Automated Trust Negotiation in Autonomic Environments

Andreas Klenk[1], Frank Petri[1], Benoit Radier[2], Mikael Salaun[2], Georg Carle[1]

[1]Wilhelm-Schickard-Institut,
Sand 13, 72076 Tübingen, Germany
[2]France Télécom R&D,
avenue Pierre Marzin 2, 22307 Lannion, France

Abstract. Autonomic computing environments rely on devices that are able to make intelligent decisions without human supervision. Automated Trust Negotiation supports the cooperation of devices with no prior trust relationship. They can reach an agreement by iteratively exchanging credentials during a negotiation process. These credentials can serve as authorization tokens or may carry information that becomes a parameter of the further service usage. A careful negotiation strategy helps in protecting sensitive credentials that must only be available to authorized entities. We introduce the *VersaTrust* framework that supports a stateless negotiation protocol to reach comprehensive agreements. We argue how this approach applies to autonomic environments and demonstrate its scalability.

Key words: attribute-based access control, stateless automated trust negotiation

1 Introduction

The growing complexity of the information technologies infrastructure leads to an increase in administrative effort to ensure availability and security of the systems. There is a lot of manual configuration associated with implementing administrative decisions. Autonomic computing research aims for facilitating administration of complex infrastructures by introducing self-management capabilities [7] into networks and devices. The coordination of autonomic entities is challenging if these entities are part of different administrative domains without unbounded mutual trust. In such scenarios, constraints of future interactions between the devices need to be considered [4] depending on the trust between the entities. The Global Grid Forum recognized the need for an automated establishment of agreements between web services with its work on the *WS-AgreementNegotiation* specification draft [5]. However, the draft neglects the protection of sensitive information during the negotiation and requires session state at the participating hosts.

The research on *Automated Trust Negotiation (ATN)* [14] [12] [2] deals with automatically establishing mutual trust between strangers by an iterative credential

exchange. *Automated Trust Negotiation* systems use a policy driven iterative negotiation process to reach an agreement between two parties that need not have a prior trust relationship. The main focus is on the protection of sensitive information (credentials and policies) and the definition of policy languages for the negotiation process. However, ATN does not help to supervise or enforce the agreement. Other techniques must complement the ATN to check if the other party adheres to its promises.
In this paper, we explore the use of Automated Trust Negotiation for autonomic systems. We show how to reach an agreement via an automated exchange of policies and credentials.

1. We introduce the *VersaTrust* framework for stateless trust negotiation, explain how policies control the negotiation process and evaluate the feasibility and the performance of the implementation.
2. We argue how to represent the final agreement and the complete negotiation in one single document. That allows to demonstrate all conditions under which the negotiation succeeded, at a later point in time, say if the terms of the agreement are under dispute. This feature is a clear advantage over current ATN implementations which can only state the results of the negotiation but lack a method to prove the interrelation of the received credentials.

In Sec. 2 we survey related work. In Sec. 3 we introduce the stateless trust negotiation and show experimental results of the implementation in Sec. 4.

2 Related Work

Winsborough and Li came from the idea of credentials as tokens for authorization and introduced the idea of Automated Trust Negotiation for establishing trust between strangers in [14]. They discussed the *parsimonious strategy* to disclose only the minimal amount of credentials necessary for the successful termination of the negotiation. Sometimes the negotiation process itself discloses private information by referring to the existence of sensitive credentials during the negotiation process. The authors enhanced their negotiation with *Ack* policies to address these privacy concerns in [8].
IBM specified the *Trust Policy Language* for a role based access control scheme that uses credentials to determine which roles a principal can obtain. *TrustBuilder*[12] uses this language to implement a trust negotiation system that incorporates trust reputation measures.
PeerTrust [10] is an ATN system that can handle X.509 certificates and import RDF for its policies. Yamaki et al introduce user preferences into the trust negotiation by assigning a cost metric to the release of a credential [15]. The authors in [16] use a locally trusted third party to break cyclic dependencies between credentials that can occur during a negotiation. Frikken et al. [6] proposed a protocol that can reach a decision if the negotiation fails or succeeds without actually revealing hidden credentials. This method is appropriate if the

