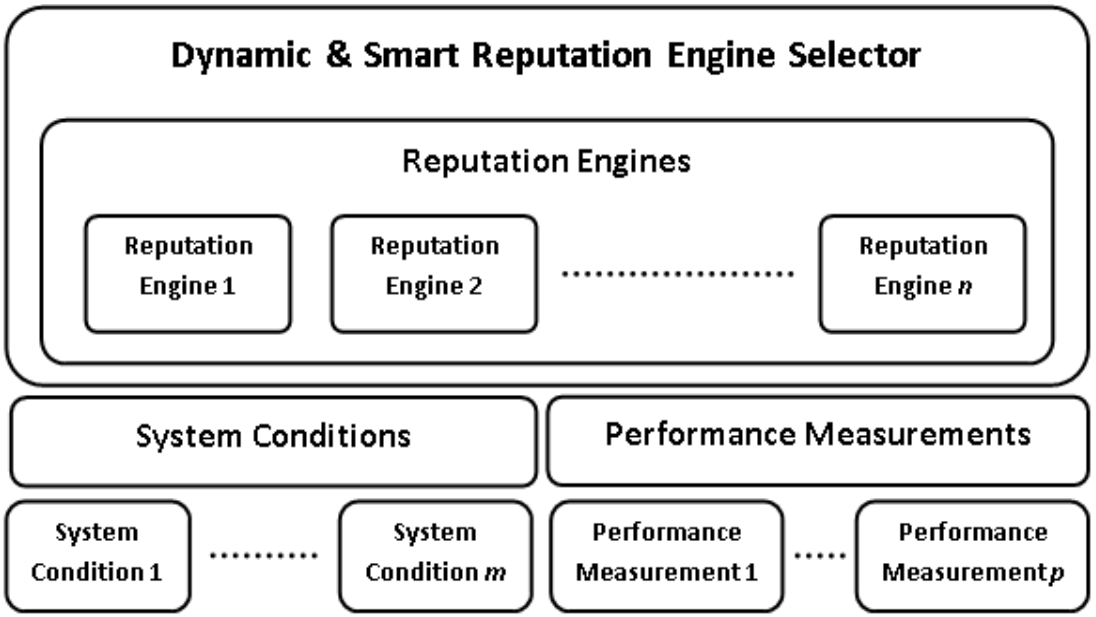
- **System conditions can vary along the time**

Dynamic & Smart Reputation Engine Selector (I)

Method to dynamically and smartly select the most appropriate reputation computation engine

- According to the current system conditions and the expected performance measurements

The system selects the most suitable reputation engine at each moment



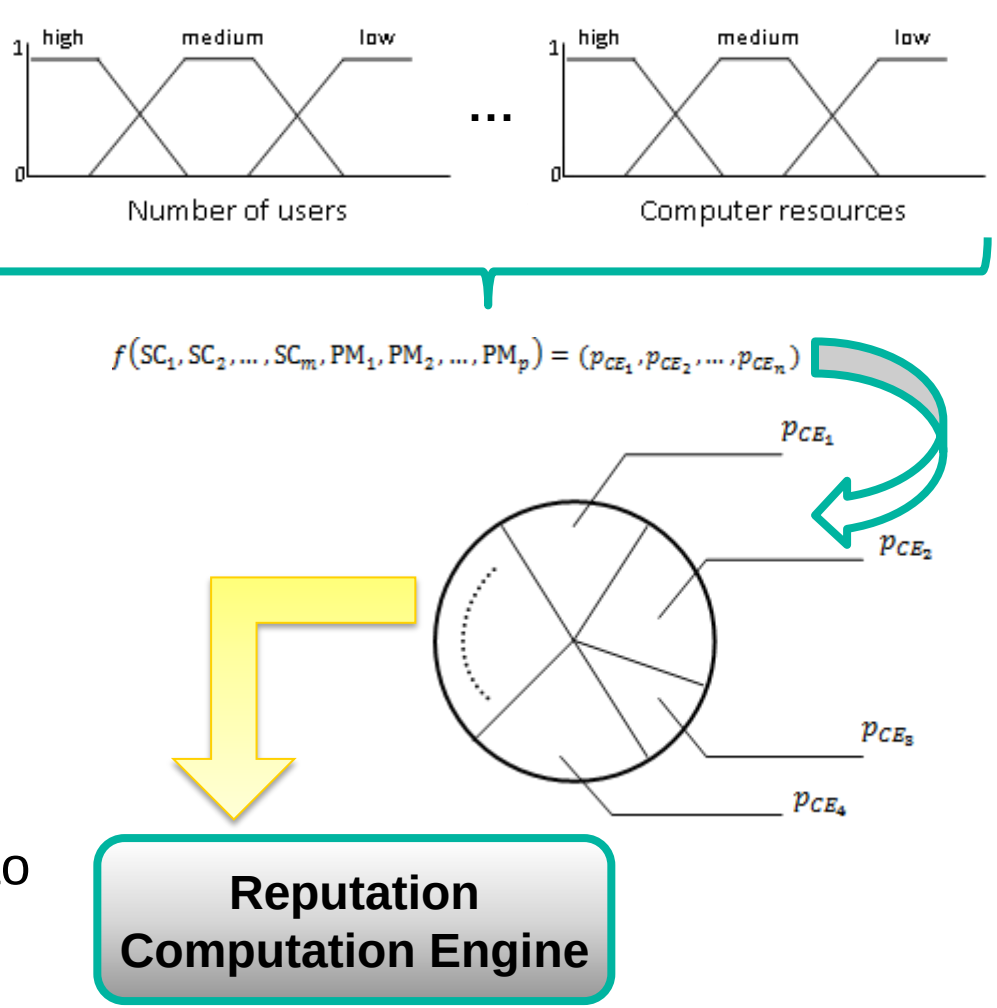
Dynamic & Smart Reputation Engine Selector (II)

