A Trust-Based Reputation Management System

A thesis submitted to the
University of Dublin, Trinity College,
in fulfilment of the requirements for the degree of
Doctor of Philosophy (Computer Science).

Elizabeth L. Gray

Distributed Systems Group,
Department of Computer Science,
Trinity College, University of Dublin

April 2006

DECLARATION

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this or any other University, and that, unless otherwise stated, it is entirely my own work.

_____

Elizabeth L. Gray,

28th April 2006

PERMISSION TO LEND AND/OR COPY

I, the undersigned, agree that the Trinity College Library may lend and/or copy this thesis upon request.

_____

Elizabeth L. Gray,

28th April 2006

ACKNOWLEDGEMENTS

Sunset is an angel weeping, holding out a bloody sword.
No matter how I squint I cannot make out what it's pointing toward…
Sometimes the best map will not guide you, you can't see what's round the bend.
Sometimes the road leads through dark places, sometimes the darkness is your friend.
Today these eyes scan bleached-out land, for the coming of the outbound stage, pacing the cage.
~ Jimmy Buffett

Many times over the past several years, I caught myself 'pacing the cage', wondering where the research path would lead. During that time, many people proffered support and guidance, and they deserve so much recognition and thanks.

First, and foremost, I would like to thank my supervisor, Prof. Vinny Cahill, who provided an excellent map to guide me, both in terms of research and in terms of life, as well as so many words of encouragement, faith, kindness, and patience when I could not see what was 'round the bend'.

Thanks also to the members of various research groups for providing opportunities to collaborate and learn. In particular, special thanks to Christian Jensen for bringing me into the SECURE consortium. I am also so very appreciative of the members of the DSG research group, especially Stefan Weber, Anthony Harrington, Peter Barron, Yong Chen, Barbara Hughes, Ray Cunningham, Cormac Driver, Gregor Gaertner, John Keeney, René Meier, Andronikos Nedos, Jean-Marc Seigneur, and Kulpreet Singh. Much appreciation is also due Alan Mullally and Denise Leahy, of the Dept. of Information Systems, for creating an environment in which I could teach and learn. Thanks also to Yong Wang of the Beijing Institute of Technology. Furthermore, to Audun Jøsang of the Queensland University of Technology, thank you for sharing your vast experience in the field of reputation management.

Thank you to the many friends who provided advice, support, caring, and willingness to listen to ramblings about trust-based decision-making models. Above all, this one's for the girls: Treasa Ní Mhíocaín, Judith Murphy, Keelin Murphy, Elizabeth Daly, and Ivana Dusparic. Each of you is a guardian angel, and even when we lost the map entirely, somehow you managed to make sure I got to the end of the journey in one piece and with so many incredible memories. Also, Geraldine McNamara and Piers Gardiner, thank you for continually opening your hearts and home to me. Gratitude and appreciation especially to Niall Rea, who stayed fast during the lowest points, pacing with me, step by step, always shining a light around corners and bringing tranquillity.

Finally, I am so grateful for the blessing of family. I am Alice's granddaughter, the spitting image of my father, and when the day is done, my mother's still my biggest fan…it's all a part of me and that's who I am. Thank you to my grandparents, who passed on the knowledge that wisdom and joy can be found in any culture, anywhere in the world; and to my extended family, who confirm that home is a

valuable place in its own right, and that the ideal of The Farm will always exist. Most significantly, to my parents, Edward and Suzanne Gray, nothing I do would be possible without you. Mom, you have given us so much creativity and lateral thinking abilities. Daddy, you have given us an engineer's intellect combined with an incomparable work ethic. You both constantly provide inspiration, love, and encouragement – thank you for teaching your children that we could achieve any dream through hard work and a positive attitude. No matter how many times I became lost, the angels and the best maps have always pointed home to you. To my sister and best friend, Rebecca Gray Bavuso, thank you for listening, especially in the middle of the night, and for consistently providing calm, grace, and beauty to our family. (And to her husband, Matthew Bavuso, thank you for always letting Becca take care of us all.) Finally, so much love to my nephews, Miles Ray and Grayson Alexander. When I look back over the entirety of these years away, your arrivals into the world will always be the two brightest highlights.

Trust one who has gone through it.

~ Virgil

SUMMARY

Since its inception in the early 1990s, e-commerce in consumer-to-consumer (C2C) markets has achieved great success, with significant projected growth. For example, the Internet auction provider, eBay, has established itself as the largest global player in this market, with $34.2 billion worth of merchandise being auctioned in 2004 and 135 million registered users in 32 markets worldwide. The C2C domain, analogous to its conventional physical marketplace equivalent, is built on trust. Buyers send payments to complete strangers from whom they have purchased goods and trust that the goods will be sent in return. Sellers trust buyers to make good on their payments. All users risk loss, both financial and of their time. Users establish reputations about their trustworthiness through an integrated feedback collection and distribution system, i.e., a reputation management system. Thus, an online marketplace approximates its traditional predecessors as a system in which the human concepts of trust, risk, and reputation are critical to performance.

The apparent benefits of interacting in such a strongly-networked global market are accompanied by innovative adaptations of traditional hazards. The Internet, while connecting disparate user groups to increase transaction potential and shared knowledge about the marketplace, also permits user anonymity and transactional intangibility, which can lead to fraud, theft, and collusion. Reputation management systems attempt to limit incorrect behaviour and to assist decision making by providing records of feedback about interactions, called recommendations, for each community participant. These systems are not without their own limitations. First, commercial reputation management systems typically promote usability over accurate evidentiary analysis, meaning that data which could be extremely useful to decision-making is disregarded by the evidence collection mechanism so that ease-of-use is maintained for community members when they are voluntarily providing feedback. This first issue leads directly to the second, which is inaccurate evidentiary analysis with regard to contextual relevance in terms of user role, timeliness of evidence, and environmental context. In this regard, trustworthiness is usually linked solely to the overall number of positive recommendations about a user, regardless of the interaction context being considered. Third, the dynamics of user interactions are not addressed, and interaction dynamics in such an evidence-rich environment are difficult, if not impossible, for an average user to manually detect. Without the ability to analyse interaction dynamics, the fourth and fifth issues arise, namely that the analysis of whether or not a user provides useful and accurate recommendations about another user or whether or not a group of users are colluding with malicious intent are both difficult to observe. Sixth, risk is not explicitly calculated by the reputation system, and may not be assessed by the user at all. Seventh, and finally, a reputation is often no more than an overall summary of a collection of thousands of individual recommendations rather than an explicit portrayal of the trust and risk involved in a context-specific interaction.

This thesis describes a trust-based reputation management system (RMS) that addresses each of the above issues. The system resolves the ease-of-use versus accuracy problem by maintaining usability

but with enhanced collection and analysis of evidence with regard to domain-specific behaviour. Furthermore, the system provides increased accuracy of evidentiary analysis with regard to context by assessing evidence in terms of role, timeliness, and environment. Interaction dynamics are also considered in the system's decision-making process, thus providing for the ability to limit exposure to risk from unreliable recommendations as well as the ability to assess the likelihood of colluding behaviour. The risk of an interaction resulting in malicious behaviour is explicitly analysed and stated to the user. Finally, the reputation summary is replaced by the explicit assessment of the trust and risk involved in interacting with another user, providing a security decision as advice to a user on whether or not to engage in an interaction.

The RMS builds on the work of the SECURE (Secure Environments for Collaboration among Ubiquitous Roaming Entities) project. Grounded on a formal model, the SECURE trust-based decision-making framework applies trust and risk to evidence in a manner comparable to the human decision-making process. We use the SECURE model as a basis with which to design our own application-specific mechanisms for reputation management in Internet auctions, and these mechanisms provide for the observation of domain-specific behaviour such as fraud and theft, assessment of contextual relevance, and analysis of risk in financial terms that is made explicit to the end user. Additionally, in the reputation management for Internet auctions application domain, SECURE is deficient in analysing the dynamic aspects of marketplace networks, and therefore we design additional techniques for interaction management. These techniques underlie an extension to the SECURE framework that includes methods for the weighting of recommendations based on the application of recommendation weighting policy to trustworthy recommendation paths within the graph of marketplace participants; and the identification of colluding behaviour between users within the marketplace community, by assessing interaction dynamics between users over time.

Our evaluation of the RMS shows that it reduces complexity, increases accuracy, and maintains usability of reputation management for Internet auction users. It validates that the RMS, in its observation and identification of normal and abnormal domain-specific behaviour, reduces complexity by providing accurate decision-making advice to users. Furthermore, the evaluation confirms that the analysis of context in terms of role, time, and environmental factors can further reduce complexity in the decision-making process while maintaining usability. Additionally, the evaluation demonstrates that recommendation weighting can protect a user against the potential unreliability of recommended evidence. Finally, the evaluation establishes that a reputation management system based on a computational trust-based decision-making model can counter the issues in existing commercial reputation management systems and provide increased benefit to users interacting in the Internet auction domain.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

The way to gain a good reputation is to endeavour to be what you desire to appear.

~ Socrates

It is estimated that in the next five years e-commerce will be marked by innovations that will make online shopping easier and more engaging, thus boosting U.S. online retail sales from $172 billion in 2005 to $329 billion in 2010 (Forrester Research 2005). Within the e-commerce domain, consumer-to-consumer (C2C) offerings have achieved great success, as evidenced by the virtual marketplace eBay Inc.'s capture of 24% of all e-commerce in the U.S. in 2004, up from 16% in 2000 (eBay Inc. 2005).

The reasons for the growth of the virtual marketplace are manifold. Users easily make the paradigm shift from physical to virtual marketplace notions, where a C2C application functions as the digital equivalent of a traditional marketplace through which community members may offer goods and services for sale or auction and browse through and purchase available goods and services through a web interface. Other benefits of moving these processes online include increased trading potential amongst a much larger consumer base than exists in traditional marketplaces, which are bound by geographic and temporal constraints, and a highly networked environment in which information about product offerings and community members can be more efficiently shared.

In a 2005 survey of Internet users, however, 42% said concerns about online attacks have affected their online shopping behaviour, leading them to be more cautious about where they shop online and to buy fewer items than they normally would (Perez 2005). Moreover, the results of a recent fraud survey carried out by CyberSource Corp. estimate e-commerce fraud losses at having increased to more than $2.8 billion for 2005, an 8% increase over the year before, and the survey notes that 'merchants used to be able to just throw people at this problem. But there's an inherent limitation to that solution. What merchants need today is greater efficiency, greater intelligence, even technology breakthroughs. The bad guys have put the larger merchants in a challenging situation' (CyberSource Corporation 2005).

These statistics reveal that users understand that the apparent benefits of interacting in such a strongly-networked global market are accompanied by innovative adaptations of traditional hazards such as fraud, theft, and collusion, which may be amplified in an environment where users are shielded by the anonymity of digital network connections as well as data overload. When deciding to interact in a virtual marketplace, users act on trust, just as they do in the physical world. Buyers send payments to complete strangers from whom they have purchased goods and trust that the goods will be sent in

return. Sellers trust buyers to make good on their payments. All users risk loss, both financial and of their time, e.g., products may be misrepresented more easily in an online environment, users may collude to defraud other community members, and payments and shipments between strangers may result in increased theft (Anderson 2005). Thus, a virtual marketplace approximates its traditional predecessors as a system in which the human notions of trust and risk in decision-making are critical to continued performance in the face of domain-specific threats.

1.1 Reputation Management

Determining whom to trust in a virtual marketplace is therefore a core issue, and users must calculate, either consciously or sub-consciously, the trustworthiness of a buyer or seller as well as the overall risk of participating in a transaction. In order to foster trustworthy behaviour and to help users make decisions about interacting with others, most virtual marketplaces incorporate some form of reputation management. Reputation management systems are proposed as a method to ensure trustworthy interactions and decreased risk in this domain, where users' reputations are formed as recommendations about them accrue from gathered historical interaction data, e.g., a reputation for consistently providing high quality goods or a reputation for not always paying on time. After each interaction, a user may evaluate the outcome of an interaction by leaving feedback, i.e., a recommendation, about one's interaction partner. The C2C application's reputation management system thus serves as a central repository of historical interaction and reputation evidence which can be used by members to make decisions about the trustworthiness and risk involved in future transactions. In this way, reputation management systems attempt to limit exposure to threats by recording and publicising each user's reputed behaviour, whether correct or incorrect.

1.2 Issues with Reputation Management Systems

Reputation management in the commercial virtual marketplace space, however, is also accompanied by several outstanding issues, namely, a lack of precision in the recommendation process; deficiencies in evidentiary analysis due to a disregard of contextual relevance, interaction dynamics, and risk; and the equating of reputation with reputation summary information that conveys very little decision-relevant information to a user trying to determine whether or not to engage in a given interaction.

First, commercial reputation systems typically promote usability over accuracy in the evidence feedback loop. Because providing post-interaction feedback is voluntary in most reputation management systems, the only incentive to record a recommendation is user goodwill. In this type of system, the feedback recording process must therefore be kept as simple as possible so that users are incentivized to participate rather than to free-ride on the goodwill of others. For example, recent research (Resnick and Zeckhauser 2002) found that 60.7% of buyers and 51.7% of sellers in the eBay community leave feedback, i.e., a record of whether the interaction outcome was positive or negative

together with a short qualitative statement, after completing transactions, suggesting that the eBay feedback model is simple and usable enough for members to operate. While maintaining usability, however, it is at the cost of not gathering observed and recommended evidence that might more accurately demonstrate relevant patterns of user behaviour.

The first issue leads directly to the second, i.e., inaccurate evidentiary analysis with regard to contextual relevance. Although in a C2C reputation management system it is possible to record much information about an interaction in addition to whether or not the outcome was simply positive or negative, e.g., the date and time a transaction took place, the roles of the users in a transaction, the details of the product or service being exchanged, etc., this context information is typically disregarded in the evidence evaluation process. In the absence of contextual relevance being incorporated into a system's reputation evaluation process, a user must base a decision about a potential interaction upon evidence ranging from no information about another user to incalculable amounts of information that cannot be manually assessed. In this regard, trustworthiness is linked solely to the overall number of positive recommendations about a user, regardless of information that is more contextually relevant to a decision at hand.

In the C2C domain, it is also possible to observe information about relationships between sets of users, i.e., the dynamics of their interactions, such as whether or not a user provides useful and accurate recommendations about another user or whether or not a user employs a specific interaction strategy, such as collusion with malicious intent, when interacting with another user. As in the case of determining which information is more contextually relevant to a decision at hand, it is impossible for a user to manually assess an overwhelming number of system observations about interaction dynamics to determine which information is relevant when considering whether or not to interact with a given set of users.

Furthermore, risk is not explicitly calculated by these reputation systems, and may not be assessed by the user at all. In the C2C domain, risk is largely attached to the value of an item or service being sold or purchased, and the risk is highly variable. For example, in the eBay marketplace, the value of products offered for sale ranges from the nearly insignificant, e.g., a pair of sunglasses being sold for one cent, through many value ranges to very high value-based risk, e.g., the Gulfstream II jet sold for $4.9 million. In this environment, risk factors that should be explicitly analysed when making a decision include, but are not limited to, the rate of virtual marketplace fraud, the risk attached to interacting with a particular user based on a user's direct and indirect experiences of interacting with that particular entity, the value of the transaction being considered, and the context in which the potential interaction is taking place.

Finally, a reputation is often no more than an overall summary of a collection of thousands of individual recommendations, of which only some are relevant, with regard to context and interaction dynamics, and with no risk being explicitly assessed. A more useful output of a reputation management system would be advice given to a user based on the analysis of trustworthiness, context,

interaction dynamics, and risk of interacting with another user or set of users in a particular transaction.

Many of the above issues arise from the current state of practice in commercial reputation management, and some methods proposed in academic research can be applied to resolve the concerns. These methods are discussed in Chapter 2. None of the reputation management systems proposed in academia, however, provides a cohesive model that unifies potential solutions to each of the issues noted above.

1.3 Aims and Objectives

The general objective of this thesis is the design of a reputation management system for virtual marketplace applications, particularly those that provide Internet auction services. The reputation management system should reduce complexity, increase accuracy, and yet maintain usability for a user wishing to determine whether or not to participate in a new interaction. Specifically, this thesis describes a reputation management system that addresses each of the issues highlighted above, i.e., maintaining the usability of evidence collection while increasing accuracy with regard to collecting evidence necessary to identify domain-specific behaviour types, and thus to limit the risk of exposure to untrustworthy entities; increasing the accuracy of evidentiary analysis with regard to contextual relevance; providing the ability to limit exposure to risk of unreliable recommended evidence; providing the ability to observe behaviour based on the assessment of interaction dynamics, and thus to limit the exposure to risk from colluding users; making risk explicit to the decision-maker; and, finally, providing advice to a decision-maker as to how to interact.

1.4 SECURE

A reputation management system should be based on a model that unifies the characteristics and properties of human trust, because such a reputation management system aims to provide computational decision-making in a manner analogous to human trust-based decision-making. Our trust-based reputation management system (RMS) builds on the work of the Secure Environments for Collaboration among Ubiquitous Roaming Entities (SECURE) project. Grounded on a formal model, the SECURE trust-based decision-making framework applies trust and risk assessment to evidence in a manner comparable to the human decision-making process. We use the SECURE model as a basis with which to design our own application-specific mechanisms for reputation management in Internet auctions, and therefore we are able to address several of the issues described above. First, we can incorporate the means to derive more accurate results from feedback by providing an evidence collection method based on properties and characteristics of trust from research, while maintaining usability. Granularity of feedback for a given application domain can be optimised according to specific parameters rather than left open to user subjectivity when only coarse-grained rating feedback

collection mechanisms are provided. Moreover, evidence collection can be tailored to extract information that is most suitable to a particular domain, and thus gathered evidence can be used to identify domain-specific threats. Second, evidentiary analysis according to contextual relevance may be incorporated while maintaining usability. Third, risk assessment is made explicit in the SECURE decision-making framework, and both actor- and context- specific risk is addressed. Finally, SECURE provides a security, or access control, decision as the output of the decision-making process. In place of a general reputation summary, this decision can be used as advice regarding how to proceed by a user as it is based on more accurate evidence that is relevant to a decision at hand in terms of trustworthiness, context, and risk.

However, SECURE provides no viable means for assessing interaction dynamics such as recommendation integrity and user group interaction profiles. Thus, the SECURE model is a good tool for assessing user-centric behaviour in terms of trust and risk, but provides no methods for capturing and analyzing the dynamic behaviour of interactions between users. In this regard, this thesis proposes an extension to the SECURE framework whereby the dynamic nature of interaction relationships is observed and assessed with the aim of better evaluating the likely outcomes of participating in a given interaction within a given user group. These techniques underlie an extension to the SECURE framework that includes methods for the weighting of recommendations based on the application of recommendation weighting policy to trustworthy recommendation paths within the graph linking marketplace participants; and the identification of anomalous user behaviour, e.g., potential colluding users by analysing evidence about user group interactions over time.

1.5 Contribution of this Thesis

This thesis describes the design and evaluation of a trust-based reputation management system (RMS) for virtual marketplace applications, particularly those that provide Internet auction services, that reduces complexity, increases accuracy, and maintains usability for Internet auction users. Specifically, the RMS addresses each of the issues highlighted above, i.e., it maintains the usability of evidence collection while increasing accuracy with regard to collecting evidence necessary to identify domain-specific behaviour types, and thus to limit the risk of exposure to untrustworthy entities; it increases the accuracy of evidentiary analysis with regard to contextual relevance assessment; it provides the ability to limit exposure to risk of unreliable recommended evidence; it provides the ability to detect behaviour types based on the assessment of interaction dynamics, and thus to limit the exposure to risk from colluding users; it makes risk explicit to the decision-maker; it provides a security decision as advice to a decision-maker as to how to interact; and finally, the RMS is based on the SECURE framework that unifies the characteristics and properties of human trust into a system that provides a decision-making mechanism in a manner analogous to human trust-based decision-making.

Our evaluation of the RMS shows that it reduces complexity, increases accuracy, and maintains usability of reputation management for Internet auction users. It validates that the RMS, in its observation and identification of normal and abnormal domain-specific behaviour, reduces complexity by providing accurate decision-making advice to users. Furthermore, the evaluation confirms that the analysis of context in terms of role, time, and environmental factors can further reduce complexity in the decision-making process while maintaining usability. Additionally, the evaluation demonstrates that recommendation weighting can protect a user against the potential unreliability of recommended evidence. Finally, the evaluation establishes that a reputation management system based on a computational trust-based decision-making model can counter the issues in existing commercial reputation management systems and provide increased benefit to users interacting in the Internet auction domain.

1.6 Organisation of this Thesis

This thesis is structured as follows. Chapter 2 introduces the characteristics of trust and reputation that are important for trust-based decision-making, and subsequently reviews and examines work related to this thesis in the areas of trust-based decision-making and reputation management. In Chapter 3, we present a discussion of the SECURE approach and the design of our reputation management system. The examination of the SECURE approach includes: a typology of the threats that SECURE is designed to address; the SECURE trust, collaboration, and risk models; the SECURE trust- and risk-based framework and its components; the implementation of the framework as a software kernel and application programming interface (API); as well as the deployment of SECURE for spam filtering. We develop a taxonomy of behaviour for the virtual marketplace reputation management domain that exposes the threats particular to this environment. The design of a trust-based Reputation Management System (RMS) based on the SECURE trust- and risk-based decision-making framework is then proposed. An overview of the design is given, and our rationale for our design decisions is presented with regard to requests, entity recognition, trust and evidence processes, risk assessment, and access control. Additionally, our rationale is described for extending the SECURE framework to include interaction management such that decision-making might be enhanced through the analysis of trustworthy recommendation paths between users and observations about potential colluding behaviour. We illustrate how these two new interaction management components may be integrated into the RMS and describe the enhanced decision-making process. In Chapter 4, we describe the evaluation of our work through simulation and case studies in which the decision-making mechanisms of the RMS are evaluated. Chapter 5 concludes this thesis with a summary of the work and a delineation of issues that remain open for future work.

Chapter 2: Trust and Reputation Management in Computer Systems

Every kind of peaceful cooperation among men is primarily based on mutual trust and only secondarily on institutions such as courts of justice and police.

~ Albert Einstein

In order to design decision-making tools, such as reputation management systems for virtual marketplaces, it is essential to understand the key features of the human decision-making processes of determining trustworthiness, assessing risk, and analysing social features of interaction groups. It is also necessary to understand what kind of evidence these processes require, how to collect that evidence, and how to determine evidentiary relevance for the types of decisions that need to be made. This chapter addresses current research into modelling and adapting traditional human decision-making processes based on trust and reputation for online interaction environments. Section 2.1 discusses various aspects of human trust, such that trust attributes and properties can be extracted and synthesised into a unified trust model. Section 2.2 presents recent research in formulating computational trust models in the computer security domain and examines whether any of these models capture the key trust characteristics. Section 2.3 explores more recent work on using human trust as the basis for devising a computational trust model. Section 2.4 addresses the use of recommendation and reputation in online environments. Note that trust-based decision-making systems and reputation management systems, sections 2.3 and 2.4 respectively, are considered separately for the purposes of presentation, even though there is ambiguity as to the definition of boundaries that would make one system type distinct from the other. Finally, section 2.5 presents some conclusions, including a list of techniques that could be used to build a reputation management system that incorporates the qualities of trust in such a way as to provide an electronic surrogate to the human trust-based decision-making process. We will incorporate these techniques as part of the design a reputation management system, discussed in Chapter 3.

2.1 Trust

Trust is a fundamental part of human interaction, and, in particular, it is an important feature in the decision-making process that humans use every day. This section presents a number of definitions of trust used in different fields in an attempt to build an understanding of the different ways in which

trust may be characterised. From these definitions, the properties and operations that apply to trust are extracted and integrated into a unified model of human trust. We then apply this trust model in an analysis of existing computational trust mechanisms with the goal of identifying trust management systems that closely approximate the human trust-based decision-making process.

2.1.1 Defining Trust

Most daily interactions are based on trusting decisions, i.e., a decision in which one person depends on another person in a particular situation even though a negative outcome is possible. This is the case even if the decision-making is done implicitly rather than explicitly. For example, a customer in a commercial transaction may trust that the goods/services he is receiving meet expectations and that his personal payment details will be kept private and secure by the selling party. In return, the seller accepts payment in return for the goods/services, trusting that the money or credit card is not stolen or forged and that the customer is able and willing to pay the price in full. Yet, because trust is an abstract concept, it would be difficult for either the seller or customer in such a transaction to definitively answer the questions 'what is trust?' or 'how would you define trustworthiness in this scenario?'

The answer to the question 'what is trust' is not easily provided. A basic definition of trust can be found in any dictionary, e.g., the Oxford English Dictionary (Oxford English Dictionary 2006) defines trust as follows:

> *1 : Confidence in or reliance on some quality or attribute of a person or thing, or the truth of a statement.*
>
> *2 : Confident expectation of something; hope.*

This general definition alludes strongly to subjective notions of *confidence* and *expectation* as well as more concrete concepts such as truth, but is insufficient to capture the various types of trust, each reflecting different dispositions, situations, evidence gathering processes, and belief updating mechanisms. While trust is difficult to explicitly define in this regard, researchers from various academic fields have attempted to do so, and consequently several definitions of trust exist. Marsh, who has contributed significantly to developing a formal model of trust (Marsh 1994), states that there are many types and views of trust as well as many fields which study the trust phenomenon, including such diverse fields as evolutionary biology, history, and economics. Main contributions emerge from work in the areas of sociology, social psychology, and philosophy, as well as work extending from these domains in the area of computational trust. These main contributions are discussed in the following, providing a base from which to extract the properties and operations possible when using trust to make decisions, which is then discussed in the following section.

2.1.1.1 Psychological Trust

Deutsch's work (Deutsch 1962) in social psychology, resulted in what is the most widely accepted definition of trust.

- *If an individual is confronted with an ambiguous path, a path that can lead to an event perceived to be beneficial (Va^+) or to an event perceived to be harmful (Va^-);*

- *He perceives that the occurrence of Va^+ or Va^- is contingent on the behaviour of another person; and*

- *He perceives the strength of Va^- to be greater than the strength of Va^+.*

> *If he chooses to take an ambiguous path with such properties, I shall say he makes a trusting choice; if he chooses not to take the path, he makes a distrustful choice. (Deutsch 1962)*

The repeated uses of the word perceive in the definition implies that trust has a quality of *subjectivity*, or *disposition* which depends on the individuals involved. That is, that trust is shaped in part by the way in which an individual forms beliefs according to his own disposition or point of view.

Moreover, this definition implies that *utility* is at the basis of a trusting decision. Because the result of following the path can be good or bad, and the negative impact of the bad result is greater than the positive impact of the good result, a person is further motivated to make the correct choice.[1] Thus, when making a trusting choice, Deutsch proposes that an individual is weighing up costs and benefits before making a final decision, i.e., that trusting decisions are based on a form of cost-benefit analysis. While *risk* is not explicitly mentioned, it is important to note that trust is linked to the perceived harmfulness of the ambiguous path as well as the fact that utility analysis is a common way to determine risk. Deutsch breaks trust down into several different circumstances where such a choice would definitely be made, but he concentrates on the fact that trust 'is strongly linked to confidence in, and overall optimism about, desirable events taking place.' If the ambiguous path were known to be entirely beneficial, i.e., without any risk, the choice to follow it would not be so great a dilemma

---

[1] The requirement that the bad outcome must have greater negative implications than the good outcome has positive implications has been countered in other work Golembiewski, R. T. and McConkie, M. (1975). The Centrality of Interpersonal Trust in Group Processes. Theories of Group Processes, Wiley. in which it is stated that the outcome disparity is not essential to the decision-making process.

for the decision-maker. Yet when risk exists, if there were no optimism or hope about the ambiguous path and/or its pay-off, a trusting choice would never be made. Furthermore, in order to perform a cost-benefit analysis, some type of risk-related information is required, and this point is revisited below when discussing the evidentiary inputs to trust-based decision-making.

Additionally, this definition takes into account *uncertainty*. There is ambiguity both in regard to the path and in regard to the other party on which an individual is dependent. If there were no uncertainty, there would be no need to make a trusting decision because the correct path to take would be known. Gambetta concurs with Deutsch regarding uncertainty, stating:

> *If we were blessed with an unlimited computational ability to map out all possible contingencies in enforceable contracts, trust would not be a problem. (Gambetta 2000)*

Thus, from Deutsch's definition, grounded in psychology, we extract three key features of what trust is and what it means to make a trusting decision: trust is *subjective* and is necessitated by the existence of *risk* present in entering into a situation in which *uncertainty* is caused by the unavailability of perfect information regarding potential outcomes. When an individual decides to enter into such a risky situation with imperfect knowledge, he makes a trusting decision.

2.1.1.2 Sociological Trust

Luhmann approaches trust from a sociological background (Luhman 1979), suggesting that trust is dependent on human relationships within societal networks. Luhmann proposes that the relation of the world as a whole to all of the individual diverse identities within it is extremely complex. This proposition implies that in order to make a trusting decision, an individual requires information, or evidence, about his environment and the diverse individuals within that environment, which can lead to complexity, or evidence that may be so abundant that an individual may not be able to process it all. In this type of environment, trust is needed as a *means to reduce complexity* and promote adaptation through increasing the possibility for experience and interaction.

This definition is reinforced by Shklar, who describes trust as a way of coping with ignorance, i.e., 'the limits of our foresight' (Shklar 1984). Similarly, Reagle states, 'trust itself represents an evaluation of information, an analysis that requires decisions about the value of specific information in terms of several factors' (Reagle 1996). Thus, *evidence gathering* and *evidentiary analysis* are two issues that are highlighted as relevant to the reduction of complexity in order to make a trusting decision. The evidence required to make a decision may take various forms. For example, if an individual has firsthand experience of the behaviour of another party, he has directly observed evidence, i.e., an *observation*. Alternatively, if an individual has no experience of interacting with another party, he may be willing to rely on indirect evidence, i.e., *recommendations*, from others who have interacted with that party. As mentioned above, *evidence about risk*, i.e., the potential costs and

benefits, is also required as an input to making a trusting decision. Moreover, we find in the discussion below that *evidence about context* also plays an important part in a trust-based decision-making process. While humans do not explicitly use processes of evidence gathering and evidentiary analysis, they do make observations, request and assess recommendations, and perceive context and risk. Humans typically implicitly analyse this evidence in some manner when making a trusting decision.

Moreover, in a situation where full information is not available, the likelihood of making an incorrect choice is increased, and this leads to *risk*. Trust has been proposed as a mechanism for the reduction of complexity, and therefore a reduction in risk exposure, in situations by making assumptions based on the environment and situation in which interaction is taking place.

Luhmann's definition also suggests that when faced with *uncertainty* due to complexity in decision-making, humans make *context*-based assumptions, i.e., assumptions about their environment, in order to proceed to the end of the decision-making process. With regard to the earlier discussion about evidence, we find that information about context is another type of evidence that needs to be assessed when making a trusting decision. For example, Alice might trust Bob to drive a car but not to fly a plane, and the context, i.e., 'drive a car' or 'fly a plane', must be specified in order for Alice to evaluate the purpose for which she is making a trusting decision about Bob.

Barber, like Luhmann, attempts to solidify a sociological definition of trust. He links trust to societal network relations rather than to individual cost-benefit assessments. He views trust 'predominantly as a phenomenon of social structural and cultural variables and not...as a function of individual personality variables' (Barber 1983). In 'The Logic and Limits of Trust', Barber gives three expectations that are taken into account when making trusting decisions:

1. *Expectation of the persistence and fulfilment of the natural and moral social orders.*

2. *Expectation of 'technically competent role performance' from those we interact with in social relationships and systems.*

3. *Expectation that partners in interaction will 'carry out their fiduciary obligations and responsibilities, that is, their duties in certain situations to place others' interests before their own. (Barber 1983)*

These expectations imply that individuals do not operate in a vacuum, reinforcing Luhmann's indication that there is societal *context* guiding the decision-making process. Moreover, individuals can be seen to fall into more abstract '*roles*' with assigned duties rather than each individual being entirely unique in his interactions.

From the field of sociology, then, we find trust characteristics in common with those identified by Deutsch, i.e., *uncertainty* and *risk*, as well as additional features of trust-based decision-making, i.e., using the processes of *evidence gathering* and *evidentiary analysis* to collect and assess *observations*, *recommendations*, and *evidence about risk and context*, as well assessing *context* and *roles* when using trust as a *means to reduce complexity*.

2.1.1.3 Philosophical Trust

In the article 'Can We Trust Trust?' which is included in the collection 'Trust, Making and Breaking Cooperative Relations,' Diego Gambetta amalgamates work from such diverse areas as biology, music, history, and economics to define trust philosophically, stating:

> *In this volume there is a degree of convergence on the definition of trust which can be summarized as follows: trust (or, symmetrically, distrust) is a particular level of the subjective probability with which an agent assesses that another agent or group of agents will perform a particular action, both before he can monitor such action (or independently of his capacity ever to be able to monitor it) and in a context in which it affects his own action... When we say we trust someone or that someone is trustworthy, 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. Correspondingly, when we say that someone is untrustworthy, we imply that that probability is low enough for us to refrain from doing so. (Gambetta 2000)*

Gambetta's definition provides several aspects of trust. This definition again reinforces the *subjective* nature of trust. The definition also takes *uncertainty* into account by stating that trust is affected by actions that an individual may not be able to monitor. Another aspect of trust relates it to the amount of information available about one's behaviour when interacting in a given situation, again alluding to *evidence gathering* and *evidentiary analysis* processes. This means that in a situation where full information is not available, the quality and meaning of the information one can monitor will have a major effect on his calculation of threshold levels of trust. The definition also recognizes that trust is necessary only when there is a possibility of *risk*, i.e., probability of someone being untrustworthy in a given context.

Finally, a heretofore unmentioned trust characteristic is identified in this definition, as Gambetta supposes that trust can be represented mathematically and therefore is *quantifiable*, or able to be measured, which means that the definition 'becomes more concrete than abstract compared to other definitions presented earlier' (Lamsal 2001).

2.1.1.4 Computational Trust

Recently, the notion of human trust has been taken up with interest by the computing community.

McKnight and Chervany define trust as:

> *… the extent to which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible. (McKnight and Chervany 1996)*

This definition, like that of the Oxford English Dictionary, demonstrates that *abstract entities* as well as humans might be trustworthy, although they do not have the ability to act honestly or dishonestly in the way that humans do. Like Deutsch, McKnight and Chervany highlight the *risk* of a negative outcome, although their definition does not explicitly express the element of uncertainty. McKnight and Chervany also specifically address the element of *context* by saying that trust is formed in light of a particular situation. Finally, mentioning the possibility of negative consequences presupposes the existence of *risk* in entering into a given situation.

In a recent survey of trust in the context of networked and distributed computing systems, Grandison and Sloman (Grandison and Sloman 2000) define trust as 'the firm belief in the competence of an entity to act dependably, securely, and reliably within a specified context (assuming dependability covers reliability and timeliness)' and they define distrust as 'the lack of firm belief in the competence of an entity to act dependably, securely, and reliably within a specified context.' The *subjectivity* of trust is implied by the word 'belief' in this definition. Again, *context* is referred to as strongly linked to trust, and a *time* element of context is indicated.

These recent definitions of trust as developed in the computer science domain are therefore not dissimilar to the definitions provided from work on trust in psychology, sociology, and philosophy.

2.1.2 A Summary of Trust Definitions

The above definitions capture several characteristics of trust, many of which are found in more than one definition. First, trust encompasses some meaning about *confidence* or *expectation* regarding the outcome of an interaction in a situation in which it is necessary to rely upon another party. Additionally, trust is *subjective* and depends on the disposition of the individual person or *entity* making a trusting decisions. Second, trust is dependent upon the *context* or environment in which a trusting decision is being made. Context includes, but is not limited to, the elements of *role* and *time*. Third, a trusting decision is made in the absence of complete information regarding potential outcomes, i.e., the decision is made when there is *uncertainty*. Uncertainty can indicate two different possibilities with regard to the availability of evidence: first, that complete evidence about trustworthiness, context, and risk is unavailable to the person making a trusting decision; or that the

amount of evidence available is so copious that a person would not have the time or resources available in order to analyse it to make a trusting decision. In both of these cases, an individual relies on trust: in the former case, a person uses trust to make a decision despite the existence of uncertainty when complete information is unavailable; and in the latter case, a person uses trust to *reduce complexity* when information is so abundant that it cannot all be analysed in time to make a trusting decision. Thus, a trusting decision is made based on the explicit or, more typically, implicit, processing of *evidence* consisting of *observations*, *recommendations*, and *evidence about risk and context*, through *evidence gathering* and *evidentiary analysis*. In fact, using trust as a means to reduce complexity implies that a person analyses only the most relevant evidence to make a decision. Furthermore, the element of *risk* is required for trust to be necessary in decision-making, because without risk, there is no need to trust. Finally, trust can be quantified in some way, i.e., a *trust measure* can be subjectively calculated based on the analysis of collected evidence about a given person or entity in a particular context for which the outcome holds uncertainty and risk.

2.1.3 Properties of Trust

The discussion in the previous section provides working definitions with which to discuss trust. A set of trust characteristics were extracted from the definitions, and in this section we augment the characterisation of trust by defining several properties of trust that, regardless of how trust is defined, are held to be true. These properties include diversity, subjectivity, symmetry, transitivity, context, and measure.

2.1.3.1 Trust Diversity

Trust is usually specified in terms of a relationship, or trust purpose, between a subject, or trust origin, and a trust target, with each being represented by a trust dimension (Grandison and Sloman 2000; Jøsang, Gray et al. 2006). Moreover, each dimension can express trust diversity, as illustrated in Figure 1 below. In the case of the first trust dimension, several trust origins may have the same trust purpose with the trust target. An example of this would be when many individuals choose the same service provider to provide telephone service. In the second case, one trust origin trusts one trust target for many different trust purposes. For instance, a person may trust another person for the purposes of recommending restaurants as well as flying a plane and driving a car. Finally, trust target diversity occurs when one trust origin trusts many trust targets for the same trust purpose. For example, trust target diversity occurs when a person is happy to purchase milk at any number of different grocery stores.

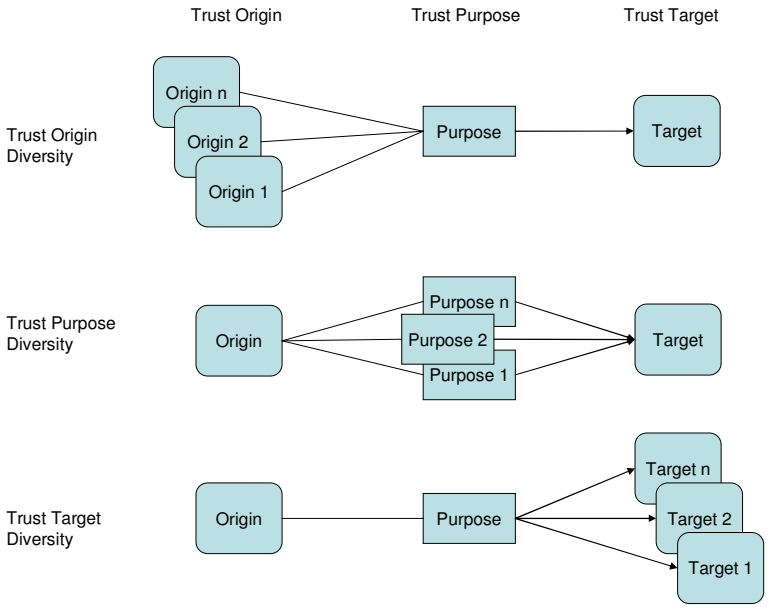

Figure 1: Diversity dimensions of trust

Having specified trust in terms of trust origin, trust purpose, and trust target, other trust properties become apparent. First, as extracted from the definitions in the previous section, we know that the trust origin is subjective in calculating trust for a trust target for a given trust purpose. Next, the properties of symmetry and context can be applied to the trust purpose. Third, a trust measure can be determined by a trust origin for a trust target for a trust purpose. Finally, a trust measure can be recommended as evidence to a third party due to the transitive trust property. Each of these properties is more fully detailed below.

2.1.3.2 Trust Subjectivity

The subjective nature of trust is evident in most of the definitions presented in the last section. A trust origin can have personal interests on several levels, e.g., economic, emotional, and social (Jonker and Treur 1999), and the notion of subjective probability, i.e., the probability that a one's subjective interest in an interaction outweighs the potential risk involved, can be used to capture those personal interests. Elofson notes that a subjective assessment when making a trusting decision is 'the outcome of observations leading to the belief that the actions of another may be relied upon, without explicit guarantee, to achieve a goal in a risky situation,' emphasizing that trust can be developed over time through a series of confirming evidentiary observations (Elofson 1998).

Jonker and Treur (Jonker and Treur 1999) attempt to analyze and formalize the subjective nature of trust as a classification of trust dynamics that are the basis for a formal model of trust updating. They

illustrate how trust-positive and trust-negative interaction results lead to different assessments of trustworthiness in agents with different subjective interpretations of observed evidence. For example, a blindly trusting agent will increase trust levels regardless of a positive or negative outcome with another agent, whereas a blindly untrusting agent will never trust another agent.

Various trust dynamic strategies lie in between these two extremes, as explored in (Axelrod 1984; Marsh 1994; Jonker and Treur 1999; Gray, O'Connell et al. 2005), and illustrate that disposition significantly impacts the evaluation of trustworthiness. For example, in (Jonker and Treur 1999; Gray, O'Connell et al. 2005), in addition to the blind trust and blind distrust extremes, trust may be evolved according to:

- a slow-positive-fast-negative policy, in which an entity is subjectively less trusting than distrusting, i.e., slow to increase trust in another party but rapid to decrease trustworthiness after having observed incorrect behaviour;

- a fast-positive-slow-negative policy, in which an entity is subjectively more trusting than distrusting, i.e., increases trust rapidly when correct behaviour is observed when interacting with another party, and giving the other party the 'benefit of the doubt' by decreasing trust slowly when incorrect behaviour is observed.

- a balanced-fast policy, in which case trust is evolved rapidly in both positive and negative directions; and

- a balanced-slow policy, in which an entity is more slow to evolve trust either positively or negatively. In (Gray, O'Connell et al. 2005), it is found that this strategy produces the smoothest trust evolution results due to the lack of large increases or decreases in trust assessment that occur when the other subjective assessments of correct or incorrect behaviour are used.

Similarly, Axelrod (Axelrod 1984) studied how cooperation evolves in groups by assessing the outcomes of interaction behaviour in a Prisoner's Dilemma tournament. The Prisoner's Dilemma game is based on a scenario in which two people are arrested for a crime and are questioned independently by authorities. Each prisoner is given the opportunity to either cooperate with his accomplice, i.e., not give any information to the authorities, or defect by snitching on his accomplice. A matrix can be constructed to illustrate the pay-off to the players depending on how they behave, given in Figure 2. A typical pay-off is to give 3 points to each player if they both cooperate; if one cooperates and the other defects, the defector gets 5 points and his partner gets none; and if both players defect, they each only receive 1 point. The total pay-off to both players is greatest for mutual cooperation, 6 points, while a cooperate-defect play results in 5 points, and mutual defection results in only 2 points. Clearly, it is best for the collective system if all players cooperate, however, each player can win at least 1 point, and potentially a maximum of 5, points by defecting in any given situation.

		Player 2	
		Cooperate	Defect
Player 1	Cooperate	R = 3, R = 3 R: reward for mutual cooperation	S = 0, T = 5 S: sucker's pay-off T: temptation to defect
	Defect	T = 5, S = 0 S: sucker's pay-off T: temptation to defect	P = 1, P = 1 P: punishment for mutual defection

Figure 2: Pay-off matrix for the Prisoner's Dilemma

In 1979, Axelrod hosted a tournament to determine which cooperation strategies would perform best, i.e., result in the largest pay-off over time, in an Iterated Prisoner's Dilemma (IPD) game. Researchers from various fields submitted IPD strategies in the form of computer algorithms, and the submitted strategies were tested against one another in repeated round-robin PD games. The strategy that accumulated the highest total pay-off in the end was declared the tournament winner.

The 14 strategies evaluated included those such as Always Cooperate and Always Defect, similar to blind trust and blind distrust, as well as more elaborate strategies, such as Nice (initial cooperation), Mean (initial defection), and Tit-for-Tat. The Tit-for-Tat strategy is one in which a player follows the Nice strategy in the first round against another player and then subsequently mirrors the strategy of the other player, i.e., if Player 2 cooperates in a game, Player 1 cooperates in the following game against Player 2; and if Player 2 defects in a game, Player 1 defects in the following game against Player 2. The result of the tournament reveals that a Tit-for-Tat cooperation strategy yields the best long-term results for interacting entities over time. Moreover, the single characteristic that distinguished the high-scoring strategies from the low-scoring strategies in Axelrod's tournament was the property of 'niceness', i.e., never being the first to defect.

Marsh (Marsh 1994) extends Axelrod's work in an IPD PlayGround for artificial agents that interact with each other according to subjective trusting strategies: optimist, pessimist, and realist. For the optimist strategy, an agent uses the highest remembered trust value for an interaction partner in a given situation. The pessimist strategy employs the lowest trust value available for an interaction partner in any situation. A realist agent uses the mean trust value for all past interactions with a particular interaction partner in a given situation, or, if the interaction partner is not known in a specific situation, the mean is taken for all other situations. Marsh's findings include: optimists are quicker to cooperate than realists and appear to benefit, i.e., accrue more points, from repeated risk-taking, which is similar to Axelrod's niceness results; if initial trust is too low, pessimists will never cooperate; trusting behaviour can be educated; cooperation can be encouraged; some of the agents behaved in a manner similar to humans; and, most importantly, that trust can be formalised and that the formalism can be embedded in artificial agents to produce subjective trust-like behaviour.

The results of the work in trust evolution and cooperation evolution policy evaluation conveys the importance of the subjectivity, or disposition, of an entity evaluating the trustworthiness of another

entity based on observed interaction behaviour. For example, an person or entity following a blind-trust or optimist trust evolution policy or an Always Cooperate cooperation policy disregard the potential risk of interacting with another party and always believe that the other party may be relied upon during future interactions. The opposite holds true for a person or entity following a blind-distrust or Always Defect policy, whereby the potential pay-off from non-interaction is held to be greater than any possible reliance on another party. Moreover, an intermediate strategy, such as balanced-slow or Tit-for-Tat, resembles a moderate stance on determining the way in which to interact with another party based on that party's past behaviour. In this way, evidence about an entity's past behaviour is subjectively assessed when making a trusting decision about future interactions with that entity. Thus, disposition affects system performance and optimal trust evolution or cooperation strategies are typically those that allow for moderation.

2.1.3.3 Trust Asymmetry

From the discussion on trust dimension diversity, it is seen that a relationship between trust origin and trust target need not be, and indeed is not typically, symmetric. In addition to the possibility of one-to-many trust relationships, even in a one-to-one relationship Alice's trust in Bob may not be the same as Bob's trust in Alice. This phenomenon is evidenced in social network analysis (Coleman and Lal 1990; Watts 1999; Watts, Dodds et al. 2002; Latora and Marchiori 2003), where there is a directional relationship between nodes, e.g., teacher to student, employer to employee, seller to buyer, parent to child, etc. Where directional relationships exist, trust may be assessed differently depending on which entity is performing the role of trust origin and which is performing the role of trust target. Trustworthiness may be assessed differently, depending on the role, e.g., a trust origin who is a buyer of goods calculates trust for the trust target, a seller of goods, based on criteria such as quality and price, while the seller evaluates the trustworthiness of a buyer based on criteria such as ability and intention to pay. The trust purpose in each case is different depending on the directionality of the relationship being evaluated. Moreover, the evidence being input to make a trusting decision is also dependent on the directionality of the relationship. Evidence about a seller's ability and intention to pay for goods when he acts in the role of a buyer will probably not be of as much importance to an entity assessing whether or not the seller can be trusted to provide goods of a high quality. Thus, trust purpose typically implies some notion of role context that is important to making trusting decisions.

2.1.3.4 Trust Context

In addition to the contextual element of role, a trust purpose usually occurs within an overall context. Context, like trust, is an abstract concept with many definitions. A general definition is given by the Merriam-Webster Dictionary which states that context is 'the interrelated conditions in which something exists or occurs: environment, setting' (Merriam Webster Dictionary 2006). That definition is made more precise by Dey, who defines context for the computing domain:

> *Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and application, including the user and applications themselves. (Dey 2001)*

Dey's definition of context highlights that context is variable and includes all of the possible information about an entity's current state, i.e., if a piece of information can be used to characterize the situation of a participant in an interaction, then that information is context. This may include location and time, but it may additionally include situation-specific information or the acknowledgement that other actors have entered the interaction thus changing the situation.

Most trust definitions include the aspect of context, demonstrating that trust is context-specific. Moreover, context is the basis for one of the dimensions of trust, i.e., the trust purpose. The following statement is an example of this: 'Alice trusts Bob to drive a car, but not to fly a plane.' As the situation, or trust purpose, changes, so may the trust level calculated by the trust origin for the trust target. Contextual changes that affect trust purpose include variations in role, time, and other environmental factors.

For example, perhaps Alice trusts Bob in the role of car driver but not in the role of plane pilot. The trust purpose that Alice, the trust origin, evaluates for each role assumed by Bob, the trust target, is changed when the context of Alice's situation is modified from deciding whether or not to get into a car with Bob to whether or not to get into a plane with Bob. In both cases, the trust purpose is essentially the same, i.e., Bob's ability to navigate a vehicle, but Bob's role modifies this general trust purpose somewhat in each case.

Time is another contextual component that affects trust because interactions between parties may take place over a period of time rather than occurring as once-off transactions (Coleman and Lal 1990; Marsh 1994; Jøsang, Gray et al. 2006). Quite obviously, a trusting relationship for a given trust purpose at one point in time might be evaluated differently at a later point in time. This means that time can be modelled as a set of discrete events taking place in which both the trust origin and trust target are involved. However, even if no transactions take place, a trust relationship will gradually change as time passes, e.g., eventually degrade. Therefore, in addition to the discrete changes that are made when events occur, it is also necessary to take into account the changes made to trust relationships when no recent evidence is available.

Other environmental factors that affect context include domain-specific factors. For example, in the domain of trade, a trust origin may evaluate the trust purpose for a trust target differently depending on the type of item being exchanged or the cost range of the item.

Environmental constraints are also part of context that affect trust assessment. If two parties are interacting in a system with enforceable rules or insurance, e.g., a credit card system with validation and stop-payment processes, they may be more willing to engage in a potentially risky interaction

than they would in a system with no enforceable rules, e.g,. black market currency exchange on a dark corner of a shady neighbourhood. Thus, assessing risk is also context-dependent, e.g., Alice may not trust Bob to fly a plane unless Bob's flying the plane is the only possible means to escape certain death. At some point, environmental constraints may become so all-encompassing that risk becomes irrelevant, meaning that trust is unnecessary when making a decision to interact because either sufficient safeguards are in place or because the interaction is essential. It is in cases below this threshold in which trust and risk assessment processes are necessary.

Finally, it has been stated that 'evidence without context is ambiguous at best' (Serafian 2003). When humans scrutinise collected observations and recommendations, each piece of information is examined in terms of its contextual relevance to the conclusion and resolution of the decision at hand. The contextual relevance of evidence, i.e., in terms of role, time, environmental factors, and environmental constraints, must be examined to lessen ambiguity when forming trusting decisions.

2.1.3.5 Trust Transitivity

Trust transitivity means that if Alice trusts Bob and Bob trusts Carl, then Alice will also trust Carl. This assumes that Bob actually tells Alice that he trusts Carl, i.e., as a recommendation based on Bob's observations about Carl. It has been shown (Christianson and Harbison 1996) that trust is not implicitly transitive. For example, as illustrated in Figure 2, the fact that Alice trusts Bob to mind her children and Bob trusts Carl to drive a car does not imply that Alice trusts Carl to drive a car nor to mind her children.

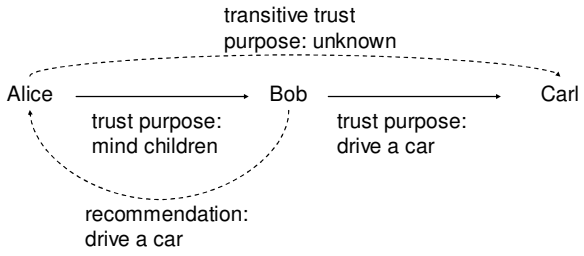

Figure 3: Trust transitivity leading to an unknown trust purpose

However, while trust is not inherently transitive, a form of transitivity can be achieved by the recommending of observations according to explicit guidelines (Jøsang, Gray et al. 2006). It is important to distinguish between a functional trust purpose, i.e., the actual ability of the trust target as observed through direct interaction; and a referral trust purpose, i.e., the ability of the trust target to recommend another party. For example, in Figure 4, Alice trusts Bob to recommend Carl as a good driver, i.e., Alice has direct observations for the trust purpose that Bob makes useful recommendations about drivers to her. This is direct referral trust. In this scenario, Bob has direct functional trust in

Carl based on his own observations about interacting with Carl for the purpose of driving. Bob's recommendation is used by Alice as evidence with which to derive indirect functional trust in Carl. The recommendation chain between Alice and Carl can be arbitrarily long, but as long as the last leg in the chain is comprised of a functional trust purpose, or observation, and the other legs in the chain are comprised of referral trust purposes, or recommendations, trust transitivity is provided. Transitive trust propagation is thus possible when two variants of trust purpose, functional and referral, are explicitly assessed.

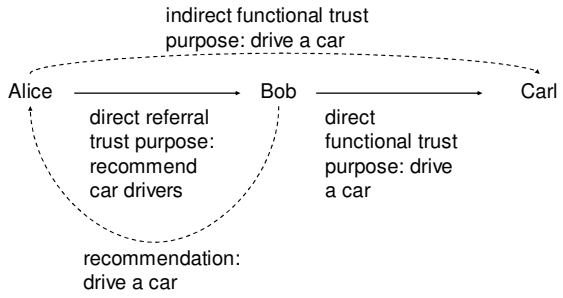

Figure 4: Trust transitivity leading to a known trust purpose

Moreover, Alice may wish to combine all collected evidence, both her own observations and recommendations from other parties, to form overall trust in a trust target. Parallel combination of evidence, i.e., the process of combining evidence from multiple paths between trust origin and trust target, is particularly useful in situations in which a trust origin has few or no observations with which to form direct functional trust in a trust target and must rely upon recommendations from other parties, i.e., rely on referral trust, in order to assess the target's trustworthiness. For example, Figure 5 illustrates the available evidence Alice may need to combine in order to assess Eric's trustworthiness when she has no direct functional trust in Eric for trust purpose σ and two indirect recommendations for Eric.

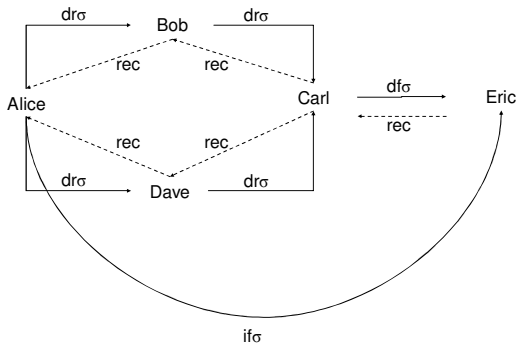

Figure 5: Parallel combination of recommendations to derive indirect functional trust

Jøsang et al. (Jøsang, Gray et al. 2006) propose notation to describe the process of deriving indirect functional trust. Principals are denoted as A, B, C, D, and E, rather than Alice, Bob, etc. A directed graph represents the transitive trust network, with a single trust relationship between trust origin and trust target for trust purpose represented as a directed edge, e.g., the edge [A, B, σ] means that A trusts B for the trust purpose σ. We recall that the trust purpose, e.g., drive a car, can have two variants, functional and referral, and that the trust purpose can be direct or indirect, giving the set of possible trust purposes {dfσ, ifσ, drσ, irσ}. Additionally, the symbol ":" denotes the transitive connection of two consecutive trust edges to form a transitive trust path. Thus, the trust relationships in Figure 5 are expressed as: ([A, E, ifσ]) = ([A, B, drσ] : [B, C, drσ] : [C, E, dfσ]) and ([A, E, ifσ]) = ([A, D, drσ] : [D, C, drσ] : [C, E, dfσ]). Referral trust is calculated for each path and then applied to the functional trust being recommended on the last edge of the path, giving trust from parallel combination as ([A, E, ifσ]) = ((A, B, drσ] : [B, C, drσ]) ◊ ([A, D, drσ] : [D, C, drσ] : [C, E, dfσ])). The parallel combination operator, ◊, is trust system dependent, and examples of specific combination methods are given in the following section when discussing trust measures.

Intuitively, A's trust in E is greater after performing the parallel combination of recommendations than if solely relying on one recommendation. Trust in E would be increased further by combining additional recommendations, as well as combining ifσ and dfσ when A has her own observations to input into the combination process. Parallel combination of transitive trust therefore has the effect of strengthening derived trust.

2.1.3.6 Trust Measure

When making a trusting decision, a human trust origin does not typically assign an explicit trust measure to a trust target. Humans generally determine trustworthiness implicitly, or intuitively, without explicitly analysing evidence about trust, risk, and context. However, it has been shown that trust can be measured in a similar manner to other abstract commodities such as information or knowledge (Dasgupta 2000). A measure can be associated with each trust relationship that captures a trust origin's subjective degree of belief based on evidence about a trust target's trustworthiness for a given trust purpose.

To explicitly calculate a trust measure, however, evidence must be analysed, i.e., a trust origin forms trust about a trust target for a trust purpose based on evidence that has been observed and/or recommended about how the trust target behaved in past interactions with similar trust purposes. The evidence is assessed for relevance according to the context of the trust purpose, i.e., the directionality (role of the trust target), environmental factors, and timeframe of interactions. Moreover, the quality of recommended evidence, i.e., recommendation integrity, is usually also taken into account. The measure is shaped by the disposition, or subjectivity, of the trust origin, as well as the context of the trust purpose. The outcome of processing the combined evidence is the formation and evolution of a measure of how much a trust origin trusts a trust target to fulfil a given trust purpose, and this measure

can be used to determine whether or not to assume the risk associated with interacting with the target for that purpose.

Many types of trust measure have been proposed in the literature, including binary measures of trusted and not trusted; discrete measures such as complete trust, high trust, medium trust, low trust, no trust (Zimmerman 1995; Abdul-Rahman and Hailes 1997; Manchala 1998; Cahill, Shand et al. 2003); or measures that are continuous in some form such as probability, Bayesian, and belief function representations (Beth, Borcherding et al. 1994; Marsh 1994; Jøsang 1996; Kohlas and Maurer 2000; Jøsang 2001; Kinateder and Pearson 2003; Gray, O'Connell et al. 2005).

Measuring Trust as a Probability

For example, Marsh uses probability to express the many aspects of social trust, i.e., representing the many facets of trust, such as 'competence' and 'risk', as continuous variables between 0 and 1. The variables are combined to produce an overall probability of trustworthiness. In this case, a trust origin calculates a trust measure as a probability that a trust target will behave competently, reliably, etc. for a trust purpose, capturing context, based on the percentage of time that trust target has behaved correctly in the past.

It has been argued (Luhman 1979; Zadeh 1986; Jøsang 1996; Abdul-Rahman and Hailes 1997) that probability is not the correct way to represent trust because it cannot express uncertainty and because computing transitivity may produce counterintuitive results, e.g., using multiplication to calculate transitive trust for a path of trust measures less than 1 will tend to produce a probabilistic trust measure that approaches 0, which expresses distrust rather than ignorance about likely outcomes of interacting with a trust target. Therefore, the Bayesian approach and belief functions have been proposed as mathematical theories of evidence to more accurately represent trustworthiness and capture uncertainty.

