Risk Models for Trust-Based Access Control (TBAC)

Nathan Dimmock*, Jean Bacon, David Ingram, and Ken Moody

University of Cambridge Computer Laboratory
JJ Thomson Ave, Cambridge CB3 0FD, UK
Firstname.Lastname@cl.cam.ac.uk

Abstract. The importance of risk in trust-based systems is well established. This paper presents a novel model of risk and decision-making based on economic theory. Use of the model is illustrated by way of a collaborative spam detection application.

1 Introduction and Background

Autonomous decision-making is an increasingly popular application of Trust Management systems. This is particularly true for security and access control in emerging fields such as pervasive and autonomic computing, where existing techniques are seen as inadequate. However, *trust* in itself is an abstract concept and, in order to take a decision, how much trust is required must be balanced against other factors. These are normally quantified as the *risk* of the various courses of action available to a decision-maker.

Proposition 1. *Trust is unnecessary unless there is something at risk.*

1.1 What is Risk?

Previously the SECURE project [1] has explored risk [2] as being a combination of two components, *likelihood* and *impact*. Economists take an alternative approach, defining it as a special type of *uncertainty* [3].

Risk applies to situations when one is unsure of the outcome, but the odds are known.

Uncertainty applies to situations when one is unsure of the outcome and the odds are unknown.

When interacting with autonomous agents in a global computing environment the uncertainty in the outcome is usually due to uncertainty as to the future behaviour of the other agents. Computational trust models of evidence and reputation ultimately allow this uncertainty about potential actions be transformed into risk, and consequently the uncertain situation can be reasoned about as a risky one, using trust-based decision engines such as SECURE [1].

* Supported by the EU SECURE project, IST-2001-32486.

1.2 Decision-Making under Uncertainty in Economics

Hirshleifer's state-preference theory [4] aims to reduce choice under uncertainty to a conventional choice problem by changing the commodity structure appropriately. Thus preferences are formed over state-contingent commodity-bundles or, if we assume the commodity is currency, then *state-payoff* bundles. To use Hirshleifer's classic illustration, "ice cream when it is raining" is a different commodity from "ice cream when it is sunny".

Given a suitable formulation of the situation, the von Neumann-Morgenstern (vNM) expected utility rule ([5], see below) can then be applied to choose the best course of action given our beliefs about which state is likely to obtain. This approach fits nicely with the concept of Trust-Based Access Control (TBAC) used in the SECURE project [6], since it allows for reasoning about outcomes where a number of different states could obtain.

1.3 Suitability of the Expected Utility Theory for TBAC

Extensive experimental research has resulted in many concluding that the expected utility theory is not an accurate model of how humans make decisions, and much debate on this subject continues [7]. We feel that the psychological side of this debate — such as the supposition that humans do not meet the criteria for being rational[1] agents, and issues of the way in which the problem is framed [8] — is irrelevant to our model. *We are looking to build autonomous agents, not artificially intelligent ones.* [7] details a number of phenomena that are not well-modelled by the vNM utility rule, but since no unified model for all the phenomena described by [7] has yet been found, for simplicity we shall restrict our initial experiments to using the vNM rule.

2 A Trust-Based Access Control Model for SECURE

The state-preference model has the following five components:

- a set of acts available (X) to the decision-maker;
- a set of (mutually exclusive) states available to Nature (S);
- a consequence function, $c(x, \varsigma)$, showing outcomes under all possible combinations of acts and states[2];
- a probability function, $\pi(\varsigma)$, expressing the decision-maker's trust beliefs;
- an elementary-utility function (or preference scaling function), $v(c)$ measuring the desirability of the different possible consequences.

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

$$U(x) \equiv \sum_{\varsigma \in S} \pi_\varsigma v(c_{x,\varsigma})$$

[1] that is, will act to maximise their personal utility

[2] NB: ς is used to denote elements of S to avoid clashing with the existing SECURE terminology of using s as part of an (s, i, c) trust-triple.

2.1 Defining Preference using Utility

Previous models of risk have expressed preference over outcomes in the form of monetary costs and benefits. This approach has been criticised as being inappropriate because in many situations financial valuation is difficult or impossible. In this new model the abstract metric of (economic) utility is used. This represents money adjusted for time-effects (such as inflation and interest) and relative wealth, thereby overcoming the drawbacks associated with using actual money.

Whilst [9] observes that economists have proved that there is no meaningful measure of utility, the following definition from page 73 of [9], seems well-suited to our purposes:

Definition 1. *A real-valued function of consequences, v, is a utility if and only if $f \leq g$ (i.e. g is preferred to f) is equivalent to $v(f) \leq v(g)$, provided f and g are both with probability one confined to a finite set of consequences.*

We note that the concept of monetary costs and benefits satisfies this definition, which could be useful in applications where financial considerations are already present, for example the e-purse scenario outlined in [1].

