Trust Negotiation in Electronic Markets

Todd Barlow, Adam Hess, Kent E. Seamons

Internet Security Research Lab
Computer Science Department, Brigham Young University
Provo, Utah, USA 84602
seamons@cs.byu.edu

Abstract

As business transactions migrate into electronic marketplaces, most interactions will occur between strangers. In order for strangers to conduct secure transactions, a sufficient level of mutual trust must be established. A new approach to gradually establishing trust between strangers is through the iterative exchange of digital credentials, known as *trust negotiation.* This paper briefly describes TrustBuilder, an architecture for automated trust negotiation that we are designing and developing. It also introduces client-initiated trust establishment, a new context for trust negotiation that has not been explored previously. When a client sends a request to a server, the request can sometimes contain sensitive content that the client must safeguard. The client must establish trust in the server before the request is sent. This paper presents the design of an architecture for client-initiated trust establishment and describes how the architecture can be used to address privacy concerns, a significant impediment to conducting on-line business transactions.

The copyright of this paper belongs to the paper's authors. Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage.

Proceedings of the Eighth Research Symposium on Emerging Electronic Markets (RSEEM 01)

Maastricht, The Netherlands, September 16-18, 2001
(M. Schoop, R. Walczuch, eds.)

http://www-i5.informatik.rwth-aachen.de/conf/rseem2001/

1 Introduction

The Internet enables a global electronic marketplace. Brick-and-mortar businesses that once serviced local customers now offer electronic services internationally. This dramatic shift in scope introduces a new challenge: how to establish trust during electronic business transactions. Our research addresses the problem of how strangers enter new business relationships and conduct secure business transactions electronically with a trust similar to traditional face-to-face business interactions.

In the physical world, physical presence, reputation, and accountability of a business all contribute to customers trusting it. When a business is completely unfamiliar to a customer, trusted third parties can play a key role in engendering consumer trust. Paper credentials, such as a membership certificate from the Better Business Bureau or International Chamber of Commerce, are prominently displayed to promote consumer confidence.

As business transactions migrate into electronic marketplaces, most interactions will occur between strangers, because the billions of Internet users do not share a common security domain. In order for strangers to conduct secure transactions, a sufficient level of mutual trust must be established. For this purpose, the *identity* of the participants (e.g., their national ID number, fingerprint, institutional tax ID) will often be irrelevant to determining whether or not they should be trusted. Instead, other *attributes* of the participants will be more relevant, such as employment status, citizenship, age, credit ratings, accreditations, certifications, memberships, and licenses.

Traditional security approaches based on identity require a new client to pre-register with the electronic market, in order to obtain a local login, capability, or credential before requesting service; but trust establishment is still an issue. For example, today the administrative effort in arranging electronic document interchange (EDI) of sensitive business data between companies and arranging on-line auctions of commodities is centered on lengthy, out-of-band trust establishment procedures to verify attributes of the participants. E-commerce needs a more scalable approach to allow automatic, on-line pre-registration or eliminate the need for pre-registration and out-of-band, manual verifications. Automated trust establishment is a potential solution to streamline these processes.

With automated trust establishment, strangers build trust by exchanging digital credentials, the on-line analogues of paper credentials that people carry in their wallets. Digital credentials are digitally signed assertions by a credential issuer about the credential owner. A credential is signed using the issuer's private key and can be verified using the issuer's public key. A credential describes one or more attributes of the owner, using attribute name/value pairs to describe properties of the owner asserted by the issuer. Each credential also contains the public key of the credential owner. The owner can use the corresponding private key to answer challenges or otherwise demonstrate ownership of the credential. Digital credentials can be implemented using, for example, X.509 certificates [X509, 1997].

Automated trust establishment between strangers promises to extend trusted interactions to a much broader range of participants than traditional security approaches based on identity and capabilities. With automated trust establishment between strangers, the number of sensitive

business processes accomplished electronically will grow substantially, leading to more efficient markets and reduced costs of doing business over time.