- Instead of developing one single parametrizable model, several models are developed
- Each model has the best performance under certain well defined circumstances or conditions
- The system administrator indicates which performance metrics are more relevant at each moment
  - Model accuracy
  - Scalability
  - Robustness
  - Resilience against attacks

The dynamic & smart reputation engine selector chooses at each moment the reputation engine that better satisfies the performance metrics indicated by the system administrator, taking into account at the same time, the current system conditions (CPU usage, storage usage, etc)

Dynamic & Smart Reputation Engine Selector (III)

- The selector makes use of fuzzy sets to categorize current system conditions and performance metrics
  - `number_of_users=low`
  - `user_participation=medium`
  - etc
- Then, it determines the suitability of each computation engine as a value which gives the probability of use such reputation engine
- Finally, a probabilistic choice is performed to determine the Reputation Computation Engine to use



Dynamic & Smart Reputation Engine Selector (IV)

- Let ( ) be the current system conditions
- Let ( ) be the performance measurements
- Each and are represented as fuzzy sets
- Let ( ) be the performance metrics set by the administrator
- Let be the -th computation engine and ( ) the probability of selecting as the current computation engine
- Let ( ) be the performance metrics of under certain system conditions
- Then we have

*MSE: Mean Squared Error

Dynamic & Smart Reputation Engine Selector (V)

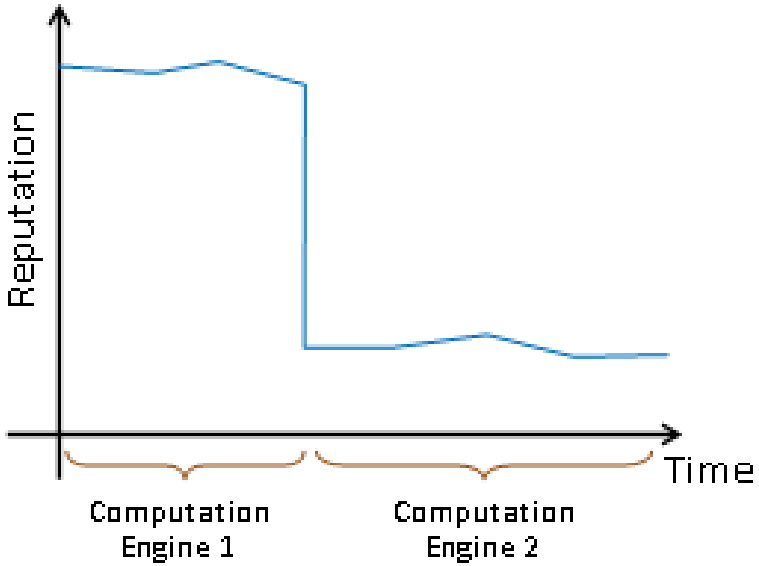
- The process would be as follows

- With
- Evaluating ( ) continuously would be very costly and resources consuming
- That is the reason why we use fuzzy sets to represent ( )

$$g_{CE_1}(\quad) \quad (PM_1^{CE_1},\dots,PM_p^{CE_1})$$
$$g_{CE_2}(\quad) \quad \dots \quad (PM_1^{CE_2},\dots,PM_p^{CE_2})$$
$$\vdots \qquad\qquad \vdots$$
$$g_{CE_n}(\quad) \quad (PM_1^{CE_n},\dots,PM_p^{CE_n})$$