A Bayesian Approach to Trust Measurement

A Bayesian approach to trust measurement has been proposed (Mui, Mohtashemi et al. 2001; Jøsang and Ismail 2002; Jøsang, Hird et al. 2003; Buchegger and Le Boudec 2004; Whitby, Jøsang et al. 2005) in which a trust measure is calculated based on the statistical updating of beta probability density functions (PDFs) in which an *a posteriori*, i.e., updated, measure is arrived at by combining the *a priori*, i.e., previous, measure with new evidence. The measure is represented in the form of a beta PDF parameter tuple (α, β) where α represents the number of observations of positive, or trustworthy, behaviour, and β represents the number of times dishonest behaviour has been observed. For example, if the *a priori* measure is (α, β), when a new observation is made, e.g., *h* honest

behaviour observations and *d* dishonest behaviour observations, the *a priori* is updated according to $\alpha = \alpha + h$ and $\beta = \beta + d$. The beta PDF denoted by $\text{beta}(p \mid \alpha, \beta)$ is given by gamma function Γ as: $beta(p \mid \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}$ where p is a probability variable representing the probability of an event and density $\text{beta}(p \mid \alpha, \beta)$ represents the probability that p has a specific value, given α and β. The expected probability of a certain outcome , i.e. a particular distribution, occurring is given by: $E(p) = \alpha/(\alpha+\beta)$. For example, if Alice has observed Bob's behaviour to be honest 8 times and dishonest 2 times, the probability expectation value is *E(p)* = 0.8, which can be interpreted as saying that the relative frequency of a positive outcome in the future is somewhat uncertain, according to the beta distribution, and that the most likely probability of a future positive interaction is 0.8.

The advantage of Bayesian systems is that they provide a statistically sound basis for calculating trust measures, they capture uncertainty, and they only require two parameters for continuous updating as new observations are made or reported. A disadvantage with this approach, however, is that the calculations may be too complex for comprehension by average system users.

Measuring Trust as a Function of Belief

A belief function assigns probability values to sets of possible outcomes rather than single events, thus encoding evidence as supporting, contradicting, or being inconclusive about a given proposition's plausibility. Based on Dempster-Shafer theory (Shafer 1976; Shafer 1990), the belief model is related to probability theory, however, the probabilities of potential outcomes do not necessarily add up to 1, and the remaining probability is assigned to the union of the outcomes. Belief calculus has been found to be suitable (Jøsang 1999) to reason about outcomes in situations in which there is uncertainty about the truth of a given proposition, such as when a trust origin is trying to calculate a trust measure for a trust target based on incomplete information. Subjective logic is a specific belief calculus that uses a metric called an 'opinion' to express beliefs. For example, an opinion denoted by $\{b,d,u,a\}$ where $b+d+u=1$ and $\{b,d,u\} \in [0,1]$, and where *b* represents belief, *d* represents disbelief, and *u* represents uncertainty. Relative atomicity, i.e., the size of the state space of the potential outcome, is represented by *a*, where $a \in [0,1]$. Also called the base rate, *a* determines how uncertainty shall contribute to the probability expectation value, and in the absence of any specific evidence about a given party, the base rate determines the general level trust that would be put in any member of the community. An opinion thus captures a trust origin's belief in the truth of a proposition about a trust target for a trust purpose.

For example, if Alice wants to reason about the proposition, *x*, 'Carl will cooperate, i.e., behave correctly, when selling items in a virtual market', she will assess observations and recommendations

in terms of the extent to which the evidence supports this proposition. Alice may have observed that Carl behaves correctly for the trust purpose, e.g., delivering goods of high quality in return for payment, 7 times, that he did not behave correctly 2 times, and that he has been in 1 interaction of which the outcome is unknown, giving uncertainty. The relative atomicity of the state of behaving correctly relative to the full state space, {cooperates, defects}, is ½, and therefore the uncertain probability of Carl cooperating is 0.5. Alice's trust measure for Carl for *x*, therefore, can be represented as the opinion ω_x^{AC} = {0.7, 0.2, 0.1, 0.5}. In order to calculate a probability expectation for Carl's behaviour in a future transaction, the formula $E(\omega_x^{AC}) = b + au$ is used, which produces 0.75, i.e., a 75% likelihood that Carl will cooperate as a seller in a future transaction based on the evidence of Alice's observations of Carl's past behaviour and Alice's uncertainty.

If Alice has no observations about Carl's past behaviour, or if she has only limited observations, she may wish to accept recommendations from other sources. For example, Bob may have observed Carl's past behaviour and have an opinion about Carl's trustworthiness for the trust purpose, *x*, ω_x^{BC} = {0.8, 0.0, 0.2, 0.5}. If Alice requests a recommendation from Bob about Carl as a seller, Bob would pass this opinion, ω_x^{BC}, to Alice. Alice might not wish to take Bob's recommendation at face value, that is, she may wish to assess Bob's trustworthiness as a recommender, i.e., Bob's recommendation integrity, and scale his recommendation accordingly. Methods proposed for discounting recommendations for trust-based decision-making include semantic distance analysis (Abdul-Rahman and Hailes 1997; Dimmock, Bacon et al. 2005; Jøsang and Pope 2005) and path distance analysis (Gray, Seigneur et al. 2003; Jøsang, Gray et al. 2006). Semantic distance is the divergence between a recommender's observations about a trust target and a trust origin's own observations about the same target. For example, Alice might observe that Carl is very trustworthy, while Bob observes that Carl is moderately trustworthy; and each time Alice receives a recommendation from Bob about Carl, she discounts it appropriately. Path distance can also be used to discount recommendations, i.e., the longer the transitive trust chain linking a trust origin to a trust target, the more uncertainty is inherent in the recommendation. When using subjective logic to measure trust, both semantic and path distance are captured by increasing the uncertainty parameter of a recommended opinion. The discounting operator, $\otimes$, is used in subjective logic (Jøsang 2001) to discount a recommendation according to the recommendation integrity of the recommender. For example, if Bob recommends the opinion $\omega_x^{BC} = \{b_x^{BC}, d_x^{BC}, u_x^{BC}, a_x^{BC}\}$ about Carl's trustworthiness as a seller, and Alice has an opinion about Bob with regard to the proposition, *r*, 'Bob will cooperate for the purpose of recommending sellers', $\omega_r^{AB} = \{b_r^{AB}, d_r^{AB}, u_r^{AB}, a_r^{AB}\}$, the discounted recommendation is $\omega_x^{ABC} = \omega_r^{AB} \otimes \omega_x^{BC}$, which is calculated as an opinion such that:

1. $b_x^{ABC} = b_r^{AB} b_x^{BC}$

2. $d_x^{ABC} = b_r^{AB} d_x^{BC}$

3. $u_x^{ABC} = d_r^{AB} + u_r^{AB} + b_r^{AB} u_x^{BC}$

4. $a_x^{ABC} = a_x^{BC}$

Where b_x^{ABC} gives Bob's belief in Carl's trustworthiness as a seller, which is discounted by Alice's belief in Bob as recommender of sellers; d_x^{ABC} gives Bob's disbelief in Carl's trustworthiness as a seller, which is discounted by Alice's belief in Bob as recommender of sellers; u_x^{ABC} gives overall uncertainty, i.e., Alice's disbelief in Bob as a recommender in sellers, which is combined with Alice's uncertainty in Bob as a recommender in sellers, which is then combined with Bob's uncertainty in Carl's trustworthiness as a seller, which has been discounted by Alice's belief in Bob as recommender of sellers. Relative atomicity, i.e., a_x^{ABC}, remains the same, as there is no change in the state space of the proposition that Carl is a trustworthy seller.

If ω_r^{AB} = {0.9, 0.0, 0.1, 0.5} and ω_x^{BC} = {0.8, 0.0, 0.2, 0.5}, then, using the above discounting formula, ω_x^{ABC} is the discounted recommendation {0.72, 0.0, 0.28, 0.5}, in which uncertainty, rather than disbelief, has been increased. This makes sense intuitively because Alice is less certain about Bob's observations of Carl rather than more disbelieving of Carl's actions.

Subjective logic also provides a method for combining opinions such that the combination of two opinions reflects both opinions in a fair and equal way (Jøsang 2001). For example, Alice could combine her own opinion of Carl, ω_x^{AC} = {0.7, 0.2, 0.1, 0.5}, with the discounted recommendation (as above) from Bob about Carl, ω_x^{ABC}, in a consensus operation that reflects both opinions, i.e., $\omega_x^{AC,ABC} = \omega_x^{AC} \oplus \omega_x^{ABC}$ such that:

1. $b_x^{AC,ABC} = \dfrac{b_x^{AC} u_x^{ABC} + b_x^{ABC} u_x^{AC}}{u_x^{AC} + u_x^{ABC} - u_x^{AC} u_x^{ABC}}$

2. $d_x^{AC,ABC} = \dfrac{d_x^{AC} u_x^{ABC} + d_x^{ABC} u_x^{AC}}{u_x^{AC} + u_x^{ABC} - u_x^{AC} u_x^{ABC}}$

3. $u_x^{AC,ABC} = \dfrac{u_x^{AC} u_x^{ABC}}{u_x^{AC} + u_x^{ABC} - u_x^{AC} u_x^{ABC}}$

4. $a_x^{AC,ABC} = \dfrac{a_x^{ABC} u_x^{AC} + a_x^{AC} u_x^{ABC} - \left(a_x^{AC} + a_x^{ABC}\right) u_x^{AC} u_x^{ABC}}{u_x^{AC} + u_x^{ABC} - 2u_x^{AC} u_x^{ABC}}$

Where $b_x^{AC,ABC}$ gives Alice's belief in Carl's trustworthiness as a seller combined with the discounted recommendation about Bob's belief in Carl's trustworthiness, and uncertainty is taken into account; $d_x^{AC,ABC}$ gives Alice's disbelief in Carl's trustworthiness as a seller combined with the discounted

recommendation about Bob's disbelief in Carl's trustworthiness, and uncertainty is taken into account; $u_x^{AC,ABC}$ gives Alice's uncertainty in Carl's trustworthiness as a seller combined with the discounted recommendation about Bob's uncertainty in Carl's trustworthiness; and $a_x^{AC,ABC}$ gives the relative atomicity of the combined opinions.

Alice's overall trust in Carl, based on her opinion combined with Bob's recommended opinion, is thus $\omega_x^{AC,ABC}$ = {0.76, 0.16, 0.08, 0.5}, which gives a probability expectation $E(\omega_x^{AC,ABC})$ = 80% that Carl will cooperate as a seller. Note that the effect of combining opinions, i.e., incorporating more information into the trust assessment, decreases uncertainty and increases trust, as would intuitively be expected.

Measuring Trust as a Discrete Representation

Because it is difficult for humans to express trust in terms of numbers, people typically prefer fuzzy terms such as 'completely trusted', 'somewhat trusted', and 'not at all trusted' to represent trust. This is captured by systems that use discrete representations of trust measures. For example, the Abdul-Rahman-Hailes trust model (Abdul-Rahman and Hailes 1997) expresses a trust measure for direct functional trust as *t(a, c, td)* where *t* is the trust that a trust origin has in trust target *a* for trust purpose, or context, *c*, to the degree *td*. The trust degree $td \in \{vt, t, u, vu\}$ where *vt* represents very trusted, *t* represents trusted, *u* represents untrusted, and *vu* represents very untrusted. Observation outcomes are accumulated and stored according to grade of outcome, i.e., $s = (vg, g, b, vb)$, where *vg* represents a very good outcome, *g* represents a good outcome, *b* represents a bad outcome, and *vb* represents a very bad outcome. These outcomes correspond to the discrete trust measure types. In order to obtain the trust measure to be used for decision-making, an agent consults the experience accumulator to determine which grade of outcome is most prevalent. If this process, i.e., max(*s*), returns more than one value, then *td* is assigned an uncertainty value according to Table 1 (in which "?" indicates 0 or one other value).

Table 1: Uncertainty in the Abdul-Rahman-Hailes trust model

Experience	*td*	Meaning
$vg \wedge g \wedge ?$	u^+	Mostly good and some bad.
$vb \wedge b \wedge ?$	u^-	Mostly bad and some good.
All other combinations	u^0	Equal amounts of good and bad.

Furthermore, to express trust in a trust target to give recommendations about other entities, recommender trust is represented as *rt(b, c, rtd)* where recommendation trust *rt* is a measure of a trust origin's trust in trust target *b* for giving recommendations in context *c* to recommendation trust degree *rtd*. The value of *rtd* indicates the semantic distance between the recommendation and the trust

origin's own observations of the recommender's trustworthiness, i.e., how close the recommender's observations are to the trust origin's own observations. The recommender's perception of 'very trustworthy' may only equate to what the trust origin perceives to be 'trustworthy', and thus when *b* makes a recommendation of 'very trustworthy', rtd is applied to the recommendation to scale it to 'trustworthy'. The authors apply a weighting system to provide the scale for discounting recommendations according to semantic distance. Recommendations are combined by obtaining the *rtd* of all known recommenders of a trust target, adjusting each recommendation according to semantic distance, summing the adjusted recommendations according to grade of outcome, and applying the max(*s*) and uncertainty processes as above. Finally, it has been shown that discrete trust measures can be mapped from numerical values (Jøsang and Pope 2005), thus allowing for the mathematical calculation of trust, e.g., by subjective logic, but outputting a discrete trust measure that is more intuitive for humans to understand.

Formation and Evolution of Trust Measures

Because trust is subjective, a given trust measure, whether discrete or continuous, may not be formed, or evolved in the same way by every entity, giving rise to divergent policy specification by trust origins for, e.g., the way in which recommendations are assessed and trust evidence is combined. Furthermore, subjectivity affects the way in which trust measures are exploited. For example, trust can have threshold values (Marsh 1994), which vary between people and situations, i.e., a trust origin will have a positive threshold and when a trust measure for a trust target for a given trust purpose is above the threshold, it will be trusted; conversely, when a trust measure falls below a negative threshold value, a trust target will not be trusted. Ultimately, a trust measure encapsulates the properties and characteristics of trust such that a trust origin may use the measure to make a trusting decision with regard to the context in which an interaction is taking place and the risk associated with that interaction.

Thus, measuring trust encompasses the trust formation and evolution processes, and the trust measure resulting from these processes is used for trust exploitation, or decision making based on trust. In measuring trust, there are several requirements that trust measures should satisfy (Jøsang, Gray et al. 2006). First, it is essential that a trust value is meaningful to and usable by trust origins in the case where recommendations are assessed in addition to observations. Otherwise, if trust is subjectively measured by independent methods, the value becomes meaningless and unusable to parties wishing to rely on recommendations to make a trusting decision. In this regard, the trust purpose and its variants, functional and referral, must be explicitly defined. Second, the context of an interaction must be captured by the trust measure, again signifying the importance of correctly specifying the trust purpose. As part of context, time should be captured, not only to demonstrate how trust is evolving but also in order to enable interaction partners to assess trust based on, e.g. the most recent trust value available. Additionally, the confidence of the trust measure is necessary. For instance, the weakening of trust through long recommendation chains should result in a discounting of the trust measure, or

reduced confidence in the trust measure. Moreover, in order to derive trust measures from recommendation chains, there should be explicit methods for combining trust measures. Finally, policies can be specified that appropriately capture subjectivity with regard to the trust formation, evolution, and exploitation processes such that a trust measure can be used to determine whether or not to accept the risk of an interaction.

2.1.4 A Summary of Trust Properties

In addition to the characteristics of trust extracted from the definitions in the previous section, we find the trust properties of diversity, subjectivity, asymmetry, context, transitivity, and measure key to building a unified model of human trust. These properties convey the following significant factors. First, a trusting relationship can be broken down into trust origin, trust purpose, and trust target. A trust origin should be able to subjectively asses evidence in the form of observations, recommendations, and context to form, evolve, and exploit a trust measure to determine whether or not to assume the risk of interacting with a given trust target for a specific trust purpose. Next, a trust purpose should capture not only context, but also the functional and referral variants of trust purpose such that a form of transitivity can be implemented to correctly propagate and exploit trust measures via recommendations between community members. Additionally, the property of asymmetry is important in defining the trust purpose, as it provides directionality to the graph of trusting relationships and helps to capture the contextual notion of role when assessing trust for a given trust target. Fourth, mechanisms are needed so that recommended trust measures can be scaled appropriately and joined via parallel combination so as to increase the amount of evidence available for analysis in making a trusting decision, thus decreasing uncertainty. Finally, a trust measure should be obtainable with respect to the trust properties, such as, subjectivity of the trust origin in specifying how to form, evolve, and exploit trust; context, including role, time, purpose, and environmental factors; and transitivity.

2.1.5 Model of Trust

The diversity of trust definitions can lead to difficulty when trying to develop a general definition of trust. However, we have shown that it is possible to extract from the definitions several trust attributes, i.e., confidence/expectation, subjectivity, context, uncertainty, evidence, risk, reduction of complexity, and trust measure, as well as to highlight the properties of trust, i.e., trust origin, trust target, trust purpose, diversity dimensions, subjectivity, asymmetry, context, transitivity, and measurability. Therefore, we propose that it would be beneficial to create a unified model of trust that captures these elements such that:

1. Specification of confidence in outcome expectations is possible;

2. Diversity dimensions of trust are captured, i.e., trust origin (individual or entity), trust target (individual or entity), and trust purpose;

3. Subjective specification of trust formation, evolution, and exploitation processes is possible, since not all entities have the same perception of evidence. Subjectivity is based on an entity's disposition and belief system;

4. Evidence that is based on past behaviour can be directly observed or indirectly recommended and used to update trust both positively and negatively in a dynamic and non-monotonic manner;

5. Processes to collect and analyse evidence are explicit, i.e., the collection of evidence types: observations, recommendation of observations through transitive trust paths, context, and risk; and the assessment of evidence according to subjective interpretation and combination as well as contextual relevance;

6. Trust is context-dependent, and context can be captured, to include asymmetrical relationships (role), time, environmental factors, and environmental constraints;

7. Both agent- and context-specific risk can be captured and assessed;

8. A meaningful and usable measure of trust can be produced from subjective analysis of contextually relevant evidence such that it may be used by a trust origin to be exploited in propagating trust to the community via recommendations and to make trusting decisions to interact for a given trust purpose with a given trust target in light of associated risk; and

9. Complexity of decision-making is reduced in environments in which uncertainty and risk are present.

McKnight and Chervany (McKnight and Chervany 1996) have proposed an approach to capture the different aspects of trust in one unified trust model, illustrated in Figure 6, which shows how various human trust processes contribute to the outcome of trusting behaviour.

The McKnight and Chervany trust model unifies six trust processes, situational trust process, dispositional trust process, belief formation process, system trust process, trusting beliefs, and trusting intention, each of which contribute to an outcome of trusting behaviour. Trusting behaviour occurs when one party 'voluntarily depends on another party in a given situation with a feeling of relative security, even though negative consequences are possible' (McKnight and Chervany 1996). Thus, trusting behaviour describes the act of trusting and implies the acceptance of risk (Povey 1999). Each of the six constructs that contribute to trusting behaviour in the McKnight and Chervany model is described below.

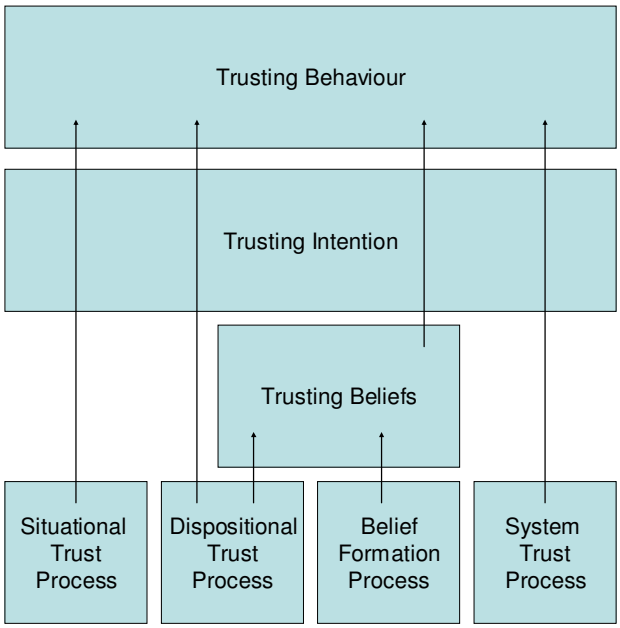

Figure 6: McKnight and Chervany model of trust

2.1.5.1 Situational Trust

The situational trust process occurs when an individual analyses his willingness to depend on a non-specific party in a given situation, i.e., situations in which an individual will trust another party, irrespective of the individual's beliefs about that party's trustworthiness, because the benefits of exhibiting trusting behaviour in these situations outweigh the possible negative outcomes. This process implicitly captures risk because if the risk of a negative outcome in a given situation is acceptable, a person may choose to make a trusting intention regardless of the other party with whom he is interacting.

For example, in a situation where a person is carrying heavy bags and it is raining, that person might always choose to hail a taxi. Though the person may or may not have specific beliefs about whether the taxi driver is a good driver, has not been drinking alcohol, or is the actual licensed driver of the taxi, the individual makes a decision to trust each time the situation arises. Thus, this trust construct also captures reasoning about the contextual facets of environment and role, i.e., given a particular context a person always makes the same trusting intention, regardless of the personal attributes of the other party involved in the interaction. Furthermore, an element of subjectivity is captured in this trust construct because individuals may asses risk differently from one another and therefore may employ subjective situational trust-based decision-making policies.

2.1.5.2 Dispositional Trust

Dispositional trust is a cross-situational and cross-personal trust analysis construct, and it contributes to the formation of a trusting intention in two ways. First, dispositional trust contributes to the formation of trusting beliefs in a general manner that, like the situational trust process, captures the contextual attribute of role, i.e., a person may be predisposed to believe that people will behave in a certain manner and is valid over a broad spectrum of situations. This value reflects whether an individual is optimistic or pessimistic in their approach to new situations. Alternatively, a person's disposition also contributes to his forming of a trusting intention when he believes that he will obtain a better outcome by trusting a specific person in a given situation. Person-specific evidence for that situation, in this case, is subjectively assessed. The dispositional trust process thus captures both the subjectivity and context aspects of trust, and, again, implies that some form of risk assessment takes place when forming an intention to trust.

2.1.5.3 System Trust

System trust represents the extent to which a principal believes that the proper impersonal structures are in place to enable him to interact successfully, regardless of the attributes of the other party on which he must depend. System trust reflects the fact that safeguards are in place to reduce the amount of risk to which an entity must be exposed. These safeguards may be in the form of regulations, guarantees, insurance, or stabilising intermediaries. Credit card companies, for example, safeguard transactions through codified processes such as authentication. Suppliers of goods and services accept credit cards for payment, trusting that proper structures are in place to ensure that payment in full and on time will be successful.

System trust captures another facet of context, i.e., environmental constraints. It is important to note that in an organisation in which sufficient safeguards are in place to mitigate all potential forms of risk, a decision to trust is no longer relevant because trust is unnecessary without risk.

2.1.5.4 Belief Formation

Belief formation is the process by which information and experience gathered from an individual's environment is evaluated to form new trusting beliefs about others. It is made up of two mechanisms: categorization mechanisms that organise information into categories, each with an equivalent response; and illusionary mechanisms, which are based on assumptions, emotions, and levels of confidence.

Categorization mechanisms include unit grouping, reputation, and stereotyping. These three methods enable one to assign a role to a new entity, i.e., to place an unknown entity into a grouping from which generalizations may be made. Unit grouping means that an individual, because of a newly-formed

relationship with another party, now perceives that party to be part of a grouping, e.g., a group in which members share common goals and/or beliefs, values, and assumptions. A second type of categorization is reputation, which may reflect professional competence, integrity, or trustworthiness. Reputation reflects what is known about an entity, based on the recorded results of interactions with other parties. Finally, the categorization process of stereotyping is fed mainly from general biases towards or against genres of people, e.g., according to broad levels of gender, nationality, and class. The resulting prejudices may cause immediate distrust between interacting parties. Each of these processes affect the way in which an individual forms trusting beliefs from gathered observations, recommendations, and environmental information. From each of these processes, generalizations may be gleaned in order to enable one to shape beliefs about another party's trustworthiness, honesty, competence, reliability, predictability, etc.

Categorization provides a rational and logical basis for determining beliefs. Human trust, however, is not developed solely on rational mechanisms, but also on illusionary beliefs, particularly in situations where information is unavailable and uncertainty is high. In the initial stages of an interaction, where one may not be aware of all of the risks involved, trust formation may be based on assumptions and overconfident judgment, wherein one forms a tentative opinion, seeks confirming evidence, and develops an over-inflated confidence in the new interaction partner once the slightest bit of confirming evidence is observed. This aspect of human trust assessment may lead to less than optimal decision-making, and indeed increase the risk of making a poor decision, but nevertheless is an influencing construct when operating in uncertain environments where complete information is not available or as a means to reduce complexity when an overwhelming amount of, perhaps contradicting, evidence is available.

Overall, the human belief formation trust construct provides guidelines, or trusting beliefs, with which to subjectively specify ways of shaping trust during trust formation and evolution.

2.1.5.5 Trusting Beliefs

Trusting beliefs are composed of cognitive and emotional components, and are described as the extent to which one believes that another party is trustworthy in a given situation, i.e., is willing and able to act in the trusting party's best interest, e.g., with benevolence, honesty, competence or predictability. Trusting beliefs are affected by the subjectivity of both the dispositional trust and belief formation processes, as well as by the context in which the beliefs are formed, and they shape the way a human makes a decision to trust, or trusting intention.

2.1.5.6 Trusting Intention

A trusting intention is formed when a trust origin is willing to depend on a trust target in a given situation, i.e., a given trust purpose, with a feeling of relative security, even though negative

consequences are a risk. The trusting intention construct captures the extent to which one party is willing to form such a dependence based on that party's disposition, his assessment of the situation and system, and the way in which he forms trusting beliefs. Trust is thus implicitly measured through a trust formation and evolution process leading to a value that can be used to make such a trusting intention. A trusting intention implies that an individual has made a decision based on the various risks and benefits of trusting, and shaped this decision through his subjective beliefs. When an individual proceeds to act on his trusting intention, trusting behaviour is exhibited, thus exploiting the trust assessment.

2.1.5.7 Summary

The trust characteristic of confidence in expected outcomes is captured by the trusting intention, i.e. a decision is made to interact that reflects some level of confidence in the outcome of the interaction. Furthermore, subjectivity is revealed in five of the trust processes: in the situational trust process, an individual subjectively perceives contextual combinations of situation and role in which he will always decide to trust, and these contextual combinations depend strongly on individual risk assessment; the dispositional trust process captures the overall disposition of an individual, i.e., very trusting, not trusting, etc., as well as the willingness of an individual, or trust origin, to form an intention to trust a trust target for a given trust purpose; the belief formation process in humans is highly subjective, as are the resultant trusting beliefs; and a trusting intention captures the subjectivity in the contributing constructs to form a trusting decision that is unique to the decision-maker. Third, the model addresses context in several ways: as environmental constraints that may affect a trusting decision, as system factors that may make a trusting decision unnecessary, as a shaper of beliefs over time, and as an important input to the dispositional trust process in which an individual is making a person-specific and context-specific trust-based decision. Evidentiary collection and analysis processes are not specified by the model, but the need for evidence and evidentiary analysis is implied in the descriptions of each of the constructs, as is a measure of this evidence implied by the formation of a trusting intention. Similarly, risk and uncertainty are not explicitly modelled, but implicit from the description of trusting behaviour, i.e., the act of trusting implies the acceptance of risk of uncertain outcomes. Likewise, while trust formation, evolution, and exploitation are not expressly modelled, they are reflected in the constructs of dispositional trust, belief formation, trusting beliefs, and trusting intention. Finally, through belief formation in particular, the McKnight and Chervany model illustrates the ways in which humans use trusting beliefs in order to simplify decision-making and reduce complexity.

McKnight and Chervany's trust model helps clear conceptual confusion by representing trust as a broad but coherent set of constructs that incorporate the major definitions from research to date. It draws together many aspects of trust to show how humans make trusting decisions as a function of available evidence, subjective beliefs, and context in the face of risk and uncertainty. We find that this model captures the characteristics and properties of trust that feed into the trust-based decision-

making process to provide an outcome of trusting behaviour, although at a very high level at which some trust attributes are implied rather than explicitly specified. A more expressive model might be defined such that the attributes and processes of human trust-based decision-making are made explicit and the output of a system based on such a model is a decision about whether or not to trust. In the next two sections we present the attempts by the computer security field to produce such a model.

2.2 Trust Models in Computer Security

Trust has not only been defined and modelled by researchers in the humanities fields. In fact, trust has long been used (Department of Defense 1985) in the computer security domain to signify competence and reliability of systems. In this arena, however, the trust attributes and properties identified earlier in the chapter are not incorporated, and trust is typically equated with authorisation and authentication (Grandison and Sloman 2000), e.g., the term 'trusted' refers to an authenticated digital key in a public-key system that authorises its holder to perform a given set of actions. This section provides an overview of the work in formal logics and trust management systems that has given rise to models of trust resulting from work in the computer security community.

2.2.1 Formal Logics and Trust Models for Authentication

The first attempts to formally describe computational trust were focused on authentication (Burrows, Abadi et al. 1989; Gong, Needham et al. 1990; Rangan 1992; Yahalom, Klein et al. 1993; Beth, Borcherding et al. 1994; Yahalom, Klein et al. 1994), and resulted in formal logics that can be used to analyse trust relationships and draw conclusions about design flaws in systems and correctness of data. The formal logics identified can specify protocol assumptions and interactions, and can be used to determine who is trustworthy, i.e., authenticated and authorised to interact, and which data belongs to whom. They are not well-suited as general computational trust models, however, as their application is for a very specific domain (Grandison and Sloman 2000).

For example, in one of the first approaches to formally describe trust, Burrows et al. (Burrows, Abadi et al. 1989) state that, while authentication would be straightforward in a sufficiently risk-free environment, such an environment cannot be assumed and therefore 'the style of precautions taken has caused it to be recognized for a long time that we are dealing with questions of belief, trust, and delegation' (Burrows, Abadi et al. 1989). In this regard, the authors propose a language to uniformly capture the steps followed in the authentication process between two entities, independently of the authentication protocol being used, such that a protocol may be analysed to determine whether design flaws exist. The result of an authentication process is that an entity is either trusted, i.e., authenticated, or not, and the formal logic serves mainly to assess protocol performance rather than to allow entities to form, evolve, and exploit trust for the purpose of decision-making.

Beth et al. (Beth, Borcherding et al. 1994) build on the work on formal logics, extending the formal representation of trust relationships proposed in Yahalom et al. (Yahalom, Klein et al. 1993; Yahalom, Klein et al. 1994), to specify trust parameters to allow for trusted communication in open networks, mainly for authentication, e.g., a trust origin, Alice, trusts a trust target, authentication server AS1, for the trust purpose of authenticating a digital signature.

Trust is assessed in Beth et al's work (Beth, Borcherding et al. 1994) as: $Ptrusts_x^{seq}QvalueV$, where P is the trust origin, Q is the trust target, e.g., an authentication server, x is the trust purpose (six trust purposes are specified: key generation, identification, keeping secrets, non-interference, clock synchronization, and performing algorithmic steps), *seq* is the recommendation path between P and Q, and V is the trust measure which is an estimation of the probability that Q behaves correctly when being trusted, e.g., $Ptrusts_x^{\varnothing}Qvalue0.55$ based on P's positive direct, i.e., not recommended, experiences with Q.

The trust measure is based on a count of positive experiences with Q that P knows about from evidence based on directly observing interactions with Q and recommendations about interactions with Q. To determine the trust measure value, experiences are evaluated monetarily (assessing cost and benefit in terms of ECU) adding the notion of utility to the trust model. A positive experience count is incremented in ECU when a positive interaction has been experienced, and a negative experience count is similarly incremented when a negative interaction has been experienced, and when positive experiences outweigh negative experiences to a given level, the trust target is trusted. This introduces the notion of 'degrees of trust' and trust thresholds, that is, that there is a maximum threshold value one would be willing to risk within a given trust relationship.

If a trust measure is recommended to a trust origin by a recommender about a trust target for a given trust purpose, it is assessed by the trust origin in terms of how well that recommender typically makes recommendations, which can be viewed as a seventh trust purpose. If a recommender typically makes recommendations that a trust origin finds valuable, then recommendations from that recommender will be assessed to form a trust measure for a trust target. If a trust origin's experiences with a recommender are more negative than positive, recommendations from that recommender will be excluded from the trust valuation process.

In the trust valuation process proposed by Beth et al., we begin to see the development of a trust model that comprises trust characteristics and properties. Confidence about the utility of expected outcomes of interacting with a trust target in a particular context is measured as a trust value. Trust is formed and evolved based on collected evidence of past behaviour that may be directly observed or recommended. Trust thresholds may be subjectively specified by a trust origin such that trust may be exploited when deciding whether or not to interact when faced with the risk of a negative outcome, i.e., a trust target is either trusted above a threshold or not. However, the notion of trust remains equated to reliability in authentication based on the assumption that all trusted entities display

consistent and predictable behaviour once they have been authenticated. This may not scale to a more complex system in which principal behaviour is more dynamic, uncertain, and potentially malicious.

Pretty Good Privacy (PGP) (Zimmerman 1995; Abdul-Rahman 1996) provides a trust model for another security domain. It uses public key cryptography to provide secure email interaction between parties. If a key, or certificate, is determined to be valid, that confirms that the key belongs to its purported owner and an email can be thus authenticated. The two main concepts in this environment are those of validity of certificates and trusting people for the trust purpose of validating other people's certificates.

A certificate authority (CA) is completely trusted to establish certificate validity by signing keys, and to introduce, or recommend, other people as validators of certificates. Establishing a line of trust with someone who is not explicitly trusted by a CA can be done according to three different trust models, the direct trust model, the hierarchical trust model, and the web of trust model. Direct trust is the simplest model in which entity A trusts that entity B's key is valid because A knows the key came from B. Hierarchical trust is analogous to a tree in which a leaf node's certificate validity is verified by tracing backwards from its certifier to other certifiers until a directly trusted certifier, i.e. the root CA, is found, as illustrated in Figure 7.

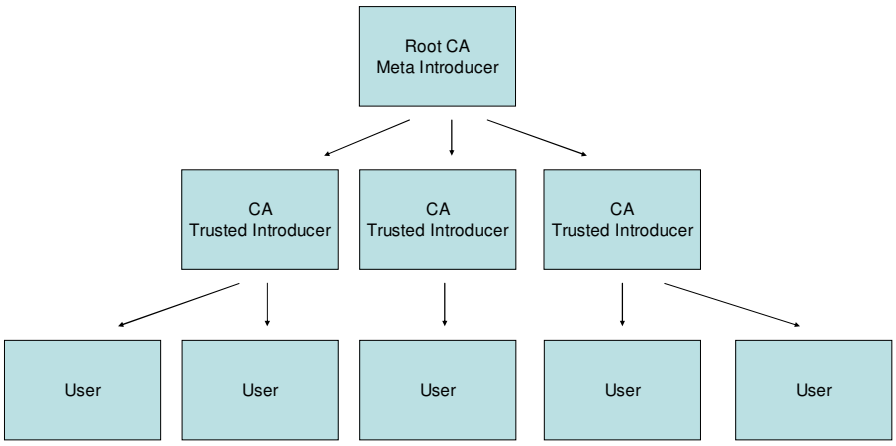

Figure 7: Hierarchical trust model for certificate validation

PGP is based on the third trust model, web of trust, which encompasses both direct trust and hierarchical trust as well as the notion that 'trust is in the eye of the beholder', i.e., a person may wish to subjectively assess the trustworthiness of another person for the purpose of recommending users. In the PGP environment, any user can act as a CA to validate another user's public key certificate. There is no central trusted authority. Individuals recommend, or digitally sign, each other's keys and progressively form a web of individual public keys linked by signatures to form a network of communities, including tight clusters of friends and bridges of inter-community links.

Each user stores evidence in the form of a public keyring on which copies of keys are kept along with a validity level, i.e., whether or not the user considers a key valid (if so, the user signs the key copy), and a level of trust placed on the key denoting to what extent the key owner is trusted to serve as an introducer of other keys. There are four discrete trust levels: the highest level is implicit trust in one's own key pair, the next highest level is complete trust, then marginal trust, and finally, untrusted. The actual meaning of the trust levels is not specified, and they are meant only as guidelines. Additionally, there are three validity levels, i.e., valid, marginally valid, and invalid. For entity A to recommend entity B as a trusted introducer, A must first have a valid key for B (signed by A or a trusted introducer) and then set a level of trust to which A feels that B is entitled. To establish a key as valid requires one complete trust signature, or two marginal trust signatures. For example, A's keyring contains B's key, which A has validated. A knows that B is highly reliable for validating other people's keys and A therefore assigns B's key complete trust, making B a CA. If B signs a third party's key, that third party's key will appear as valid on A's keyring. The element of time is captured as part of validity, i.e., a certifier can revoke his signature at anytime.

As in the model proposed by Beth et al., the PGP trust model unifies some trust characteristics and properties to allow entities to perform trust-based decision-making for securing email in the public key environment. A trust origin forms and evolves a trust measure, i.e., implicit, complete, marginal, or untrusted, for a trust target for the trust purpose of signing keys, i.e., authenticating validity of public key certificates. Evidence in the form of direct observations, i.e., direct validation of a certificate, and recommendations, i.e., signed certificates, is combined to determine an overall trust measure, although the combination method is not defined apart from noting that one completely trusted certificate or two marginally trusted certificates are sufficient recommendations of another key's validity. Trust is exploited when an entity subjectively decides whether or not to sign another entity's certificate. The web of trust model does assist in reducing complexity, inasmuch as any entity can be a trusted recommender rather than necessitating the discovery of recommendation paths to CAs outside of one's own public key domain. Risk is implied as the potential that an authenticated key is not actually valid, but risk assessment is not explicitly performed as part of the trust-based decision-making process.

2.2.2 Trust Management Systems

The trust management approach to distributed system security was developed because of the perceived inadequacy of traditional authorisation mechanisms for distributed systems (Blaze, Feigenbaum et al. 1996). Traditionally, every application implemented its own mechanisms for specifying access policy, checking compliance, and binding user authentication to authorisation to perform security-critical operations. This approach exposed the security of the application to high levels of risk because an application developer might make simple but subtle mistakes in the design and implementation of the system. The main aim in the development of trust management systems is to produce an off-the-shelf security module that can be integrated into any application, specifically as

a solution for access control in large distributed systems. The module allows each application to define application-specific policies and credentials.

Blaze et al. (Blaze, Feigenbaum et al. 1996) describe trust management as a 'unified approach to specifying and interpreting security policies, credentials, relationships which allow direct authorization of security-critical actions.' The notion of a 'resource' that a security system is designed to protect varies widely across different distributed systems. They recognised that the system should be able to define what a resource is within its local environment. The next step is to allow the system to define access and restriction policies that apply to its resources. Policies and credentials can be modified to reflect changes in the usage patterns of the system over time. Current trust management solutions include PolicyMaker, KeyNote, REFEREE, the IBM Trust Establishment Framework, and the SULTAN toolkit.

The solution proposed by Blaze et al. (Blaze, Feigenbaum et al. 1996; Blaze, Feigenbaum et al. 1998) in their trust management system, PolicyMaker, is to bind a set of specific keys directly to authorisation to perform a specific task, i.e., a credential. PolicyMaker takes as input a set of local policy statements, a collection of credentials, and a string describing a proposed trusted action. Acting as a query engine, it evaluates whether the credentials prove that a requested action complies with local policy, which is a trust assertion that is made by the local system and is unconditionally trusted by the system. The fundamental question that the trust management system asks is 'Does the set, *C*, of credentials prove that the request, *R*, complies with the local security policy, *P*?' The request, *R*, the local policy, *P*, and the set of credentials, *C*, are passed to a trust management engine, which processes the request and outputs an authorisation decision as to whether the request and credentials provided are valid according to the local policy. Moreover, PolicyMaker provides a general-purpose, application-independent algorithm for checking proof of compliance. Applications using the single general-purpose compliance checker in a distributed system can be confident that results from the checker are correct. This reduces the need to have each resource in a distributed system implement its own compliance checker.

KeyNote, the successor to PolicyMaker, addresses further design goals, i.e., easier integration for applications and standardisation (Blaze, Feigenbaum et al. 1999). KeyNote assigns more responsibility to the trust management engine, i.e., signature verification and use of a specific policy assertion language, rather than to the calling application, making it easier to integrate into applications. It requires that assertions (policies and credentials) be written in a specific language, so that processing by the compliance checker proceeds more smoothly. The application passes an 'action environment' to the KeyNote tool. This is similar to the PolicyMaker 'query' in that in contains three elements: the security policy, a list of credentials, and the proposed action. KeyNote's response to the action environment is an application-defined string. In the simplest case, the result would be 'authorised.' KeyNote, like other trust management engines, does not enforce policy directly - it only provides advice to applications that call it.

The Rule-Controlled Environment for Evaluation of Rules and Everything Else (REFEREE) is a trust management system that makes access control decisions related to web documents (Chu, Feigenbaum et al. 1997). Based on PolicyMaker, REFEREE is a query-engine that may be integrated into a host application to evaluate proposed action requests. It interprets trust policies and returns a tri-value (true, false, or unknown) and statement list. 'True' means that the requested action is approved because sufficient credentials were supplied, 'false' means that the requested action is denied, and 'unknown' means that the supplied credentials were insufficient to approve or deny the requested action. The statement list provides the context for the decision and a justification for the answer. All statements are 'two element s-expressions' in which the first item specifies the context of the statement and the second item specifies the statement's content. For example, the statement that Alice is untrustworthy in a REFEREE certification module would be ((“certification module”)(“Alice” (untrustworthy yes)).

The IBM Trust Establishment Framework (Herzberg, Mass et al. 2000) uses certificates to build trust for e-commerce from the grass roots up, i.e., certificate-issuing third parties are either known in advance or can provide sufficient certificates for a policy to consider them a trusted authority. Certificates can be issued by various users who vouch for an entity in a particular role, e.g,. for an entity's status as a seller, and the role conveys useful information about the subject (not necessarily its identity). These certificates are evaluated by IBM's role-based access control model which is comprised of a Trust Policy Language (TPL) and a Trust Establishment module. Local policy is specified in TPL, which maps unknown users to predefined business roles, e.g. for the role 'seller' there may be rules governing group membership for the seller group. After the Trust Establishment module has determined that an entity can be assigned to a certain role, this information is sent to another module that applies the access rights that are bound to that role. The authors state that their implementation may be used as an extension of a web server or as a separate server interfacing to applications.

The trust management solutions discussed thus far are closely tied to systems that implement access control or authentication (Grandison and Sloman 2000). These systems do not acknowledge that trust changes over time and have no mechanism for monitoring trust relationships to re-evaluate their constraints in light of new observations. Therefore, to monitor changing trust relationships, Grandison et al. propose the Simple Universal Logic-oriented Trust Analysis Notation (SULTAN) trust management system, a computational framework designed to facilitate 'the activity of collecting, codifying, analyzing, and presenting evidence related to competence, honesty, security, or dependability with the purpose of making assessments and decisions regarding trusting relationships for Internet applications' such as the management of trust relationships relating to employee data access and client service provision for an e-commerce vendor (Grandison and Sloman 2003). The SULTAN trust management model is comprised of four components, a specification editor, an analysis tool, a risk service, and a monitoring service.

The specification editor is used by a system administrator to define trust relationships in terms of the parties involved and the interaction context, or trust purpose. A trust statement has the following

format: $TrustPolicyName : trust(Tr, Te, As, L) \leftarrow Cs$, where *TrustPolicyName* is the unique name for the assertion that a trust origin, *Tr*, trusts a trust target, *Te*, for a trust purpose, i.e., to perform a set of actions, *As*, at trust level, *L*, if constraint(s), *Cs*, is true. A recommendation statement has the following format: $RecPolicyName : recommend(Rr, \mathrm{Re}, As, L) \leftarrow Cs$, where *RecPolicyName* is the unique name for the assertion that a recommender, *Rr*, recommends a recommendee, *Re*, for a trust purpose, i.e., to perform a set of actions, *As*, at a level of confidence in the recommender, *L*, if constraint(s), *Cs*, is true. Trust statements and recommendations are expressed in the SULTAN policy language.

The analysis tool allows the system administrator to perform simulations and property analysis to determine if specified properties hold on trust and recommendation statements or if there are conflicts between statements.

The risk service retrieves risk information and calculates risk. Risk is a measure of the probability of a transaction failing. For example, if there is high risk associated with a service provider, then trust in that entity will be lower. Risk is also related to transaction value, e.g., a client may trust a supplier who is considered a high risk when services supplied cost only $10, but may not trust a medium risk supplier when purchasing goods for $10,000. Risk is calculated based on the following elements: a list of common risks, e.g., refusal to produce goods, service failure, information theft, and fraud; the probability of each risk occurring; a list of action dependencies; a list of trust origins and their maximum allowable losses/risk thresholds; and a list of system resources.

The monitoring service monitors interactions and updates risk, experience, and system state information to an information store. For example, when a customer makes a payment to an e-commerce vendor, the customer's balance is updated by the monitoring service.

This system incorporates two concepts notably lacking from the trust management systems previously discussed, i.e., the elements of capturing trust dynamics through observation monitoring, and the element of risk information retrieval, which facilitates the evaluation of risk in trust-based decision-making. SULTAN provides limited support for decision-making in that it allows the specification of simple authorisation policies which query the trust management system for access control based on trust levels. However, it has been found (Dulay, Lupu et al. 2005) that the system is too heavyweight for automated trust decision support as SULTAN's emphasis remains on trust specification analysis rather than trust-based decision-making. Moreover, while Grandison et al. differentiate between trust and authorisation, saying 'to state that you trust someone to do something and to say that you allow someone to do something in a given situation are two different things. Trust is a statement of belief. Access control is a statement of what is permitted (or not permitted)', SULTAN is primarily a means of specifying and analysing a system administrator's beliefs about access control for system users.

2.2.3 A Summary of Trust in Formal Logics and Trust Management Systems

Overall, it has been found (Abdul-Rahman and Hailes 2000) that much of the work in trust-based security (Burrows, Abadi et al. 1989; Gong, Needham et al. 1990; Rangan 1992; Yahalom, Klein et al. 1993; Beth, Borcherding et al. 1994; Yahalom, Klein et al. 1994; Zimmerman 1995; Maurer 1996; Kohlas and Maurer 2000) has been in the area of formal logics which solely provide a notation with which to describe the process of authentication and trust models for authentication that are too application-specific to be suitable as general computational trust models. Moreover, none of these models gives an explicit definition of what it means to be trusted, i.e., '*why* is this user trusted to execute this action?', rather than simply '*is* this user authenticated or authorised?', assuming that an intuitive notion of trust is universally understood.

This lack of definition leaves the models open to subjective interpretations and incompatibility issues, as well as making it difficult to model complex trust relationships. An application developer is given the responsibility of checking the conditions of the establishment of a trust relationship which is assumed to be monotonic, which does not allow for the dynamic nature of trust which evolves over time as new evidence becomes available through observation and recommendation. Nor do these models incorporate other key decision-making factors such as risk, apart from the SULTAN system which introduces limited risk assessment component to trust management. Overall, the solutions focus on access control decisions rather than a general trust-based analysis.

2.3 Human Trust Models

It has been stated (Jøsang 1996) that security represents the idealistic side of formal modelling, design, and development of systems, showing how we would like systems to behave in theory, while, in practice, trust is not idealistic but realistic, in that no formal model can perfectly capture human behaviour and errors may occur regardless of accuracy of system design, because knowledge about other humans is imperfect. In this regard, the way in which trust has traditionally been modelled by researchers in the computer security field has primarily focussed on improving the modelling, design, and development of a very specific domain, i.e., authentication and authorisation, and does not incorporate the characteristics of human trust-based decision-making. Moreover, the results of authentication and authorisation processes are not trusting decisions, but merely statements of who or what is permitted (or not permitted) access or action.

Recently, however, research has produced models for secure decision-making that involve explicit representations of human trust characteristics. The 'human trust models' have been developed while taking into account the very attributes and properties of trust discussed earlier in the chapter, and they attempt to be expressive enough to provide trust-based decision-making capability in domains in which uncertainty and risk are prevalent. The main body of work in this area is in the models of Jøsang, Marsh, Abdul-Rahman and Hailes, and the Secure Environments for Collaboration among Ubiquitous Roaming Entities (SECURE) project, and each of these models is described below.

2.3.1 Jøsang's Trust Model

Jøsang presents a trust model (Jøsang 1996; Jøsang 1999) for distributed systems, stating that a distributed system is not unlike a social network, wherein trust in human agents is evaluated with regard to honesty and trust in systems is evaluated with regard to security, and where malicious behaviour is possible. Because a distributed system involves both human and system entities, it becomes most advantageous to interact with those that are most honest and most secure, because trustworthy entities minimize exposure to risk. Thus, trust is defined as a belief that a system is believed to be secure and a human is believed to be honest.

Jøsang identifies three issues that arise when building trust models: it is important to understand the concept of trust as a human phenomenon, it is difficult to extract real-world characteristics to use as parameters in a trust model, and it is difficult to integrate these parameters into formal models in order to be able to optimise system performance and quality of service. In this regard, attributes of trust are characterised as follows. First, trust is a positive concept that expresses a level to which one expects desired results when relying on another party. Next, if there is no malicious behaviour in a system, there is no need for trust in decision-making. Third, a trusting relationship requires at least two parties, a trust origin and a trust target. If the trust target is a system, trust is an assessment of how resistant it is to malicious manipulation by human agents, and if the trust target is a human-like entity, trust is an assessment of belief that it will behave without malicious intent.

Jøsang uses belief theory and subjective logic to model and operate on trust. Belief expresses an expectation of how an entity will behave or perform based on information about past experience, knowledge about the entity's nature, recommendations from other sources, and disposition of the trust origin. Recall that this information is captured in an opinion that one entity, *A*, has about another entity, *B*, for a trust purpose, *x*, i.e., $\omega_x^{AB} = (b, d, u, a)$ where ω_x^{AB} represents trust origin *A*'s trust in trust target *B* for trust purpose *x*, *b* represents belief in *B*'s trustworthiness, *d*, represents disbelief, *u* represents uncertainty, and *a* represents relative atomicity; and where $b + d + u = 1$ and $\{b, d, u\} \in [0,1]$. A trust measure, $E(\omega_x^{AB})$, or the expected probability of ω_x^{AB}, can be used by a trust origin to make a trust-based decision whether or not to interact with a trust target for a given trust purpose, *x*.

Decision-making in Jøsang's trust model incorporates the trust measure, $E(\omega_x^{AB})$, and the notion of risk by assessing utility. For example, in the case of transacting in a virtual market, payment for goods and services is disconnected from delivery of the goods and services. In this case, each transaction has two possible outcomes, depending on whether a trust target cooperates or defects. If the trust target is operating in the role of seller, cooperation occurs when he delivers goods for payment and defection occurs when he receives payment but does not deliver goods. If the trust target is operating in the role of buyer, cooperation occurs when he pays for goods delivered and defection occurs when he receives goods but does not make payment. A trust origin can attach utilities to the possible outcomes of a transaction, i.e., monetary units are gained if the trust target cooperates,

$U_+^A(x)$, and lost if the trust target defects, $U_-^A(x)$, and then a trust-based decision can be made according to the outcome of the formula $U_B^A(x) = E(\omega_x^{AB})(U_+^A(x) - U_-^A(x)) + -U_-^A(x)$.

For instance, if Alice's opinion about Bob's trustworthiness as a seller of goods in an online marketplace is ω_x^{AB} = {0.76, 0.16, 0.08, 0.5}, which gives a probability expectation $E(\omega_x^{AB})$ = 80% that Bob will cooperate in a future transaction, and if Alice assigns utilities to the outcomes of her next potential transaction with Bob as $U_+^A(x)$ = \$25 and $U_-^A(x)$ = -\$75, $U_B^A(x)$ = \$5. If Alice's policy is, e.g., to interact whenever utility is determined to be greater than 0, in this example the small expected gain from trusting Bob, i.e., \$5, indicates a rational choice for Alice to engage in the interaction with Bob although the risk is relatively high.

The trust model enables one to reason about the complexities inherent in a system involving both human and rational entities. Overall, this model captures key trust attributes and properties, i.e., confidence of expected outcomes, subjectivity of beliefs, context according to role and environment, uncertainty, evidence in the form of direct observations and indirect recommendations, evidentiary analysis through discounting and consensus operations, diversity dimensions, trust measurement through opinions translated into expected probability, and ways in which to form and evolve trust. Furthermore, the model provides a means for agents to reduce complexity by relying upon a sound formal process for decision-making, as well as to include the notion of risk into a trust exploitation process. However, it leaves for future work to 'find principles to correctly assess and extract trust as a parameter from the real world' as well as the implementation of the formal model in a real-world system.

2.3.2 Marsh's Trust Model

Stephen Marsh's seminal work on trust (Marsh 1994) is founded in the social sciences, from which he extracts real world parameters for evaluating trust from properties inherent in social networks. Marsh extracts trust properties from the disciplines of psychology, philosophy, sociology, game theory, and multi-agent systems, as well as incorporating concepts of risk and utility into his trust formalism to provide a tool for precise trust-based decision-making that is implementable in artificial agents.

Marsh sees trust as a means for understanding and adapting to complex environments, such that it can be used as a tool for evaluating prior experience of the behaviour of others. From the literature, then, Marsh delineates a formalism with the following elements. First, agents and situations form the basis of the model. An agent *x* has knowledge, $K_x(y)$, of another agent y if they have met at some time and if *x* can remember the interaction. Knowledge $K_x(y)$ is evaluated as a boolean variable, to either 0 or 1, meaning that an agent either knows another agent or does not. Marsh then provides a typology for trust, breaking trust down into three elements, basic, general, and situational.

Basic trust, T_x, represents the dispositional trust of x and depends on all of the experiences that have shaped x in the past, and is evaluated over the range [-1, +1), where negative values represent distrust and positive values represent trust, and where blind trust, i.e., $T_x = 1$, is not acceptable because 'blind trust is not trust, as it it does not involve thought and consideration of things' (Marsh 1994). Basic trust is not a measure an agent has in another agent, situation, or environment, but rather a representation of an agent's overall beliefs about the world. For example, if x is a seller of goods in an online environment and has only ever experienced interactions with buyers who pay on time and in full, then x may be dispositionally predisposed to believe that the world, i.e. the online selling environment, is a trustworthy place where buyers are to be trusted, and T_x will be high.

General trust, $T_x(y)$, represents the trust that a trust origin, x, has in a trust target, y, irrespective of situation. This is also evaluated over the range [-1, +1), although Marsh notes that $T_x(y) = 0$ equates to unknown trust rather than distrust, i.e., agents x and y have met for the first time and do not have any previous experience of each other to evaluate, or agent x has evaluated y's actions over time and positive and negative experiences have drawn the general trust measure for y to 0.

Situational trust introduces context into the trust measure, $T_x(y, \alpha)$, which represents the trust that a trust origin, x, has in a trust target, y, for context, α. Situational trust is also evaluated over the range [-1, +1). Marsh finds situational trust to be most important when an agent is interacting in a situation where it must determine whether or not to cooperate with another agent. It is assumed that in this scenario, an agent will cooperate if situational trust is above a certain threshold. In order to estimate situational trust, x considers the utility of the situation, $U_x(\alpha)$, with values over the interval [-1, +1]; the importance an agent assigns to the situation, $I_x(\alpha)$, with values over the interval [0, 1]; and trust in agent y based on past experience of interacting with y in all situations, giving $T_x(y,\alpha) = U_x(\alpha) \times I_x(\alpha) \times T_x(y)$. Note that the concept of importance is similar to the notion of situational trust in the McKnight and Chervany model, wherein an agent subjectively assesses not only expected utility but also the environmental constraints in place when making a decision. The estimation of general trust, $T_x(y)$, in the situational trust calculation takes into account all data relevant with respect to situations in which the context, γ, is similar or identical to the current context, α. A method for context-mapping, i.e., determining the similarity of situations, is not provided. Moreover, the representation of distrust and negative utility is problematic, because the multiplication of two negative numbers produces a positive outcome that is counterintuitive.

Marsh defines the notion of cooperation threshold as the optimal threshold of probability that an agent will trust another enough to engage in some action in a certain situation. This threshold varies according to agent disposition, i.e., not all trust origins calculate and exploit trust in the same way, and circumstances, i.e., the perceived utility varies across situations. The cooperation threshold is given as:

$$cooperation\_threshold_x(\alpha) = \frac{perceived\_risk_x(\alpha)}{perceived\_competence_x(y,\alpha) + T_x(y)} \times I_x(\alpha)$$

Where $perceived\_risk_x(\alpha)$ represents the risk of cooperation assessed by the trust target, x, for the utility of the situation, or trust purpose, α; $perceived\_competence_x(y, \alpha)$ represents the situational trust x has in y for α, $T_x(y)$ represents x's general trust in y; and $I_x(\alpha)$ represents the importance that x assigns to interacting in the situation. Trust plays a role in the mediation of the cooperation threshold such that very low trust ensures that cooperation is less likely to occur than if trust were high. A temporal index is also employed such that each trust element can be assessed in terms of timeframe.

The trust model is implemented in agents to perform simple experiments in a testbed using the Iterated Prisoner's Dilemma to show that agents correctly mimic human trust-based decision-making behaviour, i.e., cooperate when trust is high and defect when trust is low.

Marsh extracts most aspects of social trust from the real world, i.e., trust as confidence in expected outcome of interacting, agent disposition in forming subjective beliefs, context according to time and environment, evidence in the form of direct observations, evidentiary analysis assessing knowledge, diversity dimensions, trust measurement as probability, risk, and ways in which to form, evolve trust, and exploit trust, and integrates these parameters into a formal model, the implementation of which is shown to use trust to optimise agent systems,. However, critics suggest (Abdul-Rahman and Hailes 1997) the large number of variables in Marsh's model leads to it being unduly large and complex. Moreover, we have seen that probability is not an appropriate way to represent trust, as uncertainty cannot be captured and multiplication of fractional numbers leads to counterintuitive results. Additionally, the notion of recommending evidence is not taken into account, nor is the concept of role.

2.3.3 Abdul-Rahman & Hailes' Trust Model

Alfarez Abdul-Rahman and Stephen Hailes (Abdul-Rahman and Hailes 1997; Abdul-Rahman and Hailes 2000) describe an effective practical trust model for virtual environments based on the human notion of trust. This work incorporates dual aims: first, to assist users in identifying trustworthy entities within a virtual community, and second, to give artificial autonomous agents the ability to reason about trust.

Using Gambetta's definition of trust (Gambetta 2000) as a basis, the authors integrate the following real-world elements: notion of interpersonal trust, discrete degrees of belief associated with a trust range from complete trust to complete distrust, reputation, and context. Interpersonal trust in this case refers to the trust one agent assigns to another in a given context. Thus, it is both trust target- and trust purpose-specific. Reputation is defined as 'expectation about an agent's behaviour based on information about or observations of its past behaviour,' and includes personal opinions as well as the opinions of others. Context is left undefined and up to the user to customise as per application requirements. The model is illustrated in Figure 8.

