Policy Based Architecture for Class Based QoS Support

Carlos Rabadão[1, 2], Edmundo Monteiro[2]

[1] *Superior School of Technology and Management*
Polytechnic Institute of Leiria
Morro do Lena – Alto do Vieiro, 2411-901 Leiria, Portugal
crab@estg.ipleiria.pt

[2] *Laboratory of Communications and Telematics*
CISUC / DEI, University of Coimbra
Polo II, Pinhal de Marrocos, 3030-290 Coimbra Portugal
{crab, edmundo}@dei.uc.pt

Abstract

The main objective of the IETF Differentiated Services (DiffServ) model is to allow the support on the Internet of different levels of service to different sessions and information flows, aggregated in a few number of traffic classes. This model shows some security limitations that are inherent to the DiffServ model. This paper proposes an architecture for Authentication, Authorization, Admission control and Accounting (AAAA) of QoS client applications with dynamic identification of sessions and flows, based on the PBN paradigm. The proposal functionalities are described and analyzed in some detail, focusing on a set of extensions to the Common Information Model of the DMTF. The paper ends with the discussion of the main advantages of the proposal over existing solutions.

1. Introduction

In communication systems, the expression "Quality of Service" (QoS) is used to characterize the capacity of the system to support data flows with service guaranteed parameters (e.g. bandwidth, delay, jitter, losses) in a more or less strict way. The QoS mechanisms impose priorities and restrictions in the access of flows to available communication system resources. In the case of the DiffServ model [1] this traffic prioritization is supported by the identification of Classes of Service (CoS) done according specific fields of the header of IP packets [2]. As discussed in [3, 4] this approach has some security limitations, namely authentication and authorization.

The IETF DiffServ working group has considered some methods to reduce the inherent security limitations of the DiffServ model [4]. These include auditing and IPSec [5, 6]. However the vulnerabilities to security attacks, such as man-in-the-middle and Denial of QoS (DQoS), remain open issues [7].

To overcome the security limitations of DiffServ model, was proposed an architecture [8] for Authentication, Authorization, Admission control and Accounting (AAAA) of Quality of Service (QoS) client applications with dynamic identification of sessions and flows. This proposal addresses the issues related with the secure negotiation of QoS, in an intra-domain scope, namely admission control at the edge devices of DiffServ domains and the processes of authentication of the customers and authorization of flows associated with the resource reservation procedures. The QoS management inside this AAAA architecture is based on a set of policies, named Client Authentication, QoS Resources Authorization, Admission Control and Accounting policies. Large-scale systems may contain a huge number of users and resources. It is not practical to specify policies relating to individual entities – instead, it must be possible to specify policies relating to groups of entities and also to nested groups. It is also useful to group the policies pertaining to the rights and duties of a role or position within an organization such as a system administrator, a company CEO or mobile computing 'visitor' in a hotel. To implement this architecture we propose the adoption of the Common Information Model (CIM) standard [9, 10] of Distributed Management Task Force (DMTF).

Besides the present section, the paper has the following structure. Section 2 discusses relevant research work related with the improvement of management of QoS networks. Section 3 will be dedicated to the Information Model adopted to

implement the proposed architecture, based on the CIM model. Finally, Section 4 will be devoted to conclusions and directions for future work.

2. Related Work

The Common Information Model (CIM) is a conceptual framework for describing managed resources and management information in enterprise and service provider environments. CIM consists of over 1000 classes and associations defining managed resources that have been developed over the last eight years, by the DMTF. While the vast majority of the CIM standardization effort has been devoted to defining and improving the CIM models of resource instrumentation, work on management services is not so advanced [11]. Management services are generic, reusable, domain-independent functionality that should be defined and implemented only once; the best-known examples to date are the ISO–OSI Management Functions [12] for Network and Systems Management. Further examples of this kind of functionality are services for specifying and applying management policies, Service Level Agreements (SLAs), Service Level Objectives (SLOs), and capturing and manipulating managed resource metrics.

The Policy Core Information Model (PCIM) [13, 14] and CIM Policy Model [15], specified by the IETF Policy Framework Working Group and DMTF Policy Working Group, respectively, simplify the establishment of consistent behaviours between policies of different systems and allow to represent these policies independently of the manufactures and equipment types. In this way, it is possible to support policies at the services layer, independently of the network equipments features. Later, these policies will be converted in configuration parameters, according to the network equipments.

