NSIS Working Group
Internet Draft
Document: draft-ietf-nsis-req-00.txt
Expires: August 2002

M. Brunner (Editor)
NEC

February 2002

Requirements for QoS Signaling Protocols
<draft-ietf-nsis-req-00.txt>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/ietf/1id-abstracts.txt
The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html.

Abstract

This draft defines requirements for signaling QoS across different network environments. To achieve wide applicability of the requirements, the starting point is a diverse set of scenarios/use cases concerning various types of networks and application interactions. We also provide an outline structure for the problem, including QoS related terminology. Taken with the scenarios, this allows us to focus more precisely on which parts of the overall QoS problem needs to be solved. We present the assumptions and the aspects not considered within scope before listing the requirements grouped according to areas such as architecture and design goals, signaling flows, layering, performance, flexibility, security, and mobility.

1 Introduction

This draft defines requirements for signaling QoS across different network environments. It does not list any problems of existing QoS signaling protocols such as RSVP.

In order to derive requirements for QoS signaling it is necessary to first have a clear idea of the scope within which they are applicable. After defining terminology in Section 2, we therefore start in Section 3 with a set of QoS signaling scenarios. These scenarios derive from a variety of backgrounds, and help obtain a clearer picture of what is in or out of scope of the NSIS work. They illustrate the problem of QoS signaling from various perspectives (end-system, access network, core network) and for various areas (fixed line, mobile, wireless environments). As the NSIS work becomes more clearly defined, scenarios will be added or dropped, or defined in more detail.

Based on these scenarios, we are able to define the QoS signaling problem on a more abstract level. In Section 4, we thus present a simple conceptual model of the QoS signaling problem, describe the entities involved in QoS signaling, and typical signaling paths. Additionally we list our assumptions and exclusions.

The model of Section 4 allows deriving requirements from the scenarios presented in Section 2 in a coherent and consistent manner. Requirements are grouped according to areas such as Architecture and design goals, Signaling Flows, Layering, Performance, Flexibility, Security and Mobility.

QoS is a pretty large field with a lot of interaction with other protocols, mechanisms, applications etc. In the following, some thoughts from an end-system point of view and from an network point of view.

End-system perspective: In future mobile terminals, the support of adaptive applications is more and more important. Adaptively can be seen as an important technique to react to QoS violations that may occur frequently, e.g., in wireless environments due to changed

environmental and network conditions. This may result in degraded end-to-end performance. It is then up to adaptive applications to react to the new resource availability. Therefore, it is essential to define interoperability between media-, mobility- and QoS management. While most likely mobile terminals cannot assume, that explicit QoS reservation schemes are available, some access networks nevertheless may offer such capabilities. Applications subscribed to an end-system QoS management system should be supported with a dedicated QoS API to set-up, control and adapt media sessions.

Network perspective: QoS enabled IP networks are expected to handle two different kinds of QoS granularities: per-flow QoS and per-trunk/per-class QoS. Per-flow QoS might be needed in access networks and may there be subject of QoS signaling. However, in the core network only per-trunk or per-class QoS can be considered for scalability reasons. Therefore there might be different requirements on QoS signaling applying to different parts of the network. In the access network QoS signaling is an interaction between end systems and access routers or access network QoS managers (in the following we call them QoS initiator and QoS controller). In the core network QoS signaling refers to trunks or classes of traffic between core and edge systems or between peering core systems. Please note that this does not exclude the transport of per-flow signaling through core networks.

It is clear from these descriptions that the subject of QoS is uniquely complex and any investigation could potentially have a very broad scope - so broad that it is a challenge to focus work on an area which could lead to a concrete and useful result. This is our motivation for considering a set of use cases which map out the domain of application that we want to address; these use cases are given in section 3. It is also the motivation for defining a problem structure, which allows us to state the boundaries of what types of functionality to consider, and to list background assumptions. This structure is given in section 4. The requirements themselves follow in section 5. There are several areas of the requirements related to networking aspects which are incomplete, for example, interaction with host and site multi-homing, use of anycast services, and so on. These issues should be considered in any future requirement analysis work.

2 Terminology

Aggregate: a group of flows, usually with similar QoS requirements, which can be treated together as a whole with a single overall QoS requirement for signaling and provisioning. Aggregates and flows can be further aggregated together.

[QoS] Domain: a collection of networks under the same administrative

control and grouped together for administrative purposes.

Egress point: the router via which a path exits a domain/subdomain.