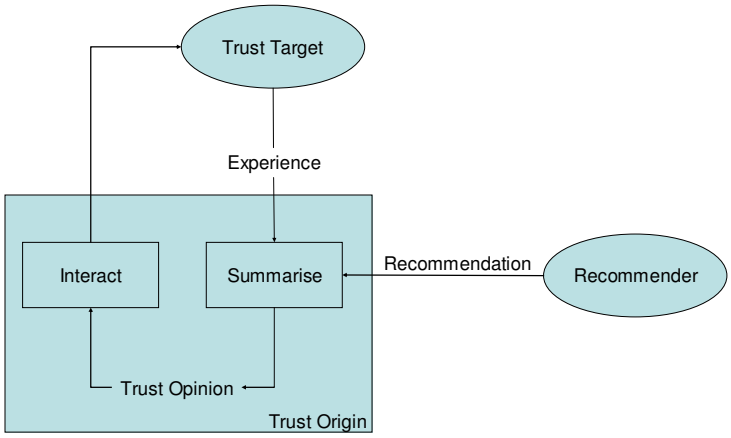

Figure 8: Abdul-Rahman Hailes trust model

The output of the model is an interpersonal trust measure based on prior experiences with a given agent in a given context and/or direct recommendations from other sources. An experience may either be direct, i.e., an agent's belief based on its own experiences with another agent, or recommended, i.e., reliance on a third party's belief about its experiences with another agent. Experiential evidence is stored in four different record sets: the set of direct trust experiences, the set of recommended trust experiences, the set of known contexts, and the set of agents with which previous interaction has occurred. These sets are updated incrementally by type counters, and a method for discounting recommendations based on 'semantic distance,' i.e., how similarly another agent grades experiences in the given context, is provided.

While the authors incorporate some real-world trust characteristics in building their model, the model in fact explicitly excludes the notions of system trust and dispositional trust, focusing solely on interpersonal trust. Without integrating the other fundamental aspects of human trust, i.e., disposition and system trust, this model is not a true representation of the human trust-based decision-making process. Moreover, although the authors establish a trust metric based on degrees of belief, they do not provide a basis for determining what, for example, 'very trustworthy' means in relation to the other degrees. This is left for the human user to approximate and therefore cannot be automated. Finally, the authors do not describe the actual decision-making process itself, i.e., trust exploitation, once a trust level has been calculated, leaving it up to the user to determine interaction policies based on his own disposition, the system in which he is interacting, and the risk of engaging in interaction.

2.3.4 Secure Environments for Collaboration among Ubiquitous Roaming Entities

The Secure Environments for Collaboration among Ubiquitous Roaming Entities (SECURE) project has developed a model that captures the key attributes and properties of human trust (Cahill, Shand et al. 2003). The SECURE model of trust was developed with explicit consideration of the results of

research into human trust in the humanity disciplines. Thus, the properties that are identified as being essential to a computational trust model overlap with those trust characteristics extracted from research into human trust. First, trust is inherently linked to risk, and there is no reason to trust if there is no risk involved in a given interaction. Second, trust is subjective, meaning that each entity makes its own decision to trust or not to trust. Moreover, it means that if trust is propagated throughout a community of diverse entities, e.g., by way of recommendations or reputation, a notion of recommendation integrity is required, i.e., in addition to subjectively assessing an entity's trustworthiness for a given task, one must be able to assess an entity's ability to recommend another entity. Finally, trust depends on the situation, e.g., context, environment, or community, in which it is established. Context adds a notion of relevance of evidence to a decision being made within a given context, including the parameters of role, time, environmental factors, and environmental constraints. Overall, the SECURE model adheres closely to the McKnight and Chervany model of human trust, incorporating elements of situation, disposition, system constraints and assurances, and belief formation into a process that results in a trusting decision from which trusting behaviour can be exhibited.

As illustrated in Figure 9, the SECURE trust model is comprised of several components, i.e., entity recognition, trust calculation, risk assessment, evidence management, and decision-making, each of which contributes to the forming of a trust-based decision.

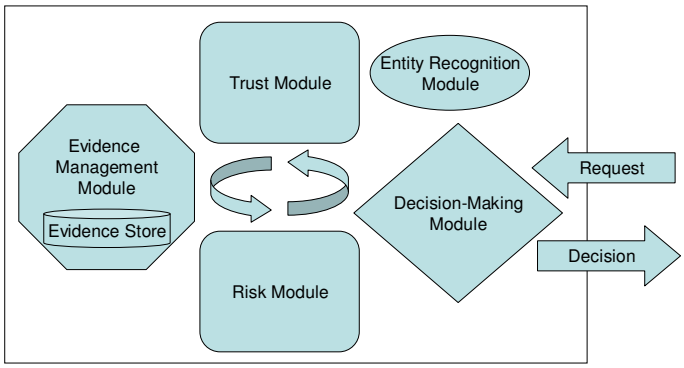

Figure 9: SECURE trust model

The entity recognition module is proposed as a generic replacement for authentication. It supposes that the 'ability to reliably recognise another entity is sufficient to establish trust in that entity based on past experiences' (Cahill, Shand et al. 2003). This component allows for the integration of other recognition schemes, e.g., existing password-based authentication mechanisms.

The trust module captures the processes by which trust is formed, evolved, and exploited. A formal model of trust (Nielsen, Carbone et al. 2004) uses a mathematical structure called an event structure, which captures the events which occur during interactions between two entities and classifies the outcomes of events. Evidence is classified as supporting (s) a hypothesis, inconclusive (i), or

contradicting (c) in terms of the possible outcomes of each interaction. An interaction history, derived from an event structure, indicates the number of occurrences of each type of outcome in a (s,i,c)-triple, similar to Jøsang's 'opinion', and this is used to compute a trust value in the interval of [0,1]. SECURE allows for the modification of triples received from other parties, depending on trust in the sending party, however no recommendation integrity mechanisms are defined. The trust model allows for the need of evidentiary assessment according to contextual relevance of evidence, although no specific methods to do this are defined. A single trust value is created from the combination of direct and indirect evidence, i.e., observations and recommendations. Observations and recommendations are evaluated independently so that a trust origin is able to keep its own direct evidence about a trust target separate from the general population's opinion about the trust target. SECURE incorporates a limited notion of reputation into the model, stating that 'an entity's reputation may be consulted in the absence of previous experience …[and] an entity may enter into collaboration with another entity, if that entity is generally trusted by all other entities in the current context' (Cahill, Nielsen et al. 2001).

A trust measure, or (s,i,c)-triple, is expressed as a value that captures combined evidence for and against a given outcome, as well as capturing uncertainty. The formal model provides two functions with which to form and evolve trust: the *effect* function is used to analyse whether a new piece of evidence supports, contradicts, or is inconclusive about the outcome of an interaction; and the *evaluation* function is used to compute the effects of updated configurations for interaction histories such that a trust value might be calculated. Trust is exploited when evidence from previous encounters with entities in certain situations is considered to make future decisions, thus incorporating a feedback loop between past and future interactions. Evidence collection and analysis is managed by an evidence manager, and observations and recommendations, as well as information about context and risk, are stored in an evidence store. Management of evidence allows for the reduction of complexity in a decision-making process in an environment in which an arbitrarily large number of observations and recommendations about a trust target may be accessed.

The risk component evaluates risk based on several factors, i.e., the trustworthiness of the trust target in the current context as expressed by the trust value, the overall risk of interacting in the current context, and the potential utility to the trust origin of interacting. By explicitly reasoning about risk, SECURE allows users to specify acceptable levels of risk for interactions with other entities in specific contexts. This reasoning process takes place in the decision-making component, and a trust- and risk-based decision is the final result output from the SECURE model.

The SECURE trust model and decision-making framework shows how trust 'can be made computationally tractable while retaining a reasonable connection with human and social notions of trust' (Dulay, Lupu et al. 2005), and is very suitable for use in applications in which decisions are made in a manner similar to the human decision-making process. Because SECURE was designed based on extensive research into human trust, SECURE integrates all of the key properties and attributes of human trust such that it can output a trust-based decision, i.e., such that a computational entity can exhibit trusting behaviour. Moreover, software has been developed that comprises the SECURE kernel with extensions for policy specification by application developers, and the SECURE

framework has been instantiated and validated in the collaborative spam filtering domain (Bryce, Cahill et al. 2005; Bryce, Seigneur et al. 2005).

2.3.5 A Summary of Human Trust Models

Models based on the human notion of trust, i.e., those developed by Jøsang, Marsh, Abdul-Rahman and Hailes, and the SECURE consortium, distinguish themselves from earlier attempts by the computer security community to use trust as a basis for computational decision making. The human trust models are founded on the attributes and properties of human trust, and they attempt to unify these trust characteristics into models that allow computational trust-based decision-making to occur in a similar manner to human trust-based processes.

Jøsang's trust model provides a means to form and evolve digital trust based on formal methods for analysing evidence and assessing trustworthiness in light of risk. This model, however, has not been applied to nor implemented in a real-world application domain. Marsh's model, implemented in a simple Iterated Prisoner's Dilemma system in which agents choose between two possible actions, i.e., cooperate or defect, captures most aspects of real-world trust as parameters that contribute to a final trust measure. The trust combination and measurement process, however, utilises probability which produces counterintuitive results in some cases and does not allow uncertainty to be captured. Moreover, Marsh does not provide for the notion of recommendation that humans often rely upon. The Abdul-Rahman-Hailes model incorporates some real-world trust parameters and provides a discrete trust measurement method that is more easily understandable than numeric measures, but this model excludes the aspects of disposition and system trust and does not provide a method for automating trust exploitation, i.e., decision-making that leads to trusting behaviour.

Only the SECURE trust model captures all of the identified trust attributes and properties. The SECURE model unifies all of the key trust characteristics and processes into one complete framework that has been implemented and validated in a real-world system such that computational trust can be used for decision-making in a manner analogous to the human process.

2.4 Trust and Reputation

In the last section, the properties of trust were explored and it was shown that a trust framework can be implemented such that trust-based decision-making can help an entity to analyze past experience, observations, recommendations, and context in terms of trust and risk to make decisions about interaction. Another important method for fostering trust between entities in virtual environments is reputation, i.e., a collection of recommendations which are aggregated to form a measure of an entity's character with regard to ability or reliability in interaction. Reputation has traditionally been used as an input to the human trust-based decision-making process when personal experience of

interacting with an individual or entity is lacking. Likewise, reputation might be used as an input to a computational trust-based decision-making process.

In this section, reputation is defined, and its properties are discussed. An overview of existing reputation management systems, both academic and commercial, is then presented. Finally, deficiencies with existing reputation systems are highlighted.

2.4.1 Defining Reputation

Reputation as a social concept is more easily defined than trust. While trust always includes a subjective element based on the disposition of the trust origin, reputation can be measured from an accumulation of independent experience-based opinions, i.e. recommendations, resulting in an evaluation of the overall character of an entity. The Merriam-Webster Dictionary (Merriam Webster Dictionary 2006) defines reputation as follows:

> *1 a : overall quality or character as seen or judged by people in general b : recognition by other people of some characteristic or ability <has the reputation of being clever>*
>
> *2 : a place in public esteem or regard : good name*

Before the Internet opened up the world to transactions unbounded by geography, local reputation was used to determine reliability of parties in interactions: vendors of goods and services provided certificates of reference from trusted third-party centralized ratings organizations, centralized evidence collection authorities such as the Better Business Bureau collected and circulated complaints, and past personal experiences relayed by word of mouth throughout clustered communities served as a basis for determining which grocer/doctor/lawyer/etc. was reliable or proficient in a given context. Collections of recommended evidence in the public domain, then, constituted a person's or organization's reputation.

Because the Internet links networks of local communities, reputation is no longer bounded by geography or special interest, and reputation management systems have been developed to provide an online equivalent to traditional means of collecting evidence, aggregating the recommended evidence into a reputation measure, and distributing reputation information. Reputation management entails recording an entity's actions and the opinions of others about those actions. These records can then be published in order to allow other people (or agents) to use the information to make informed decisions about whether to trust another party or not.

In this regard, Jøsang et al. define reputation as 'what is generally said or believed about a person's or thing's character or standing' (Jøsang, Keser et al. 2005), and they purport that reputation can be

quantitatively measured from information in an underlying social network. This information is visible to all members of the network. Therefore, in a situation where personally-observed evidence is lacking for a given entity, in some instances it is possible to seek out recommendations about that entity from one or many trusted third parties. Recommendations from third parties, when accumulated, constitute an entity's reputation that can be used in the absence of personal experience to make a decision about an entity's character.

Abdul-Rahman and Hailes extend this definition to include the notion of using reputation to search out entities that are characterised by evidence about a particular behaviour, stating that a 'reputation is an expectation about an agent's behaviour based on information about or observations of its past behaviour' (Abdul-Rahman and Hailes 1999).

Thus, for the purposes our work, a reputation is a collection of recommendations, i.e., personal observations recommended by one or more third parties, about an entity's past behaviour which are accumulated in such a way as to characterise an entity's nature with regard to ability or reliability in potential future interactions in a given context. If the accumulated recommendations are evidence of behaviour for a given trust purpose, then the resultant reputation characterising an entity's trustworthiness can be used as input to a trust-based decision-making system. Such a system could then perform trust-based reputation management such that a security decision as to whether or not to interact with a given entity might be provided to users.

In order to better understand reputation and its potential value to a trust-based decision-making process, we must first determine what kind of evidentiary information is required for reputation formation in the digital age; what mechanisms exist for the collection and distribution of evidence throughout a network; and what options are available to accumulate evidence into a reputation.

2.4.2 Evidence for Reputation

A reputation is based on the collection and evaluation of experiences over time. Recommendations, also called feedback or ratings in the literature, are records of these experiences and therefore provide the main evidence that contributes to the formation of a reputation. Components of a recommendation include content, explicitness, directness, source, and context. Moreover, recommendations can be collected and shared by centralised or distributed methods; and combined by reputation management systems to form a reputation measure that can be exploited by a system user as input to a decision-making process.

2.4.2.1 Recommendations

The Merriam-Webster Dictionary (Merriam Webster Dictionary 2006) defines recommendation as follows:

1 a : the act of recommending b : something (as a procedure) recommended

2 : something that recommends or expresses commendation

Wherein the term 'recommend' is defined as:

1 a : to present as worthy of acceptance or trial <recommended the medicine> b : to endorse as fit, worthy, or competent <recommends her for the position>

In the environment of trust-based decision-making, a recommendation may be defined as an experience-based trust assessment that provides evidence about the subject being recommended. In this domain, as distinct from the dictionary definitions, it is just as valid to recommend an observation of negative behaviour as it is to recommend a positive experience.

2.4.2.2 Content

The content of a recommendation may be qualitative, e.g., 'you would really love this restaurant because the service is great', or quantitative, e.g., a travel guide may rate a restaurant 4 out of 5 stars for a set of parameters such as menu, service, price, location, and cleanliness. Typically, the term recommendation has a positive connotation, i.e., if a service is recommended, it is a good service for some definition of 'good.' In fact, a recommendation may convey a range of evidence about its subject, positive or negative. Qualitative evidence is more difficult to analyze in an automated fashion by reputation management systems and, therefore, in these systems content parameters must be defined such that evidentiary analysis and measurement is possible.

2.4.2.3 Explicitness

A recommendation may be explicit, e.g., 'I suggest you go to this hairdresser', or implicit, e.g., Julia Roberts gets her hair done at a particular hairdresser when in New York thereby implicitly recommending the hairdresser to her fans. For automated reputation management, recommendations should be explicitly provided, e.g., a user rates another user on the correctness of behaviour in a transaction and submits this rating through a feedback mechanism within a reputation management system; although cases could be imagined in which evidence is implicitly observed, e.g., tracking the parties with whom a user repeatedly interacts.

2.4.2.4 Directness

A direct recommendation is based on direct encounters or observations, i.e., the recommender is passing on the measure of an experience he has experienced firsthand, while an indirect recommendation is based on second-hand information, e.g., Alice recommends Bob's recommendation of Carl. Indirect recommendations lead to transitivity issues that can be solved by assessing recommendation chain length and recommendation integrity in addition to analysing the observations being recommended.

2.4.2.5 Source

The source of a recommendation may be anonymous, pseudonymous, or identified. The source of the recommendation may be a local member of a community or a member of the general public. Personal experience, or an entity's observation of its own interaction with another entity, may also be captured in the form of recommendation, i.e., a self-recommendation (Abdul-Rahman and Hailes 1997), although typically personal experience is given a higher weight during the aggregation process. Moreover, confidence in the source is an important consideration, because a source may not necessarily provide accurate or useful information.

2.4.2.6 Context

A recommendation's context may be general, e.g., 'John is a good person.' However, a recommendation may be more usable when its context is more specific, e.g., including time, date, location, and service-type for a given interaction with John. Mui's typology of reputation (Mui 2003) shows that reputation is context-dependent because it is based on recommendations about a subject performing a given role at a certain date and time, in a given environment. Recommendations should be degraded as time passes so that current behaviour becomes more relevant than actions observed further in the past. Mui further states that existing commercial reputation systems provide reputation ratings without context, and that adding context, e.g., based on item value, might help mitigate bad behaviour.

Context is traditionally specified as part of the recommendation itself, e.g., Alice recommends Bob as a good driver. However, in online environments, context is typically appended to a recommendation by a system. For example, in the eBay reputation system, after a completed transaction, a user submits feedback evaluating the behaviour of the other party involved in the transaction. The eBay system appends contextual evidence to that feedback, i.e., the role and username of each party involved in the transaction, the time and date of the transaction, item category of the item involved in the transaction, and a link to a page containing the description of the item and the transaction details.

2.4.2.7 Recommendation Collection and Reputation Distribution

Traditionally, a reputation is built up by accumulation and distribution of recommendations for a given subject in a given context over time. This accumulation and distribution may occur in a centralized manner, e.g., a travel agency recording and circulating feedback for a particular resort, or in a distributed fashion, e.g., word-of-mouth propagating information from person to person. Evidence gathering and distribution for automated reputation management, similarly, can be centralized or distributed, depending on the requirements of the application in which reputation will be used. Thus, a protocol is necessary for collecting recommendations and distributing reputation. Jøsang et al. discuss the different properties of a centralized versus a distributed reputation system (Jøsang, Ismail et al. 2006).

Centralized reputation systems, such as those in commercial use today, adhere to two fundamental aspects, which are illustrated in Figure 10:

1. Centralized communication protocols that allow participating entities to provide and obtain evaluations about transaction partners to and from a central authority.

2. A reputation computation engine used by the central authority to aggregate individual transaction evaluations, i.e., recommendations, to derive overall reputation measures for each participant. Typically, a reputation measure for a given user and the corresponding list of recommendations about that user are made available to all participants in a community.

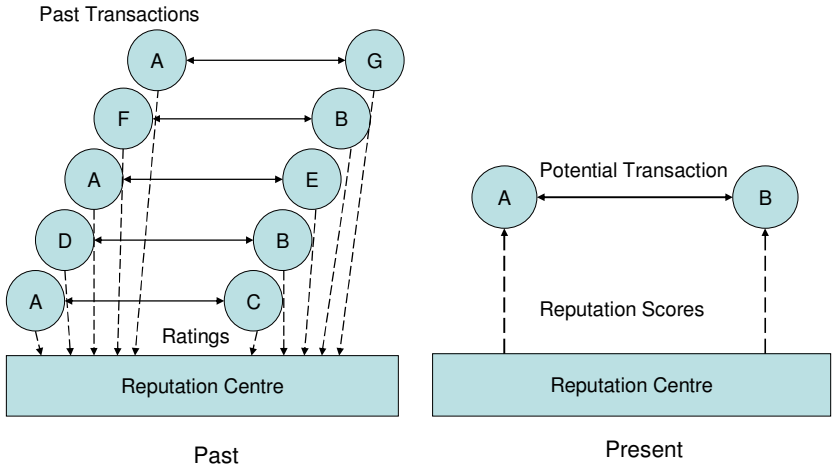

Figure 10: Evidence collection and distribution in a centralised reputation system

In centralized reputation systems, information about partner transactions is collected as recommendations from users of the system. The central authority collecting the evidence aggregates the information to provide a reputation measure for each member of the community. System users

can then obtain a reputation measure for a potential interaction partner when deciding whether or not to engage in a transaction with that entity. Commercial applications that employ centralized reputation systems include e-commerce providers such as online auctions, bookstores, etc.

Alternatively, distributed reputation systems rely on a different set of mechanisms to propagate reputation throughout a network:

1. A distributed communication protocol allowing entities to submit and obtain recommendations from other entities in the community.

2. A reputation computation method used by each individual entity to calculate reputation measures for entities with whom it has interacted based on recommendations.

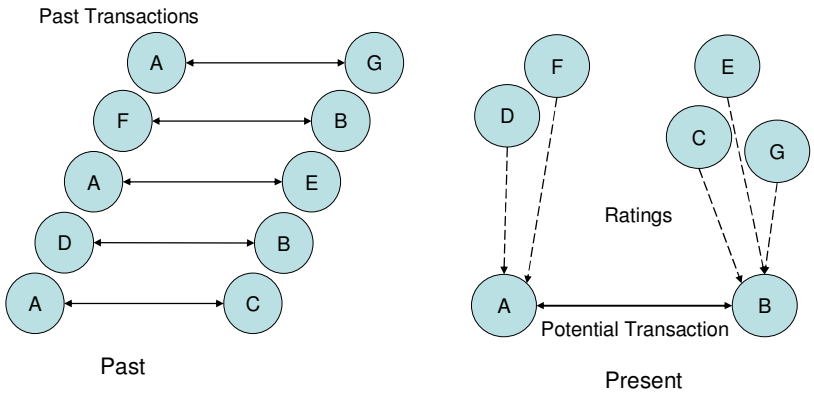

Figure 11: Evidence collection and distribution in a distributed reputation system

Distributed reputation systems, e.g., peer-to-peer (P2P) networks, function in the absence of a central authority for submitting or obtaining reputation information. Thus, there may be multiple evidence stores to which entities submit feedback about experiences and the information is available for requesting parties. A requesting entity must search out relevant reputation information from the distributed stores, as illustrated in Figure 11, and then calculate a reputation measure from the evidence.

2.4.2.8 Recommendation Combination for Reputation Formation

Recommendations are typically combined and used by two different types of systems, i.e., recommendation systems and reputation management systems. Recommendation systems are programs which attempt to predict items, e.g., films, music, news, webpages, etc., that a user may be interested in based on what other users have opined about the items. This type of system is different from a reputation management system, which records an entity's actions and the opinions of others about those actions, accumulates the opinions, or recommendations, and aggregates the

recommendations in such a way as to form a reputation measure that is presented to system users to assist them in making informed decisions about whether or not to interact with another. Thus, recommendation systems provide information that can be used to make a decision and reputation management systems provide information that can be used to make a trust-based decision. Therefore, the work in recommendation systems is outside the scope of this thesis.

In their survey of current reputation systems, Jøsang et al. (Jøsang, Ismail et al. 2006) describe the different types of evaluation mechanisms that can be embedded into a reputation engine. An evaluation, or recommendation-combination, mechanism computes a reputation measure based on an entity's firsthand observations, or private information, and recommended, or public, information. Reputation systems are typically based on public information that reflects the general consensus of the community, although evidence resulting from personal experience is usually considered more reliable than recommendations. These recommendation-combination methods include simple sum, Bayesian, discrete, belief, fuzzy, and flow mechanisms.

The most basic method of reputation calculation is that used by eBay (eBay Inc. 2006), i.e., simple summation. A recommendation from each interaction is a record of whether the interaction had a positive, negative, or neutral result. A positive recommendation is worth 1 point, negative is worth -1 point, and neutral is worth 0 points. A reputation measure is the sum of all positive unique recommendations and all negative unique feedback, where unique means that only one positive and one negative recommendation from a specific recommender count toward the total reputation, thus attempting to prevent collusion for the purpose of reputation tampering. The advantage of this system is that it is very easy for community members to use and understand. The simplistic nature of the method, however, does not necessarily give an accurate representation of reputation, i.e., it ignores aspects such as timeliness and integrity of recommendations.

A slightly more advanced method is used in reputation systems such as that of Amazon (Amazon.com 2006), in which the reputation measure is calculated as an average of all recommendation scores. This method can be extended to calculate a weighted average of all recommendations, whereby weight is derived from factors such as trust in the recommender, timeliness of feedback, or semantic distance between the recommendation measure and personal experience.

Other types of evidence aggregation proposed for reputation systems are similar to those used for combining trust measures, i.e., Bayesian approaches (Jøsang 1999; Mui, Mohtashemi et al. 2001; Jøsang and Ismail 2002; Mui, Mohtashemi et al. 2002; Jøsang, Hird et al. 2003; Buchegger and Le Boudec 2004; Whitby, Jøsang et al. 2005), belief theory (Jøsang 1999; Yu and Singh 2000; Jøsang, Gray et al. 2006), and discrete methods (Abdul-Rahman and Hailes 2000; Cahill, Shand et al. 2003; Carbone, Nielsen et al. 2003).

Moreover, fuzzy models (Manchala 1998; Sabater and Sierra 2001; Sabater and Sierra 2002; Sabater 2004) allow trust and reputation to be represented as linguistically fuzzy concepts. In this type of system, reputation is measured through membership functions that describe to what degree an agent

can be trusted. Fuzzy logic (Zadeh 1965; Zadeh 1976) provides rules for reasoning with this type of measure.

Finally, flow models can be used to compute reputation by transitive iteration through looped or arbitrarily long chains. In flow models, a constant reputation rank is assumed for the whole community and this weight is distributed among the community members. Participants only increase their reputation at the cost of others, i.e., a participant's reputation increases as a function of incoming recommendations and outgoing recommendations lead to decreased reputation.

2.4.3 Reputation Management

Reputation management systems have been proposed to provide reputation measures to assist decision-making in a variety of applications, e.g., choice of transaction partners in online auctions and e-commerce, selection of honest peers in a P2P network, and detection of misbehaving nodes in mobile ad-hoc networks. Intuitively, interacting with an entity that has a positive reputation (as opposed to no reputation or a negative reputation), i.e., a more trustworthy entity, is more likely to result in a positive transaction experience, so reputation management provides an incentive for honest behaviour as well as to deter dishonest parties from participating.

Resnick et al. (Resnick, Zeckhauser et al. 2000) identify three basic properties necessary for reputation management systems to function:

- Entities are long-lived, so that there is an expectation of future interaction
- Feedback about current interactions is captured and distributed, so that this evidence can be visible in the future.
- Past feedback guides current decisions, so that entities are forced to pay attention to reputations.

The first property relates to the contextual element of time. As demonstrated in the cooperation results of Axelrod's iterated Prisoner's Dilemma experiments (Axelrod 1984), in a given interaction between Alice and Bob, one entity may be tempted to defect, i.e., not cooperate, if the pay-off is big. However, defection is more likely to occur if Alice and Bob are unlikely to interact again; and cooperation is more likely to occur if Alice and Bob know that they must interact again, or if reputation is made available to other entities who may wish to interact with Alice or Bob in the future. By reinforcing the idea that time must be captured in trust and reputation systems, identity longevity also implies that measures should exist in the system to discourage any change to, deletion of, or tampering with identifying information that might erase links binding an entity's identity to records of its past behaviour.

The second property depends on the evidence gathering and distribution protocol specified, which is typically straightforward in centralized systems but more difficult in distributed systems.

The third property depends on how usable the reputation system is and whether or not people and systems do indeed make use of the information available. For example, while the eBay reputation system may calculate reputation based on very simplistic feedback, community members are willing to engage in the feedback process and therefore the reputation system maintains usability. Thus, a balance between granularity of evidence and usability of evidentiary process exists.

Several types of reputation management systems are proposed in the literature. In the following sections, we review several commercial and academic reputation management systems in terms of the reputation management properties identified.

2.4.3.1 Commercial Reputation Management Systems

Transacting online in the e-commerce arena can make it difficult to use traditional methods for establishing trust between parties. E-commerce vendors of third party items, such as eBay, Yahoo!Auctions, and Amazon, use reputation as a means for propagating information about buyers and sellers. Additionally, a variety of commercial rating applications, from shopping comparison sites such as BizRate to search engines like Google, utilise reputation as the basis for decision-making algorithms that rank e-commerce vendors and webpages respectively.

In the eBay reputation system (eBay Inc. 2006), a reputation is provided for each member such that other members may consider the reputation as grounds for deciding whether or not to interact. Evidence in the form of recommendations, called feedback statements, records the behaviour of a member acting in the role of buyer or seller in a given eBay transaction as observed and recorded by the other party to the transaction. For each transaction, the buyer and seller are allowed to rate each other's actions by leaving feedback consisting of the following content: a rating (positive, negative, or neutral), and a short comment (of up to 80 characters). The buyer and seller are not obliged to leave feedback to complete the transaction. Evidence left in the eBay feedback loop is highly subjective, as it is left at the discretion of the buyer or seller, and not as the result of any consistent evaluation against set criteria. To mitigate against improper actions in the feedback loop, eBay policy prohibits the following feedback policy violations:

- Shill feedback - Using secondary eBay User IDs or other eBay members to artificially raise the level of one's own feedback.

- Feedback extortion - Demanding any action of a fellow user that he or she is not required to do, at the threat of leaving negative feedback.

- Feedback solicitation - Offering to sell feedback, trade feedback undeservedly, or buy feedback.

- Feedback abuse, e.g., the use of inappropriate language – Is subject to removal.

When a complaint is made by a member about improper feedback, eBay reviews the feedback and may enforce actions including, placing limits on account privileges of the offender, suspending the account of the offender, or simply removing the offensive feedback.

Evidence is recommended explicitly via feedback statements, directly to eBay's centralised feedback accumulation system when a member submits feedback about a transaction partner. The source and recipient of the feedback are captured by the recommendation. Furthermore, contextual parameters such as role, time and date, and item category, as well as a link (available for 90 days after transaction completion) to transaction details such as bidding activity and final price, are appended to each piece of feedback by the system.

A simple sum aggregation method is used to form a reputation measure from all unique positive and negative recommendations about a given member. For example, if Alice has interacted with Bob twice and she records two negative ratings about Bob, only one will contribute to Bob's reputation. However, if Alice leaves two negatives and one positive recommendations about Bob, the negative rating will count once and so will the positive rating. Subsequent negatives or positives from Alice will not affect Bob's reputation measure. A member's reputation score is thus the total of the number of unique eBay members that are satisfied doing business with a given member. It is the difference of the number of (unique) members who recorded a positive rating and the number of (unique) members who recorded a negative rating.

Reputation is distributed in two ways. First, a reputation score is appended to each member's username, e.g., Alice(20), expressing that Alice has received at least 20 positive recommendations. Second, each member's reputation is publicly viewable in the eBay Feedback Forum, where the reputation measure is available as well as a percentage measure of overall positive recommendations, i.e., the number of positive ratings divided by the total number of ratings. For example, one might assume that a member such as Alice(20) has been involved in 20 transactions with positive outcomes. However, upon closer inspection of Alice's reputation profile in the Feedback Forum, one might find that Alice has 60 positive recommendations and 40 negative recommendations, meaning that she behaves correctly in only 60% of interactions. Moreover, a reputation profile details the basic information about the member, i.e., geographic location and date of joining the eBay community, as well as a list of each recommendation left by trading partners from previous transactions.

A member is tied to his username, thus providing identity longevity, in two ways. First, credit card details are required to register as a member of the eBay community. Second, once a member has established a good reputation, he will most likely not wish to enrol for a new username that has an unknown reputation associated with it.

Furthermore, the eBay reputation management system is highly usable. It has been found (Resnick, Zeckhauser et al. 2000; Resnick and Zeckhauser 2002) that, despite incentives to free ride, i.e., when

an individual makes use of reputation scores to assist in decision-making but does not contribute to the system by leaving feedback, recommendations were provided more than 50% of the time. Moreover, reputation profiles were typically predictive of future performance, i.e., a person with a good reputation usually behaved correctly and vice versa. Additionally, sellers with better reputations were more likely to sell their items. Also, there was a high correlation between buyer and seller feedback, suggesting that the players reciprocate and retaliate.

Several issues arise in this type of reputation management system. First, a reputation measure alone does not express the likelihood of a positive outcome from interacting with a member. Second, the accumulated recommendations in a reputation profile contain information in the qualitative statements that may provide pertinent information about specific behaviour, e.g., delivery, payment, or communication, that is not captured by the positive/neutral/negative rating method and may be relevant to a member attempting to make a decision. Next, these recommendations contain context information that may be relevant to determining trustworthiness of a member for a particular trust purpose, i.e., role, time, and item category and pricing information, that is not captured by the reputation measure, nor are recommendations degraded as time passes. Fourth, while all recommendations are available for perusal by members, they are often far too numerous for a human to sift through to determine which recommendations are pertinent to a current decision. Fifth, recommendations may be recorded by members of different levels of recommendation integrity, and, in the Internet auction domain, users typically have no observations with which to compare recommendations and are thus heavily reliant on recommendations that tend to be Pollyanna assessments (Resnick and Zeckhauser 2002), that is, overly positive to the point that a reputation may be built on many false positive recommendations. Ebay provides no method for assessing recommendations for reliability according to semantic distance or recommendation path between two members. Additionally, no method for determining the risk of interaction is provided. Finally, while eBay acknowledges the fact that collusion between members takes place in approximately 2% of transactions and employs proprietary methods to counteract collusion, no collusion assessment information is provided to a potential decision-maker.

The Yahoo!Auctions (Yahoo! 2006) reputation management system operates in the same manner as that of eBay, with identical benefits and problems.

In the Amazon Marketplace (Amazon.com 2006) reputation system, reputation is provided for potential buyers to use when determining whether or not to interact with a seller of new or used books. Recommendations take the form of a transaction rating statement indicating the behaviour of a seller in a given Amazon Marketplace transaction. For each transaction, a buyer is allowed to rate a seller's actions by leaving feedback consisting of a rating and a short comment of no more than 200 characters and submit this recommendation to the centralised reputation calculation mechanism. The rating is expressed as an explicit recommendation of one of five levels, from 5 stars (best) to 1 star (worst). Both buyers and sellers may leave feedback ratings, although Amazon only evaluates seller reputation. The source and recipient of a recommendation are recorded, as is the date that the feedback

was submitted (within 90 days after transaction completion), however no further transaction details such as price or item category are documented.

Recommendations are aggregated centrally by Amazon. Similar to the eBay reputation system, recommendations are combined in two ways, first as a total number of ratings received (overall and within the past 12 months), and also as a percentage of positive feedback, i.e., an average of stars out of five stars. Positive feedback relates to 5 or 4 stars, neutral feedback to 3 stars, and negative feedback to 2 or 1 stars. The reputation measure is appended to each seller's username and appears in each instance that the username appears. Additionally, the reputation measure and a list of all recommendations are displayed in a seller's reputation profile webpage, but only feedback ratings submitted by buyers are included in a seller's overall feedback calculations, i.e., when a member has interacted in the role of buyer and received feedback, these recommendations are not included in his reputation as a seller. In this way, Amazon reputations are role-specific.

Evidence left in the Amazon Marketplace feedback loop, like that of eBay and Yahoo!Auctions, is highly subjective, as it is left at the discretion of the buyer, and not as the result of any consistent evaluation against set criteria. However, in the Amazon system, buyers are reminded of some criteria to keep in mind while recording the star rating. Buyers are asked to consider the following criteria before leaving their transaction rating, but this information is not captured in the recommendation nor in the reputation measure: was the item shipped on time and packaged acceptably to prevent damage in transit; did the received item meet expectations of what was ordered, i.e., did it match the seller's description; if there were problems with the order, did the seller handle them in a satisfactory fashion; and would the seller be one to recommend to a friend?

Furthermore, the Amazon reputation management system suffers from similar problems as those identified for eBay and Yahoo!Auctions in terms of lack of expressiveness of reputation measure, lack of capture of contextually relevant information in the reputation measure, arbitrarily large recommendation lists that are not suited to manual assessment, no means of determining recommendation integrity, and lack of risk assessment.

Comparison shopping search engines, such as Shopzilla (Shopzilla 2006), use reputation as a means for establishing trust between buyers and sellers, such that online shopping transactions are trustworthy. Shopzilla is not a merchandise retailer like eBay, Yahoo!Auctions or Amazon. It is an online shopping search and comparison engine that provides reputation information for over 40,000 stores. Shopzilla uses the BizRate ShopRank mechanism, a proprietary search algorithm, to produce search results by weighing price, popularity and availability of products, against the reputations of the merchants that sell them.

Evidence is recommended as ratings from customers and Shopzilla members, i.e., online shoppers who have volunteered to provide ratings and feedback to help others shop. In both cases, a recommendation is an explicit rating according to 15 criteria, 8 of which are recorded by a buyer at the checkout part of a purchase, and the remaining 7 are recorded by a buyer after purchase delivery

has occurred. These merchant-specific criteria include: ease of finding product, selection of products, clarity of product information, prices relative to other online merchants, overall look and design of the online site, shipping charges, variety of shipping options, charges stated clearly before order submission, availability of the product required, order tracking, timeliness of delivery, meeting of expectations with regard to product, customer support, potential for repeat business, and overall rating. For each of these criteria, a recommender may rate a merchant as outstanding, good, satisfactory, or poor. In this way, source and context information is explicitly captured by a recommendation, according to context-specific criteria. Only ratings from the most recent 90-day period are assessed, leading to up-to-date reputation formation.

Shopzilla's evidence collection mechanism is integrated into the checkout procedure of participating merchants, and recommendations are submitted to its centralised reputation formation scheme. Shopzilla then forms a reputation from the recommendations based on the total number of individual consumer reviews that have been collected for a particular merchant. Shopzilla does not consider anything less than 20 reviews in a 90-day period to be a statistically relevant number, rather than just the opinion of a handful of buyers, and therefore does not post reputation scores for stores below this threshold of reviews. These stores are designated as: 'Not Yet Rated'. The equation that calculates a reputation score determines each merchant's rating for each of the 15 criteria:

$$\frac{(\text{Average Survey Scores} \times \text{Number of Surveys}) + (\text{Average Member Scores} \times \text{Number of Member Reviews})}{(\text{Number of Surveys} + \text{Number of Member Reviews})}$$

ShopRank, the BizRate proprietary shopping search algorithm, is used by the Shopzilla shop search and product comparison website to evaluate reputation measures in order to rank merchants for shopping search queries by relevance, weighing many factors like prices, popularity and availability of products, as well as the reputation of stores that sell them. Within equal relevance bands, merchants that pay to be Customer Certified Shopzilla members are listed above those that do not in search results. The overall reputation measure is expressed by a 'smiley face' next to a merchant's online listing, and there are four different faces to represent the four different levels of reputation formed by the consumer ratings. Maintaining longevity of merchant identity is simpler in the BizRate environment because real-world merchants are tied to their recognizable store names both on and offline. Additionally, Shopzilla advertises that ratings are collected from more than one million online buyers each month, showing that the system is highly usable. Thus, the Shopzilla system incorporates evidence gathering and combination processes that accumulate information for reputation formation which captures many of the important trust characteristics.

2.4.3.2 Summary of Commercial Reputation Management Systems

Commercial reputation management systems have been adopted for online interaction. Systems such as those used by eBay, Yahoo!Auctions, and Amazon capture important trust and context information about members and transactions, and are very understandable and usable. However, a main failing in

these systems is that, while fine-grained evidence is captured, it is not represented in a reputation measure, which is typically a simple sum of positive and negative feedback counts rather than an expression of an entity's character in terms of trustworthiness according to specific parameters of a given context. The recommendation information is available for perusal by system users who may wish to derive a more relevant reputation characterisation for a given member, but in many cases recommendation lists are too large for any human user to sift through to find recommendations pertinent to a current decision. Moreover, the notion of recommendation integrity, i.e., the honesty and capability of the user making the recommendation, is not captured.

Shopzilla attempts to integrate fine-grained evidence into the reputation formation process, and then to exploit reputation to rank merchants for search results for a given query. In this way, contextually-relevant evidence is gathered and used to form a reputation measure that characterises a particular entity's ability to meet the needs of a querying user. Moreover, the systems assesses timeliness of evidence. The output of the system is a search result ranked by reputation that assists the user in finding the most competent source of product or information according to the context in which the query is submitted. This system thus provide a better example of how reputation might be formed and exploited based on evidence.

2.4.3.3 Academic Reputation Management Sytems

Many academic reputation management mechanisms have been proposed to improve propagation and assessment of trust information for reputation formation, and a representative set of this research (Abdul-Rahman and Hailes 1997; Abdul-Rahman and Hailes 1999; Abdul-Rahman and Hailes 2000; Yu and Singh 2000; Sabater and Sierra 2001; Jøsang and Ismail 2002; Mui, Mohtashemi et al. 2002; Sabater and Sierra 2002; Xiong and Liu 2003; Buchegger and Le Boudec 2004; Sabater 2004; Whitby, Jøsang et al. 2004; Whitby, Jøsang et al. 2005) is described in the following.

As discussed previously, Abdul-Rahman and Hailes (Abdul-Rahman and Hailes 1997; Abdul-Rahman and Hailes 1999; Abdul-Rahman and Hailes 2000) model a decentralised trust management system based on distributed reputation management in which entities identify malicious parties and propagate this information around the system as recommendations about trustworthiness. A recommendation explicitly captures information about identity, trust category (context), and discrete trust value, i.e., very trustworthy, trustworthy, untrustworthy, very untrustworthy. An entity wishing to determine another entity's reputation queries a trusted recommender and receives a recommendation containing a trust measure as calculated by the recommender based on his direct observations. A method for recommendation weighting according to semantic distance is described, as well as methods for the combination of recommendations to form an overall discrete reputation measure. Combining discrete recommendations is not computationally straightforward, however, and requires heuristic methods such as look-up tables and user-specified policies that are subjective and therefore difficult to automate.

Mui (Mui, Mohtashemi et al. 2002) describes a reputation management system in which recommended evidence is captured as: $\rho : A \times O \rightarrow \{1,0\}$ where rating, ρ, is explicitly recorded by agent, a, as approval, 1, or disapproval, 0, of agent o's behaviour. Context is defined as a set of attributes about the environment in which agents are interacting and is expressed as a binary value, i.e., either an attribute is present in an environment or not. Distribution and collection of ratings occurs in a distributed manner in which opinion sharing between agents is performed through 'encounters' in which query agent, a_i, requests a recommendation from response agent, a_j, about the trustworthiness of agent, o_k, in a given context. Weighting based on recommendation integrity, i.e., to what extent a_i approves of past ratings given by a_j, is applied to recommendations, and agents implement preference-based rating based on recommendation integrity, i.e., when gathering evidence to form a reputation measure for a target entity, an agent selects the members of the community whose recommendation ability he most approves of. Recommendations are combined through a Bayesian calculation. Trust information is not modelled as multi-faceted in this model, however, and reputation is therefore very simplistically represented as a combination of counts of binary outcomes of interactions, i.e., cooperation or defection. Additionally, the model is implemented in a simulated restaurant and movie recommendation system and not in an environment, such as a virtual marketplace in which monetary exchanges occur, in which reputation management, rather than recommendation sharing, is necessitated.

Jøsang (Jøsang and Ismail 2002) proposes a reputation engine called the Beta Reputation System as a flexible and statistically sound model for reputation management. In this model, evidence is recorded by agents as positive and negative feedback, $\left(r_T^X, s_T^X\right)$, in which r_T^X represents a positive rating by X about T's observed behaviour, i.e., X's degree of satisfaction with T in an interaction, and s_T^X represents a negative rating, or degree of dissatisfaction. All feedback about T is accumulated centrally and combined according to $r_T^{X,Y} = r_T^X + r_T^Y$ and $s_T^{X,Y} = s_T^X + s_T^Y$ where both X and Y have submitted feedback about T. Bayesian techniques are used to update T's reputation with new evidence, and the reputation is expressed as an expected probability reputation measure, $\rho\left(p \mid r_T^C, s_T^C\right)$, where C represents all recommenders of T, that indicates how T is expected to behave in the future based on accumulated evidence about past behaviour. Discounting is incorporated using belief metrics such that feedback from a highly reputable agent carries more weight than that of an agent of a low reputation. The notion of forgetting is introduced for cases in which older feedback may not be as relevant as T changes behaviour over time, and is represented by the 'forgetting factor', λ, which can be adjusted in the range from 0 to 1, where 1 represents no forgetting of old feedback, and 0 represents the fact that only the most recent piece of feedback should be assessed. Note that during evidence collection, the contextual element of time must be appended to each piece of feedback such that the forgetting algorithm will make proper assessments based on the order in which recommendations were received. After a transaction, each agent may provide feedback to a central management component that stores recommendations, applies discounting and forgetting algorithms, updates a target's reputation, and provides updated reputations to all agents in the community such

that each agent may use the reputation measure as information when considering whether or not to interact with another entity. While this system is suggested for use in the commercial e-commerce domain, parameters for trustworthiness and context are not suggested, and the model has not been evaluated in a real-world scenario.

Buchegger et al. (Buchegger and Le Boudec 2004) propose a distributed reputation system for use in the P2P and mobile ad hoc network domains. In their model, a reputation rating, $R_{i,j}$, is an opinion formed by entity, *i*, about entity *j*'s behaviour as an actor in the base system, i.e., providing correct files in a P2P file-sharing network or correctly participating in a routing protocol of a mobile ad hoc network, and this rating is used to classify an entity as normal or misbehaving. Additionally, a trust rating, $T_{i,j}$, is an opinion formed by entity, *i*, about entity *j*'s behaviour as an actor in the reputation system, i.e., whether or not the first hand information provided by *j* is likely to be true, and this rating is used to classify an entity as trustworthy or untrustworthy. First hand information is combined using a modified Bayesian approach and stored as a summary record, $F_{i,j}$, by each entity. Whenever *i* makes a direct observation about *j*'s behaviour, $R_{i,j}$ and $F_{i,j}$ are updated. From time to time, nodes publish their first hand summaries to a subset of the population, and if an entity, *i*, uses a recommended summary from entity, *k*, to update reputation, it also updates $T_{i,k}$ for the recommending entity based on *k*'s recommendation integrity and closeness to *i* in the network. Similar to the forgetting factor in the Beta Reputation System, a fading method is provided so that an agent's reputation may be degraded after a period of inactivity. The reputation was plugged into the Confidant system (Buchegger and Boudec 2002), a misbehaviour detection system for mobile ad hoc networks. The performance of the reputation system was evaluated using a simulator. It was found that detection time of misbehaving nodes is improved when second hand information is used to help form reputation, false positives increase as the population becomes more untrustworthy, and overhead depends only on the Confidant system itself. Again, this model does not identify the various aspects of trustworthiness that should be captured by a trust measure in order to form reputation, and context information is very basic.

Yu and Singh (Yu and Singh 2000; Yu and Singh 2002) propose a model of reputation for agents interacting in the e-commerce domain. Evidence for reputation formation is captured as a trust assessment by source agent, *i*, regarding trust in target agent, *j,* at a given time, $T_i(j)^t$, reflecting the quality of past interactions between agents in terms of expertise, i.e., service provision, and helpfulness, i.e., ability to make good recommendations. The trust value is based on direct observations of *j*'s willingness to cooperate or defect in an IPD, as well as recommendations from other agents. A recommendation is a trust value reflecting direct observation of expertise by a recommender of the target agent. This value may be collected through a referral chain, in which case each agent in the chain is assessed in terms of ability to recommend, and the direct trust value about *j*'s expertise as recorded by the recommender of *j*, i.e., the last leg in the referral chain, is weighted according to the recommendation integrity of the other agents in the path. If there is more than one

referral chain to a recommender of *j*, it is assumed that *i* will choose the path that produces the highest trust in that recommender. Recommendations are distributed through query-response in a decentralised manner by agents, as well as through flooding of gossip, i.e., spread of recommendations when no query has been initiated, throughout the agent network. Evidence is combined using Dempster-Shafer belief theory, and the system assumes subjectively-determined cooperation thresholds, above which an agent will trust and cooperate and below which an agent will distrust and defect. While time is taken into account in this model, other context information such as role or environmental factors are not considered. Thus, trust information for reputation formation is simply a combination counts of binary outcomes of interactions, cooperation or defection.

In the PeerTrust (Xiong and Liu 2003) distributed reputation model, trust is measured as the basis for assessing reputation amongst peers in a P2P community. Reputation reflects the degree of trust one entity has in another based on past experiences. An explicit recommendation, or feedback, consists of a trust value that is the weighted average of the amount of satisfaction one peer receives from each transaction with a second peer. The weight takes into account the feedback itself, the reliability of the feedback source and the transaction and community contexts, including temporal adaptation through weighting of historical evidence. PeerTrust uses a transaction-based feedback system to bind each recommendation to the transaction to which it pertains. The system requests feedback after the interaction is complete from the two peers involved. Source credibility is calculated in one of two ways. The first method calculates source reliability implicitly, assuming that a peer with a high trust value for interacting will also be a trustworthy recommender. While this method is simpler to implement, it may be the case that a peer maintains a trustworthy reputation for interaction but reports malicious feedback about its competitors. The second credibility measure calculates personalised similarity, or semantic distance, between two peers as regards their weighting of feedback. Trust information is then combined according to:

$$T(u) = \alpha * \frac{\sum_{i=1}^{I(u)} S(u,i) * Cr\left(p(u,i) * TF(u,i)\right)}{I(u)} + \beta * CF(u)$$

Where $I(u)$ represents the total number of transactions performed by peer, *u*, during a given time period; $p(u,i)$ represents the other participating peer, *p*, in *u*'s ith transaction; $S(u,i)$ represents the normalised amount of satisfaction *u* receives from *p* in its ith transaction; $Cr\left(p(u,i)\right)$ represents the credibility of feedback submitted by *p*; $TF(u,i)$ represents the adaptive transaction context factor for *u*'s ith transaction; and $CF(u,i)$ represents the adaptive community context factor for *u*'s ith transaction. The model is implemented in a simulated complaint-based rating system, and the process of computing and distributing reputation in this manner in a decentralised environment in which there is no central database to manage trust data is found to be too computationally expensive, and thus caching of past evidence is proposed as a potential solution. Furthermore, as in Mui's model, trust information is not modelled as multi-dimensional, and reputation is therefore very simplistically

represented as a combination of counts of binary outcomes of interactions, depending on whether the outcome of a transaction was satisfactory or not.

ReGreT (Sabater and Sierra 2001; Sabater and Sierra 2002; Sabater 2004) is a reputation model based on social relations. The authors present an analysis of social network relationships to identify features that may be valuable in an e-commerce environment to increase successful negotiation between users. The model is implemented in a simple supply chain simulation in which three relationship types may exist: competition, a relation between two agents pursuing the same resources who will use all available mechanisms to gain advantage; cooperation, where sincere information exchange exists and agents are predisposed to help one another; and trade, a relation reflecting the existence of commercial transactions and which is compatible with both competition and cooperation. Within these relationships in the supply chain example, negative seller behaviour is typed as one who overcharges, delivers late, delivers lesser quality goods, or a swindler who overcharges and/or delivers lesser quality goods. A buyer's negative behaviour is limited to a buyer who does not pay in full or at all for items. Evidence used to form reputation for a buyer or seller is based on agent ratings assessed in three dimensions, i.e., individual, social, and ontological. The individual dimension comprises evidence about direct interactions between agents, in which an agent assesses the outcomes of interacting with another agent according to type of outcome, e.g., seller overcharged, and weight, giving more relevance, or weight, to more recent outcomes. The reliability of individual rating is assessed based on the number of outcomes one agent has performed with another, i.e., the greater the variability in rating values of outcomes, the more volatile an agent's behaviour is believed to be in regard to fulfilment of agreements. The social dimension is incorporated into reputation formation because direct interaction information is not always available. In this case, recommendations are requested from witnesses who have interacted with a target agent. The witness set is identified and the most reliable witnesses, i.e., those with the most interactions with the target, are queried for explicit recommendations. These recommendations are combined using fuzzy rules. The social dimension also takes into account the trustworthiness of witness clusters, again using fuzzy rules to analyse relationships between neighbours in a witness cluster, and common knowledge about the interaction domain. Finally, an ontological dimension to reputation is incorporated into reputation formation to capture more complex behaviour, e.g., when an agent delivers late and overcharges. The ReGreT system adds societal structural concepts to the typical reputation constructs of observation and recommendation as well as treating reputation as multi-faceted concept rather than a single concept. The system performs well in such a strictly defined interaction environment in which context is limited to only a few types of trust purposes and in which rules can be established in advance for combining binary, i.e., cooperate or defect, outcome results for those trust purposes. In a real-world system, however, the dynamics of agent interaction may be too complex to be managed by such a model.

2.4.4 A Summary of Academic Reputation Systems

The academic reputation systems described above are representative of current research into reputation management wherein a multitude of different systems with advanced features are being proposed by the academic community. These models make inroads into the modelling of reputation systems that capture the evidence necessary to characterise an entity's nature in terms of trustworthiness according to explicit recommendations whose representation encapsulates contextual elements such as role, time, environmental factors, and social dynamics. Moreover, these models depict ways in which to represent evidence in mathematically sound formats that can be operated on for updating, discounting according to recommendation integrity, fading according to time, and to incorporate the notion of uncertainty. Such evidence can be aggregated according to both centralised and distributed methods such that reputation formation can occur to provide a reputation measure for exploitation in trust-based decision-making by humans or agents.

These models, however, have not been implemented in real-world systems in which it is necessary to specify environmental factors as explicit parameters which contribute to forming a measure of trustworthiness. The majority of models are focused on simple scenarios in which an entity may only choose one of two interaction options, i.e., to cooperate or to defect, rather than the more complex interaction scenarios in which trust is important and in which the characterisation of an entity by reputation must capture more complex parameters. Moreover, it has been found (Jøsang, Ismail et al. 2006) that current academic proposals for reputation management lack scalability and coherence, and that the reputation measures proposed are not easily understandable by average human system users. However, Jøsang notes that, in regard to development of reputation management systems, the 'period we are in can therefore be seen as a period of pioneers, and we hope that the near future will bring consolidation around a set of sound and well recognised principles for building trust and reputation systems, and that these will find their way into practical and commercial applications' (Jøsang, Ismail et al. 2006).

2.4.5 Outstanding Issues with Reputation Management Systems

From our survey of both commercial and academic reputation management systems, we find that several issues currently exist. First, accuracy of both recommendation and reputation measure is a concern. Granularity of feedback for a given application domain should be optimised according to specific parameters rather than left open to user subjectivity when providing coarse-grained ratings. Evidence requires a formal representation so that it can be operated on for combination, discounting, time fading, and uncertainty depiction in a mathematically sound way. The way in which a reputation measure is calculated is also of importance to accurately reflect the contributing evidence. For example, in a simple sum evidence combination approach, an agent who performs hundreds of transactions and cheats 25% of the time will maintain a steadily increasing reputation whose measure appears to be representative of a good reputation, whereas combining evidence according to, e.g., belief theory, may provide a reputation measure that more accurately describes the expected

likelihood of an agent to cheat on a future transaction. Second, there is typically an assumption that feedback is honest and unbiased, which may not always be the case, especially given a tendency toward Pollyanna feedback assessment. This introduces the need for methods to assess reliability of recommenders, e.g., according to discounting approaches. Third, contextual relevance of evidence for reputation formation is not taken into account by most commercial reputation systems and is assessed in only a very basic way in most academic reputation models. This results in a reputation measure that is not wholly accurate for a specific user decision, as well as leaving human users with the sometimes impossible task of manually sifting through the entire recommendation set to retrieve evidence that is appropriate for reputation formation to a particular query. The application of context assessment to evidence introduces the need for techniques of filtering evidence by role and environment-specific parameters, as well as for temporally adapting evidence. Simply applying a role filter, for example, would avoid the phenomenon of 'bought reputations' (Cabral and Hortacsu 2004) when a user purchases many small-price items in order to accrue a large number of positive recommendations before starting to sell items. Finally, often there is no incentive for users to provide feedback and free-riding occurs. Resnick and Zeckhauer (Resnick and Zeckhauser 2002) find, however, that 60.7% of buyers and 51.7% of sellers in the eBay community leave feedback after completing transactions, suggesting that the eBay feedback model is simple and usable enough for members to operate.

Many of the above issues arise from the current state of practice in commercial reputation management, and the methods proposed in academic research can be applied to resolve the concerns. When applying more complex mechanisms for reputation formation and exploitation, however, it is imperative to consider that the resulting system should be one that reduces complexity, increases accuracy, yet maintains usability for the user community.

2.5 Chapter Summary

In this chapter we investigated current research into modelling and adapting traditional human decision-making processes based on trust and reputation for online interaction environments. First, we discussed the various aspects of human trust and extracted trust attributes and properties that should be synthesised into a unified trust model, including: the ability to specify confidence in possible outcomes of interaction; capture of trust diversity dimensions; specification of trust formation, evolution, and exploitation processes to incorporate subjectivity; direct and indirect evidence specification, collection, update, and analysis processes including discounting, weighting, and filtering; specification of context, including the elements of role, time, environmental factors, and environmental constraints; incorporation of risk assessment methods for both trust target and environment; production ability of a meaningful and usable measure of trust from subjective analysis of contextually relevant evidence such that it may be used by a trust origin to be exploited in propagating trust to the community via recommendations and to make trusting decisions to interact for a given trust purpose with a given trust target in light of associated risk; and usability through

reduction of complexity. We presented McKnight and Chervany's model of human trust as an illustration of a high level model in which each of the trust characteristics is incorporated.

Next, we evaluated trust research in the computer security community and found that, while this community has produced initial results in modelling the notion of trust, development has traditionally been in the area of formal logics, which solely provide a notation with which to describe the process of authentication, and trust models for authorisation that are too application-specific to be suitable as general computational trust models. Moreover, none of these models explicitly defined what it means to be trusted, i.e., '*why* is this user trusted to execute this action?', rather than simply '*is* this user authenticated or authorised?', assuming that an intuitive notion of trust is universally understood.

Third, we explored more recent research in the computer science domain into using human trust as a basis for devising a computational trust model. We found that models based on the human notion of trust distinguish themselves from earlier attempts by the computer security community to use trust as a basis for computational decision making because the human trust models are founded on the attributes and properties of human trust, and they attempt to unify these trust characteristics into models which allow computational trust-based decision-making to occur in a similar manner to human trust-based processes. Of the models evaluated, we found that only the SECURE trust model captured the important trust attributes and properties. The SECURE model provides mechanisms to assess trustworthiness and capture trust evidence as a trust measure, assess risk, observe and recommend evidence, filter evidence for trust, risk, and contextual relevance, and exploit trust to provide a security decision to a querying user. Thus, the SECURE model unified all of the key trust characteristics and processes into one complete framework that has been implemented and validated in a system such that computational trust can be used for decision-making in a manner analogous to the human process.

Section 4 defined reputation and presented the mechanisms needed to form reputation in online environments, including an analysis of the types of evidence and evidence collection, assessment, and exploitation processes required. An overview of existing reputation management systems, both academic and commercial, was presented, which highlighted deficiencies with existing reputation systems.

Having identified the properties of trust and reputation necessary to build a computational trust-based reputation-management system that provides accurate and usable functionality to users of online communities, we propose the extension of the SECURE trust framework to include a richer notion of reputation analysis. In this way, the SECURE reputation management system can incorporate measures to set right the current deficiencies in reputation formation and implement trust- and reputation-based decision-making to provide security decisions to end users.

Chapter 3: Design of the Reputation Management System

Character is like a tree and reputation like a shadow.
The shadow is what we think of it; the tree is the real thing.
~ Abraham Lincoln

As highlighted in the previous chapter, a number of issues remain outstanding in the development of reputation management systems for virtual marketplace applications, mainly the promotion of usability over accuracy in terms of evidence gathering, assessment of contextual relevance, and assessment of interaction dynamics, as well as the lack of making risk explicit or providing decision support to users through reputation summary information.