The remainder of this paper is organized as follows: Section 2 describes trust negotiation and the TrustBuilder architecture for trust negotiation, and provides some real-world examples of a trust negotiation. Section 3 discusses privacy, a vital issue to individuals and corporations operating on-line. Privacy is one significant example that motivates the need for client-initiated trust establishment, outlined in section 4. Section 5 contains related work, and section 6 provides conclusions and future work.

2 Trust Negotiation

While some resources are accessible to all, many require protection from unauthorized access. Access control policies can be used for a wide variety of "protected" resources, such as services accessed through URLs, roles in role-based access control systems, and capabilities in capability-based systems. Since digital credentials themselves can contain sensitive information, their disclosure will often also be governed by access control policies. For example, access to a credential containing medical information could be restricted to primary care physicians and HMO staff. Access to a credit card credential could be limited to businesses authorized to accept credit cards and committed to adhere to guidelines governing disclosure of credit card numbers.

When businesses and consumers both possess digital credentials, a new approach to establishing trust gradually between strangers is through an iterative exchange of digital credentials known as a *trust negotiation* [Seamons, 2001][Winsborough, 2000][Yu, 2001]. We are designing and developing the TrustBuilder architecture (see figure 1) to support automated trust negotiation between strangers. A security agent manages credentials and access control policies that govern access to sensitive resources, credentials, and access control policies. Access control policies specify the combination of credentials that must be disclosed during a negotiation in order to establish sufficient trust to be granted access to a sensitive resource, such as a credential, access control policy, or service. The architecture supports the disclosure of access control policies themselves in order for strangers to learn about the security requirements of their counterpart during a negotiation and to determine in private whether they possess the credentials required by their counterpart. Like credentials, access control policies might contain sensitive information that should not be disclosed to a stranger.

The following is an example of trust negotiation in TrustBuilder. Assume that a client must disclose personal information while making an on-line purchase request. The client has previously established a policy to disclose personal information to a server only when the server adheres to certain privacy practices. The TrustBuilder client's security agent sends an access control policy to the server, requesting a credential from the server that specifies its privacy practices. The client does not forward the purchase request to the server until the desired trust has been established. Once the server receives the policy from the client requesting a server credential, the server discloses the credential satisfying that policy. Now