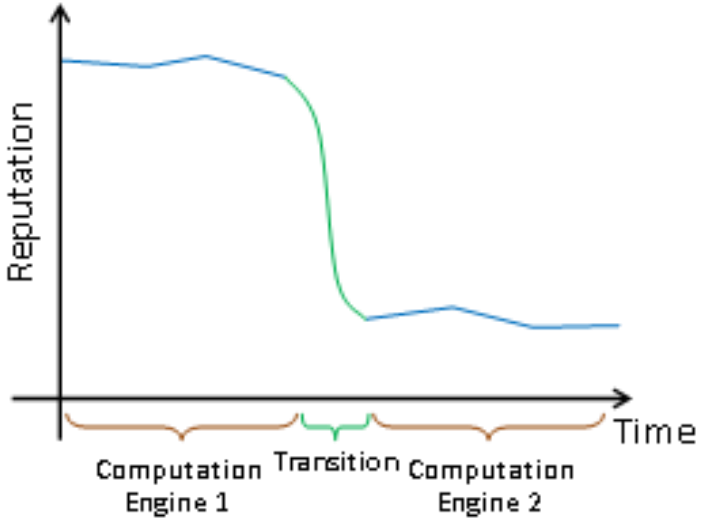
Smooth transition between different reputation computation engines (I)

- When switching to the selected best fitting computation engine it might happen that the computed reputation scores differ too much from the ones obtained with the previous computation engine
- We want to avoid an abrupt change in the computed reputation score

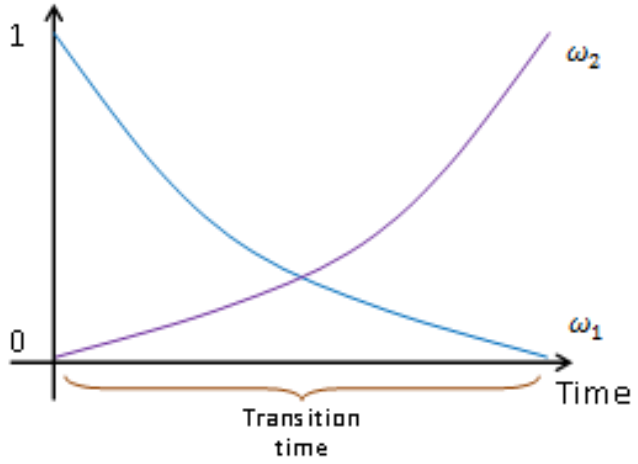


Smooth transition between different reputation computation engines (II)

- We propose a smooth transition between the old computation engine and the new one
  - For a while, both reputation values are taken into account
  - To do so, we weight the reputation scores given by both computation engines



  - Weights decreases during the transition time as increases, fulfilling that

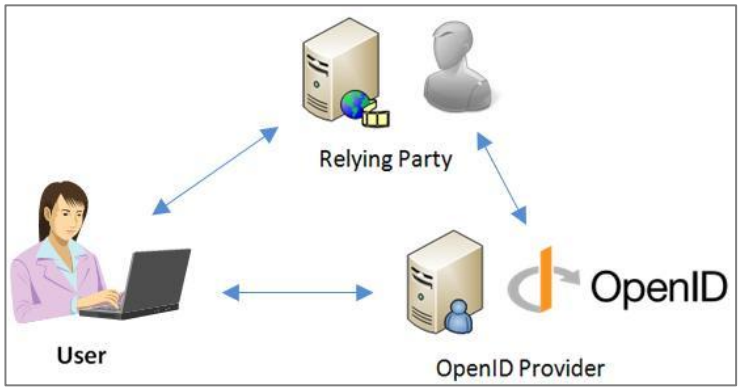


Trust and Reputation Management in Distributed Systems

Dynamically adaptable Reputation Systems (IX)

Integration tests within identity management systems

- Developed four different trust and reputation models
- Several simulations performed to analyze the behavior within IdM systems
- Analyzed these four models according to different **system conditions** and **performance measurements**



