Verification of Admission Control implementation in federated Testbedding Environment

DAMIAN DUDA, PIOTR PYDA, ANDRZEJ STAŃCZAK, MAREK AMANOWICZ

Military Communication Institute,
ul. Warszawska 22A, 05-130 Zegrze Południowe

Abstract: This paper is focused on the development issues and the presentation of testing results of Admission Control implementation in the Network Centric Operations Consortium laboratory at the Military Communication Institute. It is the continuation of work concerning the concept of Admission Control for federated systems presented during the previous MCC conference. The proposed Admission Control implementation constitutes one of the key modules of the QoS IP network. The module allows to evaluate the user’s application demand for specific QoS parameters. The Admission Control module is responsible for checking the availability of network resources necessary to handle a specific request. The main assumption is that the request, its verification and network resources reservation are processed on-line. Examples of results of the AC module verification are presented for selected end-user services. The results of testing prove the concept of Admission Control for the VoIP and VoD services in federated systems.

Keywords: Admission Control, Federation of Systems, Quality of Service, VoIP, VoD

Introduction

The communications and information infrastructure used to support the coalition operations is multi-domain by nature. The domains – military, government, public and civilian systems – are administratively independent and heterogeneous in respect of technology used. However the systems cooperate creating the logical structure to achieve common objectives. Such structures are considered by NATO NC3A [1] as the Federation of Systems (FoS). They propose an overarching architecture composed of information assurance and integration services, communications services, system management and control. The communications component encompasses the communications and information infrastructure with a guaranteed Quality of Service (QoS). There are QoS mechanisms proposed to guarantee required end-to-end and network performance parameters.

1 The work is supported by Polish Ministry of Science and Higher Education funds for project PBZ-MiNISW-DBO-02/I/2007
The necessity of using QoS mechanisms comes from the hallmark of military networks such as:

- limited network resources such as capacity of the main communications links,
- damages, malfunctions and limited availability of communications infrastructure,
- limited applicability of network over-provisioning due to a common use of wireless links.

The propositions in [1] follow the architecture of IP QoS network with heavy influence of current research and standardization works for military as well as for commercial networks. However, it should be noted that commercial works in the area of IP QoS do not take into account the detailed security requirements. For example, providing the QoS in single domain and enterprise class networks was the subject of AQUILA project [3]. The multi-domain and security issues were not considered. The representative research projects in the area of commercial multi-domain QoS IP networks are QBone and EuQoS. The first project conducted research on the QoS provision in Internet [4]. The latter focused on multi-domain systems, built from access networks connected to a common over-provisioned core network [5]. Both projects did not consider the mechanisms for securing user and system information as well as the impact of a limited capacity of communications and information infrastructure. The generic architecture of the QoS packet network was proposed by ITU-T in the recommendation Y.1291, however there are no detailed guidelines regarding the development of or cooperation with security mechanisms [6].

In the area of military networks the TACOMS project conducted research and standardization works on the interoperability between heterogeneous systems in coalition structures [7]. TICNET project was focused on the QoS provisioning in IPv6 tactical networks [8] and INSC – on the technical architecture of IPv6 interoperable and secure networks [9].

The documents mentioned above do not define the detailed architecture for a federated IP QoS network with a limited capacity. It was the main reason of conducting our research work. The results presented in this paper are focused on Admission Control (AC), which is one of the most important QoS mechanisms in the process of providing QoS for the users and preventing network overloads.

This paper presents the functional description of the QoS networking infrastructure and its implementation in the laboratory environment of the Military Communication Institute (MCI). The structure of the paper is as follows: following a brief introduction, the authors propose the architecture of the QoS IP networking infrastructure for the federation of systems. Next they describe the Admission Control module implementation. The consecutive chapter presents the verification results of AC implementation and the last one recapitulates the paper and proposes directions for future work.
QoS Network Architecture

According to ITU-T the QoS IP network architecture should contain a complete set of mechanisms in the management, control and data plane [6]. However, the independence of systems inside the federation forces requires some modifications to the architecture because of:

- decentralized management due to administrative constraints,
- dynamic changes of the federation's structure, which involves network topology changes,
- limited accessibility of information about the structure and state of the systems due to security and administrative constraints.

The proposed architecture of the QoS IP infrastructure takes the above into account and enhances the set of traffic control mechanisms proposed in the previous author's MCC papers [2], [10]. In particular the architecture consists of:

- Management plane elements:
  - network manager entity – part of Management&Signalling (M&S) module,
  - SLA management entity,
  - network devices configuration module – Remote Procedure Call-based configuration Server (RPCS),
  - external security server (AAA).