2.2 Access Control Policy

A simple access control policy would be to choose any act, $a \in A$ where:

$$A = \Big\{ a \mid a \in X \land U(a) = \max_{X}[U(x)] \Big\}$$

If $|A| > 1$ then we choose the action $a' \in A$ with the smallest variance of utility.

Unfortunately this model does not yet address the problem of reasoning about the uncertainty and information content of trust-values. Intuitively a person will not feel confident in taking a course of action based upon a decision about which they have too little information. The quantity of the information required to satisfy them that this is the correct course of action will depend on their risk-aversity, and therefore must be encoded as part of the person's policy. Therefore we allow the policy-writer to set a lower-bound l for the number of bits of information known about the chosen action a, in order for it to be executed. If this check fails, then the system may test other actions in the set A or fall back to a pre-defined, "default" action.

2.3 Integration with the SECURE Trust Model

While this model is designed to be used with any suitable trust metric, we will now summarise how it is expected to integrate with the SECURE Trust Model [10]. This trust model provides for a principal to compute a set of beliefs about the likely outcome of interacting with another principal. Outcomes are defined in event structures and beliefs have the form (s, i, c) where s is the number of pieces of evidence supporting that outcome, c is the number of pieces of evidence contradicting that outcome, and i is the number of inconclusive pieces of evidence. By defining a mapping from the event structure of outcomes, E, to the states of Nature, S, it is trivial to compute a similar mapping from the set of beliefs in (s, i, c) form to a set of $\pi(\varsigma)$.

Entity Recognition (ER) also plays an important part in the SECURE model and the required level of confidence in the recognition of a principal should be determined by the risk associated with the decision. Intuitively, if an entity is incorrectly recognised, then the estimates for $\pi(\varsigma)$ are likely to be completely invalid. One way of representing this in the model would be to use the confidence in entity recognition to scale the measure of information the decision is based upon — the decision to perform act a is known to be based upon trust-values of the form (s, i, c) and since states are mutually exclusive, $\sum_S \pi(\varsigma) = 1, \forall \varsigma \in S, s + i + c = I$, which is the quantity of information, upon which the decision is based. This is then reduced in magnitude by the probability of incorrect entity recognition, $(1 - p_{er})$ before being compared to the information threshold l, described above.

An alternative method would be to perform a second risk analysis using the state-preference approach, once the set A has been determined. In this second analysis there would be just two states, $\{$entity recognised correctly, entity recognised incorrectly$\}$ and the acts would be A plus a fall back act for the eventuality that the risk of mis-recognition is too high to take any of the acts in A.

3 Selection under uncertainty as a decision problem

One of the weaknesses of the current SECURE trust model is that it is optimised for yes/no style decisions. For example, making a selection, such as choosing the best principal (or principals) from which to request a service, is very inefficient. This new model may be used to make a selection in the following way:

- The set of acts available (X) to the *selector* is equal to the set of potential service providers, P, or the power-set of P if combinations of providers are to be considered.
- The set of (mutually exclusive) states available to Nature (S). Each state represents whether each particular principal was "good" or "bad". For example, $\varsigma_1 = \{p_1\}, \varsigma_2 = \{p_1, p_2\}$ means that only p_1 satisfied us in state ς_1, while in state ς_2, p_1 and p_2 satisfy us, and so on.
- The consequence function, $c(x, \varsigma)$, showing outcomes under all possible combinations of acts and states — so $P \times S$ in the simple case where $X = P$. Good consequences, (x_i, ς_j) are those where $p_i \in \varsigma_j$.
- The probability function, $\pi(\varsigma)$, expressing the user's beliefs.
- The elementary-utility function (or preference scaling function) $v(c)$, measuring the desirability of the different possible consequences — so those that are "good" consequences, as mentioned above, have higher utility than those that are "bad". This function could also encode constraints such as a budget.

4 Example Application: Spam Detection

4.1 Risk Analysis of Spam Filtering

When considering whether an e-mail is spam or not, the decision-maker must decide whether to *mark* a message as spam or allow it to *pass* into the inbox. The situation can therefore be modelled as follows:

- The set of acts available to the decision-maker is $X = \{mark, pass\}$.
- The set of (mutually exclusive) states available to Nature is $S = \{spam, notspam\}$.
- The consequence function, $c(x, \varsigma)$ with example utilities for each consequence shown (where E is a parameter that allows the sensitivity of the filter to be adjusted):

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

- The probability function, $\pi(\varsigma)$, expressing the decision-maker's beliefs in whether the message is spam or not. How this is determined is discussed below.