The QoS Policy Information Model (QPIM) [16], supported by the IETF, allow to model how policies can be established to manage and configure QoS mechanisms of network devices, like classification, marking, metering, dropping, queuing, and scheduling functionalities.

The CIM model, by the way of is “QoS sub-model”, follow the information model proposed by the IETF Policy Framework Working Group, around QoS architectures. Through the “User Model”, it also provides classes to manage and retrieve organizational data and information about "users" of services and their credentials, allowing the modelling of systems’ accounts for users, and the key services involved in managing authentication and authorization. However, to implement the proposed AAAA architecture with dynamic authentication and authorization of sessions and flows based on the client privileges’ and availability of QoS resources, some extensions will be proposed to the CIM User and Security Schema, at following session.

3. Policy for QoS Support Management

This section describes our AAAA architecture and the proposed extension to the CIM model for QoS negotiation and support.

3.1. Overview of the AAAA architecture

In the DiffServ model flow authentication is carried out on a per packet basis, at the entrance of each domain. Flow classification is supported by some of the IP packet header fields. As said before, this approach has some security limitations that are inherent to the DiffServ model.

Being the edge routers (ER) the responsible for the admission and packet marking according to flow’s quality of service, they are the most vulnerable element to attacks and security holes.

To overcome these limitations, was proposed an architecture [8] for QoS negotiation with authentication, authorization, admission control and accounting of client applications in a dynamic way, at the entrance of DiffServ domains. The architecture will basically focus in questions of secure negotiation of QoS, in an intra-domain scope, addressing the questions related to admission control at the edge devices of DiffServ domains and with the procedures for customer authentication and resources reservation authorization.

The proposed architecture, shown in Figure 1, has seven main modules: Policy Repository, QoS Client, Authentication, Authorization, Admission Control, Accounting and Router PEP (Policy Enforcement

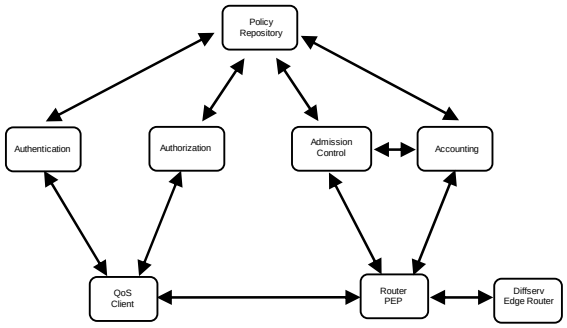

Figure 1. AAAA Architecture for QoS applications.

Point). The AAAA modules interact with the Policy Repository module of the domain and with the QoS Client component located at each client with QoS capabilities. The Router PEP and the DiffServ Edge Router (of the network provider) are the responsible for the enforcement of the QoS definitions in the border between the user domain and the network provider.

3.2. Information Model

The AAAA architecture proposal is based on the unification of the administration systems, using an integrated system for authentication, authorization and resources reservation with QoS and also accounting of resources, based on certain policies. These policies should be independent from the equipments, applications and networks. Afterwards, specific configurations will be created for the several manageable entities, in accordance with the specificity of each one. The result of the cooperation between the several entities will produce a consistent behavior for the system as a whole.

The Policy Core Information Model and CIM Policy Model, simplifies the establishment of consistent behaviours between policies of different systems and allows representing policies independently of the manufacturers and equipments types. In this way, it is possible to support policies at the services layer, independently of the network equipments features. The QPIM, supported by the IETF, allow modelling how policies can be defined to manage and configure the QoS mechanisms of devices. However, it does not support functionalities like client authentication and authorization that are prerequisites to implement the Authentication, Authorization and Accounting systems of our architecture.

We adopted the Common Information Model as the information model of our architecture due to a set of circumstances. Firstly, because the work developed by the IETF Policy Framework Work Group, around QoS architectures, heavily influenced the "CIM QoS Sub-Model", allowing it to model how policies can be defined to manage and configure the QoS mechanisms. Moreover, the "CIM User Model", proposed by CIM User and Security Working Group, provides classes to manage and retrieve organizational data and information about "users" of services and their credentials, allowing to model systems accounts for users, and the key services involved in managing authentication and authorization. However, some extensions must be proposed to "CIM User Model" in order to fully support the proposed architecture, namely the definition and the specification of authorization to establish a network data flow, from a subject to a target and the specification of the networks resources associated to the flows (QoS, maximum bandwidth authorized per session and duration of the flow).

