The right type of trust for distributed systems

Audun Jøsang

Department of Telematics
The Norwegian University of Science and Technology
N-7034 Trondheim

Abstract. *Research in information security has traditionally focused on where to place or how to propagate trust. In that sense, a cryptographic algorithm or protocol is simply a mechanism to transfer trust from where it exists to where it is needed. This paper puts the focus on trust itself and shows that it is a very complex concept with many interesting and important implications. We do not attempt to define a formal trust model, but rather examine the types of trust and trust relationships which are relevant for information security. It is shown that the existence of trust as a phenomenon depends on the existence of malicious behaviour. This observation leads to the distinction between passionate entities with human-like capabilities, and rational entities which basically are systems. Trust can then be defined as the belief that a rational entity will resist malicious manipulation or that a passionate entity will behave without malicious intent. It is also shown that trust relationships exhibit a great diversity, that they are based on knowledge and that they contain aspects in common with strategy games.*

1 Introduction

What is trust, why is it needed and how do we use it? These questions may seem so difficult to answer that many would like to avoid them altogether. In many scientific studies and research papers on In many scientific studies and research papers on information security, this is exactly what is being done. By assuming absolute trust in some parts of the system, one can concentrate on the more concrete problems of where to put trust and how to propagate it in order to obtain the most optimal security schemes and protocols. In this paper, we will open the Pandora's box, and look into the concept of trust itself.

Trust is an essential factor for human interaction, but is trust still meaningful when systems interact with other remote systems? To the degree that system entities are tools used by humans, a distributed system still constitutes social interaction indirectly and we can conclude that trust is meaningful and necessary here also.

From an information security point of view, it can be observed that humans are trusted because they are believed to be honest whereas systems are trusted because they are believed to be secure, and this will form the basis of defining two different types of trust. In a distributed system involving both human-like and system-like entities, it will be advantageous to interact with the most secure or honest, and thereby trustworthy entities, because it minimises the exposure to risky transactions. Three basic problems must be addressed in this respect. Firstly, it is important to properly understand the concept of trust and how it manifests itself as a human phe-

nomenon. Secondly it is necessary to know how trust can be extracted as a parameter from the real world. Finally it should be investigated how trust can be integrated in formal models in order to use it as a parameter among others to optimise system performance and quality of service.

This paper mainly focuses on the first and most fundamental problem of properly understanding trust in the real world. This topic lays on the edge of the computer science and information security disciplines, and it is with recognition of our limited insight in the fields of psychology and behavioural science that we humbly attempt this study.

Section 2 below gives a short survey of recent papers on the subject of trust relative to information security. In section 3 we define the two main types of trust relationships as trust in human agents and trust in systems. This is further developed in section 4 where trust relationships are analysed from a role playing point of view. Section 5 looks at malicious behaviour from a philosophical viewpoint. Section 6 illustrates the diversity of trust, section 7 explains why trust should be based on knowledge, and section 8 shows that trust and strategy games have several aspects in common. Section 9 compares security with the related concept of reliability and illustrates their difference relative to trust. A postscript is added after the conclusion to reflects the discussion that the presentation of this paper induced during the workshop.

2 Brief literature survey on trust

Yahalom *et al.* have published two interesting papers [YKB93, YKB94] which propose a formal model for deriving new trust relationships from existing ones. In [YKB93] a *Trust Classification* is defined. According to this, being trusted for a particular class means that an entity can be trusted to perform a specific task like e.g. key generation, keeping secrets or clock synchronisation, without necessarily being trusted for other tasks. There can in this way be multiple trust relationships between the same pair of entities. In [YKB94], rules and algorithms for obtaining public keys based on trust relationships are developed. Neither of these papers attempt to define trust itself.

Beth *et al.* [BBK94] present an extension of [YKB93] which assumes relative trust. The paper presents a method for extracting trust parameters from the real to be used in formal models such as [YKB93] and [YKB94], and the authors claim that this method can be used to accept or reject an entity as being suitable for a sensitive task. The method is almost purely statistical and is based on the assumption that all trusted entities have a consistent and ultimately predictable behaviour. It is doubtful whether this simplistic approach is adequate to quantise the complex behaviour of potentially malicious agents. In our view, the method is better suited to evaluate reliability, or it can at most provide a necessary but not sufficient test of trustworthiness.

Denning [Den93] has analysed the concept of trust related to trusted systems and market requirements. She argues that trust is not a property of a system, as usually assumed, but the result of an assessment made by an observer about a person, organisation or object being observed. This has important implications for the way security evaluations are being conducted, in that the focus is shifted from the system to the relationship between the observer and the system. Security evaluation according to ITSEC [EC92] TCSEC [USD85] produces a certain assurance level and is thereby an example of a method for extracting trust parameters from the real world, and Denning's observation invokes the difficulty of defining a firm and absolute basis for the evaluation and determination of security assurance.

Simmons *et al.* [SM95] have analysed the propagation of trust in access control systems in which an action can only be performed by certain individuals acting in concert. If a security protocol has been designed only to be executable with a particular combination of participants, it is shown that additional trust between the participants can create unintended combinations of participants which are able to execute the protocol, and that this can weaken the security scheme. A method is presented to determine all such combinations as a function of the additional trust relationships that can exist, and this can be used to verify a protocol's strength against unintended executability.

