Extending the Inter-domain PCE Framework for Authentication and Authorization in GMPLS Networks

Silvana Greco Polito, Mohit Chamania and Admela Jukan
Technische Universität Carolo-Wilhelmina zu Braunschweig, {polito, chamania, jukan}@ida.ing.tu-bs.de

***Abstract*— IETF is working on the design of new architectures and signaling solutions to support inter-AS (Autonomous System) GMPLS-TE (Generalized Multi Protocol Label Switching with Traffic Engineering) for multi-domain, multi-carrier connection setup with guaranteed quality of service (QoS). In addition, the Path Computation Element (PCE) working group is developing the framework for inter-domain path computation. However, many issues are still open regarding the joint path computation and path setup signaling solutions for inter-carrier authentication and authorization (AA). In this paper, we propose the first security solution which integrates inter-domain AA features in the PCE path computation framework. Specifically, we define a new architecture for inter-domain QoS path provisioning based on an extension of the PCE framework to include features that allow domains interested in inter-domain resources to get AA for end-to-end path provisioning over multiple domains belonging to different carriers. In addition, we introduce a mechanism to tie policies controlling path setup with the AA mechanisms introduced in the PCE framework. While at present provisioning of inter-domain paths is based on rather static settlements between neighboring domains that make end-to-end QoS provisioning a challenge, we propose an AA framework that allows domains interested in setting an inter-domain QoS path to have guarantees about resource provided by each domain along the path from source to destination. Simulation results show the performance of the model proposed in networks having different size and connectivity.**

Keywords **-** Authentication, Authorization, QoS, Path Setup, Signaling, Policy

I. INTRODUCTION

In connection-oriented networks, inter-domain QoS path provisioning is a complex procedure in the data and control planes, as it includes trust relationships and QoS agreements between domains, end-to-end path computation procedures, setup procedures that allocate resources along the multi-domain path computed, as well as authentication and authorization (AA) models that allow domains to verify the identity and authorization rights of any entity requesting resources [1]. Efforts are underway to develop standards for constraint-based inter-domain path computation and path setup signaling within the Path Computation Element (PCE) framework [2] and the Inter-AS GMPLS-TE protocol [3]. What is still absent from current efforts, and is critical towards the commercial implementation of automated inter-domain control and management, is the integration of AA features. Although existing models implemented in the IP layer use bilateral QoS agreements and trust relationships between neighboring domains, they only authenticate and authorize any neighboring pair of domains and cannot guarantee end-to-end QoS. In this paper, we propose a solution that guarantees end-to-end QoS provisioning and allows source domains to have knowledge about availability of resources in all the domains along the inter-domain path desired. We apply this solution to the emerging networking technologies in the layers below the IP, most notably the GMPLS-based networks. Our model integrates multi-domain authentication and authorization with the control plane of the emerging PCE framework [2] for inter-domain path computation. This is a fundamentally new approach since most existing works [4, 5] decouple authentication and authorization from the control plane and propose service planes that perform resource negotiation and inter-domain authentication and authorization, whereas the control planes of individual domains work together for inter-domain path computation and setup. Instead, our approach is based on property of path computation signaling to encompass resource requests and responses. We take advantage of the already existing path computation element signaling protocol (PCEP) [6], which allows the source domain to ask for computation of paths across multiple domains, and for these domains to provide information about the available paths. We enhance this signaling to include authentication and authorization for resources requested inside each domain. We also propose to use the AA mechanisms integrated in path computation to govern path setup. Our ultimate goal is the introduction of a new architecture for end-to-end QoS provisioning in GMPLS networks, which is based on an extension of the PCE control plane to support inter-domain AA.

The paper is organized as follows. We introduce the PCE framework and the objectives of the model proposed in Section II. In Section III we define the architecture and the security procedures of the model proposed for QoS path provisioning. Specifically, we describe the procedures addressing the security objectives of the paper in Section III.A, and we define the management and the control planes of the architecture proposed in Section III.B and III.C, respectively. We then define the interworking between the entities of the architecture and the PCE-based signaling solution for AA in Section IV. In Section IV.A, we describe the intra-domain and inter-domain interactions for multi-domain path computation and setup with AA, while in Section IV.B, we define the required extensions to the PCE protocol (PCEP) to be used as carrier of AA information between domains. Finally, we provide a performance study related to the AA load and delay of the model proposed in Section V and conclude the paper in Section VI.