The AAAA architecture proposes the clients' association to groups and roles, because when a huge number of clients is involved is rather difficult to manage privileges and policies associated to each one, individually. The creation of groups of clients reduces substantially the management effort, allowing to group persons or not human entities. These groups are implemented in the "CIM User Model" by it subclass `Group`. The definition of Role, represented in CIM by the subclass `Role`, allows representing a position or a set of responsibilities in an organization, which can be associated to persons or not human entities. This association can be done in an explicit way, associating identities to `Roles`. The position or set of responsibilities of a `Role` are represented as a set of privileges by instances of class `Privilege`. If an explicit relation does not exist, then relations should be established between the instances `Authorization Rule` and `Roles`, through `AuthorizationRule AppliesToRole` associations. These rules define in which way persons or not human entities (subjects) are authorized to perform certain roles and to which targets those roles should be applied.

3.3. CIM analyses for AAAA support

Next, will be analyzed the requisites for each system, in order to identify the CIM classes and subclasses able to satisfy it.

Authentication System

The information model requirements for the Authentication System are:

- Verification of the client's authenticity;
- Association of the client to Groups and Roles (subject);
- Verification of the privileges to receive credentials to the intended QoS authorization server (target).

The "CIM User Model", more specifically the `AuthenticationService` class, contemplates several security mechanisms to authenticate clients based on symmetric and asymmetric cryptographic techniques and on biometry. Through the definition of authentication policies, whose conditions define which are the authentication requirements, will be settled the mechanisms to the authentication process. The Clients'

association to Groups and Roles (subjects) are also supported as shown above. The verification of the privileges to access to the authorization server can be implemented through the use of the class `AuthorizationService`, being possible to create, modify and eliminate instances `Authorized Privilege`, gotten from the class `Privilege`, which is the base for all the types of allowed or denied activity, by Role or Identity. The class `Previlege` allows to define if a certain resource (target) can or not be used by one identity (subject), giving in detail the set of allowed activities.

Authorization System

The information model requirements for the Authorization System are:

- Validation of the credentials presented by the Client;
- Association of Clients to Groups and Roles;
- Authorization of QoS resources.

The two first ones are in all similar to the ones of the authentication system, support by the CIM model. However, with the actual class `Previlege`, it is not possible to specify the characteristics of the networks resources to be available to those flows, such as: the QoS, the maximum authorized bandwidth per session and its maximum duration. To enable these functionalities, extensions will be proposed to the "CIM User Model", more specifically to the `Privilege` and `PrevilegeManagement Service` classes.

Admission Control System

This system has as main responsibilities:

- Validate the authenticity and integrity of the ticket;
- Select the router interface;
- Verify the resources availability;
- Add and remove configuration parameters to the edge router;
- Registering of all the sessions with resources attributed and all the client requests denied.

To validate the authenticity and integrity of the tickets, this system will analyze the information given by the clients, e.g. the ticket. For that purpose, a security relationship with the Authorization system should be established. This relationship enables the validation of the Authentication_Data field which contains an authenticator of the information transported by the ticket [17]. The "CIM User Model" defines an authentication service named `Authentication Service`, able to satisfy the security requirements involved in this phase.

In what concerns to the interface selection and verification of resources availability, they are mutually dependent, because the selection of the interface to forward the session flows depends of the traffic destination and resources availability. The traffic's destination allows choosing the interface able to forward the flows, without verifying the availability of resources on those interfaces. This availability will be verified using information about QoS services implemented by ER interfaces, by the supported PHB and by the allocated and used bandwidth per PHB and per interface.