Campbell *et al.* [CSNP92] have taken a probabilistic approach to trust related to security protocols. They build on the BAN-logic [BAN89] and attach probabilities to the sentences and rules of the logic in order to

determine a minimum trust in the goal of the protocol. However, the assumption that trust can be modelled as probability is very simplistic and does not take into account its human aspects.

3 Defining trust from a malicious point of view

At this place, it is appropriate to define some concepts which will be used throughout the paper. The characteristics *honest*, *dishonest*, *straight*, and *crooked* can only be used to describe human agents as opposed to systems. I am honest if I keep my word and dishonest if I don't. I am straight if I follow the rules and crooked if I don't. It is then perfectly possible to trust a crooked person, and an example will illustrate this. If I tell you that I am going to steal your car, and then do it, I am an honest crook because I kept my word and broke the law at the same time, and interestingly, my honesty was trustworthy in that particular case.

The two most relevant combinations are honest/straight which we will call *benevolent* and dishonest/crooked which we will call *malicious*. In the rest of the paper only these two combinations will be considered, as probably nobody would interact with honest/crooked or dishonest/straight agents.

Trust is a positive concept. It expresses that we expect something positive from the trusted entity, or in other words, we expect it to have a desired property or to behave the way we want. To be "fault free" or to "behave correctly" would be too general, which for reasons given below belongs to the concept of reliability and the more general concept of dependability. The main rationale behind information security is that some agents will behave maliciously in a given situation, and try to attack or manipulate IT systems. Trust relative to IT security must therefore somehow reflect the resistance against malicious threats.

If there was no malicious behaviour in the world, trust would no longer be a useful concept because everything could be trusted without exception. On the other hand, the total lack of trust would signify that malice and betrayal have penetrated everything and everybody. This indicates that the relevance of trust depends on the uncertainty of whether somebody is benevolent or malicious, or simply on the existence of both types of behaviour in society. We will assume this to be true and let it form the basis of the further analysis of trust below.

A trust relationship requires at least the involvement of two parties. We will first focus on the trusted party, subsequently on the trusting party, and in each case ask what is required of it in order to be part of the trust relationship.

3.1 The trusted party

By observing that it is possible to trust humans as well as systems we notice that the nature of the trusted party can vary over a wide range. The distinction between a pure system and the human agents who use is not obvious because their actions and involvement are often deeply integrated in the operations of the system. Nevertheless, by defining a pure system as the aspects of a system which during operation are unaffected by human involvement, we will distinguish between a class of human-like entities called passionate entities and a class of system entities called rational entities, and define trust according to each class.

3.1.1 Trusting a passionate entity

By considering a human, I will trust him if I believe him to be benevolent, and mistrust him if I believe the opposite. A human entity is thus expected to be either benevolent or malicious. One can never be absolutely sure about somebody else's benevolence, and trust can thus be no more than a belief. It may be true that ultimately there is a finite number of factors which determine whether a human behaves in a benevolent or malicious way, but it is impossible to obtain perfect knowledge about his nature and of all other influencing factors. A human agent's behaviour is therefore impossible to predict, even for the agent himself.

For all practical purposes, whatever the underlying mechanism may be, we will call the human-like mechanism which chooses between benevolent and malicious behaviour the *free will*. We define entities possessing this kind of free will as *passionate*. One usually considers passion to be a purely human characteristic, but it

would be arrogant to exclude certain animals. For that reason, when using expressions like "human behaviour" in the rest of the paper, we also refer to certain aspects of animal behaviour. The borderline between passionate and rational entities necessarily becomes blurred and maybe someday one will have to accept machines as passionate. For the time being however, we will consider humans, human organisations or a combination of systems, humans and human organisations as passionate entities, but never pure systems alone. This kind of trust relationship illustrated in figure 1 leads to the definition of the first type of trust.

Trust in a passionate entity is the belief that it will behave without malicious intent.

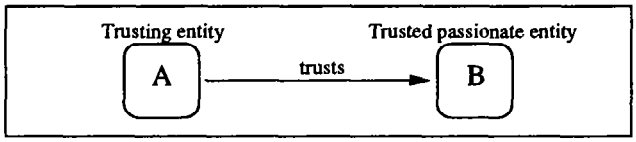

Figure 1: Trusting a passionate entity

There exists no universal principle or method to consistently determine trust in humans. Malicious but trusted employees are therefore common in most organisations and this is a particularly dangerous threat because it can not be stopped by traditional security mechanisms.

3.1.2 Trusting a rational entity

Algorithms, protocols, software, hardware or even the most complex computers can hardly be characterised as passionate or having a free will, but we still would like to be able to trust those as well.

The simplest definition of a *rational* entity relative to trust is to call it an entity which is not passionate. In other words, a rational entity lacks the human aspect of spirit and free will. A rational entity will usually be a pure system although it is possible to consider the inclusion of human involvement by excluding the passionate aspects of human behaviour.

Because a rational entity has no free will, it is not expected to be benevolent or malicious. What exactly is being trusted is not how it will behave, but rather that it will resist any attempt of malicious manipulation by an external malicious agent. There is thus a third party involved, namely the external threat. This kind of trust relationship is illustrated in figure 2.

