Multi-domain Information Exchange in a Bandwidth on Demand Tool

Giorgos Adam, Christos Bouras, Ioannis Kalligeros, Kostas Stamos, and Giannis Zaoudis
Computer Technology Institute and Press "Diophantus", N. Kazantzaki Str, University Campus 26504, Rio Greece
Computer Engineering and Informatics Dept., University of Patras
kallige@ceid.upatras.gr,{adam,stamos,bouras,zaoudis}@cti.gr

***Abstract*—In this paper we describe the multi-domain information exchange aspects of the AutoBAHN tool, which is used for implementing a Bandwidth on Demand service as part of the GEANT project. This project was driven by the increasing demand for reliable and guaranteed data transportation for scientific purposes. In order to implement the process of information exchange in a multi-domain heterogeneous environment several design decisions have to take place, in order to balance and satisfy by the needs of information security, fast information dissemination, efficient request processing, reliability and robustness. The paper describes how AutoBAHN implements a flexible AAI in a distributed environment, how it exchanges and updates information and how it handles incoming requests in an efficient and timely manner. The multi-domain aspect of this effort stems from the fact that the tool is used in the production environment of several cooperating National Research and Educational Networks (NRENs) across Europe. Each NREN may have different underlying technologies, different policies and different use cases, but they all have to coordinate through usage of the AutoBAHN tool in order to produce a multi-domain service of circuit reservation.**

***Index Terms*—Bandwidth on Demand; AAI; Security; Quality of Service; Performance Evaluation**

I. INTRODUCTION

The GN3 and GN3plus European projects [1] are research projects funded by the European Union and Europe's NRENs (National Research and Education Networks). They aim at building and supporting the next generation of the pan-European research and education network, which connects universities, institutions and other research and educational organizations across Europe and interconnects them to the rest of the Internet using high-speed backbone connections.

In the context of GN3, a Bandwidth on Demand (BoD) service is being developed, and the service is supported by the AutoBAHN tool. The aim of the service is to provide dedicated channels for data transport, which are necessary for demanding applications and research fields such as radio astronomy, high-energy physics and general Grid applications with strict demands for the provisioning of guaranteed and dedicated capacity. The provisioning of such circuits is done dynamically, so that both the performance limitations of IP networks and the inflexibility of fixed circuits are addressed as it is also presented in [2]. As soon as a circuit's resources are no longer necessary, they are released for another potential transfer between different end-points utilizing the same resources.

The AutoBAHN system is capable of provisioning circuits in the heterogeneous, multi-domain environments that constitute the European academic and research space and allows for both immediate and advanced circuit reservations. The overall architecture of the AutoBAHN system, its goal and the network mechanisms it employs are thoroughly presented in [3]. A similar architecture is also presented in [4].

This paper examines all the appropriate mechanisms that exist among AutoBAHN's different modules and also the external tools and technologies that are used for the purpose of exchanging information which is needed for AutoBAHN to complete its task and reserve a circuit with specific bandwidth across a network path that connects two end points. In such a complicated, modular and robust application it is a common fact that there is a lot of information that needs to be addressed, processed and exchanged. We will discuss this issue from the aspects of authorization and authentication infrastructure (AAI), secured communication between system components and a common registry service which holds data accessible to everyone who is involved. We will also evaluate the performance of the tool in two different real life simulation scenarios and prove that AutoBAHN can handle properly large number of submitted reservations in a two-domain testbed.

The rest of the paper is structured as follows: In Section 2 we analyse related work while Section 3 describes the GN3 project. Section 4 presents the general design implementation for AutoBAHN BoD system and in Section 5 we present the general architecture and the procedures that take place during authentication and authorization process. Section 6 describes the way that communication between system components can be considered secure. In Section 7, we examine the use of Lookup Service as a common registry mechanism and how it is used by AutoBAHN instances in order to exchange information. In Section 8 we describe the process of converting the actual network topology to a more abstract representation of the resources. Section 9 is dedicated to the evaluation of performance metrics regarding network traffic, processing time for pathfinding,

doi:10.4304/jnw.9.5.1075-1085

resources reservation and memory consumption. Finally, Section 10 concludes the paper and Section 11 presents future fields of this study.

II. Related Work

The AutoBAHN BoD system has been influenced by a number of other projects dealing with similar challenges for bandwidth on demand provisioning. In this section we present some of the most closely related ones.

The Dynamic Resource Allocation across GMPLS Optical Networks (DRAGON) project [5] is also conducting research and developing technologies to enable dynamic provisioning of network resources on an inter-domain basis across heterogeneous network technologies. It mainly deals with GMPLS enabled domains and in a smaller scale compared to AutoBAHN. It consists of almost 100 miles of dark fiber configured as two intersecting rings within the Washington DC metropolitan area and five core nodes incorporate wavelength selective switching capabilities in the form of multi-degree "reconfigurable optical add drop multiplexors" (ROADMs). These ROADMs are built by Movaz Networks and support the GMPLS routing and signaling protocols for dynamic provisioning. The DRAGON testbed is also connected to several other networks including Abilene, the Internet2 Hybrid Optical Packet Infrastructure (HOPI) and the Global Information Grid - Experimental Facility (GIG-EF).

One of DRAGON's core components is the Network Aware Resource Broker (NARB). The NARB provides several important functions to enable routing, path computation and signaling in this environment. The NARB is an agent which represents a local Autonomous Domain (AD) and acts as a protocol listener to the intradomain routing protocols. In this implementation, the intradomain protocol is OSPF-TE [22]. AutoBAHN was also using OSPF routing protocol for interdomain topology exchange which was later deprecated by Lookup Service. The NARB is also responsible for inter-domain routing. The NARB's utilize a modified version of OSPF-TE to share a link state database between domains. This inter-domain topology exchange can be based on the actual topology as discovered by listening to the local OSPF-TE protocol, or optionally based on an "abstracted" view of the domain topology (generated by configuration file or automatic synthesis of the OSPF-TE link state database). Domain abstraction provides mechanisms for an administrative domain to advertise to the outside world a highly simplified view of its topology. This allows domains to hide their real topologies as well as minimize the amount of external updates required. The trade-off is reduced accuracy for path computations. AutoBAHN is also adopting this approach of topology abstraction in order to unify and distribute the actual network elements into a more simple form without transmitting sensitive network information. The NARB also plays an important role in distribution of policy information to LSR's so that appropriate action can be taken when processing provisioning messages. To accomplish this, the NARB translates the complex AAA and schedule information located in the 3D RCE into a simple policy directive which is distributed to the appropriate LSR's.