First, commercial reputation systems typically promote usability over accuracy in the evidence feedback loop that provides evidence for reputation evaluation. Although it is important to maintain usability through a simple feedback recording process, it is at the cost of gathering observed and recommended evidence that might more accurately demonstrate patterns of user behaviour. That is, current reputation systems capture only a high level of information about an interaction and thus lose accuracy.

A second issue arises when contextual information about a user's role and type of interaction is captured by a recommendation but not incorporated into the reputation evaluation process. In this case, a reputation is based on all available evidence rather than that evidence that is contextually relevant with regard to role, time, and environment to making a given interaction decision.

Third, inaccurate evidentiary analysis also occurs with regard to interaction dynamics, i.e., information about relationships between pairs of users. In the C2C domain, it is possible for a system to observe the dynamics of user interactions, e.g., whether or not a user provides useful and accurate recommendations about another user or whether or not a user employs a specific interaction strategy, such as collusion with malicious intent, when interacting with another user. As in the case of determining which information is more contextually relevant to a decision at hand, it is impossible for a user to manually assess an overwhelming amount of system observations about interaction dynamics to determine which information is relevant when considering whether or not to interact with a given set of users.

Fourth, interacting online carries with it an associated cost, or risk, that an interaction will result in an undesirable outcome. Existing reputation systems do not make risk explicit to the user.

Finally, in existing reputation systems, a reputation for a given application participant consists of a set, potentially very large, of recommendations about that user. When making a decision to interact with someone with a reputation based on a large number of recommendations, a user would need unlimited time and resources in order to manually analyse each individual recommendation. Therefore overall reputation is typically evaluated by a reputation management system as a summary of the recommendation set, with a reputation score provided as the output of the evaluation process. However, the reputation score often conveys no more information than a count of positive recommendations or an average of positive recommendations out of all recommendations in the set. A simple summary reputation can be very deceptive because it allows a user to behave incorrectly some of the time while maintaining an overall good reputation. A more useful output of a reputation management system would be a secure decision that is a guideline to a user based on the analysis of trustworthiness, context, interaction dynamics, and risk of interacting with another user or set of users in a particular transaction.

We propose a trust-based reputation system that addresses the above issues, and this chapter describes the design of such a reputation management system. First, the approach taken by the Secure Environments for Collaboration among Ubiquitous Roaming Entities (SECURE) project is described, including a discussion of the threats SECURE is designed to address; the SECURE trust, collaboration, and risk models; the SECURE framework components and decision-making process; and a description of how SECURE is implemented as a software kernel with application program interface. Next, this chapter depicts the deployment of SECURE in the spam filtering domain, highlighting the decisions taken with regard to assessing trust, collaboration, and risk in this domain, as well as how SECURE decision-making for spam filtering is implemented and evaluated. Third, we provide a general description of reputation management in virtual marketplaces application domain, highlighting the processes intrinsic to Internet auctions and reputation management which leads to our development of a taxonomy of behavioural threats in this area. Then, the design of a trust-based Reputation Management System (RMS), based on the SECURE model, for virtual marketplaces is put forward. An overview of the design is given, and our rationale about design decisions is presented with regard to requests, entity recognition, trust and evidence processes, risk assessment, and access control. Fifth, we propose an extension to the reputation management system to allow enhanced decision-making through interaction management for recommendation weighting and collusion detection. We illustrate how the two new interaction management components may be integrated into the reputation management system and detail the enhanced decision-making process. Finally, a chapter summary is provided.

3.1 The SECURE Approach

The SECURE trust/risk-based security framework (SECURE TSF) (Cahill, Shand et al. 2003) is at the state of the art in trust-based security and is designed to be deployed in application domains characterised by a large population of diverse entities needing to make autonomous security decisions in uncertain environments wherein perfect information is not always available. In such domains, SECURE applies a formal approach to reasoning about trust and risk in a manner analogous to the human decision-making process.

This section discusses the approach taken by the SECURE project. First, a discussion of the threats SECURE is designed to address is presented. Next, the SECURE trust, collaboration, and risk models are explained. Third, we describe the the SECURE framework components and decision-making process. Finally, a description is provided of how SECURE is implemented as a software kernel with an application program interface.

3.1.1 Threat Taxonomy

This section presents a taxonomy of the application-independent threats that may occur in environments, i.e., global computing environments[2], in which SECURE is intended for use (Ingram, Dimmock et al. 2005). The general threats, summarised in Table 2, fall into three main groups, i.e., behavioural threats, that is, those that arise due to the behaviour of a single malicious entity; recommendation threats, i.e., those attacks in which collusion is employed in order to subvert the proper functioning of a recommendation system, i.e., a system for distributing and gathering recommended evidence; and system threats, i.e., attacks against the system as a whole rather than against a specific component or entity.

Table 2: Threats in the global computing environment

Behavioural threats	Recommendation threats	System threats
Bad guys	Collusion clique	Human error
Newcomer attack	Collusion with supporters	Software faults
Basic Sybil attack	Collusion with camouflage	Identity theft
Waiting attack	Defamation	Routing attacks
Oscillation attack	Indirect Sybil attack	Privacy attacks
Mixed behaviour attack	General Sybil attack	Denial of service
Chaotic behaviour		Resource costs
Misconfiguration		Deployment costs
Carelessness		
Peer-specific attacks		

[2] According to the European Union's Future and Emerging Technologies Global Computing Initiative, global computing refers to computation over 'global computers', i.e., computational infrastructures available globally and able to provide uniform services with variable guarantees for communication, co-operation and mobility, resource usage, security policies and mechanisms.

Behavioural attacks may occur when trustworthiness of entities is not assessed, when there is a lack of evidence about a given entity, or when a discrepancy occurs between pieces of evidence available about a given entity's behaviour. For example, bad guys are principals who behave badly consistently over time, and a trust-based system should exclude their participation. Newcomers are principals who are new participants in the system and for whom no evidence has yet been gathered for analysis. The basic Sybil attack is one in which the attacking principal employs a succession of badly behaved principal identities, that is, benefiting from protracted exploitation of the newcomer attack. In a waiting attack, the attacker typically accumulates good recommendations until he is poised to make a big pay-off through bad behaviour. A stronger version of this attack is the oscillation attack, in which a principal switches between well-behaved and hostile modes of behaviour in an attempt to manipulate his perceived trustworthiness. Similarly, the mixed behaviour attacker chooses a mode, e.g., to cooperate or defect, for each separate interaction probabilistically with an objective to manipulate his perceived trustworthiness to stay just below the threshold of being identified as a bad guy. Attacks based on chaotic behaviour or misconfiguration lead to random or constant incorrect behaviour respectively. A careless attacker is generally good, but occasionally suffers from 'trembles', behaving badly by mistake at random. Finally, the peer-specific attack is aimed at specific participant(s), e.g., only cheating a subset of principals that are seen as easy targets, e.g., naïve newcomers who are unsure of what constitutes correct behaviour, while behaving well with other, more experienced principals to gain good recommendations.

Recommendation attacks are those in which entities collude to undermine the correct performance of a recommendation system. A collusion clique is a group of principals in which all behave badly but provide false positive recommendations for each other, leading to their initial success at bad behaviour until negative recommendations result from interactions with principals outside of the clique. Collusion with supporters is a case in which only one principal behaves badly while its colluders continue to recommend it positively, highlighting the necessity of calculating trust in recommenders. Collusion with camouflage is a combination of collusion with supporters and mixed behaviour, wherein the active principal only misbehaves some of the time, and the supporters never misbehave but supply the principal with good recommendations, such that he may escape detection while trying to profit by bad behaviour. Defamation involves falsely attacking a principal's good reputation. The indirect Sybil attack occurs when a stream of colluding recommenders, i.e., one colluding recommender after another over time, boost the trust value of one badly-behaved principal, and the general Sybil attack extends this to many teams of colluding recommenders and an arbitrary number of bad principals.

System threats enable attacks against the system as a whole. Human error is a major problem with security systems and can be particularly problematic if users have to encode their own policies. Software faults include failure and recovery issues. Identity theft occurs when an attacker breaks into a system and harvests a private key associated with an identity, e.g., through keyboard sniffers or spyware. Routing attacks target an overlay network in which a routing protocol is used to distribute routing information within networks, e.g., shortest paths and advertising routes out from the local

network. In this type of attack, an attacker could forge a routing packet to provide false network information. Privacy attacks occur when behavioural profiling information is revealed by evidence gathered by a person or program, giving a behaviour pattern about a given principal. This pattern may highlight, e.g., shopping behaviour or even link a user to his real-world identity. Denial of service attacks against a third party may prevent the ability to take certain actions. Finally, resource and deployment costs, while not related to specific attacks, are additional constraints which may affect the feasibility of system performance.

SECURE was designed against a backdrop of these application-independent threats. Later in this chapter, we elaborate on this threat taxonomy for the e-commerce context, that is, we put forward an application-specific behaviour taxonomy that classifies different malicious behaviour types in the reputation management for virtual marketplaces domain.

3.1.2 The SECURE Trust Model

This section describes the formulae and methods that SECURE trust model employs to describe and measure trust. An explanation is given of each of the techniques used in the trust model, including event structures and event configurations which describe the events occurring in an interaction between principals; interaction histories which describe a collection of observations about interactions; trust and information orderings which are used to determine when more or less evidence exists; and the effect and evaluation functions which are used to update trust values.

3.1.2.1 Event Structure

Trust calculation should be based on a model general enough that it may be instantiated in a variety of applications, and the SECURE trust structure provides such versatility. SECURE's trust components are based on a *formal model of trust* (Nielsen, Carbone et al. 2004) that is related to the work of Dempster and Shafer on a mathematical theory of evidence (Shafer 1976) and to the work of Jøsang on subjective logic (Jøsang 2001). SECURE's formal trust model uses *event structures* (Nielsen, Plotkin et al. 1981; Winskel and Nielsen 1995) to support the reasoning process and to capture the dynamic nature of relationships in which trust evolves over time. An event structure is a triple ($E, \leq, \#$) consisting of a set E of *events* that are partially ordered by $\leq$, the *necessity relation*, and in which $\#$ is a binary, symmetric, reflexive relation called the *conflict relation*. Two events are *independent* if they are not ordered by either of the two relations. In the sample event structure in Figure 12, as well as in subsequent figures, necessity, or causality, is denoted by an arrow and conflict is denoted by the # symbol.

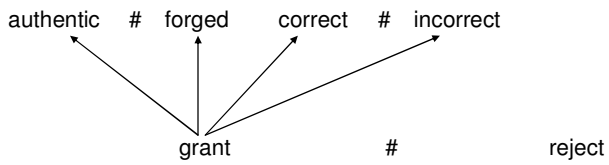

Figure 12: $ES_{e\text{-}cash}$ event structure for a simple e-cash scenario

An interaction between two principals is made of up a sequence of actions, called events, which can be observed. The outcome of an interaction in SECURE is described with an event structure. This is a tree that starts with no knowledge about the outcome at the root. Each event observed by the system narrows down the position of an outcome by traversing a branch of the tree. Leaves represent a completely resolved interaction in which the outcome is fully specified. Thus, an event structure models all of the possible actions a principal can observe within a single interaction.

During an interaction between two principals, each observed action is modelled as an event within the event structure. The event structure illustrated in Figure 12 models a simple scenario in which principal *p* requests a transfer of electronic cash (e-cash) from another principal, *q*. After making the request, *p* observes whether the request was granted or rejected, which is an example of conflicting events as the occurrence of the grant event excludes the possibility of the rejection event. If the request is granted, *p* might observe that the e-cash is authentic or forged, as well as whether or not the correct amount of e-cash was transferred. These latter events can only be observed if the request is initially granted, thus demonstrating the causal relation. The authenticity and correctness of the e-cash are two independent observations, i.e., they are neither causal nor conflicting.

3.1.2.2 Event Configurations

The trust model also introduces the notion of *configurations of an event structure*. Configurations model the sets of events that a principal could possibly observe during an interaction. As time passes, each principal will have observed some set of occurring events as part of an interaction, e.g., {grant, authentic, correct} or {reject}. Each such set of events constitutes an event configuration. For example, $C_{ES_{e-cash}}$, illustrated in Figure 13, is the set of possible event configurations (using abbreviations of the names of the events) of the event structure, $ES_{e\text{-}cash}$, for the e-cash example.

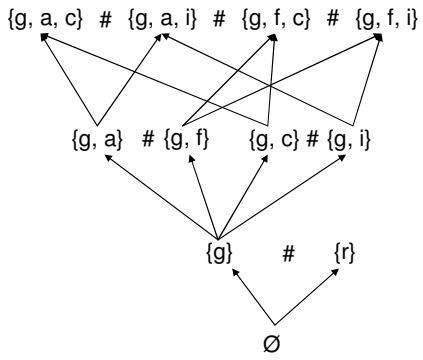

Figure 13: $C_{ES_{e-cash}}$ Event configurations for a simple e-cash scenario

It is important to note that the possible endpoints of an interaction are described by the possible configurations of an event structure. Event configurations are elaborated as time passes during an interaction. At a given point in time during an interaction between two principals, the outcome, i.e., configuration, observed may or may not provide everything possible about the interaction. For example, an e-cash user may complete an interaction having only observed $\{g, c\}$ and have no knowledge about whether the e-cash was forged or authentic. As time passes, further observations may or may not be made. As illustrated, it is possible to see that no information has been observed, i.e., Ø, through a range of observable events and event combinations at varying levels of granularity.

3.1.2.3 Interaction History

The trust model assumes some mechanism for the recording of events during interactions between principals. An *interaction history* is defined as a formalisation of what a principal has recorded about an interaction with another principal. An interaction history is a finite sequence, *H*, of interactions ordered over time, e.g., $H = x_1, x_2 \ldots x_n$. The individual elements, x_i, in the interaction history, *H*, are called interactions. Each interaction, x_i, in *H* is a configuration. For example, in the e-cash scenario, *q*'s recorded interaction history for *p* might be $H_{e-cash} = \{g\}\{g,a,c\}\{r\}\{g,a,i\}\{g,c\}$. If *q* records information from a further interaction with *p*, the configuration observed is appended to H_{e-cash}.

Each configuration in *H* is a piece of evidence that can be used to provide a decision about a given proposition. A proposition is a statement, which is itself a configuration, that is used to determine the likelihood of a principal observing a particular configuration in a future interaction with another principal. Each configuration recorded in *H* is either supporting, contradicting, or inconclusive about a proposition. A principal would like to estimate the likelihood of ending up in a particular configuration, e.g., proposition *w*, in a future interaction with another principal given that a piece of evidence shows that the last interaction with that principal resulted in configuration *x*. If the

configuration *x* contains all of the events in *w* and therefore supports the likelihood of proposition *w* occurring. If *x* instead contains an event which rules out the configuration *w*, then *x* contradicts the occurrence of *w*. If neither of these are the case, *x* is inconclusive about *w*. For example, if a proposition *w* states that an interaction with a principal is likely to result in configuration {*g, a, c*}, and if $H_{e-cash} = \{g\}\{g,a,c\}\{r\}\{g,a,i\}\{g,c\}$, then there is one piece of evidence that supports the occurrence of *w*, i.e., {*g, a, c*}; there are two piece of evidence, i.e., {*g*} and {*g, c*}, that are inconclusive about the occurrence of *w*; and there are two pieces of evidence, i.e., {*r*} and {*g, a, i*}, that contradict the occurrence of *w*.

3.1.2.4 Trust and Information Orderings

Thus far, the trust model provides a way to model evidence that may be observed about a given interaction. *Trust values* may be derived from this evidence. Trust values are equated with the notion of evidence values, i.e., values that express evidence about a particular configuration. If we consider an event structure, *ES*, a trust value will be a function from its configurations, C_{ES}, into a domain of trust values. The function applied to a configuration, $x \in C_{ES}$, produces a value that reflects the evidence about *x*. This is a trust value which is a triple of natural numbers, $(s,i,c) \in \mathbb{N}^3$. The interpretation is that out of *s*+*i*+*c* interactions, *s* interactions support the occurrence of *x*, *c* interactions contradict the occurrence *x*, and *i* interactions are inconclusive about the occurrence of *x*.

The trust model defines two orderings on the set of trust values, an ordering expressing more information overall, i.e., an *information ordering*, and an ordering expressing more evidence in favour of a proposition, i.e., a *trust ordering*. Consider the comparison of two (s,i,c) triples as evidence about a particular proposition, e.g., *x*. A partial order, $\sqsubseteq$, is defined on the triples to determine which triple expresses 'more information' about *x* as follows:

$$(s,i,c) \sqsubseteq (s',i',c') \Leftrightarrow (s \le s') \wedge (c \le c') \wedge (s+i+c \le s'+i'+c')$$

According to the information ordering, (s',i',c') represents the existence of more evidence about the truth of proposition *x* than (s,i,c) if it is possible to start out from (s,i,c) and end up with the value (s',i',c') by observing some additional number of supporting or contradicting events, or refining some events, or both. Refining of an event means to change an inconclusive observation to a supporting or contradicting one.

The trust ordering, $\preceq$, expresses more evidence in favour of a proposition and is defined as follows:

$$(s,i,c) \preceq (s',i',c') \Leftrightarrow (s \le s') \wedge (c \ge c') \wedge (s+i+c \le s'+i'+c')$$

Here, (s',i',c') expresses at least as much evidence in favour of the proposition in question as (s,i,c) if there are at least as many supporting events, no more contradicting events, and at least as many total events. Intuitively, (s',i',c') can be obtained from (s,i,c) by refining inconclusive events to supporting, or by observing further positive or inconclusive events, or both.

Surprisingly, we find that this formula can lead to counterintuitive results. For example, consider (s,i,c) = (10,1,1) and (s',i',c') = (100,1,2). According to the trust ordering, (s',i',c') does not express more evidence in favour of a proposition than (s,i,c) because there are more contradicting observations in (s',i',c'). However, the number of supporting observations in (s',i',c') far surpasses the number of supporting observations in (s,i,c), and there is a much more significant increase in supporting observations in (s',i',c') than there is in conflicting observations. Intuitively, one would believe that there is more evidence in favour of a proposition in (s',i',c') even though the trust ordering disagrees.

An alternative trust ordering formulation that produces more intuitive results might be:

$$(s,i,c) \preceq (s',i',c') \Leftrightarrow \left(\frac{s}{c} \le \frac{s'}{c'}\right) \wedge (s+i+c \le s'+i'+c')$$

In the alternative trust ordering, the changes in supporting and contradicting evidence is measured as a ratio, thereby allowing for the case in which there are more contradicting observations accompanied by significantly more supporting observations. As long as the $\frac{s}{c}$ ratio increases and evidence is at least as numerous in (s',i',c'), we can say that (s',i',c') expresses more evidence in favour of a proposition.

3.1.2.5 Effect and Evaluation Functions

The trust model provides the effect, *eff*, and evaluation, *eval*, functions with which to update trust values. The effect function is defined as:

$$\mathrm{eff}_x(w) = \begin{cases} (1,0,0) \text{ if } w \subseteq x \\ (0,0,1) \text{ if } x \# w \\ (0,1,0) \text{ otherwise} \end{cases}$$

Consider *x*, a configuration that has already been observed when *p* has interacted with *q*. *p* may be considering another interaction with *q*, and wants to determine whether the future interaction will result in proposition *w*. To determine the likelihood of the future interaction resulting in *w*, given that the last interaction resulted in configuration *x*, three effects are possible in *eff*: if $w \subseteq x$, then the fact

that x occurred in a previous interaction supports the likelihood of the occurrence of w; if $x \# w$, then x contains an event that rules out the configuration w meaning that x contradicts the occurrence of w; if neither $w \subseteq x$ nor $x \# w$, then x is inconclusive about w.

The *eval* function is used to map an interaction history to a trust value. The evaluation function is defined as:

$$\text{eval}(x_1, x_2 \ldots x_n) = \lambda w. \sum_{i=1}^{n} \text{eff}_{x_i}(w)$$

Thus, the evaluation of a particular interaction history returns a (s,i,c) triple that comprises the sum of updated evidence in favour of, contradicting, and inconclusive about all observed events. The *eval* equation is configurable so that it is able to calculate trust based on a subset of interactions from the interaction history as well as permitting the weighting of interactions. For example, q may only be interested in p's most recent interactions. In order to model memory such that a principal only 'remembers', or takes into account during decision-making, the most recent $M+1$ interactions, the evaluation function can be changed to:

$$\text{eval}^M(x_1, x_2 \ldots x_n) = \lambda w. \sum_{i=n-M}^{n} \text{eff}_{x_i}(w) = \text{eval}(x_{n-M}, x_{n-M+1} \ldots x_n)$$

In another example, q may wish to weight recent interactions with p more strongly and interactions that are more than, e.g., a year old, more weakly, and this could be modelled by incorporating scaling or fading weights into the equation.

3.1.3 The SECURE Collaboration Model

Using the formal trust model as a basis for reasoning about trust allows us to be more precise about how evidence is processed by SECURE, however, as mentioned before, the trust model assumes the existence of mechanisms to collect and store evidence. The SECURE *collaboration model* (Terzis, English et al. 2005c; Terzis, English et al. 2005a; Terzis, English et al. 2005b) describes the way in which evidence is gathered, stored, and processed for trust formation, evolution, and exploitation.

The collaboration model distinguishes between two types of evidence, i.e., direct evidence, or *observations*, and indirect evidence, or *recommendations*. The former result from the monitoring of observations of outcomes during direct interaction with a given principal, while the latter are other principals' opinions about a given principal, i.e., their trust values for that principal based on their own observations. This distinction is important because the two types of evidence may be treated differently. Direct evidence may be accepted as fact (assuming that a principal is satisfied that its own observations are fully reliable) while the value of indirect evidence depends on the reliability of the recommending principal, i.e., recommendation integrity, thereby introducing the need for the

adjustment of recommendations in accordance with one's subjective assessment of recommendation integrity.

In order to distinguish between the two types of evidence, the collaboration model specifies two different types of evidence procedures, i.e., *interaction monitoring* and *evidence gathering*, as well as an *evidence store* for storing evidence of different formats. The interaction monitoring process occurs during an interaction between two principals, in which one principal records in an interaction history the configurations, i.e., outcomes, observed while interacting with the other principal. The evidence gathering process comprises the approach a principal might use to request and receive recommendations, in the form of trust values, about another principal from third parties.

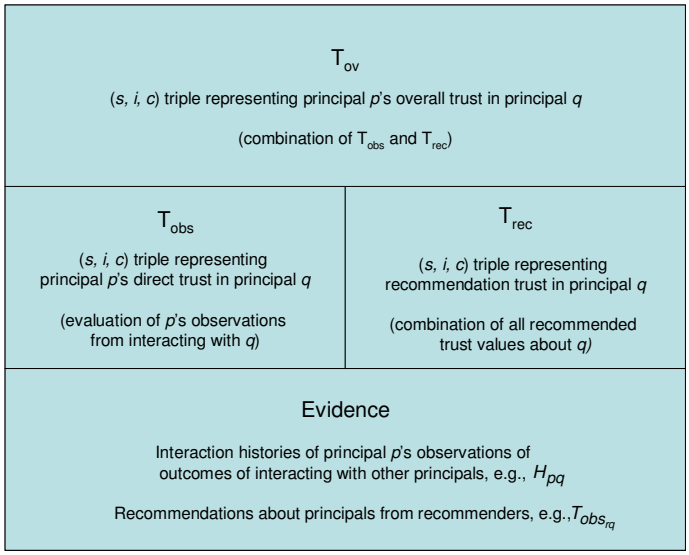

Figure 14: SECURE evidence architecture

The evidence architecture, as illustrated in Figure 14, provides a layered model of the way in which evidence is structured for use by the SECURE system. The bottom level of the evidence architecture represents the totality of information stored in an evidence store. It contains unevaluated lists of all observed and recommended evidence. This layer of evidence is updated with observations provided by the interaction monitoring process, i.e., updates to an interaction history that is a record of *p*'s interactions with another principal, and with recommendations provided by the evidence gathering process, i.e., trust values based on other principals' observations.

The second layer of the evidence architecture contains evaluated evidence. T_{obs}, a SECURE trust value, i.e., a (s,i,c) triple, is derived from an interaction history using the *eff* and *eval* functions specified in the trust model. T_{rec} expresses the combination of recommended trust values, i.e., observations made by other principals when interacting with *q* that are evaluated as trust values and passed as recommendations. The point at which SECURE derives T_{rec} from all of the

recommendations is the stage at which recommendation integrity would be assessed, although no such methods to perform assessment of recommender reliability currently exist in the SECURE model.

Observations and recommendations are evaluated independently, thus giving T_{obs} and T_{rec}, respectively. In this way, p is able to keep its own opinions about, e.g., q, separate from the general population's opinion about q. Therefore, when recommending principal r is required to provide a recommendation about q, it passes T_{obs}, which is the evaluated result of its direct interaction with q, which ensures that the rules of transitive trust management (see Section 2.1.3.5) are upheld.

The third layer represents p's overall trust value for another principal, i.e., T_{ov}, which is a combination of T_{obs} and T_{rec}.

The collaboration model describes how principals can form an initial trust value for previously unknown principals, update or evolve that trust value based on new evidence, and exploit a trust value for decision-making. The *trust formation* process is a special case of trust evolution in which no observations have yet been made about a given principal's behaviour, i.e., T_{obs} is empty. In this case, a principal must rely upon recommendations to derive a trust value for the unknown entity.

Trust evolution (Terzis, English et al. 2005a) is the process by which principals update trust values in light of new evidence. Trust evolution is enabled by the *eff* and *eval* functions specified by the trust model, through which new observations and recommended evidence are assessed as supporting, conflicting, or inconclusive with regard to a proposition and then used to update T_{obs} and T_{rec} accordingly. SECURE treats observations as fact and therefore fully reliable. SECURE assumes that recommendations, however, are adjusted according to an undefined recommendation integrity assessment process, or assessment of trust in the recommender, before being taken into account to derive T_{ov}.

During the trust evolution process, the collaboration model incorporates the notions of *trust subjectivity* and *contextual parameterisation* with regard to processing evidence. Trust subjectivity is addressed through the ability of principals to specify policies for trust disposition, i.e., whether or not a principal is generally trusting or distrusting, and trust dynamics, i.e., policies that determine how quickly or slowly a principal builds and erodes trust in light of ongoing positive and negative experiences. The situational nature of trust-based decision-making is captured by the use of context to parameterise interactions such that evidence that is most contextually relevant to a given decision may be extracted during the decision-making process.

The notion of contextual parameterisation is also related to the process of *trust exploitation* (Terzis, English et al. 2005b), i.e., the interpretation of trust values in a given context. Again, during this process, the situational character of trust is incorporated, noting the fact that a principal's trustworthiness changes depending on the context of an interaction. Because an event structure models events for a particular type of interaction, context is inherently captured by the trust model, meaning that information is monitored and gathered as events of a specific interaction type. Thus, this

evidence is contextually relevant to a specific interaction type, so the trust value derived from the evidence can be exploited appropriately for future interactions of the same type. A contextually relevant, subjectively-evolved trust value is then exploited in conjunction with risk assessment to make a trust-based decision.

3.1.4 The SECURE Risk Model

Traditionally, risk is determined as a function of likelihood of exposure to a certain outcome and the predicted impact of that outcome should it occur. Risk applies to situations when one is unsure of which outcome will result from an interaction, but the likelihood of each potential outcome is known.

The SECURE risk model (Bacon, Dimmock et al. 2005) assumes the input of principal-specific evidence, i.e., a trust value, and context-specific evidence, i.e., information about the interaction type, application domain, etc., to derive the risk of engaging in a particular interaction with a specific principal. Risk is then assessed according to a state-preference model incorporating five components:

- A set of acts (*X*) available to a decision-maker, i.e., $X = \{x_1, x_2, \ldots, x_n\}$. In an e-cash payment scenario, it might be that the acts available to a bus company are x_1 = permit e-cash user to board bus, and x_2 = refuse boarding of e-cash user.

- A set of mutually exclusive states (*Z*), that is, potential outcomes of interaction, i.e., configurations of events, available to Nature i.e., $Z = \{\zeta_1, \zeta_2, \ldots, \zeta_n\}$. In the e-cash domain, let us say that the potential outcomes of interacting with an e-cash user are ζ_1 = e-cash payment accepted and ζ_2 = e-cash payment denied.

- A consequence function, $c(x, \zeta)$, showing the estimated consequence in terms of the cost of each possible combination of decision and state. This is described by the cost matrix below.

X/Z	x_1	x_2
ζ_1	c_{x_1,ζ_1}	c_{x_2,ζ_1}
ζ_2	c_{x_1,ζ_2}	c_{x_2,ζ_2}

We put forward a sample cost matrix for the e-cash example as follows.

X/Z	*permit*	*refuse*
accepted	$c_{permit,accepted}$ = -(fare)	$c_{refuse,accepted}$ = fare
denied	$c_{permit,denied}$ = fare	$c_{refuse,denied}$ = 0

In this cost matrix, a consequence that is a cost to the bus company is $c_{permit,denied}$, in which the company decides to permit an e-cash user to board a bus and subsequently the user's e-cash payment is denied. In this case, the bus company loses the price of the bus fare. A benefit, i.e., negative cost, is gained by the bus company when $c_{permit,accepted}$ is the consequence of the company's decision to permit an e-cash user to board a bus and subsequently the user's e-cash payment is accepted. If the bus company refuses boarding of a user whose payment would have been accepted, i.e., $c_{refuse,accepted}$, the bus company incurs a cost of the potential fare had it made a decision to permit. Finally, $c_{refuse,denied}$ results in a cost of zero as the bus company risks nothing and loses nothing in this situation.

- A probability function, $\pi(\zeta)$, expressing the beliefs of the decision-maker about the likelihood of an outcome occurring in a future interaction with a particular principal. This probability is derived from the SECURE trust value. For example, $\pi(\zeta)=\frac{s\zeta}{s_\zeta + i_\zeta + c_\zeta}$. To continue with our e-cash example, $\pi(\zeta_1)=\pi_{accepted}$ and $\pi(\zeta_2)=\pi_{denied}$.

- And an elementary utility function, $v(c)$, measuring the utility for each consequence. Cardinal utility (Wikipedia 2006) is a measure of the happiness or satisfaction gained from consuming goods and services. Conventional economic theory imagines that each individual is continuously maximising his utility and that cardinal utility is a quantity that has units and is measurable. Thus, cardinal utility can be used to quantitatively measure the preference of an individual toward a certain commodity. Clearly, a negative consequence is typically undesirable and a positive consequence is most desirable. In the e-cash example, utility could be assigned on a scale of units from 0-10, with 0 being the least desirable measure and 10 being the most desirable measure, as follows:

  $v\left(c_{permit,accepted}\right)$: most desirable consequence, e.g., utility of 10 units.

  $v\left(c_{permit,denied}\right)$: most undesirable consequence, e.g., utility of 0 units.

  $v\left(c_{refuse,accepted}\right)$: somewhat undesirable consequence, e.g., utility of 3 units, as the bus company could have profited more from a decision to permit rather than to refuse.

  $v\left(c_{refuse,denied}\right)$: somewhat desirable consequence, e.g., utility of 7 units, as a good decision was made but this consequence is still not as desirable as $c_{permit,accepted}$.

The von Neumann-Morganstern theory then gives the utility of an act, $x \in X$, as:

$$U(x) \equiv \pi_1 v\left(c_{x,\zeta_1}\right) + \pi_2 v\left(c_{x,\zeta_2}\right) + \ldots + \pi_\zeta v\left(c_{x,\zeta_n}\right)$$
$$\equiv \sum_{\zeta \in Z} \pi_\zeta v\left(c_{x,\zeta}\right)$$

To apply the von Neumann-Morganstern formula to the e-cash example, utility for each possible act is:

$$U(permit) = \pi_{accepted} v\left(c_{permit,accepted}\right) + \pi_{denied} v\left(c_{permit,denied}\right)$$

$$U(refuse) = \pi_{accepted} v\left(c_{refuse,accepted}\right) + \pi_{denied} v\left(c_{refuse,denied}\right)$$

If $\pi_{accepted}$ is 1.0 and π_{denied} is 0.0, that is, an e-cash user has an excellent history of accepted payments using e-cash, and if we apply the utility weights given in the above example, $U(permit)$ will result in a greater result than $U(refuse)$.

A simple decision-making, or access control, policy would be to choose any decision, $a \in A$ where:

$$A = \left\{ \left(a \mid a \in X\right) \wedge \left(U(a) = \max_X [U(x)] \right) \right\}$$

This policy selects from the set of all possible acts the act with the greatest utility. In the e-cash scenario, this policy can select whether it is more desirable to permit or to refuse boarding to an e-cash user based on the past history of that user, i.e., the evidence supporting the likelihood of his payment being accepted or rejected, and the utility metrics of the bus company.

If, however, $|A| > 1$, that is, there are two acts with the same utility, a further policy is required to determine which act to choose, e.g., a policy exposing the bus company to the least risk, i.e., to refuse boarding.

This model can also be used for the assessment of monetary costs and benefits in applications where financial considerations are present. However, the concept of cardinal utility suffers from the absence of an objective measure of utility when comparing the utility gained from the consumption of a particular good by one individual as opposed to that of another individual. The way in which risk is estimated is heavily dependent upon the values used to weight utility of the different consequences. An alternative solution may be to use the traditional security risk assessment method of calculating risk as a product of likelihood to exposure and cost, and we examine this solution later on when discussing risk assessment for reputation management in virtual marketplaces.

3.1.5 The SECURE Framework Components

The trust, evidence, and risk models provide the basis for the specification of the SECURE framework and its components, which are illustrated in Figure 15.

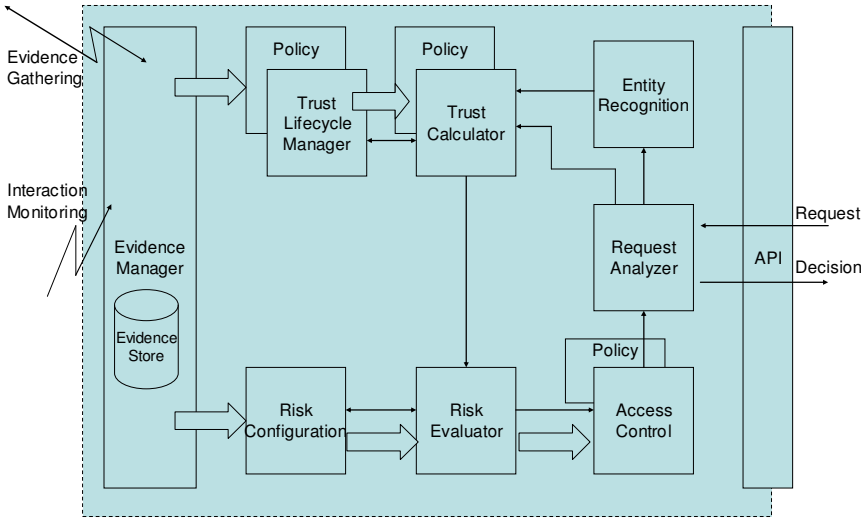

Figure 15: SECURE framework

The *Application Program Interface* (API) links an application to the SECURE decision-making process.

The *Request Analyser* (RA) is responsible for accepting the request parameters from the API and returning a security decision to the API.

The *Entity Recognition* component (ER) is responsible for identifying the requesting principal via some specified authentication mechanism based on identification information provided by the requesting principal. The ER component may also be used to calculate a level of confidence in the recognition of a principal based on the assessed reliability of the authentication scheme being used, and if this is done, the confidence factor may also be used in calculating trustworthiness.

The *Trust Calculator* (TC) is responsible for calculating an overall trust value, i.e., T_{ov}, for the identified requesting principal.

The role of the *Trust Lifecycle Manager* (TLM) is to evaluate observed evidence, i.e., to use the *eff* and *eval* functions to produce T_{obs} from an interaction history, and to combine recommendations to produce T_{rec}. Additionally, the TLM is the component in which contextual relevance of evidence is assessed.

The *Evidence Manager* (EM) is responsible for three processes, i.e., the monitoring of direct interactions via an interaction monitoring process, the gathering of recommendations via an evidence gathering process, and the storing of all unevaluated evidence in an Evidence Store (ES). The EM can also be used to store general context information that may be useful to the risk assessment process.

The *Risk Configuration* component (RC) updates context-specific risk, i.e., the likelihood of threat occurrence based on the interaction context regardless of interacting principals.

The *Risk Evaluator* (RE) accepts a trust value from the TC and a context-specific risk value from the RC, and derives the overall risk, in terms of utility, of the requested interaction. The risk is output to the Access Control component.

The role of the *Access Control* component (AC) is to apply decision-making policies with respect to trust-based risk of interacting in the requested interaction with the requesting principal and to pass the resulting security decision to the RA.

3.1.6 The SECURE Decision-Making Process

The SECURE framework components described above are used in the decision-making process as follows.

A request from a principal using an application is input to the SECURE kernel, via the API, and a security decision about whether or not to proceed with a requested application-specific interaction is output to the requesting principal via the API. When principal, *q*, requests an interaction with principal, *p*, the request passes through the API to the RA of *p*'s SECURE kernel. A request may be divided into the request itself, e.g., 'may I access this document?', and additional parameters, including identifying information about the requesting principal, e.g., the username of *q*, and context- or application-specific information. The RA passes the complete query to the ER component.

The ER performs a recognition process, e.g., password-based authentication, on the identity information of *p* contained within the interaction request. The ER component may also calculate a level of confidence on the reliability of the authentication scheme being used, e.g., it may ascribe 100% confidence to one password-based authentication method but only 80% confidence in another method. The ER outputs the interaction request for the identified requesting principal to the TC.

The TC receives the request from the ER and requests updated trust information, i.e., T_{obs} and T_{rec} from the TLM. The TC calculates a trust value for *q* by combining T_{obs} and T_{rec} to produce T_{ov}, a (s,i,c)-triple. Policies may be specified to allow the TC to determine subjective factors, e.g., a memory window, when calculating T_{ov}. T_{ov} is output to the RE.

The TLM requests evidence about *p* from the EM. The TLM receives an interaction history describing *p*'s observations of interacting with *q*, and calculates T_{obs} for *q,* using the *eff* and *eval*

functions. The TLM also receives recommendations from the EM. A recommendation-integrity assessment function may be applied to discount recommended evidence appropriately before combining recommendations to produce T_{rec}. Furthermore, policy may be specified to allow the TLM to evolve T_{obs} and T_{rec} according to p's subjective trust disposition and trust dynamics.

The EM stores trust and risk information in the ES. This information may be collected via interaction monitoring, i.e., interaction histories of p's records of observations about interacting with q, and evidence gathering, i.e., a process through which p can request and receive recommendations about q from other principals who have interacted with q. The EM may also collect risk-related information about the application domain that can be used to update the RC. When a request for evidence comes from the TLM or RC, the EM provides evidence to these components.

The RC updates context-specific risk based on application-specific evidence received from the EM. The RC provides this risk value to the RE when requested.

The RE accepts T_{ov} from the TC and a context-specific risk value from the RC, and performs a risk assessment. The risk assessment, i.e., the utility of each possible act, is output to the AC.

The AC receives the risk assessment from the RC and applies decision-making policies to formulate a security decision as to whether or not p should interact with q in the requested interaction. If the utility of interacting with q in the given situation is above a specified threshold, the AC passes a security decision to the RA to interact with the requesting principal. If the risk is not acceptable, a decision not to interact is passed.

3.1.7 The SECURE Kernel and API

The SECURE TSF is implemented in a security kernel that may be consulted by an application user for the purpose of trust- and risk-based decision-making during an interaction (Bryce, Cahill et al. 2005). The term kernel is used to denote the operational implementation of the SECURE model. The role of the SECURE kernel is to store all security-related data, e.g., received evidence and trust values, as well as the policy components that are consulted by a principal during a decision-making process.

A principal in SECURE is any autonomous entity that can initiate an action, is capable of making a decision, and in which another principal may have to place trust. At the implementation level, a principal is an execution entity with support for protecting its data from others. Once a principal has been recognised according to the entity recognition process, the trust and risk evaluation processes of the SECURE decision-making process can proceed.

The ability of a system to run on different hardware and operating system environments requires portability measures. Also, the notion of portability extends to applications and policies, i.e., it should be possible to reuse the same kernel for different applications or for one application with different

policies, e.g., to enable different principals interacting in the same application to configure different trust and access control policies. SECURE is implemented as a policy-neutral kernel that is fixed for all environments and applications, along with configurable modules for entity recognition, trust management, risk evaluation, etc. that are 'pluggable' into the kernel.

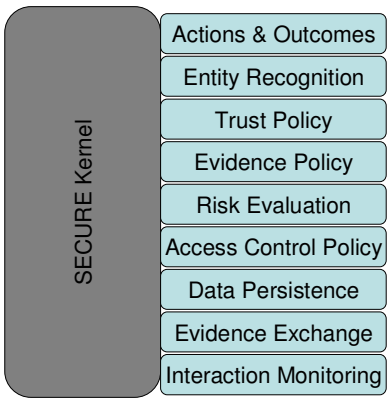

Figure 16: SECURE kernel and its configurable and pluggable modules

Obtaining portability in practice also requires that non-security features be integrated into the kernel, i.e., storage, network, and logging. These are relevant to SECURE for the persistence of principal data, exchanging of evidence between principals, and monitoring of interactions. Thus, an application developer must not only specify security policies in terms of trust, risk, evidence, and access control, but also in terms of how data is stored and exchanged. The pluggable modules mirror the components of the SECURE framework model, along with portability modules, as illustrated in Figure 16.

The implementation of SECURE does not follow precisely the idealised design specification, but does captures the spirit of the design to incorporate trust and risk assessment for decision-making. The components of the SECURE framework design are implemented as programmable policy components in the SECURE API, allowing the implementation of a decision-making process for a principal that is configurable according to the context of a specific application. Each class in the SECURE API corresponds to an abstraction of a framework component. The main class is SecureKernel that implements a decide method to be invoked by a principal's application code whenever a decision needs to be made by an application. Using the API, a developer defines polices for trust, risk, access control, etc., and 'codes an Action…in which one specifies Action.Outcomes and Outcome-Costs, as well as coding the appropriate subclasses of TrustLifecycleManager, AccessController, etc. to represent the trust, access control policies, etc.' (Bryce, Seigneur et al. 2005). A class must be coded for each component to be plugged into the kernel.

3.2 SECURE for Spam Filtering

The SECURE TSF has been successfully instantiated in the spam filtering application domain (Bryce, Cahill et al. 2005; Bryce, Seigneur et al. 2005), as an application coded over the SECURE kernel to enable trust-based decision-making based on the collaboration of autonomous email users. The spam application is representative of an autonomous system because each email user makes his own decision about whether or not a message is spam, perhaps relying on advice from others. In this environment, observations about interactions with a given email address are shared amongst users as recommendations, which are then used as evidence with which to reason about the trustworthiness of an email sender. Trust, evidence, and risk for SECURE-enhanced spam filtering are described in this section, as well as the SECURE decision-making process in the spam filtering application and its implementation and evaluation.

3.2.1 Trust

Recall that SECURE uses an event structure to model all of the possible observations that a principal can make about a particular interaction. The event structure, $ES_{spam\text{-}filter}$, illustrated in Figure 17 models the spam filtering scenario as described in (Bryce, Cahill et al. 2005; Seigneur, Bryce et al. 2005). Principal *q* sends an email to principal *p*. After the mail receive request enters the SECURE TSF, *p* observes whether the email is legitimate, yet to be read, or spam. Each of these three possible events has a causal relationship with the receive mail event, as well as conflicting relationships with each other.

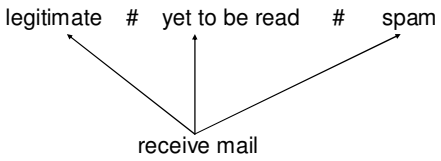

Figure 17: $ES_{spam\text{-}filter}$

As events occur in an interaction, *p* might observe that no information has been observed, i.e., Ø, through a range of observable events about whether the message is legitimate, not yet read, or spam. The event configurations for $ES_{spam\text{-}filter}$, i.e., $C_{ES_{spam-filter}}$, the sets of events that potentially can be observed, is given by Figure 18.

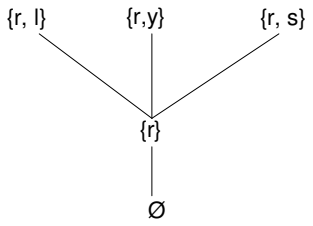

Figure 18: $C_{ES_{spam-filter}}$

A sample interaction history for the spam filtering application, i.e., a finite ordered sequence of interactions that *p* records about *q*, may be as follows:

$$H_{spam-filter} = \{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,l\}\{r,s\}\{r,l\}\{r,l\}\{r,l\}\{r,y\}\{r,y\}$$

The interaction history is updated with new evidence each time *p* observes a new event about *q*. Application of the *eff* and *eval* functions to $H_{spam\text{-}filter}$ produces a trust value, a (s,i,c)-triple, of (14, 2, 1) for *q* based on *p*'s evidence, i.e., 14 observations, $\{r, l\}$, of evidence in favour of an email being legitimate; 2 observations, $\{r, s\}$, contradicting that *q* sends legitimate email; and 1 inconclusive observation, $\{r, y\}$.

3.2.2 Collaboration

Evidence in the spam filtering scenario may be comprised of direct observations about incoming email being legitimate or spam from which a trust value based on observations may be formed, i.e., $T_{obs\text{-}spam}$, and recommendations from other email users, i.e., $T_{rec\text{-}spam}$. This evidence is then used to derive overall trust in an email sender, i.e., $T_{ov\text{-}spam}$. $T_{ov\text{-}spam}$ is exploited in conjunction with risk assessment to make a trust-based decision.

Events are observed by the email system according to user actions. If a principal *p* has yet to read an email from principal *q*, then that event is regarded as inconclusive with regard to the proposition of *q* being a legitimate email sender. If *p* has read an email from *q* and keeps it in the inbox, then that event is regarded as evidence that supports *q*'s legitimacy as a sender. If *p* has read an email from *q* and moves that message to the spam folder, then that event increases the count of contradicting evidence about *q*. If a message has been automatically routed to a spam folder, then that event also increases the count of contradicting evidence about *q*. If *p* reads a message in the spam folder and then moves it to the inbox, then that event is regarded as evidence that supports *q*'s legitimacy as a sender, so a contradicting event is changed to a supporting one.

Where no directly observed evidence is available for trust formation, recommendations become useful in determining whether or not an email sender is a spammer. In the spam filtering application, a static

approach is taken to recommenders, i.e., p manually specifies the email addresses of the mail server administrator and a set number of email addresses of chosen friends are considered to have full recommendation integrity, that is, a recommended trust value from any of these friends is considered to be as reliable as p's own observations. For example, if principal r sends p a recommendation about q, e.g., (0, 0, 1000), it means that r believes (or wants p to believe) that q is a spammer. If r is specified as one of p's friends, r's recommendation will weigh as much as p's own observations. If r is not specified as one of p's friends, however, r's recommendation will be rejected.

3.2.3 Risk

Risk analysis as described in the SECURE risk model is applied to the spam filtering scenario as follows:

- The set of acts available to a principal is $X = \{mark, pass\}$.
- The set of mutually exclusive states available to Nature is $Z = \{spam, notspam\}$. The two possible outcomes this instantiation of SECURE is concerned with are that the message is legitimate or that it is spam.
- Costs are expressed relative to what cost would be incurred if no spam filter were integrated. The consequence function, $c(x, \zeta)$, showing sample costs under all possible combinations of acts and states, is:

X/S	spam	notspam
mark	-1	*E*
pass	0	0

- The probability function, $\pi(\zeta)$, expressing the principal's beliefs in whether the message is legitimate or not, where π is the probability that a user is a legitimate email sender, rather than a spammer, based on the derived trust value. For example, π may be simply calculated as $s/(s+i+c)$ based on supporting, inconclusive, and contradicting evidence.

In this scenario, a utility weighting metric is not proposed, but rather it is assumed that the act with the highest utility is also the one that has the least associated cost. In the sample cost matrix above, passing a message always has an associated cost of zero, because this models the outcome if the SECURE TSF were not being consulted. Marking a spam message as spam provides a benefit, i.e., a negative cost, of 1, which was arbitrarily set to be the unit cost in this application. Marking a legitimate email as spam has an associated cost of E, the false-positive error cost. It is suggested that E is likely to be considerably larger than 1 and that E may be configured by the user based on, for example, the average severity of the consequence of losing a legitimate email relative to the cost of his time.

The act of marking a legitimate message as spam has a higher associated cost than that of the act of allowing a spam message to pass unmarked into *q*'s email inbox. The expected cost of marking a message as spam is then given by: $(\pi \times E) + (1-\pi)(-1) = \pi \times (E+1) - 1$. A message is only marked as spam if the expected cost is negative (that is, the expected benefit is positive) so the access control policy is to mark a message as spam when $\pi \times (E+1) - 1 < 0$, i.e., $\pi \times (E+1) < 1$. If the cost to *p* to lose an e-mail (be it spam or not) is high, e.g., *E*=100, that means that even email from an untrustworthy sender, e.g., $\pi(\zeta)$ for *q* is 0.05, will not be marked, i.e., 0.05(100+1) > 1, because *p* wants to receive all mail into the inbox. However, if the cost to *p* of mismarking legitimate email as spam is not very high, e.g., *E* = 10, i.e., *p* does not care if some legitimate mails get marked as spam as long as he does not get too much spam in his inbox, a spam message from most untrustworthy senders will be marked successfully and messages from legitimate senders will pass to the inbox.

3.2.4 Decision-Making

The components of the SECURE TSF process trust and risk information about incoming email messages to determine whether the message is legitimate, i.e., a message that is not spam, or spam.

In this application, a principal represents an email user, i.e., an email user's email address. Principal *p* runs an instance of the SECURE kernel in his mail server proxy. Each time an email message is received, the receive action instigates a query to SECURE to make a decision as to whether the message should be passed, as normal, into *p*'s inbox or marked as spam and taken out of the normal stream. When principal *q* sends an email to *p*, the request passes through the SECURE API to the RA in *p*'s SECURE kernel. The request includes the actual query, 'should this message be passed or marked?', and some identifying information about *q* such as *q*'s email address. The RA passes the complete request to the ER component.

The ER component recognises *q*, i.e., distinguishes whether or not *q* has sent a mail to *p* before. The recognised principal email address and the query are passed to the TC. The TC, supported by the TLM, the ES, and its local trust policy, computes a trust value based on observed and recommended evidence about *p*.

When an email from a known email address is received, {r} is added to the interaction history in the ES, and the TLM increases the *i* element of T_{obs} by 1 regardless of whether or not the message is marked or passed. As soon as *p*'s opinion on the decision is captured, the interaction history is updated, and 1 is subtracted from *i* and added to *c* or *s* according to the *p*'s opinion which is captured by a move request. The move request is intercepted by the proxy and interpreted as an observation of an outcome (of a message being spam or legitimate). These observations are used to change evidence from being inconclusive to being supporting or conflicting. Note that in this implementation, the SECURE *eval* function considers all interactions rather than isolating a time window of interactions for evaluation.

When an email is received from a new address, the proxy sequentially polls its list of trusted recommenders until a recommendation about the address is received or the list is exhausted. If a recommendation is found, then the trust value of the newcomer is set to the trust value in the recommendation.

The TC combines observed and recommended evidence to form T_{ov} about q, which is passed to the RE which calculates $\pi = {}^{s}\!/_{(s+i+c)}$, i.e., the likelihood of q being a legitimate email sender. The risk component outputs π to the AC to apply the access control policy, i.e., to mark a message as spam if $\pi \times (E+1) < 1$. At the end of the trust- and risk-based decision-making process, the RA receives a security policy decision from the AC, e.g., 'mark the message as spam' or 'let the message pass into the inbox', and provides the decision to the mail proxy via the SECURE API.

3.2.5 Implementation and Evaluation

The structure of the SECURE-enhanced spam filtering application is illustrated in Figure 19. In this application, a principal represents an email user who has an email client that is configured to use the Simple Mail Transfer Protocol (SMTP) proxy, a protocol for sending email messages between mail servers and the Internet Message Access Protocol (IMAP) proxy, a protocol for retrieving email from a mail server and routing it to an email client. IMAP also allows for the creation of email folders on a server and the copying of messages between folders. SECURE is called by the proxy to make a decision as to whether or not to mark a message as spam.

Messages marked as spam are routed by IMAP to a spam folder, while legitimate non-spam messages are routed to the user's email inbox. In the case of a false positive, i.e., a legitimate email is marked as spam, or a false negative, i.e., a spam email is let pass as legitimate, the user can move the message from or to the spam folder, and the move request is intercepted by the proxy and captured by SECURE in order to update trust evidence.

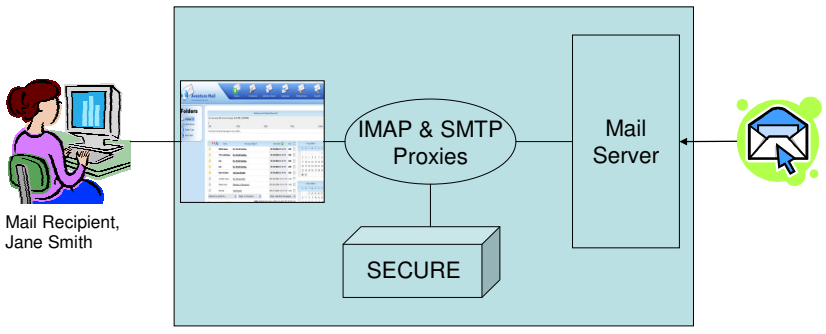

Figure 19: SECURE spam filter

The SECURE-enhanced spam filtering application has been evaluated in the SECURE Evaluation Framework (Bryce, Cahill et al. 2005). This experimental environment is illustrated in Figure 20.

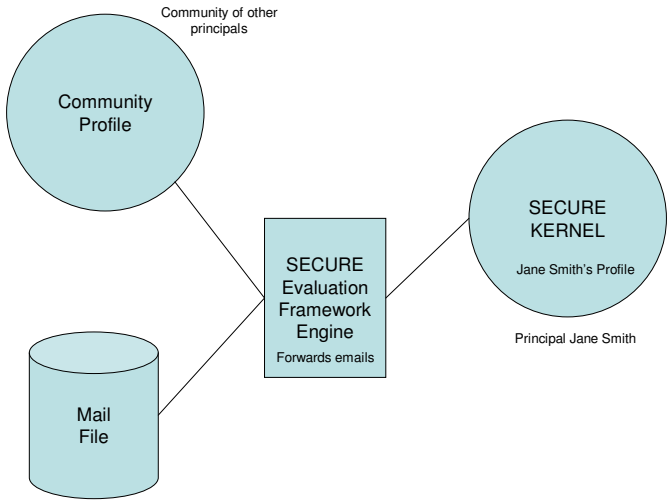

Figure 20: The SECURE evaluation framework configuration

The environment models and evaluates mail messages sent to principal Jane Smith, and uses a fixed set of messages that are stored in a mail file. Each of these stored messages is tagged *a priori* as spam or legitimate. Messages are routed from the mail file to Jane Smith's email client via the mail proxy, which calls SECURE to determine whether or not a message is spam. Each message is processed by the SECURE kernel's trust and risk policies, and a classification is assigned, i.e., spam or valid.

The results of the SECURE classification are compared to the pre-assigned spam tags which allows for the evaluation of SECURE's accuracy with regard to correctly identifying both spam and legitimate email messages.

The mail file is compiled from the SpamAssassin benchmark using the easy_ham and spam files, composed respectively of valid messages and spam messages. There are 3051 messages in the benchmark, of which 2551 are legitimate and 500 are spam. The messages are sent from a community of 846 unique senders, of which 425 are spammers and 51 are repeat offenders, i.e., send more than one spam message. The number of valid mail senders is 421, of which 213 are once-off message senders and 208 send more than one message.

Two cases are evaluated. In the first case, Jane Smith uses no spam filter. All messages come into her inbox. She correctly identifies 2551 emails as valid and leaves them in her inbox. She also identifies 500 emails as spam and moves these messages to a spam folder, meaning that there were 500 false negatives in this case (spam emails that passed to the inbox as valid). No false positives occurred. The classification process in this case is 71.84% accurate.

In the second case, SECURE is used to make trust-based decisions about filtering spam based on Jane Smith's observations about past mail sender behaviour. SECURE correctly classifies 2626 messages, with false negatives falling to 425 and false positives remaining at zero. The accuracy in this case rises to 86.07%, thus improving spam filtering by 14.23%.

With regard to recommended evidence, it is found that in the case where Jane Smith consults recommendations when she has no experience with a given email sender and when recommenders are fully trusted, i.e., are reliable to identify spammers, very high spam filtering accuracy can be achieved. In the case where some recommendations are false, if Jane Smith has some observations to rely on these observations act as a safety net against poor recommendations and valid classification can still be maintained, thus illustrating the importance of 'buddy principals' in whom Jane Smith can trust.

Other important results of the evaluation are that SECURE can be implemented in a real autonomous system, and that SECURE provides a complementary approach to Bayesian spam filtering systems. Whereas Bayesian spam filters classify a mail message as spam based on a statistical analysis of message content, it was found that the SECURE spam filter can classify a message as spam based on a number of additional criteria, including trust in the message sender, acceptable risk levels for false positive or false negative errors, and confidence in the underlying system's ability to recognise a mail sender.

However, we note that full SECURE capability is not utilised. For example, context is not used in this application, but it could be used, e.g., for providing information about the priority level associated with an incoming mail. Moreover, an alternative method for evaluating recommendations would have been to implement methods to assess recommendation integrity, especially given the evaluation result concerning the importance of recommendations when assessing email validity. Finally, it may have proven beneficial to provide the ability to evolve trust according to trust dynamic policies.

3.3 Reputation Management in Internet Auctions

Although SECURE was developed for distributed systems applications within the global computing environment, the SECURE TSF is also well-suited to optimise decision-making in centralised applications, for instance, reputation management in virtual marketplaces such as Internet auctions.

A centralised system is one that has a major hub with which entities communicate, e.g., to access information or services. Centralised applications are more easily regulated and organised than applications for distributed systems, and all participants in a centralised system need only to refer to the system for updated information and services. In this type of environment, SECURE can be used to evaluate evidence from the databases of the centralised application, rather than evidence that has been shared in a distributed manner.

SECURE can provide improved decision-making potential to users of marketplace applications through trust-, risk-, and context-based analysis of evidence captured by the reputation systems of such centralised systems. This section describes the processes involved in Internet auctions, including reputation management. We then describe the instantiation of SECURE for reputation management.

3.3.1 Internet Auctions

Internet auctions quickly became one of the most successful virtual marketplace applications after being introduced in the mid-1990s. For example, eBay is generally held up as an exemplar for the Internet auction industry, having captured approximately 75-85% of the consumer-to-consumer (C2C) market (Walker 2003; Wurman 2004; Forbes.com 2005) since its launch in September 1995. eBay had 135.5 million registered users at the end of 2004, an increase of 43% from 2003. New auction listings in the first quarter of 2005 increased 39% from 2004 to 404.6 million (CNN Money 2005). Moreover, in the business-to-business (B2B) marketplace, it is predicted that potential transaction volume of online auctions, e.g., the WorldWide Retail Exchange, which was founded in 2000 by a group of retailers including J.C. Penney, Gap, Auchan, Marks & Spencer and Tesco, will eventually overtake that of the C2C channel (Keenan 2000; Rosenthal 2002). eBay, as the leader in the C2C auction market, currently sets the standards for the B2B channel as well, as consumers continue to adapt to the usability standards it sets and expect to see such standards emulated in other application areas.