- Control plane elements:
  - Admission Control (AC) module,
  - signalling node – part of M&S module,
  - Network State Database (NSDB),
  - Established Connection Database (ECDB).

- Data plane elements:
  - routers with traffic control mechanisms at the IP packet level,
  - manageable transmission devices.

- Protocols between functional elements of architecture:
  - user access protocol – enhanced Session Initiation Protocol [10],
  - inter-nodal protocol – between autonomous systems, enhanced SIP,
  - end-to-end signalling protocol – standard SIP,
  - internal control and management protocols – XML-RPC for reconfiguration and dedicated protocol between AC and NM modules.

The architecture was enhanced with a signalling layer distributed among autonomous systems. The layer is used for creating relations between management and control entities of cooperating systems inside the FoS.

The user request is carried by an enhanced Session Initiation Protocol [10]. The enhancements allow to explicitly pass the traffic descriptor and required QoS parameters to the control plane mechanisms. The first enhancement was the definition of a new SIP method called "ORDER". The message with the "ORDER"
method is passed by the SIG-Proxy directly to the Network Manager. The second enhancement allows to encode the traffic descriptor and QoS parameters in the Session Description Protocol (SDP) fields. The detailed contents of an enhanced SIP signalling message for a sample VoIP session is presented below:

```
ORDER sip:1280@192.1.2.92 SIP/2.0
Via: SIP/2.0/UDP 192.1.1.93;branch=z9hG4bK57D486F7
CSeq: 4407 ORDER
To: <sip:1280@sig2.las2.wil.pl>;tag=8450b0a2-d200-1910-8f43-002308b77b15
Content-Type: application/sdp
From: "Damian Duda na Ubuntu" <sip:1170@sig1.las1.wil.pl>;tag=2E29C601
Call-ID: 965699608@192.1.1.93
Subject: sip:1170@sig1.las1.wil.pl
Route: <sip:192.1.1.2;ftag=2E29C601;lr=on>, <sip:192.1.2.2;ftag=2E29C601;lr=on>
Content-Length: 275
User-Agent: kphone/4.2
Contact: "Damian Duda na Ubuntu" <sip:1170@192.1.1.93;transport=udp>

v=0
o=username 0 0 IN IP4 192.1.1.93
s=The NEC Kphone
c=IN IP4 192.1.1.93
t=0 0
m=audio 50280 RTP/AVP 8 97 3 0
c=IN IP4 192.1.1.93
b=AS:100000
q=telephony resv
m=audio 5064 RTP/AVP 8
c=IN IP4 192.1.2.92
b=AS:100000
q=telephony resv
```

Definitions of particular fields are available in [10] and [17].

An application that demands specific QoS parameters for its traffic must first request a connection, which involves informing the network management entity about the characteristics of user traffic and required QoS parameters. The AC module checks whether the network has enough resources available to accept the connection, and then either accepts or rejects the connection request. The AC module introduced in [2] consists of a decision algorithm and a security component. The task of the decision algorithm is to check available bandwidth and report the decision to the management entity. The verification of the request's conformance with security policy is the role of a security component. The AC module implementation is distributed among independent domains. However, security components are
intended to be close to the source of requests in order to filter out the requests from unauthenticated and unauthorized users. Security component is not an autonomous mechanism but cooperates with a security server (AAA). The AAA server is a functional module implemented by another team from the Military Communication Institute. It will be included in the testbed in near future.

**Admission Control implementation**

Figure 1 shows a functional diagram of the experimental system. In our implementation the applications send requests to M&S module which forwards them to AC module. AC module receives the application request from M&S and – after processing it – returns an answer to M&S.

![Functional diagram of QoS IP architecture for FoS](image)

Figure 1. The functional diagram of QoS IP architecture for FoS.

Abbreviations: SLA – Service Level Agreement, NSDB – Network Status Database, ECDB – Established Connections Database, AAA – Authentication, Authorization and Accounting. 

ER – Edge Router of autonomous system [2].