Another highly active BoD project is ESnet's On-Demand Secure Circuits and Advance Reservation System (OSCARS). OSCARS provides multi-domain, high-bandwidth virtual circuits that guarantee end-to-end network data transfer performance. As of November 2010, ESnet traffic topped 10 petabytes a month. In 2010, ESnet operated over 30 (up from 26 in October 2009) long-term production OSCARS virtual circuits supporting scientific areas including High Energy Physics: (Large Hadron Collider) Computational Astrophysics (OptiPortal), Biological and Environmental Research, Genomics, Climate (GFD and Earth Sciences Grid). Approximately 5000 in total OSCARS virtual circuit reservations have been created for demos, transient experiments, and projects, etc. but 5000 are not all currently in use today [28].

OSCARS supports user driven advanced reservations of dynamic VCs at layer 2 (Ethernet VLANs), and layer 3 (IP). In this capacity, OSCARS is used as a domain controller for network resources within ESnet. It also functions as an Inter-Domain Controller (IDC) which has the capability to communicate with other domain controllers like AutoBAHN does. One of the core objectives of OSCARS is to provide easily used functionality that allows application programmers and end-users without a network engineering background to set up reservations for VCs. These can be requested on-demand, or scheduled in advance to support a workflow pipeline, for example to coincide with experimental data generation [28].

OSCARS is considered the most closely related BoD tool comparing to AutoBAHN. The latter's main advantage is that it can operate across different network technologies such as MPLS, SDH and Ethernet where no other tool can achieve the same level of interoperability. Furthermore AutoBAHN has been designed from scratch with an emphasis on a heterogeneous multi-domain environment, with different administrative domains that wish to collaborate for the specific purpose of Bandwidth on Demand without sharing more information than necessary.

III. GN3

GÉANT brings together over 400 participants from 32 NRENs, TERENA, DANTE, and over 20 subcontractors and third parties. It provides a dedicated, high capacity, 50,000 km data network that brings together 40 million users in research institutions across 40 countries, underpinning critical projects that would previously have been impossible without its reach, capacity and reliability. In the past, average bandwidths were at the order of 155 Mbps but since 2004, GÉANT became able to transmit data at speeds of up to 10 Gbps as standard. Even faster capacities of 40 Gbps have been successfully piloted across distances of up to 750km, and 100 Gbps connections are in planning.

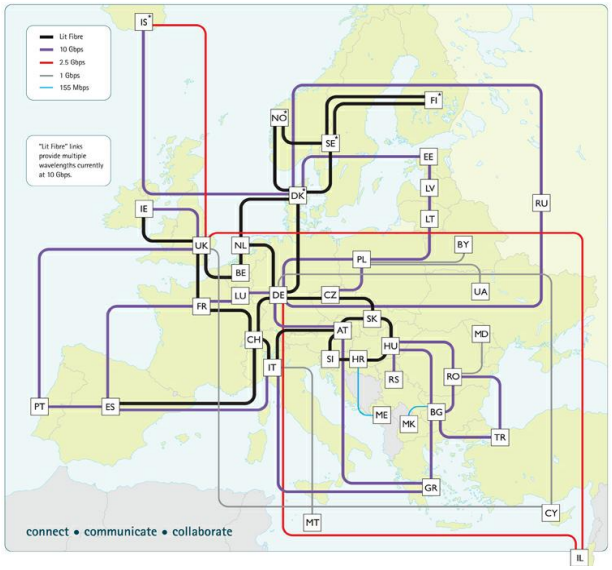

Figure 1. GÉANT's topology map (www.geant.net)

The GEANT BoD service is an end-to-end, point-to-point bidirectional connectivity service for data transport. The BoD service allows users to reserve bandwidth on demand between the end points participating in the BoD service. The data transport capacity dedicated to a connection can range from 1 Mbps up to 10 Gbps. This service is offered collaboratively by GÉANT and a set of adjacent domains (NRENs or external partners) that adhere to the requirements of the service. These joint networks form a multi-domain area where the service is provided.

The service is designed for situations where users have frequent transfer of large data sets between two end points. The data transport capacity is negotiable per request and is either accepted or rejected when the request is made. The service offers a high security level in the sense that the carried traffic is isolated from other traffic. It has to be noted that the traffic is isolated at logical layer and not necessarily at physical layer. It means that the core network will carry data from multiple users, but there will be no "crosstalk" between these traffic streams. From the users' perspective, each instance of the service is a virtual circuit between the two endpoints among which the traffic is exchanged in a manner isolated from other data flowing within the involved networks. The bandwidth offered by the service is not over-subscribed in any of the networks carrying the traffic hence making it possible to deliver deterministic service in terms of throughput capabilities.

Figure 2 presents the data plane for the end-to-end transport of the Ethernet Frames by the BoD service. The two End-Users are connected on their respective End-User domains. The BoD Service area comprises several participating domains. The BoD Service receives transports and delivers Ethernet Frames between Service Demarcation Points (SDPs) "A" and "Z", where the End-User sites are connected. The end-to-end transport service is implemented through several intra domain transport services, stitched together at the Service Stitching Points (SSPs) between the domains. Both SDPs are connected using infrastructure provided by the federated domains. The domains that participate in the joint BoD service offer are connected together at SSPs.

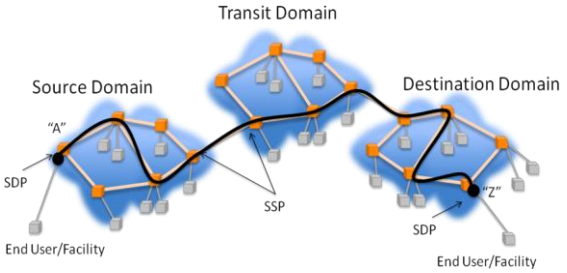

Figure 2. End to end path of BoD instance in multi-domain infrastructure. This BoD service for GEANT is supported by AutoBAHN system [22]

IV. AUTOBAHN BOD SYSTEM

The AutoBAHN system has to coordinate a distributed circuit reservation process. Participating domains may not wish to advertise their internal structure, technology or topology details. This has led to the adoption of a distributed implementation, where each participating domain hosts an identical instance of the AutoBAHN software. For the purpose of flexibility, the software has been designed in a modular way, and is comprised of independent components, each taking care of a specific set of functionalities. The multi-domain aspect of the system is mainly handled by the Inter-Domain Manager (IDM), a module responsible for inter-domain operations of circuit reservation on behalf of a domain. This includes inter-domain communication, resource negotiations with adjacent domains, request handling, and topology advertisements.

