The Solar Trust Model: Authentication without Limitation

Michael Clifford
Dept. of Computer Science
University of California, Davis
Davis, CA 95616
macliffo@engr.ucdavis.edu[1]

Charles Lavine
The Aerospace Corporation
2350 East El Segundo Blvd.
El Segundo, CA 90245
lavine@aero.org

Matt Bishop
Dept. of Computer Science
University of California, Davis
Davis, CA 95616
bishop@cs.ucdavis.edu

Abstract

The PEM and PGP/X.509 authentication models and the Biba Integrity Model have limitations inherent in their design that diminish their practicality in real world applications. The ICE-TEL trust model addresses some of these difficulties, and introduces a few new limitations. The Common Security Services Manager's Trust Policy Interface Specification provides the guidelines with which new trust policies may be encoded, but does not implement an actual policy. This paper describes a new model that permits both the identity of the sender of a message, and the trustworthiness of the sender of the message to be determined. The model works regardless of whether or not the message was signed by a certificate authority with which the recipient has a relationship. The model can be implemented without changing the format of certificates that are currently in use, and could be used as a module in a broader security framework, such as the Common Security Services Manager.

Introduction

This paper presents an alternative model for verifying the identity of the sender of a message that has been signed using a digital certificate [7,8]. The PGP/X.509 model [1] assumes an implicit transitivity of trust between participating parties. The PEM certification model [2] assumes that everyone in the world trusts one ultimate authority to verify the identities of other certificate senders. The PEM model also does not allow for multiple levels of trust within its certification hierarchy. Neither of these models provides a practical, universally applicable method of verifying the identity of the sender of a message. In the real world, trust is rarely transitive, and the concept of a single hierarchy has already given way to multiple corporate and governmental certifying authorities, each of which issues their certificates without being required to meet the standards of some ultimate certifying authority [5].

The Biba Integrity Model [3] provides for multiple levels of trust within an organization or system, but not between two autonomous systems. The Common Security Services Manager's Trust Policy Interface Specification [4] provides an API in which an open set of authentication policies may be implemented, and permits multiple certificate authorities (CAs) to sign the same message, but provides no method for verifying the trustworthiness of any of the CAs. The ICE-TEL trust model [9] eliminates the need for transitivity of trust, and permits certification along paths rather than in global CA hierarchies, but it does not provide for multiple levels of trust .

The model described in this paper enhances verification capabilities beyond other models by providing a method for designating many levels of trust, for permitting unlimited numbers of independent CAs with no requirement for a central authority, and for determining the trustworthiness of a message that has been signed by a CA with whom the receiver of the message has no direct relationship [5, 9, 10].

Background

The PEM certification model is described in RFC1422 [2]. It describes a certification hierarchy in which one root level authority, called the Internet Policy Registration Authority (IPRA), certifies a small number of lower level authorities. Each of these lower level authorities, known as Policy Certification Authorities or PCAs, can certify even lower level authorities known as Certification Authorities or CAs. CAs can then certify individuals. Certification requirements are passed down to lower level authorities, and can become increasingly restrictive as each authority

[1] Portions of this work were performed while the author was at The Aerospace Corporation.

0-8186-8789-4/98 $10.00 © 1998 IEEE

adds new restrictions onto those imposed upon it from higher level authorities. In this model, the transitive application of trust from higher level authorities to lower level authorities, and the dependence upon one root level authority, imply that all users of the model must implicitly place their trust and confidence in the certification policies of the IPRA. If a user either can not, or chooses not to accept the ultimate authority of the IPRA then, because of the transitivity of trust inherent in the PEM model, the user can not accept as valid any certificate signed by a CA under the IPRA.

The PGP/X.509 Web of Trust model derives from the ITU X.509 Authentication Framework's strong authentication procedure [1]. Messages are digitally signed using public key encryption. If a user does not know the sender of a message, they can find someone who does know the sender, either directly, or though one or more other people. Each person in the chain verifies the identity of the next person in the chain. Because there is no higher level certification authority, users are forced to trust the identity of the sender at position n in a chain, simply because the next closest person to them (at position *n-1*) says that person *n* can be trusted. The "web of trust" can verify the identity of a sender only if the recipient and the sender can be connected to each other through a chain of people who know each other. This requirement greatly limits its functionality.