Recent research (Wurman 2004) describes the architecture, mechanisms, processes, and rules required to replicate traditional auction methods in virtual environments. An Internet auction is defined by a set of rules that regulate several processes, including the listing of, bidding on, and clearing of saleable goods. All auctions share the same core functionality of admitting bids, generating information, and clearing transactions. The sequence and frequency with which an auction system performs these actions is governed by auction rules. The rules also determine what types of bids are acceptable, what format intermediate information should take, and how trade prices are computed. Complementary features of most online auctions include cataloguing, searching, and managing reputation of participants. Thus, flexibility and ability to integrate complementary features are key requirements of auction system development.

3.3.1.1 Internet Auction Systems

In an auction system, three types of user role are defined: a bidder who bids on items; a buyer, who is the high bidder when an auction has finished; a seller, who creates a new auction, lists the item description, and sets the price. Moreover, there must be an auction system administrator who installs, configures, and maintains the auction application, but does not directly interact with bidders, buyers, or sellers.

Auction formats include: combinatorial, where a bidder may place an offer on a set of items; multi-attribute, where factors in addition to price are considered, e.g. product quality; sealed bit auctions, where each bidder submits his best bid to the auction initiator and all bids are evaluated at a specified time; and the single-seller-single-item auction, which is most typical of traditional auctions where a single item is offered for auction and bidders place bids until the auction ends or until there are no counter-offers to the highest bid.

An auction is created when the seller, i.e., the auction initiator, interacts with the auction system to specify and launch a new auction. The seller must clearly describe the product, including all necessary details for bidders to know exactly what is on offer, what shipping and handling procedures are available, return policy, etc. The initiator also selects the auction rules, including auction duration, bid increment, and bid format.

Bidding rules define the types of bids allowed in an auction system, as well as which users are permitted to participate and when bids can be accepted. The seller is the only user who can place a sell offer, and if the sell offer price is non-zero, it is called the reserve price. The simplest type of bid is an offer to buy one unit of an item at a specified price, and the bid language becomes increasingly complex for auctions such as combinatorial or multi-parameter types. Activity rules for bidding determine what types of bids are permitted, e.g., the improve-your-bid rule permits only strict improvements on the previous bid state.

The bidding component is responsible for enforcing the bidding rules, admitting bids that satisfy the rules, and extracting the auction description and current state from the database. Once a bid meets the rule conditions, it is considered to be a valid bid, and it is admitted into the current bid set and stored in the database. Any bid that does not meet rule conditions is rejected. A bid database is updated to track which bids are currently winning.

Clearing is the act of computing trades when an auction completes. In eBay, for example, an auction clears at a prescribed fixed time, the auction end. At this time, the highest bid above the reserve wins, i.e., the high bidder becomes the buyer of the item being auctioned.

Auction systems must also provide notification functionality, either by pushing current information to the user via email or by pull methods wherein the user checks current auction status.

Finally, auction systems typically provide integration for complementary features such as personalization, cataloguing and search, payment functions, and reputation management. Personalization includes the registration and authentication processes, as well as the capability for users to track specific auctions and items. Cataloguing and search functionality allows users to browse current auctions through hierarchies of categories or to search for auctions by key word or favourite seller. While smaller Internet auction providers relegate payment settlements to sellers and bidders, e.g., payment by cheque or postal order, larger Internet auction providers have joined together with electronic payment service providers to integrate the payment process into the auction system.

Reputation management has become a critical process for Internet auctions, and is discussed in more detail below.

3.3.2 Reputation Management in Internet Auctions

The traditional marketplace was a point where members of the public could come together to sell their wares, to browse available items, to haggle over price, to outbid neighbours, and to spread recommendations about which sellers and buyers acted in a trustworthy manner. In such a marketplace, reputation is formed as these recommendations accrue from gathered historical interaction data – cheats are identified, for example, as well as honest traders. In the virtual world, these communities are organized around a centralised marketplace application which provides to community members the computational equivalent to the real-world process of evaluating transactions and interaction partners based on observations about behaviour and context. The reputation management application thus serves as a central repository of historical transaction and reputation information.

When real-world marketplace paradigms are transferred online, complexity increases. The advantages of virtual marketplaces are clear, e.g., reaching massive new markets and increasing the ability to network socially, thus more rapidly spreading reputation information. However, disadvantages also appear. Traditional marketplace threats such as fraud and theft can now be perpetrated on a much larger scale from a position of relative anonymity. Similarly, collusion and reputation tampering scams become harder to detect. Additionally, because it is a virtual marketplace rather than a physical one, consumers must rely on virtual product descriptions and digital images rather than being able to physically inspect an item before purchase, which adds to the potential for various types of fraudulent behaviour. According to a recent Gartner survey, in the second half of 2003, only 4% of hacker attacks were launched at e-commerce sites, whereas in the first half of 2004, 16% were targeted at e-commerce, making it the single most assaulted industry (Lacy 2004). Because the system controls that are in place in some online marketplaces are not as extensive as they would be in the real world, community members may be forced to rely mainly on their trust in each other rather than in the system.

Additionally, in reputation management for Internet auction users, it is difficult to assess the ability of a recommender to recommend another user. This is due mainly to the fact that a typical auction user will have little or no observations of any other given user, i.e., a typical user will not have interacted with the majority of other users. This results in a lack of observations with which to compare recommendation ability. That is, for example, if I have not personally observed Alice's ability as a seller, how can I objectively judge Bob's recommendation of Alice's selling abilities? Furthermore, the sheer magnitude of recommendations forming a user's reputation may lead to scalability problems when trying to integrate a mechanism that allows users to rate a recommender's ability to recommend.

Moreover, in this domain, risk is largely attached to the value of an item or service being sold or purchased, and the risk is highly variable. For example, nearly 95 million eBayers sold approximately $24 billion worth of goods in 2003 (Steiner 2004), and the value of the sold goods in that marketplace ranges from the nearly insignificant, e.g., a pair of sunglasses being sold for one cent, through many value ranges to very high value-based risk, e.g., a Caribbean island. In this environment, risk factors include but are not limited to the rate of virtual marketplace fraud (informed to the entire community via reputation), the value of a transaction, and the context in which an interaction is taking place.

As discussed in the previous chapter, a reputation system is a mechanism that captures the reputation of actors within an application domain. In these systems, reputation is defined as an overall quality or character of a given party as observed or judged by a group. Reputation management, therefore, involves recording observations about a person's actions and the opinions of others about those actions. These records can then be published in order to allow members of the community to make informed decisions about whether to trust a particular person or not.

Reputation management is especially important in virtual communities in which different members come and go daily, and in which most members have only interacted personally with a small fraction of the whole. In virtual marketplaces such as Internet auctions, reputation management consists of two main processes, the recommendation, or feedback, process, and the reputation evaluation process. The recommendation process allows each user to post his opinion about the person with whom he transacted once an auction has completed. A recommendation typically includes the time and date of the recommendation, time and date of the transaction, usernames of recommender and the party being recommended, the item that was the subject of the transaction, and a rating, e.g., positive or negative, about the party being recommended. The evaluation process collates the recommendations about a given user into a recommendation list and forms a reputation score based on the evaluation of the list. Since having primarily positive recommendations will improve a user's reputation and therefore make other users more comfortable in dealing with him, users are encouraged to behave correctly in order to achieve a good reputation.

Thus, a reputation in the Internet auction domain is usually an overall summary of a collection of recommendations about a user, e.g., that user Alice has behaved well in 98% of her interactions with other marketplace members, as well as a list of all recommendations about Alice. For a participant who interacts frequently in a marketplace, this recommendation list may be a collation of thousands of unique recommendations. A user trying to determine whether or not to interact with Alice may be in the position of relying on Alice's reputation summary, which, while more usable than manually processing an arbitrarily large recommendation set, lacks accuracy and provides no contextual relevance to the decision at hand. Moreover, as highlighted in the previous chapter, current research exposes other deficiencies in reputation management, including an absence of processes to make explicit risk management and collusion detection. These deficiencies leave current commercial reputation management systems ill-suited to counter threats that are specific to the virtual marketplace domain, as discussed in the following section.

3.3.3 Domain-Specific Behaviour Taxonomy

In reputation management in a virtual marketplace, a reputation marks the community's judgment as to the trustworthiness of an actor, i.e., seller, buyer, or bidder, based on behaviour exhibited in interactions with that actor. Thus, another role emerges when Internet auctions utilise reputation management systems for enabling feedback about entity trustworthiness, i.e., the role of recommender. Trustworthiness for each of these roles can be evaluated by typifying correct behaviour and evaluating observed behaviour in relation to a taxonomy of possible behaviour classifications. In this section, we propose such a taxonomy of behaviour in virtual marketplaces, Internet auctions in particular, that use reputation management to encourage correct behaviour in interactions. The behaviour types are highlighted in the Venn diagram in Figure 21, and detailed in Table 3.

While the taxonomy includes the classification of 'normal', i.e., correct, behaviour types, it also classifies types of anomalous behaviour that exist in this specific application domain. Where relevant, similarities are highlighted between Internet auction domain-specific behaviour and the more general types of malicious behaviour and system threats classified in Table 2. We note also that some behaviour types are sub-classed according to an entity's role. For example, normal behaviour differs according to role, i.e., a seller typically carries out different duties than a buyer.

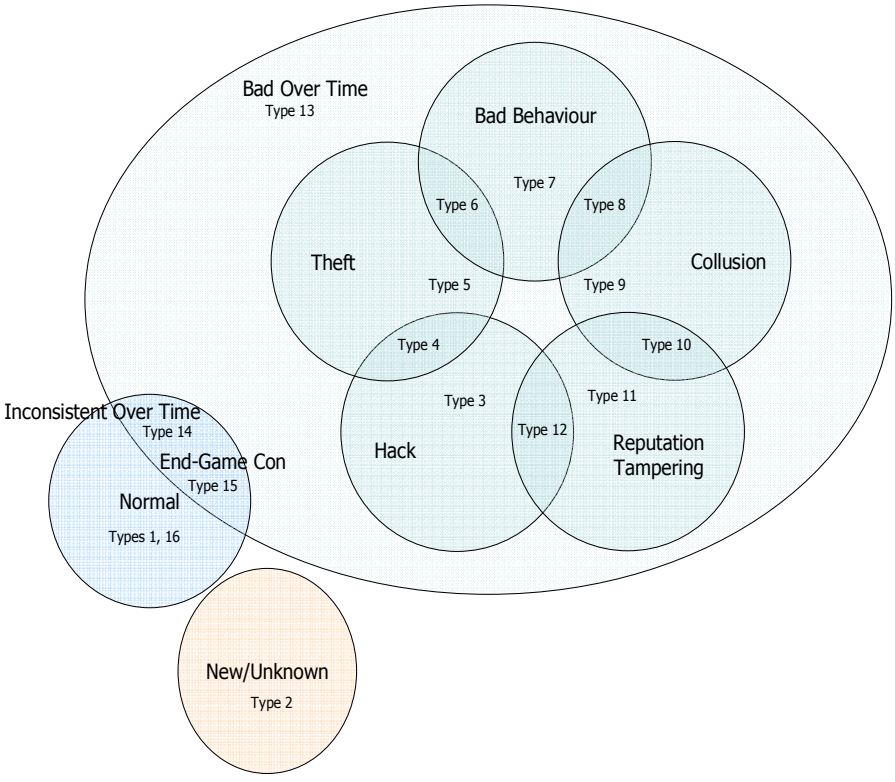

Figure 21: Behaviour types in virtual marketplaces with reputation management

Table 3: Behaviour types in virtual marketplaces with reputation management	
Behaviour Type	Behaviour
Type 1: Normal Behaviour	Seller • Sells a product/service to a buyer. • Accepts bids from a bidder in an auction transaction. • Does not misrepresent self. • Describes item/service correctly. • Ships on time, appropriately packaged. • Adheres to stated return policy. • Communicates appropriately before, during, and after the transaction. • Leaves appropriate feedback. Buyer • Purchases a seller's product/service. • Does not misrepresent self. • Has ability and intention to pay for the product/service. • Pays in full, on time, and payment clears. • Communicates appropriately before, during, and after the transaction. • Leaves appropriate feedback. Bidder • Special case of buyer, seen in online auctions rather than in non-auction e-commerce transactions. • Bids on a seller's product/service. • Does not misrepresent self. • Bids genuinely (i.e., does not display bad behaviour and bids with the intention and ability to pay for the item if bidding is successful). Recommender • Passes a recommendation regarding a seller or buyer with whom he has interacted in the past. • Does not misrepresent self. • Is accurate and truthful.
Type 2: New/Unknown	New entity, seller/buyer/bidder, in the marketplace. No information (feedback/reputation) yet. *Related to Table 2: Newcomer attack. Protracted exploitation of the newcomer attack is a basic Sybil attack.*
Type 3: Hacker	Hacks bid (bid tampering, e.g., seller hacks a bidder's bid to make it look higher). Hacks seller/bidder account. *Related to Table 2: Identity theft.*

Table 3: Behaviour types in virtual marketplaces with reputation management	
Behaviour Type	Behaviour
Type 4: Hacker Thief	Seller sells on hacked account and does not deliver goods. Buyer pays with hacked/stolen credit card/Paypal account. *Related to Table 2: Identity theft and bad guys.*
Type 5: Thief	Buyer receives goods and does not pay. Seller receives payment and does not deliver goods. Actor poses as escrow service to do either of above. Seller accepts return but does not credit buyer. Buyer accepts return payment but does not return item. *Related to Table 2: Bad guys.*
Type 6: Thieving Bad Behaviour	Selling stolen goods, e.g., advertising original/genuine version of MS Office and really selling pirated CD. *Related to Table 2: Bad guys.*
Type 7: Bad Behaviour (General)	Sells counterfeit goods. Sells goods not as described. Spurious bidding. Improper bid retraction. Non-paying bidder/buyer (NPB) wins auction and does not pay. Unwelcome bidder/buyer, for some seller specified criteria of unwelcome. *Related to Table 2: Bad guys.*
Type 8: Colluding Bad Behaviour	Shilling, e.g., seller uses conspirators or alternate identities in order to bid up the prices in his auctions. Bid retraction/default scam, e.g., two bidders collude to result in item being sold for very low price. *Related to Table 2: Collusion clique, collusion with supporters, collusion with camouflage, indirect Sybil attack, and general Sybil attack.*
Type 9: Collusion	Trades on a new/alternate identity after one identity's account is suspended for engaging in bad behaviour. *Related to Table 2: basic Sybil attack.*
Type 10: Colluding Reputation Tampering	Increases positive feedback by trading between conspirators or aliases Launches defamation attack via multiple conspirators or aliases. *Related to Table 2: Collusion clique, collusion with supporters, collusion with camouflage, defamation, indirect Sybil attack, and general Sybil attack.*

Table 3: Behaviour types in virtual marketplaces with reputation management	
Behaviour Type	Behaviour
Type 11: Reputation Tampering	Inappropriate/inaccurate/defamatory feedback. Feedback solicitation, i.e., propositioning entities to engage in interaction for the purposes of enhancing reputation. For example, including language like "Build your feedback score quickly" in the listing title of a very inexpensive item might be considered to be feedback solicitation. After accumulating positive feedback in this way, they might immediately begin selling more expensive items. Feedback extortion, i.e., when a seller or a buyer threatens to leave negative feedback in order to force a result, e.g., a buyer threatening to leave a negative recommendation unless he gets a discount on his purchase. *Related to Table 2: Collusion clique, collusion with supporters, collusion with camouflage, defamation, indirect Sybil attack, and general Sybil attack.*
Type 12: Hacked Reputation Tampering	Hacked feedback database for purpose of falsely increasing or decreasing a entity's reputation. *Related to Table 2: Identity theft.*
Type 13: Bad Over Time	Consistent bad behaviour, e.g., Types 3 – 12, over time, in any of the domain-specific roles.
Type 14: Inconsistent Over Time	Fluctuations between various types of behaviour over time. For example, an eBay PowerSeller may only cheat 2% of the time but still maintain a very good reputation. *Related to Table 2: Oscillation, mixed behaviour, chaotic behaviour, and misconfiguration attacks. It is difficult to determine the motivation behind inconsistent behaviour over time, although, as evidence accumulates, it may be possible to subclass inconsistent behaviour according to one of the more fine-grained attack profiles in the Table 2 correlation.*
Type 15: End-Game Con	Builds up a good reputation over time (Type 16) and then uses the good reputation for a rip-off sale/purchase in a high profit context before discontinuing the account. *Related to Table 2: Waiting attack.* Hacks the account of an actor who has built up a good reputation over time (Type 16) and then uses the good reputation for a rip-off sale/purchase in a high profit context before discontinuing the account. *Related to Table 2: Identity theft.*
Type 16: Good Over Time	Consistent good behaviour (Type 1), for some definition of 'good', over time, in any of the domain-specific roles.

3.4 SECURE for Reputation Management in Internet Auctions

The previous section described reputation management in the virtual marketplace domain, and presented a proposed taxonomy of behaviour in this domain. This section describes the design of a reputation management system based on the SECURE decision-making framework that may be used in Internet auctions to support more accurate decision-making based on the analysis of contextually-relevant evidence about trustworthiness and risk. By supporting better decision-making, we aim to assist users in the avoidance of the risk of interacting with malicious behaviour types.

First, an overview of the SECURE reputation management system for Internet auctions is illustrated. Next, the ways in which context is captured and requests are dealt with by SECURE in this domain are described, as well as the role of entity recognition in the system. Then, the design of the event structures, event configurations, and the *eff* and *eval* functions is presented. Finally, the design of the evidence collection, risk assessment, and access control processes is detailed. Extensions to the SECURE model for application in this domain will be discussed in the section following this discussion of our design decisions.

3.4.1 Design Overview

The SECURE reputation management system for Internet auctions, illustrated in Figure 22 below, interacts with the auction system and its databases to provide interaction security guidelines to an auction user.

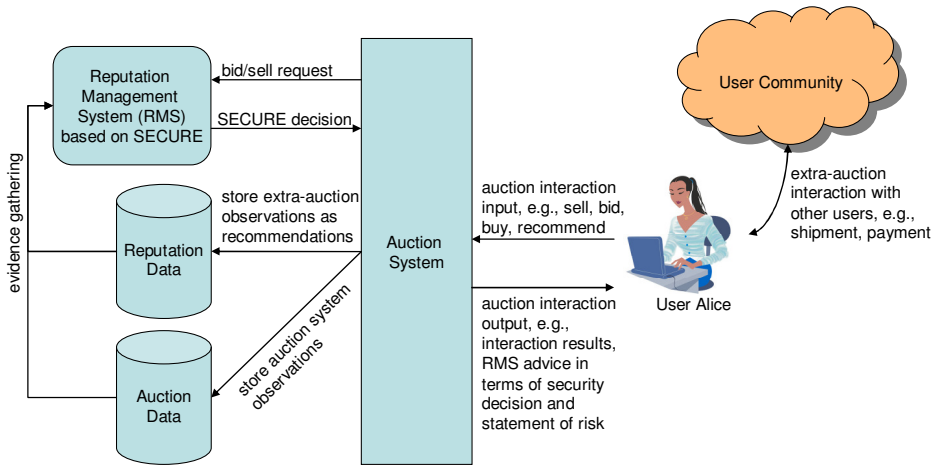

Figure 22: SECURE enhanced reputation management for Internet auctions

The user community consists of members of a given Internet auction who engage in interactions with one another via an auction system. Users participate in an interaction under the guise of one of three roles, i.e., bidder, buyer, or seller. Before engaging in any interaction, a user in any role must be

authenticated by the auction system to prove that he is a member of the Internet auction and therefore authorised to interact. When interacting in a role, an authenticated user, Alice, may perform auction actions that are role-specific. For example, as a bidder, Alice may place bids in a current auction. As a buyer, Alice may purchase goods. In the seller role, Alice may list items for sale and accept bids on current auctions. Observations about these interactions are made by the auction system and recorded.

Moreover, when Alice interacts in a buyer or seller role, she may interact with other users outside of the auction system, i.e., extra-auction interactions, when an auction transaction has completed. Extra-auction interactions consist mainly of the shipment of and payment for the goods transacted upon during an auction transaction. When Alice provides feedback about her observations of an extra-auction interaction, she is acting in a recommender role, and the auction system stores her recommendation in a reputation database. Note that auction-duration interactions may also occur, that is, the dynamic interactions between bidders or a bidder and seller. This is not captured by feedback, however, but perhaps may be detected by assessment of specific system observations. This is discussed later in the context of a SECURE extension for collusion detection.

The auction system accepts user input and outputs interaction results. It also stores evidence in reputation and auction stores, sends bid and sale requests to SECURE for decision-making, and relays SECURE decisions to users. Essentially, SECURE can be integrated to operate in between the auction system and reputation system components to perform additional decision making which is beneficial to the end user.

As illustrated in Figure 15 above, several subcomponents operate within the SECURE TSF lifecycle. To use the SECURE approach when interacting in a Internet auction, at the point of deciding whether or not to interact, i.e., to bid or to accept a bid, a request is made to the SECURE TSF. A request includes a query, some contextual information, and some information about the requesting entity. One subcomponent is a trust module that dynamically computes and updates trust values for actors based on pieces of evidence, i.e., reputation information gathered from the reputation management database of the online marketplace. Another subcomponent dynamically evaluates the risk involved in marketplace interactions. A third subcomponent, the evidence manager, collects evidence from the reputation and auction databases, and filters out the evidence that is relevant for updating trust and risk for a given request, on a request-by-request basis. An access control module then evaluates the trust, risk, and user access control policies and outputs a decision to the decision-making entity. The design of the specific SECURE algorithms and processes for use in Internet auction reputation management is described in detail in the following.

3.4.2 Requests

A request to SECURE consists of three parts: the request query itself, identification information about the principal who is the subject of the request, and information about the context of the interaction being requested. Therefore, in order to design a SECURE request for reputation management in the

Internet auction domain, we must determine what types of decisions are required, what type of principal identification information is necessary, and what context information should be provided as part of the request.

3.4.2.1 Request Types

The type of request query depends on the type of decision a user is trying to make. At the most basic level, a user, represented as a principal, is trying to decide whether or not to interact with another user in an auction. The most general decision to be made in the auction domain is whether or not principal *p* should interact with principal *q*, and thus a general request query would be: 'should *p* interact with *q*'?

In an Internet auction, two, more specific, types of requests are typical and they are tailored according to role context, i.e., 'should *p* sell an item to *q*?' and 'should *p* buy an item from *q*?'. A principal may be deciding whether or not to bid on (with intent to buy) an item for auction, in which case a *bid request* would be made. Alternatively, a principal may be considering whether or not to accept another principal's bid on an item for auction, in which case a *sale request* would be made. Thus, we require two types of requests, a bid request and a sale request.

The request types could be even more finely specified, i.e., 'if I buy from *p*, what is the likelihood that he will send me a counterfeit item?' might be another type of request that could be designed. However, this type of request is very specific and requires the availability of a specific type of evidence. We therefore choose to limit the decision types, and thus the request types, to bid and sale requests, as this captures the majority of decisions users of an Internet auction are interested in making.

3.4.2.2 Identification Information

In current Internet auction systems, a user's identification information usually consists of a username that has been authenticated with a password. In our design, the entity recognition component assigns full confidence to the authenticated username because the username is authenticated by the auction system's own authentication mechanism which is assumed to be reliable.

3.4.2.3 Request Context

Because context is an important part of trust-based decision-making, a request must convey to the SECURE framework the context of the decision the user is trying to make. In the Internet auction domain, context includes information about user role, as discussed with regard to request type; the environment, e.g., category id of the item being auctioned, financial information; and temporal

information. This will be used to filter out evidence that is contextually relevant to the decision being made.

Thus, the following elements comprise request context: the contextual element of role, which is essential to determining request type; the two main environmental factors of item category, determined by the seller at the time of listing an item for sale, and price, which are key parameters in determining contextual relevance of evidence and risk; and the contextual factor of time, such that temporal degradation of evidence is possible.

It is important to note that the context information, i.e., role, environment, price, and time, could be extended to include, for example, more fine-grained item subcategories. Too fine a granularity with regard to item category, however, may limit the amount of evidence that may be used for decision-making. While it is important to assess evidence that is contextually relevant to making a given decision, a very limited evidence set may hamper decision-making.

3.4.2.4 Request Design

When principal *p* is considering placing a bid on an item for sale by principal *q*, *p* initiates a bid request to the SECURE TSF that is comprised of information about request type, identification information, and context, as illustrated in Fig. 23.

Bid request		
Request query	Should I place a bid?	
Identification information	Authenticated username of principal *q*	
Context information	Role context, i.e., *q*'s role	Seller
	Environmental context (category)	Item category number
	Environmental context (price)	Current price of item
	Temporal context	Current date/time

Figure 23: Bid request

On the other hand, if *p* has received a bid from *q* for an item that *p* is selling, *p* initiates a sale request to SECURE, as illustrated in Fig. 24.

Sale request		
Request query	Should I accept a bid?	
Identification information	Authenticated username of principal *q*	
Context information	Role context, i.e., *q*'s role	Buyer
	Environmental context (category)	Item category number
	Environmental context (price)	Current price of item
	Temporal context	Current date/time

Figure 24: Sale request

The design of this request structure captures the necessary information for SECURE decision-making, and is extensible should further application-specific information be required for decision-making.

3.4.3 Entity Recognition

The ER component processes identification information about principal q, e.g., q's authenticated username for the marketplace in which he is participating. Because ER allows for existing recognition schemes to be plugged in to the SECURE TSF, it is compatible with the password-based authentication schemes currently employed by commercial Internet auction applications. We assume that the existing mechanisms are completely accurate for password-based authentication and that this authentication process occurs before a bid or sale request is initiated. Therefore, the ER component allocates full confidence in a requesting principal's identity.

3.4.4 Event Structures

Event structures and their configurations are the basis for representing evidence that SECURE uses to reason about trust. In order to design the most appropriate event structures for reputation management in the Internet auction domain, this section first examines what events occur in typical interactions in existing commercial reputation management systems for Internet auctions and shows how the SECURE event structure can be used to model these events. Next, we describe how the notion of context is incorporated into event structures, depending on entity role. Then, our rationale is presented about what events should be captured in order to enable SECURE to make decisions about whether or not a principal should interact with another principal given the possible behaviour types of principals in this domain. Additionally, we describe event structures that balance granularity levels such that increased accuracy of decision-making might be provided without hindering application usability. Finally, conclusions are presented as to the choice of event structure design.

3.4.4.1 Representation of Events in Existing Reputation Management Systems

In existing reputation management systems, a recommendation is used to record events that occurred in extra-auction interactions between users. Typically, relevant transactional data, such as time and date of interaction, item details, and user role, are appended by the reputation system to the recommendation. The SECURE event structure can model such a recommendation to varying degrees of granularity of the events that are captured by existing reputation systems. For example, Figures 25-28 illustrate event structures ES_E, ES_A, ES_Y, and ES_B that model the observations captured by the eBay, Amazon, Yahoo!, and BizRate reputation systems respectively.

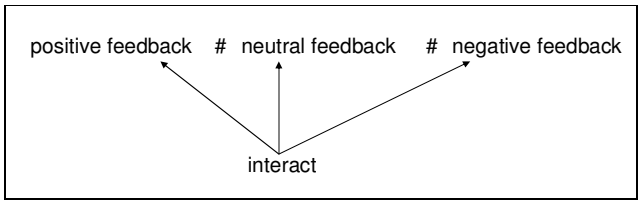

Figure 25: ES_E modelling feedback events in the eBay auction reputation system

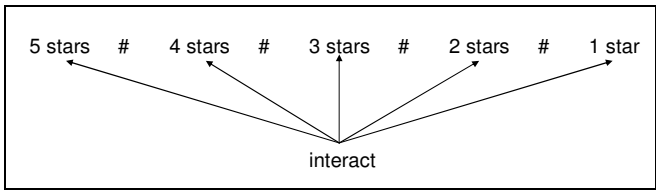

Figure 26: ES_A modelling feedback events in the Amazon auction reputation system

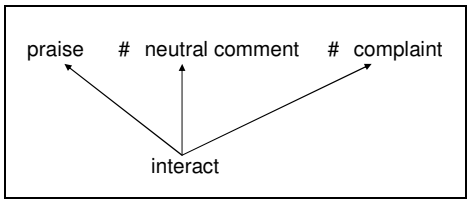

Figure 27: ES_Y modelling feedback events in the Yahoo! auction reputation system

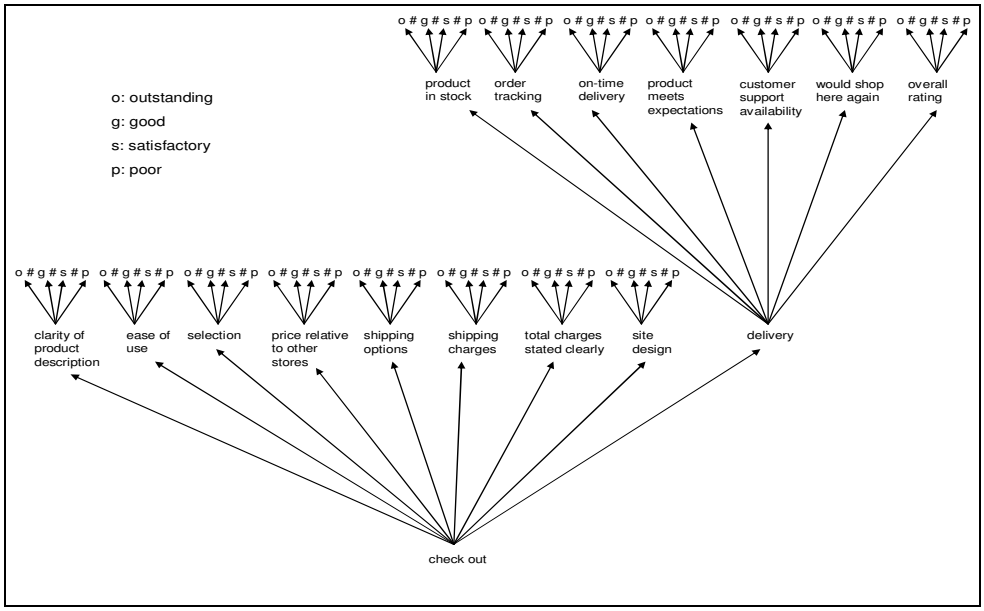

Figure 28: ES_B modelling feedback events in the BizRate merchant rating reputation system

In the case of the eBay, Amazon, and Yahoo! reputation systems, the event structures illustrate that only a basic level of observed behaviour is captured by a recommendation about extra-auction interaction. In the eBay and Yahoo! systems, the observation recorded by a recommendation is simply whether or not the interaction was positive, neutral, or negative. That is, *p* observes whether an interaction with *q* is positive, neutral, or negative, and records this outcome in a recommendation. Similarly, in the Amazon system, a recommendation captures whether the interaction received a rating of 1 – 5 stars, with 5 stars representing that *p* observed 'excellent' behaviour by *q* during the interaction, 4 stars representing 'good' behaviour, 3 stars representing 'fair' behaviour, 2 stars representing 'poor behaviour', and 1 star representing 'awful' behaviour. The BizRate reputation system, perhaps because it is used to rate online merchants rather than auction participants, captures a far more precise level of observation in a recommendation, including both pre- and post-sales behaviour of the vendor. Although this type of recommendation allows for significantly more detail in recording observations throughout the course of an interaction, it also demands significantly more time on the part of the user when relaying feedback about an interaction.

It is important to note that, should a user of any of the current Internet auctions have unlimited time and ability to manually process an arbitrarily high number of individual recommendations from the reputation recommendation list, it is possible to glean more information from each individual recommendation than just the overall rating of positive, neutral, or negative. As mentioned above, a typical recommendation in a recommendation list includes not only the rating, but also context information regarding role, time/date, item category, and price. However, in forming a reputation from the recommendation lists, all current Internet auction applications employ reputation systems that only summarise rating information rather than assessing the recommendations that are contextually relevant to a future potential interaction to provide a reputation statement about a given member.

3.4.4.2 Event Structure Type with Regard to Context

A buyer's behaviour during an interaction is different than a seller's behaviour, and we therefore propose that, as with request types, two different event structure types are required by a SECURE reputation management system, i.e., differentiating events according to role. Therefore, we define two different event structures for the representation of possible interactions, one to model the events that may occur when a seller sells an item and one to model the events that may occur when a buyer buys an item.

In addition to role, we have determined that other context types are important to the SECURE decision-making process, i.e., time, price, and item category. Rather than encoding this into an event structure, however, this information will be appended to recommendations and used by the SECURE components that deal with analysing contextual relevance and risk. This allows the event structures to remain generic enough to use in many contexts. If we had instead encoded all context types into an event structure, cases could arise where an event structure was too limiting to model events across

different contexts, i.e., events that might be observable in one context may not be observable in another.

3.4.4.3 Events Necessary to Capture Trust

Taking on board the types of Internet auction domain threats against which we are trying to protect, we note that, in addition to normal behaviour, the areas that should be targeted for observation are those that can assist in the assessment of potential fraud, theft, and collusion, i.e., the behaviour types described in Table 3 above. The events necessary to characterise this behaviour are specified as follows.

It may be observed that the outcome of an interaction with a seller is a set of events that characterises normal behaviour, as defined in Table 3. This set of events corresponds with the normal selling behaviour described by Types 1 and 16. At a coarse-grained level, one might observe two events, i.e., that a seller interacts with a buyer and that there is a positive outcome. At a more fine-grained level, one might observe a series of events, e.g., a seller accepts a buyer's high bid in an auction, communicates well with that buyer, packages the item well, ships the item, ships the item in a timely manner, ships an item that is as it was described in the auction details, etc. The evidence that is needed for events that describe this behaviour are gleaned from extra-auction user feedback.

If, however, no feedback records exist about a user's selling behaviour, one can observe nothing about that behaviour, which relates to Type 2.

If a seller does not ship an item for which a buyer has paid, then he is exhibiting thieving behaviour, i.e., Type 5. This means that it would be useful to be able to observe the events that a seller accepted a buyer's winning bid on an item and did not ship the item. Again, the evidence about the events that describe this behaviour would be taken from extra-auction user feedback.

Other types of events that allow us to characterise seller behaviour include observations that a seller has delivered counterfeit, broken, or otherwise not-as-described merchandise, or that a seller has delivered stolen merchandise. These events characterise behaviour Types 6, 7, and 13. However, we assume that it would be difficult for the recipient of merchandise to know whether or not the merchandise was truly stolen, and so we dismiss the creation of an event for this type, i.e., Type 6.

Furthermore, there may be the case, i.e., Type 14, where, having observed several interactions with a given seller, it is observed that a seller is inconsistent over time, or oscillates between normal behaviour, i.e., Type 1, and bad behaviour, i.e., Types 7 and 13. We note that in this instance, it is not necessary to create a new event type in an event structure.

On the buyer side, we also require events that capture normal behaviour, e.g., interacts with a seller and pays for an item won in an auction, as well as newcomer behaviour, bad buyer behaviour, i.e., does not pay for an item that a seller has shipped, and oscillating behaviour. As in the case of the

seller, the evidence about the events that describe a buyer's behaviour would be taken from extra-auction user feedback.

In each of the cases thus far, the evidence used to populate potential event structures is based on extra-auction observations that are stored as recommendations in the reputation management system. To identify other types of behaviour, however, such as collusion between a seller and bidder(s), i.e., Types 8 and 9, it is necessary to describe auction duration events between user pairs. Auction duration event observation is different than the rest because events are observed by the auction system during an auction, and it requires a special mechanism that is described in the SECURE extension.

Furthermore, malicious behaviour that involves reputation-tampering, i.e., Types 10-11, requires observations about user pairs such that recommendation integrity can be determined and this behaviour is protected against by a recommendation weighting mechanism that is described in the SECURE extension.

Anomalous behaviour such as hacking for the purposes of stealing or reputation-tampering, i.e., Types 3 and 4, and 12, comprise threats that are addressed by conventional security mechanisms such as authentication methods and therefore we do not incorporate events regarding these behaviour types into our event structure. Similarly, it is impossible to predict an end-game con, i.e., Type 15, based on observed behaviour in this domain.

Finally, recent research (Anderson 2005) shows that, out of the possible types of malicious behaviour observed via Internet auction complaints, 'item not received' comprised 74.1%, 'quality of the item' complaints amounted to 16.1%, and 'payment not received' was noted 5.4% of the time. Complaints about other types of malicious behaviour comprised only 4.5% of the total. Therefore, we focus our reputation system design on capturing evidence and identifying patterns for behaviour of these three main types.

3.4.4.4 Granularity of Events

As illustrated by the modelling of existing reputation system events using the SECURE event structure, our event structure is capable of modelling varying levels of interaction granularity. Therefore, it is possible to design a recommendation that captures a finer level of granularity about the events of an interaction than is currently used in commercial reputation systems, thus allowing an increase in the accuracy of evidence about observed behaviour. However, at the same time our system aims to maintain usability, i.e., we do not wish to further disincentivize a user by forcing the use of a very detailed feedback mechanism. Moreover, we wish to be able to perform quantitative analysis of evidence rather than subjecting a user to qualitative evidence that, in bulk, cannot be manually processed.

Therefore, our event structure design must capture the key events that would suggest a pattern of malicious behaviour. To do this, a typical positive-or-negative recommendation is extended in part by

adding minimal additional user input into the recommendation process. The event structure that captures this extended recommendation will allow for increased accuracy in decision-making while maintaining usability in the feedback loop.

3.4.4.5 Construction of Event Structures and their Configurations

Based on the above rationale, appropriate SECURE event structures, and their resultant configurations, might be as follows.

ES_{RS}, the event structure that models the observations a user in the buyer role can make about a user in the seller role once an auction interaction has completed, is illustrated in Figure 29 below. When p has completed an interaction as a buyer with seller q, p can observe that q either ships or does not ship the auctioned item. If q does ship the item, p can observe whether the item met expectations, i.e., was as described in the auction, or whether the item was counterfeit, broken, or otherwise not as described.

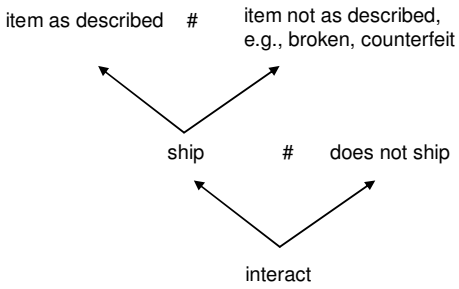

Figure 29: ES_{RS} modelling observations about a seller by a buyer

ES_{RB}, the event structure that models the observations a user in the seller role can make about a user in the buyer role once an auction interaction has completed, is illustrated in Figure 30 below. When p interacts as a seller with buyer q, p can observe that q either pays or does not pay for the auctioned item.

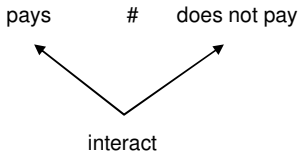

Figure 30: ES_{RB} modelling observations about a buyer by a seller

Clearly, in each case, over the passing of time during and after the auction, *p* could also observe other behaviour, such as *q*'s communication behaviour, return policy of the seller, etc. In order to maintain usability of the recommendation component of the reputation system, however, we isolate extra-auction observations that will help us protect against fraud, and theft, and leave out other potential observations, e.g., whether *p* communicates well, that would be disincentivizing to a user to be forced to comment on when recording feedback.

We recall that an *event configuration (C_{ES})* models the possible states in which a principal may be regarding its knowledge about an interaction. The event configuration, $C_{ES_{RS}}$, models the possible configurations of ES_{RS}, and is illustrated in Figure 31. (Abbreviations of event names are used in the event configuration diagrams.)

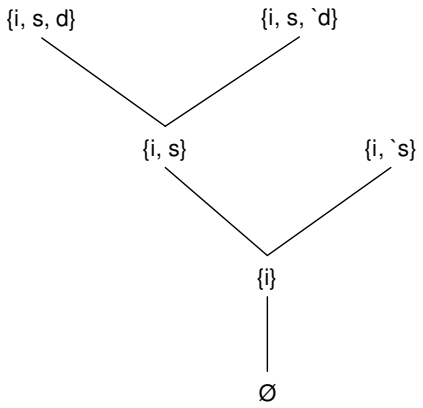

Figure 31: $C_{ES_{RS}}$ modelling the possible configurations observable by a buyer about a seller

In $C_{ES_{RS}}$, the following event configurations are observable by buyer *p* about seller *q*

Ø *p* has never interacted with *q* and has no observations about *q*

{i} *p* and *q* interacted in an auction, i.e., *q* sold an item to *p*

{i, s} *q* sold an item to *p*, and *q* shipped the item

{i, `s} *q* sold an item to *p*, and *q* did not ship the item

{i, s, d} *q* sold an item to *p*, and *q* shipped the item, and the item received by *p* was as described, thus meeting *p*'s expectations

{i, s, `d} *q* sold an item to *p*, and *q* shipped the item, and the item received by *p* was not as described, e.g., counterfeit, non-functioning, broken, etc.

Similarly, the event configuration, $C_{ES_{RB}}$, models the possible configurations of ES_{RB}, and is illustrated in Figure 32.

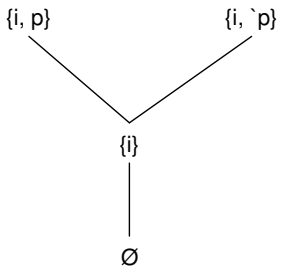

Figure 32: $C_{ES_{RB}}$ modelling the possible configurations observable by a seller about a buyer

In $C_{ES_{RB}}$, the following event configurations are observable by seller *p* about buyer *q*

Ø *p* has never interacted with *q* and has no observations about *q*

{i} *p* and *q* interacted in an auction, e.g., *p* was the high bidder and bought an item from *q*

{i, p} *p* bought an item from *q*, and *q* paid for the item

{i, `p} *p* bought an item from *q*, and *q* did not pay for the item

Given the event configurations $C_{ES_{RS}}$ and $C_{ES_{RB}}$, we can determine which configurations will best support the trust calculation process in supplying a decision for the Bid and Sale requests respectively. To provide a trust value for *q* when a Bid Request is initiated, *p* is interested in the proposition that *q* will behave correctly in the seller role. Evidence supporting this proposition maps to {i, s, d} in $C_{ES_{RS}}$, while evidence that contradict this proposition include all observations of {i, `s} and {i, s, `d}. (If *p* were specifically concerned with the risk, e.g., of non-shipment by *q*, the proposition might address the likelihood of {i, `s} occurring instead.) All observations of {i} are inconclusive with regard to the proposition.

3.4.4.6 Conclusions Regarding Event Structures

In this section, we modelled event structures, and their resulting event configurations, for a reputation management system that allows for the capture of more fine-grained observation in key areas of an Internet auction application, thus allowing more accurate reputation-based decisions to be made. Moreover, two event structures are defined based on role context for this domain, such that the events possible to observe during an interaction are more accurately reflected by contextually independent

event structures. Finally, while increasing accuracy through the incorporation of more fine-grained observations and context, usability is maintained in the recommendation loop by choosing not to construct an event structure that would require significantly finer-grained user feedback than users are currently accustomed to.

We note that this event structure pair captures extra-auction observations, i.e., observations *p* makes about *q* outside of the auction system which *p* then records as a recommendation through the feedback mechanism of the auction's reputation management system. Auction-duration observations, e.g., if *q* placed a bid of a certain amount at a certain time in *q*'s auction, would not only be significantly difficult for a user to manually record feedback about, but also form part of the interaction dynamics between two users, and these observations are incorporated into the SECURE extension regarding collusion detection, described below.

3.4.5 Evidence

In order to update evidence represented by the event structures and their configurations, evidence about extra-auction interactions must be designed. Having determined the design of the event structures for the SECURE reputation management system, it is straightforward to determine the design of a feedback record that will provide an evidence reporting structure.

Feedback, or the recording of one's observations for recommendation to the user community, is different according to whether one is recording observations about an interaction with a seller or with a buyer. For an observation about a seller, one can record the observation that the seller shipped an item or did not ship an item, and that if the item was shipped, whether it was as described or not as described. Given $C_{ES_{RS}}$, it is possible to observe, and therefore report through feedback, {i}, {i, s}, {i, `s}, {i, s, d} or {i, s, `d}. Similarly, for an observation about a buyer, one can record that the buyer paid or did not pay for an item won in auction, and, given $C_{ES_{RB}}$, feedback may consist of {i}, {i, p}, or {i, `p}. When an auction completes, {i} is recorded by the auction system as a default piece of feedback to show that two users have interacted so that it does not have to be done manually.

We note that evidence provided by the extra-auction feedback loop corresponds precisely to our event structures. We note also that each event in the feedback record is true or false, giving a binary result, thus circumventing the difficulty of analysing qualitative feedback data such as a textual statement about the trustworthiness of an interaction partner.

3.4.6 The Evidence Gathering Process

Evidence about the trustworthiness of a seller or buyer is provided via feedback, i.e., the recording of one's extra-auction observations. The feedback mechanism is part of the auction system, and once a

user has recorded his observations, the auction system manages the appending of context information about the transaction to which the feedback refers, as well as the storing of the evidence in the reputation database.

SECURE's evidence manager accesses evidence from the reputation and transaction databases and updates an interaction history with relevant evidence when requested to do so by the TLM.

Moreover, risk data may be gathered from the Internet auction application domain, and is used to update the risk configuration component, e.g., the risk of interacting in one context, independent of interaction partner, may be different than that of interacting in another context.

3.4.6.1 Updating Interaction Histories with New Evidence

We recall that an interaction history is a finite ordered sequence H of configurations: $H = x_1\, x_2 \ldots x_n$, where recorded observations are the individual components x_i in history H. A sample interaction history is given denoting p's past interactions with q:

$$H_{pq} = \{i,\ s,\ d\}\{i,\ s,\ d\}\{i,\ s,\ d\}\{i,\ s,\ d\}\{i,\ s,\ d\}\{i,\ s,\ \grave{}d\}\{i,\ p\}\{i,\ p\}\{i\}$$

H_{pq} is updated with new evidence each time p observes a new event about q and records it. Note that an interaction history contains all evidence about a principal, regardless of context. Note also that for each H_{pq}, there will also be an interaction history H_{qp} that contains configurations of q's observations about interactions with p.

3.4.6.2 Evidence Context

Each observation in the form of a piece of evidence of an interaction history is associated with context information according to role, environmental, and temporal parameters.

By modelling ES_{RS} and ES_{RB}, role context is captured in the event structures to provide a subset of more contextually relevant evidence with which to assess trust, but to make this explicit, each piece of evidence is tagged according to the role context of the user about to whom it pertains. An interaction history is further associated with both environmental and temporal context, and we annotate it as follows:

$$H_{pq} = \{i,\ s,\ d\}_{rct}\{i,\ s,\ d\}_{rct}\{i,\ s,\ d\}_{rct}\{i,\ s,\ d\}_{rct}\{i,\ s,\ \grave{}d\}_{rct}\{i,\ p\}_{rct}\{i,\ p\}_{rct}\{i\}_{rct}$$

In this model of an interaction history, r refers to role (buyer or seller) of q in the interaction the configuration describes, c refers to item category, and t refers to the date and time, i.e., a timestamp, of the interaction about which the observation is made.

3.4.6.3 Recommended Evidence

Thus far, we have described only observed evidence, that is prima facie evidence. Each user has an interaction history containing observations about each user with whom he has interacted. Because feedback is made available to the entire Internet auction community, we can also say that the auction system records an interaction history for each user pair. A given user q's total reputation consists of all of the interaction histories pertaining to q. When user, p, is making a decision about whether or not to interact with q, he consults his own interaction history of observations about q, as well as others' interaction histories, which p treats as recommendations about q. The sum total of evidence about q, then, is an interaction history containing both observations and recommendations. To model this, we annotate H_{pq}, according to which user, u, is the provider of evidence, e.g.:

$$H_{pq} = \{i,\ s,\ d\}_{rctu}\{i,\ s,\ d\}_{rctu}\{i,\ s,\ d\}_{rctu}\{i,\ s,\ d\}_{rctu}\{i,\ s,\ d\}_{rctu}\{i,\ s,\ `d\}_{rctu}\{i,\ p\}_{rctu}\{i,\ p\}_{rctu}\{i\}_{rctu}$$

In this model of an interaction history, r refers to role (buyer or seller), c refers to item category, and t refers to the date and time, and u represents the provider of the evidence about the interaction about which the observation refers. In the above example, if $u = p$, then the piece of evidence is treated as p's observation about q, and if u refers to a username other than that of p, then the piece of evidence is treated as a recommendation from that user about q.

However, while SECURE allows for the possibility of discounting, or weighting, recommended evidence, it does not provide a mechanism for assessing recommendation integrity. Therefore, for the remainder of this section, we assume that evidence is comprised entirely of observations, and we extend SECURE to include a recommendation integrity assessment method in the following section.

3.4.7 The *eff* Function

The effect, or *eff*, function calculates the effect a piece of evidence will have on a trust value. That is, *eff* determines which pieces of evidence support, contradict, or are inconclusive about the likelihood that a new interaction will result in a specified way. We extend the *eff* function to also determine which pieces of evidence from an interaction history are contextually relevant to a request. The extended *eff* function is defined as:

$$\text{eff}_{x_{rc}}(w_{rc}) = \begin{cases} (1,0,0) \text{ if } w_{rc} \subseteq x_{rc} \\ (0,0,1) \text{ if } x_{rc} \# w_{rc} \\ (0,1,0) \text{ if neither } w_{rc} \subseteq x_{rc} \text{ nor } x_{rc} \# w_{rc} \end{cases}$$

If x_{rc} is a configuration that has been observed when q has interacted with p, q may be considering another interaction with p which will result in proposition w_{rc}. To estimate the likelihood of the interaction resulting in w_{rc}, given that the last interaction resulted in configuration x_{rc}, three effects are possible in *eff*: if $w_{rc} \subseteq x_{rc}$, then the fact that x_{rc} occurred in an interaction supports the likelihood of

the occurrence of w_{rc}, i.e., x_{rc} contains all of the events in w_{rc}; if x_{rc} # w_{rc}, then x_{rc} contains an event that contradicts the occurrence of w_{rc}; if neither $w_{rc} \subseteq x_{rc}$ nor x_{rc} # w_{rc}, then x is inconclusive about w_{rc}. Note that the *eff* function also assesses context in terms of role and item category to determine which evidence in an interaction history meets the contextual relevance rules. If a piece of evidence does not correspond to the role, r, or item category, c, specified in the request, it is rejected. Evidence timestamps are not assessed for timeliness at this stage, but rather in the next step of evaluation.

For example, when a bid request is initiated, p is interested in the proposition that q will behave correctly in the seller role, s, for a given item category, e.g., *c4*. Evidence supporting this proposition maps to $\{i, s, d\}_{sc4}$, while evidence that contradicts this proposition include all observations of $\{i, \grave{s}\}_{sc4}$, and $\{i, s, \grave{d}\}_{sc4}$. All observations of $\{i\}_{sc4}$ are inconclusive with regard to the proposition. Evidence that relates to q acting as a buyer or interacting in another item context is not deemed to be contextually relevant to make a decision for this specific bid request. Given the following interaction history,

$$H_{pq} = \{i, s, d\}_{sc4}\{i, s, d\}_{sc4}\{i, s, d\}_{sc1}\{i, s, d\}_{sc2}\{i, s, d\}_{sc3}\{i, s, \grave{d}\}_{sc4}\{i, p\}_{bc4}\{i, p\}_{bc3}\{i\}_{sc4}$$

two observations, $\{i, p\}_{bc4}$ and $\{i, p\}_{bc3}$, are not contextually relevant due to role context; and two further observations, $\{i, s, d\}_{sc1}$, $\{i, s, d\}_{sc2}$, and $\{i, s, d\}_{sc3}$, are not contextually relevant due to item category context. Of the remaining, contextually relevant configurations, i.e.,

$$H_{pq_{sc4}} = \{i, s, d\}_{sc4}\{i, s, d\}_{sc4}\{i, s, \grave{d}\}_{sc4}\{i\}_{sc4}$$

two observations of $\{i, s, d\}_{sc4}$ support proposition $\{i, s, d\}_{sc4}$, one observation of $\{i, s, \grave{d}\}_{sc4}$ contradicts the proposition, and one observation, $\{i\}_{sc4}$, is inconclusive.

Thus, the evidence being analysed to calculate trust is the contextually relevant subset of all evidence. The *eff* function is performed on all evidence as part of the calculation done by the *eval* function described below.

3.4.8 The *eval* Function

The *eval* function is used to map an interaction history to a trust value. The evaluation function is defined as:

$$\text{eval}\left(x_{1_{rc}}, x_{2_{rc}} \ldots x_{n_{rc}}\right) = \lambda w_{rc}.\left(\sum_{i=1}^{n} \text{eff}_{x_{i_{rc}}}\left(w_{rc}\right)\right)$$

Thus, the evaluation of a particular interaction history returns a (s,i,c)-triple that comprises the sum of updated evidence in favour of, contradicting, and inconclusive about all observed event

configurations that are contextually relevant for role and item category. Returning to the earlier example, given interaction history $H_{pq_{sc4}} = \{i, s, d\}_{sc4}\{i, s, d\}_{sc4}\{i, s, \grave{}d\}_{sc4}\{i\}_{sc4}$, there is evidence of two observations of $\{i, s, d\}_{sc4}$ supporting the proposition *{i, s, d}sc4*, one observation of $\{i, s, \grave{}d\}_{sc4}$ contradicting the proposition, and one observation, $\{i\}_{sc4}$, that is inconclusive. Applying the *eval* function to *p*'s observations about *q* produces the trust value $T_{OV_{pq\,sc4}}$ = (2, 1, 1) for *q* in the roll of seller of an item in item category 4 for all temporal contexts.

3.4.8.1 Incorporating Time Fading

The *eval* function is configurable so that it is able to calculate a trust value based on a subset of interactions from the interaction history as well as permitting the weighting of interactions. However, no time fading algorithms have yet been designed for SECURE. Therefore, we configure the *eval* function such that time fading is introduced according to the following method:

$$\begin{aligned}
&\text{eval}\left(x_{1_{rct_0}}, x_{2_{rct_0}} \ldots x_{n_{0\,rct_0}}, x_{1_{rct_1}}, x_{2_{rct_1}} \ldots x_{n_{1\,rct_1}}, \ldots x_{1_{rct_m}}, x_{2_{rct_m}} \ldots x_{n_{m\,rct_m}}\right) \\
&= \text{eval}\left(\delta^m x_{1_{rct_0}}, \delta^m x_{2_{rct_0}} \ldots \delta^m x_{n_{0\,rct_0}}, \delta^{m-1} x_{1_{rct_1}}, \delta^{m-1} x_{2_{rct_1}} \ldots \delta^{m-1} x_{n_{1\,rct_1}}, \ldots \delta^0 x_{1_{rct_m}}, \delta^0 x_{2_{rct_m}} \ldots \delta^0 x_{n_{j\,rct_m}}\right) \\
&= \lambda w_{rct_m}\left(\sum_{j=0}^{m} \delta^{m-j}\left(\sum_{i=1}^{n_j} \text{eff}_{x_{i_{rct_j}}}\left(w_{rct_m}\right)\right)\right)
\end{aligned}$$

Where the temporal context of each piece of evidence is annotated as t_j, and *j* is the time step associated with a given timestamp, e.g., evidence can be grouped according to time steps by hour, day, month, year. The time step representing the current time is *m*, i.e., the time at which the proposition w_{rct_m} is being evaluated. δ is the time fading factor. Note that this formula takes into account that more than one interaction might be recorded at the same time step about a user interacting in a given role and item category. Using our formula, the weight of older evidence is faded according to $\delta \in [0,1]$ so that the most recent evidence counts more heavily toward assessment of trustworthiness based on current behaviour. In fact, evidence that falls into time step *m*, the current time period, is not faded at all because it is considered to be as recent as possible.

In the following example, interaction history, $H_{pq_{sc4t}}$, is given, in which *p* is evaluating *q*'s trustworthiness as a seller of items in item category 4 and for which time of interaction is recorded.

$$H_{pq_{sc4t}} = \{i, s, d\}_{sc4t_0}\{i, s, d\}_{sc4t_0}\{i, s, \grave{}d\}_{sc4t_0}\{i\}_{sc4t_0}\{i, s, d\}_{sc4t_1}\{i, s, d\}_{sc4t_1}\{i, s, \grave{}d\}_{sc4t_1}\{i\}_{sc4t_1}$$

If no time fading were applied, i.e., δ = 1, the result of the evaluation would be a trust value, contextually relevant for role and item category, $T_{OV_{pq\,sc4t}}$ = (4, 2, 2). We now apply time fading set at δ = .99 to this evidence as follows. In this example, evidence is being assessed from two time period groups, t_0 and t_1. The oldest evidence, at time step t_0, evaluates to (1.98, .99, .99) when faded. The

most recent evidence, at time step, t_1, is in the same time step as proposition w_{rct_1}, and is not faded, evaluating to (2, 1, 1).

$$\begin{aligned}
&\text{eval}\left(x_{1_{sc4t_0}}, x_{2_{sc4t_0}}, x_{3_{sc4t_0}}, x_{4_{sc4t_0}}, x_{1_{sc4t_1}}, x_{2_{sc4t_1}}, x_{3_{sc4t_1}}, x_{4_{sc4t_1}}\right) \\
&= \lambda w_{rct}\left(\delta^1 \sum_{l=1}^{4} \text{eff}_{x_{i_{sc4t_0}}}\left(w_{sc4t_0}\right) + \delta^0 \sum_{k=1}^{4} \text{eff}_{x_{k_{sc4t_1}}}\left(w_{sc4t_1}\right)\right) \\
&= (3.98, 1.99, 1.99)
\end{aligned}$$

Therefore, after all of the evidence has been faded, $T_{OV_{pq\,sc4t}}$ = (3.98, 1.99, 1.99). The oldest evidence has been time faded the most heavily, and the newer evidence therefore weighs and contributes more toward the assessment of *q*'s present trustworthiness.

We note that the outcome of Jøsang et al.'s experiments with the Beta Reputation System (Jøsang and Ismail 2002; Jøsang, Hird et al. 2003) showed that the most reasonable results of time fading were produced when δ was set at .99. Moreover, when defining the period of time steps, it is most effective when specified at the expected rapidity of behavioural change.

3.4.9 Risk Assessment

According to the Federal Trade Commission (Anderson 2005) and Internet Fraud Watch (Fraud.org 2006) , the most frequently reported form of Internet fraud is that related to fraudulent schemes appearing on online auction sites. During 2004, Internet auction fraud comprised 71.2% of referred fraud complaints, a 16.7% increase from 2003 (Federal Bureau of Investigation (FBI) and National White Collar Crime Center (NW3C) 2006). These schemes typically purport to offer high-value items at significantly reduced prices so as to attract many consumers. Victims are induced to send money for the promised items, but then the seller delivers nothing or only an item far less valuable than what was promised e.g., counterfeit or altered goods. The SECURE risk assessment mechanism might be used to assess the impact, in terms of cost, of such an event occurring.

First, there are a number of acts, *X*, available to a SECURE decision-maker in the Internet auction environment. As outlined above, a seller may either decide to accept or reject a bid from a particular buyer, i.e., X_1 = {*accept_bid, reject_bid*}; and a buyer may decide to either place or not place a bid for an item/service with a given seller, i.e., X_2 = {*bid, don't_bid*}.