information of the credentials is of no importance for the further service usage. Within the scope of multi-agent systems, a large body of work exists on the negotiations between distributed agents to reach some specific goals [3]. Negotiations in multi-agent lack the capabilities of ATN systems for the protection of sensitive information and are not specifically fit to deal with credentials. ATN systems are comparable lightweight, because they reach a binary decision, (e.g. access granted/access denied), in contrast to multi-agent systems which negotiate about complex tasks, for instance, the market price of goods [9].
The *Trust-X* of Bertino, Ferrari and Squicciarini [1][2] is a recent ATN framework that had a strong influence on our work. This framework uses XML for its Trust Negotiation Language, disclosure policies and credentials. It uses DTD to specify credential types. It supports different negotiation strategies and optimization mechanisms. An important difference is that the *Trust-X* transmits individual disclosure policies and credentials during each round and relies on local state during the negotiation. Hence, it is not obvious how to proof the interrelation of the credentials retrospectively. *VersaTrust* in contrast can represent all conditions under which promises were made, that led to a specific agreement, within one single digitally signed document. Another difference is that *VersaTrust* allows for an easy recovery from system failure during the negotiation due to the stateless realization of its negotiation process.

3 Mutual Agreement with Automated Trust Negotiation

Automated Trust Negotiation governs the access to resources by attribute based authorization. Authorization decision use properties connected to a subject in contrast to solely the identity. This functionality can be useful for the self-management in environments where autonomic devices without prior trust relationship join the network and establish trust at the time they interact with other services. Another scenario is the collaboration of autonomic services across administrative domains without the need for manual configuration. An important property of ATN is the disclosure of only the minimal set of credentials and the protection of sensitive information within credentials. It is even possible to authorize a resource access without revealing the actual identity of the requester.

3.1 Credentials and Disclosure Policies

ATN systems use digital credentials usually signed by a trustworthy third party. *VersaTrust* utilizes currently a XML data structure for the credentials; for real world use other credential formats, for instance, X.509, or SAML are preferable. We denote the credential set of the party that initiates the request by $\mathcal{C}_L$ and the credential set of the the remote party by $\mathcal{C}_R$.
Disclosure policies define logical conditions that must be met before a resource can be accessed or a credential can be released. Propositional formulas help to

express the conditions of the disclosure policies [13][17] using the logical symbols $\wedge$, $\vee$, $\leftarrow$ and parentheses. The formula $O \leftarrow F_O(R_1, R_2, R_3..., R_k)$ governs the access to an object O. The propositional variable R_i is true if the associated credential $C_i \in \mathcal{C}_R$ can be offered by the other party and if conditions regarding the attributes of the credential C_i are satisfied. The expression $C_j \leftarrow F_{C_j}$ states that the disclosure of credential $C_j \in \mathcal{C}_L$ is regulated by the formula F_{C_j}. Credentials without protection requirements are called *unprotected* and are by default $C_k \leftarrow true$. The implementation uses XML for the disclosure policies and the negotiation state. The formula $R_x \wedge (\bigvee_{0<y<n} R_y)$ is equivalent to the XML representation of a node R_x having a number n of children R_y.

3.2 Iterative negotiation process

The objective of Automated Trust Negotiation is to find a *safe disclosure sequence* of credentials $(C_1, C_2, ..., C_n)$ in a way that all preconditions attached to the release of credentials are met before releasing them. This strategy is known as *parsimonious strategy* [13]. Before a negotiating party is willing to release a credential it must check that $C_i \leftarrow F_{C_i}(\bigcup_{k>i} C_k) = true, C_i \in \mathcal{C}_L, C_k \in \mathcal{C}_R$.