In addition, the PGP/X.509 model does not offer any way to compute more than one level of trust. Users are forced to make a judgment as to how much they trust messages signed by the people that they know, and have no guidance in determining how much they can trust messages signed by someone elsewhere on the web of trust.

The Biba Integrity Model uses the concept of integrity labels to provide an ordering of data objects, and the subjects that may try to read from or write to the data objects [3]. Two properties, the Simple Integrity Property, and the Integrity *-Property, are used to determine the subjects that are permitted to read from, or write to, each object. The Simple Integrity Property permits a subject to write to an object only if the subject has an equal or higher integrity label than the object. The Integrity *-Property prevents a subject with read access to an object from modifying an object with a higher integrity label than the object that it can read from. While the Biba Integrity Model does provide a form of trust ordering if all of the subjects and objects involved have been assigned integrity labels within the same organization or system, it fails to create orderings of trust for subjects and objects that exist within completely autonomous systems. Subjects that have been labeled with an integrity level in one system have no basis for being labeled with any particular integrity label in another system. If subjects in one system were to be labeled at the lowest integrity label in another system by default, then subjects would be denied the ability to even read data in other systems. This prevents people in two independent organizations from being able to share data easily.

If, on the other hand, subjects were labeled with an integrity label in another system based on the integrity level that they have been assigned within their own system, then the two organizations must have some method of implicitly ensuring that they label subjects with integrity labels in exactly the same way within both of their organizations (or provide a mapping between labels). If the two organizations do this, then they would also have to agree that subjects at the same integrity label within one organization would have the same read and write privileges within the other organization. If these requirements are not met, then the notion of levels of integrity is rendered meaningless between separate systems or organizations.

The Common Security Services Manager's Trust Policy Interface Specification is an API released by the Intel corporation, that provides a wide variety of functions and services that support applications that require security features in their implementation [4]. Security policy modules can be plugged into a program using the API, and then used to calculate the level of trust that can be assigned to a message. Signatures by multiple certificate authorities can be handled by the API. However, the API does not specify a method of determining whether or not a CA with which the user has no relationship can be trusted. In addition, even though it provides functions with which trust policies can be built, it does not specify any specific procedure for implementing a trust policy.

The European Union's ICE-TEL project addresses many of the shortcomings in the other security models by combining some of the best features of some of the other models [4]. In the ICE-TEL model, users determine if they can trust a message that is signed by a particular CA. A message sent to a user from certificate authorities other than their own is signed first by the sender's CA, then by a chain of intermediate CAs until the message reaches the user's CA. The user's CA signs the message, and delivers it to the user. A user can verify the identity of the sender of a message by walking through the chain of signatures on the message, and verifying that each signature is valid. Cross certification is used to promote efficiency in the implementation of the model. Users are grouped into security domains, that are designed to aid the users in determining how the model should be applied to a particular message.

The ICE-TEL model provides a solution to some of the most significant problems with the other security models. Users are given the ability to trust messages that have not been signed by their own CA. The ICE-TEL model eliminates the transitivity of trust problem found in the PEM model. Additionally, a single CA hierarchy is not required for the ICE-TEL model to function, although hierarchies are present in the model.

The major drawback in the design of the ICE-TEL model is that it provides users with a flat model of trust.

When a certification path is considered trusted, it is trusted to the same degree as any other certification path. ICE-TEL provides users with no method of determining whether one certification path might be worthy of a higher degree of trust than another path. Although ICE-TEL does provide the ability to limit the length of a certification path, this only addresses the issue of how many CAs are permitted to sign a particular message, and not how differences among the CAs themselves might affect the trustworthiness of the message.

This presence of multiple levels of trust is an extremely important property, because it permits the computation of a most trusted or secure path for a message to take, and because it permits comparisons of the authenticity of data from multiple sources, or from the same source along multiple paths.. Without this ability, the ICE-TEL model can not determine whether or not a message should be trusted in a truly flexible manner. Instead, it must resort to a somewhat inflexible and complex array of cross certifying hierarchies and complicated algorithms that have limited efficiency and versatility.

A New Approach: The Solar Trust Model

The Solar Trust Model overcomes many of the limitations inherent in the designs of the other trust models. It does this by providing a simple and efficient method by which many levels of trust can be implemented, by permitting an unlimited number of independent CAs with no requirement for a central PCA, and by defining a procedure for determining the trustworthiness of a message that has been signed by a CA with whom the receiver of the message has no direct relationship.