The "CIM Network Specification Model", allows the definition of QoS services associated to routers interfaces, through its class `QoSService` and it association with a set of conditioning factors. These factors define the treatment to be applied to the traffic at the routing equipment interfaces, with the help of mechanisms able to classify, measure, mark, queue, discard and scheduling the packets' forwarding. This way, it is possible to represent the QoS services and associated PHB, supported by a router. The `RoutingPolicy` class, defined in the "CIM Network Specification Model", allows to adequate the routing policies to the values of several attributes, in a consistent way, like the QoS services implemented by the routers interfaces, the supported PHB and the allocated and used bandwidth in each one of those PHB and interfaces. Nevertheless, it does not assure information about the number of traffic classes implemented by router interface and the allocated and used bandwidth, in each class and per interface. Works on Ponder policies specification language scope propose the implementation of a sub-model to extend CIM, named DiffServ metrics, created to store a set of statistics [18], which answer to the previous requirements.

When the resource reservation requests are authorized, configurations to adjust the routing tables and the traffic conditioning at the output interfaces are generated by the way of a set of instances of `ConditioningService`, defined in the "QoS Conditioning and Scheduling Services" of the CIM Network Specification Model.

To finalize, as the validity of the authorization is time limited, registers of all the sessions with authorized resources and associated duration should be kept. This information will allow to free resources when the validity is void and allow feed the Accounting System. They should integrate the following information: `SessionId`, `State`, `Cause`, `FlowId`, `Resources Specification` and `Validity`.

The CIM has no predefined class able to store this information, being necessary to propose extensions to it.

Accounting System

Firstly, the Accounting System will proceed with the selection of the accounting mechanism to be adopted for each resources reservation request demanded by the client. The selection will take in account the traffic source and destination, the type of service and the time period reservations occur. After that, will be necessary to store a set of information, composed by: user identification, adopted accounting type, class of service associated to the reservation in question, session reserved bandwidth, measured traffic and reservation start and end time.

The CIM model does not have any predefined class for this purpose being necessary to propose extensions.

3.4. Proposed Extensions to CIM

The actual specification of the CIM `Previlege` class, allows specify flow establishment authorization, by a Subject, through certain Targets. Although, it is not possible to specify the characteristics of the networks resources to be available to those flows, such as the QoS, the maximum authorized bandwidth for session and its maximum duration. In order to overcome these limitations, a set of extensions is proposed to enable the `Privilege ManagementService and Privilege` classes, with support to:

- QoS to be assigned to IP services;
- Maximum bandwidth authorized per session;
- Upper limit of authorization time;

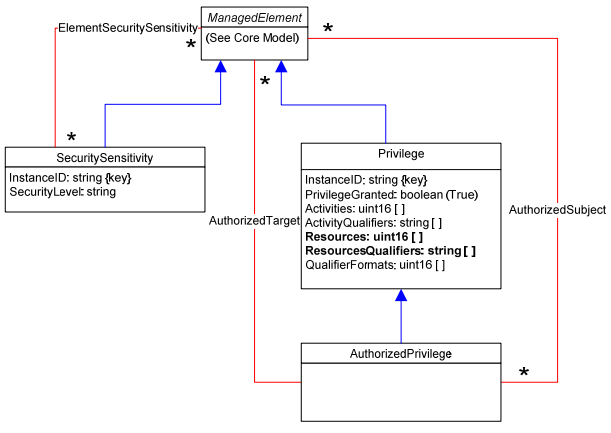

Figure 2. Privilege class and it relationships with proposed extensions.

- Limit number of Subject authorized sessions, per service group of IP services.

The proposed extension to the CIM `Previlege` class, presented on Figure 2, add to this class the following functionality: permit or deny communication of IP flows, per IP service; specify the networks resources to be available to authorized session. The first one, with the goal of specify which network resources are available to the authorized activities, such as the maximum bandwidth allowed per session (`SessionBw`), the associated QoS (`QoSService`), the maximum authorization time (`Duration`) and the maximum number of simultaneous sessions. The second one is to qualify and specify the authorized resources. Extensions are also proposed to the supervisor class `PrivilegeManagement Service`, because it is its responsibility to create, modify and eliminate the `AuthorizedPrivilege` instances obtained from the `Privilege` class.