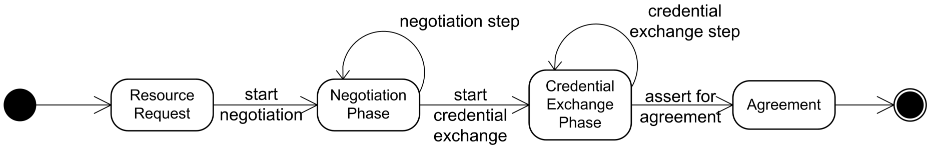

Fig. 1. State diagram of the negotiation process

The iterative exchange of *Negotiation State* messages during the automated trust negotiation contains all information about a particular negotiation process and can be evaluated without the need for session state. This is in contrast to related ATN systems which work on a tree data structure in local memory and exchange only incremental messages. The negotiation process itself is a transition of four states as depicted in the state transition diagram in Figure 1:

- **Resource Request**: The service requests access to the resource. As the resource is protected by a disclosure policy, a trust negotiation is initiated.
- **Negotiation Phase**: The objective of this phase is to find the *safe disclosure sequence* by evaluating requested credentials and their local disclosure policies.
- **Credential Exchange Phase**: This phase starts after at least one *safe disclosure sequence* was identified. The credentials that were requested most recently in the negotiation are now transmitted first. The credential exchange happens iteratively in reverse order until all credentials are disclosed.

- **Agreement**: After all required credentials were successfully exchanged, the trust negotiation terminates with a positive outcome. The objective of the negotiation is reached, for example, access to the storage service is permitted.

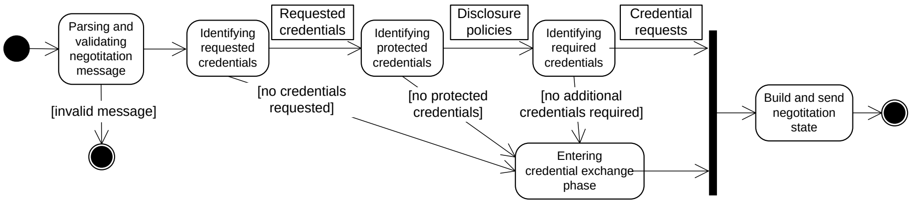
Parsing and validating negotitation message
[invalid message]
Identifying requested credentials
Requested credentials
[no credentials requested]
Identifying protected credentials
Disclosure policies
[no protected credentials]
Identifying required credentials
Credential requests
[no additional credentials required]
Entering credential exchange phase
Build and send negotitation state

Fig. 2. Activity diagram of a processing step during the Negotiation Phase

The *Negotiation Phase* is critical for the discovery of a safe disclosure sequence. The algorithm that processes a received *Negotiation State* is depicted in Figure 2. The first task is to assure syntactical and logical correctness and discard invalid messages. The next activity is to identify the requests R_i of the remote party for credentials. If a credential C_i is protected by a *Disclosure Policy* P_i, the algorithm extends the tree structure appending P_i to all leafs containing C_i in the path from root to lead. The algorithm marks leafs as *failed* that contain credentials that cannot be offered. After completion of the processing the state is sent to the other party. This algorithm iterates till a safe disclosure sequence is found, that means there are no additional credential requests for the path.
A negotiation fails during *Negotiation Phase* if the parties cannot reach an agreement. However, if there is a technical failure, or one party tries to cheat, the negotiation process can also fail at another point in time. One precaution is to exchange credentials in reverse order during the *Credential Exchange Phase*, processing the *safe disclosure sequence* in the tree from the corresponding leaf to the root. That implies that all required credentials are present and the conditions on the values of the credentials are met.

3.3 Security Aspects of the Negotiation

Security is especially challenging in trust negotiations, due to the large potential negative impact and the legal dimension of the negotiation. Both parties can protect the integrity and confidentiality against a malicious third party by using asymmetric cryptography and digital signatures with cryptographic protocols, like TLS/SSL or WS-Security.
It is more difficult to protect the negotiation against manipulations of the other negotiating party. The *VersaTrust* relies solely on the received *Negotiation State*.

