Negotiation for Authorisation in Virtual Organisations

Shamimabi Paurobally

Department of Information Systems and Computing, University of Westminster
115 New Cavendish Street, London W1W 6UW, U.K.
S.Paurobally@westminster.ac.uk

Abstract. In virtual organisations, the authorisation and expression of policies in terms of direct trust relationships between providers and consumers have the problems of scalability, flexibility, expressibility, and lack of policy hierarchy because of interdependent institutions and policies [7]. This paper proposes a bilateral negotiation protocol and an English auction to negotiate a list of credentials to be exchanged after a service level agreement has been drafted, and that would provide sufficient trustworthiness for the parties in the negotiation. We implement and evaluate our algorithms as grid services in a virtual organisation (VO) to show the effect of negotiation on the trustworthiness achieved within a VO.

1 Introduction

The long-term Grid vision involves the development of "large-scale open distributed systems, capable of effectively, *securely* and dynamically deploying Grid resources as required, to solve computationally complex problems" [1]. Thus, traditional centralised methods needing complete information for system wide optimisation of performance, reliability and security are not enough. In current Grid applications, heterogeneity and dynamic provisioning are limited, and dynamic virtual organisations (VOs) are restricted to those parties with a priori agreements to common policies and practice. To remedy this, there is a drive towards service-oriented architectures and virtual organisations which can support a broad range of commercial applications and authorisation mechanisms [9], [5]. Grid computing research is investigating applications of virtual organisations for enabling flexible, secure and coordinated resource sharing among dynamic collections of individuals, institutions and resources. The virtual organisations and usage models include a variety of owners and consumers with different security requirements, credentials, usage, access policies, cost models, varying loads, resource requirements, and availability. The sharing and coordinated use of resources may involve not only file exchange but also direct access to computers, software, data, and other resources, as is required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science, and engineering, each with their own access policies and credentials.

Please use the following format when citing this chapter:

Paurobally, S., 2007, in IFIP International Federation for Information Processing, Volume 238, Trust Management, eds. Etalle, S., Marsh, S., (Boston: Springer), pp. 107–122.

In this paper, we focus on facilitating secured and trustworthy interactions between grid services, since a key challenge for the Grid in the coming decade is adaptability to varying security requirements and capabilities. There is a need for complex and dynamic policies governing access to resources. In virtual organisations, the authorisation of policies to form direct trust relationships between producers and consumers has the problems of scalability, flexibility, expressibility, and lack of policy hierarchy because of interdependent institutions and policies [7]. As members of institutions or VOs change, policies change accordingly. We use negotiation mechanisms to address the problem of scalability for authorisation within distributed virtual communities. Another advantage to using negotiation for bringing about trust lies in the inability of current systems to establish trust between complete strangers, as explained in Winsborough et. al. [10]. The requirement for a-priori knowledge to establish trust between interaction partners cannot be met in truly open distributed systems.

There are significant differences between our approach for deploying negotiation in VO authorisation and the current work on trust negotiation by Winslett et. al. [6] and Winsborough et. al. [10]. In contrast to these latter works, we do not send a user's credentials and certificates during the negotiation to respond to a request, rather the user and service provider (or authorisation server) negotiate and agree on a suitable set of credentials for resource access. The actual credentials are exchanged on both sides only after the negotiation has terminated with a service level agreement. Our approach has the advantage of preventing malicious parties from obtaining sensitive information from others through negotiation without having any intention of reaching an agreement. Thus, the certificates are sent at the end, and thereby also preventing repeated authentification and decreasing the probability for the encryption keys to be compromised. This paper also advances the state of the art by considering *1-many* negotiations in the form of English Auctions. Here the auctioneer is a service provider whose goal is to maximise secure access to its resources by choosing out of a number of consumers the most trustworthy one. The bids are offers with a list of credentials that the auctioneer evaluates. Finally, our implemented negotiations are at a higher level than at the Operating Systems or Hardware level. Here, the negotiations are concerned with agreeing on a set of credentials for authorisation to securely access a resource.

This paper is structured as follows. Section 2 provides critical analysis of current forms of negotiation in VOs. Section 3 describes our approach and two negotiation protocols that we deploy in a grid framework. Section 4 describes the strategies used for evaluating and generating credentials offers. Section 5 presents an evaluation of the two negotiation protocols. Section 6 concludes.

2 Related Work

In this section, we analyse the current state of the work on trust negotiation and the need for more flexible negotiation in this area. Our work is mostly relevant

to the current research by Winslett et al. [6] and Winsborough et. al. [10]. The former proposes an authorisation broker service that uses negotiation to issue tokens for resources on consumers' requests. The latter defines an architecture to establish mutual trust through negotiation and specifies various strategies, assuming cooperation between participants. In our framework, we do not assume cooperation but rather allow for self-interested agents that most probably have different goals.