In order to support registering of reservation resources requests and associated parameters at the Admission Control System, later used by the DiffServ metrics sub-model and by the Accounting System, a

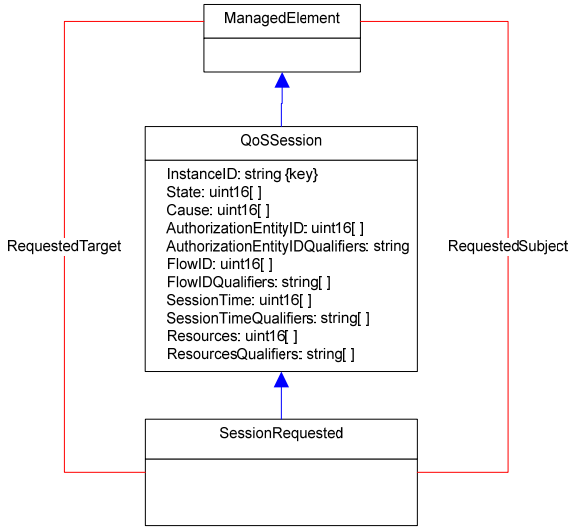

Figure 3. Proposed QoSSession class.

new class is proposed, which allow to define all the QoS sessions parameters, its state and the origin causes of denied sessions, named `QoSSession`. This class presented on Figure 3, defines a set of variables as follows:

- InstanceID, which identifies the instance and session;
- State, that presents the state of the session;
- Cause, which identifies the rejection cause of a

session;

- AuthorizationEntityID and AuthorizationEntityID Qualifier, identifying the authorization entity;
- FlowID and FlowClassifiers, that define the flow identification;
- SessionTime and
- SessionTime Qualifiers, which define the reservation time;
- Resources and ResourcesQualifier, which identify the specified resources.

Every time there is a session request it will be created an instance of the `SessionRequested`, associating resources requested by the client to the network equipments able to provide the resources. These instances are always created, whether the request is accepted or denied. To manage these instances, a new sub-class of `AuthorizationService`, is proposed, `QoSSessionManagementService`. It

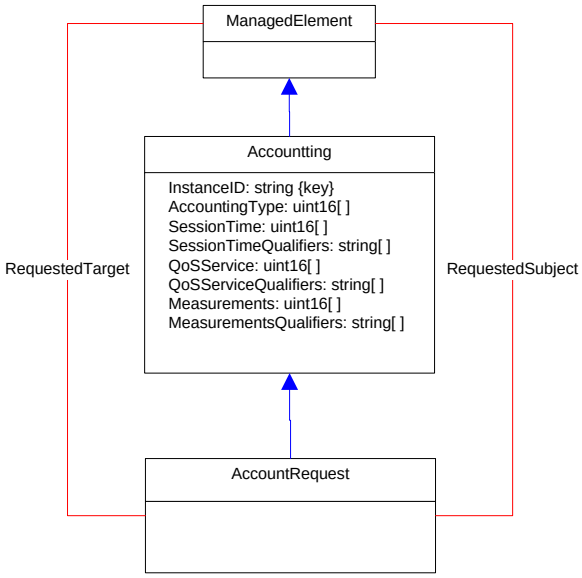

Figure 4. Proposed Accounting class.

will allow the register of new reservation requests and it modification every time the ticket is revalidated or the session is finalized.

The Accounting System must identify the resources used by each client, such as: client identification, type of accounting adopted, class of service, bandwidth allocated, measured traffic and reservation start and end time. For this purpose it's proposed a new class named `Accountting`, to collect accounting information of the requested resources, associated to the clients. This class presented on Figure 4, derivates from the `StatisticalData` class being composed by:

- AccounttingType, which identify the accounting methods to be adopted;
- SessionTime and SessionTimeQualifiers, which define the session time duration;
- QoSService and QoSServiceQualifiers, which define the type of service associated to a certain accounting rule and its parameters;
- Measurements and Measurements Qualifiers, which define the measured information type and its value.

At the end of the session, it will be registered the time when that has occurred and, if the accounting type used is Measured, it will be also registered the packets transmitted and drooped, measured by the edge router. To manage these instances, a new sub-class of `AuthorizationService` is proposed, named `AccounttingManagementService`. This class will allow registering new accounting process and to modify and finalize the existent ones.

4. Validation

This section discussed the prototype implementation and functional evaluation of the proposed architecture and CIM extensions.