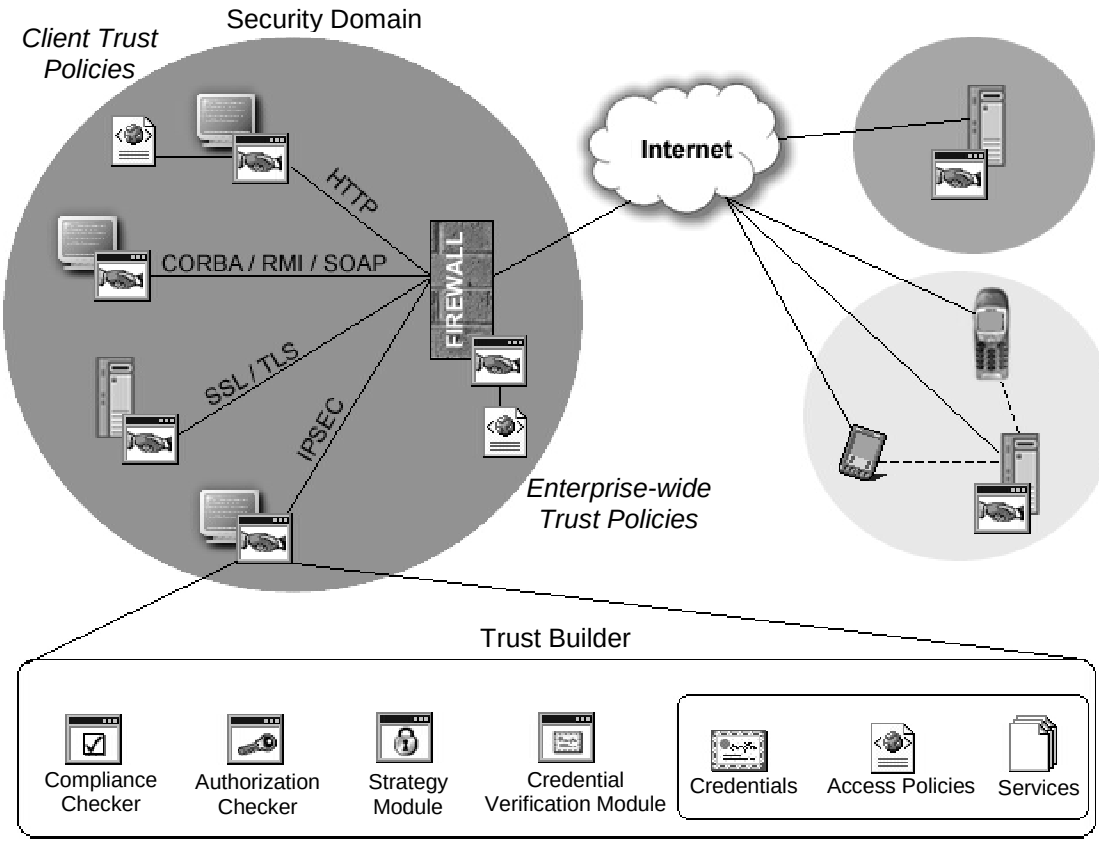

Figure 1. The TrustBuilder trust negotiation architecture for establishing trust between strangers. Using TrustBuilder, negotiation participants exchange digital credentials to gradually establish trust in order to conduct sensitive business transactions.

the client can safely forward the purchase request to the server. After receiving the purchase request, the server determines the client is requesting a membership discount. The server then requests a membership credential from the client to assure that the client is entitled to the discount before the server processes the purchase request. In this example, both the client and the server establish trust in each other. Although the credentials exchanged during this simple example negotiation are not sensitive, TrustBuilder also has support for sensitive credentials and policies.

The following is an example of trust negotiation in TrustBuilder involving sensitive credentials. Suppose an individual is making an airline reservation on-line and the server requires a credit card credential and a proof of identity credential, such as a passport or driver's license, in order to process the reservation. When the client initially requests the reservation, the server issues a counter-request for a digital credit card and ID. The client considers those credentials to be sensitive and will only disclose them to a server that is

certified by TRUSTe. The client requests a TRUSTe credential from the server. Next, the server discloses a TRUSTe credential that is satisfactory to the client. Now, the client can safely disclose a credit card credential and an ID credential to the server. Finally, the server completes the reservation.

3 Privacy Issues

Trust negotiation is based on the premise that credentials and credential disclosure policies may contain sensitive information that must be kept private from negotiation counterparts until they demonstrate that they are qualified to have that sensitive information disclosed to them. In a global electronic marketplace, the handling of private information is a key concern for business and consumers. According to Forrester Research, only six percent of North Americans have a high level of confidence in the way Internet sites manage personal information [Forrester Research, 2001]. In 1999, it was estimated that privacy issues would cost on-line commerce $18 billion in potential sales by 2002, unless something were done [Sandeep, 1999].

Privacy concerns cause some users to operate only under complete anonymity or to avoid going on-line altogether. Without adequate privacy protection, the most sensitive on-line transactions will be inhibited. Since no single global approach addresses privacy concerns today, electronic marketplaces must be flexible to handle varying legal and social requirements pertaining to privacy. Participation in a global economy will require that businesses and consumers be in a position to verify and prove adherence to current privacy laws and practices.

Internet users want control over what information they disclose to web sites, and they want assurance about what those sites will do with their personal information. However, control over personal information on-line is handled in an ad hoc fashion today. In Europe, governments have taken an active role in legislation that protects individual privacy. The European Union (EU) directive on data privacy [European Union, 2001] restricts the exchange of private information between any EU country and any non-EU country. This directive led to the development of the Safe Harbor Framework [Safe Harbor, 2001], a set of guidelines to govern privacy policies of companies outside the EU.