To demonstrate how the Solar Trust Model works, let a CA be defined as any entity which issues digital certificates. If CA_1 and CA_2 are two certificate authorities, then CA_2 can establish a set of rules to determine how much it trusts messages signed by CA_1. This set of rules is called a trust relationship.

For example, suppose that Bob wants to read a document sent by Alice and signed by CA_1, which uses a fixed procedure to ensure Alice's identity. Now, if CA_1 can prove to CA_2 that the procedure used to verify Alice's identity meets CA_2's criteria, then CA_2 can be certain (i.e., to an adequate level of certainty) that Alice's document (which has been signed by CA_1) is indeed Alice's. On the other hand, if CA_3 is another certificate authority, and its policy for verifying the identity of a second person, say Ted, is unknown to CA_2, and Ted is sending a message to Bob, then CA_2 will not have adequate assurance to believe any claims about the identity of Ted made by CA_3.

If CA_2 has a set of rules that say that any certificate authority that uses the same procedures as itself for verifying the identity of the sender of a message can be trusted more than a certificate authority that does not use those procedures, and if CA_2 can verify that CA_1 uses these procedures, then CA_2 can say that it has a stronger trust relationship with CA_1 than with CA_3.

Let a solar system be defined as the representation of an ordering of trust relationships with respect to a specific certificate authority. It is helpful to think of a solar system as a series of objects that exist within concentric orbits around a central body, much as the planets in a solar system orbit around the sun. For any set of certificate authorities CA_1 through CA_n, CA_i is the central body or primary in its own solar system, and all of the other certificate authorities with which CA_i has established a trust relationship are objects or planets in orbit around the primary.

An ordering of trust can now be established for all certificate authorities which are planets in a solar system, with distance from the solar system's primary indicating the level of the trust relationship between the primary and a planet. A certificate authority places itself in the 0'th orbit of itself, because it trusts itself completely. Orbits 1 through n are occupied by all other certificate authorities in the solar system, where a certificate authority in orbit m is more trusted than a certificate authority in orbit m+1. It is possible for two or more certificate authorities to share the same orbit, or for orbits to be empty. If certificate authority CA_i does not have a trust relationship with certificate authority CA_x, then CA_x is not a planet in CA_i's solar system.

Note that since every certificate authority has its own solar system, the primary certificate authority in one solar system can (and often will) be a planet in another solar system, and that a certificate authority can be a planet in many different solar systems. It is also important to recognize that two certificate authorities do not necessarily have the same trust relationship with each other. Since a trust relationship is derived from a set of rules which each certificate authority independently establishes (although common rule sets can be established), there is no guarantee that two certificate authorities will ever have the same trust relationship with each other. Furthermore, if one certificate authority has a trust relationship with a second certificate authority, then it is not guaranteed nor is it necessary that the second certificate authority has a trust relationship with the first.

Although a certificate authority can establish direct trust relationships with many other certificate authorities, it is infeasible that it will establish such relationships with **all** other certificate authorities. The solar trust model solves this problem by establishing *indirect trust relationships*. For example, if CA_2 is a planet in the solar system of CA_1, and CA_3 is a planet in the solar system of CA_2, then CA_3 is a "*moon*" of CA_2 in the solar system of CA_1. This can be extended through any number of iterations. Since a certificate authority can be a planet in many different solar systems, that certificate authority can be a moon of many different certificate authorities in the same solar system. Although the moon of a planet is regarded as being in the same orbit as the planet, the moon is not considered to be the same entity as the planet.

It is important to understand that an indirect trust relationship does not imply a transitivity of trust. When trust is transitive, then if CA_1 trusts CA_2, and CA_2 trusts CA_3, then CA_1 must trust CA_3 in the same way that CA_2 trusts CA_3. In an indirect trust relationship, if CA_1 trusts CA_2, and CA_2 trusts CA_3, then CA_1 may or may not trust CA_3. Furthermore, since CA_1 relates to CA_3 indirectly, CA_1 is unlikely to trust CA_3 as much as it does CA_2.