To build a real end-to-end circuit, the Domain Manager (DM) module is also required to manage intra-domain resources. The DM has an interface to the local IDM from which it undertakes all intra-domain functions (abstracting the topology towards the IDM, scheduling and pre-reserving resources, monitoring etc.). This southbound interface of the DM is the part of the AutoBAHN system that needs to be tailored to the domain-specific conditions. The DM also has an interface towards the Topology Abstraction module. Its role is to take the actual domain topology and produce a generic graph (without any device details) that can be safely shared with any other domain participating in the service. The purpose of this topology abstraction process is two-fold:

Make sure that each domain's detailed network topology is not advertised to the rest of the domains that participate in the service, which may not be desirable.

Limit the size of the overall topology that is used for the initial inter-domain pathfinding, thus making the algorithm faster (in exchange for sub-optimal path computation results). A similar issue is discussed in [32].

In each domain, the data plane is controlled by the DM module using a range of techniques, including interfaces to the Network Management System (NMS), signalling protocols or network elements. As part of the AutoBAHN, a dedicated and independent Technology Proxy module

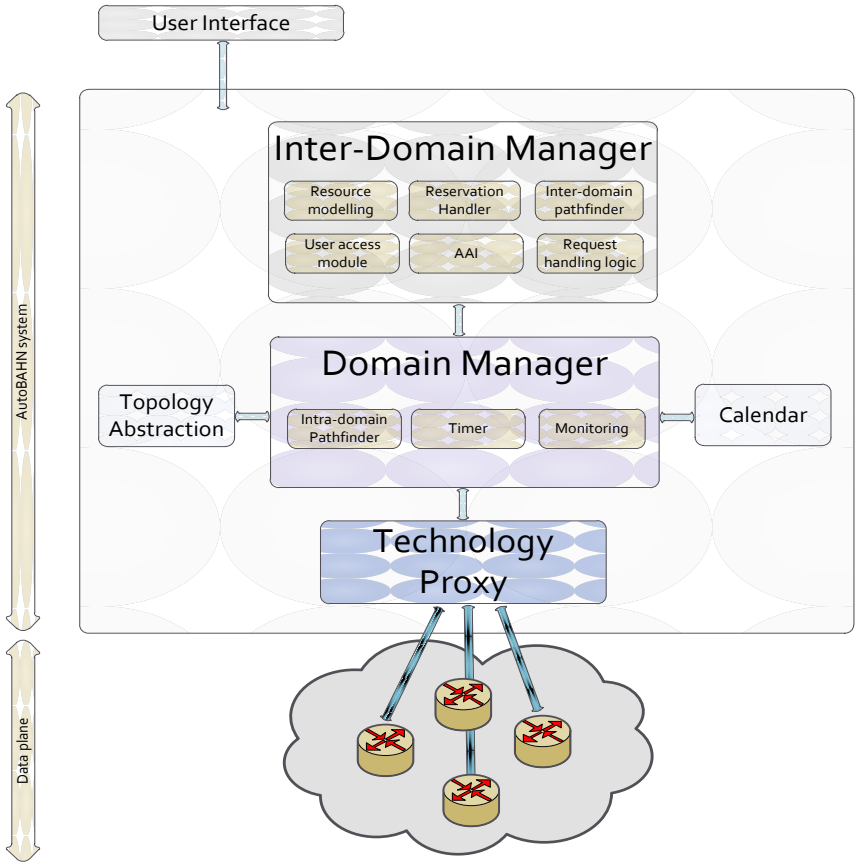

Figure 3. Basic architecture of AutoBAHN

allows the AutoBAHN system to support a range of technologies and vendors according to domain and global requirements. The Technology Proxy is the only module that is not identical between various instances, but has to be tailored to each domain's network technology characteristics. Thus, this modular approach allows the custom part of the AutoBAHN system to be reduced to the minimum possible functionality set. Furthermore, AutoBAHN provides the framework which takes care of typical technical implementation aspects of the Technology Proxy module (such as communication to other modules and reservation request parameter processing) so that the only work that needs to be done separately per domain is limited to specifying the underlying network equipment configuration commands.

Access to the Bandwidth on Demand service by end users is achieved through a web-based User Interface (BoD Portal), which can communicate to IDM instances at all participating domains. Similarly, an API (Application Programming Interface) is available for machine-to-machine communication that can be used to by-pass reservation request via the BoD Portal.

V. AUTOBAHN AAI

Authentication and Authorization in a multi-domain environment is a challenge due to the distributed nature of the service and the heterogeneity of policies among the domains. It is also a vital component of a service that can reconfigure network devices and has deep level of access to the network production data plane. AutoBAHN uses part of GÉANT AA Framework for this purpose, which provides several implementation choices regarding the Authentication and Authorization providers that can be used. Each of these choices has been considered taking into account the characteristics of a multi-domain BoD service. AutoBAHN can be configured to use XML or Atlassian Crowd [18] for Authentication and User Attributes Provider. It can also support the existing eduGAIN [16] infrastructure for authentication and authorization. In addition, LDAP or Relational Databases can be used as Authentication and User Attributes Providers. They also have the additional capability of supporting Access Control Lists for flexible definition of authorization policies. The rationale for this selection is that using local XML as an Authentication and Authorization Provider is a fast and efficient way to handle users centrally and configure a consistent cross-domain policy during the initial service rollout. However it is not scalable for a fully functional production service, where a dedicated solution such as LDAP or Atlassian Crowd is needed. Finally, eduGAIN is a promising effort from GEANT that provides the most flexibility and convenience upon its wide adoption by end users due to the Single Sign-On (SSO) feature.

A. User Authentication

The AutoBAHN system has been designed in such a way so that multiple authentication methods may be used in a modular way as described above, based on the Spring Security Java framework [17].