Meanwhile, the current approach to addressing privacy concerns in the United States is through industry self-regulation rather than government regulations. The self-regulatory approach to privacy has resulted in a number of companies that encourage suitable privacy practices through privacy seal programs. For example, TRUSTe [TRUSTe, 2001] requires a web site to publish and adhere to a privacy policy, and in return, the site can display the TRUSTe seal of approval, intended to generate trust with potential clients. Human users can visually recognize the logo and can read the associated privacy policy. Perhaps many users never bother to read and understand the policy. The mere presence of the logo and privacy policy link may alleviate privacy concerns so that users will interact with the server. However, the logo itself is susceptible to forgery, and the logo's presence says nothing concerning the details of the web site's privacy policy.

On-line customers need more than simple assurance that a web site has a privacy policy. For more sensitive interactions, clients will need to verify *automatically* that Internet sites agree to follow certain procedures that meet the client's needs or that Internet sites are certified to adhere to standardized, well-known privacy practices and procedures. The trust negotiation technology in TrustBuilder can be used to create a more systematic approach to handling privacy on the web. For instance, standardized credentials can be introduced and automatically verified to adhere to specific privacy practices, such as a credential indicating a web site has an opt-in procedure for clients to consent to disclosure of their personal information to a third party.

4 Client-initiated Trust Establishment

A client's initial request to a service provider sometimes includes sensitive information in the body of the request message, such as credit card numbers, passport numbers, travel destinations, monetary amounts, or medical information. Clients often want to be assured that a service provider is authorized to receive sensitive content and is committed to handling the content appropriately according to standardized policies that the client or the client's institution establishes. Clients need assurance prior to the server receiving the sensitive information. Thus, clients must proactively establish trust based on the content of a request before that request is delivered to the server.

Trust negotiation capabilities must be expanded to a broader context than in the past in order to handle privacy requirements for sensitive content effectively. Previous work in trust negotiation dealt only with servers who initiate trust negotiations in response to a client's attempt to access a sensitive service without providing the requisite credentials. During the negotiation, either party could attempt to establish trust in the other party prior to disclosing a credential to them. Thus, clients could establish trust in servers only in the context of credential disclosure during a trust negotiation. Servers could establish trust in the client for other purposes in addition to credential disclosure, but only in response to a request for service, never proactively.

The need for client-initiated trust establishment for sensitive content goes beyond the context of a web browser contacting an unfamiliar web server, to include any process that pushes sensitive content to an unfamiliar destination. For instance, web servers that push sensitive content to web caches or any point-of-presence on the Internet will have similar trust requirements. An automated solution to client-initiated trust establishment, rather than a manual solution, will be advantageous for potentially sensitive content that must be distributed rapidly to meet business needs.

Prior advances in trust negotiation offer a partial solution to the problem of client-initiated trust establishment. For example, languages for expressing access control policies and developments in negotiation strategies can all be applied in this new context. However, earlier work in trust negotiation has not addressed two aspects of the problem. First is how to determine the policy that specifies the trust requirements associated with sensitive content. This determination may need to be made dynamically at runtime when the content is

generated, instead of relying on static associations between policy and content made in advance by security administrators. Second is how to determine the appropriate trust negotiation representative associated with the service that will receive the sensitive content, so that trust can be legitimately established prior to sending the sensitive service request.

4.1 Dynamic Policy Association

Before anyone can access a sensitive service or credential, a user or administrator creates an access control policy to protect it. The association between the policy and the sensitive resource is static. Managing static associations at runtime is straightforward. However, some sensitive content may not exist until runtime, such as when a web-based purchase order is created just before invoking the purchasing service. The client's security agent that must proactively establish trust may need to dynamically associate a policy with the content at runtime. It cannot rely on static policy associations established a priori, but must create the association on demand. We call this a content-based access control policy.

Determining the appropriate content-based policy for establishing trust in a server must be under client control. The appropriate policy cannot be statically associated with the server or service and be delivered to the client on demand before a service request is made, because different clients may demand different assurances from the same server. For example, with respect to privacy policies, clients in the United States will be interested in knowing if servers adhere to self-regulating guidelines such as TRUSTe or BBB*OnLine*. European clients will be interested in following EU privacy directives and should divulge private information only to servers that adhere to the principles of Safe Harbor [Safe Harbor, 2001].