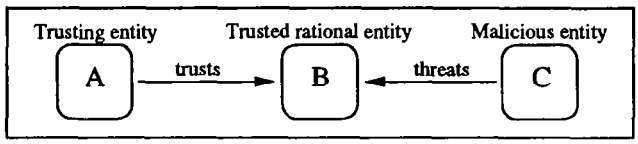

Figure 2: Trusting a rational entity

A threat is a potential malicious manipulation or attack. Even if it is known that somebody out there will attempt an attack, it is a priori impossible to know who exactly will realise the attack or what the attack will be. Again, the decision whether an external entity will attempt a malicious attack or not, must be credited to its free will, and the threatening entity must therefore be considered as passionate according to the definition given above. This leads to the definition of a second type of trust.

Trust in a rational entity is the belief that it will resist malicious manipulation by a passionate entity.

3.2 The trusting party

Trust is a belief that one entity has about another entity. Firstly, there must be a reason behind the belief, and secondly, the belief expresses an expectation of how an entity will behave or perform. The reason behind the trust can be composed of many elements, like past experience, knowledge about the entity's nature, recommendations from other entities or some kind of faith. This indicates that the reason behind trust is complex and often is based on unquantifiable amounts of information, and that it requires human-like capabilities to be able to trust. We therefore claim that only human entities are able to assess trust as defined in the previous section and that trust only makes sense to humans. This rather philosophical assumption can be supported by a few observations.

The previous section showed that there always is a

passionate entity involved on the side of the trusted party, either directly as the trusted entity itself or as the malicious threat. It therefore seems natural that the trusting entity needs similar reasoning faculties to properly assess whether an entity can be trusted, or said in other words, passionate behaviour can only be appraised by other passionate entities. The trusting entity must therefore necessarily be passionate too, and trust becomes a relationship involving peer passionate entities. If we admit that trust to some degree results from belief or faith rather than a simple assessment of probability and that belief and faith are human aspects, we also reach the conclusion that the ability to trust and mistrust essentially is a human faculty which can not be possessed by computers. An immediate implication is that a proper establishment of trust relationships can never be entirely automated. A few examples can illustrate this.

If a machine is instructed to check persons for trustworthiness using for instance multiple choice questionnaires, is the machine then trusting those who pass the test? Or if a machine checks fingerprints and retina patterns against stored values of known trusted persons, is the machine then trusting the persons who match some entry in the list of values? The answer in both cases, as explained below, is no!

In the first case, any person would be able to pass as trustworthy by simply learning a set of correct answers because they are not secret. Even if other criteria were added, it would still be possible to fool the system by learning how it works. As a result, system designers would need to constantly upgrade and modify the criteria, so that the system finally could not be called automated anymore.

The second case is the well known authentication by something you are. In reality, it is the uniqueness of fingerprints and retina patterns which is trusted, as well as the integrity of the system and the list of pre-stored values. The system is simply transferring this trust, by a formal model implemented in the system, onto the person possessing the matching fingerprints and retina patterns.

A rational entity can be instructed to trust other entities, but it will always be on behalf of passionate entities. How trust parameters can be entered into systems in this way corresponds to the problem of extracting trust from the real world into formal models. Once rational systems have been given instructions about initial trust relationships, one can imagine systems that automatically derive new trust relationships according to some formal model. This corresponds to the problem of integrating trust into formal models in order to optimise system security. As already mentioned, these topics are outside the scope of this paper.

4 Trust relationships

We will in this section give an overview of the main types of trust relationships from a role playing point of view. According to the definitions in section 3, trust in a system involves three entities; a passionate trusting entity, a rational trusted entity and a passionate external threat. By combining these three roles in different ways, we will illustrate various trust relationships, of which two are generalised versions of those already described in section 3. The description is only illustrative and is not intended to be exhaustive.

Since a malicious passionate entity is a threat to systems, it is natural to ask whether a passionate entity can also be a security threat to humans. Obviously the answer is yes, if the first entity were only a little bit smarter than the second. It then becomes possible to trust passionate entities not only to be benevolent, but also to be smart enough not to be fooled by others with malicious intent. This could also be explained by letting the same entity play two separate roles, where one is the trusting and the other the trusted, or in other words by giving it a split personality. When an entity plays two or three roles simultaneously, we will assume it to be passionate as long as one of the roles is passionate.

I can trust myself to be rationally smart enough to resist malicious manipulation by others, or I can trust somebody to use their rational reason to resist the temptation of behaving maliciously in a given situation. In this way it is possible to combine the roles into entities in various ways as illustrated in figure 3.