II. PCE BACKGROUND AND OUR OBJECTIVES

A. Path Computation Element (PCE) framework

The Path Computation Element (PCE) framework defined by the IETF PCE working group, introduces an architecture (see Figure 1) in which Path Computation Clients (PCC) re-

978-1-4244-3435-0/09/$25.00 ©2009 IEEE

ceive service requests from users and ask PCEs located in the control plane to compute paths that satisfy the users' requests. PCEs compute single-domain paths using information stored in their Traffic Engineering Database (TED), whereas multi-domain paths are computed by interacting with PCEs of other domains. To compute a multi-domain path, the PCE sends a path computation request [7] to the PCEs along a pre-selected chain of domains to the destination, asking them for information about paths available in their domains satisfying a set of constraints. The PCE of the destination domain issues a path computation response [7] that arrives to the source PCE following the path of the relative request and each domain inserts information about optimal paths inside their domain to create a Virtual Shortest Path Tree (VSPT) [7]. To do so, each intermediate domain intercepts the path computation response, reads the paths described in this response and computes local paths that extend the VSPT issued by the downstream domain till the border nodes connected to the upstream domain. The source PCE uses the virtual shortest path tree to compute the optimal inter-domain path.

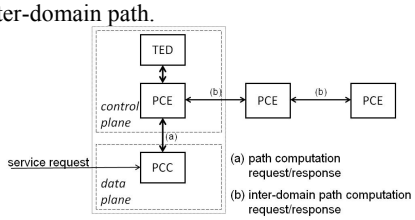

Figure 1: The basic PCE architecture.

Paths computed by the source PCEs are given as input to setup protocols for initiating resource allocation. In this work, we use the GMPLS [8] control plane, which is emerging as the de-facto standard for path setup in Layer 1 and 2 networks. The Resource Reservation Protocol with Traffic Engineering Extensions (RSVP-TE) [9] is used to set up traffic engineered label switched paths (TE-LSPs) inside a single domain, and recent standardizations have extended the capability of GMPLS with RSVP-TE to set up inter-domain TE paths [3].

B. Our Objectives

The main objective of this work is the design of a control and management plane architecture and a signaling solution for authentication and authorization between neighboring as well as non-neighboring domains to provide the end-to-end QoS. We explore the possibility of integrating authentication and authorization signaling for resource reservation with the path computation requests and responses and define a framework that uses the PCE protocol as carrier of AA data between domains. The solution proposed includes a mechanism to establish QoS agreements between source and transit domains, a mechanism to authenticate domains and to make policies of the control signaling dependent on the authorization right of the source domain and a mechanism that allows source domains to provide transit domains with guarantees against payment for the resources obtained.

We propose a security model that combines features of symmetric and asymmetric key-based methods to allow authentication between domains, guarantee integrity and non-repudiation of security information exchanged between them and protect sensitive information against eavesdropping. We propose an asymmetric key based-method that uses digital signatures for authentication of the issuer of path computation and setup requests and verification of integrity of the request content. A keyed-hash method is used to authenticate transit domains and protect information provided by them to source domains. The model proposed is composed of four main procedures which are: (i) setting of QoS agreements between domains, (ii) authentication and authorization of path computation requests, (iii) authorization of path setup requests and (iv) provisioning of guarantees against payment. In the following section we describe these procedures along with the proposed PCE-based architecture.