Computation engine	Number of Users	Number of OPs	User Participation	Network Resources	Computer resources
Average	*	*	*	*	*
Weighted average	**	*	**	*	**
Preferences Weighted Average	***	**	**	***	***
User Weighted Average	****	***	****	***	****

* few ** considerable *** many **** a vast amount

Advantages and Limitations

Advantages

- Flexible mechanism to select the most appropriate trust and reputation model to apply at each moment considering both the system conditions and the performance measurements
- Resources consumption adaptation and optimization by applying the most suitable trust and reputation model at each moment
- Improvement and optimization of the performance of the trust and reputation management model applied at each moment

Limitations

- Reputation computation engines should be developed and analyzed beforehand in order to determine under which conditions they provide the best outcomes for each of the desired performance measurements.

Trust and Reputation Management in Distributed Systems

CONCLUSIONS

Trust and Reputation Management in Distributed Systems
Conclusions

- Current challenges
  - Many authors focus on the “scoring and ranking” step, neglecting the other ones
  - Reputation bootstrapping is also a commonly obviated issue
  - Security threats and design recommendations are also usually not considered
  - Weak support from the standardization community
    - OASIS ORMS (Open Reputation Management Systems)
    - IETF Reputation Services (Repute)

Trust and Reputation Management in Distributed Systems

BIBLIOGRAPHY

Trust and Reputation Management in Distributed Systems
Bibliography (I)

- OASIS ORMS (Open Reputation Management Systems) www.oasis-open.org/committees/orms
- Kevin Hoffman, David Zage, Cristina Nita-Rotaru, ”A survey of attack and defense techniques for reputation systems”, ACM Computing Surveys, 42 (1), 2009
- Audun Josang, Roslan Ismail, Colin Boyd, “A survey of trust and reputation systems for online service provision”, *Decision Support Systems*, 43 (2), 618-644, 2007
- M. Carmen Fernandez-Gago, Rodrigo Roman, Javier Lopez, “A survey on the applicability of trust management systems for wireless sensor networks”, In *International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing*, pages 25-30, 2007
- Yan Sun, Zhu Han, K.J.R. Liu, “Defense of trust management vulnerabilities in distributed networks”, *IEEE Communications Magazine*, 46 (2), 112-119, 2008
- Yan Sun Yafei Yang, “Trust Establishment in Distributed Networks: Analysis and Modeling”, In *Proceedings of the IEEE International Conference on Communications (IEEE ICC 2007)*, Glasgow, Scotland, 2007
- Shyong K. Lam, John Riedl, “Shilling recommender systems for fun and profit”, In *Proceedings of the 13th international conference on World Wide Web*, pages 393-402, New York, 2004
- Sepandar D. Kamvar, Mario T. Schlosser, Héctor Garcia-Molina, “The EigenTrust Algorithm for Reputation Management in P2P Networks”, In *Proc. of the International World Wide Web Conference (WWW)*, Budapest, Hungary, 2003

Trust and Reputation Management in Distributed Systems
Bibliography (II)

- Li Xiong, Ling Liu, “PeerTrust: Supporting Reputation-Based Trust in Peer-to-Peer Communities”, *IEEE Transactions on Knowledge and Data Engineering*, 16 (7), 843-857, 2004
- Runfang Zhou, Kai Hwang, “PowerTrust: A Robust and Scalable Reputation System for Trusted Peer-to-Peer Computing”, *Transactions on Parallel and Distributed Systems*, 18 (4), 460-473, 2007
- Félix Gómez Mármol, Gregorio Martínez Pérez, “Security Threats Scenarios in Trust and Reputation Models for Distributed Systems”, Computers & Security, 28 (7), 545-556, 2009
- Félix Gómez Mármol, Gregorio Martínez Pérez, “Towards Pre-Standardization of Trust and Reputation Models for Distributed and Heterogeneous Systems”, Computer Standards & Interfaces, 32 (4), 185-196, 2010
- Félix Gómez Mármol, Gregorio Martínez Pérez, “Trust and Reputation Models Comparison”, Emerald Internet Research, 21 (2), 138-153, 2011
- Félix Gómez Mármol, Gregorio Martínez Pérez, “TRMSim-WSN, Trust and Reputation Models Simulator for Wireless Sensor Networks”, IEEE International Conference on Communications (IEEE ICC 2009), Dresden, Germany, 2009
- IETF Reputation Services (Repute) http://www.ietf.org/proceedings/81/repute.html