In every case, the two passionate roles are always the trusting (A) and the threat (C). The rational role (B) is either a separate entity or a sub-entity of a passionate entity. The real entities in figure 3 are thus the rounded rectangles or squares containing a combination of the

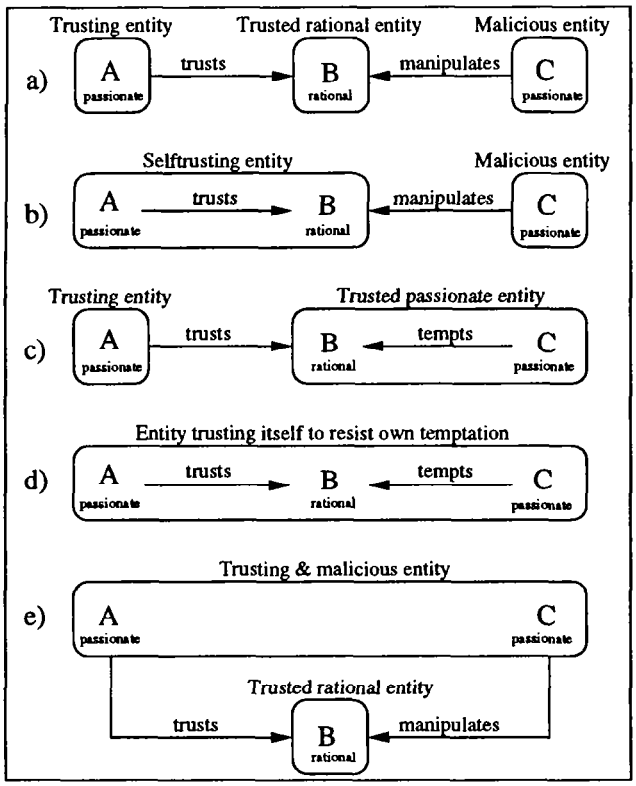

Figure 3: Role based trust relationships

a) A rational entity is being trusted to resist malicious manipulation.
b) An entity trusts itself to resist malicious manipulation.
c) An passionate entity is trusted to resist temptation of becoming crooked.
d) An entity trusts itself to resist the temptation of becoming crooked.
e) An entity who is tempted to become a crook trusts a rational entity of resisting his own malicious attacks.

roles A,B or C. As already explained, an entity is considered passionate if it contains at least one passionate role.

It can be observed that the situation in figure 3.a is the same as that in figure 2 and that the trust relationship in figure 3.c in reality is a generalisation of the trust relation from figure 1. In figure 3.b, the trusting entity can either be viewed as a human or as a human organisation trusting itself for being resistant or secure against external threats. The trust relationship in figure 3.d can be relevant for policy making and training of personnel because it illustrates the constraints placed upon a human, causing it to either cooperate or to defect. Figure 3.e illustrates the hackers view of a system he wants to attack. It also illustrates how system designers must put themselves in the role of the attacker in order to understand the potential threats.

5 Malice and Kant's categorical imperative

In section 3, malice was defined as the combination of dishonesty and crookedness. This section takes a closer look at what it actually means to be malicious.

What exactly constitutes malicious behaviour can never be absolute and can only be defined relative to a security policy, moral rules, contracts and legislation. By consequence, security domains with different policies can have conflicting views on malicious behaviour, illustrating the great challenge of establishing a firm basis for secure inter-domain transactions. Rather than discussing multi-domain security which is a problem of formal modelling, we will discuss the more philosophical problem of establishing a more absolute basis for defining malicious behaviour in general.

No human is perfectly benevolent, and everyone of us must admit that even they could become malicious. As figure 3.d illustrates, it is possible to trust oneself to have the necessary power of judgement to resist the temptation of becoming malicious, but this implies a rather schizophrenic personality. Who am I when I want to be a crook, and who am I when I resist it? Am I able to separate malicious behaviour from good behaviour? If even I can not agree with myself on what good behaviour should be, can we all agree on it? It is perfectly possible to imagine entities belonging to different political or economical domains where the respective publicly accepted norms for good behaviour can be incompatible on certain points. What do we do then? Are malicious

entities necessarily conscious of being malicious? These uncomfortable questions tend to seek an answer in something more absolute and universal which always can tell the difference between good and malicious behaviour. One possible solution to this fundamental problem can be found in Kant's Categorical Imperative [Kan] which can be summarised as follows:

> *Act only on that maxim whereby thou canst at the time will that it should become a universal law.*

Assume a man with a split personality as described above, and name the two personalities as Mr.Passion and Mr.Rational. Mr.Passion, as his name indicates, is totally governed by his passions, and Mr.Rational applies his rational reason to judge whether the actions proposed by Mr.Passion are a good thing to do. Mr.Rational can actually apply different criteria for his judgement, but Kant is never clear about what rational reason specifically instructs. He found that since we are all equipped with rational reason, it could only ever tell us to do something which it would be possible for everyone to do. This is the test provided by the categorical imperative, and reason guides us by telling us to exclude those actions which do not pass the test. Thus we should not want to do something which we could not wish would be done by everyone.

As an example[1] on how the categorical imperative might be applied, consider our man with a split personality wondering whether to pay his taxes. Mr.Passion proposes non-payment, and if Mr.Reason uses short term profit as criterion for his judgement, non-payment would be executed. However, such an action would not pass the test of the categorical imperative. By considering not paying while at the same time accepting the premise that others use the same criteria, our candidate would be committed to the predictable result that society would break down without the necessary funding from taxes, and we will assume that this is not something he would like to see happen.

The application of the test will not always be as simple as in this example, but it still puts the highly relative concept of malicious behaviour in a more absolute and general frame. As a final remark it can be said that Kant's categorical imperative test hardly will have to be called upon in practice for defining what constitutes malicious behaviour in a security policy document.

[1] Example taken from [HHV95] p.16.