III. PROPOSED PCE-BASED ARCHITECTURE

Figure 2.a describes the details of the architecture proposed, with a functional decomposition within the control- and management planes. In the control plane, we propose to add the following components to the existing PCE framework: (i) the policy server that controls policy functions at the edge nodes to restrict access to authorized flows; (ii) the AA server that performs procedures for mutual authentication and authorization with other domains, (iii) the resource monitoring agent (RMA) that provides statistical information about resource availability. The management plane is proposed to include two new components: (i) the resource provisioning agent (RPA) responsible for setting path provisioning agreements with other domains, and (ii) the charging server that performs billing functions. The main difference between the control and management planes is the time scale at which the new functions are invoked. Whereas the management plane is supposed to work within the larger time scales, the control plane is typically dynamic, and request/response event driven, typically on a per-flow basis. In the following subsections, we first review the proposed security procedures in line with our objectives, and with these objectives in mind we describe the new functions of the management and control planes.

A. Security procedures

1) *Setting QoS agreements.* In our model, source domains need to establish QoS agreements with transit domains before a path request can be sent. A QoS agreement defines the QoS services that the requester (source domain) can ask the responder (transit domain) for. While subscribing to a QoS agreement, the two parties issue a security object called AA token, which encodes information used to perform AA each time a resource request has to be processed. The AA token, which is described in Figure 2.b, contains a unique identifier, the authorization profile of the source domain, its certificate and a key shared by the two parties. The authorization profile defines the amount of resources of one or more QoS classes that the source domain is authorized to use as well as their price. The certificate contains the public key of the source domain which is used for authentication of requests issued and signed by it. The key shared by the two parties, called symmetric key in the following, is used by a keyed-hash mechanism that allows authentication of the transit domain and protection of sensitive information exchanged between the two domains. In this paper we do not enter into details on the method used to exchange the token, but we want to underline that such a method has to guarantee protection against eavesdropping to avoid malicious subjects from obtaining the symmetric key.

2) *AA for path computation.* The AA procedure in the path computation request allows transit domains to verify the identity and authorization right of the requesting domain. The authentication procedure uses a digital signature computed on the content of path computation requests by the source domain. We recall that path computation requests are forwarded by

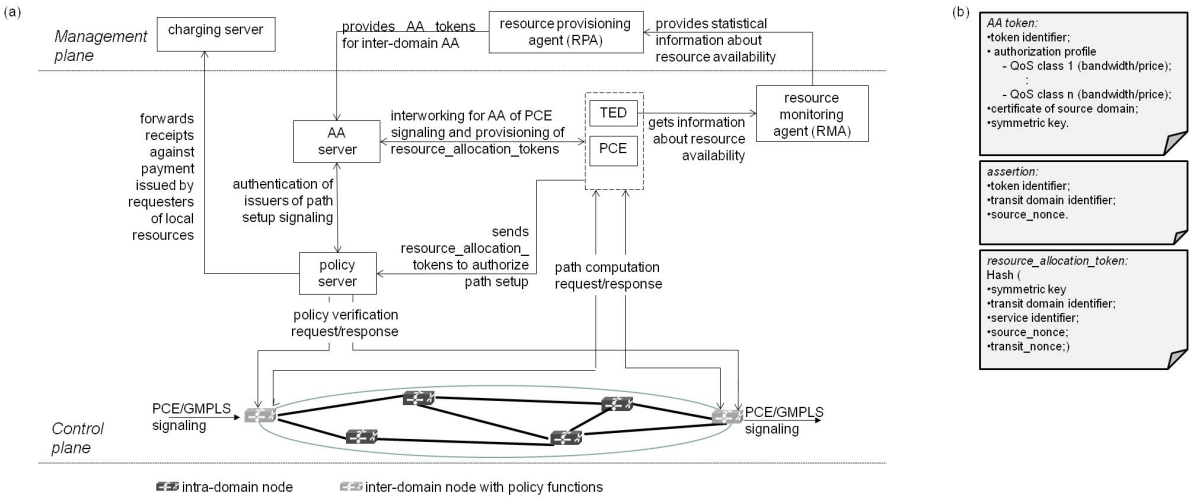

Figure 2: (a) Architecture for inter-domain QoS path provisioning; (b) Security objects.

transit domains until the destination and that no changes are made to the content of this message by transit domains. This allows the source domain to use the same signature to be authenticated by all the transit domains. The source domain issues a security object called assertion for each transit domain and includes it in the path computation request. The assertion (see "assertion", Figure 2.b) contains the identifier of the AA token corresponding to the agreement with the remote domain. The remote domain uses the token identifier carried in the assertion to individuate the relative token containing information for AA of the source domain stored in its servers. The assertion also contains the identifier of the transit domain and a pseudo random string called source_nonce which allows the source domain to make an association between the assertion and the request for which it was computed. The transit domain identifier is used to identify the domain to which the assertion is addressed to.