The AC module works as a server in a typical client-server mode. Initially it waits for the requests from the Network Manager. On request from NM the AC starts processing the call according to the following sequence:
1. Analyses the request from NM.
2. Connects to Network Status Data Base, NSDB for:
   a. checking the correctness of request,
   b. checking available capacity on the end-to-end inter-domain path.
3. Runs the decision algorithm based on:
   a. parameters in the request,
   b. answer from NSBD.
5. Depending on decision the following actions are undertaken:
   – Decision 0: AC sends the answer to NM about request rejection,
   – Decision 1: AC sends the answer to NM about request acceptance and updates NSBD records with the number of active connections and a network state.
6. AC returns to idle state, waiting for a new request from NM.

The AC module requires that a user application sends the signalling message which includes the Class of Service (CoS) and peak bit rate requested.

The decision algorithm of AC supports providing the exact guarantees for streaming as well as elastic applications. A new request is accepted when the following formula is fulfilled:

\[ C_{req} \leq C_{path} - C_{reserv} \]

where:
- \( C_{req} \) – capacity requested for a new connection,
- \( C_{path} \) – capacity in the selected path for a given CoS,
- \( C_{reserv} \) – capacity reserved for current connections.

To implement the AC module a number of tools were used: a free, open-source Integrated Development Environment Netbeans in version 6.7.1 (Build 200907230101), Java 1.6.0_16; Java HotSpot(TM) Client VM 14.2-b01, database jdbc:derby and driver org.apache.derby.jdbc.ClientDriver. Because of limitations of testbed environment the tasks of AC do not include the communication with security mechanisms.

**Experimental results**

The experimental testbed was developed in the Military Communication Institute's laboratory in co-operation with the Military University of Technology, Warsaw. The laboratory setup consists of the following main functional modules:

- Control and management software modules
  - Network Manager – Win32 application developed in Borland C++ environment, with embedded communication clients to AC and RPCS servers [12],
- Admission Control module – JAVA application server [14],
- Signalling Proxy – based on SIP Express Router [15] [13],
- RPCS server – Python-based XML-RPC server [13].

- Communications and information infrastructure
  - Software routers based on GNU/Linux 2.6.21 kernel (Slackware Linux 12.0),
  - Traffic generators and analyzers – Spirent Test Center SPT-2000,
  - DNS, DHCP and routing software from Slackware distribution.

- User applications with QoS signalling
  - Modified soft-phone – Kphone adapted by MUT with additional software modules [11] [17],
  - VoD – Video LAN Client (VLC) with external SIP signalling module (Python language used) [16].

Scenario #1: QoS support for Video on Demand service

The goal of this test is to show the impact of background traffic on the available bit rate for video streaming. The testbed setup is presented in Figure 2. The VoD source is located in LAS2 and accessible at address rtsp://vod.las2.wil.pl. The client accesses the video content through the limited bandwidth link. The reservation request is sent from an external call generator. The rest of setup is as follows:

- VoD source
  - Video LAN Server (VLS),
  - VBR mpeg1 stream from file:///~wil.mpg,
  - Variable bit rate from 0.9 to 1.4 Mbps (MPEG-1 stream),
  - time of streaming 100 sec.

- Background traffic:
  - Interwatch IW95000 hardware generator and analyzer,
  - constant bit rate traffic, UDP at 1.8 Mbps, PDU=1484 bytes.

The results of test are presented in Figure 3. It shows the impact of background traffic – without appropriate QoS mechanisms the video stream competes for link sharing with background traffic and its bit rate drops to about 0.6 Mbps. The QoS request was generated at about 55 second of the test. After the acceptance of a request and when appropriate reservation mechanisms were run, the video bit rate returned to its nominal value. It was at the cost of heavy traffic loss of background traffic.
Figure 2. The testbed setup for testing the Video on Demand service [own source]

Figure 3. The output router’s bit rate of video stream [own source]

Scenario #2: QoS support for VoIP service

In this scenario we examine the capability of guaranteeing QoS parameters like IP packet losses, delay and bit rate for the VoIP test traffic. Two cases were considered and tested: 1) without QoS mechanisms, i.e. FIFO scheduling, no classifiers, shap-
ers and policers were used and 2) with packet level QoS mechanisms configured in Linux routers. The tests present the impact of the background traffic on the QoS metrics for the VoIP service. The setup of test is presented in Figure 4.

The VoIP traffic and background traffic sources were provided by two Spirent Test Center SPT-2000 network analysers. Other parameters were set as follows:

- **Test traffic**
  - UDP/IP traffic, 200B fixed size payload;
  - Constant generator rate at 100 kbps;
  - Time of VoIP streaming 300 sec.

- **Background traffic**
  - UDP/IP traffic, 1500B fixed size payload;
  - Constant generator rate at 2000 kbps.