At this point, it should be noted that if a message comes from a moon, it may appear to come from many different orbits. To resolve this issue, we define the *path of trust* taken by a message as the set of certificate authorities that sign the message in the order in which they are signed. For example, if a message is signed first by CA_3, then by CA_2, then by CA_1, the path taken by the message is CA_3, CA_2, CA_1. Given any two certificate authorities with an indirect trust relationship, it is likely that there is more than one path that a message could take between the two certificate authorities. However, the only path that counts is the one that the message actually takes. If a CA appears more than once on the same path of trust, it is regarded as a different CA each time that it appears.

In order to improve efficiency, it may be desirable for paths of trust between CAs to be computed in advance. There are several methods by which this may be implemented. In the first method, a three way handshake is used to send a trusted path to a CA that sends a message. For example, if CA_2 wished to send a message to CA_1, CA_2 would send a request for a trusted path to CA_1. CA_1 would then send an acceptable path of trust back to CA_2. CA_2 would then send its message to CA_1 along the path of trust. Note that CA_1 does not have to believe the origin of the path request from CA_2, since CA_1 can determine whether or not the final message came from CA_2, and can determine whether or not it trusts messages from CA_2. Another method would involve the computation of trust tables, which would be similar in form to the routing tables used in IP protocol routers. Finally the ICE-TEL trust model proposes publishing paths using public forums. [9]

In addition to establishing the orbit from which a message derives, the concept of a path of trust also allows the determination of the levels of trust for messages that are sent between certificate authorities that do not have direct relationships. When a message is first signed by a certificate authority, that certificate authority can attach a copy of its rule set to the message. As the message is passed from certificate authority to certificate authority, each certificate authority concatenates its rule set to the rule set that is passed to it, forming a *composite rule set.* The certificate authority that ultimately receives the message applies all of the rules in the composite rule set, until the message is either rejected as untrustworthy, or is accepted after meeting the requirements of all of the rules. The rule set for each individual certificate authority is represented using the Solar Trust Model Rule Set Header shown below.

Field A: Local Direct Range
Field B: Local Indirect Range *(m)*
Field C: Maximum Path Length *(p)*
Field D: Permitted Local Range for Next CA *(q)*
DATA

Figure 1. The Solar Trust Model Rule Set Header.

The following fields are represented in the Solar Trust Model Rule Set Header:

- *Field A*: Local Direct Range: Trust all messages that have been directly signed by a CA in an orbit with a number (*k*) no greater than this value.

- *Field B*: Local Indirect Range: Trust messages that have been indirectly signed by a CA in an orbit with a number that is less than or equal to this value. Do not trust any messages that have been signed by a CA in an orbit with a number that is greater than this value.

- *Field C*: Maximum Path Length: A message can be trusted only if the total number of CAs that have signed the message is no greater than this value *(p)*.

- *Field D*: Permitted Local Range for Next CA: A message that has been indirectly signed by a CA inside the local indirect range can be trusted only if it came from an orbit in that CA's system with a number no
- greater than this value *(q)*.

The examples that follow demonstrate the major classes of rules from which certificate authorities can derive their own rule sets. The figures beneath each new case show the order and rules whereby each new certificate authority adds its own signature to the data that it receives.

Case 1: Direct transmission through one certificate authority

In this case, a message is sent between two individuals who use the same certificate authority, and the certificate

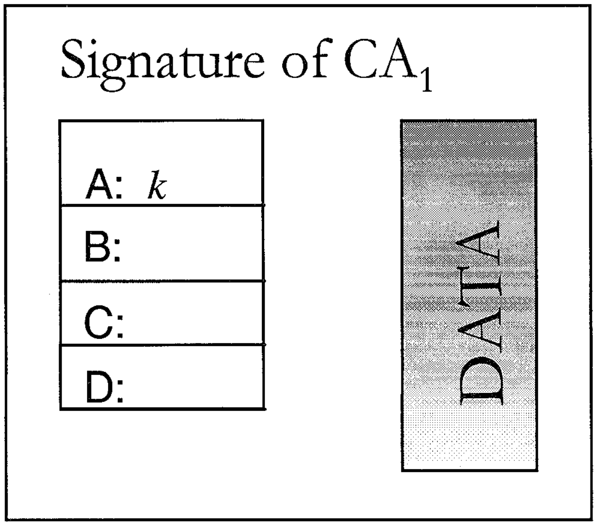


Figure 2. Format of a message sent between two users with the same certificate authority. CA$_1$ guarantees the identity of the sender of the message.