3) *AA for path setup requests.* Authentication and authorization of path setup requests allow setup signaling coming from the source domain to overcome the policy control at the domain boundary. Since in our model, the AA mechanisms used in path computation and path setup are coupled, we introduce authorization information in the path computation response, which can then be used during path setup to authorize the incoming connection request. Transit domains issue a security object called resource_allocation_token each time a path computation request is authenticated and authorized. The resource_allocation_token is univocally associated to the paths computed and is provided to the source domain as an authorization key for setup signaling. The transit domain keeps a local record of the association between the resource_allocation_token, the requesting domain and the paths computed. The source domain, on receiving the path computation response, computes a digital signature on the resource_allocation_token and includes the signed resource_allocation_token in the path setup signaling. When a transit domain receives the setup signaling, it verifies the validity of the signature and checks the requested path against the paths associated with the resource_allocation_token. The digital signature is used to verify the identity of the source domain and the resource_allocation_token mapping stored locally ensures that the path was originally computed for the requesting domain.

4) *Guarantees against payment.* In the model proposed, the resource_allocation_token signed digitally by the source domain is also used as a receipt against payment. The receipt describes the service provided by transit domain and is used to prove that the source domain will pay for it. The resource_allocation_token is obtained (see "resource_allocation_token", Figure 2.b) by applying a hash function on a string composed of the symmetric key, the identifier of the transit domain, the identifier of the service provided, the source_nonce and a pseudo random string called transit_nonce issued by the transit domain to guarantee univocity of resource_allocation_tokens computed for different requests. The source_nonce is included to allow the source domain to verify that the resource_allocation_token was computed in response to the request for which the source_nonce was computed and the service identifier encodes the resource offer since it is associated with a pre-specified amount of resources and cost. The symmetric key allows verification of data origin of the resource_allocation_token. In the model proposed, the source domain, on receiving the resource_allocation_token, applies the hash function to the same information used by the transit domain and checks if the result obtained matches the resource_allocation_token received to verify its integrity and data origin.

B. Management plane and interactions with the control plane

As introduced above, the entities running on the management plane are the resource provisioning agent (RPA) and the charging server. The RPA regulates the subscription of path provisioning agreements with other domains. These agreements allow transit domains to provide the source domain with information about their resources and are also used by source domains to provide credentials proving their capacity to pay for the resources requested. The subscription of a path provisioning agreement does not imply static assignment of resources, and resources are allocated upon receiving path setup requests. The agreement also does not guarantee that a path will always be available. This requires that RPAs control subscription of path provisioning agreements on the base of estimation of incoming resource requests and resource availability in their domains. The resource monitoring agent uses the in-

formation stored in the TED to compute statistical information about resource availability which is used by the RPA to make decisions regarding subscription of agreements. When a path provisioning agreement is subscribed, a AA token is issued which describes the resources that the source domain is authorized to get and information for AA (see Section III.A.1). The RPAs then provide the local AA servers with the AA token to allow AA during path computation and setup.

C. Control plane and interactions with the data plane

The control plane of the proposed path provisioning architecture is obtained by extending the PCE based control plane architecture. It is composed of the RMA that provides information about resource availability to the management plane, the PCE responsible for path computation, the policy server that interacts with the boundary nodes to control access to the network, and the AA server that performs inter-domain AA. The security framework proposed requires intra-plane and inter-plane interactions. The AA server stores the AA tokens provided by the RPA and interacts with the PCE to perform authentication and authorization of the PCE signaling. The AA server also interacts with the policy server for authentication and authorization of the path setup signaling. The policy server is responsible to police the path setup requests at the borders of the domain, and interacts with the AA server to check the authorization of incoming path setup requests. It interacts with the PCE to get information about the paths authorized, and sets policy rules in the edge nodes to restrict entry to authorized connections only. The policy server also provides the charging server with information about resources used by authorized domains which is used to compute the billing information.