- **Traffic control mechanism**
  - Hierarchical Token Bucket (HTB) with three classes:
    - Class 1: Telephony, 1 Mbps reserved,
    - Class 2: Low Latency Data, 1 Mbps reserved,
    - Class 3: Standard, 1 kbps reserved with 1 Mbps ceil rate.
  - Admission Control configuration
    - Reserves 100 kbps per single VoIP connection;
    - Maximum 10 connections were allowed.

The measurement equipment allowed to gather the following IP level metrics:

- End-to-end one-way IP packet level average, minimum and maximum delay,
- End-to-end one-way IP packet level loss ratio.

---

Figure 4. The testbed setup for testing the VoIP service [own source]
For the verification of the AC module the following test was performed. The AC module was configured to accept connections not exceeding the total rate of 1 Mbps for the Telephony class of service. The reserved bit rate per connection equals 100 kbps which roughly corresponds to the bit rate of a single G.711 stream. Thus ten simultaneous connections were allowed. The AC module worked as expected and rejected all excessive requests.

The results of the test are presented in Table 1. Without the configured QoS mechanisms the impact of background on the VoIP traffic is significant – delay is longer than 600 ms and losses are greater than 5%. The value of delay comes from the configured length of the router’s buffers and the transmission delay of a single packet at line speed. The 100 packet length buffer is shared with background traffic so the expected delay is mainly enforced by the delay of large background packets, which is about 100 times 6 ms. With traffic control mechanisms turned on and configured for the VoIP traffic the resulting values of metrics are acceptable. The average latency is 4 ms with a maximum value about 13 ms. The packet level losses were not observed during the test.

The background traffic suffered large delay and losses. The losses near the 50% come mainly from the configured ceil rate for Standard traffic at 1 Mbps.

<table>
<thead>
<tr>
<th>Case</th>
<th>Parameter</th>
<th>VoIP traffic</th>
<th>Background traffic</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1 no background</td>
<td>Average latency [ms]</td>
<td>1</td>
<td>N.A.</td>
</tr>
<tr>
<td></td>
<td>Minimum latency [ms]</td>
<td>1</td>
<td>N.A.</td>
</tr>
<tr>
<td></td>
<td>Maximum latency [ms]</td>
<td>1</td>
<td>N.A.</td>
</tr>
<tr>
<td></td>
<td>Dropped frames [%]</td>
<td>None observed</td>
<td>N.A.</td>
</tr>
<tr>
<td>#2 With background</td>
<td>Average latency [ms]</td>
<td>614</td>
<td>617</td>
</tr>
<tr>
<td>No QoS mechanisms</td>
<td>Minimum latency [ms]</td>
<td>600</td>
<td>605</td>
</tr>
<tr>
<td>(FIFO scheduling)</td>
<td>Maximum latency [ms]</td>
<td>632</td>
<td>633</td>
</tr>
<tr>
<td></td>
<td>Dropped frames [%]</td>
<td>5.1</td>
<td>4.5</td>
</tr>
<tr>
<td>#1 With background</td>
<td>Average latency [ms]</td>
<td>4</td>
<td>1188</td>
</tr>
<tr>
<td>and with QoS</td>
<td>Minimum latency [ms]</td>
<td>1</td>
<td>1167</td>
</tr>
<tr>
<td>mechanisms (HTB</td>
<td>Maximum latency [ms]</td>
<td>13</td>
<td>1192</td>
</tr>
<tr>
<td>scheduling with three classes)</td>
<td>Dropped frames [%]</td>
<td>None observed</td>
<td>48.6</td>
</tr>
</tbody>
</table>

**Table 1. Results of VoIP service testing**

**Conclusions**

This paper presents a functional description of the QoS network and its implementation in the laboratory environment of the Military Communication Institute. The network contains the minimum set of QoS mechanisms. However, the set allows for on-line requesting and reservation of appropriate network resources for end-user applications. The paper described the details
of the AC module implementation and shows the sample laboratory results of the VoIP and the VoD services' tests.

The results gathered for defined test scenarios proved the concept of the Admission Control for federated systems presented in [2]. As some works are conducted in the area of co-operation with security mechanisms for authentication and authorization of the QoS requests, we may soon expect the first results.

The future work includes: implementing modules to communicate with security mechanisms, the QoS-based routing, handling of IPv6, as well as the implementation and testing of a logical interface for webservices-based applications.

REFERENCES