6 Trust diversity and interdependency

In the previous sections, only *target diversity* has been discussed, i.e. the fact that trust varies as a function of the trusted entity. Obviously it is also important to consider what exactly is being trusted. Denning [Den93] has observed that trust is relative to a *domain of action*, and Yahalom *et al.* [YKB93] have defined a *trust classification* which in a more formal way expresses the same thing. We will use the term *trust purpose* for this concept. In security evaluation criteria such as ITSEC[EC92] and CC[ISO96], the same aspect is reflected by the specification of *functionality*[2]. The trust purpose then expresses exactly what the target is being trusted for. But the diversity does not stop here, because trust also depends on the trust origin, i.e. not every trusting entity will have equal trust in the same trusted entity for the same purpose, and we will call this *origin diversity*. The three diversity types are illustrated in figure 4.

Without going too much into detail one can observe that separate trust relationships often are related and interdependent. For target diversity, this can mean that if one particular target is being trusted, then others are too.

Concerning purpose diversity, it is useful to consider humans and systems separately. If an employee is trusted to operate a high security system, can he be trusted not to cheat with his travel vouchers? Both actions depend on his benevolence and will be relatively but not absolutely dependent. In case of systems which generally provide different and independent services, there may still be dependencies. If for instance several services have low quality, it is likely that the whole system is badly designed and implemented, causing a reduced trust in every service.

[2] The earlier set of criteria TCSEC[USD85] on the other hand does not contain function diversity, because increased trust depends directly on enhanced functionality.

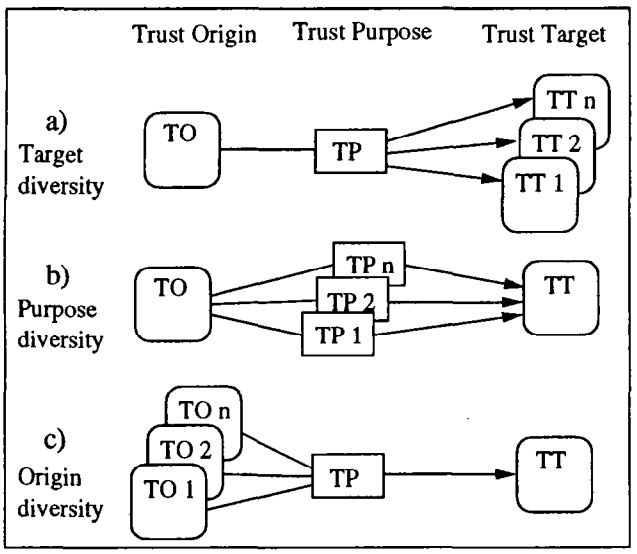

Figure 4: Trust diversity

Origin diversity is primarily caused by differing knowledge among trusting entities causing them to have different trust in the target. Another aspect to consider, when the target is passionate, is the target's knowledge about the trusting entity, i.e. that the target's benevolence depends on the trusting party. Then even in case of identical knowledge at the origin, different trust may result.

It should not be necessary to mention that a combination of the three types of diversity can generate a very large number of trust relationships, and that there even may be other types of diversity than those mentioned here. This is the reality one has to take into account when trying to understand and use trust as a parameter in system modelling.

7 Trust as knowledge about security

There is a significant difference between what trust is based on in real life, and what it should be based on for the purpose of information security. Humans can be irrational, and so can trust. Irrational trust is not based on knowledge, but e.g. on faith, and can sometimes persist in spite of knowledge. This type of trust may be valuable in other situations but can be risky for for security. The right type of trust for distributed systems should as much as possible be based on knowledge.

We will define knowledge as information which can be used for a specific purpose. In this case we are interested in information which can be used for determining trustworthiness. Any information which contributes to this task then becomes knowledge.

A user of a system can never obtain perfect knowledge of the system he uses nor of the threats, and he is therefore unable to exactly evaluate the system's security. By gathering as much knowledge as one can about the system, a user will get an idea or a belief about the security, or in other words, a certain trust in the system. The trust thus reflects the user's knowledge about the system's security. Trust and security can be said to represent two sides of the same thing. Security reflects the idealistic side like e.g. formal modelling, design and development, or in short how we would like the systems to be in theory. Trust on the other hand reflects the realistic side of system knowledge taking into account that no formal model is perfect and that errors will always persist no matter how strict the design procedures are.

It is interesting to notice that in common language usage, both humans and systems can be trusted, but only systems can be secure. When seen from a knowledge point of view, the reason for this difference is probably because any realistic knowledge about humans always will be imperfect and very limited, whereas knowledge about systems can reach a high degree of correctness and completeness.

In a distributed system there can be a hierarchy of trust relationships, where some represent more or less the final trust a user needs in a particular application, whereas others only are useful as underlying support in order to establish the final trust. Examples of a supporting trust can be the trust in a cryptographic key or even the trust in a system, because a key or a system are only tools for providing services to a user. Examples of final trust can be the trust in the authenticity of a document or the confidentiality of a message transfer.

An explicit trust relationship can be characterised as *direct trust*. Trust relationships can also be implicit, i.e.