Figure 1 depicts the type of trust negotiation developed in [6] in a stock exchange scenario. Alice has a broker ID credential, protected by a policy that requires a certificate from the SEC (Securities and Exchange Commission) showing that the holder is authorized as a stock exchange. Bob is a stock exchange that offers an internet-based exchange service to stock brokers. Bobs authorization policy for the service requires that the customer present a current broker ID from a recognized brokerage firm. Alice tries to access the service and Bob responds by sending Alice the authorization policy that protects this service. Alice is not willing to disclose her credential to Bob, because its authorization policy is not yet satisfied. Alice sends her broker ID policy to him and he sends Alice his certification issued by the SEC and a proof that he owns the certificate. Alice sends her broker ID credential to Bob who grants Alice access.

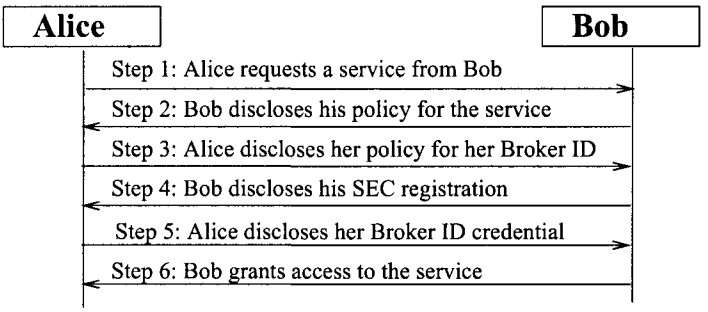

Fig. 1. Example of current forms of Trust Negotiation [6]

Winsborough et. al. [10] also follows such a model of trust negotiation where credentials are requested and exchanged during the whole interaction, whenever required by the participants policies. Although such approaches provide flexibility in acquiring trust between strangers, there are situations that could prove to be insecure, as argued below:

Malicious agents. There can be malicious agents whose goals are to gather as much information as possible about other users without intending to reach the end of the interaction and sending their final credentials. For example in figure 1, Alice could be a duplicitous agent intending to obtain Bob's SEC registration for illegal access or impersonation. After Bob has disclosed his SEC registration, Alice does not disclose her Broker ID credential, and stops communicating with Bob, blaming this on a fake faulty communication or that

she is no longer interested in the resource. Bob would not suspect any foul intentions. Here Alice has only disclosed non-vital information about her broker ID policy whilst she has gathered secure information about Bob's policy and his SEC registration. Thus, the problem in this type of interaction is that users are disclosing their credentials during the interaction without any guarantee of a successful negotiation and of a binding agreement.

Non flexible interaction. Negotiation is defined as a form of decision making where two or more parties jointly search a space of possible solutions with the goal of reaching a consensus [3]. Moreover, there is not only one agreement that is possible but a space of possible agreements that represents a convergence between the preferences of two or more parties during the negotiation. The concessions depend on constraints such as time, resource availability and utility valuations. Thus in an auction or a bargaining situation, at the beginning of a negotiation, the final agreement is unknown to all parties. The scenario in figure 1 is a discrete form of negotiation of the request-response type, where neither Alice nor Bob can choose what to request or offer from a set of possible agreements. The contents and sequence of messages are fixed in the above scenario for the exchange of the policies and credentials. What would turn this in a full-fledged negotiation would be if Alice and Bob bargain about what type of credentials each is willing to disclose and to be disclosed in exchange, without any pre-determination of what credentials are to be sent.

Unnecessary credential disclosure. Consider the situation where Alice has disclosed credentials {`Ca,Cb`}, but does not have the required {`Cc`}. So Bob back-tracks and instead asks for credential {`Cd,Ce`}. However, credential `Cd` subsumes `Ca`, and `Cb` is no longer relevant if {`Cd,Ce`} are sent. Thus, it can again be seen that sending credentials during the negotiation can disclose some credentials that could later prove to be unnecessary. Here also, a malicious agent can exploit this situation by negotiating as long as possible and asking the disclosure of different sets of credentials to gather as much information about other agents. Thus parties should be unwilling to readily disclose credentials. Each exchange of credentials and decryption of the private key of a sender provides another opportunity for the information and the key to be compromised.

No public auctions. Disclosure of credentials prevents the use of open-cry auctions, such as the English auctions, because the bidders will see each other's certificates which will have to be advertised publicly in the bids. Thus the above scenario cannot use auctions for negotiations and misses the advantages that are associated with auctions such as a competitive market.