Second, in an Internet auction application, there are certain mutually exclusive states, *Z*, available to Nature. The most basic result of an auction is an interaction involving payment and delivery. That interaction results in a positive or negative state, which is recorded as feedback according to the specified event configurations. Each principal, having decided to engage in an interaction, may act in a positive or negative manner, wherein positive and negative behaviour can be defined according to various ranges of granularity, as illustrated in the discussion about event structures. For example, if a

seller accepts a high bid from a buyer, that buyer may choose to pay or not pay, i.e., Z_1 = {*pay, not_pay*}. Similarly, if a buyer chooses to bid on a seller's item and wins the auction, the seller may ship the item as described, may ship an item that does not meet the description, or may not ship the item at all, i.e., Z_2 = {*ships_as_described, ships_not_as_described, does_not_ship*}.

Next, we give two cost matrices below, expressing the results of the consequence function, i.e., sample costs under all possible combinations of acts and states.

X_1/Z_1	pay	not_pay
accept_bid	0	current_price
reject_bid	0	0

X_2/Z_2	ships_as_described	ships_not_as_described	does_not_ship
bid	0	current_price	current_price
don't_bid	0	0	0

A probability function then expresses the decision-maker's beliefs about the likelihood of an outcome occurring. The SECURE risk assessment mechanism takes a trust value, T_{ov}, as input. T_{ov} provides evidence about the likelihood of the interaction outcome being, e.g., {*i, s, d*} for *q* in the role of a seller in item category 4 at time, *t*. This process exposes the information necessary to accurately derive risk in the risk engine.

The SECURE model incorporates a utility function at this stage, in order to assess the desirability of each given outcome. However, utility is highly subjective, and, in the Internet auction domain, is likely to differ greatly according to user. Rather than incorporating a utility function, therefore, we propose the use of an alternative risk assessment mechanism, i.e., the standard risk assessment calculation used by the information security community where risk is a function of the probability of a given outcome occurring and the amount of potential financial loss should that outcome occur. Therefore, at its most basic level, the overall risk in an Internet auction interaction is a function of the probability of a negative outcome occurring based on the trustworthiness of an interaction partner, *p*, and the amount of potential financial loss (current price), *l*, i.e., $R \equiv (p_t, l)$. For example, *p* considers bidding in an auction in which *q* is the seller of an item in item category 4 for which the cost context is \$100. $T_{OV_{pq\,sc4t}}$ = (2, 1, 1). The risk of a negative outcome occurring is 25%, i.e., there is a 25% likelihood of *p* incurring a negative outcome, that is, an outcome that is not {*i, s, d*}, with an expected financial loss of \$100. Traditionally in the cost-benefit analysis domain, risk is measured as $R = (p \times l)$. Using this formula in our example, then, *p* is exposing himself to a risk of $.25 \times \$100$, or \$25, by entering into an interaction with *q* given $T_{OV_{pq\,sc4t}}$. A statement of financial risk can then be provided as output to the user.

3.4.9.1 Incorporating Context-Based Risk

Although the seller-related likelihood of fraud occurring can be captured by analyzing the seller's trust value, it is also important to note that some categories, or contexts, of goods being auctioned online are more high-risk, regardless of which seller a buyer interacts with. For example, in a recent lawsuit against eBay, luxury jeweller Tiffany & Co. alleged that 73% of the 'Tiffany' jewellery sold on eBay in 2004 was counterfeit, 5% of it was genuine, and the rest was promoted as 'Tiffany-like' but not promoted as genuine (Reuters 2004). Similar cases demonstrate consumer electronics, sports memorabilia, and luxury handbags, among others, as categories with a higher risk of fraud. Unfortunately, in many of the categories in which counterfeiting is rampant, specific sellers may have good reputations because buyers are unable to distinguish, e.g., a real Louis Vuitton handbag from a fake one.

Therefore, in such categories, the SECURE risk assessment could be better informed by adding a context-based risk parameter to the calculation. To do so, the item context parameter is re-used. Therefore, the risk is evaluated as a function of likelihood of loss with respect to the trust value, likelihood of loss with respect to the item context, and the value of potential financial loss, i.e., $R \equiv (p_t, p_c, l)$.

We extend the previous example: *p* considers bidding in an auction in which *q* is the seller of an item in item category 4, i.e., Tiffany jewellery, for which the current price is \$100 and $T_{OV_{pq\,sc4t}}$ = (2, 1, 1). Therefore, $R(pq_{sc4t}) \equiv (.25, .73, \$100)$.

In order to arrive at a measure of expected financial loss in this dual-risk environment, we first configure a decision tree, illustrated in Fig. 33, which allows us to examine the different combinations of possible outcomes in order to make the most logical decision.

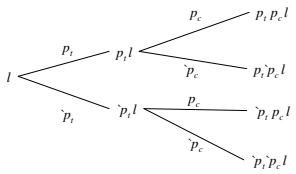

Figure 33: Risk decision tree

The decision tree illustrates that there are four possible combinations of likelihood factors: the likelihood of a good outcome in both trustworthiness and context, $p_t p_c$; the likelihood of a good outcome based on trustworthiness and a bad outcome based on context, $p_t \grave{p}_c$; the likelihood of a bad outcome based on trustworthiness and a good outcome based on context, $\grave{p}_t p_c$; and the likelihood of a bad outcome in both trustworthiness and context, $\grave{p}_t \grave{p}_c$. We determine risk of financial loss

according to the complement of the dual-good outcome, $p_t p_c$, i.e., $R = \left(1 - \left(w_{p_t}(p_t) + w_{p_c}(p_c)\right)\right) l$, where $\left(w_{p_t} + w_{p_c}\right) = 1$ and $\left\{w_{p_t}, w_{p_c}\right\} \in [0,1]$. In this way, risk based on trustworthiness and risk based on context may each be weighted according to significance. In the above example, assigning equal weight to w_{p_t} and w_{p_c}, i.e., $w_{p_t} = 0.5$ and $w_{p_c} = 0.5$, risk would be measured as $R(pq_{sc4t}) = \left(1 - \left(0.5(0.75) + 0.5(0.27)\right)\right)(\$100) = \$49$. This makes sense intuitively, as the amount of financial loss risked is higher than if contextual risk were not taken into account, but lower than the overall risk of interacting in item category 4 due to the trustworthiness of the seller. Based on this design of the risk assessment component, a statement of combined user- and context-specific financial risk can then be provided to the user.

3.4.10 Access Control

Access control applies security policy to the risk assessment input and assesses whether or not *p* should interact with *q*, i.e., whether or not secure interaction with *q* is likely based on the risk to which *p* would be exposed. A basic set of security policies might be: high security, e.g., never risk more than 1% of the current price; medium security, e.g., never risk more than 50% of the current price; and low security, e.g., never risk more than 95% of the current price. Thus, the output of the access control mechanism is a security decision that is provided as advice to *p*, along with a statement regarding the potential risk exposure.

3.4.11 Conclusions

This section described the way in which the SECURE trust, evidence, risk, and access control methods can be incorporated into the design of a reputation management system for Internet auctions. An overview of the SECURE-enhanced reputation management system was given, showing how SECURE can be integrated into a typical Internet auction system. Then, each of the design decisions required for the deployment of SECURE in this application domain were presented, including the identification of request type, relevant information, and context; entity recognition; the design of event structures, event configurations, and evidence types to support the trust model; the design of enhanced *eff* and *eval* functions that produce contextually relevant results when analysing evidence; and the design of risk and access control methods that are suited to the Internet auction domain.

In the following section, we discuss the design of extensions to SECURE to allow for recommendation weighting and collusion detection.

3.5 Extending the SECURE Model with Interaction Dynamics

As described in the previous section, the SECURE TSF can provide optimised reputation management. However, it provides no viable means for recommendation weighting or protection against collusion attacks in the Internet auction application domain, i.e., Types 8 – 10 in Table 3. Thus, the SECURE model is a good tool for assessing user-centric behaviour in terms of trust and risk, given records about extra-auction behaviour as evidence, but the model provides no methods for capturing and analyzing the dynamic behaviour of interactions between pairs of users, i.e., interaction dynamics. In the Internet auction application domain, much information can be gleaned from such interaction dynamics, e.g., recommendation paths linking nodes and auction-duration system observations about user-pair behaviour. This section describes an extension to the SECURE framework supporting interaction management for the purpose of allowing recommendation weighting and collusion detection.

First, we put forward our rationale for a recommendation integrity assessment technique based on trustworthy paths and recommendation policy specification. Then, we present a mechanism to detect collusion based on the SECURE observation method. Finally, we illustrate how the interaction management extension can be integrated into the SECURE framework.

3.5.1 Recommendation Weighting

Recall that recommending evidence for the purposes of trust-based decision-making must be done in a manner that takes into account recommendation integrity, i.e., recommendation reliability, in order to properly adhere to the rules of trust transitivity. This section discusses recommendations and the methods available to determine recommendation integrity, as well as the way in which recommendation assessment is incorporated into the extension to SECURE.

3.5.1.1 Recommendations

It is important to recall that evidence consists of not only a decision-maker's, *p*'s, observations about principal *q*, but also recommendations, i.e., the observations about *q* made by principals other than *p*. In the Internet auction application domain, other principals' recommendations are the primary source of information about a given user, that is, in this domain, a user is typically interacting for the first time with another user and has no observations upon which to rely.

In the current SECURE model, each principal uses an interaction history to capture his observations about another principal. Thus, there is an interaction history for each recommender, *r*, of *q* that captures *r*'s observations about interacting with *q*. These observations are used to form *r*'s trust value for *q*. Because the application domain is centralised, interaction histories captured by principals are public knowledge, i.e., these interaction histories are made available to the entire Internet auction

community as feedback, that is, recommendations about *q*. In order to get a recommended trust value from that feedback, *p* must perform the *eval* function on recommendations to get a trust value for *q* for each recommender of *q*, i.e., $T_{rec_{rq}}$. In performing the *eval* function on evidence about *q* given by *r*, the *eff* function determines which feedback *r* has recorded about *q*, i.e., in *r*'s interaction history for *q*, that is contextually relevant and what recommended evidence supports, contradicts, or is inconclusive with regard to the decision being made about *q* in a specific role in a specific item category over a given time period. Then, *p* combines $T_{rec_{rq}}$ for each recommender of *q* to form T_{rec_q}, which *p* then combines with his own observations, if any, of *q*, i.e., $T_{obs_{pq}}$, to produce an overall trust value for *q*, $T_{ov_{pq}}$.

3.5.1.2 Recommendation Integrity in Current Commercial Reputation Management Systems

Current commercial reputation systems for Internet auctions provide no recommendation integrity assessment. Each unique observation, i.e., only one positive and one negative observation by *p* about *q*, counts in full to a reputation. Each unique recommendation also is counted in full. This results in two issues. First, the reliability of recommendations is not assessed. Second, there is a loss of information about repeat interactions between *q* and his recommenders, i.e., the number of times two users have interacted (and information about the context of those interactions) is not considered when summarising reputation.

3.5.1.3 Recommendation Integrity Using Semantic Distance

One method to provide recommendation integrity is semantic distance, in which *p* analyses each recommendation from recommender *r* about *q* to see how close that recommendation is to *p*'s own observations of *q*. A ratio of semantic distance between the two is produced that *p* may use to weight recommendations from *r* about *q*. However, in the Internet auction domain, it is unlikely that *p* will have observations with which to compare recommendations, which means that most, if not all, recommendations about *q* will be weighted at 100%, providing no information about recommendation integrity. Moreover, even if *p* does have an observation with which to compare *r*'s recommendation, it is not scalable in this domain to compare *p*'s own observation to potentially thousands of recommendations about *q*.

3.5.1.4 Determining Recommendation Integrity Using Subjective Logic

Recall that the discounting operator, $\otimes$, is used in subjective logic to discount a recommended opinion, in belief-disbelief-uncertainty format, according to the recommendation integrity of the

recommender. For example, if *r* recommends the opinion $\omega_x^{rq} = \{b_x^{rq}, d_x^{rq}, u_x^{rq}\}$ about *q*'s functional trustworthiness in performing task *x*, and *p* has an observed opinion about *r* with regard to *r*'s ability to recommend, i.e., task *y*, $\omega_y^{pr} = \{b_y^{pr}, d_y^{pr}, u_y^{pr}\}$, the discounted recommendation is $\omega_x^{prq} = \omega_y^{pr} \otimes \omega_x^{rq}$

For example, if ω_y^{pr} = {0.9, 0.0, 0.1} and ω_x^{rq} = {0.8, 0.0, 0.2}, then, using the subjective logic discounting formula, ω_x^{prq} is the discounted recommendation {0.72, 0.0, 0.28}, in which uncertainty, rather than disbelief, has been increased. This makes sense intuitively because *p* is less certain about *r*'s observations of *q* rather than more disbelieving of *q*'s actions.

This is a sound mathematical method for calculating a trust value that increases uncertainty rather than disbelief about an entity when recommendations are used in addition to or in place of observations. As in the case of semantic distance, though, scalability is an issue. It is not feasible in a large Internet auction to record feedback about a recommender's ability to recommend.

In a system in which a recommender's ability to recommend is not, or can not be, captured, the application of this method depends on a decision-maker's disposition. For example, if a decision-maker *p* believes that all recommenders are honest and good at recommending, *p* could use a recommendation policy would reflect that disposition, e.g., ω_y^{pr} = {1.0, 0.0, 0.0}. If *p* believes that all recommenders are dishonest and unreliable in recommending, a recommendation policy may be ω_y^{pr} = {0.0, 1.0, 0.0}. If *p* believes that all recommenders are honest and good at recommending but is slightly uncertain that a recommender will always behave honestly, the policy may be ω_y^{pr} = {0.9, 0.0, 0.1}.

However, an unfortunate drawback of this proposal is that subjective logic operates on (*b, d, u*) triples, which measure trust in terms of probability of belief, disbelief, and uncertainty. SECURE measures trust with (*s, i, c*) triples, which are not probabilities. We can simply convert a SECURE (*s, i, c*) triple to a (*b, d, u*), but information loss occurs, e.g., (1, 0, 0) and (100, 0, 0) both convert to (1.0, 0.0, 0.0) and we have lost information about evidence in terms of the number of interactions that support, contradict, and are inconclusive about a potential outcome.

3.5.1.5 Recommendation Integrity Using Trustworthy Path Analysis

Another potential solution to assessing recommendation integrity is based on path analysis, a concept used in the field of social network analysis to analyse interaction dynamics between users of a network. We base this solution on the work of Latora and Marchiori (Latora and Marchiori 2001; Latora and Marchiori 2003) who put forward methods for finding path efficiency in, e.g., neural, social, communication, and transport networks.

Latora and Marchiori demonstrate that a network can be characterised by the introduction of an efficiency variable which measures how efficiently nodes exchange information. An unweighted graph requires only an adjacency matrix to be described, i.e., a matrix $\{a_{ij}\}$ containing information about whether or not a link exists between any two nodes i and j and defined as a set of numbers a_{ij} = 1 when there is a vertex joining i and j, and $a_{ij} = 0$ when there is no vertex between i and j. The Latora-Marchiori model considers a network as a weighted graph needing two matrices to be described. First, an adjacency matrix is required, as in the unweighted graph. Additionally, a matrix is needed that associates weights to each link between i and j, $\{l_{ij}\}$, named the matrix of physical distances because the number l_{ij} can be thought of as the spatial distance based on information annotating the link between i and j. For example, l_{ij} in a social network can be set equal to the inverse number of edges between i and j, e.g., the inverse of the number of times i and j have interacted in a specified way. Nodes that are closer will thus have a smaller physical distance between them, i.e., smaller weight in the matrix of physical distance.

The matrix of shortest path lengths $\{d_{ij}\}$ is then calculated by using information contained in both the adjacency and physical distance matrices. The shortest path, E, is the one that covers the least physical distance between i and j, rather than the path with the fewest vertices. That is, the path with the smallest sum of the physical distances throughout all the possible paths in the graph from i to j.

An Internet auction community is a social network that can be analysed according to the above concepts, in particular, to find the most efficient, i.e., most trustworthy, paths between a decision-maker, p, and a recommender, r, of user q, thus gleaning trust information from interactions of users in the path between a decision-maker and recommender. We propose the trustworthy path method discussed as follows.

If p has interacted directly with q, we assume that his own observations will be weighted more highly than recommendations, e.g., 100%. All recommendations can be discounted according to their trustworthiness, i.e., recommendations made by principals that are 'closer' to p will be weighted more strongly than those recommendations at the end of paths that are less trustworthy.

In order to weight a recommendation about q made by recommender r_2, $T_{obs_{r_2q}}$, decision-maker p considers the paths linking him to r_2. Along this path, p has interacted directly with r_1 who has interacted directly with r_2 who has interacted directly with q, and p has interacted directly with r_3 who has interacted directly with r_2. The two paths, annotated with SECURE trust values resulting from interactions between each node pair, are illustrated in Figure 35.

$$\text{Path A}: \; p \xrightarrow{(1,0,0)} r_1 \xrightarrow{(3,0,0)} r_2 \xrightarrow{(1,0,0)} q$$

$$\text{Path B}: \; p \xrightarrow{(1,0,0)} r_3 \xrightarrow{(1,0,0)} r_2 \xrightarrow{(1,0,0)} q$$

Figure 34: Recommendation paths annotated with trust values

As per the Latora-Marchiori model, the path trustworthiness can be found for each path by combining the inverse of the number of positive, or trustworthy, interactions between each node pair along the path. In Path A, the distance between p and recommender r_2 is the sum of 1/1 and 1/3, or 1.33. The distance between p and r_2 in Path B is 2. Therefore, Path A is the most trustworthy path of the two and is the path that should provide trust information with which to weight r_2's recommendations about q.

Unfortunately, two issues arise when considering path analysis as a recommendation weighting scheme. First, the weight of each leg of a trustworthy path is based on trust values pertaining to functional trust rather than referral trust. The same problem of lack of evidence about the ability of a recommender to recommend due to scalability arises here as it did in the other recommendation assessment methods. Second, the distance measure that is produced by computing path trustworthiness is counterintuitive when used as a weight. That is, in a long path the distance measure is likely to be higher than 1, which would amplify r_2's recommended trust value rather than discounting it.

Of the three methods discussed, i.e., semantic distance, subjective logic discounting, and trustworthy paths, the trustworthy path analysis mechanism provides a basis to form a recommendation weighting solution in systems, such as reputation management in Internet auctions, in which it is infeasible to rate the ability of recommenders to recommend due to lack of observations or lack of a mechanism that captures referral trust. Although the distance measure is not useful for weighting recommendations, the finding of trustworthy paths can be combined with a recommendation integrity policy specification method, thus leading to the design of a solution that uses trust and interaction dynamics to provide recommendation weighting. The proposal of the design of such a recommendation weighting method is described in the following section.

3.5.1.6 Design of the Recommendation Weighting Method

Having examined the benefits and difficulties of using various recommendation discounting methods, we propose to use a combination of path analysis and recommendation integrity policy specification based on the disposition of the decision-maker.

First, we use trustworthy path analysis to find most trustworthy path between a decision-maker, p, and each recommender, r, of q. We consider an Internet auction community to be a graph, with each node representing a user, each vertex representing a link between users who have interacted, and each vertex annotated with trust values, extracted from interaction histories, representing the number of times users have interacted in a trustworthy manner. Although it is inappropriate to use functional trust in place of referral trust when discounting recommendations, we propose the assessment of the general trustworthiness of a path between a decision-maker and a recommender based on functional trust as part of a solution in the absence of referral trust.

We then use Dijkstra's shortest path algorithm (Dijkstra 1959) to find the most trustworthy path (shortest weighted path) between decision-maker, *p*, and recommender, *r*, on the weighted graph of trustworthy paths described above. This algorithm then outputs trusted path length, i.e., the number of hops between *p* and *r* in the most trustworthy path joining the two..

Having found a trustworthy path between *p* and *r*, *r*'s observations about *q* must then be weighted. In the absence of referral trust, to weight each recommendation, a recommendation integrity policy, $\pi \in [0,1]$, is used. π expresses *p*'s subjective view of recommenders in the Internet auction domain as a probability of the level of honesty and reliability of each recommender along the trustworthy path found between *p* and *r*. For example, if *p* believes that all recommenders are completely honest and reliable, π may be set to 1, in which case recommendations are treated as observations. On the other hand, if *p* believes that all recommenders are completely dishonest and unreliable, π may be set to 0, in which case all recommendations are ignored and *p* relies solely on observations. Alternatively, π may be expressed as a probability that lies between these two extremes.

We extend the *eff* function to incorporate recommendation weighting.

$$\text{eff}_{x_{rc}}(w_{rc}) = \begin{cases} \pi^{l}(1,0,0) \text{ if } w_{rc} \subseteq x_{rc} \\ \pi^{l}(0,0,1) \text{ if } x_{rc} \# w_{rc} \\ \pi^{l}(0,1,0) \text{ if neither } w_{rc} \subseteq x_{rc} \text{ nor } x_{rc} \# w_{rc} \end{cases}$$

Where π is the recommendation weighting policy selected, e.g., .99, and l is the number of hops between *p* and *r*, and where evidence about *q* is analysed to determine whether it is supporting, contradicting, or inconclusive about a proposition as well as whether or not it is contextually relevant to a proposition. If a piece of evidence is *p*'s own observation about *q*, then $l = 0$ and $\pi = 1$, which means that the evidence is not weighted, as is intuitively true. If $l = 1$, i.e., *p* is 1 hop from *r*, then $\pi^l = .99$, and so on. When performed as part of the *eval* function, the results of passing into the formula H_{rq} results in a (*s, i, c*)-triple that is the weighted recommended trust value $T_{rec_{rq}}$ which *p* will combine with $T_{obs_{pq}}$ to form an overall trust value, $T_{ov_{pq}}$, which is a measure of the trust *p* has in *q*. This process must be performed for every recommender of *q* in the relevant role and item category.

3.5.2 Collusion Detection

Recent research (Kauffman and Wood 2003; Shah, Joshi et al. 2003; Rubin, Christodorescu et al. 2005) considers the issue of anomalous behaviour in Internet auctions. Specifically, competitive shilling, collusion between Internet auction users to drive up the price in an auction, may be used by auction sellers to ensure that a legitimate bidder pays a higher price for the item than in the case where no shilling occurs. Trying to detect or predict such opportunistic behaviour is beyond the scope of most Internet auction research, mainly because, according to the National Consumer League, shilling is the hardest type to detect of the various types of fraud to occur in this domain (Fraud.org 2006).

Shilling occurs when a seller bids in his own auction via aliases or friends. This form of collusion becomes even more difficult to detect in Internet auctions, where alias identities are easily obtained and any user with multiple accounts (and IP addresses) can shill without assistance of friends. Internet auction providers typically employ undisclosed proprietary methods for shill prevention, however, shilling may still occur to some extent.

This section addresses the need for a collusion detection method to be designed to identify anomalous auction-duration behaviour and to be incorporated in decision-making for reputation management. We first discuss the results of the very limited current research in the area of Internet auction shilling. We then propose the design of a collusion detection method, which is modelled using SECURE event structures to capture auction-duration events.

3.5.2.1 Recent Research into Collusion Detection in Internet Auctions

In Shah et al. (Shah, Joshi et al. 2003), it was discovered that attributes of bidding behaviour, mined from eBay's public auction records, could be codified such that different types of bidders could be classified according to bidding strategy. This result is particularly relevant to the detection of fraud such as shilling because the following characteristics of a shill bidder can be identified from auction system data: first, there is a strong association between a seller and a bidder, or a ring of bidders, i.e., the shill(s) appears very frequently in auctions hosted by the seller; next, the shill wins auctions infrequently, if at all; third, the bids placed by a shill are significantly higher than the current asking price; finally, a shill will eschew sniping and late bidding to permit legitimate buyers enough time to respond to his bid increment, as well as to avoid winning the auction.

Research by Kauffman and Wood (Kauffman and Wood 2003) also addresses the characteristics of a shill bidder. They show how to detect such opportunistic behaviour by first examining what a market would look like if shilling behaviour existed and then to test for that behaviour. The identified behaviour is similar to that pinpointed by Shah et al., i.e., the characteristics of 'questionable bids' are: first, a shill bidder bids in a colluding seller's auction regardless of other auctions of similar items; next, a shill bidder concentrates on fewer sellers than other bidders; third, shill bids are usually placed early in the auction and are incremented in large increments, i.e., an average of 62% per increment rather than the non-shill average of 38%; finally, and most importantly, shill bidders try *not* to win the auction in which they are bidding, i.e., average win rates of shills is 23%, versus non-shills, 35%. These results, based on the analysis of over 10,000 eBay coin auctions, allow the prediction of the presence of shilling in an auction based on the assessment of past behaviour of auction participants.

Finally, Rubin et al. (Rubin, Christodorescu et al. 2005) designed and developed a behaviour-based reputation system to help buyers identify sellers whose auctions seem price-inflated. The design is based on models that characterize sellers according to statistical metrics related to price inflation and with anomaly detection techniques to identify suspicious sellers. The characteristics of a suspicious seller in this model is one who lists many auctions which have many bids, do not start auctions with a

relatively low starting bid, and has a group of bidders who repeatedly participate in his auctions and lose. The reputation system outputs to the user a set of values representing the confidence with which the system can say that the auctions of a particular seller are price-inflated. This system was evaluated on over 600 high-volume sellers who listed over 37,500 auctions on eBay. The system automatically detected a small set of sellers whose auctions contained potential shill bidders. We note that this system does not output the cause of inflation, whether legitimate or fraudulent.

In each of these cases, similar types of behaviour attributes are used to characterise questionable bidders, leading us to propose the design of a collusion detection method based on amalgamation of the behaviour attributes identified in this work, as discussed in the following section.

3.5.2.2 Design of the Collusion Detection Method

Based on the results of the research into the area of anomalous behaviour in Internet auctions, we put forward a simple design for the detection of colluding sellers and shills that focuses on the key characteristics of a questionable bid: that shills experience less-than-average win rates for the auctions in which they participate, shills tend to make large bid increments, shills tend to bid early and not late, and that a shill typically interacts with fewer sellers than legitimate bidders. The detection process is done on the buyer side, i.e., when a bid request is put to SECURE for decision-making.

First, in line with the SECURE event-based trust model described earlier, we identify the possible auction-duration events that will allow the profiling of a user-pair, i.e., seller-bidder, behaviour. We base the events for this design on an amalgamation of the attributes of anomalous bidding behaviour put forward by Shah et al., Kauffman and Wood, and Rubin et al., i.e., bidder-seller relationship, bid amount, bid timing, and loss rate. An event structure that captures these attributes as observable events is illustrated in Figure 35.

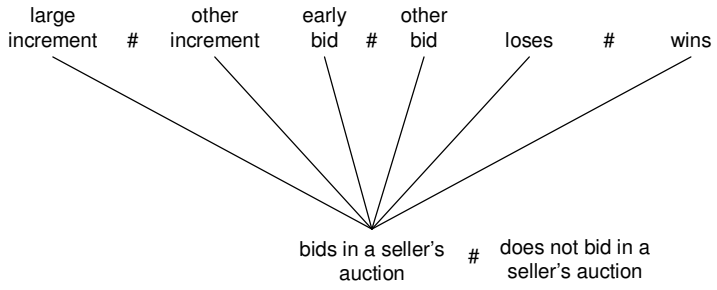

Figure 35: ES_{BS} modelling a bidder-seller relationship

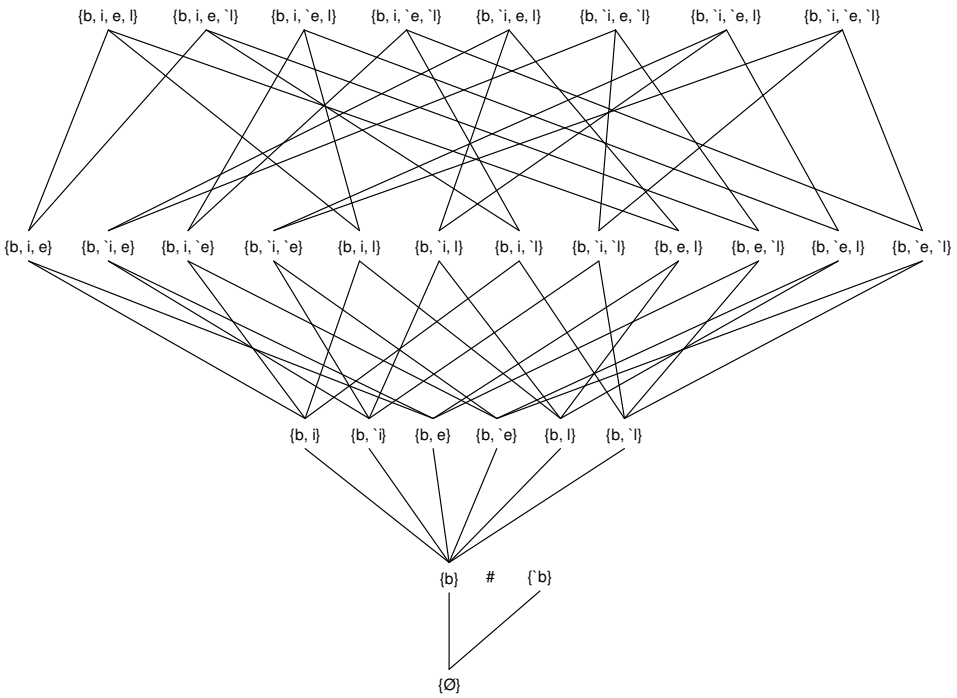

Figure 36: $C_{ES_{BS}}$ modelling auction-duration event configurations

The event configurations possible for ES_{BS} , i.e., $C_{ES_{BS}}$, are illustrated in Figure 36 wherein event names are abbreviated. As illustrated, many event configurations are observable by the auction system about a bidder-seller relationship. At the bottom level of $C_{ES_{BS}}$, no observations about a bidder-seller relationship are observed. At the second level of $C_{ES_{BS}}$ are the outcomes {b} and {`b}, representing observations of whether or not a bidder bids in a seller's auction. At the next level, when a bidder bids in a seller's auction, it is possible to observe whether the bid was a large or small increment, whether the bid was made early or late in the auction, and whether the bidder won the auction or not. The fourth level of $C_{ES_{BS}}$ shows configurations representing outcomes in which combinations of three behavioural events are observed, i.e., that a bidder bid in a seller's auction and whether or not that bid was a large increment; or that a bidder bid in a seller's auction and whether or not that bid was made early in the auction. At the top-most level of $C_{ES_{BS}}$, configurations express outcomes in which all possible behaviour events are observed.

In order to determine which event configurations are most relevant to the decision-making process, Table 4 lists the attributes necessary for detecting colluding behaviour, gives the event configurations needed to describe each attribute, and classifies both normal and anomalous behaviour in bidder-seller relationships based on the results of the research described in section 3.6.2.1. From this table, we find

that the following event configurations can be used to indicate the likelihood of normal or colluding seller-bidder behaviour: {b}, {b, i}, {b, e}, and {b, l}, giving information to assess bidder-seller interaction dynamics in the form of relationship attributes: *P*, the likelihood of a bidder-seller relationship being strong; *B*, the likelihood of a bidder's increments being large; *T*, the likelihood of a bidder's bids being made early in an auction; and *Λ*, the likelihood of a bidder losing in a seller's auction. Therefore, even though it is possible to observe more complex configurations, our detection method is satisfied by this set of configurations. Moreover, as illustrated in $C_{ES_{BS}}$, the observation of outcomes at levels 4 and 5 captures combinations of behaviour events that detract from the flexibility needed to assess relationship attributes individually.

Table 4: Event configurations necessary for detecting colluding behaviour

Attribute	Event configuration per attribute	Normal behaviour	Anomalous behaviour
Bidder-Seller Relationship, P	Bidder appears (bids) in seller's auction, {b} Bidder does not appear in seller's auction, {`b}	P = observations of {b} = low	P = observations of {b} = high
Bid Amount, B	Bidder bids large increment in seller's auction, {b, i} Bidder bids 'normal' increment in seller's auction, {b, `i}	B = observations of {b, i} = low	B = observations of {b, i} = high
Bid Timing, T	Bidder bids before a specified point in time in a seller's auction, {b, e} Bidder bids after a specified point in time in a seller's auction, {b, `e}	T = observations of {b, e} = low	T = observations of {b, e} = high
Loss Rate, Λ	Bidder bids in seller's auction and loses, {b, l} Bidder bids in seller's auction and wins, {b, `l}	Λ = observations of {b, l} = low	Λ = observations of {b, l} = high

Because our collusion detection method is based on the SECURE trust model, a *(s, i, c)*-triple captures evidence which supports, contradicts, or is inconclusive about the configuration describing each bidder-seller relationship attribute, *P*, *B*, *T*, and *Λ*. This evidence results from auction system observations about auction-duration events.

Finally, *P*, *B*, *T*, and *Λ* are each given a weight, with weight total to be 100%, so that each parameter can contribute to the total expected probability of collusion in a flexible way. For example, each attribute may be equally important when determining overall likelihood of collusion between a seller, *q*, and a bidder, *b*. In this case, the evidence about collusion, Φ_{qb}, is expressed as

$\Phi_{qb} = .25\mathrm{P} + .25\mathrm{B} + .25\mathrm{T} + .25\Lambda$. Clearly, these weights can be adjusted in favour of some attributes over others. Moreover, as other anomalous behavioural attributes arise in research in this domain, they may easily be added to our model.

The final combined (*s, i, c)*-triple of evidence about Φ_{qb} is then used to derive an expected probability of the occurrence of collusion in a given auction. This probability is derived by $R(\Phi_{qb}) = \frac{s}{s+i+c}$.

3.5.3 Extending the SECURE Framework for Interaction Management

Two components are required to extend SECURE for interaction management to provide recommendation weighting and collusion detection methods to the trust- and risk-based decision-making framework. These components are the Interaction Manager and the Interaction Calculator. The Interaction Manager assesses trustworthy paths and evaluates auction-duration evidence about collusion. A recommendation weighting is then determined according to path length and recommendation weighting policy, and this weighting is passed to the Trust Lifecycle Manager. The results of evaluating collusion evidence are passed to the Interaction Calculator, which applies a weighting policy to P, B, T, and Λ, and passes the result, Φ, to the Trust Calculator.

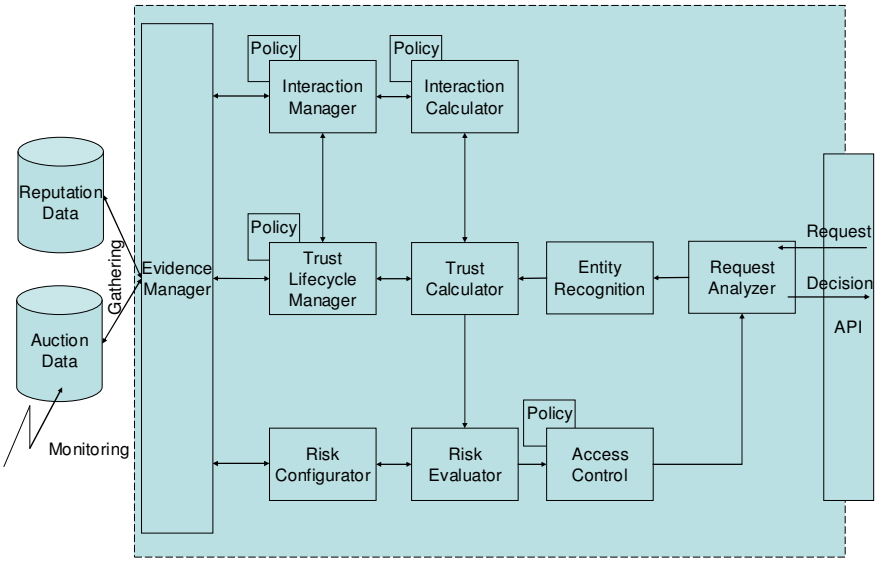

Figure 37: The SECURE framework, extended for interaction management

The additional framework components are illustrated in Figure 37, which shows the integration of the Interaction Manager and the Interaction Calculator with the other SECURE components, giving the extended SECURE framework. We use the extended framework as the basis for our Reputation Management System (RMS), in which decision-making occurs as described in the following section.

3.5.4 Decision-Making in the Reputation Management System (RMS)

Decision-making in the Reputation Management System (RMS) is enacted on behalf of an Internet auction user deciding whether or not to bid on an item or whether or not to sell an item to a given high bidder. The decision-making process is different in each case, and each process is described as follows.

3.5.4.1 Decision to Bid

The Application Program Interface (API) links an Internet auction user, *p*, in the role of buyer to the SECURE decision-making process at the point when the user is determining whether or not to place a bid on an item that auction user, *q*, is selling. The request passes from the API to the SECURE Request Analyzer (RA), and contains all of the information SECURE needs to initialise decision-making, as illustrated in Figure 38.

Bid request		
Request query	Should I place a bid?	
Identification information	Authenticated username of principal *q*	
Context information	Role context, i.e., *q*'s role	Seller
	Environmental context (category)	Item category number
	Environmental context (price)	Current price of item
	Temporal context	Current date/time
	Interaction dynamic context	Current bidders $\{b_1, b_2, \ldots b_n\}$

Figure 38: Bid request

The bid request contains the query 'should I place a bid', which means 'what is the likelihood based on evidence about *q* that *q*, interacting in the role of seller of an item in the specified item category of the specified price at the specified date and time will behave in a trustworthy manner and that *q* is not colluding with any other current bidder, *b*?' In our application, trustworthy means that a seller ships an item as described, i.e., {*i, s, d*}.

The RA passes the request to the Entity Recognition (ER) component, which allocates 100% reliability to *q*'s username and passes the request to the Trust Calculator (TC).

The TC queries the Trust Lifecycle Manager (TLM) for evidence about *p*'s observations about *q*, i.e., $T_{obs_{pq}}$, and about other user's recommendations about *q*, i.e., $T_{rec_{rq}}$, where *r* is a recommender of *q*. The TC also queries the Interaction Calculator (IC) for Φ_{qb} about each seller-bidder pair in the current auction.

The TLM requests evidence about *q* from the Evidence Manager (EM), which retrieves evidence about *q* from the auction system reputation database. Evidence is returned to the TLM in the form of interaction histories H_{pq}, i.e., event configurations comprising *p*'s observations of *q*, and H_{rq}, i.e., event configurations comprising recommender *r*'s observations of *q*. (H_{rq} is returned for each

recommender of q.) The TLM also queries the Interaction Manager (IM) for π^l_{pr} for each recommender of q for which it has received an interaction history, and is returned π^l_{pr} for each recommender.

The TLM evaluates H_{pq} and H_{rq} using the *eff* and *eval* functions. The *eff* function analyses the effect of each piece of evidence, i.e., event configuration, in an interaction history to determine whether the evidence is supporting, contradicting, or inconclusive about the request proposition {*i, s, d*}, as well as whether or not the piece of evidence is contextually relevant to the request according to role and item category. Additionally, for each interaction history, H_{rq}, the *eff* function applies recommendation weighting using π^l_{pr} to discount recommendations. The *eval* function sums the effects of all evidence for H_{pq} and H_{rq}, independently of one another, and applies time fading. This process results in trust values, i.e., (*s, i, c*)-triples $T_{obs_{pq}}$ and $T_{rec_{rq}}$, which are returned to the TC.

In addition to dealing with the TLM's request for π^l_{pr}, the IM receives a request from the IC to evaluate the likelihood of collusion between q and each current bidder, b. The IM requests evidence in the form of auction system observations about interactions between q and b, i.e., the interaction history H_{qb}, and applies the *eff* and *eval* functions (no contextual relevance, recommendation weighting, or time fading occur in this application of these functions, as we are interested in all of q's potential colluding interactions with b and as they are system observations rather than recommendations). The resulting collusion factors, P, B, T, and Λ are passed to the IC.

The IC weights P, B, T, and Λ according to its weighting policy and combines them to produce Φ_{qb}, which it passes to the TC.

The TC combines $T_{obs_{pq}}$ and $T_{rec_{rq}}$ to produce $T_{ov_{pq}}$, a trust value that represents p's overall trust in q. The TC passes $T_{ov_{pq}}$ and Φ_{qb} to the Risk Evaluator (RE).

The RE queries the Risk Configurator (RC) for the contextual risk of interacting in the item category specified in the request. The RC queries the EM for updated information regarding the category risk and returns category risk to the RE. The RE calculates the risk of interacting with q in this context as well as the likelihood of collusion in the current auction. It passes the results to the Access Control (AC) component.

The AC enacts policy to determine whether or not p should enter into an interaction with q and the current bidset based on the risk levels of q acting in an undesirable manner and of collusion between q and any bidder, b. The decision, as well as a statement about the risk of exposure to untrustworthy behaviour and colluding behaviour, is passed to the RA, which outputs the decision to the user via the API.

3.5.4.2 Decision to Sell

The Application Program Interface (API) links an Internet auction user, *p*, in the role of seller to the SECURE decision-making process at the point when *p* is determining whether or not to accept a high bid made by *q* at the end of the auction. This request is illustrated in Figure 39.

Sale request		
Request query	Should I accept a bid?	
Identification information	Authenticated username of principal *q*	
Context information	Role context, i.e., *q*'s role	Buyer
	Environmental context (category)	Item category number
	Environmental context (price)	Current price of item
	Temporal context	Current date/time

Figure 39: Sale request

The sale request contains the query 'should I accept a bid', which means 'what is the likelihood based on evidence about *q* that *q*, interacting in the role of buyer of an item in the specified item category of the specified price at the specified date and time will behave in a trustworthy manner, i.e., {*i, p*} meaning pays for the item?'

The RA passes the request to the Entity Recognition (ER) component, which allocates 100% reliability to *q*'s username and passes the request to the Trust Calculator (TC).

The TC queries the Trust Lifecycle Manager (TLM) for evidence about *p*'s observations about *q*, i.e., $T_{obs_{pq}}$, and about other user's recommendations about *q*, i.e., $T_{rec_{rq}}$, where *r* is a recommender of *q*. The Interaction Calculator is not invoked for this type of decision.

The TLM requests evidence about *q* from the Evidence Manager (EM), which retrieves evidence about *q* from the auction system reputation database. Evidence is returned to the TLM in the form of interaction histories H_{pq}, i.e., event configurations comprising *p*'s observations of *q*, and H_{rq}, i.e., event configurations comprising recommender *r*'s observations of *q*. (H_{rq} is returned for each recommender of *q*.) The TLM also queries the Interaction Manager (IM) for π^{l}_{pr} for each recommender of *q* for which it has received an interaction history, and is returned π^{l}_{pr} for each recommender.

The TLM evaluates H_{pq} and H_{rq} using the *eff* and *eval* functions. The *eff* function assess each piece of evidence, i.e., event configuration, in an interaction history to determine whether the evidence is supporting, contradicting, or inconclusive about the request proposition {*i, p*}, as well as whether or not the piece of evidence is contextually relevant to the request according to role and item category. Additionally, for each interaction history, H_{rq}, the *eff* function applies recommendation weighting using π^{l}_{pr} to discount recommendations. The *eval* function sums the effects of all evidence for H_{pq}

and H_{rq}, independently of one another, and applies time fading. This process results in trust values, i.e., (*s, i, c*)-triples $T_{obs_{pq}}$ and $T_{rec_{rq}}$, which are returned to the TC.

The TC combines $T_{obs_{pq}}$ and $T_{rec_{rq}}$ to produce $T_{ov_{pq}}$, a trust value that represents *p*'s overall trust in *q*. The TC passes $T_{ov_{pq}}$ to the Risk Evaluator (RE).

The RE queries the Risk Configurator (RC) for the contextual risk of interacting in the item category specified in the request. The RC queries the EM for updated information regarding the category risk and returns category risk to the RE. The RE calculates the risk of interacting with *q* in this context. It passes the result to the Access Control (AC) component.

The AC enacts policy to determine whether or not *p* should enter into an interaction with *q* based on the risk levels of *q* acting in an undesirable buyer manner. The decision, as well as a statement about the risk of exposure to untrustworthy behaviour, is passed to the RA, which outputs the decision to the user via the API.

3.6 Design Conclusions

In this section, we proposed the design of a Reputation Management System (RMS) for Internet auctions based on extending the SECURE trust- and risk-based decision-making framework. We tailored, and extended, the SECURE trust, evidence, and risk mechanisms to this application domain and our contributions in this regard are described as follows.

First, we designed the RMS in such a way as to increase accuracy in decision-making for users of Internet auctions while maintaining usability. Initially, we classified application-specific behaviour as a taxonomy of normal and anomalous user behaviour types. Then, we designed SECURE event structures and configurations to model the event types needed to predict the likelihood of role-specific user behaviour in future interactions. We adapted the event structures and the associated evaluation methods of the trust model, i.e., interaction histories and the *eff* and *eval* functions, to allow for the assessment of gathered extra-auction evidence in the form of observations and recommendations at a level of granularity that permits both usability and increased precision in decision-making. The assessment of evidentiary relevance based on contextual aspects further increases accuracy. Furthermore, we designed components with which to analyse interaction dynamics, i.e., recommendation weighting based on path analysis and collusion detection based on auction-duration events, to enhance the precision of the decision-making process. Automating the analysis of evidence using the RMS allows for increased accuracy as well as reducing the complexity in decision making as more evidence may be assessed than a human user is capable of manually processing.

Second, the RMS is designed to incorporate contextual elements in the decision-making process. Although SECURE was designed to allow for the assessment of context, it did not include the

specification of a mechanism to perform this task. Our application of the SECURE approach proposes the design of a mechanism with which to assess contextual relevance, with which we extend the SECURE *eff* function. Therefore the evidence being assessed for an interaction decision is related to the context of the decision rather than obliging a user to make a decision in the absence of context, as is the case in existing reputation management systems. We analyse context based on user role, environmental factors of item category and price, timeliness of evidence, and the context of interaction dynamics, i.e., the relationships between members of a group of users participating in a given auction. SECURE was designed to allow for the assessment of temporal context, and our application proposes a new time fading mechanism that we use to extend the *eval* function so as to allow evidence to be weighted based on timeliness. Thus the output security decision is less ambiguous, and again more accurate, than a decision made in which context is disregarded. Additionally, evidence is often too abundant for a human user to filter out evidence according to contextual relevance, and hence this aspect of our design also permits increased usability while reducing complexity.

Third, the RMS design provides methods, based on our extension to the original SECURE framework, for the analysis of evidence about interaction dynamics, i.e., information about relationships between pairs of users. Internet auction system observations about the dynamics of user interactions are evaluated, e.g., whether or not a user provides useful and accurate recommendations about another user and whether or not a user employs a specific interaction strategy, such as collusion with malicious intent, when interacting with another user. As in the case of determining which information is more contextually relevant to a decision at hand, it is impossible for a user to manually assess an overwhelming amount of system observations about complex interaction dynamics to determine which information is relevant when considering whether or not to interact with a given set of users. Therefore, the RMS further adds usability while reducing complexity.

Fourth, the RMS uses risk analysis methods based on the SECURE risk model to assess user trustworthiness, thus making explicit to a user the expected cost of an interaction resulting in favourable or undesirable result. Moreover, we extend the SECURE approach to include methods with which to analyse both contextual risk and risk based on interaction dynamics. We designed the risk methods in such a way as to expose risk in financial terms, according to traditional security risk assessment techniques that users comprehend, as an alternative to the more subjective utility theoretic risk methods incorporated in the original SECURE design.

Finally, the result of the RMS decision-making process is advice that is provided to a user considering an Internet auction interaction. This decision guides a user to make a correct decision about entering into an interaction based on the RMS's automated evaluation of contextually relevant evidence in terms of trust, risk, and interaction dynamics. The security decision is therefore more useful to and usable by a user than the current reputation summary information currently provided by commercial reputation management systems.

3.7 Chapter Summary

This chapter addressed issues outstanding in the reputation management systems currently used to support decision-making in Internet auction applications and presented a design for a reputation management system based on extending the SECURE decision-making framework.

First, the SECURE approach was described, including its trust, collaboration, and risk models, framework components, decision-making processes, and implementation. This description also covered a discussion of the general threats that SECURE was designed to protect against.

Next, the deployment of SECURE in the spam filtering domain was discussed, which encompassed the description of how the SECURE trust, collaboration, and risk models were applied in this domain, as well as a brief examination of the implementation and evaluation of the SECURE spam filtering application.

Third, a general description of the reputation management in virtual marketplaces application domain was put forward, including our development of an application-specific taxonomy of behaviour that classifies both normal and anomalous types of behaviour in this domain.

Then, we applied the SECURE approach to a new domain, i.e., that of reputation management in Internet auctions. The design of a Reputation Management System (RMS) based on the SECURE trust- and risk-based decision-making framework was proposed. An overview of the design was given, and our rationale for our design decisions was presented with regard to requests, entity recognition, trust and evidence processes, risk assessment, and access control. Additionally, our rationale was described for extending the SECURE framework to include interaction management such that decision-making might be enhanced through the analysis of trustworthy recommendation paths between users and observations about potential colluding behaviour. We illustrated how these two new interaction management components may be integrated into the SECURE framework and detailed the enhanced decision-making process. The proposed design of the RMS addressed existing issues with regard to reputation management in Internet auctions in terms of reducing complexity, increasing accuracy, and maintaining usability.

Chapter 4: Evaluation

It takes 20 years to build a reputation and five minutes to ruin it.
If you think about that, you'll do things differently.
~ Warren Buffett

This chapter discusses the evaluation of the RMS reputation management system, whose design was described in Chapter 3. First, we present an evaluation plan that identifies success criteria for a reputation management system and discusses how they may be evaluated. Then, we describe the evaluation of the proposed RMS design in three parts: simulations of user behaviour in Internet auctions such that the mechanisms of the RMS for analysing trust, context, timeliness, and recommendation integrity of reputation evidence used to measure trust can be assessed; a case study using data that is publicly available on a well-known Internet auction website to evaluate the potential benefits of the anomaly detection capability of our collusion detection mechanism; and a qualitative analysis of the RMS as a trust-based decision-making system in the context of the key trust characteristics identified in Chapter 2.

4.1 Evaluation Plan

The evaluation plan defines success criteria, based on the goals identified in Chapter 1, for building a reputation management system for Internet auction users. The overall success criteria are three-fold, i.e., to reduce complexity, to increase accuracy, and to maintain usability. Each of these criteria is more finely specified according to our detailed aims and objectives in the following subsections. In addition, we describe the methods used to conduct the evaluation.

4.1.1 Domain-Specific Evidence Collection and Analysis

Evaluation Criterion 1: The trust value calculation mechanism of the RMS should be evaluated to assess its accuracy in producing a trust value that can be used as a basis to calculate the likelihood of domain-specific entity behaviour based on observations of that behaviour.

The RMS should maintain the usability of the evidence collection and analysis mechanisms while increasing accuracy by collecting evidence that is of a suitable type and sufficient granularity to identify domain-specific behaviour types and thus to limit risk of exposure to user interaction with

untrustworthy entities. Not only should the observation of certain domain-specific behaviour assist in increasing accuracy while maintaining ease-of-use of the feedback mechanism, but also it should allow the RMS to calculate a trust value with which the risk mechanism could assess the likelihood of behaviour in future interactions.

The domain-specific behaviour we are interested in includes behaviour types in virtual marketplaces with reputation management, i.e., those behaviour types described in Table 3 in Chapter 3, specifically: normal behaviour over time, new or unknown behaviour, bad behaviour over time, and behaviour that oscillates between normal and bad behaviour over time.

4.1.2 Context-Specific evidence Collection and Analysis

Evaluation Criterion 2: The contextual relevance assessment mechanism of the RMS should be evaluated to assess the results of using a contextually relevant trust value to calculate the likelihood of domain-specific entity behaviour in a given environmental context based on observations of behaviour in the context.

The RMS should provide for evidentiary analysis with regard to contextual relevance assessment in terms of role and environment. By automating the analysis of the context information that is appended to evidence, the RMS can not only increase the accuracy of its decision-making ability, but also decrease complexity and increase usability for users who would no longer need to manually filter through an abundant body of evidence to determine which recommendations were more relevant to a decision at hand.

4.1.3 Time-Specific Evidence Collection and Analysis

Evaluation Criterion 3: The time-fading mechanism of the RMS should be evaluated to assess the results of using a time-faded trust value to calculate the likelihood of domain-specific entity behaviour in a given time period based on observations of behaviour in the time period.

The RMS should provide for increased accuracy of evidentiary analysis with regard to temporal assessment. By incorporating time-fading, the RMS should, again, be able to increase the precision of the automatic decision-making process in a manner that diminishes complexity for the user. Trust values produced by the system may be weighted toward current behaviour, and the time-faded trust value should recognise and reflect recent trends in entity performance while fading evidence about past behaviour.

4.1.4 Limiting Exposure to Risk of Unreliable Evidence

Evaluation Criterion 4: The trust value calculated by the RMS should be evaluated to assess the results of the application of a recommendation weighting policy that may be specified at various levels of risk aversion.

The RMS should provide the ability to limit exposure to risk from unreliable recommendations. Recall that in reputation management for Internet auction users, it is difficult to assess the ability of a recommender to recommend another user. This occurs primarily because a typical auction user will have no observations of any other given user, i.e., a user will not have interacted with the majority of other users. This results in a lack of observations with which to compare recommendation ability. For example, if Alice has not personally observed Bob's ability as a seller, how can Alice objectively judge Carl's recommendation of Bob's selling abilities? By the same token, in this domain, Alice typically will have no observations about Bob, and must rely on recommendations as the sole form of evidence about Bob. Recall also that in the Internet auction domain, recommendations tend to be Pollyanna-type assessments, leading to a reputation that may be inflated by false positive evidence. By weighting recommendations according to recommendation integrity policy specification, the RMS should produce a trust value that exposes a user to less risk in the case where recommendations, e.g., false positives, may not be wholly reliable. This objective also contributes to the reduction of complexity and maintenance of usability for users interacting in the Internet auction domain.

4.1.5 Colluding Behaviour Evidence Collection and Analysis

Evaluation Criterion 5: The collusion detection mechanism of the RMS should be evaluated to assess its ability to capture and assess the interaction dynamics that make up the profile of colluding behaviour.

The RMS should provide the ability to detect anomalous behaviour with respect to interaction dynamics between groups of users, specifically, collusion for the purpose of artificially increasing the price of an item being auctioned, and thus limit exposure to risk from colluding behaviour. As in the case of determining which evidence is more behaviour-specific, contextually relevant, or reliable, it would be nearly impossible for a user to manually assess an overwhelming number of system observations about complex interaction dynamics to determine which users may be colluding. By including a mechanism to automatically assess interaction dynamics, the RMS is further diminishing complexity and providing ease-of-use to users.

4.1.6 Making Risk Explicit

Evaluation Criterion 6: The RMS should be evaluated in terms of its ability to make risk explicit to the user.

The RMS should make risk explicit to the decision-maker, both in terms of the risk of financial loss due to untrustworthy behaviour based on the calculation of a trust value that is behaviour-specific, contextually relevant, and based on reliable evidence and in terms of the risk of paying an artificially inflated price due to colluding behaviour between a seller and bidder. This factor also concerns the reduction of complexity and provision of usability, i.e., analysing risk in terms of trust-based evidence and presenting that analysis to a user.

4.1.7 Advice Provision

Evaluation Criterion 7: The RMS should be evaluated in terms of its ability to provide advice to the user.

The RMS should provide advice based on trust, risk, context, and interaction dynamics to the decision-maker, guiding the user on whether or not to proceed with an interaction. While this criterion embodies all three overall goals of our work, it mainly signifies that a user of the RMS should be guided by advice to make a more informed, and therefore more accurate, decision about whether or not to proceed with an interaction.

4.1.8 Trust-Based Decision-Making

Evaluation Criterion 8: The RMS should be evaluated in terms of its adherence to the characteristics of a trust-based decision-making system that unifies the characteristics and properties of human trust.

As discussed in Chapter 2, a computational trust-based decision-making system should unify key characteristics of human trust. First, a trust-based decision-making system should allow for the specification of confidence in outcome expectations, i.e., that the system can assess how reliable its decision is based on the amount of evidence used to derive the decision. Second, the diversity dimensions of trust should be captured, i.e., trust origin (individual or entity), trust target (individual or entity), and trust purpose. Next, the system should allow for the subjective specification of trust formation, evolution, and exploitation processes, since not all entities have the same perception of evidence. Subjectivity is based on an entity's disposition and belief system, and therefore a system's trust calculation process should allow for user-specific policy to be specified which represents a user's disposition with regard to analysing evidence and making decisions. Fourth, evidence that is based on past behaviour can be directly observed or indirectly recommended and used to update trust both positively and negatively in a dynamic and non-monotonic manner. Fifth, there should be processes to collect and analyse evidence which are made explicit. Sixth, that context can be captured, to include asymmetrical relationships (role), time, environmental factors, and environmental constraints. Seventh, that both agent- and context-specific risk can be captured and assessed. Eighth, that a meaningful and usable measure of trust can be produced from subjective analysis of contextually relevant evidence such that it may be used by a trust origin to be exploited in propagating trust to the

community via recommendations and to make trusting decisions to interact for a given trust purpose with a given trust target in light of associated risk. Finally, the complexity of decision-making is reduced in environments in which uncertainty and risk are present.

4.1.9 Evaluation Methods

In order to evaluate the RMS according to the specified success criteria, we conducted a three-part analysis. First, we describe a set of Internet auction user simulations that we ran to evaluate evaluation criteria 1-4. Next, we present a case study using publicly available Internet auction data to evaluate the potential benefits of our collusion detection mechanism, i.e., evaluation criterion 5. Third, to validate evaluation criteria 6-8, we present a qualitative analysis of the decision-making capability of the RMS in terms of the trust characteristics described in Chapter 2.

4.2 Simulations

This section presents the results of a series of experiments in which extra-auction behaviour is simulated and trustworthiness is assessed based on feedback evidence about that behaviour. First, the simulation environment is described. Then, experimental results and analyses are presented with regard to calculating trustworthiness in terms of the expected likelihood of behaviour using: the basic trust value calculation method on its own; the basic method with the contextual relevance assessment extension; the basic method with the time fading extension; and the basic method with the recommendation weighting extension.

4.2.1 TNG Simulation Environment

We conducted a series of experiments in which extra-auction behaviour is simulated, using a version of the Trade Network Game (McFadzean and Tesfatsion 1999) simulator that has been extended to simulate a simple Internet auction environment and to include the RMS algorithms with regard to identifying domain-specific behaviour based on evidence, assessing contextual relevance in terms of role, time, and environment, and weighting recommendations.

The extended Trade Network Game (TNG) is an evaluation framework for studying the formation and evolution of behaviour among interacting users, i.e., buyers and sellers, of an Internet auction community. Interactions are modelled as 2-person games, in which a seller may behave well, i.e., interact with a buyer and ship to that buyer an item that is as described; may commit fraud, i.e., ship an item that was not as described; or cheat, i.e., not ship any item at all. After each interaction, a buyer records observations about a seller's extra-auction behaviour, thus simulating a feedback process. A buyer records his observations of a seller's behaviour, which updates a trust value for that seller. The observation is appended with information about the time and the item category in which

the interaction took place. A trust value is computed from the observations and is used to calculate a probability that reflects the likelihood of the seller engaging in one of the three specific behaviour types in the future. Using these simulations, we evaluate the RMS's ability to accurately calculate the likelihood of an entity's role-based behaviour in future interactions based on observed domain-specific evidence. Furthermore, we evaluate the contextual relevance assessment mechanism of the RMS with regard to both environment and time. Additionally, we describe the effects of the RMS recommendation weighting method on the simulation results.

4.2.2 TNG Parameters

The following parameters can be specified in the TNG simulations, such that the RMS mechanisms are amenable to statistical evaluation, i.e., allow the identification of factors that contribute to adjusting trust values differently.