4.2 Collaborative Spam Detection

Determining the belief function, $\pi(\varsigma)$, is difficult when dealing with spam. The lack of reliable authentication mechanisms in Internet e-mail makes forgery of existing identities and the creation of new ones too easy. This makes it difficult to make trust assessments about the sender of a message. An alternative mechanism, described in [11], uses a peer-to-peer (p2p) collaborative network to detect spam. It takes the view that, since identifying spam is an AI problem that can never be entirely solved using rule-based systems, even advanced ones based on Bayesian inference, the best method of identifying spam is still a human. Therefore the first human to identify a spam publishes a hash of the message to a p2p network and then each member of that network compares their incoming e-mail with the published hashes, as shown in figure 1. A trust and risk analysis is used to determine whether to mark the message as spam or allow it to continue into the user's inbox, given the opinion of other trusted nodes on the p2p network. Since spam e-mails increasingly have added random noise to try to defeat filters, fuzzy signature algorithms such as Nilsimsa[12] must be used to generate the *fingerprint*.

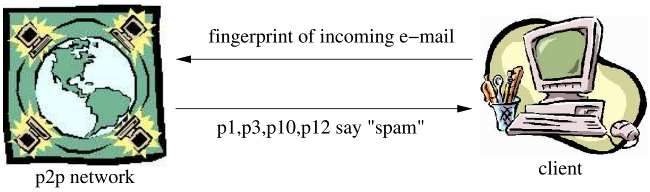

Fig. 1. Overview of the operation of a Collaborative Spam Detector

4.3 Trust and Identity

For simplicity each principal is represented by a public key so entity-recognition does not have a large role to play in the current implementation of this application.

In the trust model, each client locally stores a (s_j, i_j, c_j) triple for each principal, p_j, indicating how much they trust their judgement. s_j is the number of times they have given a correct opinion on whether an e-mail is spam or not, c_j is the number of times they have incorrectly described an e-mail and i_j is the number of opinions received that have yet to be confirmed as correct or incorrect.

The probability that a mail is spam given that p_j says it is spam is then given by:

$$\rho_j = p(\text{spam}|p_j) = \frac{s_j}{s_j + c_j}$$

The probability that a mail is spam given that p_j says it is **not spam** is given by:

$$\rho_j = p(\text{spam}|\overline{p_j}) = \frac{c_j}{s_j + c_j}$$

The information from each principal is then weighted, based upon the number of previous interactions we have had with them and using a weighted-mean, to determine an overall probability of whether a message is spam or not:

$$p(\text{spam}) = \frac{w_1\rho_1 + w_2\rho_2 + ... + w_j\rho_j}{\sum w_j}$$

where $w_j = s_j + c_j$. Information from principals with w_j below the information threshold, l, is ignored.

When the owner of the system reads their mail they give feedback to the system as to whether or not it made the correct decision. This may be implemented implicitly by monitoring how the user deals with the e-mail (such as moving it to a certain folder), or an explicit feedback channel exclusively for correcting errors (as used in [11]).

Once a message has been verified by the human operator of the system, the (s, i, c) triples for each principal that published an opinion are updated.

4.4 Recommendations

If there is insufficient information (as determined by the information threshold, l) in the local trust database about principal Alice who offers an opinion on a particular e-mail, then the trust engine may also query the p2p network for recommendations from other principals about their experiences with Alice. To avoid problems with second-hand information and loops, only information from direct experiences is published, in the form of (s, i, c) triples.

Clearly, naïvely trusting recommendations from other p2p users is dangerous. In order to incorporate these recommendations into the probability calculation, they are first discounted [13], using the belief in the principal's *recommendation integrity*; they are then averaged over all principals who supplied a recommendation, before being added to any local trust information.

Recommendation integrity (also known as “meta-trust”) is calculated using the concept of *semantic distance* from [14]. Received recommendations are stored in a cache and then, after a certain number of interactions with the subject of the recommendation have taken place (arbitrarily chosen as five in the initial prototype), the value of the (newly obtained) local *observed* trust value is compared with the *received* trust value. The main difficulty here is that the information content of the received value is likely to be far higher than the experience value. The two trust values are therefore normalised, giving (where info $= s + i + c$):

$$\left(\frac{s_o}{\mathrm{info}_o}, \frac{i_o}{\mathrm{info}_o}, \frac{c_o}{\mathrm{info}_o}\right), \left(\frac{s_r}{\mathrm{info}_r}, \frac{i_r}{\mathrm{info}_r}, \frac{c_r}{\mathrm{info}_r}\right)$$