3 Our Approach: Negotiation of Credentials

To remedy to the above problems, we do not pass actual credentials during a negotiation, but negotiation is on what credentials may be sent at the end and after reaching a binding agreement between the parties. The credentials are exchanged at the end of a successful negotiation, reducing the risks of ex-

ploitation from malicious agents and of unnecessary information disclosure in case of back-tracking in a negotiation. Our approach can help advanced Grid applications in which a single interaction may involve the coordinated use of resources at many locations and allows users to access resources without repeated authentication.

We also do not assume cooperation, but instead consider the case of each party being self-interested and having their own goals, as would normally be the case in grid and e-commerce scenarios. On the one hand, consumers have digital certificates that attest to their attributes, membership in VOs, and requirements for resources. A consumer might want to get access the resource as soon as possible. On the other hand, resource owners have access policies for resources, sharing constraints for their resources and wants to collect as much information as possible about the clients attributes before granting access to resources. In some cases, a consumer trusts the server apriori but the server does not trust the client. In our case, we do not assume any prior trust on either side. An acceptable level of trust is established between the parties based on their properties proved through credential disclosure at the end. Negotiation allows to determine in an iterative manner which credentials and policies are to be disclosed between parties at the end. No sensitive credentials are disclosed if anyone party terminates the negotiation prematurely.

In this section, we describe two negotiation protocols which we implement as grid services – the time-constrained bilateral negotiation and English auction.

3.1 Bilateral Negotiation

A bilateral negotiation occurs between two agents, a consumer and a service provider or an authorisation server. Figure 2 shows such a protocol where at the beginning the service provider advertises the set of credentials it recognises. Agreement will be on a subset of these credentials. The service consumer A obtains the interface of the service provider B through service discovery and makes an initial offer with the credentials, for example $\{C^B_{ibm}, C^A_{GSI}\}$, where the super-script to a credential denotes the owner and sender of that credential. The service provider B evaluates these credentials and either accepts if the proposed credentials from the consumer is sufficiently secure, or it counter-offers with other credentials, for example $\{C^A_{SSL}, C^B_{TSL}, C^A_{prima}\}$. The negotiation continues until either party accepts or rejects a counter-offer. Acceptance could occur because the counter-offer has reached a required trust threshold for a party, where as rejection occurs if the deadline of the negotiation arrives for a party without reaching its trust threshold.

3.2 English Auction

The other negotiation protocol we implement is the English auction which is a 1-many protocol. This protocol is used when there is a scarcity or excess of resources. For example, if resources are scarce, a service provider acts as the

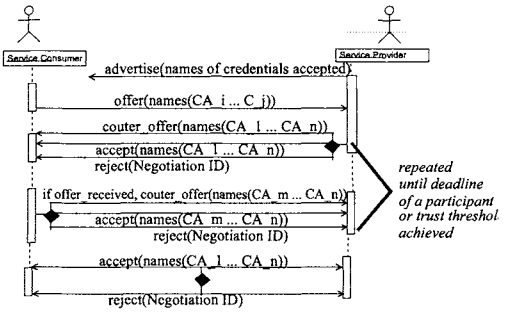

Fig. 2. Bilateral Negotiation Protocol

auctioneer and auctions access to its resources to many consumers and the consumer offering the highest security options wins the auction and thus the service provider maximises the security access to its resources. On the other hand if supply exceeds demand for resources, then there is a reverse auction between 1 consumer and many providers. Here the consumer acts as the auctioneer and has two options – either it chooses the provider that requires the least number of credentials from the consumer, or the consumer chooses the provider offering the most secure information about itself. As future work, we intend to develop many-many double auctions.

In an English auction with time constraints, the auctioneer grid service, for example the service provider, informs bidder consumer grid services the start of the auction and the issues on which to bid, that is the advertised credentials list. Bidder consumer services reply with bids offering their sets of credentials. The auctioneer evaluates the bids, chooses the best bid, called `highest_bid`, and invokes the `Submit_Bid` method on the bidder consumer services with `highest_bid` as parameter. By doing so, the auctioneer is invoking the bidder services to submit counter-bids again respective to `highest_bid`, so as to offer a better *ordered* list of credentials than those listed in the `highest_bid`. The bidder services evaluates whether they can improve on the highest bid and if so send their revised bids. The auctioneer again evaluates received bids and requests higher bids from the remaining bidders with respect to the new highest bid. This process of requesting new bids and submitting higher bids continues until the auction deadline or there is only one bidder left. At these terminating conditions, the auctioneer invokes the `Auction_Result` method on the bidder services indicating the end of the auction and informing the winning bidder of the agreed credential list. Figure 3 shows the English Auction port-type.