First of all, the number of interactions between the two user roles, seller and buyer, is configurable, which allows us to set a number that will show how trustworthiness for a given behaviour type is formed and how it evolves over time.

Next, the behaviour profile of a user acting in the role of seller may be set. This allows us to simulate different behaviour patterns according to:

$$(seller\_behaviour)\begin{cases} B_G \text{ the probability of observing good behaviour, or } \{isd\} \\ B_F \text{ the probability of observing fraudulent behaviour, or } \{is`d\} \\ B_C \text{ the probability of observing cheating behaviour, or } \{i`s\} \end{cases}$$

Our design proposed an effect, or *eff*, function that calculates the effect a piece of evidence will have on a trust value. That is, *eff* determines which pieces of evidence support, contradict, or are inconclusive about the likelihood that a new interaction will result in a specified outcome. Once evidence has been assessed for its effect, the likelihood of observing a behaviour type using the RMS trust value calculation is $\frac{s+1}{s+i+c+2}$, where s observations support the likelihood of observing a behaviour type in a future interaction, i observations are inconclusive about the likelihood of observing a behaviour type in a future interaction, and c observations contradict the likelihood of observing a behaviour type in a future interaction. In our simulations, there is no inconclusive evidence, and an initial likelihood of observing a behaviour type is set to 0.5 such that trustworthiness can be assessed in the case where no observations have yet been made. If we are interested in decision-making based on observations of good behaviour, i.e., that a seller interacts with a buyer and ships an item as described, or $\{isd\}$, then all observations of $\{isd\}$ are supporting evidence and all observations of other behaviour are contradicting, e.g., a seller shipping an item not as described, $\{is`d\}$, or not shipping an item at all, $\{i`s\}$. Alternatively, if we are interested in determining the likelihood of $\{is`d\}$ occurring, all observations of $\{is`d\}$ are supporting evidence and all observations of the other two behaviour types, i.e., $\{isd\}$ and $\{i`s\}$ contradict the likelihood of $\{is`d\}$ occurring.

Finally, if we are interested in assessing the likelihood of $\{i`s\}$ occurring, all observations of $\{i`s\}$ are treated as evidence supporting the proposition and all observations of the other two behaviour types, i.e., $\{isd\}$ and $\{is`d\}$, are treated as evidence contradicting the proposition.

Note that when a seller's behaviour profile is set, he acts randomly in each interaction according to that profile. For example, if a seller's behaviour profile is set to:

$$(seller\_behaviour)\begin{cases} B_G = 0.90 \\ B_F = 0.07 \\ B_C = 0.03 \end{cases}$$

then the seller would ship an item as described in 90% of interactions, ship a fraudulent or otherwise-not-as-described item in 7% of interactions, and not ship any item at all in 3% of interactions. The pattern of individual interactions can change randomly within the profile.

Furthermore, a seller's behaviour profile can also change over a period of interactions, i.e., an update cycle, according to:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } B_G = B_G + \Delta, B_F = B_F - \Delta/2, B_C = B_C - \Delta/2 & \text{(with probability } p_1) \\ \text{Decreases honesty: } B_G = B_G - \Delta, B_F = B_F + \Delta/2, B_C = B_C + \Delta/2 & \text{(with probability } p_2) \\ \text{Unchanging profile: } B_G = B_G, B_F = B_F, B_C = B_C & \text{(with probability } p_3) \end{cases}$$

where the honesty delta, Δ, represents the amount of behavioural change in each update cycle. This parameter allows us to simulate scenarios in which behaviour profiles change over time, e.g., a seller moves from a pattern of good behaviour to a pattern of bad behaviour. Additionally, the update cycle can be specified, so as to control the number of interactions after which a behaviour profile is changed.

For example, if the behaviour profile changes according to:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.33 \\ \text{Decreases honesty: } 0.33 \\ \text{Unchanging profile: } 0.34 \end{cases}$$

when the update cycle = 1 interaction and $\Delta = 0.02$, then after each interaction, 33% of the time there will be in increase in the seller's behavioural likelihood of shipping items as described and he will decrease both the likelihood of shipping items not-as-described and the likelihood of not shipping; 33% of the time, there will be a decrease in the seller's behavioural likelihood of shipping items as described and he will increase both the likelihood of shipping items not-as-described and the likelihood of not shipping; and 34% of the time the seller's profile will not change.

Next, we may specify the number of contexts in terms of environmental factors, e.g., item categories, and the probability of a seller interacting in each item category according to:

$$prob(context\_of\_\text{int}eraction)\begin{cases} prob(ItemCategory_1) \\ prob(ItemCategory_2) \\ prob(ItemCategory_3)\ldots \\ prob(ItemCategory_n) \end{cases}$$

This allows context information about item category for each piece of feedback a buyer records to be appended to feedback, such that evidence can be analysed according to contextual relevance in terms of item category. For example, if a seller typically sells items in Item Category 1 90% of the time and Item Category 2 10% of the time, this behaviour can be specified as:

$$prob(context\_of\_\text{int}eraction)\begin{cases} prob(ItemCategory_1) = 0.90 \\ prob(ItemCategory_2) = 0.10 \end{cases}$$

Recall that our design of the *eff* function also integrates the assessment of which pieces of evidence from an interaction history about a user are contextually relevant to a request. If a piece of evidence has been appended with context information ItemCategory1, then, that information is contextually relevant to a decision being made about interacting in the environmental context of Item Category 1. Thus, the evidence being analysed to calculate trust is the contextually relevant subset of all evidence.

A seller's behaviour profile for a given item category can also be specified, according to:

$$prob(seller\_behaviour\_given\_context)\begin{cases} prob(ItemCategory_nB_G) \\ prob(ItemCategory_nB_F) \\ prob(ItemCategory_nB_C) \end{cases}$$

For example, if a seller is 90% likely to ship items as described in Item Category 1, but only 10% likely to ship items as described in Category 2, this behaviour can be specified as, e.g.,:

$$prob(seller\_behaviour\_given\_context)\begin{cases} Category1B_G = 0.90 \\ Category1B_F = 0.05 \\ Category1B_C = 0.05 \\ Category2B_G = 0.10 \\ Category2B_F = 0.45 \\ Category2B_C = 0.45 \end{cases}$$

A time step and time fading factor may also be specified. The time step is set equal to the number of interactions that make up a given time period. For example, if the time step is 1, then each interaction occurs at a different point in time. If the time step is 100, then a time period refers to 100 interactions within the same step. When the timestamp of each piece of evidence is annotated as t_j, j is the time step associated with a given timestamp, e.g., evidence can be grouped according to time steps by hour, day, month, year. If the time step representing the current time is m, evidence that occurs in a time period before m, i.e., m-j, is faded according to the time fading factor, $\delta \in [0,1]$. Evidence that is further time steps away from the current time step is faded more strongly, whereas evidence that is closer in time to the temporal context in which a decision is being made is faded less strongly and

therefore affects trust calculation more than older evidence. In this way, using the RMS time fading algorithm to assess evidence, the weight of older evidence is faded according to δ and the most recent evidence counts more heavily toward assessment of trustworthiness based on current behaviour. In fact, evidence that falls into time step *m*, the current time period, is not faded at all because it is considered to be as recent as possible.

When $\delta = 1$, no evidence is faded and evidence from every time step is given equal weight. When $\delta = 0$, all evidence that occurs in a time step before *m* is completely ignored. We recall that the outcome of Jøsang et al.'s experiments with the Beta Reputation System (Jøsang and Ismail 2002; Jøsang, Hird et al. 2003) showed that the most reasonable results of time fading were produced when δ was set at .99. Moreover, when defining the period of time steps, it is most effective when specified at the expected rapidity of behavioural change. Therefore, in our simulation environment, the time step should be set to the number of interactions that the update cycle is set to.

Next, a recommendation weighting policy, $\pi \in [0,1]$, may be specified according to how risk averse a user is. When $\pi = 1$, a user considers recommendations to be wholly reliable they are not discounted. When $\pi = 0$, a user considers recommendations to be wholly unreliable and ignores them entirely. However, cases might exist in between these two extremes, e.g.:

$$\pi \in \left\{ \begin{array}{l} 0.99 \text{ - believes recommenders are highly reliable} \\ 0.90 \text{ - believes recommenders are mostly reliable} \\ 0.75 \text{ - believes recommenders are somewhat reliable} \\ 0.50 \text{ - believes recommenders are somewhat unreliable} \\ 0.10 \text{ - believes recommenders are mostly unreliable} \end{array} \right\}$$

Finally, the probability of false positive recommendations may also be specified, according to:

$$prob(false\_positive\_rec) \left\{ \begin{array}{l} prob(false_{FG}) \\ prob(false_{CG}) \end{array} \right.$$

where $false_{FG}$ is the probability that a recommender observes fraudulent behaviour, B_F, but records observations of normal behaviour, *{isd}*; and $false_{CG}$ is the probability that a recommender observes theft, B_C, but records observations of normal behaviour, *{isd}*.

The following sections detail the parameter specification for each set of simulation experiments, as well as presenting results and analyses.

4.2.3 Domain-Specific Evidence Collection and Analysis

The following two experiments produce results that allow us to describe the accuracy of the basic RMS trust calculation mechanism, i.e., the trust calculation with no assessment of contextual relevance, time, or recommendation reliability, with regard to its ability to assess the likelihood of the occurrence of a certain kind of behaviour. The experiments allow us to evaluate how well the

mechanism calculated the likelihood of observing a specific behaviour based on evidence about a user's trustworthiness in past interactions. We are interested in seeing how the RMS performs in each of the following four cases: case 1, in which a user acting in a given role, e.g., seller, exhibits consistently normal behaviour; case 2, when a user exhibits consistently malicious behaviour; case 3, in which a user is new or unknown to the community; and case 4, the case in which a user oscillates unpredictably between normal and malicious behaviour.

The first experiment simulates the case in which a seller starts out as a new user and subsequently his behaviour over time is consistent for each of the three behaviour types, thereby allowing us to analyse the performance of the RMS trust calculation mechanism for cases 1-3. The second experiment simulates the case in which a seller's behaviour fluctuates randomly between good and bad over time, thereby allowing us to analyse the performance of the RMS trust calculation mechanism for case 4. For each experiment, the parameters of the simulation environment are noted, results are presented, and an analysis of results is given.

4.2.3.1 Experiment 1.1: consistent behaviour

In this experiment, a seller starts out as a new user interacting with a buyer in the Internet auction simulations, and subsequently his behaviour over time is consistent for each of the three behaviour types.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.90 \\ B_F = 0.07 \\ B_C = 0.03 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 90% of the time, ships a lesser-quality item 7% of the time, and not ship any item at all 3% of the time.
- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.0 \\ \text{Decreases honesty: } 0.0 \\ \text{Unchanging profile: } 1.0 \end{cases}$$

- No contextual relevance assessment.
- No time fading.
- No recommendation weighting.

Results

We ran this simulation experiment 10 times, and the aggregated results are illustrated in Figures 40 and 41. For each domain-specific behaviour type captured, i.e., {*isd*}, {*is`d*}, and {*i`s*}, the mechanism was able to record observations and to analyse the observations to calculate the likelihood of the behaviour type's occurrence.

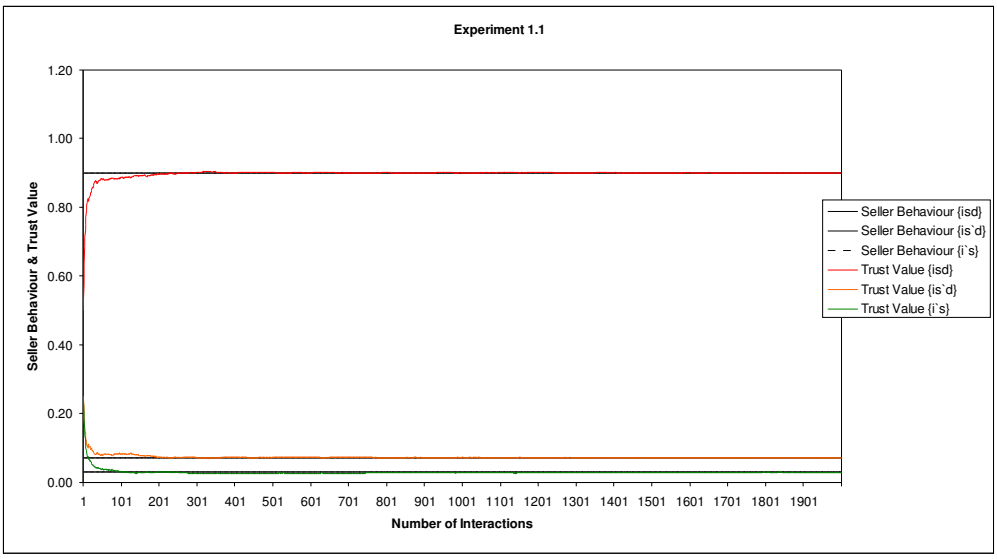

Figure 40: Experiment 1.1 consistent behaviour

When a seller exhibited consistent good behaviour, that is, he shipped items as described, in 90% of interactions; the mechanism calculated the likelihood of good seller behaviour with a mean of 0.897, standard deviation of 0.017, and mean absolute difference (MAD) between likelihood of behaviour and actual behaviour of 0.003.

When a seller exhibited consistent bad behaviour of the type {*is`d*}, i.e., he shipped items that were not as described, in 7% of interactions, the mechanism calculated the likelihood of seller behaviour of {*is`d*} with a mean of 0.073, standard deviation of 0.008, and MAD of 0.003.

When a seller exhibited consistent bad behaviour of the type {*i`s*}, that is, he did not ship items at all, in 3% of interactions, the mechanism calculated the likelihood of seller behaviour of {*i`s*} with a mean of 0.029, standard deviation of 0.010, and MAD of 0.003.

For each of the behaviour types, likelihood of behaviour occurring based on the RMS trust calculation converges on the mean entity behaviour. Based on these results, we can say that the basic RMS trust calculation mechanism produces a trust value that reflects the likelihood of entity behaviour to nearly

100% accuracy for each domain-specific type of role-based behaviour. However, it is important to note that this trust value reflects mean entity behaviour over time, and it does not react strongly to actual rises and falls in a seller's behavioural honesty in each individual interaction.

Figure 2 allows us to better view the results of initial interactions between entities. When nothing is known about a user, the RMS calculates a trust value reflecting uncertainty with regard to the likelihood of what behaviour will occur. The likelihood of good behaviour when there is no evidence is calculated as 0.5, and the initial likelihood of bad behaviour is split equally between the two types of bad behaviour, at 0.25 likelihood of observing $\{is`d\}$ and 0.25 likelihood of observing $\{i`s\}$. For each of the three behaviour types, as evidence is accrued about the seller's exhibited behaviour, the likelihood of observing behaviour quickly converges on the mean actual behaviour. By also considering the total number of observations being used as evidence to calculate a trust value, the RMS trust calculation mechanism can distinguish between the case where there is no evidence about a user as opposed to the case in which a seller acts badly, in which the RMS adjusts the trust value to reflect the bad behaviour.

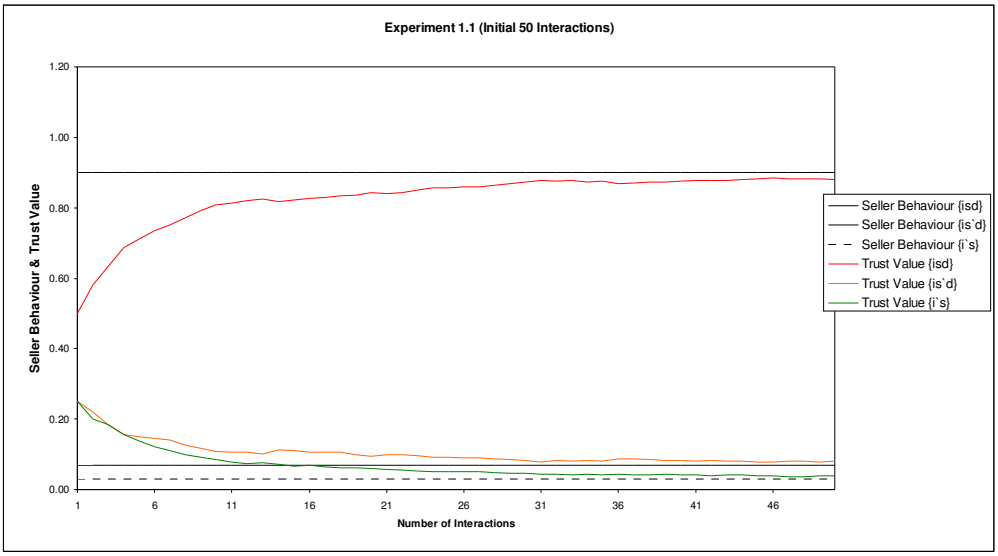

Figure 41: Experiment 1.1 (initial 50 interactions) consistent behaviour

Having demonstrated the ability of the RMS to accurately assess the likelihood of occurrence of key domain-specific behaviour, the remainder of the simulations will assess the likelihood of occurrence of good behaviour, i.e., $\{isd\}$.

4.2.3.2 Experiment 1.2: oscillating behaviour

In this experiment, a seller starts out as a new user interacting with a buyer in the Internet auction simulations, and subsequently his behaviour oscillates between the three seller behaviour types, thereby allowing us to analyse the performance of the RMS trust calculation mechanism in assessing the likelihood of observing oscillating behaviour.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.90 \\ B_F = 0.07 \\ B_C = 0.03 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 90% of the time, ships a lesser-quality item 7% of the time, and not ship any item at all 3% of the time.
- Seller's behaviour profile oscillates in terms of honesty, i.e., it is updated over time according to:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.33 \\ \text{Decreases honesty: } 0.33 \\ \text{Unchanging profile: } 0.34 \end{cases}$$

- Δ, amount of behavioural change in each update cycle: 0.02.
- Update cycle: 1 interaction.
- No contextual relevance assessment.
- No time fading.
- No recommendation weighting.

Results

We ran this simulation experiment 10 times, and the aggregated results are illustrated in Figure 3.

The seller's behaviour changes in a highly random manner from one interaction to the next, with mean behaviour being 0.79, but oscillating frequently between increasing and decreasing levels of honesty. The RMS calculates a basic trust value based on observations about a seller's behaviour and computes likelihood of observing $\{isd\}$ with a MAD from seller behaviour of 0.067. It is important to note that this trust value reflects mean entity behaviour over time, and it does not react strongly to actual rises and falls in a seller's behavioural honesty. Thus in the case where an entity's behaviour fluctuates greatly between honest and dishonest, a trust value can approximate the mean behaviour but is an

average of nearly 6% less accurate for predicting the actual likelihood of an interaction with the seller resulting in {*isd*} than in the experiment in which a seller's behaviour was consistent over time.

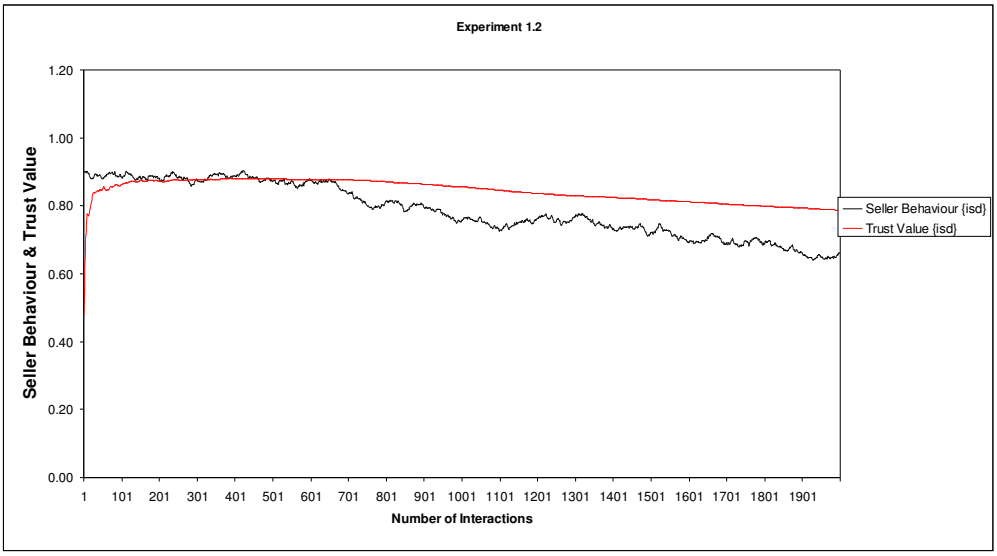

Figure 42: Experiment 1.2 oscillating behaviour

4.2.3.3 Conclusions about the basic RMS trust calculation mechanism

In the simulations, the RMS feedback mechanism collects simulated observation records of a type and granularity required to identify domain-specific behaviour types, such as normal behaviour, fraudulent behaviour, and thieving behaviour, and thus to limit risk of exposure to user interaction with untrustworthy entities. The basic RMS trust calculation mechanism is highly accurate for identifying behaviour types for sellers with consistent behaviour patterns, e.g., a seller who acts honestly 90% of the time, delivers counterfeit goods 7% of the time, and does not ship any goods at all in 3% of interactions. In these cases, while sensitive to new evidence during initial interactions, over time the trust value converges on mean behaviour. Furthermore, the RMS can calculate initial trustworthiness in the case where no observations have yet been made about a user.

When a seller's behaviour profile is extremely unpredictable, i.e., changes in a highly random manner with each interaction, as in Experiment 1.2, the basic RMS trust value becomes less accurate as a basis for calculating likelihood of expected behaviour on a per-interaction basis, and is on average 93% accurate with respect to actual seller behaviour.

This experiment also allows us to conclude that when observations are recorded about domain-specific behaviour, the RMS can identify user behaviour patterns based on the evidence that it would not be able to identify when only general positive or negative feedback statements are collected. By

simply slightly increasing the granularity level of the type of observations recorded, i.e., not reducing ease-of-use of the evidence collection process, the mechanism can therefore increase the accuracy of making decisions based on the actual types of behaviour that occurs in the Internet auction domain. Moreover, when observations are recorded about specific behaviour types, the mechanism can analyse evidence to make decisions about the likelihood of occurrence of normal behaviour, fraud, or theft, thus reducing complexity for a user who no longer would have to manually assess feedback records to identify behaviour patterns.

4.2.4 Context-Specific Evidence Collection and Analysis

The following two experiments produce results that allow us to evaluate the ability of the RMS contextual relevance mechanism to assess the likelihood of domain-specific entity behaviour in a given context based on observations of behaviour in that context. The experiments allow us to evaluate how well the mechanism calculates the likelihood of observing normal behaviour in more than one environmental context, i.e., item category, based on observed evidence about a user's trustworthiness in past interactions in that context. We are interested in comparing the calculation of the likelihood of non-context-specific behaviour with that of context-specific behaviour to evaluate which calculation provides a more accurate analysis of actual behaviour.

The first experiment simulates the case in which a seller's behaviour over time is consistently mostly good and only sometimes bad, but in which the good behaviour occurs in one context while the bad behaviour occurs in a second context. The second experiment extends the first experiment to simulate a scenario in which a seller varies behaviour across four contexts.

4.2.4.1 Experiment 2.1: behaviour in two environmental contexts

In this experiment, a seller interacts with a buyer in two environmental contexts within the Internet auction simulations, and his behaviour over time is consistent for each context, thereby allowing us to analyse the performance of the RMS contextual relevance assessment mechanism's ability to produce a contextually relevant trust value that can be used to calculate the likelihood of observing consistent behaviour in two different contexts.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Number of contexts, i.e., item categories: 2.
- Probability of seller interacting in each context according to:

$$prob(context\_of\_interaction)\begin{cases} prob(ItemCategory_1) = 0.90 \\ prob(ItemCategory_2) = 0.10 \end{cases}$$

- Seller behaviour profile in each context according to:

$$prob(seller\_behaviour\_given\_context)\begin{cases} Category1B_G = 0.90 \\ Category1B_F = 0.05 \\ Category1B_C = 0.05 \\ Category2B_G = 0.10 \\ Category2B_F = 0.45 \\ Category2B_C = 0.45 \end{cases}$$

  That is, the seller interacts over the period of interactions according to the following: in Category 1, the seller exhibits good behaviour 90% of the time, ships a lesser-quality item 5% of the time, and not ship any item at all 5% of the time; and in Category 2, the seller exhibits good behaviour 10% of the time, ships a lesser-quality item 45% of the time, and does not ship any item at all 45% of the time.

- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: 0.0} \\ \text{Decreases honesty: 0.0} \\ \text{Unchanging profile: 1.0} \end{cases}$$

- Contextual relevance is assessed.
- No time fading.
- No recommendation weighting.

Results

We ran this experiment 10 times, and the aggregate results of the simulations are illustrated in Figure 4. The RMS's contextual relevance assessment mechanism was able to analyse observations that had been recorded and appended with context information in terms of environmental context and to calculate the contextually relevant likelihood of a behaviour type's occurrence in two contexts.

When a seller exhibited consistent good behaviour in Item Category 1, that is, he shipped items as described, in 90% of interactions, the mechanism calculated the likelihood of good seller behaviour in Item Category 1 with a mean of 89%. The likelihood of expected good behaviour varies from actual good behaviour with a MAD of 0.009.

When a seller exhibited consistent good behaviour in Item Category 2 in only 10% of interactions, the mechanism calculated the likelihood of good seller behaviour in Item Category 1 with a mean of 11%. The likelihood of expected good behaviour varies from actual good behaviour with a MAD of 0.011.

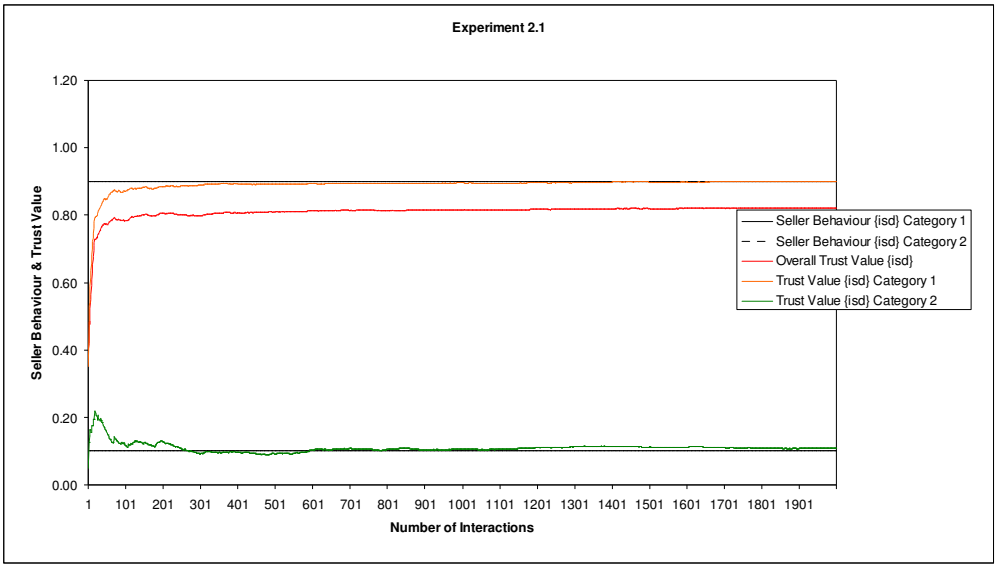

Figure 43: Experiment 2.1 behaviour in two environmental contexts

For the purpose of comparison, Figure 4 also gives the plot of overall trust, i.e., the likelihood of expected behaviour when contextual relevance has not been assessed. Mean likelihood of expected behaviour when item category is not taken into account is 0.81. This varies from actual good behaviour in Item Category 1 with a MAD of 0.09, and it varies from actual good behaviour in Item Category 2 with a MAD of 0.79. Therefore, when contextual relevance is not assessed, the mechanism produces a trust value which is less accurate for calculating likelihood of context-specific behaviour than the trust values which are produced when contextual relevance is assessed.

This evaluation produces an intuitive and yet significant result: the assessment of contextual relevance is key to decision-making in an environment in which entities may interact in more than one context. In this experiment, were contextual relevance not assessed, the level of trust a buyer would calculate about a seller of the specified profile would be 81% trustworthy with regard to delivering goods as described. This is nearly 10% inaccurate for the seller's actual trustworthiness in Item Category 1, and even more inaccurate when one considers extremely high levels of bad behaviour the seller exhibits in Category 2. In a system in which contextual relevance is not assessed, a seller such as this could build up a high reputation in one context so as to mask his bad behaviour in another context. This threat is mitigated against when contextual relevance assessment is used to detect such behaviour.

4.2.4.2 Experiment 2.2: behaviour in four environmental contexts

This experiment extends Experiment 2.1. In this experiment, a seller interacts with a buyer in four environmental contexts within the Internet auction simulations, and his behaviour over time is consistent for each context, thereby allowing us to analyse the performance of the RMS contextual relevance assessment mechanism's ability to produce a contextually relevant trust value that can be used to calculate the likelihood of observing consistent behaviour in several different contexts.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Number of contexts, i.e., item categories: 4.
- Probability of seller interacting in each context according to:

$$prob(context\_of\_\text{int}eraction)\begin{cases} prob(ItemCategory_1) = 0.4 \\ prob(ItemCategory_2) = 0.3 \\ prob(ItemCategory_3) = 0.2 \\ prob(ItemCategory_4) = 0.1 \end{cases}$$

- Seller behaviour profile in each context according to:

$$prob(seller\_behaviour\_given\_context)\begin{cases} Category1B_G = 0.90 \\ Category1B_F = 0.05 \\ Category1B_C = 0.05 \\ Category2B_G = 0.70 \\ Category2B_F = 0.10 \\ Category2B_C = 0.20 \\ Category3B_G = 0.5 \\ Category3B_F = 0.25 \\ Category3B_C = 0.25 \\ Category4B_G = 0.10 \\ Category4B_F = 0.45 \\ Category4B_C = 0.45 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: in Category 1, the seller exhibits good behaviour 90% of the time, ships a lesser-quality item 5% of the time, and not ship any item at all 5% of the time; in Category 2, the seller exhibits good behaviour 70% of the time, ships a lesser-quality item 10% of the time, and does not ship any item at all 20% of the time; in Category 3, the seller exhibits good behaviour 50% of the time, ships a lesser-quality item 25% of the time, and does not ship any item at all 25% of the time; and in Category 4, the seller exhibits good behaviour 10% of the time, ships a lesser-quality item 45% of the time, and does not ship any item at all 45% of the time.

- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases}\text{Increases honesty: } 0.0\\ \text{Decreases honesty: } 0.0\\ \text{Unchanging profile: } 1.0\end{cases}$$

- Contextual relevance is assessed.
- No time fading.
- No recommendation weighting.

Results

We ran this experiment 10 times, and the aggregate results of the simulations are illustrated in Figure 5. As in the previous experiment, the mechanism was able to analyse observations that had been recorded and appended with context information in terms of environmental context and to calculate the contextually relevant likelihood of a behaviour type's occurrence in multiple contexts.

When a seller exhibited consistent good behaviour in Item Category 1, that is, he shipped items as described, in 90% of interactions; the mechanism calculated the likelihood of good seller behaviour in Item Category 1 with a mean of 89%. The likelihood of expected good behaviour in Item Category 1 varies from actual good behaviour in Item Category 1 with a MAD of 0.014.

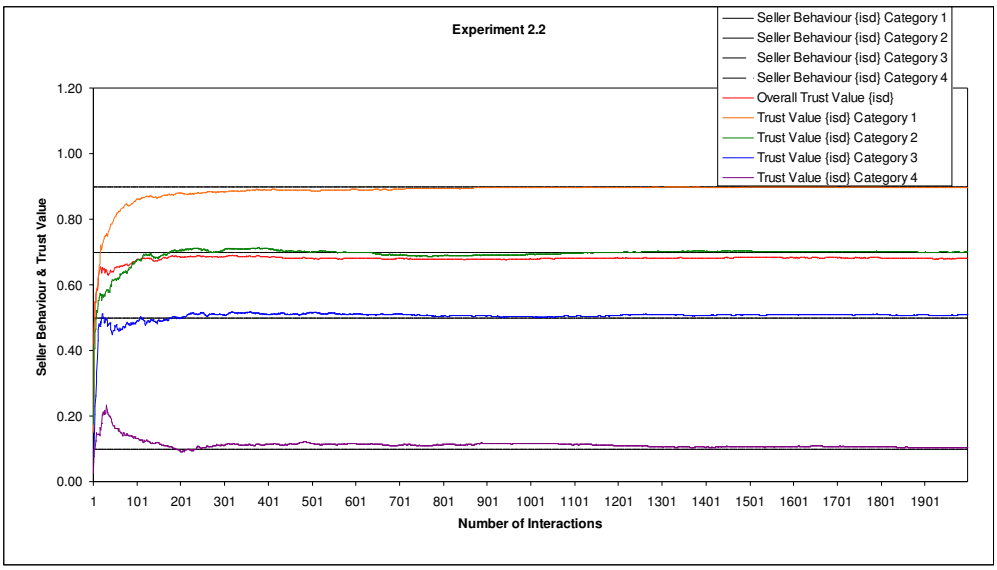

Figure 44: Experiment 2.2 behaviour in four environmental contexts

When a seller exhibited consistent good behaviour in Item Category 2 in only 70% of interactions, the likelihood of good seller behaviour in Item Category 2 was calculated with a mean of 69%. The

likelihood of expected good behaviour in Item Category 2 varies from actual good behaviour in Item Category 2 with a MAD of 0.010.

When a seller exhibited consistent good behaviour in Item Category 3 in only 50% of interactions, the trust value calculation mechanism calculated the likelihood of good seller behaviour in Item Category 3 with a mean of 50%. The likelihood of expected good behaviour in Item Category 3 varies from actual good behaviour in Item Category 3 with a MAD of 0.010.

When a seller exhibited consistent good behaviour in Item Category 4 in only 10% of interactions, the mechanism calculated the likelihood of good seller behaviour in Item Category 4 with a mean of 11%. The likelihood of expected good behaviour in Item Category 4 varies from actual good behaviour Item Category 4 with a MAD of 0.013.

As in the previous experiment, for the purpose of comparison, Figure 5 also gives the plot of overall trust, i.e., the likelihood of expected behaviour when contextual relevance has not been assessed. Mean likelihood of expected good behaviour when item category is not taken into account is 0.68. This varies from actual good behaviour in Item Category 1 with a MAD of 0.221, in Item Category 2 with a MAD of 0.021, in Item Category 3 with a MAD of 0.180, and in Item Category 4 with a MAD of 0.579. Therefore, as were the conclusions in Experiment 2.1, in Experiment 2.2 when contextual relevance is not assessed, the mechanism produces a trust value which is less accurate for calculating likelihood of context-specific behaviour than the trust values which are produced when contextual relevance is assessed.

In this experiment, were contextual relevance not assessed, the level of trust a buyer would calculate about a seller of the specified profile would be 68% trustworthy with regard to delivering goods as described. This is nearly 22% inaccurate for the seller's actual trustworthiness in Item Category 1, 2% inaccurate for the seller's actual trustworthiness in Item Category 2, 18% inaccurate for the seller's actual trustworthiness in Item Category 3, and 58% inaccurate for the seller's actual trustworthiness in Item Category 4. In a mechanism in which contextual relevance is not assessed, a seller such as this could maintain a reasonably high reputation for good behaviour while behaving quite differently across different contexts.

This experiment further supports the results in Experiment 2.1, i.e., a seller can behave very well in many transactions in some contexts in order to maintain a high overall reputation while behaving badly in a few interactions in other contexts. Through examining evidence for contextual relevance and calculating a contextually relevant trust value, this kind of reputation-masking behaviour can be identified.

4.2.4.3 Conclusions about the RMS contextual relevance mechanism

These two experiments illustrate that a basic overall trust calculation is not as accurate for a given context as a contextually relevant trust value. A seller can maintain a fairly high overall trust value by

behaving well in a context(s) in which he mainly interacts, e.g., Item Category 1, which can mask his untrustworthy behaviour in another context, e.g., Item Category 4. This scenario becomes more interesting if we say that Item Category 1 is typically a low-cost context, i.e., one in which items are low in price, and Item Category 4 is a high-cost context, i.e., one in which items are high-priced. In the case where context is not assessed (and especially if the risk of potential financial loss is not made explicit), a seller could be making some very big financial gains by acting badly in Item Category 4 while losing very little by acting well in Item Category 1. We can also apply these results to the case in which the contextual element of role is not assessed, i.e., an entity could build a very high reputation acting in the role of a buyer, e.g., of very low-cost items, and then use that high level of trustworthiness to act badly in the role of a seller of a high ticket item. Our mechanism increases accuracy of trust value calculation for decision-making in such scenarios.

The RMS can calculate a trust value that assesses the likelihood of domain-specific entity behaviour in a given context based on observations of that behaviour in the context. This calculation provides evidentiary analysis with regard to contextual relevance in terms of role-specific behaviour, e.g., the likelihood of a user in the seller role shipping an item as described, and environment, e.g., the likelihood of a user behaving well in Item Category 1. By automating the analysis of the context information that is appended to evidence, the RMS contextual relevance mechanism can produce a context-specific trust value that is more accurate than a non-context-specific trust value for calculating the likelihood of a user's behaviour in a given context. Moreover, the mechanism's ability to automatically assess contextual relevance also decreases complexity and increases usability for users who would no longer need to manually filter through an abundant body of evidence to determine which evidence was more relevant to a decision at hand.

4.2.5 Time-Specific Evidence Collection and Analysis

The following two experiments produce results that allow us to evaluate the ability of the RMS time-fading mechanism to produce a time-faded trust value that is used to calculate the likelihood of domain-specific entity behaviour in a given time period based on observations of that behaviour in the time period.

We constructed both experiments with the same parameter specifications as those of Experiment 1.1 and 1.2, except that we change the time-fading parameter so that we can evaluate the effects of time-fading on a trust value to calculate expected likelihood of good behaviour in both the case in which a seller's behaviour profile is consistent and in the case in which a seller's behaviour profile oscillates randomly with regard to honesty levels. That is, the first experiment simulates the case in which a seller's behaviour over time is consistently mostly good and only sometimes bad. The second experiment simulates the case in which a seller's behaviour fluctuates randomly between good and bad over time.

4.2.5.1 Experiment 3.1: consistent behaviour with time fading

In this experiment, a seller interacts with a buyer within the Internet auction simulations, and his behaviour profile over time is consistent. Evidence about the seller's behaviour is assessed according to timeliness, i.e., evidence about current behaviour is given more weight than older evidence, thereby allowing us to analyse the performance of the RMS's time-fading mechanism's ability to produce a time-faded trust value that can be used to calculate the likelihood of observing specific behaviour when actual behaviour remains consistent.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.90 \\ B_F = 0.07 \\ B_C = 0.03 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 90% of the time, ships a lesser-quality item 7% of the time, and does not ship any item at all 3% of the time.
- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.0 \\ \text{Decreases honesty: } 0.0 \\ \text{Unchanging profile: } 1.0 \end{cases}$$

- No contextual relevance assessment.
- Time fading is applied to evidence.
- Time step: 1.
- Time fading factor, δ: 0.99.
- No recommendation weighting.

Results

We ran this experiment 10 times, and the aggregate results of the simulations are illustrated in Figure 6. The RMS's time-fading mechanism was able to analyse observations that had been recorded and appended with information in terms of temporal context and to produce a the time-faded trust value that was used to calculate the likelihood of a behaviour type's occurrence over time. We compare the results of this experiment with those of Experiment 1.1, in which seller behaviour is also consistent over time.

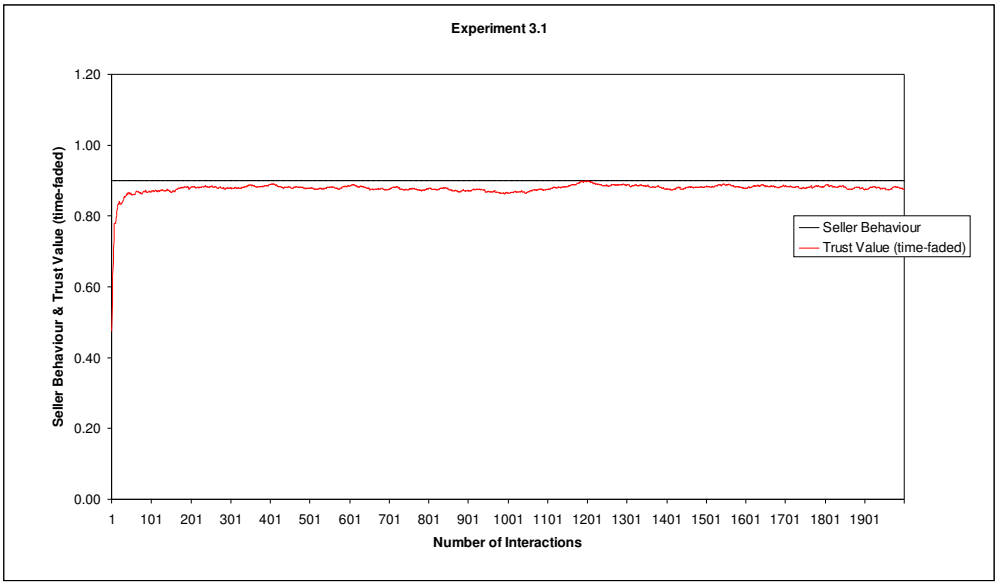

Figure 45: Experiment 3.1 consistent behaviour with time-fading

When a seller exhibited consistent good behaviour, that is, he shipped items as described, in 90% of interactions; the mechanism produced a time-faded trust value used to calculate the likelihood of good seller behaviour with a mean of 0.88, standard deviation of 0.017, and mean absolute difference (MAD) between likelihood of behaviour and actual behaviour of 0.022. Thus, the RMS's time fading mechanism produces a time-faded trust value that reflects mean entity behaviour that is 2% less accurate than the non-time-faded trust value in the case where a seller is behaves consistently over time. Although the trust value is less accurate according to mean entity behaviour than the non-time-faded trust value calculated in Experiment 1.1, the trust value is weighted towards current behaviour and therefore reacts more quickly to entity behaviour changes. This result becomes more apparent in the following experiment where seller behaviour oscillates between profiles.

4.2.5.2 Experiment 3.2: oscillating behaviour with time fading

In this experiment, a seller interacts with a buyer within the Internet auction simulations, and his behaviour profile over time is consistent. Evidence about the seller's behaviour is assessed according to timeliness, i.e., evidence about current behaviour is given more weight than older evidence, thereby allowing us to analyse the performance of the RMS's time-fading mechanism's ability to produce a time-faded trust value that can be used to calculate the likelihood of observing behaviour when actual behaviour oscillates unpredictably.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.90 \\ B_F = 0.07 \\ B_C = 0.03 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 90% of the time, ships a lesser-quality item 7% of the time, and not ship any item at all 3% of the time.
- Seller's behaviour profile oscillates in honesty, i.e., it is updated over time according to:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.33 \\ \text{Decreases honesty: } 0.33 \\ \text{Unchanging profile: } 0.34 \end{cases}$$

- Δ, amount of behavioural change in each update cycle: 0.02.
- Update cycle: 1 interaction.
- No contextual relevance assessment.
- Time fading is applied to evidence.
- Time step: 1.
- Time fading factor, δ: 0.99.
- No recommendation weighting.

Results

We ran this experiment 10 times, and the aggregate results of the simulations are illustrated in Figure 7. The RMS's time-fading mechanism was able to analyse observations that had been recorded and appended with information in terms of temporal context and to produce a time-faded trust value that was used to calculate the likelihood of a behaviour type's occurrence over time. We compare the results of this experiment with those of Experiment 1.2, in which seller behaviour is also oscillating unpredictably over time.

The seller's behaviour changes in a highly random manner from one interaction to the next in this experiment, with mean behaviour being 0.75, but oscillating frequently between increasing and decreasing levels of honesty with a standard deviation from mean behaviour of 0.08. The RMS time-fading mechanism produced a time-faded trust value used to calculate the likelihood of good seller behaviour with a mean of 0.75, standard deviation of 0.079, and mean absolute difference (MAD) between likelihood of behaviour and actual behaviour of 0.018. Recall that the MAD in Experiment 1.2 was of 0.067.

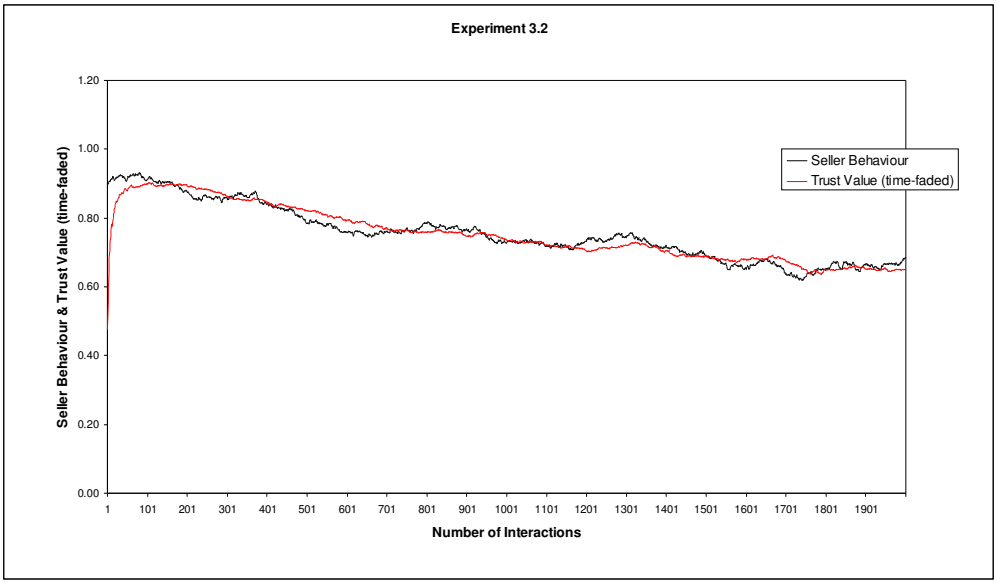

Figure 46: Experiment 3.2 oscillating behaviour with time-fading

Comparing the graphs for Experiments 1.2 and 3.2, it is immediately obvious that the time-faded trust value which is weighted toward current behaviour more closely tracks actual entity behaviour and is more responsive to behavioural changes when honesty levels increase or decrease. In fact, when we compare the MADs for these experiments, we find that when the value of old evidence is faded and a trust value is weighted toward observations about current behaviour, the RMS produces a trust value that is approximately 5% more accurate than a non-time-faded trust value for calculating the likelihood of seller behaviour.

4.2.5.3 Conclusions with regard to the accuracy of RMS when time fading is assessed

In Experiment 3.2, the time-faded trust value reacts more strongly to fluctuations in behaviour, i.e., the trust value is weighted towards current behaviour and therefore follows more closely to entity behaviour patterns rather than converging on mean behaviour as does the basic RMS trust value calculation. In domains in which current behaviour is perceived to be a better indicator than past behaviour of likely future behaviour, the RMS's time fading mechanism allows evidence about past behaviour to be faded and thereby gives more weight to evidence about current behaviour.

Moreover, we see a 5% increase in accuracy from Experiment 1.2 to Experiment 3.2, whereby the time-faded trust value that is weighted in favour of most recent behaviour gives a more accurate calculation of the likelihood of *actual* seller behaviour rather than *mean* seller behaviour. Again, automating the evidentiary analysis process, in this case with regard to temporal context, assists in providing usability through complexity reduction.

4.2.6 Limiting Exposure to Risk of Unreliable Evidence

The following four experiments were constructed to evaluate the ability of the recommendation weighting mechanism to limit a user's exposure to the risk of unreliable evidence, i.e., false positive recommendations. The first experiment demonstrates the effects of applying different recommendation weighting policies as the number of hops of a recommendation path is extended. In this experiment, no false positive recommendations are recorded. The three subsequent experiments demonstrate the result of weighting recommendations in the case where recommenders record false positive recommendations and the recommendation path length is a constant of one hop.

4.2.6.1 Experiment 4.1.1: varying recommendation weighting policy and path length

In this experiment, we demonstrate how exposure to risk of unreliable evidence may be limited when a recommendation weighting policy is applied. Different recommendation weighting policies are specified, and recommendation path length is specified to two hops.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.90 \\ B_F = 0.07 \\ B_C = 0.03 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 90% of the time, ships a lesser-quality item 7% of the time, and not ship any item at all 3% of the time.
- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.0 \\ \text{Decreases honesty: } 0.0 \\ \text{Unchanging profile: } 1.0 \end{cases}$$

- No contextual relevance assessment.
- No time fading.
- Recommendation weighting policy, π, where $\pi \in \{0.99, 0.90, 0.75, 0.50, 0.10\}$
- Recommendation path length, l, where $l \in \{1, 2\}$
- No false positive recommendations.

Results

Figure 8 illustrates the decrease in risk exposure when recommendation weighting is utilised with different recommendation weighting policies. For example, when the recommendation weighting policy is optimistic, i.e., $\pi = 0.99$, and a recommender is one hop away in a recommendation path

from a decision-maker, the recommender's recommendations are weighted at 99% of the full weight. In this case, the recommendation-weighted trust value produced by the trust calculation mechanism calculates likelihood of good seller behaviour at 1% less than a non-recommendation-weighted trust value. Thus forming a recommendation-weighted trust value based on recommendations provided from a recommender that is one hop away in a recommendation path decreases risk exposure to a decision-maker by 1% when 0.99 is the recommendation weighting policy specified.

To explain further, the unweighted trust value in this experiment is used to calculate likelihood of expected behaviour with a mean of 90% {*isd*}, which converges on actual seller mean behaviour. If this trust value were based on the recommendations of a recommender that is one hop away, however, the mean recommendation-weighted trust value would calculate likelihood of expected behaviour with a mean of 89%. As a recommender becomes further away from a decision-maker in terms of path length, risk exposure is further limited. For example, for a path length $l = 2$, recommendations are weighted by 0.98, and the mean absolute difference between the likelihoods based on the recommendation-weighted trust value and the unweighted trust value is 2%. Thus, as is intuitive, as a recommender becomes further away from a decision-maker in a recommendation path, less weight is given to that recommender's recommendations, thereby exposing the decision-maker to decreased risk of interacting based on recommendations that may be somewhat unreliable. Because the decision-maker's view of the recommender world is very optimistic in this scenario, i.e., the decision-maker believes that recommenders are mostly reliable, as the path increases between decision-maker and recommender, trustworthiness is underestimated which results in a risk exposure limitation of about an additional 1% for each hop in the path.

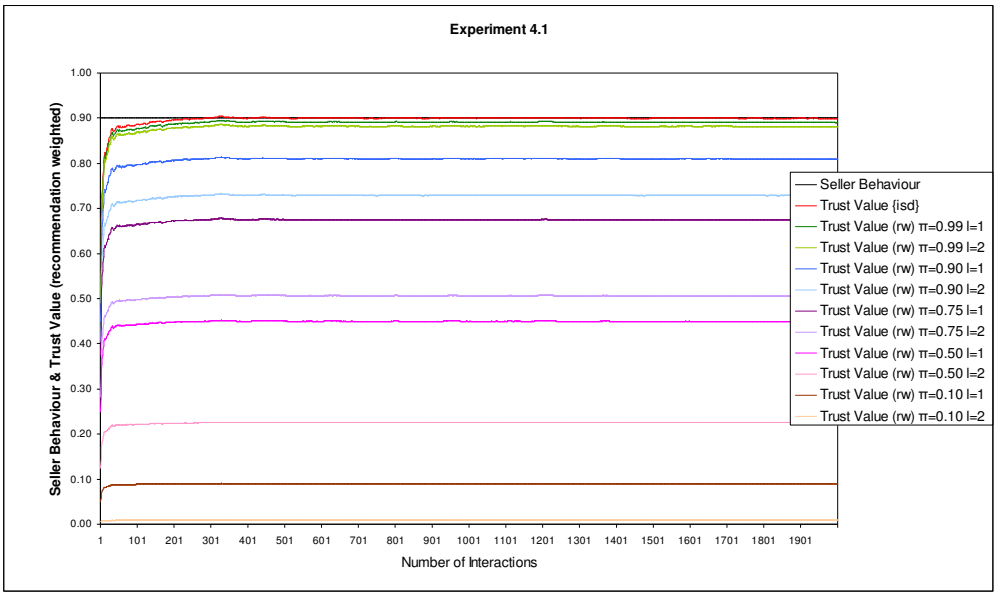

Figure 47: Experiment 4.1 varying recommendation weighting policy and path length

As illustrated, when the recommendation weighting policy is only slightly less optimistic, such as 0.90, reasonable results for risk limitation are produced, i.e., as recommendation path length increases, trustworthiness is decreased increasingly in regular steps. However, more risk averse recommendation weighting policy specifications, e.g., 0.75, 0.50, or 0.10, decrease trust in large steps that quickly result in a such a low estimation of trustworthiness as to leave a decision-maker exposed to extremely little risk that may prevent the decision-maker from the benefits of interacting. These policies parallel real world human risk strategies, whereby some people are more trusting and interact often, even if sometimes interaction results in negative results, and others choose to assume less risk and by the same token significantly underestimate trustworthiness and pass up interactions that could have been beneficial.

4.2.6.2 Experiment 4.2.1: varying recommendation weighting policy; 10% unreliable recommendations

In this experiment, a seller's behaviour is consistent. The recommendations of the buyer with whom the seller interacts are falsely positive in 10% of interactions. The recommendation weight is varied to demonstrate the effects of using a different recommendation weighting policy in the given scenario. Recommendation path length is fixed at one hop.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.50 \\ B_F = 0.0 \\ B_C = 0.50 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 50% of the time, ships a lesser-quality item in no cases, and not ship any item at all 50% of the time.
- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.0 \\ \text{Decreases honesty: } 0.0 \\ \text{Unchanging profile: } 1.0 \end{cases}$$

- No contextual relevance assessment.
- No time fading.
- Recommendation weighting policy, π, where $\pi \in \{0.99, 0.90, 0.75, 0.50, 0.10\}$
- Recommendation path length, l, where $l \in \{1\}$
- False positive recommendations:

$$prob(false\_positive\_rec)\begin{cases} prob(false_{FG}) = 0.0 \\ prob(false_{CG}) = 0.10 \end{cases}$$

Results

Figure 9 illustrates the decrease in risk exposure when recommendation weighting is utilised with different recommendation weighting policies. In this case, the unweighted trust value produced by the RMS trust calculation mechanism calculates likelihood of good seller behaviour at an amount that is nearly 10% greater than actual mean seller behaviour. This makes sense intuitively, because the trust value is based on evidence that includes 10% false positive recommendations. The recommendation-weighted trust values have the effect of decreasing the calculated likelihood of good behaviour and underestimating trustworthiness so as to ensure risk limitation. For example, when the trust calculation mechanism weights the trust value according to π = 0.99, a likelihood of good seller behaviour is produced that limits risk of unreliable recommendations by 1%, and more severe recommendation weighting policies reduce risk even further.

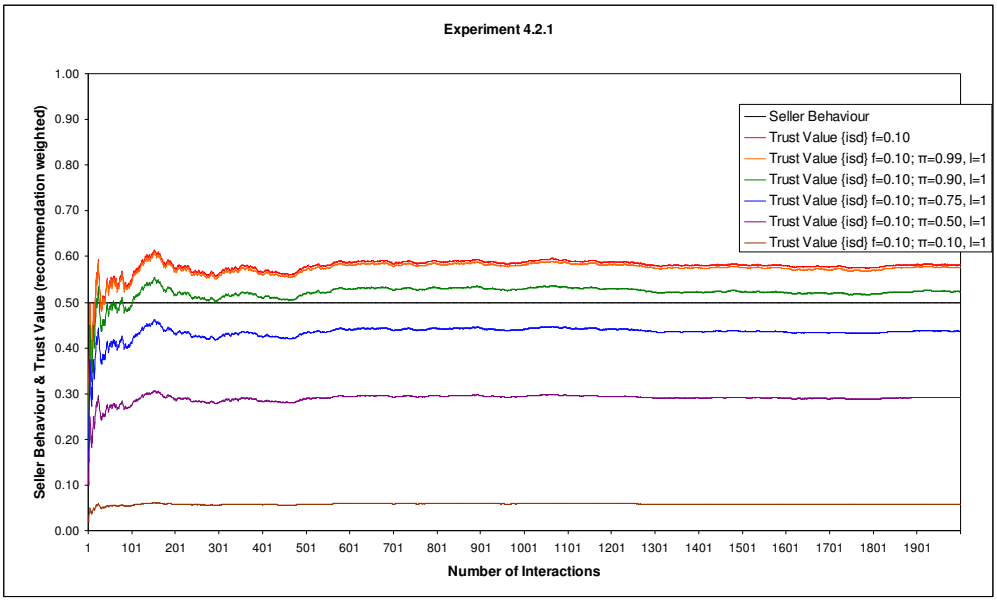

Figure 48: Experiment 4.2.1 10% unreliable (false positive) recommendations

In this scenario, when π = 0.99, the mean likelihood of good seller behaviour is only slightly lower than that produced by the unweighted trust value. When π = 0.75, 0.50, and 0.10, the calculated likelihood of behaviour is increasingly lower than actual mean behaviour, respectively. When the recommendation weighting policy, π = 0.90, is used to weight a trust value, a likelihood of seller

behaviour is produced that is more accurate to actual seller behaviour. Note that when $\pi = 0.90$, trust values are discounted 10%, an amount that is equal to the amount of false positive recommendations used as evidence in this experiment.

4.2.6.3 Experiment 4.2.2: varying recommendation weighting policy; 25% unreliable recommendations

In this experiment, a seller's behaviour is, again, consistently 50% good, 50% bad, but in this case, the recommendations of the buyer with whom the seller interacts are falsely positive in 25% of interactions. The recommendation weight is again varied, and the recommendation path length is fixed at one hop.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.50 \\ B_F = 0.0 \\ B_C = 0.50 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 50% of the time, ships a lesser-quality item in no cases, and not ship any item at all 50% of the time.
- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.0 \\ \text{Decreases honesty: } 0.0 \\ \text{Unchanging profile: } 1.0 \end{cases}$$

- No contextual relevance assessment.
- No time fading.
- Recommendation weighting policy, π, where $\pi \in \{0.99, 0.90, 0.75, 0.50, 0.10\}$
- Recommendation path length, l, where $l \in \{1\}$
- False positive recommendations:

$$prob(false\_positive\_rec)\begin{cases} prob(false_{FG}) = 0.0 \\ prob(false_{CG}) = 0.25 \end{cases}$$

Results

The results of this experiment are illustrated in Figure 10, where the unweighted trust value produced by the RMS trust calculation mechanism calculates likelihood of good seller behaviour at approaching

25% more likely than is the case in actual mean seller behaviour. Again, this result is intuitive, because the trust value is calculated according to evidence that is made up of 75% accurate recommendations and 25% false positive recommendations. As in the previous experiment, the recommendation-weighted trust values have the effect of decreasing the calculated likelihood of good behaviour and underestimating trustworthiness so as to ensure risk limitation. For example, when π = 0.99, the calculated likelihood of good seller behaviour limits risk of exposure to unreliable evidence by 1%. The other recommendation policies, when applied to the trust value, result in further reducing risk.

In this scenario, when π = 0.99 and 0.90, the likelihood of good seller behaviour is slightly lower than that produced by the unweighted trust value but still greater than actual mean behaviour. When π = 0.50 and 0.10, the calculated likelihood of behaviour is increasingly lower than actual behaviour, respectively. When the recommendation weighting policy, π = 0.75, is used to weight a trust value, a likelihood of seller behaviour is produced that is more accurate to actual mean seller behaviour. We note a similar result to that of Experiment 4.2.1, i.e., when π = 0.75, trust values are discounted 25%, an amount that is equal to the amount of false positive recommendations used as evidence in this experiment.