IV. PCE Protocol and the Proposed Extensions

A. PCE signaling flow

We now describe the interworking between the PCE, the AA server and the policy server to process path computation signaling in source, transit and destination domains, as illustrated in Figure 3.

Source domain: issuing path computation requests.

For any inter-domain path computation request, the PCE of the source domain sends the request issued to the AA server asking for AA data that allows transit and destination domains to authenticate and authorize the source domain. The AA server issues a list of assertions, one for each domain along the inter-domain path (see Section III.A.2), and the pseudo random source_nonce. The source_nonce is used to make an association between the path computation request and the assertions issued for it, and this mapping is stored locally on the AA server. The AA server adds the assertion list to the path computation request along with a digital signature computed on the content of the message. The signature guarantees data origin, integrity and non-repudiation of the message. To include the digital signature and the assertions in the PCE message, we introduce extensions to the PCE protocol, which are further explained in Section IV.B. The AA server returns the PCE request message enhanced with AA content to the PCE which sends it to the downstream PCE of the inter-domain chain.

Transit and destination domains: processing incoming path computation requests.

On receiving an inter-domain path computation request, the PCE verifies the identity and authorization right of the source domain before activating the path computation procedure. It also issues a resource_allocation_token for the corresponding response. The source domain can then use this object for authorization during path setup. In the model proposed, transit domains forward the path computation request as soon as they receive it and perform the AA procedures while waiting for the response. By doing so, downstream domains do not have to wait for the AA mechanisms to finish at each node before getting the path computation request. To authenticate the incoming request, the PCE forwards it to the AA server. The AA server mines the assertion addressed to its domain, and uses the token identifier to get the token of the source from its local database. The certificate described in the token allows verification of the signature on the message. If the signature verification is successful, the AA server verifies if the source domain is authorized to get the resources required. If authoriza-

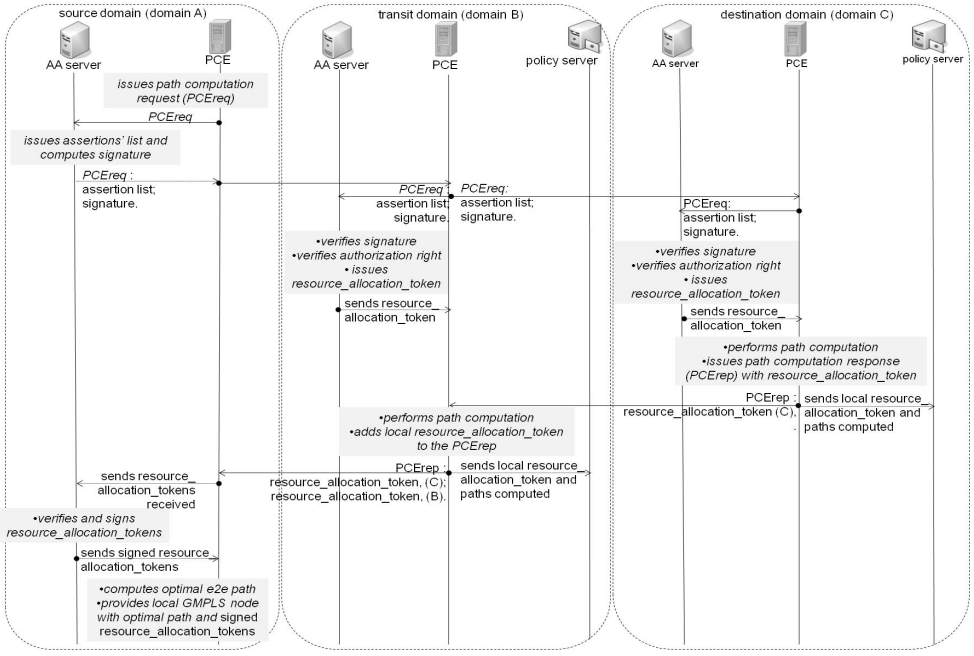

Figure 3: PCE signaling and interaction between PCE, policy and AA servers for path computation.