3.3 Negotiation Subject

We define the contents of offers and bids for both negotiation protocols. Figure 4 gives the WSDL specification of the credentials that are exchanged between the consumers and service providers. We extend Jonczy and Haenni's definition of

```
<wsdl:portType name="Bidder">
 <wsdl:operation name="Submit_Bid" parameterOrder="bid">
   <wsdl:input message="SubmitBidIn" name="SubmitBidIn"/>
   <wsdl:output message="SubmitBidOut" name="SubmitBidOut"/>
 </wsdl:operation>
 <wsdl:operation name="Auction_Result" parameterOrder="bid">
  <wsdl:input message="ResultIn" name="ResultIn"/>
  <wsdl:output message="ResultOut" name="ResultOut"/>
 </wsdl:operation>
 <wsdl:operation name="getBidderID">
  <wsdl:input message="getBidderIDIn" name="getBidderIDIn"/>
  <wsdl:output message="getBidderIDOut" name="getBidderIDOut"/>
 </wsdl:operation>
 <wsdl:operation name="getMember">
  <wsdl:input message="getMemberIn" name="getMemberIn"/>
  <wsdl:output message="getMemberOut" name="getMemberOut"/>
  </wsdl:operation>
</wsdl:portType>
```

Fig. 3. English Auction Service PortType

credentials [4]. In their work, a credential can have a class, and either a positive, negative or a mixed sign as rating. The class of a credential could be either be a statement about the trustworthiness or the authenticity of the recipient. In contrast to their definition, in our case a credential owner do not disclose to other parties the weight it assigns to the credential, i.e. the importance it attaches to the credential. This is a private matter for the credential owner and would be different for another user.

In figure 4, in the `CredentialType` which specifies a credential, we include fields to specify membership to any VO, the period of validity of that credential and any registration or cryptographic keys. We also include the negotiable field which if `false` means that this credential is compulsory the requester of that credential. For example, if A's offer include C^B_{IBM} and `IsNegotiable` is false, then A regards C^B_{IBM} as a compulsory credential to be provided by B. It is very important to note that during negotiation, information-sensitive fields in the `CredentialType` are not disclosed. Sensitive fields, such as sign and private keys are only instantiated when an agreement is reached. Thus there is enough information for a user or a server to evaluate a credential during the negotiation, but not enough to disclose any private information in case of malicious agents. `CredentialListType` is a list of credentials and *Negotiation_Subject* is what is exchanged in offers and bids. In addition to the list of credentials, the *Negotiation_Subject* include the sender of the bid or offer, which may not be the same as the recipient in `CredentialType`, for example in the case of a broker negotiating on behalf of another party. The `NegotiationID` is an identifier for a negotiation instance where a user may be involved in concurrent negotiations.

```
<wsdl:types>
<xsd:complexType name="CredentialListType">
 <xsd:sequence>
   <xsd:complexType name="CredentialType">
    <element name="class" type="xsd:string"/>
    <element name="sign" type="xsd:string"/>
    <element name="issuer" type="wsa:EndpointReferenceType"/>
    <element name="recipient" type="wsa:EndpointReferenceType"/>
    <element name="negotiable" type="xsd:boolean"/>
    <element name="VO_membership" type="wsa:EndpointReferenceType"/>
    <element name="validity_period" type="date"/>
    <element_name="private_key" type="String"/>
    <element_name="policy_details" type="URI"/>
    <element_name="any_other_details" type="any"/>
   </xsd:complexType>
 </xsd:sequence>
</xsd:complexType>

<xsd:element name="Negotiation_Subject">
   <sequence>
     <element name="sender" type="wsa:EndpointReferenceType"/>
     <element name="NegotiationID" type="xsd:string"/>
     <xsd:element ref="tns:CredentialListType"/>
   </sequence>
 </xsd:element>
</wsdl:types>
```

Fig. 4. WSDL Types for the Negotiation Subject

In both the negotiation subject and in the service level agreement, the list of credentials is an *ordered* list. For example, the credential list in the agreement $\{C^A_{SSL}, C^B_{TSL}, C^A_{prima}, C^A_{IBM}\}$ would mean that service A sends its SSL certificate, followed by B sending its TSL certificate and finally by A sending both its PRIMA and IBM-signed certificates. The service provider advertises the negotiation subject allowing the service consumer to share the same structure for the credential list.

4 Negotiation Strategies - Evaluation and Generation of Credentials

We implement strategies for evaluating and generating bids and offers of the credentials names list. In the evaluation strategy, we use a classical summation of the weighted utility of each issue in the negotiation subject, here a credential being an issue. For generation of bids and offers, we implement four strategies: 1) the truth-telling strategy, 2) the constant decrement strategy, 3) the time dependent strategy, and 4) experience-dependent strategy.