A run-time architecture that provides dynamic policy association for sensitive content is shown in figure 2. The architecture allows a client to determine and enforce a content-based access control policy before forwarding a service request containing sensitive information. A content analysis engine inspects out-bound content to map it to a content disclosure policy enforced by the local security agent before the content is disclosed.

The content analysis engine must be able to handle many types of data to be useful in practice. In object-oriented systems such as Common Object Request Broker Architecture (CORBA) [OMG, 1999], Enterprise JavaBeans (EJB) [Monson-Haefel, 2000], or Java, the object model can be leveraged to determine the semantics of content. Content-based policies can be mapped to sensitive data types, such as a credit card object type, allowing content disclosure policies to be associated with application-level objects. The analysis engine can map content to appropriate security policies based on the type or value of the content. For instance, if the value of the age object indicates an individual is a minor, then a trust policy appropriate to minors can be enforced automatically. Structured data formats, such as XML, are suitable for reliable and efficient policy association if the semantics of the data are captured in the structured data representation. For instance, XML data tags and HTTP POST messages can capture semantic information in tags or labels.

An important design consideration for client-initiated trust establishment is where to position the content analysis engine in the trust negotiation architecture. CORBA supports

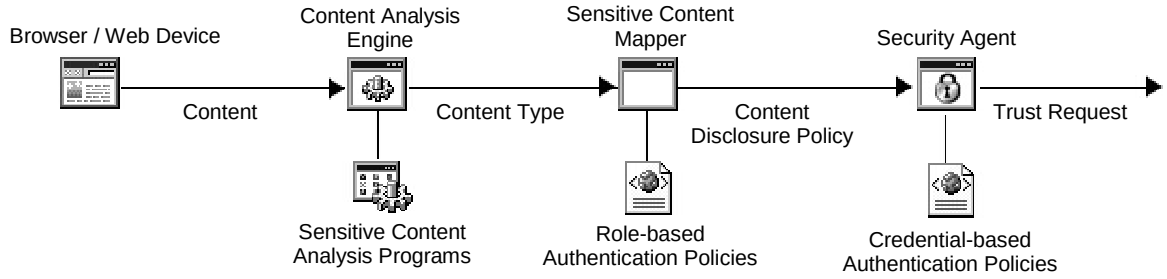

Figure 2. An architecture for determining dynamic content-based access control policies for sensitive content during client-initiated trust establishment. When a request for service contains sensitive content, the client enforces the policy before the sensitive content is disclosed.

interceptors [OMG, 1999] that allow security to be transparently inserted into the dynamic method invocation process. Using interceptor technology, a CORBA Object Request Broker (ORB) that supports dynamic method invocation could identify sensitive method arguments at the client side and establish trust in a CORBA server before invoking a remote method. A browser plug-in could offer similar services to web browser clients.

A proxy server on the client side could intercept client requests and determine if the content is sensitive so that additional trust in the server can be established before forwarding the request. A proxy server can be configured to sit at the edge of the network inside a firewall and provide consistent, mandatory content disclosure policy association for an entire organization. This eases administrative overhead, because it does not rely on clients configuring their own local environment or require the installation of additional software on each machine. It also permits transparent interception of all outgoing requests, with potential to dramatically improve the control that organizations and households have over their sensitive content. However, the overhead of examining each request could be prohibitive. A browser plug-in offers similar functionality, but at the individual browser level. This would allow individual users fine-grained control over their personal content at the cost of more administrative overhead.

The trust establishment architecture for sensitive content must be highly scalable in terms of administration and runtime performance. To provide administrative scalability, the architecture will leverage a role-based access control model [Sandhu, 1996]. Roles can be thought of as the intensional analogue to the extensional groups widely used for access control in file systems. The process for protecting sensitive content at runtime proceeds as follows. First, a content-analysis engine maps sensitive content to a role-based authorization policy associated with the content. The policy specifies the roles to which outside parties must authenticate to receive the content. Next, the appropriate authentication policy information in terms of credentials is combined with the role expression to be sent to the trust establishment agent that represents the server.