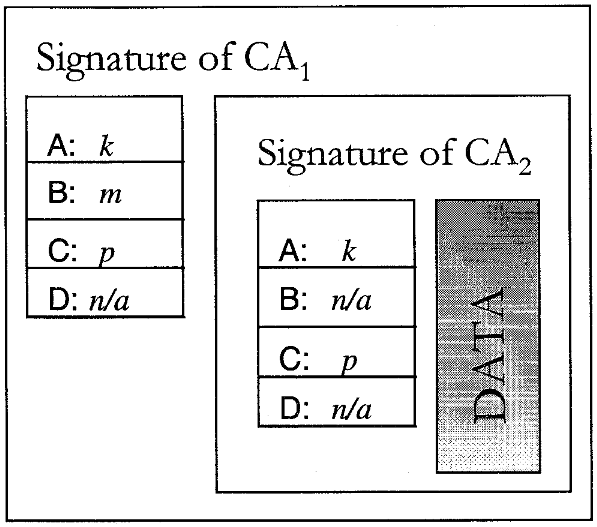


Figure 3. Format of a message sent between two users with different certificate authorities who maintain a direct trust relationship..

authority guarantees the identity of the message sender. (see Figure 2).

Case 2: Direct transmission within a solar system

In this case, a message is first directly signed by a certificate authority CA$_2$ which is a planet in the solar system of CA$_1$, and is then signed by CA$_1$. CA$_2$ might be located in any orbit in CA$_1$'s solar system. In this case, the policy of CA$_1$ is directly applied. (see Figure 4). The possible policies are:

a) Trust only transmissions that are directly signed by CA$_1$ (Case 1).

b) Trust transmissions that are directly signed by any CA in the solar system, but only if they originate within some range of orbits between orbit 0 and some orbit m. Trust transmissions that are directly signed by any CA in the solar system. As long as CA$_2$ is within one of the orbits permitted by CA$_1$'s policy, the receiver can trust the message. Otherwise, the receiver can not trust the message.

Case 3: Indirect transmission through multiple certificate authorities

Cases 1 and 2 assume that a message was directly signed by some certificate authority with a direct relationship to another certificate authority. In this case, an indirect relationship between certificate authorities is demonstrated. Additional policies are needed to handle messages that are signed by a certificate authority that CA$_i$, orbiting in the solar system of CA$_2$, does not have a direct relationship. These policies are designed to interpret the composite policies of other certificate authorities in order to achieve an appropriate composite rule set (see Figure 6). Examples of possible policies for this case include:

a) Trust only transmissions which were trusted under Case 2.

b) Trust transmissions which are trusted under Case 2, or which come from another CA's solar system, so long as that CA is trusted by CA$_i$ under the rules in Case 2.

c) Trust transmissions that are trusted under Case 2, or that come from another CA's solar system, but only if they originate within some orbit, say orbit 3, in the other solar system.

d) Trust transmissions that are trusted under Case 2, or that have been signed by not more than four other certificate authorities.

e) Trust transmissions which pass through a CA in CA$_i$'s solar system, so long as that CA falls within orbit 4 of CA$_i$'s solar system.

f) Trust transmissions which pass through some certificate authority CA$_j$ which lies within orbit 4 of CA$_i$'s solar system, so long as the transmissions are trusted according to the policy of CA$_j$.