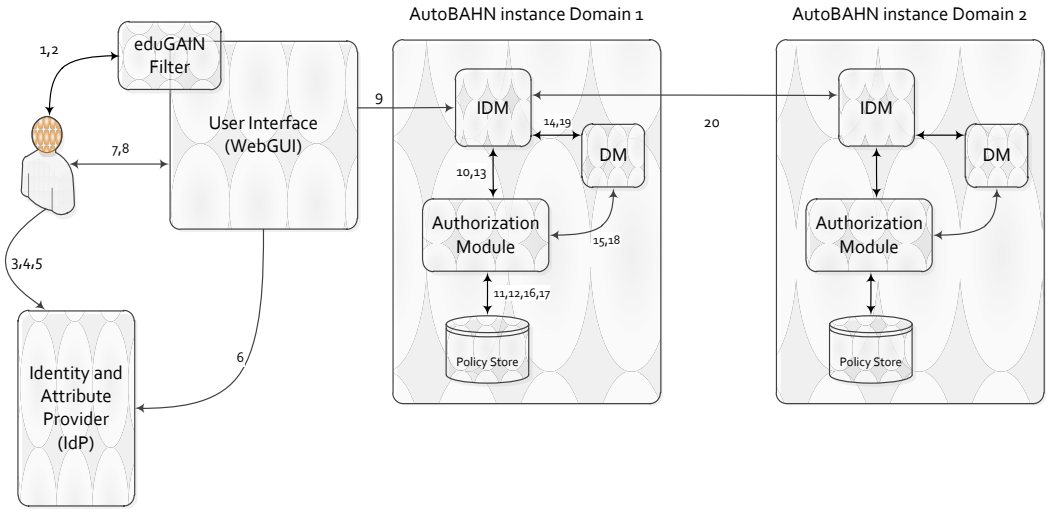

Figure 4. The user creates a reservation

The main system components that are invoked during the authentication procedure are the Authentication Manager and the Authentication Provider. The user submits his credentials to the AutoBAHN Server and an Authentication Request is created at server-side. This request is sent to the Authentication Manager which is responsible for forwarding requests through a chain of Authentication Providers. The provider will request from the UserDetails Service to provide the granted authorities for this user. These authorities are later used at the Authorization phase. The Authentication Manager receives back the result of the described procedure and decides whether the authentication is successful or not.

When a user wishes to connect to the graphical environment for making a resource reservation, his supplied credentials can be authenticated against any Authentication Provider like LDAP or Crowd, against the eduGAIN Single Sign-On (SSO) infrastructure, or even against a local XML file containing user information. The authorization procedure is then able to cooperate with interchangeable authentication modules, as long as the authentication provider also supplies the necessary attributes that enable authorization decisions. As the eduGAIN scenario is the most complicated and interesting one, it is described in more detail below.

In principle, when a user tries to access the AutoBAHN system, the (human) user is redirected to the (SSO) service of his/her federation. Then the user is authenticated through the federation software which sends the SSO response and SAML 2.0 authorization back to the AutoBAHN system. The response contains both authentication and authorization information as SAML 2.0 attributes. Finally, the AutoBAHN system checks the SSO response and SAML 2.0 attributes and responds to the user with a permission or denial to access the resource. The attributes that are transmitted are the following:

Identifier: A unique id of the user wanting to make a reservation. This could be either the name or the email of the user, or a combination of both.

Organization: The organization/domain/federation of which the user is a member.

Project Membership: This attribute contains a specified value (e.g. AUTOBAHN) that demonstrates that this user is an authorized AutoBAHN user.

Project Role: This attribute offers granularity in terms of the subset of available actions that the user is allowed to perform, and can contain values such as Service user, AutoBAHN administrator, etc.

The various project roles currently supported are:

Service User (people from e-science communities, other BoD systems, external client applications) that become “service owners”

Network Administrator (people responsible for the data plane – the underlying data network infrastructure)

Autobahn Administrator (people responsible for the control plane software)

In AutoBAHN system, the above attributes are considered to be equivalent of granted authorities meaning that based on the policy that is defined by AutoBAHN’s administrator, those attributes also define the appropriate jurisdictions and capabilities that a user can have during the usage of the system. Similar policy-based policy resource allocation issue is presented in [19].

B. Multi-domain User Authorization

Authorization is the function of specifying whether a user has the access rights to perform an action on the system resources. AutoBAHN implements multi-domain user authorization, which means that the above procedure is done on every single Domain Manager.

After the authentication phase, the user is able to request access to the available resources. The authorization procedure takes place at the Domain Manager which determines whether the user is authorized to access the requested resource. This decision is based on the access policies that each DM has defined.

For operations that are decided along a multi-domain path, such as a reservation request (Figure 4), there are multiple Domain Managers. Thus, the decision has to be taken in every domain along the reservation path, based on user attributes that have to be transmitted with the reservation request and mapped to the policies implemented by each domain.

As described earlier, the user attributes are retrieved during the authentication phase. These attributes are then forwarded to each domain along the reservation path at server-side. Before a request is examined by the system at each domain, the attributes are compared against the policy module to check whether the user has the required privileges. The policy is based on logical operations among the user attributes: identifier, organization, project membership and role.

Figure 4 shows the whole procedure for authentication and authorization when a human user want to create a service request. At step 1, the Service user (through a web browser) tries to access the service submission interface in the WebGUI and the request is intercepted by the eduGAIN filter. The eduGAIN filter redirects the user to his local AAI (step 2). The user's web browser sends an http request to the IdP server (step 3). The IdP server sends to the web browser a page to authenticate the user and the user submits his credentials to the IdP server (steps 4 and 5). The IdP server redirects the user to the WebGUI request page, and associated attributes are also sent (step 6). The user fills in necessary parameters and submits the service request which may bundle multiple circuit reservation requests (steps 7 and 8). The WebGUI forwards the service request and user attributes to the initiating IDM (step 9). The IDM deals with each reservation in the service separately. For the first reservation, it forwards the user attributes and reservation parameters to the AuthR module and the AuthR module constructs a policy evaluation query (steps 10 and 11). The query is checked against the existing policies stored in the Policy Store and the AuthR module returns an answer (steps 12 and 13). Assuming the response is to permit the request, the IDM forwards it to the DM for intradomain checking (step 14). The DM calculates possible paths and forwards the reservation parameters to the AuthR module (step 15). The AuthR module constructs a policy evaluation query and the query is checked against the existing policies stored in the Policy Store (steps 16 and 17). The AuthR module returns an answer and the DM replies to the IDM about the feasibility of the reservation (steps 18 and 19). The IDM forwards the request and the user attributes to the next domain along the path for further processing (step 20). Then the process is repeated for all domains until the first reservation request has been evaluated and then for all reservations within the service request.

VI. Secured Communication between Components

Each AutoBAHN instance initializes different communication channels among its modules with the usage of web services. Through these channels, crucial information is exchanged such as reservation request attributes, user and domain attributes, network topology elements (ip addresses, ports, capacity etc.). To ensure a secure and trusted communication between system components, WS-Security standard is used in addition with Edugain PKI infrastructure.

WS-Security (Web Services Security, short WSS) is a flexible and feature-rich extension to SOAP to apply security to web services. It is a member of the WS-* family of web service specifications and was published by OASIS.

The protocol specifies how integrity and confidentiality can be enforced on messages and allows the communication of various security token formats, such as SAML, Kerberos, and X.509. Its main focus is the use of XML Signature, XML Encryption and XML Timestamp to provide end-to-end security and in our case all three options are available for use.

WS-Security makes heavy use of public/private key cryptography [20]. A central problem for use of public-key cryptography is confidence (ideally proof) that a public key is correct, belongs to the person or entity claimed (i.e., is 'authentic'), and has not been tampered with or replaced by a malicious third party. The usual approach to this problem is to use a public-key infrastructure (PKI), in which one or more third parties, known as certificate authorities, certify ownership of key pairs.