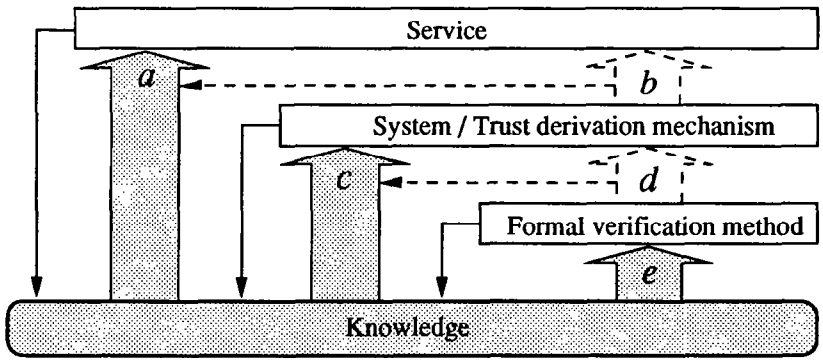

Figure 5: Trust based on knowledge

Legend:

	a	Final trust
	b	Derived trust
	c	Trust in system or trust derivation mechanism
	d	Trust in system derived by formal verification
	e	Trust in the formal verification methods
$- - \rightarrow$		Trust derivation
$\longrightarrow$		Knowledge extraction

potential but not yet existing, when there is an indirect trust path between the trusting and the trusted party. When using the term *derived trust* we want to invoke the establishment of a new explicit trust relationship, based on other already existing trust relationships or system knowledge. Once a new trust relationship has been derived and established, it immediately becomes direct. Deriving trust therefore only has meaning as a way of establishing new trust based previously existing trust. This is very similar to the concept of knowledge extraction because the reason to trust existed before it was derived, but the trust had not yet been made explicit.

Figure 5 illustrates how knowledge extraction or trust derivation at different levels create a hierarchy of trust. Each trust type is identified by the letters a through e. Trust a is the user's final trust in a service. This trust can be established by informal methods or it can be derived by a formal system as illustrated with trust b. The horizontal arrow $a \leftarrow - - b$ symbolises that the derived trust is made explicit and becomes direct once it has been derived. Obviously, the trust derivation methods themselves also need to be trusted. This is illustrated by trust c. Again this trust can be established by formal methods as symbolised with trust d and the horizontal arrow $c \leftarrow - - d$. Finally, the verification methods need to be trusted, as illustrated by trust e. The process of deriving or making trust explicit is in reality knowledge extraction as indicated by the solid arrows into the knowledge base.

It should be noticed that the trust hierarchy in figure 5 is recursive in that derived trust can be used for further trust derivation, and that formal verification also can be formally verified. The underlying trust e, and to a certain extent c, not only reflects security, i.e. strength against malicious attacks, but also other aspects of dependability as long as these aspects contribute to increasing the final trust a.

Because of its recursive character, the model in figure 5 is a highly dynamic one. Any change in trust e has direct influence on d and by consequence on c. Similarly any change in trust c influences b and a. The dynamism in the model is amplified by the fact that the model is recursive. When stable security is a goal, the trust

should be stable too. For this, the knowledge must be as complete as possible because it reduces the likelihood of being surprised by new knowledge revealing security weaknesses or breaches.

8 Trust as a strategy game

Passionate entities can have an incentive to misuse the trust of others. As an example from the world of distributed systems, one can imagine a malicious entity which behaves correctly during a certain period in order to accumulate high trust from other entities, then suddenly defects for a transaction with very high value, and subsequently disappears from the network.

This example shows that trust can be manipulated, and who will win in the end may depend on who is the smartest. It does not take much reasoning to see that we may end up in an endless feedback loop between the trusting and the trusted entity. The loop is caused by the knowledge the peer entities have about each other combined with their power of reasoning. The first loop says: Entity A trusts entity B, but when entity B knows that it is being trusted, it can manipulate A. The second loop says: A knows that B is planning to defect based on the trust he believes having, so finally A does not trust B anymore. But then again, B knows that he is not being trusted, and decides to cooperate in order to gain some trust. This reasoning can continue *ad infinitum*, and takes all the characteristics of a strategy game. The relationship suddenly becomes recursive and in theory infinitely complex. This is obviously undesirable, so the loop should be broken.

It seems that the only way to break the loop is to restrict the knowledge the entities have about their trust relationships. This is indeed paradoxical, and may seem impossible to achieve, because one prefers to interact with those one trusts the most, but by doing so we reveal our trust. This seems to create very pessimistic perspectives for the future of distributed systems, and if indeed the entities were perfectly selfish and only cared about their short term profit and individual advantage, it would lead to the certain breakdown of any distributed system.

So far in this discussion the question of what motivates benevolent or malicious behaviour has not been thoroughly considered. Since most distributed systems do work and trust is maintained, it seems that most players are inherently benevolent, or that they see the benefit to the whole system, and ultimately to themselves, of benevolent behaviour. In this way, the loop invoked above looses its vicious character and becomes: I trust you, and you trust me, and we are both happy knowing it.