The recommendation integrity, RI, is then given by the ratio:

$$\frac{s_o}{\mathrm{info}_o} : \frac{s_r}{\mathrm{info}_r}$$

We observe that if $RI < 1$ then the recommendations tend to be better than the behaviour we experience, and so any recommendation should be scaled down accordingly. In contrast, if $RI > 1$ then the recommendations are more negative than our observations indicate. Note, we do not scale up in this instance as this would allow an attacker to manipulate trust values by giving negative recommendations about those it wished to make appear more trustworthy. Instead we observe that anyone who is guilty of “bad-mouthing” another principal is unlikely to be very trustworthy at all and ignore any recommendation they may give if $RI > 2$.

The RI for a recommender is calculated at most once per recommendation received to ensure that principals are not overly penalised if the principal they recommend subsequently changes their behaviour. Since principals are likely to make many recommendations, the RI value for a principal is an average over all recommendations that principal makes.

4.5 Evaluation

A comprehensive evaluation of this application has been undertaken, using both analytical and empirical approaches. Unfortunately space constraints preclude any detailed discussion here. The approach was threat-based, analysing what type of behaviour individual attackers might use to influence decisions, and simulation was then used to determine how these individual behaviours would interact at a system-wide level.

The results indicated that the attackers operating in isolation were quickly identified and ignored and so their best strategy was to co-operate in a co-ordinated manner. To mitigate the Sybil Attack [15], policies were used that forced principals to supply more correct than incorrect information in order to be able to affect decisions.

5 Research Context and Conclusions

This paper has presented a novel model of risk and trust-based access control based on the economic theory of decision-making under uncertainty. This model has been evaluated through the implementation of a p2p collaborative spam detection application which demonstrated its effectiveness.

This work builds on the authors' experiences of developing and using the SECURE risk model [2], which whilst being flexible and expressive was found to have a number of weaknesses in practical deployment. Jøsang and Presti [16] also use the expected utility theory to model agent risk aversity, but their model concentrates on using risk to deduce trust rather than for access control. Further work could involve integrating our approach with Jøsang and Presti's, and an investigation of more complex expected utility functions.

References

1. Cahill, V., et al.: Using trust for secure collaboration in uncertain environments. IEEE Pervasive Computing **2** (2003) 52–61
2. Dimmock, N.: How much is 'enough'? Risk in trust-based access control. In: IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises — Enterprise Security. (2003) 281–282
3. Knight, F.H.: 1. In: Risk, Uncertainty, and Profit. Library of economics and liberty. 8 September 2004 edn. Hart, Schaffner & Marx; Houghton Mifflin Company, Boston, MA (1921)
4. Hirshleifer, J., Riley, J.G.: The analytics of uncertainty and information. Cambridge surveys of economic literature. Cambridge University Press (1992)
5. von Neumann, J., Morgenstern, O.: Theory of Games and Economic Behavior. Second edn. Princeton University Press (1947)
6. Dimmock, N., Belokosztolszki, A., Eyers, D., Bacon, J., Moody, K.: Using trust and risk in role-based access control policies. In: Proceedings of Symposium on Access Control Models and Technologies, ACM (2004)
7. Machina, M.J.: Choice under uncertainty: Problems solved and unsolved. The Journal of Economic Perspectives **1** (1987) 121–154
8. Simon, H.A.: Models of bounded rationality. Volume 1. MIT Press (1982)
9. Savage, L.J.: The foundations of statistics. Second edn. Dover, New York (1972)
10. Carbone, M., Dimmock, N., Krukow, K., Nielsen, M.: Revised computational trust model. EU IST-FET Project Deliverable (2004)
11. Dimmock, N., Maddison, I.: Peer-to-peer collaborative spam detection. ACM Crossroads **11** (2004)
12. cmeclax: Nilsimsa codes. `http://ixazon.dynip.com/~cmeclax/nilsimsa.html` (2004) Accessed 22 November 2004 17:30 UTC.
13. Jøsang, A.: A logic for uncertain probabilities. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **9** (2001)
14. Abdul-Rahman, A., Hailes, S.: Supporting trust in virtual communities. In: Hawaii International Conference on System Sciences 33. (2000) 1769–1777
15. Douceur, J.R.: The Sybil attack. In: Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS'02). Number 2429 in LNCS, Springer-Verlag (2002) 251–260
16. Jøsang, A., Presti, S.L.: Analysing the relationship between trust and risk. In: Proceedings of the Second International Conference on Trust Management (iTrust'04). Number 2995 in LNCS, Oxford, UK, Springer (2004) 135–145