AutoBAHN makes use of eduGAIN PKI for validating the identity of the components. The trust establishment process is enabled by means of using TLS connections for each eduGAIN interaction and including XML-Sig digital signatures for the appropriate protocol elements and assertions.

eduGAIN inter-component trust is based on X.509 certificates. It is rooted at a set of Certification Authorities (CA) created and maintained within the project. This set will be referred to as the eduGAIN truststore and all AutoBAHN components accept any of the CAs contained by the truststore as valid roots of trust. CAs in the eduGAIN truststore conform to the eduGAIN Certificate Policy, a document defining the rules and procedures agreed by the eduGAIN participants to rely on digital public certificates issued to the components of the infrastructure.

At least one of these CAs will be specifically established and run by the project. This root CA is referred as the eduGAINCA. The self-signed certificate of the eduGAINCA is the minimum content of the eduGAIN truststore.

A. Trust Validation Procedures

Trust validation is performed by eduGAIN components according to a two-step procedure:

The received certificate shall be evaluated to check whether the whole trust path correctly resolves to the eduGAIN root of trust.

The eduGAIN component identifier contained in the Subject Alternate Name extension of the received certificate shall be evaluated against the metadata available for this interaction. It must match with the component identifier as stored in these metadata.

A failure in any of the verifications above causes a rejection of the requested operation with a TrustError result. This procedure implies that, for a proper trust evaluation, all metadata exchange through the MDS must

contain the eduGAIN component identifiers applicable in each case.

Unless otherwise specified in the corresponding profile, all connections between any two eduGAIN components uses TLS and perform two-way certificate validation (both initiator and responder) according to the above procedures.

Validation of the certificates associated with XML Signatures follow the procedures described above.

In principle, when the client module wants to communicate with another module (the resource), it sends its request to the required resource along with its X.509 certificate signed by eduGAIN CA. The resource authenticates the client by validating its certificate using eduGAIN API. The certificate contains identification information that allows the resource to authenticate only designated clients.

Below the detailed procedure in the context of the AutoBAHN system for the trusted communication between AutoBAHN modules is presented.

The AutoBAHN module that wants to communicate (client) must have a certificate, so no interaction for credentials is needed. The X.509 certificate is issued by a Certificate Authority (CA) subordinated to one of the eduGAIN roots of trust.

The client module sends its request and the certificate to the resource.

The resource module performs trust validation by checking that the whole trust path of the certificate correctly resolves to the root(s) of trust defined by eduGAIN.

The resource checks that the client module is allowed to access it.

The resource provides the requested answer to the client module.

VII. Lookup Service

The Lookup Service (LS) is a key element of the perfSONAR measurement framework which is also a GEANT product and it allows every independent service to be a visible part of the system. In essence, the LS acts as a service directory for perfSONAR, where services can advertise themselves (provide their lookup information) and requestors are able to find any service they need.

In Autobahn, the Lookup Service is utilized as a common centralized registry space where all instances have to ability to write some appropriate attributes that needs to be advertised to the rest of the group. More specifically, LS is used in order to exchange the following:

The web services endpoints of the IDMs

The abstract topology

The edge links

All those descriptions and information is defined using XML schema and encoded into XML. Storage of these descriptions is performed in native XML databases and querying is performed using XPath and XQuery.

The LS can be considered as a messaging service through which each participant can inform others about crucial information. Because the information resides as XML, distillation is performed through XSLT transformation.

A. Core Functionalities

The following are the main functionalities of the LS.

Location of IDM modules for neighboring domains: In order to perform an inter-domain reservation, the IDM modules of the corresponding domains have to communicate (in a chain-like fashion). The unique identifier of an AutoBAHN domain is defined by the web service address of its specified IDM, as the identifier is in the form of a URL, containing the DNS name of the host running the IDM system. This approach implicitly uses the existing DNS service but requires manual insertion of the URLs. Lookup service in this case helps in the way that each IDM is able to retrieve the location (URLs) of the IDM modules of neighboring domains, using only its knowledge of the neighboring domain names.

Interdomain links identification: AutoBAHN faced the problem of matching an edge interface from one domain with the correct interface belonging to a neighboring domain in order to properly identify interdomain links. The lookup service made possible the transition to a semi-automated approach. The lookup service functions as the repository where edge interfaces are registered. The records in the lookup service contain the pair of domains that the interdomain link joins and an identifier for the edge interface.

Exchange of abstract topology between domains: Each domain uploads its abstracted topology to the LS, where it is merged with the global topology and then retrieved by all domains participating in the service.

In order to meet these requirements, the interface to / from the lookup service offers the following functionalities:

Registration of Domain name / IDM instance URL mapping: Upon initialization, each IDM instance registers this information to the lookup service.

Removal of Domain name / IDM instance URL mapping: Upon normal shutdown, an IDM instance notifies the lookup service and the lookup service removes the corresponding record.

Registration of Start domain / End domain / public edge port identifier mapping: Each AutoBAHN instance registers to the lookup service one record per edge interface, which contains the public name of the edge interface and the names of the domains it connects. If there are no multiple links connecting the same domains, this information can be used to automatically identify interdomain links.

Update or Removal of Start domain / End domain / public edge port identifier mapping: When there is a topology update, an interdomain link can be updated or cease to exist. The Topology Abstraction module propagates this information to the lookup service which correspondingly updates or removes the affected record.

All the above functionalities rely on the assumption that domain names are unique.

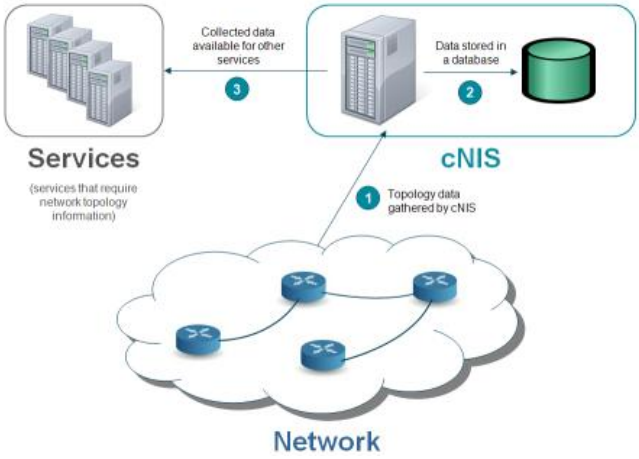

Figure 5. Basic cNIS functionality

VIII. Topology Abstraction

As it is already mentioned, each NREN deploys its own AutoBAHN instance. This results in a set of different network topologies in a total number that equals the number of running AutoBAHNs. The Domain Manager module keeps the actual topology information locally and is able to handle it appropriately but there is also a need to exist a total and more "simplified" union of these topologies and this has to be in public view from all AutoBAHNs in order to take it into account when they try to find a valid path to process a reservation request. This procedure takes place in Topology Abstraction module (TA).