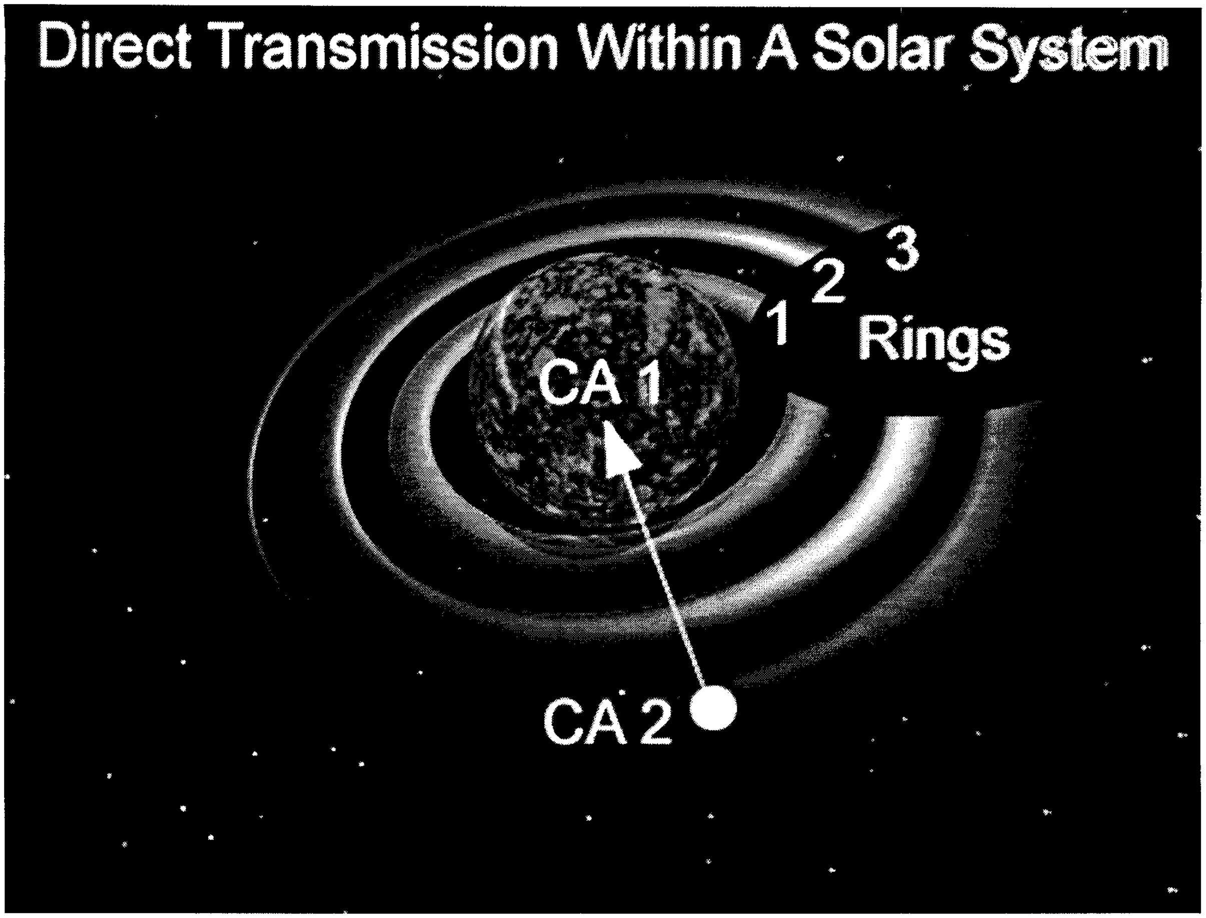
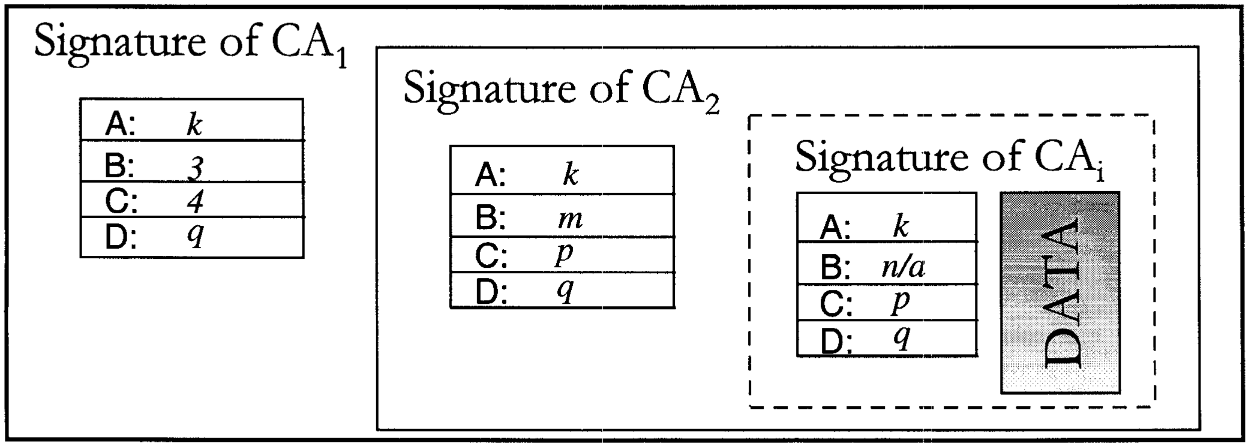