4.1 Preferences Modeling for Credentials

In order for an agent to evaluate and generate bids and offers, it needs to know what is a good enough bid and what is a good deal. To this end, a grid service has preferences which captures the profile of the owner of the grid service. From these preferences, a grid service can decide if a bid/offer is good enough by calculating whether its trust threshold has been reached. If a bid/offer is not good enough, then a service can calculate what to send as a counter-offer/bid. These preferences are private and are not disclosed to the other parties. The preferences of a grid service for a credential as an issue are:

- *Preferred value* for an issue is the ideal value for an issue, for example ideally a user might prefer a SSL certificate and a certificate from IBM.
- *Reserve value* defines the limit to which a participant is willing to concede. For example, a service provider will not accept any credentials less secure than those issued by IBM.
- *weight* is used to assign the relative importance of a credential with respect to other credentials.
- *Utility* of a credential specifies how much that credential is worth to a service provider or consumer. A higher utility means a higher worth and utility may change over time or with other environment factors.
- *IsNegotiable* is a Boolean that if false means that this credential must be provided by the other party, and if true means that it can be replaced by another credential. In our evaluation and generation strategies, the first step is to always check that the non-negotiable credentials in the received negotiation subject can be met by the receiver's preferences.
- *PrefersHigh* specifies if a user prefers a high evaluation for that credential or not. For example, a service provider may prefer to receive a high value credential from a consumer, but may also prefer to send a low value credential about itself to disclose the least amount of secure information.

Note that a service provider has to assign quantitative preferences to the list of credentials it advertise at the beginning of a negotiation, and similarly a consumer has to assign values to its known list of credentials, specially to those advertised and known credentials. For example, a service provider knows how much it values certificates from IBM, SSL, PRIMA if these are the certificates it advertises. We denote such a personal valuation for certificate C_i as $v(C_i)$.

4.2 Evaluation of Credentials

We now provide a mechanism for a service provider or consumer to evaluate a list of credentials. Evaluating a list of credentials in a received bid or offer is dependent on a user evaluating each credential in that list. In turn the evaluation of a credential in an offer/bid depends on the specifics of that credential in that negotiation subject and the user's personal preferences. More specifically,

we differentiate between the evaluation, $V(C_i)$, of a credential, C_i, in the context of a credential list and a negotiation, and the personal evaluation $v(C_i)$ which is independent of the context and the reserve preferences. This personal evaluation allows a user to know which credential it prefers itself out of 2 or more credentials, but it does not know its opponent's private valuation. It may be that two parties agree on which credentials they prefer which means that their preferences are not always opposing. Moreover, $v(C_i)$ is independent of time or other factors where as the utility can be a function changing over time or with environmental factors.

Evaluation of the credential list yields a quantitative value to the trust level achievable from an offer. The evaluation of a credential, C_i, is given as a function of the absolute value of the difference between the personal value of the credential, $v(C_i)$, and the reserve value, *reserve*, divided by the reserve value and multiplied by the utility of that credential, U_{C_i}.

For example, evaluation of credential C_{IBM} as an issue in an offer is as follows:

$$V(C_{IBM}) = (|v(C_{IBM}) - reserve|/reserve) * U_{C_{IBM}} \quad (1)$$

The valuation of a list of credentials for agent a, $V^a(cred\_list)$, called the trust valuation of such a list, is calculated by the summation of the weighted utility of credentials in that list. Let $V_j^a(cred\_list[j])$ denote the valuation of credential j in list *cred_list* for service a.

$$\texttt{Trust valuation of cred\_list} \quad V^a(cred\_list) = \sum_{1 \leq j \leq n} \omega_j^a V_j^a(cred\_list[j]) \quad (2)$$

We define the *trust threshold* to be the trust valuation of a list consisting only of preferred and ideal credentials for that user. If the trust threshold for a user is reached in a negotiation, then the opponent evaluating the trustworthiness of that user may accept the user's offer or bid. On the other hand, when when the credential list consist only of reserve values for its constituent credentials, then the *minimum acceptable trustworthiness* is obtained. Any offers or bid that are below that *minimum trustworthiness* are rejected.

4.3 Generation Strategies

We specify four strategies for generating a bid/offer in an increasing order of complexity and negotiation cost – truth-telling, constant decrement, time-dependent and experience-dependent strategies.

Truth Telling Strategy. In the truth telling strategy, the participants send their preferred list of credentials, then if an agreement has not yet been reached, then they send the reserve credentials in the second round. The first offer/bid from a service consumer is instantiated with its preferred credentials. On receiving the offer, the service provider evaluates the list according to equation

2 to obtain $V^a(cred\_list)$. If $V^a(cred\_list)$ is less than the service provider's minimum trustworthiness, then the service provider counter-offers with its own list of preferred credentials. The service consumer evaluates the provider's credential list and if this valuation is not equal/greater than its minimum trustworthiness, then the consumer counter-offers with a list of credentials where the issues are now given the consumer's reserve values. If this new counter-offer is not equal/greater than the provider's minimum trustworthiness, then the provider counter-offers with a credential list with its reserve values. This time, the consumer on evaluating the received credential list accepts the provider's counter-offer if it is within its minimum trustworthiness leading to an agreement, otherwise it rejects the received offer.