During the abstraction, a representation of the physical topology is created for each NREN which only contains the most significant attributes and components of the actual topology. The abstract topology consists of Nodes, Links and Ports. Also, AutoBAHN creates Internal Identifiers for these objects which are linked locally to the real network topology data. This also helps in keeping away from public view information such as the IPs of the routers, their port names etc. mostly for security reasons but also to have a unique and unified naming convention scheme.

In order to describe the abstraction process, first, we need to define how AutoBAHN gets the network topology. For this problem, AutoBAHN operates in conjunction with another tool that is also part of GN3 project which is called Common Network Information Service (cNIS) [27]. The aim of the cNIS is to provide a unified repository of all relevant network information about a single administrative domain. cNIS was expected to be the "single point of storage", but in fact it is more than just a database. Apart from the internal functionality required for populating, validating and updating the database, it is equipped with modules for analyzing the topology data and presenting the data in a client-specified format (graphical, tabular or even XML for external applications). During the deployment phase, the NREN's administrator has to insert the physical topology in cNIS which then AutoBAHN reads it and creates the abstraction during start-up time. cNIS supports different network technologies such as Ethernet, SDH, MPLS and IP. It also provides the user tools in form of plugins for automatic network discovery which simplifies even more the whole process. The later enhances the ability of AutoBAHN to operate across different heterogeneous networks.

AutoBAHN reads the topology from cNIS, converts it on each own elements such as Nodes, Links etc. and then, creates an abstract representation of this topology which later on sends it to Lookup Service for other domains to be aware of the new available topology.

During the abstraction, the following objects are created; Interdomain/Client Ports, Virtual Ports, Client/Virtual Links and Client/Edge Nodes. The abstraction process first of all distinguishes the edge nodes, which are the nodes that have either a client (end-point) attached or a link to another domain. After that, it creates virtual links that form a one-to-one mesh that connects any edge node directly to the other.

For each object, an identifier is created based on information such as the name that was given to a node in combination with its ports or similar combinations. The resulting string is then represented in CRC32 format to keep the string always 8 characters long. In that way we ensure two things; first of all, the uniqueness of the identifiers and secondly, in every restart and since the abstraction process has to take place from scratch (meaning erasing the existing abstract topology), if we don't have any significant changes in the topology, we ensure that the result of the abstraction will be the same.

IX. Performance Evaluation

The purpose of this section is to show that AutoBAHN as a distributed system can handle use cases where multiple users submit requests to reserve a circuit and that information exchange is done efficiently enough in order not to affect end-user perception of the service. The following tests were conducted using actual software deployments, but with the underlying data plane (Technology Proxy) disabled. This means that control plane decisions did not lead to re-configuration of network devices (switches or routers). Our main goal was to investigate the performance of the software itself, and

not the performance of the network devices themselves which depends on the equipment and vendor used.

For the purposes of running the tests we created a testbed which included two interconnected domains (AutoBAHN instances) and a web portal to manage them and submit requests. The basic benchmarking tool that was used for measuring the performance is Apache JMeter [26]. It has the ability to create simulation tests for Web (HTTP/HTTPS), SOAP, databases and other types of requests. In our case, we are mostly interested in Web Services requests through SOAP for accessing main AutoBAHN resources and also HTTP request to test the response of AutoBAHN's graphical web interface.

We run individual tests to see the actual performance of each communication channel and processing unit but we also created a test scenario that invoked all modules from Client portal to the core mechanisms of AutoBAHN in order to have a realistic view of the processing time and cost that is needed in a situation that resembles production environment scenarios.

The experiments were performed on a standardized Virtual Machine that is recommended for production usage of AutoBAHN, using a single CPU and 2GB of RAM. Our main performance metrics are:

CPU utilization for core AutoBAHN system
Memory consumption
Java memory heap size
Number of threads
Response latencies
Throughput
Average response time

Also, for quality evaluation we will examine the reliability of the system through the percentage of successfully processed requests.

With JMeter, we simulated two test cases; the first one had to do with 50 users that submitted one request each in a total time of 25 seconds, resulting in 2 requests per sec. According to the expected usage of the service at production and current experience, this is a demanding but plausible scenario. As a second test, we decided to simulate an even more demanding scenario to verify scalability during rare or unpredictable burst demand periods. More specifically, this test simulated 200 requests in a time period of 50 seconds resulting in 4 requests per sec.

A. First Scenario (50 requests within 25 seconds)

During our first simulation scenario, we recorded a number of results in order to evaluate the system responsiveness and scalability.

Figure 6 shows the number of active threads in the BoD portal handling incoming requests. Its purpose is to visualize the rate of incoming demand for system resources. The demand is increasing as new requests are incoming and then this number remains steady since each request is processed and forwarded to the source domain until the moment where client portal has transmitted all the requests to the domains. Most of the time there are 2-4 active threads for processing requests.

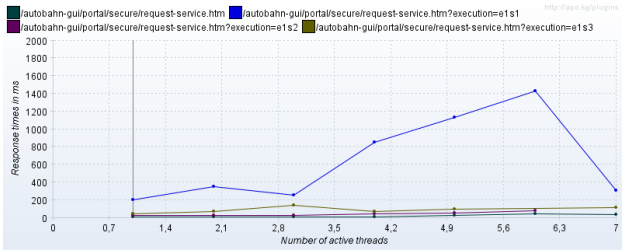

Figure 6. Response times vs threads

Figure 7 can be read in conjunction with the above information. Coloured lines indicate response time on different user activities on the service. We can see that for most of the experiment duration the response time for most actions does not exceed 0.2-0.4 seconds. Under heavy load (more than 4 simultaneous threads) response time may climb up to 1.4 seconds.

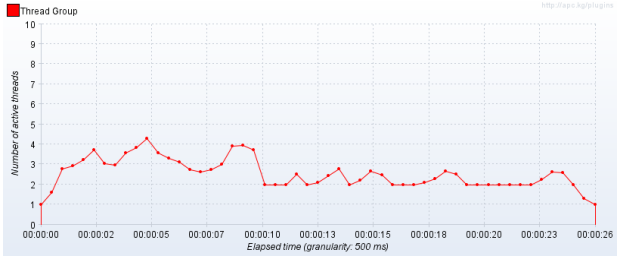

Figure 7. Number of threads for client portal

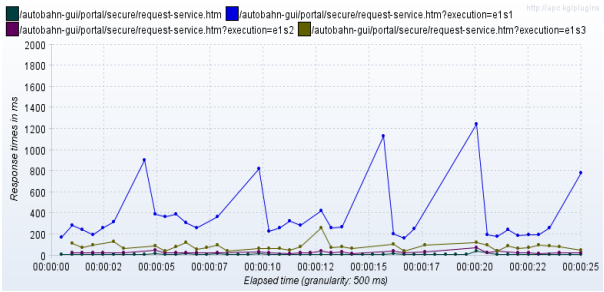

Figure 8. Response times over time

Figure 8 shows a similar result this time drawn against experiment time. The most time consuming steps of the requests process are when the end-user portal has to communicate with AutoBAHN and fetch information that needs to be processed and presented to the user.