tion is successful, the AA server computes the transit_nonce and the resource_allocation_token (see Section III.A.4) and provides the PCE with them. The PCE computes the local paths and includes them in the path computation response along with the domain identifier, the resource_allocation_token, and the transit nonce. In Section IV.B we define the extensions to the PCE protocol required to use it as carrier of these information. Information about the paths computed and the resource_allocation_token are also provided to the local policy server which stores them for admission control of the path setup request. A suitable timeout is applied to this information, after which they are removed from the policy server to avoid overloading.

Source domain: processing path computation response and issuing path setup requests.

Upon reception of the path computation response, the source domain verifies the validity of the resource_allocation_tokens issued by all the transit domains and the destination one before computing the optimal path. As introduced in Section III.A.4, the signed resource_allocation_token is used as payment receipt and a keyed-hash method is used to verify the identity of the issuers of the resource_allocation_tokens. If the verification is successful, the AA server computes a digital signature on each resource_allocation_token and returns them to the PCE. The PCE provides the GMPLS-enabled head-end node with the computed path along with the signed resource_allocation_tokens. These allow the GMPLS signaling to get authorization from each domain and are used as proof against payment by transit domains.

Transit and destination domains: processing incoming path setup requests.

As introduced above, setup signaling carries a list of signed resource_allocation_tokens, one per each domain. These tokens are used to restrict access to authorized setup signaling only. Upon receiving a path setup request, the edge node mines from this message the signed resource_allocation_token that refers to its domain and provides the policy server with it. If the resource_allocation_token is alive (available in the policy server), the policy server asks the AA server to verify the signature. The signature guarantees data origin, integrity and non-repudiation. If the signature is valid, the charging server is provided with the signed resource_allocation_token as proof against payment for the resource provided and the policy server sets the policy function of the edge node to allow access to the setup signaling.

B. Proposed PCE Protocol Extensions in Support of AA

The PCE protocol (PCEP) [6] is a request/response communication model for path computation signaling between PCEs. PCEP messages are composed of a common header describing the type of the message, its length and the version of the protocol used, followed by a list of objects. Each object has a common header describing the type of the object and its length and a body whose content depends on the type of the object. The body can contain a set of Type-Length-Value elements. The PCEP messages used to describe path computation requests and responses are the PCEreq and PCErep, respectively. The PCEreq message has two mandatory objects which are the END-POINTS object describing the source and destination addresses, and the BANDWIDTH object describing the bandwidth requested. The ERO (Explicit Route Object) is the mandatory object of a PCErep that encodes the requested end-to-end path.

In this paper we define two new PCE objects called AUT_REQ and AUT_REP to describe the AA content of the model proposed. The AUT_REQ describes the AA content issued by the source PCE while the AUT_REP object describes the AA content issued by each transit domain. The AUT_REQ (see "AutReq object", Figure 4) encodes the assertion list and the signature computed on the PCEreq message. Each element of the assertion list is described through a TLV element called Assertion_TLV (see "Assertion TLV", Figure 4) having the token identifier, the identifier of the transit domain and the source_nonce as sub elements. The AUT_REP (see "AutRep object", Figure 4) object contains four fields that carry the identifier of the transit domain, the identifier of the QoS service, the transit_nonce and the resource_allocation_token.

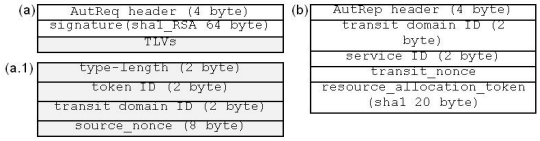

Figure 4: (a) AutReq object; (a.1) Assertion TLV; (b) AutRep object.