The English auction truth telling strategy resembles the bilateral protocol truth telling strategy. The first bid contains the preferred values of each bidder. The auctioneer evaluates each bid and chooses the highest bid. If the auction deadline is reached, the auctioneer declares the overall winning bid to be the highest received bid. Otherwise, if there is still time, then the auctioneer calls to submit another round of bids and passes to the other bidders the credential list in the highest bid. In the next round, the bidders submit bids with their reserve values. The auctioneer evaluates the highest bid using equation 2 and if the highest bid is equal/greater than the auctioneer's minimum trustworthiness then it declares the overall winning bid as the second round's winning bid.

Decrement Strategy. In this strategy, the participants evaluate and generate a bid/offer using the reserve values and the minimum trustworthiness, and also using a pre-defined margin above or below the reserve values. This gives the parties a chance to converge to an agreement during the negotiation even though the initial offers/bids are below the minimum trustworthiness, instead of rejecting such bids/offers in the first rounds as would occur in truth-telling. The pseudocode for the evaluation of a credential list here is summarised below:

```
for each issue in the credential list
   if non-negotiable issues in the credential list do not
              match non-negotiable issues in preferences
      return cannot accept
   else {
     if prefers high for that issue {
       marked_reserve = reserve value * (1-margin_outside_preferences)
       if value of issue in subject < marked_reserve
       return cannot accept
       } else { // prefers low
        marked_reserve = reserve value * (1+margin_outside_preferences)
       if value of issue in subject > marked_reserve
       return cannot accept
       }
return can accept
```

The generation of an offer/bid follows the same trend as for the evaluation. First non-negotiable issues are instantiated with the preferred credential

for that user. As for negotiable credentials, the credential that is offered is one with valuation closest to
(*margin_for_generation*∗*average*(*V*(*preferred credential*), *V*(*credential in received offer*))).

Time Dependent Strategy. Both the bilateral protocol and the English auction are dependent on the time left for the end of the negotiation. In the English auction, there is a deadline for receiving bids for each round and an overall deadline for the auction to end. A bidder only has its personal deadline for the overall auction. The generation of a bid/offer depends on the time left for the end of the negotiation. A bidder service determines which credential to send in a bid/offer by calculating how much to concede over the valuation of the whole credential list and over each credential in that list. Let $V^a(new\_cred\_list)$ denote the evaluation of agent a for the new credential list such that $V^a(new\_cred\_list)$ incorporates the concession from the previously received credential list. Also, the credential that has the closest match to the valuation in equation 3 is chosen to form part of the new credential list, such that the evaluation of the generated credential list as calculated in equation 2 is nearest to $V^a(new\_cred\_list)$. The bid $bid_a^t(new\_cred\_list)[cred_j]$ of bidder service b at time t with deadline t_{max} is calculated from equation 3 for each credential $cred_j$, where max_j and min_j are preferred and reserve values for that credential.

$$bid_a^t(new\_cred\_list)[cred_j] = min_j + \frac{min(t, t_{max})}{t_{max}}(max_j - min_j) \qquad (3)$$

The pseudocode for this strategy in a bilateral protocol is given below:

```
receive offer with credential list credList_i from opponent
if deadline has been reached
   evaluate credList_i using equation (2) to obtain V(credList_i)
   accept if V(credList_i) ≥ minimum trustworthiness
    otherwise reject
 else  // more time available
   if non-negotiable credentials in credList_i do not match
         non-negotiable credentials in preferences then reject; break;
   accept if V(credList_i) ≥ threshold trustworthiness; break;
   else
     generate counter-offer according to equation (3)
     send counter-offer to opponent
```

From the pseudocode, the time dependent strategy for a bidder implies that the bidder evaluates the current highest bid through equation 2 and decides whether to send a counter-bid or not. A bidder does not send a higher bid if the evaluation of the current highest bid is below its minimum trustworthiness value. If a bidder decides to send a counter-bid, then it uses equation 3 to generate a bid.

Experience Strategy. In the experience strategy, a user determines its concessions for the credential list to send based on the previous attitudes and credential lists received from its opponents two or more steps ago. In the auction, the opponent's offers are taken as the previous highest bids two or more rounds ago. If there has only been two offers/bid rounds before, then the time dependent strategy is used. Otherwise, if three or more offers/bid rounds have occured, then the experience strategy is used to generate the bid/counter-offer. As for the time dependent strategy, the credential that is to form part of the generated credential list is a concession on the credential list in the previous offer/bid. So we generate $bid^t_{a\to b}(new\_cred\_list)[cred_j]$ (or offer) from service a to service b at time t through equation 4.3. max^a_j and min^a_j are preferred and reserve values for that credential for a. The set of credentials service a generates at time t_{n+1} is within a's acceptable values for credentials.