B. Second Scenario (200 requests within 50 seconds)

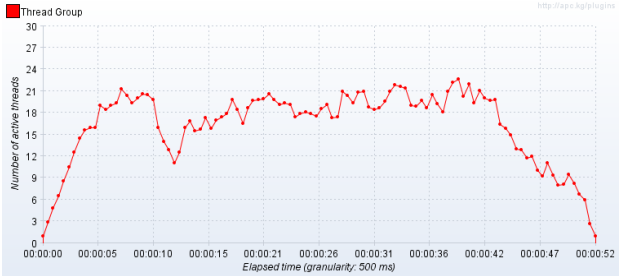

Figure 9. Number of active threads through Time

Increasing the rate of request arrivals by an order of 2, results to an increase to the number of simultaneous

active threads almost by an order of 5. This indicates the formation of a build-up of queued requests which however for the duration of the experiment is bounded by an upper threshold.

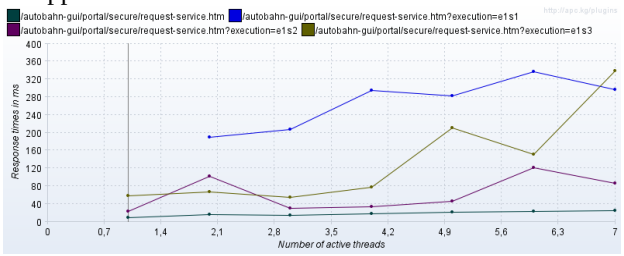

Figure 10. Response times VS threads

Figure 10 and Figure 11 demonstrate that response time still however remains low despite the increased number of simultaneous threads, which indicates that the request load is still well within the system's capability to handle without unbounded delay build-ups.

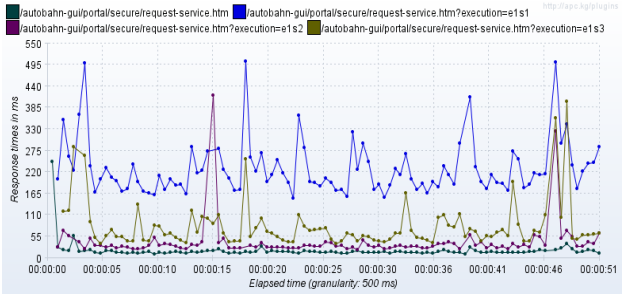

Figure 11. Response times over time

In general, the results presented above demonstrate that AutoBAHN is a system that can guarantee a successful bandwidth on demand service in terms of speed and reliability. AutoBAHN's multithreaded approach for handling the request as an autonomous process, results in storing them in a queue which later on tries to serve them in a sequential and serialized way. In AutoBAHN, these requests are stored in a FIFO queue and served in an autonomous manner through a reservation processor which created multiple threads to serve the requests.

In this way, we ensure a hierarchical approach in accommodating the reservation requests and also in situations where lots of concurrent requests are submitted, AutoBAHN can easily handle all of them based on the time of arrival and the available resources.

X. Conclusions

In this paper we presented the way that a distributed tool called AutoBAHN performs multi-domain communication exchange in order to support a federated BoD service. Lookup Service acts as a common registry space that is needed for all instances to have a single record for sensitive information such as the abstract topology. It is also a common discovery service because through Lookup Service, each AutoBAHN discovers its neighbors.

AAI is crucial in the application field of AutoBAHN and therefore it supports user authentication in addition to multi-domain user authorization approach ensuring that this communication is secured by WS-Security specification.

Performance evaluation of AutoBAHN has shown that it is capable of handling and process a large amount of simultaneous requests. Production usage is still significantly lower than the simulated loads due to the recent introduction of the service so these results have not yet been validated in a real production environment.

XI. Future Work

There are improvements planned for AutoBAHN tha aim to make it support the BoD service reliably and effectively. It is palnned to fully integrate it with supporting AA Infrastructure [21], handle topology updates in a fully dynamic manner and be able to reroute existing scheduled or active reservations in case of topology updates or NREN administrators who want to make changes in the allocated reservation paths because of heavy traffic or other causes.

Finally, the AutoBAHN tool is developed with the purpose of being interoperable with other similar tools and protocols such as OSCARS / InterDomain Controller Protocol (IDCP) [30] and Open Grid Forum's Network Service Interface WG (NSI-WG) [31].

References

[1] "GN3 European Project," [Online]. Available: http://www.geant.net/pages/home.aspx.

[2] Manish Mahajan, Ananthanarayanan Ramanathan, Manish Parashar, "Active resource management for the differentiated services environment", *International Journal of Network Management*, Volume 14, Issue 3, pages 149–165, May/June 2004

[3] M. Campanella, R. Krzywania, V. Reijs, A. Sevasti, K. Stamos, C. Tziouvaras and D. Wilson, "Bandwidth on Demand Services for European Research and Education Networks," *in 1st IEEE International Workshop on Bandwidth on Demand, San Francisco (USA)*, 2006.

[4] Vasil Hnatyshin, Adarshpal S. Sethi, "Architecture for dynamic and fair distribution of bandwidth", *International Journal of Network Management*, Volume 16, Issue 5, pages 317–336, September/October 2006

[5] F. Leung, J. Flidr, C. Tracy, X. Yang, T. Lehman, B. Jabbari, D. Riley and J. Sobieski, "The DRAGON Project and Application Specific Topologies," *in Broadnets, San Jose, California, USA*, 2006.

[6] Xi Yang, Tom Lehman, Chris Tracy, Jerry Sobieski, Shujia Gong, Payam Torab, Bijan Jabbari, "Policy-Based Resource Management and Service Provisioning in GMPLS Networks", *IEEE INFOCOM* 2006.

[7] C. Guok, "ESnet On-Demand Secure Circuits and Advance Reservation System (OSCARS)," *in Internet2 Joint Techs Workshop, Salt Lake City, Utah*, 2005.

[8] Chin Guok; Robertson, D.; Thompson, M.; Lee, J.; Tierney, B.; Johnston, W. "Intra and Interdomain Circuit Provisioning Using the OSCARS Reservation System", *Broadband Communications, Networks and Systems, 2006. BROADNETS 2006. 3rd International Conference.*

[9] "Argia," *Inocybe technologies Inc.*, [Online]. Available: http://www.inocybe.ca/.