Figure 4. The solar system of CA$_1$. Ring 0 (CA$_1$ itself), plus the three next innermost rings are shown. CA$_2$ is located in ring 3 of CA$_1$'s solar system. The arrow indicates that a direct path is taken by messages sent from CA$_2$ to CA$_1$.



Signature of CA$_1$

A:	*k*
B:	*3*
C:	*4*
D:	*q*

Signature of CA$_2$

A:	*k*
B:	*m*
C:	*p*
D:	*q*

Signature of CA$_i$

A:	*k*
B:	*n/a*
C:	*p*
D:	*q*

Figure 5. Format of a message sent between two users with different certificate authorities that have an indirect trust relationship. CA$_1$'s rule set header requires that the message be signed by a CA in an orbit less than or equal to 3, and that the length of the path taken by the message be no greater than 4.

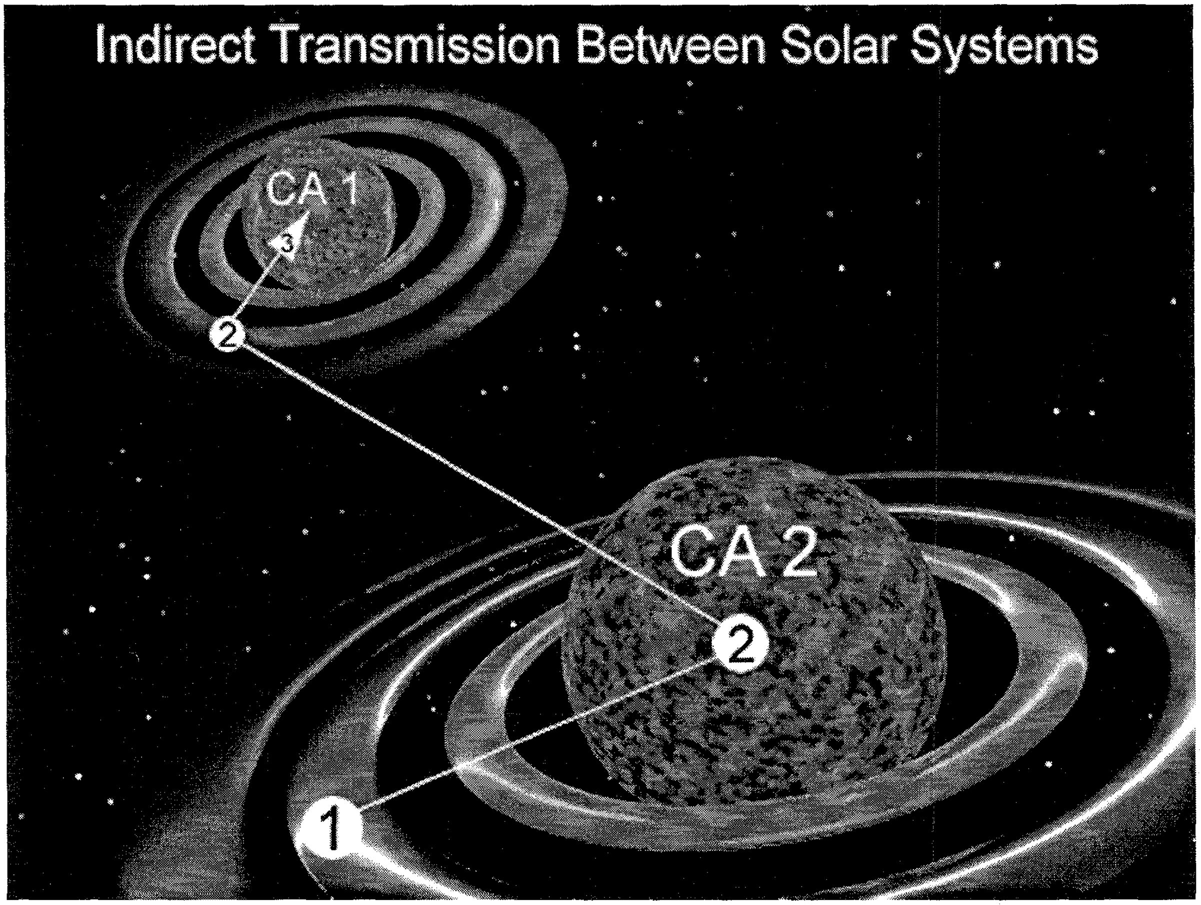


Figure 6. Path taken by a message signed by a certificate authority (1), in ring 2 of CA$_2$ (2), with an indirect trust relationship with CA$_1$ (3). CA$_2$ is in ring 2 of CA$_1$.

Conclusion

The Solar Trust Model addresses the shortcomings of the PEM, PGP/X.509 and ICE_TEL authentication models. It eliminates the need for a hierarchy of certificate authorities, while removing the limitations inherent in the design of the web of trust, and integrating dynamically computed levels and orderings of trust.

Rule set data can be added to a message just before the message is signed by a certificate authority, and can be stripped off and processed by the final recipient certificate authority. This ensures that a certificate authority signs its own rule sets, and provides for an easy method for the transmission of rule sets to the final certificate authority. In addition, it should be reasonably simple to implement software to process rule set data as a module inside of an API such as the Common Security Services Manager's Trust Policy Interface. Digital signatures and rule set data can be wrapped around any message, including other certificates, thus permitting the Solar Trust Model to be implemented on top of existing authentication services. Therefore, the Solar Trust Model can be implemented with no changes to existing hardware, and very minimal additions to existing software. Furthermore, it can be shown that the orbital structure of the Solar Trust Model can be reduced to a directed graph structure, with nodes on the graph representing certificate authorities, and lines on the graph representing trust relationships.