$$bid^t_{a\to b}(new\_cred\_list)[cred_j] = min(y, max^a_j)$$

$$\text{where y} = max\left(\frac{new\_cred\_list^{t_{n-2\delta}}_{b\to a}[j]}{new\_cred\_list^{t_{n-2\delta+2}}_{b\to a}[j]} \times new\_cred\_list^{t_{n-1}}_{a\to b}[j], min^a_j\right) \quad (4)$$

5 Evaluation of Protocols through Trustworthiness

We evaluate the English auction by deploying one auctioneer and 10 bidders negotiating on a list of credentials. Similarly we evaluate the bilateral protocol in 10 different cases by changing the parties. The advertised credential list of the service provider contains 7 possible credentials and an agreement can be reached on a subset of the advertised list.

Parameters and Metrics for Evaluation. In our evaluation, in addition to varying the negotiation protocols and the strategies, we vary the personal deadlines of the participants and their reserve and preferred values for a credential. The preferred and reserve preferences for credentials will in turn yield different values for the threshold and minimum trustworthiness and influence whether an agreement is reached or not. We also consider how far apart are the preferences of the service provider with the consumers and how this factor affects performance. More specifically, to measure the performance of our protocols, we consider the following metrics:

- The number of agreements reached.
- The time to do a negotiation and especially to reach an agreement.
- The quality of an agreement and of exchanged offers and winning bids per round, calculated from equation 2. We call this metric the trustworthiness value. The trustworthiness value shown in our results are from the provider's preferences, and the same trends are obtained when trustworthiness is calculated from the consumer's preferences.

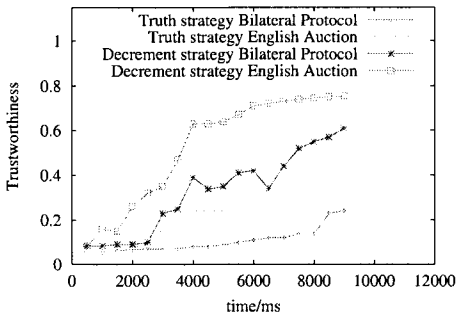

Fig. 5. Trustworthiness of offers/winning bids v/s Time for time-independent strategies

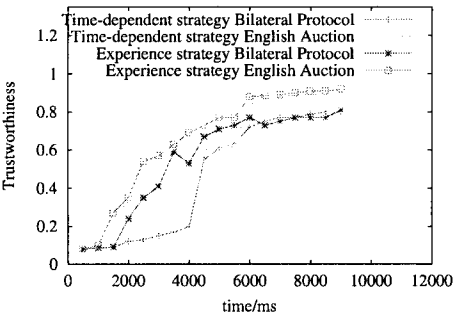

Fig. 6. Trustworthiness of offers/winning bids v/s Time for time/experience strategies

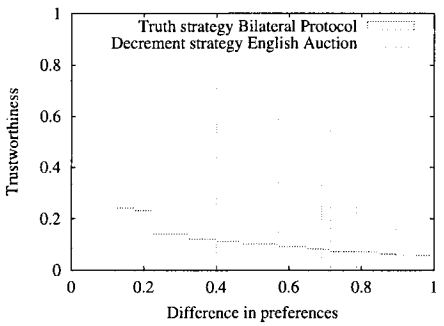

Fig. 7. Trustworthiness v/s preferences differences for time-independent strategies

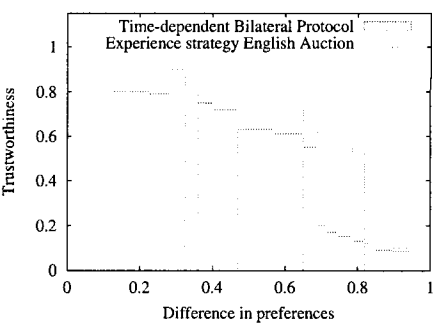

Fig. 8. Trustworthiness v/s preferences differences for time/experience strategies

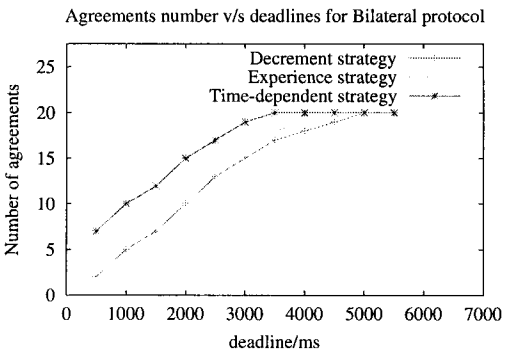

Fig. 9. Number of agreements v/s deadlines for the Bilateral protocol

Figures 5 and 6 show the trustworthiness level reached for a service provider as time elapses, by evaluating received offers and winning bids in each round for time-independent strategies (truth and decrement strategies) and for time or experience dependent strategies. The truth-telling strategy achieves a lower level of trustworthiness than the Decrement strategy, and the time-dependent strategy yields a lower trustworthiness level than the experience strategy. Also the English auction achieves a higher level of trustworthiness in a shorter time than the bilateral protocol. This is because in an English auction, there are competition between the bidders which can see each other's bids and so the trustworthiness level rises more sharply than when the consumers do not compete with each other.