[10] J. Wu, S. Campbell, J. M. Savoie, H. Zhang, G. v. Bochmann and B. S. Arnaud, "User-managed end-to-end lightpath provisioning over CA*net 4," *in Proceedings of*

the National Fiber Optic Engineers Conference (NFOEC), Orlando, FL, USA, 2003.
[11] C. Cavazzoni, "MUPBED Overview and Architecture," *in TERENA Networking Conference, Copenhagen, 2007.*
[12] L. Gommans, C. de Laat, and R. M[eijer, "Token based path authorization at interconnection points between hybrid networks aind a lambda grid," *in Proceedings of IEEE GRIDNETS 2005.*
[13] Y Demchenko, L. Gommans, C. de Laat, A. Tokmakoff, and R. van Buren, "Policy based access control in dynamic Grid-based collaborative environment," *in International Symposium on Collaborative Technologies and Systems*, pp. 64-73, 2006.
[14] J Vollbrecht P Calhoun S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, and D. Spence, "AAA authorization framework," *IETF RFC 2904*, Aug. 2000.
[15] J.J van der Ham, F. Dijkstra, F. Travostino, H.M.A. Andree, and C.T.A.M de Laat "Usin RDF to describe nfetworks'" *iGrid 2005 special issue, Future Generation Computer Systems*, vol. 22, no. 8, pp. 862-867, 2006.
[16] "Deliverable DJ5.2.3,3: Best Practice Guide - AAI Cookbook - Third Edition", [Online]. Available: http://www.geant2.net/upload/pdf/GN2-08-130-DJ5-2-3-3_eduGAIN_AAI_CookBook-1.pdf
[17] "Spring Security Framework," [Online]. Available: http://static.springsource.org/spring-security/site/.
[18] "Crowd," [Online]. Available: http://www.atlassian.com/software/crowd/.
[19] Kamel Haddadou, Samir Ghamri-Doudane, Yacine Ghamri-Doudane, Nazim Agoulmine, "Practical and analytical approaches for designing scalable on-demand policy-based resource allocation in stateless IP networks", *International Journal of Network Management*, Volume 22, Issue 2, pages 131–149, March/April 2012
[20] "Public Key Cryptography," [Online]. Available: http://en.wikipedia.org/wiki/Public-key_cryptography.
[21] Geant Forge, "GÉANT AA Framework," [Online]. Available: https://forge.geant.net/forge/display/AAI/Home.
[22] GN3 deliverable, "GN3 Bandwidth on Demand Service: Service Descriptions and Service Level Specification" "eduPKI", [Online]. Available: http://www.geant.net/SERVICES/ENDUSERAPPLICATIONSERVICES/Pages/eduPKI.aspx
[23] D. Katz, D. Yeung and K. Kompella, "Traffic engineering extensions to OSPF version," IETF Internet Draft, Work in Progress, draft-ietf-ospfospfv3-traffic-05.txt, March 2005.
[24] "Greek NREN (GRNet)", [Online]. Available: http://www.grnet.gr
[25] "Apache JMeter", [Online]. Available: http://jmeter.apache.org/index.html
[26] "Common Network Information Service", [Online]. Available: https://forge.geant.net/forge/display/cNIS/Home
[27] "OSCARS", [Online]. Avai able: http://www.es.net/services/virtual-circuits-oscars/
[28] Chin P. Guok, David W. Robertson, Evangelos Chaniotakis, Mary R. Thompson, William Johnston, Brian Tierney , "A User Driven Dynamic Circuit Network Implementation", DANMS 2008, IEEE conference 2008
[29] "IDCP", [Online]. Available: http://www.controlplane.net/
[30] "NSI-WG", [Online]. Available: http://www.gridforum.org/gf/group_info/view.php?group=nsi-wg
[31] Isabel Amigo, Sandrine Vaton, Thierry Chonavel, Federico Larroca, "Maximum delay computation for interdomain path selection", *International Journal of Network Management*, Volume 22, Number 2, page 162--179 - Mar/apr. 2012

Giorgos Adam was born in Athens, Greece in 1987. He graduated from the Computer Engineering and Informatics Department, University of Patras, Greece in 2010, and obtained his Masters Degree in Computer Science in 2013 from the same department. His major field of study was computer networks.

He is currently working for H&S Technology Solutions S.A. in Athens. His research interests include Information Retrieval, Data Mining and Mobile Ad Hoc Networks.

Christos Bouras was born in Kalamata, Greece in 1962. He graduated from the Computer Engineering and Informatics Department, University of Patras, Greece in 1985, and obtained his PhD in Computer Science in 1993 from the same department. His major field of study was computer networks.

He is Professor in the University of Patras, Department of Computer Engineering and Informatics. Also he is a scientific advisor of Research Unit 6 in Computer Technology Institute and Press - Diophantus, Patras, Greece. His research interests include Analysis of Performance of Networking and Computer Systems, Computer Networks and Protocols, Mobile and Wireless Communications, Telematics and New Services, QoS and Pricing for Networks and Services, e-learning, Networked Virtual Environments and WWW Issues.

Prof. Bouras has been a member of editorial board for international journals and PC member and referee in various international journals and conferences.

Ioannis Kalligeros was born in Kalamata, Greece in 1982. He graduated from the Computer Engineering and Informatics Department of University of Patras in 2009. His major fields of study were computer networks and routing protocols.

He was a member of Computer Technology Institute and Press "Diophantus" (CTI) in Patra until 2012, and he worked as a Software Engineer and researcher for the Greek Research and Educational Network (GRNET). Right now he is working as a Lead Release Engineer for Citrix Bytemobile. His research interests include bandwidth on demand services, network simulations, mobile networks optimization and media caching.

Kostas Stamos was born in Patra, Greece in 1978. He graduated from the Computer Engineering and Informatics Department of University of Patras in 2001, and obtained his Masters Degree in 2003 and PhD in 2007. His major fields of study were computer networks and admission control.

He is currently working for the Computer Technology Institute and Press "Diophantus" (CTI) in Patra, and for the Greek Research and Educational Network GRNET. His research interests include bandwidth on demand services, power management for wireless devices and network simulations.

Giannis Zaoudis was born in Chios, Greece in 1987. He graduated from the Computer Engineering and Informatics Department of University of Patras in 2009, and obtained his Masters Degree in 2001. His major fields of study were computer networks and transmission of video in wireless networks using cross-layer techniques.

He is currently working for the Computer Technology Institute and Press "Diophantus" (CTI) in Patra, and for the Greek Research and Educational Network GRNET. His research interests include cross-layer techniques over wireless networks and QoS in video transmission.