A separation of roles and credentials improves administrative scalability. One administrator can manage role definitions in terms of the credentials required to enter each role, while another administrator, perhaps even in a different organization, manages the policies that govern the disclosure of content, expressed in terms of the roles defined by the first administrator. Mentioning credentials only in the policies that govern access to roles also allows the design to be adapted more easily to protect the exchange of sensitive content in systems where the entities share a common security domain.

Scalability is also important for managing policy information. Many users will share some policies. Rather than replicate policy contents, references to shared policies could allow policy updates to be reflected immediately in the runtime environment. Separating policy information into role expressions in authorization policies and credential combinations in authentication policies allows the authentication policies to be distributed across different locations and accessed on demand. For instance, a government agency may be responsible for specifying the authentication policy for US citizens, and the US Post Office may specify the authentication policy for a US resident. When an authorization policy includes both of these roles, the security agent that is proactively establishing trust could gather the authentication policies on demand in order to submit them to the server. The resulting increase in ease of administration must be weighed against the increased runtime costs. In sum, the design for dynamic policy association provides administrative scalability through role-based authorization policies and distributed authentication policies.

4.2 Determining an Appropriate Trust Negotiation Agent

Previous work in trust negotiation did not need to address the issue of determining the trust negotiation agent with which to initiate a negotiation. The server simply reacted to a client request for service and initiated a trust negotiation in-band with that client. With client-initiated trust establishment, the server cannot rely on an existing connection with the client because the negotiation may take place before the client makes first contact with the service. Standard conventions need to be adopted to prescribe the relationship between a service and the associated trust establishment agent that represents the service. The solution cannot require clients to disclose, even inadvertently, sensitive information about the service request they intend to submit.

To find the appropriate trust negotiation agent, object-oriented systems such as CORBA and Java could provide a standard object interface to invoke trust establishment services before invoking a method with sensitive argument values. Objects providing client-initiated trust establishment could inherit from a class that supports an interface for client-initiated trust establishment. If the method name is standardized, client code that dispatches dynamic method invocations could examine the object definitions, determine dynamically at run time if the object supports an interface for client-initiated trust establishment, and invoke it before invoking a method for accepting sensitive content as an argument.

A web server could support trust establishment in advance of a sensitive request in several ways. First, a new HTTP message type could be introduced to establish trust in the web server

prior to making a specific request of the server. Second, a new Servlet API could be introduced to establish trust in a web server. Third, the TLS/SSL protocol [Dierks and Allen, 1998] could be enhanced to support a more sophisticated version of server authentication through trust negotiation rather than the limited client/server authentication it supports today. The rehandshake facility in TLS/SSL can be exploited to provide confidential trust negotiations before any application data flows to the server, the subject of ongoing research in our group [Hess, 2002]. Fourth, servers can provide a standardized directory service where clients obtain a reference to the authorized trust establishment agent for a given service. This will allow greater flexibility and local autonomy at the overhead of a directory lookup.

5 Related Work

Credential-based authentication and authorization systems fall into three groups: identity-based, attribute-based, and capability-based systems. Originally, public key certificates, such as X.509 [X509, 1997] and PGP [Zimmerman, 1994], simply bound keys to names, and X.509 version 3 certificates later extended this binding to general attributes. Such certificates form the foundation of identity-based systems, which authenticate an entity's identity or name and use it as the basis for authorization. Identity is not a useful basis for our goal to establish trust among strangers. Bina et al. introduced digital credentials to allow the binding of arbitrary attributes and support trust negotiation between strangers. Systems have emerged that use these attribute-describing credentials to manage trust in decentralized, distributed systems [Winslett, 1997][Seamons, 1997][Johnston, 1998][Herzberg, 2000].

Winslett et al. [Bina, 1994][Ching, 1996][Winslett, 1997] focuses on establishing trust with no prior acquaintance between client and server. They present an architecture for using credentials to authorize access to distributed resources. Client and server security assistants manage both the credentials and the policies governing access to sensitive resources. They emphasize the need for credential and policy exchange with little intervention by the client. Seamons et al. [Seamons, 1997] continue in this vein, developing policies that use credentials and credential attributes to authenticate clients to roles with attributes that can be used in authorization decisions.