We are currently investigating a strategy to apply digital signatures to the *Negotiation State* to detect manipulations.

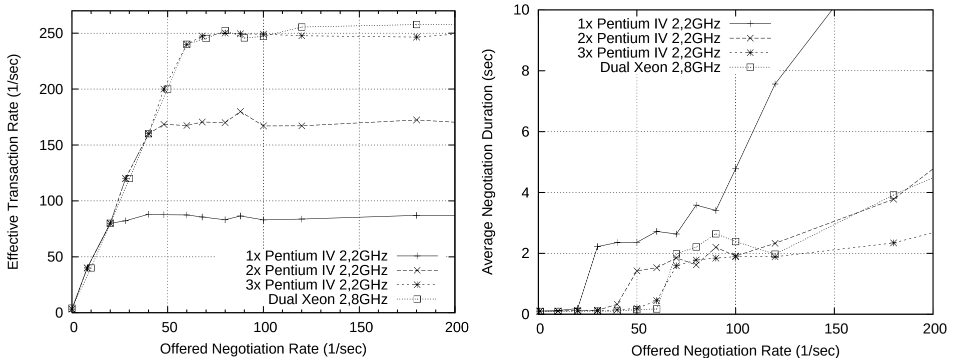

Fig. 3. Scalability under varying Load Conditions: (a) Effective Transaction Rate (b) Average Negotiation Duration

4 Experimental Results

A short overall negotiation time is important for fast service access. The outcome of one negotiation can serve as authorization for a long lasting service usage, and thereby reduce the number of required negotiations. The time for an Automated Trust Negotiation results from the iterative exchange of the negotiation messages.

As ATN is a young direction little experience exists on the characteristics of real negotiations. We used the reference example as one test case for our measurements. It allows for a negotiation consisting of 4 transactions: 2 for the *negotiation phase* and 2 for the *credential exchange phase*. It performs additionally a constraint check on an attribute of the credential. In the first experiment, one server (2,8 GHz Dual Xeon, 2x1024KiB L2 cache) was put under stress by 5 clients (2,2 GHz Pentium IV); all running with a standard configuration of Fedora Core 4, being connected in a local area network with RTTs below 0.1 ms. Both, server and client were multi threaded to support parallel processing of requests. The clients started trust negotiations at a defined rate; each experiment lasted for 600 seconds.

The left-hand figure 3(a) shows the effective transaction rate for different negotiation rates. The Xeon server scales for up to 60 complete negotiations per second in this experiment, totaling to 240 transactions per second. Another important metric is the total negotiation time - that is the time between the construction of

the request till the receipt and interpretation of the last negotiation message at the requester. Figure 3(b) shows that the average negotiation of a single server stays below 0.3 seconds for the whole negotiation till it gets into overload beyond 60 requests per second, after that point the server starts queuing.
Another experiment concerns the scalability of the system. How does the system scale with off-the-shelfe standard hardware? We used *haproxy*[1] for load balancing of up to three Pentium IV machines (see Figure 3(a)) One system can handle 80 concurrent transactions per second, two 160 and three 240, demonstrating the linear scalability of *VersaTrust*. The results in figure 3(b) show that despite the additional latency by the load balancer, the negotiation duration stays beyond 0.3 seconds besides overload conditions.
It is difficult to put these results into perspective; performance evaluations of ATN systems are rare. Certain results are published about a system that uses TrustBuilder in [11]. One single negotiation without integrity protection and about the release of one credential took already 7 second, and 0.5 seconds for each additional credential on comparable hardware. The comparison with the measures of our system is not fair, because we do not use X.509 certificates but much smaller proprietary XML certificates without cryptographic protection. We expect a performance decrease in our system when we introduce real certificates and cryptographic integrity protection of the negotiation.