Both the Solar Trust model and Biba's model provide a partial ordering of a set of trust levels. The critical difference is that Biba does not specify any partial ordering of trust levels among different autonomous domains. The Solar Model allows such an ordering and, indeed, represents the ordering as a path among components of a solar system. Hence the Solar Trust Model is a generalization of Biba's model and can be used in contexts other than certification.

Future research directions include the implementation of the Solar Trust Model as a module in an API, and as an independent software application. Additionally, a detailed comparison between the Solar Trust Model and other security models will be performed using formal methodology to verify the correctness of the model. An analysis to determine the most efficient method for distributing trusted paths in advance of message transmissions will also be carried out.

References

[1] International Telecommunication Union. "Information Technology - Open Systems Interconnection - The Directory: Authentication Framework," *ITU-T Recommendation X.509*, pages 7-17. 1995.

[2] Kent, S. "Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management." *Internet Report*, RFC 1422, February 1993.

[3] Biba, K. "Integrity Considerations for Secure Computer Systems." *U.S. Air Force Electronic Systems Division Technical Report* 760372, 1977.

[4] Intel Corporation. "Common Security Services Manager Trust Policy Interface (TPI) Specification Draft for Release 1.2", page 7, March 1997.

[5] L. Lpez, J. Carracedo. "Hierarchical Organization of Certification Authorities for Secure Environments." Proceedings of the Symposium on Network and Distributed System Security, Februrary, 1997, San Diego, CA.

[6] American National Standard Institute. "American National Standard X9.57. Public Key Cryptography for the Financial Services Industry: Certificate Management". 1996.

[7] R. Rivest, A. Shamir, and L. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". Comm. of the ACM, Vol. 21, No. 2 (pp. 120-126), Feb. 1978.

[8] Birrell, A., Lampson, B., Needham, R. and Schroeder, M. "A Global Authentication Service without Global Trust". Proceedings of the 1986 IEEE Symposium on Security and Privacy, April 1986.

[9] Young, A. Cicovic, N.K. and Chadwick, D. "Trust models in ICE-TEL". Proceedings. 1997 Symposium on Network and Distributed System Security. (pp. 122-133), Feb. 1997.

[10] Mendes, S. and Huitema, C. "A New Approach to the X.509 Framework: Allowing a Global Authentication Infrastructure without a Global Trust Model". Symposium on Network and Distributed System Security, Feb. 1995.