Winsborough et al. [Winsborough, 2000] introduced trust negotiation and provided support for sensitive credentials. In their work, each credential is protected by a credential disclosure policy that controls disclosure based on credentials received from the other negotiation participant. Policies and policy extracts can be disclosed in order to guide the credential exchange. Seamons et al. [Seamons, 2001] extend trust negotiation to limit the disclosure of sensitive access control policies.

Yu et al. [Yu, 2001] introduce the notion of a family of trust negotiation strategies, overcoming the problem that none of the strategies proposed in any earlier work would interoperate. They introduce the notion of a general trust negotiation protocol that supports a variety of negotiation strategies. In this paper, we introduce an architecture for client-initiated trust establishment that can leverage earlier work on negotiation strategies, along with credential and policy sensitivity.

The Platform for Privacy Preferences (P3P) standard [P3P, 2000] defined by the W3C focuses on negotiating the disclosure of a user's sensitive private information based on the privacy practices of the server. Unlike TrustBuilder, P3P does not rely on information certified by a trusted third party. Using TrustBuilder, both parties can establish trust in each other based on any set of criteria, including privacy practices. Thus, trust negotiation generalizes P3P to base disclosure on any server property of interest to the client that can be represented in a credential.

6 Conclusions and Future Work

Trust negotiation is a new approach to establishing trust between strangers that can be used in business-to-business and business-to-consumer secure transactions. Negotiation participants exchange digital credentials describing attributes of the participants in order to build trust during a negotiation. The TrustBuilder architecture is being designed and developed to support on-line trust negotiation.

Privacy is an important issue to clients and servers in open systems like the Internet. A web server's privacy handling practices is one factor a web user can consider in determining whether or not to trust a server. We propose the creation of specific web server privacy practices credentials that clients can verify automatically during trust negotiation to make that determination. An approach that relies on certified information from trusted third parties may result in a greater level of trust between the client and server compared to the ad hoc approach currently in use where web servers displays the TRUSTe logo for users to visually inspect or where clients rely on P3P claims made directly by the server.

This paper presents the design and rationale for client-initiated trust establishment, a new approach to trust negotiation that allows a client to establish trust in a web server prior to disclosing sensitive content in a request to the server. The design scales in terms of the administration requirements for maintaining authorization policies. The design for client-initiated trust establishment includes an architecture for dynamically associating a disclosure policy with sensitive content, as well as design alternatives to locate the security agent that negotiates trust on behalf of a web server.

In the future, we will implement the architecture for client-initiated trust establishment within TrustBuilder. We have been designing and developing reusable trust negotiation components in various computational environments: web application servers, SOAP remote procedure calls, CORBA interceptors, SSL/TLS, and IPSec. We have two working prototypes that extend a web application server and a TLS client/server to support trust negotiation. Experience with these alternatives will promote the design of reusable trust negotiation components and help to realize the promise of ubiquitous trust negotiation. Future results from our research in trust negotiation will be made accessible on the web at http://isrl.cs.byu.edu/.

Acknowledgements

This research was supported by DARPA through AFRL contract number F33615-01-C-0336 and through Space and Naval Warfare Systems Center San Diego grant number N66001-01-18908. The authors thank Marianne Winslett for her valuable feedback on an earlier version of this paper. The authors also express their thanks to Marianne Winslett and Ting Yu for helpful discussions on trust negotiation and to Ryan Jarvis and Bryan Smith for their help in preparing the final version of this document.

References

[Bina, 1994] E. Bina, V. Jones, R. McCool and M. Winslett. Secure Access to Data Over the Internet. *Proceedings of the Third ACM/IEEE International Conference on Parallel and Distributed Information Systems,* Austin, Texas, September 1994.

[Ching, 1996] N. Ching, V. Jones, and M. Winslett. Authorization in the Digital Library: Secure Access to Services across Enterprise Boundaries. *Proceedings of ADL '96 --- Forum on Research and Technology Advances in Digital Libraries,* Washington, DC, May 1996.