It seems that cooperation in a strategy game is based on a special kind of relationship. It is the expectation that the other entity also will cooperate because it will profit from cooperation. But if trust relationships are based on strategic considerations like this, then distributed systems seem to become huge strategy games resembling battlefields rather than stable environments for interactions and service exchange. It may be questioned how it is possible to base security on this kind of trust because it would be too unstable and unpredictable. It could even be questioned whether this can be called trust at all in the sense it was defined in section 3, because it no longer seems to have benevolence as reference. Nevertheless, by adopting benevolence as a strategic consideration in itself, it is still possible to define trust in this sense, because honest and straight behaviour, which is what we have called benevolence, ultimately can be seen as selfish and also the most profitable in the long run. This illustrates that benevolence perfectly well can be rooted in purely strategic and selfish considerations, and that it does not need to be an inherent metaphysical property of the trusted party.

In order to have a stable security level the trust relationships also need to be stable, as already mentioned. When assessing somebody's trustworthiness as a result of strategic considerations, it must be focused on whether benevolent behaviour is the trusted entity's most fundamental and inalterable strategic principle. This simple criterion, although difficult to assess and formalise, would then be crucial when considering a passionate entity's trustworthiness.

	What is trusted for the purpose of:	
	Security:	**Reliability:**
Passionate entities:	Benevolence	Skill, experience
Relation type:	$P \rightarrow P$	$P \rightarrow P$
Rational entities:	Strength against attack	Continuous operation
Relation type:	$P \rightarrow R \leftarrow P$	$R \rightarrow R$

Table 1: Comparison between security and reliability

9 Comparison between security and reliability

By excluding the malicious entity from the model, and making rational entities our objects of study, the term reliability is more appropriate than security. Trusting a system to be reliable would give a different meaning to the concept of trust than we have used so far, because it no longer would be related to security. Reliability can be analysed with statistical methods where the amounts of information is finite. It covers aspects such as failure rate and repair time. Both the concept of security and reliability fit in under the more general term *dependability* as defined for example in [Lap92].

When studying the reliability of a system, all malicious threats must be totally ignored, even if they in reality are present. Doing that puts entities in a different light. A system can be trusted for not being vulnerable to malicious manipulation, although it may be totally unreliable and crash all by itself. The reliability of the system then covers the degree to which the non-malicious designers have succeeded in making it fault free. Assessment of reliability can be modelled as a binary relation as illustrated in figure 6.

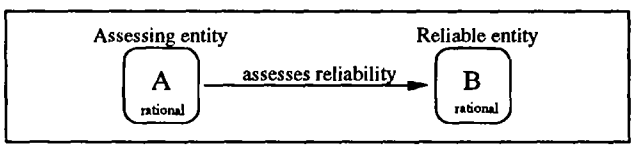

Figure 6: Reliability assessment

By definition, the assessed entity is always rational. A natural question is therefore whether the assessing entity is rational or passionate. We believe that ideal[3] reliability assessment should be a purely rational activity and thus that reliability assessment of systems is a relation between rational entities.

So far in this and previous sections, trust in passionate and in rational entities have been described, as well as the reliability of rational entities. In order to complete the picture, it should be investigated to what degree it is possible to assess the reliability of human or passionate entities. Which human characteristics are relevant for this purpose, or what makes people reliable except for benevolence? Human qualities like skill and experience seem important, but there are probably other characteristics to consider. Intuitively it again requires a passionate entity to assess such human qualities, indicating that this is a relation between passionate entities.

Without taking this discussion any further, the difference between security and reliability can be resumed in a table as illustrated in table 1.

For this purpose, the term trust is given a more general meaning differing from what has been used in previous sections, so that also the reliability of systems and the skill of humans can be trusted.

Each table entry also indicates the relationship type, i.e. whether it is between passionate or rational entities. The expression $P \rightarrow R \leftarrow P$ for instance indicates that a passionate entity P trusts a rational entity R to resist attacks from a passionate entity P.

[3] in the sense "theoretically perfect"

10 Conclusion

When studying trust relative to information security, one idealistic objective is to find out how to correctly estimate trust, because it would be a powerful tool when operating in a distributed system. A good understanding of how trust relationships work in the real world is a necessary first step, and this study is meant as a contribution to that task. We have found that trust essentially is and should be based on knowledge. The next step could be to find principles to correctly assess and extract trust as a parameter from the real world. These parameters could further be integrated in formal models with the goal to optimise system performance and service quality. It is our hope that the ideas presented in this paper will inspire further work in this direction.

11 Postscript

The discussions during and after the presentation of this paper brought up two issues which are worth mentioning. Firstly that the lack of trust due to incomplete knowledge or ignorance can be viewed as information entropy, and secondly the question whether trust can be modelled as probability.

The idea of viewing uncertain trust as entropy seems interesting. Indeed, if the amount of knowledge which is possible to obtain about an entity could be determined and also was finite, the amount of ignorance would be known and the task would simply be to replace as much ignorance as possible by knowledge. This approach is illustrated in figure 7.