4.1. Prototype Implementation

The main prototype implementation objective is the validation of the architecture and the proposed CIM extensions. Figure 5 shows a view of the prototype system developed to support the evaluation of the proposals. The QoS Client is implemented on a Kanotix Linux operating system, with support of Kerberos V5 (Authentication module). The Authorization system, Authentication system and Policy Repository are implemented in a single PC with Kanotix Linux and MySQL. The Admission Control and Accounting systems and Router PEP are implemented in a single PC equipped with Kanotix Linux with support of DiffServ and Linux Traffic Control package to allow the configuration of domain ingress routers. The Router is responsible for packet classification, flow analysis, and for packet marking (or remarking) and treatment in accordance with the QoS agreement established during the authorization of the session.

4.2. Functional Evaluation

This section addresses the validation of the implemented prototype. The behavior of the protocol is

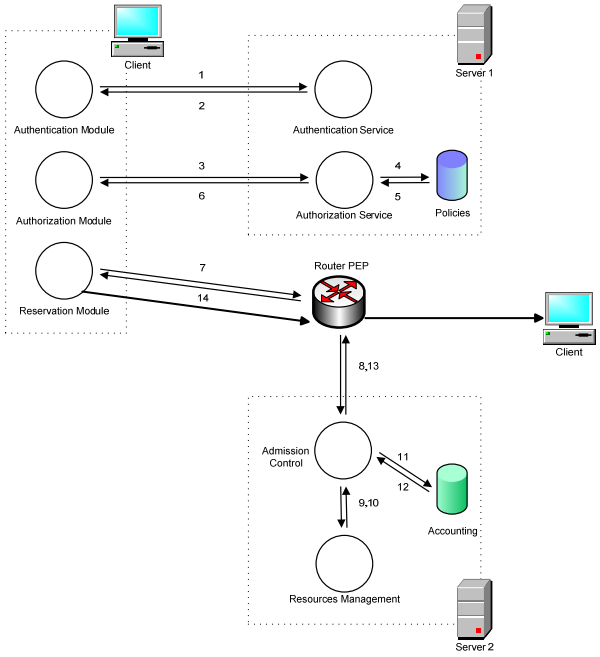

Figure 5. Prototype functional modules.

analyzed to access its functionality. The description of protocol messages and events is made with reference to Figure 5.

To evaluate the behavior of the prototype a set of policies has been created to support Authorization and Admission Control of QoS sessions. The description of these policies is kept in the MySQL database (DB). When the Client wants to establish a QoS session an authentication is needed in the Authentication Server to get the credentials (1) to communicate with the Authorization system (AS). After getting the credentials (2), the client will request authorization (3), specifying the QoS parameters for the session or leaving the decision to the Authorization system. The AS will identify the client who originated the request and request the policies applied to it (4) (5). Afterwards, it verifies if the client has authorization to establish the session. If the answer is positive, the AS issues a ticket to the client (6). With this ticket the Client, through its reservation module, will issue a request for resources reservation, to the Admission Control system (ACS). This request is intercepted by the Router PEP (7) that follows it to the ACS (8). This system will analyze the availability of network resources to the new session (9, 10). If resources are available, the ACS selects the accounting method to be used to this session, analyzing a set of accounting policies (11, 12). After that, configurations parameters are sent to the router PEP to reconfigure the filtering rules of the ingress router (13). The router receives these configurations and informs the QoS Client that it request was accepted (14).

```
QoSAuthd       : Waiting for new request

QoSAuthd       : QoSReq <- lct-sec02
Authorization  : clientValidationOK
Authorization  : requestAnalyses
(newSession to 208.38.61.228:80:6)
Authorization  : policyRequest -> PS
Authorization  : policyReply <- PS
Authorization  : policiesVerified
(sessionRefused-maximum_sessions)
Authorization  : qosReply -> lct-sec02

QoSAuthd       : Waiting for new request

QoSAuthd       : QoSReq <- lct-sec02
Authorization  : clientValidationOK
Authorization  : requestAnalyses
(newSession to 10.0.0.1:5060:17)
Authorization  : policyRequest -> PS
Authorization  : policyReply <- PS
Authorization  : policiesVerified
(noPolicies-service_denied)
Authorization  : qosReply -> lct-sec02

QoSAuthd       : Waiting for new request
```

Figure 6. Authorization Server log file extract.

Figure 6 shows an extract of the Authorization Server log file. This figure results from two answers to client requests, whose authorization was denied. The first one is because the maximum number of sessions specified for HTTP, to this client in simultaneous, was reached. The second one is because the authorization policies did not authorize the SIP service to this client.