[Dierks and Allen, 1998] T. Dierks and C. Allen. The TLS Protocol Version 1.0. draft-ietf-tls-protocol-06.txt, November 12, 1998.

[European Union, 2001] Data Protection. European Union -- Data Protection, http://europa.eu.int/comm/internal_market/en/dataprot/.

[Forrester Research, 2001] Companies Must Adopt A Whole-View Approach To Privacy. According to Forrester Research. Forrester Press Release. March 5, 2001, http://www.forrester.com/ER/Press/Release/0,1769,514,00.html

[Herzberg, 2000] A. Herzberg, J. Mihaeli, Y. Mass, D. Naor, and Y. Ravid. Access Control Meets Public Key Infrastructure, Or: Assigning Roles to Strangers. *IEEE Symposium on Security and Privacy*, Oakland, May 2000.

[Hess, 2002] A. Hess, J. Jacobson, H. Mills, R. Wamsley, K. E. Seamons, and B. Smith. Advanced Client/Server Authentication in TLS. *Network and Distributed System Security Symposium*, San Diego, CA, February 2002.

[Johnston, 1998] W. Johnston, S. Mudumbai, and M. Thompson. Authorization and Attribute Certificates for Widely Distributed Access Control. *Proceedings of the IEEE 7th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises --- WETICE '98.*

[Monson-Haefel, 2000] R. Monson-Haefel, Enterprise JavaBeans 2nd edition. O'Reilly & Associates, 2000.

[OMG, 1999] Object Management Group. The Common Object Request Broker: Architecture and Specification. Rev. 2.31, October 1999. Available at http://www.omg.org/technology/documents/specifications.htm.

[P3P, 2000] The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. W3C Candidate Recommendation. 15 December 2000, http://www.w3.org/TR/P3P/.

[Sandhu, 1996] R. S. Sandhu and E. J. Coyne. Role-Based Access Control Models. *IEEE Computer*, Volume 29, Number 2, February 1996, pages 38-47.

[Safe Harbor, 2001] Safe Harbor. U.S. Department of Commerce - Safe Harbor. http://www.export.gov/safeharbor/.

[Sandeep. 1999] J. Sandeep. Report: Half of Net users mistrust sites. http://news.cnet.com/news/0-1007-200-346152.html, Aug. 17, 1999.

[Seamons, 2001] K. E. Seamons, M. Winslett, and T. Yu. Limiting the Disclosure of Access Control Policies During Automated Trust Negotiation. *Network and Distributed System Security Symposium,* San Diego, February 2001.

[Seamons, 1997] K. E. Seamons, W. Winsborough, and M. Winslett. Internet Credential Acceptance Policies. Proceedings of the Workshop on Logic Programming for Internet Applications, Leuven, Belgium, July 1997

[TRUSTe, 2001] TRUSTe Seal Programs. TRUSTe, http://www.truste.com/programs/pub_how.html.

[Winsborough, 2000] W. Winsborough, K. E. Seamons, and V. Jones. Automated Trust Negotiation. *DARPA Information Survivability Conference and Exposition,* Hilton Head, SC, January 2000.

[Winslett, 1997] M. Winslett, N. Ching, V. Jones, and I. Slepchin. Using Digital Credentials on the World-Wide Web. *Journal of Computer Security*, **5**, 1997, 255-267.

[X509, 1997] Recommendation X.509--Information Technology—Open Systems Interconnection--The Directory: Authentication Framework. International Telecommunication Union. August 1997.

[Yu, 2000] T. Yu, X. Ma, and M. Winslett. PRUNES: An Efficient and Complete Strategy for Automated Trust Negotiation over the Internet. *7th ACM Conference on Computer and Communication Security,* Athens, Greece, November 2000.

[Yu, 2001] T. Yu, M. Winslett, and K. E. Seamons. Interoperable Strategies in Automated Trust Negotiation. *8th ACM Conference on Computer and Communications Security*, Philadelphia, Pennsylvania, November 2001.

[Zimmerman, 1994] P. Zimmerman. PGP User's Guide. MIT Press, Cambridge, 1994.