V. SIGNALING OVERHEAD AND PERFORMANCE

We study the performance of the model proposed in terms of increase in delay and load due to introduction of AA mechanisms in the PCEP signaling. We first perform simulations on the NSFNet topology [10], where each node represents a domain (e.g., ISP) and each link an inter-domain connection. Later we also show results for Internet-like topology, with higher number of domains and inter-domain links, in order to evaluate the scalability of the proposed model. We made the following assumptions regarding the traffic: (i) the arrival of path requests follows a Poisson distribution, (ii) each PCE has an independent request arrival rate, which is the same for all PCEs, (iii) all destinations are equally probable and (iv) the arrival rate at each PCE is equal to 1 request per sec for all simulations. We used the OpenSSL Library [11] to estimate the delay for signature verification and hash computation to be 200 μs and 60 μs respectively. Inter-domain paths were chosen based on the minimum number of hops. The parameters used to evaluate the model are: 1) the AA delay due to signature verification and hash computation in all the domains along the path, 2) the average AA request load, i.e., the number of AA requests per second that each domain has to process, and 3) the average AA signaling load. Simulations on the NSFNet topology (see Table 1) show that the average delay for PCE signaling with and without authentication is 14.27 and 14.83 ms, respectively, indicating an increase of 3.92 percent. The transmission delay of each individual link was set as 3.33 ms (~1000 km) which is significantly higher than the AA delay experienced, justifying the low increase observed.

	without AA	with AA
AA delay	14.27 ms	14.83 ms
Signaling load	79.03 bytes/s	286.59 byte/s
AA request load	-	2.14 req/s

Table. 1: Performance of the NSFNet.

For the same topology, the AA signaling load is 207.56 bytes per second while the request load is 2.14 requests per second. These results show that the model proposed is well suited for NSFNet like topologies.

In order to evaluate the scalability of the model proposed with increase in size and connectivity of the network, we use three sets of six different AS level topologies, with 100, 120, 140, 160, 180 and 200 domains respectively. The topologies in the three sets have different degrees of connectivity, with the ratio of number of inter-domain links (L) to number of domains as 2 in Set 1, 3 in Set 2 and 4 in Set 3. The Waxman [12] topology generation model was used to generate the random topologies with parameters alpha = 0.15, beta = 0.2, and the minimum distance scale in the topology was assumed to be 100 kilometers. All domains are placed randomly inside a square of edge length of 10,000 km. Figure 5 shows the variation of the average AA delay with the number of domains in the network for different degrees of connectivity. The AA delay depends on the average number of hops in the inter-domain path. We observe that the addition of domains in networks with a low average interconnections, such as the ones with 2 links per domain (L=2), may reduce the average hop count which causes a fluctuation in the AA delay. On the other hands, the hop count and therefore the AA delay is seen to increase linearly with the number of domains in networks with high degree of interconnectivity (L=3 and L=4).

Networks with low degree of interconnectivity have more hops between source and destination leading to larger average end-to-end delays for a given network size. Therefore the AA delay for Set 1 with two inter-domain links per domain on average (L=2) is greater than that for Set 2 (L=3) while Set 3 (L=4) has the smallest AA delay.

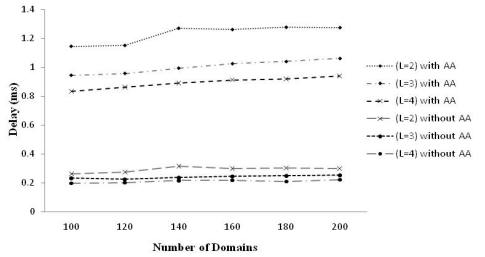

Figure 5: Delay for different network topologies.