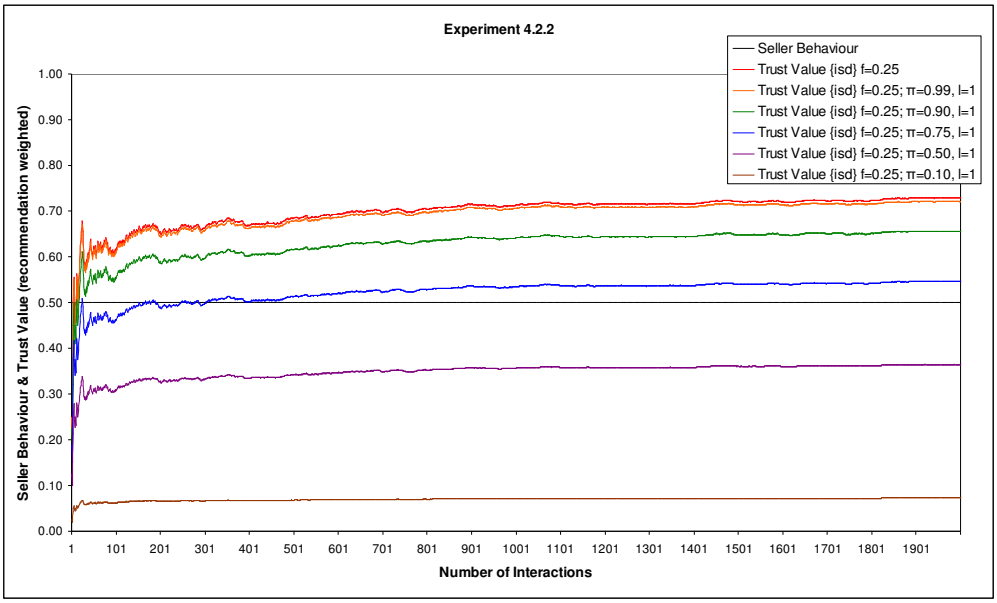

Figure 49: Experiment 4.2.2 25% unreliable (false positive) recommendations

4.2.6.4 Experiment 4.2.3: varying recommendation weighting policy; 50% unreliable recommendations

In this experiment, a seller's behaviour is, again, consistently 50% good, 50% bad. In this scenario, 50% of the buyer's recommendations are false positives. The recommendation weight is again varied, and the recommendation path length is fixed at one hop.

Parameter Specification

- 2000 interactions.
- 1 seller, 1 buyer – each maintains the same role for all interactions.
- Seller's initial behaviour profile:

$$(seller\_behaviour)\begin{cases} B_G = 0.50 \\ B_F = 0.0 \\ B_C = 0.50 \end{cases}$$

  That is, the seller acts randomly over the period of interactions according to: good behaviour 50% of the time, ships a lesser-quality item in no cases, and not ship any item at all 50% of the time.
- Seller's behaviour profile remains constant:

$$(seller\_behaviour\_change)\begin{cases} \text{Increases honesty: } 0.0 \\ \text{Decreases honesty: } 0.0 \\ \text{Unchanging profile: } 1.0 \end{cases}$$

- No contextual relevance assessment.
- No time fading.
- Recommendation weighting policy, π, where $\pi \in \{0.99, 0.90, 0.75, 0.50, 0.10\}$
- Recommendation path length, l, where $l \in \{1\}$
- False positive recommendations:

$$prob(false\_positive\_rec)\begin{cases} prob(false_{FG}) = 0.0 \\ prob(false_{CG}) = 0.50 \end{cases}$$

Results

Figure 11 illustrates the results of this experiment, in which the unweighted trust value produced by the RMS trust calculation mechanism produces a likelihood of good seller behaviour that is approximately 50% higher than actual seller behaviour. As in the previous two experiments, this result is intuitive, because the trust value is calculated according to evidence that is made up of 50% accurate recommendations and 50% false positive recommendations. Again, the recommendation-weighted trust values have the effect of decreasing the calculated likelihood of good behaviour in order to limit risk to unreliable evidence.

In this scenario, when π = 0.99, 0.90, and 0.75, a weighted trust value is produced that calculates the likelihood of good seller behaviour at a lower rate than that produced by the unweighted trust value; however, the weighted trust values in these three cases still calculate likely behaviour that is much greater than actual behaviour. When π = 0.10, the calculated likelihood of behaviour is 40% lower than actual behaviour. When the recommendation weighting policy, π = 0.50, is used to weight a trust value, a likelihood of seller behaviour is produced that converges on actual mean seller behaviour. This further validates the results of the previous two experiments, i.e., an optimal level of risk limitation is provided when a recommendation policy is specified at a weight that approaches the degree of false positive recommendations.

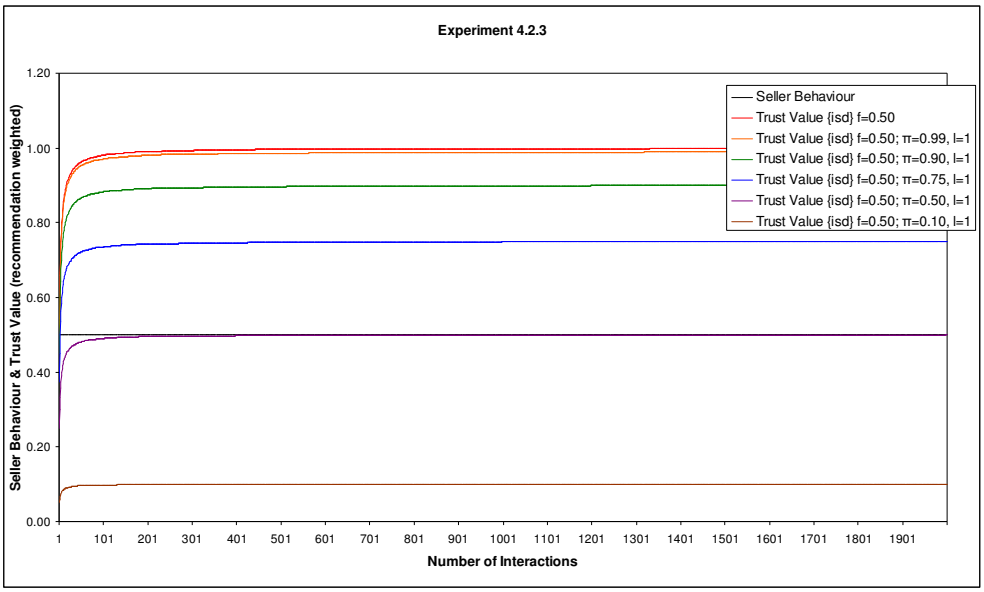

Figure 50: Experiment 4.2.3 50% unreliable (false positive) recommendations

4.2.6.5 Conclusions with regard to RMS trust calculation when recommendation weighting is utilised

Applying recommendation weighting to trust value calculation allows a user to limit his exposure to risk of unreliable recommended evidence based on the stringency of a recommendation weighting policy and how far away in a recommendation path a recommender is. These policies parallel real world human risk strategies, whereby some people are more trusting and others choose to underestimate trustworthiness. The former group interact more often, even if sometimes interaction results in a negative outcome. The latter group assumes less risk but may also be restricting themselves from potentially beneficial interactions.

In a scenario in which there are no false positive recommendations, e.g., Experiment 4.1, weighting trust values according to recommendation policy exposes a user to less risk by underestimating trustworthiness, but this occurs at the cost of accuracy loss. That is, the recommendation-weighted trust value calculates likelihood of behaviour at a rate that is lower than actual behaviour. However, when false recommendations are used as evidence with which to calculate trust, as in Experiments 4.2.1-4.2.3, weighting trust values can result in calculating the likelihood of behaviour at a rate that approaches actual behaviour, thereby increasing accuracy while reducing risk exposure. The results demonstrate that an optimal level of risk limitation is provided when a recommendation policy is specified at a weight that approaches the degree of false positive recommendations. Therefore, if a system is informed as to an expected rate of false positive recommendations within a given application domain, the recommendation weighting policy for use in that domain can be tailored appropriately.

By weighting recommendations according to recommendation integrity policy specification, the trust calculation mechanism can produce a trust value is more accurate with regard to likelihood of behaviour and that exposes a user to less risk in the case where recommendations may not be wholly reliable. Additionally, complexity is reduced and the mechanism is usable, because while any given user will not be able to manually assess the reliability of recommendations, the mechanism is able to do so according to the policy specified.

4.2.7 A Summary of the Simulation Experiments and Results

By simulating Internet auction user interactions and feedback provision with the TNG simulator, we have evaluated the mechanisms of the RMS with regard to evaluation criteria 1-4.

First, the trust value calculation of the RMS was evaluated and its accuracy assessed in producing a trust value that can be used as a basis to calculate the likelihood of domain-specific entity behaviour based on observations of that behaviour. We found that the observation records collected by the RMS feedback mechanism were of a type and granularity required to identify domain-specific behaviour types, such as normal behaviour, fraudulent behaviour, and thieving behaviour, and thus to limit risk of exposure to user interaction with untrustworthy entities. Furthermore, we found that the basic RMS trust calculation mechanism is highly accurate for identifying behaviour types for sellers with consistent behaviour patterns and that the mechanism can calculate initial trustworthiness in the case where no observations have yet been made about a user. The basic RMS trust value becomes less accurate as a basis for calculating likelihood of expected behaviour on a per-interaction basis, but this was corrected by assessing evidence for timeliness, as shown by the time-fading experiment.

Experiments 1.1 and 1.2 also allowed us to conclude that when observations are recorded about domain-specific behaviour, the RMS can identify user behaviour patterns based on the evidence that it would not be able to identify when only general positive or negative feedback statements are collected. Moreover, when observations are recorded about specific behaviour types, the mechanism

analysed evidence to make decisions about the likelihood of occurrence of normal behaviour, fraud, or theft, thus increasing usability and reducing complexity for a user who no longer would have to manually assess feedback records to identify behaviour patterns.

Second, the contextual relevance assessment mechanism of the RMS was evaluated and we assessed the results of using a contextually relevant trust value to calculate the likelihood of domain-specific entity behaviour in a given environmental context based on observations of that behaviour in the context. We found, as is intuitive, that a basic overall trust calculation was not as accurate for behaviour in a given context as was a contextually relevant trust value. A seller can maintain a fairly high overall trust value by behaving well in a context(s) in which he mainly interacts which can mask his untrustworthy behaviour in another context. Our mechanism increases accuracy of trust value calculation for decision-making in such scenarios while automating the complex and unusable task of manually assessing recommendations for contextual relevance.

Third, the time-fading mechanism of the RMS was evaluated and we assessed the results of using a time-faded trust value to calculate the likelihood of domain-specific entity behaviour in a given time period based on observations of behaviour in the time period. We found that the time-faded trust value is more responsive over time to fluctuations in behaviour, i.e., the trust value is weighted towards current behaviour and therefore reacts more quickly to entity behaviour changes rather than converging on mean behaviour as does the basic RMS trust value calculation. Moreover, we found a 5% increase in accuracy from Experiment 1.2 to Experiment 3.2, whereby the time-faded trust value, which was weighted in favour of more recent behaviour, gave a more accurate calculation of the likelihood of *actual* seller behaviour rather than *mean* seller behaviour when a seller acted in an oscillating manner. Again, automating the evidentiary analysis process, in this case with regard to temporal context, results in the provision of usability through complexity reduction.

Fourth, the trust value calculated by the RMS was evaluated to assess the results of the application of a recommendation weighting policy that was specified at various levels of risk aversion. Analogous to real world human risk strategies, applying recommendation weighting to trust value calculation allows a user to limit his exposure to risk of unreliable recommended evidence based on the stringency of a recommendation weighting policy and how far away in a recommendation path a recommender is. The results demonstrate that an optimal level of risk limitation is provided when a recommendation policy is specified at a weight that approaches the degree of false positive recommendations, i.e., if all recommendations are reliable, then weighting will limit risk exposure but also underestimate trustworthiness, whereas if some recommendations are unreliable, weighting can result in a more accurate estimation of behaviour. Therefore, if a system is informed with the expected rate of false positive recommendations within a given application domain, the recommendation weighting policy for use in that domain can be appropriately customised. Additionally, complexity is reduced and the mechanism is usable, because while any given user will not be able to manually assess the reliability of recommendations, the mechanism is able to do so.

In conclusion, our experiments demonstrated that the trust value calculation mechanism of the RMS is amenable to statistical evaluation, under which we identified factors, i.e., behaviour, context, time, and recommendation weighting, that can make the trust value calculation mechanism adjust trust values differently.

4.3 Case Study: Evaluation of the Collusion Detection Mechanism

Evaluation criterion 5 states that the RMS should be evaluated in terms of its ability to collect evidence about and to detect anomalous behaviour with respect to interaction dynamics between groups of users, specifically, collusion for the purpose of artificially increasing the price of an item being auctioned, and thus limit exposure to risk from colluding behaviour. We evaluate our collusion detection mechanism against the backdrop of anomaly detection principles, i.e., we assume that most seller-bidder pairs are honest and their behaviour does not lead to fraudulent price inflation through shilling, and that behaviour that is dishonest, i.e., anomalous, can be detected. A classification of normal behaviour for a sample of Internet auction data is performed, and then seller-bidder pairs are evaluated to assess if any pairs exhibit anomalous behaviour, i.e., shilling interaction dynamics.

Over a three-week period from 08 July 2005 through 31 July 2005, we collected data from 2791 auctions in one item category in which bidding occurred in seven-day auctions of a well-known Internet auction provider. This dataset excludes auctions in which no bidding occurred, i.e., in which there is no evidence with which to analyse interaction dynamics according to our approach. The following information was collected for each auction: auction number, seller username, and bid history for each auction including: bidder username bid amount, bid time, and whether or not the bid was the winning bid. We then identified that 35 of the 137 sellers in the data set controlled 94% of the market, i.e., 2622 auctions. The remaining sellers typically had interacted in only one auction each, i.e., no trend with regard to interaction dynamics could be demonstrated in these cases. Therefore, we selected the auction and bid data of the 35 top sellers as a representative sample of activity.

First, we define our statistical models, $P, B, T,$ and $\Lambda,$ based on the characteristics of questionable bids identified by our RMS design:

1. $P,$ which correlates the number of a seller's auctions in which a bidder interacts to the overall number of a seller's auctions.
2. $B,$ the bid increment pattern of a bidder in a seller's auctions.
3. $T,$ the bid timing pattern of a bidder in a seller's auctions.
4. $\Lambda,$ which correlates a bidder's appearance in a seller's auction to his loss rate (percentage of time a bidder loses out of all of the auctions in which he participates).

For each seller in the sample, we analysed each bidder in each auction to determine a threshold of normalcy. To do this, we first identified the bidder that exhibited the most abnormal behaviour in each auction, i.e., $\frac{\{b\}+\{bi\}+\{be\}+\{bl\}}{\{b\}+\{bi\}+\{be\}+\{bl\}+\{\text{`}b\}+\{b\text{`}i\}+\{b\text{`}e\}+\{b\text{`}l\}}$, or, the number of observations of abnormal behaviour between the bidder and seller in all auctions in the dataset in which the two interacted over the number observations of all behaviour between the bidder and seller in all auctions in the dataset. For each auction of a given seller, we calculated the number of observations of this most abnormal seller-bidder pair according to: the number of observations of {b}, i.e., a seller's auctions in which the bidder bid; the number of observations of {`b}, i.e., a seller's auctions in which the bidder did not bid; the number of observations of {bi}, i.e., a bidder's bid is over a high bidding threshold (when a bid is increased by an amount greater than or equal to 50% of the current price of the item on auction); the number of observations of {b`i}, i.e., a bidder's bid is under a high bidding threshold; the number of observations of {be}, i.e., a bidder's final bid is before a bid timing threshold (set to one day before the end of the auction); the number of observations of {b`e}, i.e., a bidder's final bid is after the bid timing threshold; the number of observations of {bl}, i.e., a bidder loses in the seller's auction; and the number of observations of {b`l}, i.e., the bidder wins in the seller's auction.

Next, we calculated four seller-specific thresholds of normalcy according to:

1. $\rho_{seller} = \frac{\{b\}}{\{b\}+\{\text{`}b\}}$, average repeat bidding in a seller's auctions.
2. $\beta_{seller} = \frac{\{bi\}}{\{bi\}+\{b\text{`}i\}}$, average high bidding in a seller's auctions.
3. $\tau_{seller} = \frac{\{be\}}{\{be\}+\{b\text{`}e\}}$, average early bidding in a seller's auctions.
4. $\lambda_{seller} = \frac{\{bl\}}{\{bl\}+\{b\text{`}l\}}$, average loss rate in a seller's auctions.

We then classified four normal behaviour thresholds for the sample according to:

1. $\rho_{sample} = \frac{\rho_{seller_1} + \rho_{seller_2} + \ldots + \rho_{seller_n}}{n}$
2. $\beta_{sample} = \frac{\beta_{seller_1} + \beta_{seller_2} + \ldots + \beta_{seller_n}}{n}$
3. $\tau_{sample} = \frac{\tau_{seller_1} + \tau_{seller_2} + \ldots + \tau_{seller_n}}{n}$
4. $\lambda_{sample} = \frac{\lambda_{seller_1} + \lambda_{seller_2} + \ldots + \lambda_{seller_n}}{n}$

Finally, we compared each seller-specific threshold to the corresponding sample threshold. If the seller-specific threshold is greater than the sample threshold, we say that the seller is abnormal with respect to the model, i.e.,

If $\rho_{seller} > \rho_{sample}$, then we say that the seller is abnormal with respect to *P*.

If $\beta_{seller} > \beta_{sample}$, then we say that the seller is abnormal with respect to *B*.

If $\tau_{seller} > \tau_{sample}$, then we say that the seller is abnormal with respect to *T*.

If $\lambda_{seller} > \lambda_{sample}$, then we say that the seller is abnormal with respect to *Λ*.

Results

We first calculated the normal behaviour thresholds for the sample, which are as follows:

1. $\rho_{sample} = 0.20$
2. $\beta_{sample} = 0.15$
3. $\tau_{sample} = 0.56$
4. $\lambda_{sample} = 0.61$

That is, for the sample population of sellers and their auctions, the average percentage of repeat interaction in a seller's auction by a bidder is 20%; the average amount of time there was bidding over the high bid increment, i.e., 50% or more of the current price, is 15%; early bidding occurred, on average, 56%; and the average loss rate was 61%.

We then evaluated each model independent of the others, with the following results. 23% of sellers were abnormal with respect to *P*. While this may initially suggest the possibility of collusion between 23% of sellers and their repeat bidders, an alternative explanation may be that a good seller commands repeat business from bidders, especially given that the item category is one in which items are highly collectible. 54% of sellers were abnormal with respect to *B*. An alternative to collusion, again, is possible, e.g., a bidder who wants to win an item may place higher-than-normal bid increments in order to drive away competition. 51% of sellers were abnormal with respect to *T*, which could suggest a colluding bidder making early bids, or that a bidder drops out of an auction for other reasons, e.g., the price gets too high. Finally, 54% of sellers were abnormal with respect to *Λ*, that is, bidders in these sellers' auctions typically lost more than the 61% average loss rate for the sample. Again, an alternative to collusion presents itself, e.g., a bidder may have been unlucky, or there were many auctions in which there was a unique bidder who lost, which is intuitive.

Because each model on its own suggests an alternative explanation to behaviour apart from collusion, we then analysed the models together. 65% of sellers were abnormal with respect to at least two models, and 43% of sellers were abnormal with respect to at least three models.

It is interesting to note that in the 15 cases in which a seller was abnormal with respect to at least three models, 10 sellers were normal with regard to *P,* but abnormal with regard to *B, T,* and *Λ*. This suggests that in each of a seller's auctions, there was a bidder who bid in high increments before the

bid timing threshold and lost, but that this bidder did not repeatedly interact with the seller in a manner that would cause suspicion to be raised. If a seller were trying to keep suspicion of collusion at bay while enjoying the benefits of receiving artificially inflated prices for his auctioned goods, one way of doing this would be to use many colluding bidders, i.e., colluding with a different bidder in each auction. Unfortunately, our mechanism does not protect against such a Sybil attack, although it does highlight this type of trend in behaviour.

Only one seller, S32, was abnormal with respect to all four models. In this seller's auctions, one bidder, B09, appeared in 58% of the auctions; B09 placed high bids 33% of the time, i.e., more than twice the normal high bidding threshold; B09 bid before the bid timing threshold 100% of the time; and B09 lost in 100% of these interactions. Similarly, two other bidders repeatedly bid early and lost in 100% of interactions with S32. Therefore, we feel that there is an extremely strong case for the likelihood of collusion between S32 and B09, as well as a fairly strong case for the likelihood that S32 colludes with more than one other user to artificially inflate prices of auctioned items through shill bidding.

This evaluation demonstrates that the RMS's collusion detection mechanism is able to assess the interaction dynamics that make up the profile of colluding behaviour based on auction-duration observations collected by an Internet auction system. Even though most Internet auction systems typically collect such evidence, without a mechanism for detecting colluding behaviour types, it is infeasible that a user could manually detect colluding profiles. Thus, without the use of a collusion detection mechanism, colluding behaviour can go unnoticed and a seller who appears to be trustworthy may be able to dupe legitimate buyers into paying artificially-inflated prices for items. By incorporating a collusion detection mechanism into the RMS, however, increased awareness with regard to colluding behaviour types can be provided to legitimate users, thus allowing a more accurate decision with regard to interaction to be made. Furthermore, by including a mechanism to automatically assess interaction dynamics, the RMS is further reducing complexity and providing ease-of-use to users.

4.4 Qualitative Analysis

This section provides a qualitative analysis of Evaluation Criteria 6 - 8, i.e., that the RMS should make risk explicit to a decision-maker; that the RMS should provide advice based on trust, risk, context, and interaction dynamics to a decision-maker; and that the RMS should adhere to the characteristics of a trust-based decision-making system that unifies the characteristics and properties of human trust. The analysis of each of these criteria is described as follows.

4.4.1 Making Risk Explicit

Both agent- and context-specific risk can be captured and assessed by the RMS design, as explained in section 3.4.9. The risk component evaluates risk based on several factors, i.e., the trustworthiness of the trust target in the current context as expressed by the trust value and the overall risk of interacting in a given context. By explicitly reasoning about risk, the RMS allows users to specify acceptable levels of risk for interactions with other entities in specific contexts. The RMS makes risk explicit to the decision-maker, both in terms of the risk of financial loss due to untrustworthy behaviour based on the calculation of a trust value that is behaviour-specific, contextually relevant, and based on reliable evidence and in terms of the risk of paying an artificially inflated price due to colluding behaviour between a seller and bidder. By analysing evidence on behalf of users, the RMS reduces complexity because a given user is no longer confronted with an abundance of evidence that he could not manually process and evaluate in terms of risk. Moreover, complexity is reduced and usability ensured by the analysis of risk in terms of trust-based evidence and the presentation of that analysis to a user.

4.4.2 Advice Provision

As described in section 3.5.4, the RMS is designed to provide advice based on trust, risk, context, and interaction dynamics to the decision-maker, guiding the user on whether or not to proceed with an interaction, instead of providing a reputation summary or a trust value to a decision-making user. Therefore, a user of the RMS can be guided by advice based on an automated analysis of evidence to make a more informed, and therefore more accurate, decision about whether or not to proceed with an interaction.

4.4.3 Trust-Based Decision Making

This section discusses the ability of the RMS to meet the trust-based decision-making criteria identified in Chapter 2. First, the specification of confidence in outcome expectations is possible. As described in section 3.4.8, the RMS uses a trust measure, i.e., the (s,i,c)-triple, that is expressed as a value that captures combined evidence for and against a given outcome, as well as capturing uncertainty, and implicitly communicates the amount of evidence, i.e., number of interactions, used to determine trustworthiness. We do, however, note a deficiency in that confidence is not explicitly specified or used by our system in this regard. We suggest that this could be easily resolved through the integration of a confidence assessment policy specification capability in the access control component of the RMS. Clearly, further experiments would need to be performed to determine general confidence thresholds, i.e., the amount of evidence required to shift from low to high confidence in an RMS security decision may be subjective and therefore vary across users.

Next, the RMS was designed to capture the diversity dimensions of trust, i.e., trust origin (individual or entity), trust target (individual or entity), and trust purpose. This is done in part through the design of role- and environment-appropriate event structures, described in section 3.4.4, to provide a flexible and extensible model of the types of events a trust origin would be interested in analysing when considering an interaction with a trust target for a given trust purpose.

Because not all entities have the same perception of evidence, the RMS is designed to allow for the subjective specification of trust formation, evolution, and exploitation processes in terms of context, timeliness, and recommendation integrity, as described throughout sections 3.4 and 3.5. For example, Alice might only be interested in the most current evidence about interactions with Bob in one context, while Carl might choose more lax contextual relevance and time-fading policies to assess historical evidence about Bob. Because subjectivity is also based on an entity's disposition and belief system, our design may easily be extended to allow for policy to be specified such that the *eff* function may adjust trust values according to various trust dynamics. Furthermore, because the RMS provides a security decision in terms of advice, a user can still act according to his own disposition, either heeding or ignoring the advice of the RMS.

Evidence that is based on past behaviour can be directly observed or indirectly recommended and used to update trust both positively and negatively in a dynamic and non-monotonic manner. A single trust value is created from the combination of direct and indirect evidence, i.e., observations and recommendations. Observations and recommendations are evaluated independently so that a trust origin is able to keep its own direct evidence about a trust target separate from the general population's opinion about the trust target, and so that recommendation integrity can be assessed and calculated into the final trust value. Furthermore, the processes to collect and analyse evidence are explicit.

Trust is context-dependent, and the RMS captures context, including asymmetrical relationships (role), time, and environmental factors. As discussed in sections 3.4.7 and 3.4.8, the design provides for evidentiary assessment according to contextual relevance of evidence according to role, time, and environmental factors.

Both agent- and context-specific risk can be captured and assessed, and the RMS makes risk explicit to the decision-maker, as discussed above in this section.

The RMS produces a meaningful and usable measure of trust based on the subjective analysis of contextually relevant evidence, and exploits this measure on behalf of a trust origin in two ways, i.e., the measure may be propagated to the community via recommendations (see section 3.4.6), and the measure may be used to making trusting decisions to interact for a given trust purpose with a given trust target in light of associated risk, as detailed in section 3.4.8.

Finally, the RMS significantly reduces the complexity of decision-making in an environment in which uncertainty and risk are present. The RMS is able to automatically process large amounts of evidence

that a human user could not manually process in order to make a timely decision about whether or not to interact. Furthermore, as noted earlier in this section, instead of providing a reputation summary or a trust value to a decision-making user, the RMS outputs actual advice to a user about how best to proceed in an interaction, given trust, context, interaction dynamics, and risk.

Our model helps clear conceptual confusion by representing trust as a broad but coherent set of constructs that incorporate the major definitions from research to date. It draws together the many aspects of trust to make trusting decisions as a function of available evidence, subjective beliefs, and context in the face of risk and uncertainty. This design captures the characteristics and properties of trust that were suggested in Chapter 2, and that the RMS provides a trust-based decision-making process to advise users regarding trusting behaviour. In conclusion, the RMS allows for trust to be made computationally tractable in the domain of reputation management for Internet auctions while retaining a reasonable connection with human and social notions of trust that guide human users to make decisions in that domain.

4.5 Chapter Summary

In this chapter, we evaluated the RMS according to an extensive evaluation plan addressing eight evaluation criteria. After describing the evaluation plan, we evaluated the RMS design in three parts. First, simulations of extra-auction user behaviour allowed us to evaluate our trust value calculation mechanism and its extensions to assess contextual relevance, timeliness, and recommendation weight. Moreover, the results of the simulations assisted in our demonstration that the RMS mechanisms reduce complexity, increase accuracy, and maintain usability in the assessment of evidence to determine trustworthiness and therefore limit exposure to the risk of interacting with untrustworthy entities. Next, we ran our collusion detection algorithms over real Internet auction data which allowed us to demonstrate the ability of the RMS to suspect colluding behaviour types based on the analysis of interaction dynamics. In fact, a highly probable instance of collusion was detected using this mechanism. Finally, we provided a qualitative analysis of the design of the RMS in terms of the trust characteristics and properties identified in Chapter 2, which allowed us to demonstrate that the RMS adheres to the characteristics of a trust-based decision-making system.

Chapter 5: Conclusions and Future Work

There's another way to survive. Mutual trust - and help.
James T. Kirk

This thesis presented the RMS reputation management system, a trust-based decision-making system designed to support users interacting in virtual marketplaces, in particular, Internet auctions. This chapter reviews the most significant achievements of our work, as described in this thesis, and outlines our contributions. We then conclude with a discussion of related research issues that remain open for future work.

5.1 Contributions

The work in this thesis addressed issues arising in existing systems for reputation management in consumer-to-consumer (C2C) virtual marketplace applications. In Chapter 1, we noted that a virtual marketplace such as an Internet auction approximates its traditional predecessors as a system in which the human notions of trust and risk in decision-making are critical to continued performance in the face of domain-specific threats; and that, in order to help users make decisions about interacting with others, most virtual marketplaces incorporate some form of reputation management to ensure trustworthy interactions and decreased risk in this domain. We also identified several core issues with reputation management systems that motivated the work in this thesis. First, commercial reputation management systems typically promote usability over accurate evidentiary analysis, resulting in the lack of collection of data which could better assist in identifying domain-specific behaviour such as fraud and theft. Second, these systems ignore valuable context information in terms of user role, timeliness of evidence, and environmental context. Third, the dynamics of user interactions are not detectable, leading to the fourth and fifth issues, namely that it is nearly impossible for a human user to manually assess whether or not a user provided useful and accurate recommendations about another user or whether or not a group of users are colluding with malicious intent. Sixth, risk is not explicitly calculated by such reputation management systems, and may not be assessed by the user at all. Seventh, a reputation is often no more than an overall summary of a collection of thousands of individual recommendations rather than an explicit portrayal of the trust and risk involved in a context-specific interaction.

In order to design decision-making tools, such as a reputation management system for an Internet auction, we believe that it is essential to understand the key features of the human decision-making

processes of analysing trustworthiness, assessing risk, and evaluating social features of interaction groups. It is also necessary to understand what kind of evidence these processes require, how to collect that evidence, and how to determine evidentiary relevance for the types of decisions that need to be made.

Therefore, in Chapter 2, we described the results of our examination of current research in modelling and adapting traditional human decision-making processes based on trust and reputation for online interaction environments. From this examination, we concluded that the diversity of trust definitions leads to difficulty when trying to develop a general definition of trust, but that it is possible to extract from the definitions several trust attributes and properties with which to create a unified model of human trust. Using these trust characteristics to evaluate research into the formulation of computational trust models, we discovered that only the Secure Environments for Collaboration among Ubiquitous Roaming Entities (SECURE) trust model captured all of the identified trust attributes and properties. The SECURE model unifies all of the key trust characteristics and processes into one complete framework that had been implemented and validated in a real-world system such that computational trust could be used for decision-making in a manner analogous to the human process, and we decided to use this model as the basis for the design of a trust-based reputation management system.

In Chapter 3, we addressed the issues identified as outstanding in the reputation management systems currently used to support decision-making in Internet auction applications and presented a design for a reputation management system based on extending the SECURE decision-making framework. In the first several sections of this chapter, we described the SECURE approach, including its trust, collaboration, and risk models, framework components, decision-making processes, and implementation. This description also covered a discussion of the general threats against which SECURE was designed to protect. We discussed the deployment of SECURE in the spam-filtering domain, which encompasses a description of how the SECURE trust, collaboration, and risk models were applied in this domain, as well as a brief examination of the implementation and evaluation of the SECURE spam filtering application.

Having described the SECURE approach and its application, we then put forward a general description of reputation management in virtual marketplaces application domain, which includes our development of an application-specific taxonomy of behaviour that classified both normal and malicious types of behaviour in this domain. Then, the design of a trust-based reputation management system, called the RMS, based on the SECURE trust- and risk-based decision-making framework was proposed. An overview of the design was given, and our rationale for our design decisions was presented with regard to requests, entity recognition, trust and evidence processes, risk assessment, and access control. Additionally, our rationale for extending the SECURE framework to include interaction management such that decision-making might be enhanced through the analysis of trustworthy recommendation paths between users and observations about potential colluding behaviour is described. We illustrated how these two new interaction management components may be integrated into the SECURE framework, and we detailed the enhanced decision-making process.

The proposed design of the RMS addresses existing issues with regard to reputation management in Internet auctions in terms of reducing complexity, increasing accuracy, and maintaining usability. Specifically, we tailored and extended the SECURE trust, evidence, and risk mechanisms to the reputation management for Internet auctions application domain and our contributions in this regard are described as follows.

The RMS is designed in such a way as to increase accuracy in decision-making for users of Internet auctions while maintaining usability. Based on our application-specific taxonomy of normal and anomalous user behaviour types, we designed SECURE event structures and configurations to model the event types needed to predict the likelihood of role-specific user behaviour in future interactions. We then adapted the event structures and the associated evaluation methods of the trust model to allow for the assessment of extra-auction evidence gathered in the form of observations and recommendations at a level of granularity that permitted both usability and increased precision in decision-making. Moreover, we incorporated a contextual relevance assessment mechanism to analyse the context of evidence in terms of role, time, and environment, such that accuracy of evidentiary analysis might be further increased. Furthermore, we designed components with which to analyse interaction dynamics, i.e., recommendation weighting based on path analysis and collusion detection based on auction-duration events, to enhance the accuracy of the decision-making process. Automating the analysis of evidence using the RMS allowed for increased accuracy as well reduced complexity in decision making, while promoting usability, because more evidence may be assessed than a human user would be capable of manually processing and users no longer must be confronted with the complexity of such abundant evidence.

Moreover, we extended the SECURE risk model to include methods with which to analyse user-, context-, and interaction dynamic-specific risk. We designed the risk assessment methods in such a way as to expose risk in financial terms, according to traditional security risk assessment techniques that users comprehend. Risk is made explicit in a statement of potential financial loss that is output to the end user along with advice to guide a user in making a decision about entering into an interaction based on the RMS's automated evaluation of contextually relevant evidence in terms of trust, risk, and interaction dynamics. This advice is therefore more useful to and usable by a user than the current reputation summary information currently provided by commercial reputation management systems.

Having proposed the design of the RMS, we then evaluated its mechanisms in three parts in Chapter 4. First, we used simulations of extra-auction user behaviour to evaluate the RMS's trust value calculation mechanism and its extensions to assess contextual relevance, timeliness, and recommendation weight. Next, running our collusion detection algorithms over real Internet auction data allowed us to demonstrate the ability of the RMS to assess colluding behaviour types based on interaction dynamics. Finally, we provided a qualitative analysis of our system in terms of the trust characteristics and properties identified in Chapter 2.

With regard to the simulation experiments, the trust value calculation mechanism of the RMS was evaluated to assess its accuracy in producing a trust value that could be used to calculate the

likelihood of domain-specific entity behaviour based on observations of that behaviour. First, we found that the basic RMS trust calculation mechanism is highly accurate for identifying key behaviour types for users with consistent behaviour patterns , i.e., the trust value converges on mean behaviour, and that the mechanism can calculate initial trustworthiness in the case where no observations have yet been made about a user. Second, when we incorporated the contextual relevance assessment mechanism, accuracy is increased for decision-making in scenarios in which entities behave differently in different contexts, i.e., the mechanism is a useful control against reputation masking threats. Third, we found that accuracy is increased 5% by incorporating the time-fading method in a scenario when a seller acts in an unpredictable oscillating manner. In that scenario, when the time-faded trust value is weighted in favour of most recent behaviour it gives a more accurate calculation of the likelihood of *actual* seller behaviour rather than *mean* seller behaviour, i.e., it follows the pattern of actual behaviour more closely than in the case in which time fading is not utilised. Fourth, the results of evaluating the recommendation weighting method demonstrate that an optimal level of risk limitation is provided when a recommendation policy is specified at a weight that approaches the degree of false positive recommendations, i.e., if all recommendations are reliable, then weighting will limit risk exposure but also underestimate trustworthiness, whereas if some recommendations are unreliable, weighting can result in a more accurate assessment of actual behaviour. Overall, these experiments demonstrate that automating the evidentiary analysis process results in the provision of usability through complexity reduction for users. Finally, the simulations confirm that the mechanisms are amenable to statistical evaluation, under which we identified parameters, i.e., behaviour, context, time, and recommendation weighting, that could make the system adjust trust values differently when adjusted.

Next, we evaluated the RMS collusion detection mechanism to demonstrate its ability to assess the interaction dynamics that make up the profile of colluding behaviour based on auction-duration observations collected by an Internet auction system. Even though most Internet auction systems typically collect such evidence, without a mechanism for assessing colluding behaviour types, it is infeasible that a user could manually detect colluding profiles, which can lead to colluding behaviour going unnoticed and result in legitimate buyers paying artificially-inflated prices for items. By incorporating a collusion detection mechanism into the RMS, such behaviour profiles can be suspected, thus allowing users to make more accurate decision with regard to interaction. Furthermore, by including a mechanism to automatically assess interaction dynamics, the RMS further reduced complexity and provided usability.

Finally, we evaluated our model qualitatively. We found that both entity- and context-specific risk can be captured and assessed by the mechanisms designed, and that the RMS design stipulates the provision of advice based on trust, risk, context, and interaction dynamics to the decision-maker instead of providing a reputation summary or a trust value to a decision-making user. Therefore, a user of the RMS can be guided by advice based on an automated analysis of evidence to make a more informed, and therefore more accurate, decision about whether or not to proceed with an interaction. Furthermore, we found that the design of the RMS captures the characteristics and properties of trust

that were suggested in Chapter 2, and that the RMS provides a trust-based decision-making process to advise users regarding trusting behaviour. Our model helps clear conceptual confusion by representing trust as a broad but coherent set of constructs that incorporate the major definitions from research into the human notion of trust. It draws together the many aspects of trust to make trusting decisions as a function of available evidence, subjective beliefs, and context in the face of risk and uncertainty.

In conclusion, we found that the RMS allows for trust-based decision making to be made computationally tractable in the domain of reputation management for Internet auctions while retaining a reasonable correlation with the human and social notions of trust that guide human users to make decisions in that domain.

5.2 Open Research Issues

The trust-based reputation management system for Internet auctions presented in this thesis supports many of the ideas put forward in the computational trust-based decision-making research. Having undertaken the design and evaluation of the RMS, however, we propose that there are some areas open for possible future work, primarily including issues of usability testing, trust mapping between contexts, and deployment of the RMS in other application domains.

The main area for future exploration involves the issue of deploying the RMS design in a real-world implementation such that user interaction with the RMS might be studied. In this regard, usability studies with human users could be beneficial such that our reputation management system can be comparatively assessed side by side with existing commercial reputation management systems.

Next, we believe that it may be possible to extend the current work on trust-based security measures such that the notion of context is further integrated into decision-making models. We foresee the development of an ontology for mapping trust information across e-commerce contexts, such that relevant information from one context could be appropriately mapped to another context for use in decision-making. For example, in our own work, it may be possible to form a trust value about an entity's trustworthiness in one context, e.g., an Internet auction item category such as low-end electronics, and to map this trust value to make a decision about an entity's likely behaviour in another context, e.g., an Internet auction item category such as kitchen appliances. In this regard, it may also be possible to extend trust-based security models to allow for such context-mapping and to specify policy for users interacting in multiple contexts within the e-commerce domain.

Finally, we believe that our work may have application in domains other than that of Internet auctions, i.e., other data-intensive environments in which automated trust-based decision making for reputation management could be beneficial, that is, domains in which an abundance of evidence is provided to end users for the purpose of decision-making. Instinctively, we foresee an application of our work in other e-commerce domains, for example, virtual marketplaces that specialise in the direct sale, rather

than the auctioning, of goods and services. However, it is possible that other applications of our work may be found, e.g., trust-based reputation management may be applicable in the areas of computer forensics, sensor fusion, or financial services.

5.3 Conclusion

This chapter reviewed the motivations for and the most significant achievements of the work presented in this thesis. In particular, it outlined how our work contributes to the state of the art in reputation management in the C2C domain by providing trust-based decision-making capability that reduces complexity, increases accuracy, and maintains usability for Internet auction users. Finally, the chapter was concluded with some suggestions for future work arising from our research.

Bibliography

Abdul-Rahman, A. (1996). The PGP Trust Model, http://www.cs.ucl.ac.uk/staff/F.AbdulRahman/docs/pgptrust.html.

Abdul-Rahman, A. and Hailes, S. (1997). A Distributed Trust Model. The New Security Paradigms Workshop, ACM.

Abdul-Rahman, A. and Hailes, S. (1999). Relying on Trust to Find Reliable Information. International Symposium on Database, Web and Cooperative Systems (DWACOS'99), Baden-Baden, Germany.

Abdul-Rahman, A. and Hailes, S. (2000). Supporting Trust in Virtual Communities. The 33rd Hawaii International Conference on System Sciences, Maui, Hawaii.

Amazon.com (2006). Amazon.com Website, http://www.amazon.com.

Anderson, K. B. (2005). Internet Auction Fraud: What We Can Learn from Consumer Sentinel Data, Bureau of Economics, Federal Trade Commission.

Axelrod, R. (1984). The Evolution of Cooperation. New York, Basic Books.

Bacon, J., Dimmock, N., Cvrcek, D., Ingram, D. and Moody, K. (2005). Definition of trust-based access control model. SECURE Deliverable (IST-2001-32486).

Barber, S. (1983). Logic and Limits of Trust. New Jersey, Rutgers University Press.

Beth, T., Borcherding, M. and Klein, B. (1994). Valuation of Trust in Open Networks. ESORICS, Brighton, U.K.

Blaze, M., Feigenbaum, J., Ioannidis, J. and Keromytis, A. D. (1999). The KeyNote Trust-Management System, version 2. RFC, IETF.

Blaze, M., Feigenbaum, J. and Lacy, J. (1996). Decentralized Trust Management. IEEE Symposium on Security and Privacy, IEEE.

Blaze, M., Feigenbaum, J. and Strauss, M. (1998). "Compliance Checking in the PolicyMaker Trust Management System." Financial Cryptography: 254-274.

Bryce, C., Cahill, V., Dimmock, N., Kruckow, K., Seigneur, J.-M. and Wagealla, W. (2005). Final Validation Report. SECURE Deliverable (IST-2001-32486).

Bryce, C., Seigneur, J.-M. and Cahill, V. (2005). A Case Study Implementation of a Trust Engine. The 3rd International Conference on Trust Mangement (iTrust05), Rocquencourt, France.

Buchegger, S. and Boudec, J.-Y. L. (2002). Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes --- Fairness In Dynamic Ad-hoc NeTworks. The IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC), Lausanne, IEEE.

Buchegger, S. and Le Boudec, J.-Y. (2004). A Robust Reputation System for P2P and Mobile Ad-hoc Networks. The Second Workshop on the Economics of Peer-to-Peer Systems.

Burrows, M., Abadi, M. and Needham, R. M. (1989). A Logic of Authentication. Symposium on Operating Systems Principles (SOSP), ACM.

Cabral, L. and Hortacsu, A. (2004). The Dynamics of Seller Reputation: Theory and Evidence from eBay, National Bureau of Economic Research, Inc.

Cahill, V., Nielsen, M., Nixon, P., Bacon, J., Jensen, C. D. and Bryce, C. (2001). Annex 1 - Description of Work, SECURE. Information Society Technologies (IST) Programme Contract. Dublin, Ireland, Trinity College Dublin.

Cahill, V., Shand, B., Gray, E., Bryce, C., Dimmock, N., Twigg, A., Bacon, J., English, C., Wagealla, W., Terzis, S., Nixon, P., Serugendo, G. D. M., Seigneur, J.-M., Carbone, M., Krukow, K., Jensen, C. D., Yong, C. and Nielsen, M. (2003). "Using Trust for Secure Collaboration in Uncertain Environments." IEEE Pervasive Computing **2**(3): 52-61.

Carbone, M., Nielsen, M. and Sassone, V. (2003). A Formal Model for Trust in Dynamic Networks. The IEEE International Conference on Software Engineering and Formal Methods (SEFM '03), Brisbane, Australia, IEEE Computer Society.

Christianson, B. and Harbison, W. S. (1996). Why Isn't Trust Transitive? Security Protocols Workshop.

Chu, Y.-H., Feigenbaum, J., LaMacchia, B., Resnick, P. and Strauss, M. (1997). "REFEREE: Trust Management for Web Applications." Computer Networks and ISDN Systems **29**: 953-964.

CNN Money (2005). eBay profits miss the mark. CNN Money. New York.

Coleman, J. S. and Lal, D. (1990). Foundations of Social Theory, Harvard: Belknap Press.

CyberSource Corporation (2005). CyberSource: Annual eCommerce Fraud Survey Results. Yahoo! News.

Dasgupta, P. (2000). Trust as a Commodity. Trust: Making and Breaking Cooperative Relations. D. Gambetta. Oxford, Dept. of Sociology, University of Oxford.

Department of Defense (1985). Department of Defense Trusted Computer System Evaluation Criteria. DOD 5200.28STD (The Orange Book). Washington, D.C., Department of Defense.

Deutsch, M. (1962). Cooperation and Trust: Some Theoretical Notes. Nebraska Symposium on Motivation, Nebraska University Press.

Dey, A. (2001). "Understanding and Using Context." Personal and Ubiquitous Computing **5**(1): 4-7.

Dijkstra, E. W. (1959). "A note on two problems in connection with graphs." Numerische Mathematik **1**: 83–89.

Dimmock, N., Bacon, J., Ingram, D. and Moody, K. (2005). Risk models for trust-based access control (TBAC). The 3rd Annual Conference on Trust Management (iTrust 2005), Springer-Verlag.

Dulay, N., Lupu, E., Sloman, M., Bacon, J., Moody, K. and Ingram, D. (2005). CareGrid: Autonomous Trust Domains for Healthcare Applications, http://www.doc.ic.ac.uk/~nd/projects/CareGrid.html.

eBay Inc. (2005). eBay Outlines Global Business Strategy at 2005 Analyst Conference. eBay News.

eBay Inc. (2006). eBay Website, http://www.ebay.com.

Elofson, G. (1998). Developing Trust with Intelligent Agents: An Exploratory Study. The 1st International Workshop on Trust.

Federal Bureau of Investigation (FBI) and National White Collar Crime Center (NW3C) (2006). IC3 2004 Internet Fraud - Crime Report, National White Collar Crime Center.

Forbes.com (2005). EBay's 'Theoretical Fair Value' Seen At $105 Forbes.com.

Forrester Research (2005). U.S. Online Retail Sales To Reach $329 Billion By 2010, Says Forrester Research. Tekrati: The Industry Analyst Reporter.

Fraud.org (2006). National Internet Fraud Watch Information Center, http://www.fraud.org/.

Gambetta, D. (2000). Can We Trust Trust? Trust: Making and Breaking Cooperative Relations. D. Gambetta. Oxford, Dept. of Sociology, University of Oxford**:** 213-237.

Golembiewski, R. T. and McConkie, M. (1975). The Centrality of Interpersonal Trust in Group Processes. Theories of Group Processes, Wiley.

Gong, L., Needham, R. and Yahalom, R. (1990). Reasoning About Belief in Cryptographic Protocols. The IEEE Symposium on Research in Security and Privacy, IEEE Computer Society.

Grandison, T. and Sloman, M. (2000). "A Survey of Trust in Internet Applications." IEEE Communications Surveys & Tutorials **3**(4th Quarter).

Grandison, T. and Sloman, M. (2003). Trust Management Tools for Internet Applications. The 1st International Conference on Trust Management (iTrust), Crete, Springer-Verlag.

Gray, E., O'Connell, P., Jensen, C., Weber, S., Seigneur, J.-M. and Yong, C. (2005). Trust Evolution Policies for Security in Collaborative Ad Hoc Applications. First International Workshop on Security and Trust Management, Milan, European Research Consortium in Informatics and Mathematics.

Gray, E., Seigneur, J.-M., Yong, C. and Jensen, C. (2003). Trust Propagation in Small Worlds. The 1st International Conference on Trust Management (iTrust), Crete, Springer-Verlag.

Herzberg, A., Mass, Y., Mihaeli, J., Naor, D. and Ravid, Y. (2000). Access Control Meets Public Key Infrastructure, or: Assigning Roles to Strangers. IEEE Symposium on Security and Privacy, IEEE Computer Society Press.

Ingram, D., Dimmock, N., Bacon, J., Bryce, C., Seigneur, J.-M., Maddison, I., Cvrcek, D. and Moody, K. (2005). Definition of Security Policy Evaluation Model. SECURE Deliverable (IST-2001-32486).

Jonker, C. M. and Treur, J. (1999). Formal Analysis of Models for the Dynamics of Trust Based on Experiences. The 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World : Multi-Agent System Engineering ({MAAMAW}-99), Berlin, Springer-Verlag.

Jøsang, A. (1996). The Right Type of Trust for Distributed Systems. The New Security Paradigms Workshop, ACM.

Jøsang, A. (1999). An Algebra for Assessing Trust in Certification Chains. The Network and Distributed Systems Security Symposium, The Internet Society.

Jøsang, A. (2001). "A Logic for Uncertain Probabilities." International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems **9**(3): 279-311.

Jøsang, A., Gray, E. and Kinateder, M. (2006). "Simplification and Analysis of Transitive Trust Networks (to appear)." Web Intelligence and Agent Systems Journal.

Jøsang, A., Hird, S. and Faccer, E. (2003). Simulating the Effect of Reputation Systems on e-Markets. The 1st International Conference on Trust Management (iTrust), Crete, Springer.

Jøsang, A. and Ismail, R. (2002). The Beta Reputation System. The 15th Bled Conference on Electronic Commerce, Bled, Slovenia.

Jøsang, A., Ismail, R. and Boyd, C. (2006). "A Survey of Trust and Reputation Systems for Online Service Provision (to appear)." Decision Support Systems.

Jøsang, A., Keser, C. and Dimitrakos, T. (2005). Can We Manage Trust? The 3rd International Conference on Trust Management (iTrust 2005), Springer-Verlag.

Jøsang, A. and Pope, S. (2005). Semantic Constraints for Trust Transitivity. The 2nd Asia-Pacific Conference on Conceptual Modelling (APCCM2005), Newcastle, Australia.

Kauffman, R. J. and Wood, C. A. (2003). Running up the bid: detecting, predicting, and preventing reserve price shilling in online auctions. The 5th International Conference on Electronic Commerce, Pittsburgh, Pennsylvania, ACM International Conference Proceeding Series.

Keenan, V. (2000). B2X emerges as new industry exchange transactions. San Francisco, Keenan Vision Inc.

Kinateder, M. and Pearson, S. (2003). A Privacy-Enhanced Peer-to-Peer Reputation System. The 4th International Conference on Electronic Commerce and Web Technologies (EC-Web 2003), Prague, Springer-Verlag.

Kohlas, R. and Maurer, U. (2000). Confidence Valuation in a Public-Key Infrastructure Based on Uncertain Evidence. The International Workshop on Theory and Practice of Public-Key Cryptography, Springer.

Lacy, S. (2004). Dangerous Days on the World Wild Web. Business Week.

Lamsal, P. (2001). Understanding Trust and Security, http://www.cs.helsinki.fi/u/lamsal/papers/UnderstandingTrustAndSecurity.pdf.

Latora, V. and Marchiori, M. (2001). "Efficient Behaviour of Small-World Networks." Physical Review Letters **87**(19).

Latora, V. and Marchiori, M. (2003). "Economic small-world behavior in weighted networks." European Physics Journal B **32**: 249-263.

Luhman, N. (1979). Trust and Power, Wiley.

Manchala, D. W. (1998). Trust Metrics, Models and Protocols for Electronic Commerce Transactions. The 18th International Conference on Distributed Computing Systems.

Marsh, S. (1994). Formalising Trust as a Computational Concept. Dept. of Computer Science and Mathematics, University of Stirling.

Maurer, U. (1996). Modelling a Public-Key Infrastructure. European Symposium on Research in Computer Security (ESORICS), Springer-Verlag.

McFadzean, D. and Tesfatsion, L. (1999). "A C++ Platform for the Evolution of Trade Networks." Computational Economics **14**: 109-134.

McKnight, D. H. and Chervany, N. L. (1996). The Meanings of Trust, Management Information Systems Research Center, University of Minnesota.

Merriam Webster Dictionary (2006). Merriam Webster Dictionary.

Mui, L. (2003). Computational Models of Trust and Reputation: Agents, Evolutionary Games, and Social Networks. Dept. of Electrical Engineering and Computer Science. Boston, Massachusetts Institute of Technology. **Ph.D.**

Mui, L., Mohtashemi, M., Ang, C., Szolovits, P. and Halberstadt, A. (2001). Ratings in Distributed Systems: A Bayesian Approach. Workshop on Information Technologies and Systems (WITS'2001).

Mui, L., Mohtashemi, M. and Halberstadt, A. (2002). A Computational Model of Trust and Reputation. The 35th Hawaii International Conference on System Science (HICSS).

Nielsen, M., Carbone, M., Kruckow, K. and Dimmock, N. (2004). Revised computational trust model. SECURE Deliverable (IST-2001-32486).

Nielsen, M., Plotkin, G. and Winskel, G. (1981). "Petri nets, event structures and domains." Theoretical Computer Science **13**: 85-108.

Oxford English Dictionary (2006). Oxford English Dictionary.

Perez, J. C. (2005). Gartner: Security concerns to stunt e-commerce growth. IDG News Service.

Povey, D. (1999). Developing Electronic Trust Policies Using a Risk Management Model. CQRE Secure Congress.

Rangan, P. V. (1992). "An Axiomatic Theory of Trust in Secure Communication Protocols." Computers & Security **11**: 163-172.

Reagle, J. M. (1996). Trust in a Cryptographic Economy and Digital Security Deposits: Protocols and policies. Boston, Massachusetts Institute of Technology. **Master of Science in Technology and Policy**.

Resnick, P. and Zeckhauser, R. (2002). Trust Among Strangers in Internet Transactions: Empirical Analysis of eBay's Reputation System. Advances in Applied Microeconomics. M. R. Baye. Amsterdam, Elsevier Science. **11**.

Resnick, P., Zeckhauser, R., Friedman, E. and Kuwabara, K. (2000). "Reputation Systems." Communications of the ACM **43**(12): 45-48.

Reuters (2004). Tiffany sues eBay in counterfeit items suit. cnet News.com.

Rosenthal, R. (2002). 2002 Competitive Assesment of eRFx Solution Providers, IDC.

Rubin, S., Christodorescu, M., Ganapathy, V., Giffin, J. T., Kruger, L., Wang, H. and Kidd, N. (2005). An Auctioning Reputation System Based on Anomaly Detection. The 12th ACM conference on Computer and communications security, Alexandria, VA, ACM Press.

Sabater, J. (2004). "Evaluating The Regret System." Applied Artificial Intelligence **18**(9-10): 797-813.

Sabater, J. and Sierra, C. (2001). REGRET: reputation in gregarious societies. Agents, Montreal.

Sabater, J. and Sierra, C. (2002). "Social ReGreT, a Reputation Model Based on Social Relations." ACM SIGecom Exchanges **3**(1): 44-56.

Seigneur, J.-M., Bryce, C., Cahill, V., Yong, C., Dimmock, N., Gray, E. and Jensen, C. D. (2005). SECURE Framework Validation. SECURE Deliverable (IST-2001-32486).

Serafian, D. (2003). Precious Metal. Crime Scene Investigation (CSI). USA, CBS Worldwide Inc.

Shafer, G. (1976). A Mathematical Theory of Evidence. Princeton, NJ, Princeton Univ. Press.

Shafer, G. (1990). "Perspectives on the Theory and Practice of Belief Functions." International Journal of Approximate Reasoning **3**: 1-40.

Shah, H., Joshi, N. R., Sureka, A. and Wurman, P. R. (2003). Mining eBay: Bidding Strategies and Shill Detection. The 4th International Workshop on Mining Web Data for Discovering Usage Patterns and Profiles, Edmonton, Alberta, Springer-Verlag.

Shklar, J. N. (1984). Ordinary Vices, Harvard: The Belknap Press.

Shopzilla (2006). Shopzilla Website, http://www.shopzilla.com/.

Steiner, I. (2004). eBay Users Buy $24 Billion in Goods in 2003, Company Raises Guidance. AuctionBytes.com.

Terzis, S., English, C., Wagealla, W. and Nixon, P. (2005a). Definition of trust evolution model. SECURE Deliverable (IST-2001-32486).

Terzis, S., English, C., Wagealla, W. and Nixon, P. (2005b). Definition of trust exploitation model. SECURE Deliverable (IST-2001-32486).

Terzis, S., English, C., Wagealla, W. and Nixon, P. (2005c). Definition of trust formation model. SECURE Deliverable (IST-2001-32486).

Walker, A. (2003). eBay: Bidding for success. BBC News.

Watts, D. J. (1999). Small Worlds, The Dynamics of Networks Between Order and Randomness. Princeton, New Jersey, Princeton University Press.

Watts, D. J., Dodds, P. S. and Newman, M. E. J. (2002). "Identity and Search in Social Networks." Science **296**: 1302-1305.

Whitby, A., Jøsang, A. and Indulska, J. (2005). "Filtering Out Unfair Ratings in Bayesian Reputation Systems." The Icfain Journal of Management Research **4**(2): 48-64.

Whitby, A., Jøsang, A. and Indulska., J. (2004). Filtering Out Unfair Ratings in Bayesian Reputation Systems. The Workshop on Trust in Agent Societies, at the 3rd International Joint Conference on Autonomous Agents & Multi Agent Systems (AAMAS2004), New York.

Wikipedia (2006). Utility Theory, Wikimedia Foundation, Inc.

Winskel, G. and Nielsen, M. (1995). "Models for concurrency." Handbook of Logic in Computer Science **4**: 1-148.

Wurman, P. R. (2004). Online Auction Site Management. The Internet Encyclopedia, Wiley. **2:** 709-719.

Xiong, L. and Liu, L. (2003). A reputation-based trust model for peer-to-peer ecommerce communities. The IEEE Conference on ECommerce (CEC'03).

Yahalom, R., Klein, B. and Beth, T. (1993). Trust Relationships in Secure Systems - A Distributed Authentication Perspective. The IEEE Computer Society Symposium on Research in Security and Privacy, IEEE Computer Society.

Yahalom, R., Klein, B. and Beth, T. (1994). "Trust-Based Navigation in Distributed Systems." Computing Systems **7**(1): 45-73.

Yahoo! (2006). Yahoo! Shopping Auctions, http://auctions.shopping.yahoo.com/.

Yu, B. and Singh, M. (2002). "Distributed Reputation Management for Electronic Commerce." Computational Intelligence **18**(4): 535-549.

Yu, B. and Singh, M. P. (2000). A Social Mechanism of Reputation Management in Electronic Communities. Cooperative Information Agents IV, The Future of Information Agents in Cyberspace, 4th International Workshop, Boston, Springer.

Zadeh, L. A. (1965). "Fuzzy Sets." Information Control **8**: 338-353

Zadeh, L. A. (1976). "A fuzzy-algorithmic approach to the definition of complex or imprecise concepts." International Journal of Man-Machine Studies **8**: 249-291.

Zadeh, L. A. (1986). Is Probability Theory Sufficient for Dealing with Uncertainty in AI? A Negative View. Amsterdam, Elsevier Science Publishers.

Zimmerman, P. (1995). The Official PGP User's Guide, MIT Press.