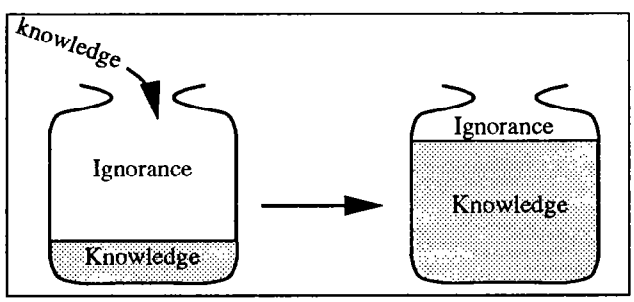

Figure 7: Over-simplistic knowledge model

Unfortunately, the amount of obtainable knowledge is impossible to determine, and it might even be infinite, so that no matter how much knowledge one acquires ignorance will always persist. The problem with applying Shannon entropy to trust is that Shannon entropy is based on statistical probability over a known universe like an alphabet or a message space. In the case of trusting a system or a human, the universe is extremely difficult to determine. Only if the universe was known, i.e. how much knowledge is obtainable, could the entropy be measured based on what one actually knows. When using the concept of entropy for the purpose of studying trust, entropy must therefore be understood in a wider sense than the purely statistical Shannon sense.

The question whether trust can be modelled as probability attracted mostly negative opinions during the discussions. In order to determine to which extent the two concepts do overlap, one first of all needs a deep understanding of both. This paper has tried to elucidate trust, but has only briefly mentioned probability. Our view is that probability always is a subjective notion, inasmuch as it is the measure of uncertainty felt by a given person facing a given event. Objective or physical probability is a meaningless notion. This view is shared by e.g. de Finetti [dF74]. In this sense, trust corresponds well with probability, in that it is a subjective belief.

However, subjective probability has the objective requirements of respecting rules of coherence such as the axioms and theorems of probability theory, and of basing its estimations on objective evidence. We have argued that trust should be based on knowledge, which can be understood as objective evidence, so in this sense, trust and probability still correspond. But on the other hand, the types of evidence used for trust may not be suitable for making probability theoretic estimations. For instance, it is hard to see how a security evaluation assurance level could be translated into something like a probability estimate. Regarding the axioms and theorems of probability theory, one example will prove that they can not be directly applied to trust.

If person A has made a good deal with a particular shopkeeper S at the market, and A then recommends the shopkeeper to person B, probability theory would

require B's final trust in the shopkeeper S to be:

$$t(B \rightarrow S) = t(B \rightarrow A) \cdot t(A \rightarrow S)$$

However, it may be that the shopkeeper does not like person B and therefore would cheat him. Anyboy with this knowledge would also know that the formula does not hold.

The problem is that probability theory does not consider the entity doing the estimations as relevant for the probability estimates, whereas it may be relevant for trust, as already mentioned in section 6. Said on other words, trust is not necessarily transitive, whereas probability is. This example does not say that probability theory can never be used to model trust, but simply that it can not be applied directly and generally. It may be possible if one puts restrictions on its use, as for instance only to apply probability theory in case of trust in rational entities where the influence of the relationship between the trusting and the threatening party can be ignored. These ideas seem interesting and should be explored further.

References

[BAN89] Michael Burrows, Martín Abadi, and Roger Needham. A logic of authentication. Technical report, DEC Systems Research Center, February 1989. Research Report 39.

[BBK94] Thomas Beth, Malte Borcherding, and Birgit Klein. Valuation of trust in open networks. In *ESORICS 94. Brighton, UK*, November 1994.

[CSNP92] E.A Campbell, R. Safavi-Naini, and P.A. Pleasants. Partial belief and probabilistic reasoning in the analysis of secure protocols. In *Proceedings. Computer Security Foundations Workshop V*, pages 84–91. IEEE Comput. Soc. Press, Los Alamitos, CA, USA, 1992.

[Den93] Dorothy Denning. A new paradigm for trusted systems. In *Proceedings 1992-1993 ACM SIGSAC New Security Paradigms Workshop*, pages 36–41, New York, NY, USA, 1993. ACM.

[dF74] Bruno de Finetti. The value of studying subjective evaluations of probability. In Carl-Axel Staël von Holstein, editor, *The concept of probability in psychological experiments*, pages 1–14, Dordrecht, Holland, 1974. D.Reidel Publishing Company.

[EC92] EC. *Information Technology Security Evaluation Criteria (ITSEC).* The European Commission, 1992.

[HHV95] Shaun P. Hargreaves Heap and Yanis Varoufakis. *Game Theory, A Critical Introduction.* Routledge, 1995.

[ISO96] ISO. *Evaluation Criteria for IT Security (Common Criteria), documents N-1401/1404.* ISO/IEC JTC1/SC 27, 1996.

[Kan] I. Kant. *Kritik der praktischen Vernunft.* 1788. trans. and ed. by Lewis W. Beck, *Critique of Practical Reason and Other Writings in Moral Philosophy.* The University of Chicago Press, 1949.

[Lap92] J.C. Laprie. *Dependability: Basic Concepts and Terminology.* Springer-Verlag, 1992.

[SM95] G.J. Simmons and C. Meadows. The role of trust in information integrity protocols. *Journal of Computer Security*, 3(1):71–84, 1995.

[USD85] USDoD. *Trusted Computer System Evaluation Criteria (TCSEC).* US Department of Defence, 1985.

[YKB93] R. Yahalom, B. Klein, and Th. Beth. Trust relationships in secure systems - a distributed authentication perspective. In *Proc. 1993 IEEE Symp. on Research in Security and Privacy*, pages 150–164, 1993.

[YKB94] Raphael Yahalom, Birgit Klein, and Thomas Beth. Trust-based navigation in distributed systems. *Computing Systems*, 7(1):45–73, Winter 1994.