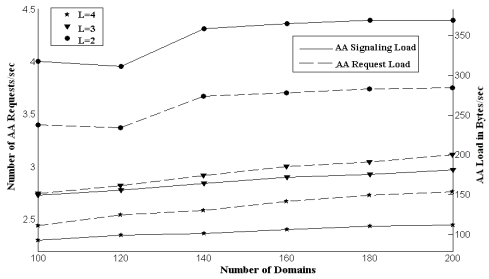

Figure 6: AA request load and AA signaling load for different topologies and degrees of connectivity.

The AA request and signaling load also depend on the number of hops in the inter-domain path. Figure 6 exhibits the linear increase in the request and signaling load with the number of domains for networks with high degree of interconnectivity (L=3, L=4). Fluctuations are observed in networks with L=2 links per node while increasing the number of domains from 100 to 140. It is also observed that increase in interconnectivity significantly reduces the average request and signaling load in the network.

The results show that the performance of the model increases with the degree of interconnectivity. Increase in signaling load, delay and AA computations were seen to follow a linear trend for networks with higher interconnectivity, suggesting that the model can scale well with size of the networks. For model with low interconnectivity, the trend observed for the increase was not uniform. However, the AA request load, signaling load and delay were observed to be within acceptable limits, suggesting that the model could be applied to these networks.

VI. CONCLUSION

In this paper we proposed the first inter-domain QoS path provisioning solution which integrates security features for inter-domain authentication and authorization in the PCE-based path computation framework. We defined a QoS provisioning architecture suitable for GMPLS networks which is composed of a control plane responsible for path computation, path setup and inter-domain AA and a management plane that controls inter-domain QoS agreements and charging functions. The AA solution proposed allows authentication and authorization between neighboring as well as non-neighboring domains and couples AA for path setup with path computation. The security solution proposed includes features to guarantee integrity, privacy and non-repudiation of sensitive information exchanged between domains for QoS path provisioning. We studied the performance of the model proposed and we provided results demonstrating that it can scale in networks with different sizes and connectivity. The main objectives of our future work are to study the robustness of the model proposed against attacks from malicious parties and solutions that prevent or block such attacks.

VII. REFERENCES

[1] T. Lehman, J. Sobieski and B. Jabbari, “DRAGON: A Frameworkfor ServiceProvisioninginHeterogeneous GridNetworks,” IEEE Communications Magazine, Vol. 44, No. 3, pp. 84-90, March 2006.

[2] A. Farrel, J.P. Vasseur and J. Ash, “A Path Computation Element (PCE)-Based Architecture, ” IETF RFC 4655, August 2006.

[3] A. Farrel, A. Ayyangar and J. P. Vasseur, "Inter domain Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineering - RSVP-TE extensions," RFC 5151, February 2008

[4] R. Douville, J.-L. Le Roux, J.-L. Rougier and S. Secci, “A Service Plane over the PCE Architecture for Automatic Multidomain Connection-Oriented Services,” IEEE Communications Magazine , Vol. 46, No. 6, pp. 94-102, June 2008.

[5] L. Gommans, F. Dijkstra, C. de Laat, A. Taal, A. Wan, T. Lavian, I. Monga and F. Travostino, “Applications drive secure lightpath creation across heterogeneous domains,” IEEE Communications Magazine, Vol. 44, No. 3, pp. 100-106, March 2006.

[6] J.P. Vasseur and J.-L. Le Roux, “Path Computation Element (PCE) Communication Protocol (PCEP),” draft-ietf-pce-pcep-15.txt (work in progresss), September 2008.

[7] J.P. Vasseur et al., “A Backward Recursive PCE-based Computation (BRPC) Procedure To Compute Shortest Constrained Inter-domain Traffic Engineering Label Switched Paths,” draft-ietf-pce-brpc-09.txt (work in progress), April 2008.

[8] E. Mannie, “Generalized Multi-Protocol Label Switching (GMPLS) Architecture,” RFC 3945, October 2004.

[9] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels," RFC 3209, December 2001.

[10] NSFnet topology, www.optical-network.com/topology.php.

[11] OpenSSL Project, http://www.openssl.org/source/.

[12] B. Waxman. Routing of Multipoint Connections. IEEE J. Select. Areas Commun., SAC-6(9): 1617-1622, December 1988.