5. Conclusions and Future Work

In this paper we proposed an information model to support the AAAA architecture proposed in our previous work [8]. A set of extensions was proposed to the CIM in order to support QoS negotiation with Authentication, Authorization, Admission control and Accounting (AAAA), based on Policies, for client application session with QoS needs.

The proposal seeks to overcome the security limitations of current DiffServ model and to manage the admission control of sessions in a dynamic way, according to the defined Authorization Policies and to the Admission Control Policies. The proposal addresses the issues of secure negotiation of QoS in an intra-domain scope, the issues related to admission control in the edge devices of DiffServ domains and the procedures of authentication of the clients and authorization of resources reservation for sessions establishment.

With the proposed architecture, a dynamic and safer admission control to network resources with QoS is

obtained, attributing resources only and exclusively to previously authenticated and authorized customers. These resources are attributed at the establishment of a new session and will be removed as soon as the session ends. A more accurate and secure management of networks resources is achieved comparatively to the conventional procedure used at the DiffServ edge routers admission control.

Future work (already ongoing) will address experimentation and evaluation of scalability and performance behavior of the proposed architecture.

Acknowledgment

This work was partially financed by the PRODEP program supported by the Portuguese Government and the European Union FSE Programme.

References

[1] S. Blake et al, "An Architecture for Differentiated Services", RFC 2475, IETF, Dec. 1998.

[2] K. Nichols et al, "Definition of the Differentiated Services Fields (DS Fields) in the IPv4 and IPv6 Headers", RFC 2474, IETF, Dec. 1998.

[3] C. Rabadão, E. Monteiro, "Segurança e QoS no Modelo DiffServ (Security and QoS in the DiffServ Model)", 5th Conference on Computer Networks (CRC2002), Faro, Portugal, University of Algarve, 26-27 Sep. 2002.

[4] Zhi Fu et al, "Security Issues for Differentiated Service Framework", Internet Draft (expired), Oct. 1999.

[5] S. Kent, R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 2406, Nov. 1998.

[6] R. Atkinson, "IP Authentication Header", RFC 1826, IETF, Aug. 1995.

[7] A. Striegel, "Security Issues in a Differentiated Services Internet", Proc. of Trusted Internet Workshop - HiPC, Bangalore, India, Dec. 2002.

[8] C. Rabadão, E. Monteiro, "Authentication, Authorization, Admission and Accounting for QoS applications", Proc. of 3rd European Conference on Universal Multiservice Networks (ECUMN), Porto, 25-27 Oct. 2004.

[9] "DMTF Common Information Model (CIM) Specification, Version 2.9", Distributed Management Task Force, January 2005.

[10] V. Sander et al, "End-to-End Provision of Policy Information for Networks QoS", Proc. of 10th IEEE International Symposium of High Performance Distributed Computing, San Francisco-California, 07-09 Aug. 2001.

[11] W. Bumpus, J. W. Sweitzer, P. Thompson, A. R. Westerinen, and R. C. Williams, "Common Information Model: Implementing the Object Model for Enterprise Management", Wiley, 2000.

[12] "Information Technology – Open Systems Interconnection – Systems Management – Management Functions," IS 10164-x, ISO/IEC, 1991-94.B.

[13] "B. Moore, E. Ellesson, J. Strassner, A. Westerinen, "Policy Core Information Model -- Version 1 Specification", RFC3060, IETF, Feb. 2001.

[14] Moore, Editor, "Policy Core Information Model (PCIM) Extensions", RFC3460, IETF, Jan. 2003.

[15] "CIM Policy Model, Version 2.9", Management Task Force, Dec 2004

[16] Y. Snir, Y. Ramberg, J. Strassner, R. Cohen, B. Moore, "Policy Quality of Service (QoS) Information Model", RFC3644, IETF, Nov. 2003.

[17] L-N. Hamer, B. Gage and H. Shieh, "Session Authorization Policy Element", RFC3521, IETF, Apr. 2003.

[18] L. Lymberopoulos, E. Lupu, M. Sloman, "Ponder Policy Implementation and Validation in a CIM and Differentiated Services Framework", Proc. of the 9th IEEE/IFIP Network Operations and Management Symposium (NOMS 2004), Seoul, Korea, 19-23 Apr. 2004.