5 Conclusion

This paper presented and studied a new Automated Trust Negotiation framework for attribute based resource access, called *VersaTrust*. Our approach reaches binding agreements by using a policy driven and privacy preserving negotiation. We introduced a novel stateless trust negotiation algorithm that operates on messages that encompass the complete negotiation state. The agreements in *VersaTrust* demonstrate the relationship between the credentials. Measurements of our prototype showed the scalability. Future work includes support of the security strategy and of other credential formats. We are hopeful that automated trust negotiation can become an important technology for the self-management of autonomic networks.

References

1. Elisa Bertino, Elena Ferrari, and Anna Cinzia Squicciarini. Trust Negotiations: Concepts, Systems, and Languages. *Computing in Science and Engineering*, 06(4):27–34, 2004.
2. Elisa Bertino, Elena Ferrari, and Anna Cinzia Squicciarini. Trust-X: A Peer-to-Peer Framework for Trust Establishment. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):827–842, July 2004.
3. H. Bui, S. Venkatesh, and D. Kieronska. An architecture for negotiating agents that learn, 1995.

[1] The Reliable, High Performance TCP/HTTP Load Balancer, http://haproxy.1wt.eu/

4. D. M. Chess, C. Palmer, and S. R. White. Security in an autonomic computing environment. *IBM Syst. J.*, 42(1):107–118, 2003.
5. Alain Andrieux et al. Web Services Agreement Negotiation Specification (WS-AgreementNegotiation). Technical report, Global Grid Forum, 2007.
6. Keith B. Frikken, Jiangtao Li, and Mikhail J. Atallah. Trust Negotiation with Hidden Credentials, Hidden Policies, and Policy Cycles. In *Proceedings of the Network and Distributed System Security Symposium, NDSS 2006, San Diego, California, USA*. The Internet Society, 2006.
7. A. G. Ganek and T. A. Corbi. The dawning of the autonomic computing era. *IBM Syst. J.*, 42(1):5–18, 2003.
8. N. Li and W. Winsborough. Towards Practical Automated Trust Negotiation. In *POLICY '02: Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Networks (POLICY'02)*, page 92, Washington, DC, USA, 2002. IEEE Computer Society.
9. Fernando Lopes, Nuno Mamede, A.Q. Novais, and Helder Coelho. A negotiation model for autonomous computational agents: Formal description and empirical evaluation, 2002.
10. W. Nejdl, D. Olmedilla, and M. Winslett. PeerTrust: automated trust negotiation for peers on the semantic web, 2003.
11. Lars Olson, Marianne Winslett, Gianluca Tonti, Nathan Seeley, Andrzej Uszok, and Jeffrey Bradshaw. Trust Negotiation as an Authorization Service for Web Services. In *ICDEW '06: Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06)*. IEEE Computer Society, 2006.
12. Bryan Smith, Kent E. Seamons, and Michael D. Jones. Responding to Policies at Runtime in TrustBuilder. In *POLICY*, pages 149–158, 2004.
13. W. Winsborough, K. Seamons, and V. Jones. Automated Trust Negotiation. Technical report, North Carolina State University at Raleigh, Raleigh, NC, USA, 2000.
14. William H. Winsborough and Ninghui Li. Protecting sensitive attributes in automated trust negotiation. In *WPES '02: Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*, pages 41–51, New York, NY, USA, 2002. ACM Press.
15. Hirofumi Yamaki, Masao Fujii, Kousuke Nakatsuka, and Toru Ishida. A Dynamic Programming Approach to Automated Trust Negotiation for Multiagent Systems. *rrs*, 0:55–66, 2005.
16. Song Ye, Fillia Makedon, and James Ford. Collaborative Automated Trust Negotiation in Peer-to-Peer Systems. In *P2P '04: Proceedings of the Fourth International Conference on Peer-to-Peer Computing (P2P'04)*, pages 108–115, Washington, DC, USA, 2004. IEEE Computer Society.
17. Ting Yu, Marianne Winslett, and Kent E. Seamons. Interoperable strategies in automated trust negotiation. In *CCS '01: Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 146–155, New York, NY, USA, 2001. ACM Press.