Figures 7 and 8 show that the trustworthiness achieved in the offers and winning bids each round for the service provider decreases with increase in the difference between the provider's and the consumers' preferences. In fact, figure 7 shows that the English auction performs better than the Bilateral protocol, especially when with the added benefit of a more complex strategy for the English auction. With more complex strategies, such as time and experience strategies, there is lesser difference in the level of trustworthiness achieved, although the experience strategy for the English auction performs better of all strategies. This performance occurs because, in an English auction, the experience strategy takes full advantage of watching other bids in addition to choosing the winning bid in each round.

Figure 9 shows the number of agreements achieved with varying deadlines in the bilateral protocol, given that the two participants' preferences intersect and allow for an agreement. As a party's deadline increases, more agreements are arrived upon. However in this case, the time-dependent strategy yields more agreements than the experience strategy if a deadline less than 4000ms. The maximum number of agreements possible, which is 10 here for all the executed 10 bilateral protocols, is achieved within a smaller deadline for the time-dependent strategy than for the other two strategies as shown in figure 9. This is explained by the fact that a time-dependent strategy performs better with a time constraint parameter such as deadline.

6 Conclusions

Virtual Organisations require increasingly complex grid systems and scalable authorisation mechanisms for resource access. Negotiation is a technique that leads to contracts and SLAs between service providers and consumers in a VO, not only for sharing resources but also, as shown in this paper, for agreeing on a list of credentials that would bring about sufficient trustworthiness for the participants. To this end, we have described our development of the time-constrained bilateral negotiation protocol and the English auction. The participants in our approach do not exchange credentials during the negotiation, but they only exchange the names of the credentials that they are willing to disclose

at the end of the negotiation once a binding agreement or contract has been achieved. We implemented decision making strategies of varying complexity for these protocols. Evaluation of our negotiation protocols shows that both competition in English auctions and the experience strategy yield a higher level of trustworthiness in a shorter time.

As future work, we intend to perform a more thorough evaluation of our protocols, and to analyse the inter-dependencies between the credentials in an offer or a bid, and to adapt the evaluation and generation decision functions to consider such inter-dependencies.

References

1. I. Foster and C. Kesselman. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 2003.
2. Keith B. Frikken, Jiangtao Li, and Mikhail J. Atallah. Trust negotiation with hidden credentials, hidden policies, and policy cycles. In *NDSS*, 2006.
3. N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge. Automated negotiation: prospects, methods and challenges. *International Journal of Group Decision and Negotiation*, 10(2):199–215, 2001.
4. J. Jonczy and R. Haenni. Implementing credential networks. In K. Stølen, W. H. Winsborough, F. Martinelli, and F. Massacci, editors, *iTrust'06, 4rd International Conference on Trust Management*, LNCS 3986, pages 164–178, Pisa, Italy, 2006. Springer.
5. N. Kelly, P. Jithesh, P. Donachy, T. Harmer, M. Perrott, R.and McCurley, M. Townsley, J. Johnston, and S. McKee. Genegrid: A commercial grid service oriented virtual bioinformatics laboratory. In *Proceedings of the 2005 IEEE Conference on Services Computing, Orlando*, pages 43–50, 2005.
6. Lars Olson, Marianne Winslett, Gianluca Tonti, Nathan Seeley, Andrzej Uszok, and Jeffrey M. Bradshaw. Trust negotiation as an authorization service forweb services. In *ICDE Workshops*, page 21, 2006.
7. Laura Pearlman, Von Welch, Ian T. Foster, Carl Kesselman, and Steven Tuecke. A community authorization service for group collaboration. *CoRR*, cs.DC/0306053, 2003.
8. Daniele Quercia, Manish Lad, Stephen Hailes, Licia Capra, and Saleem Bhatti. Strudel: supporting trust in the dynamic establishment of peering coalitions. In *SAC*, pages 1870–1874, 2006.
9. Simon Firth. *The Future is Grid.* Hewlett-Packard (HP) Labs, `http://www.hpl.hp.com/news/2003/oct_dec/grid.html/`, 2003.
10. W. Winsborough, K. Seamons, and V. Jones. Automated trust negotiation. In *DARPA Information Survivability Conference and Exposition (DISCEX 2000)*, 2000.