End Host: the end system or host, for whose flows QoS is being requested and provisioned.

End-to-End QoS: the QoS delivered by the network between two communicating end hosts. End-to-end QoS co-ordinates and enforces predefined traffic management policies across multiple network entities and administrative domains.

Edge-to-edge QoS: QoS within an administrative domain that connects to other networks rather than hosts or end systems.

Flow: a traffic stream (sequence of IP packets between two end systems) for which a specific level of QoS is to be provided. The flow can be unicast (uni- or bi-directional) or multicast.

Flow Administration: represents the policy associated with how flows should be treated in the network, for example whether and how the flows should be aggregated. It may consist of both user and local network management information.

Higher Layers: the higher layer (transport protocol and application) functions that request QoS from the network layer. The request might be a trigger generated within the end system, or the trigger might be provided by some entity within the network (e.g. application proxy or policy server).

Indication: feedback from QoS provisioning to indicate the current QoS being provided to a flow or aggregate, and whether any violations have been detected by the QoS technology being used within the local domain/subdomain.

Ingress point: the router via which a path enters a domain/subdomain.

Mapping: the act of transforming parameters from QSCs to values that are meaningful to the actual QoS technology in use in the domain/subdomain.

Path: the route across the networks taken by a flow or aggregate, i.e. which domains/subdomains it passes through and the egress/ingress points for each.

Path segment: the segment of a path within a single domain/subdomain.

QoS Administration Function: a generic term for all functions associated with admission control, policy control, traffic engineering etc.

QoS Control Information: the information the governs the QoS

treatment to be applied to a flow or aggregate, including the QSC, flow administration, and any associated security or accounting information.

QoS Controller: this is responsible for interpreting the signaling carrying the user QoS parameters, optionally inserting/modifying the parameters according to local network QoS management policy, and invoking local QoS provisioning mechanisms.

QoS Initiator: this is responsible for generating the QSCs for traffic flow(s) based on user or application requirements and signaling them to the network as well as invoking local QoS provisioning mechanisms. This can be located in the end system, but may reside elsewhere in network.

QoS Provisioning: the act of actually allocating resources to a flow or aggregate of flows, may include mechanisms such as LSP initiation for MPLS, packet scheduler configuration within a router, and so on. The mechanisms depend on the overall QoS technology being used within the [sub]domain.

QoS Service Classes (QSC): specify the QoS requirements of a traffic flow or aggregate. Can be further sub-divided into user specific and network related parameters

QoS Signaling: a way to communicate QSCs and QoS management information between hosts, end systems and network devices etc. May include request and response messages to facilitate negotiation/re-negotiation, asynchronous feedback messages (not delivered upon request) to inform End Hosts, QoS initiators and QoS controllers about current QoS levels, and QoS querying facilities.

[QoS] Subdomain: a network within an administrative domain using a uniform technology/QoS provisioning function to provision resources.

QoS Technology: a generic term for a set of protocols, standards and mechanisms that can be used within a QoS domain/subdomain to manage the QoS provided to flows or aggregates that traverse the domain. Examples might include MPLS, DiffServ, and so on. A QoS technology is associated with certain QoS provisioning techniques.

QoS Violation: occurs when the QoS applied to a flow or aggregate does not meet the requested and negotiated QoS agreed for it.

Resource: something of value in a network infrastructure to which rules or policy criteria are first applied before access is granted. Examples of resources include the buffers in a router and bandwidth on an interface.

Resource Allocation: part of a resource that has been dedicated for the use of a particular traffic type for a period of time through the application of policies

3 Scenarios/Use cases

In the following we describe scenarios, which are important to cover, and which allow us to discuss various requirements. Some

regard this as use cases to be covered defining the use of a QoS signaling protocol.

3.1 Scenario 1: Terminal Mobility

The scenario we are looking at is the case where a mobile terminal (MT) changes from one access point to another access point. The access points are located in separate QoS domains. We assume Mobile IP to handle mobility on the network layer in this scenario and consider the various extensions (i.e., IETF proposals) to Mobile IP, in order to provide 'fast handover' for roaming Mobile Terminals. The goal to be achieved lies in providing, keeping, and adapting the requested QoS for the ongoing IP sessions in case of handover. Furthermore, the negotiation of QoS parameters with the new domain via the old connection might be needed, in order to support the different 'fast handover' proposals within the IETF.

The entities involved in this scenario include a mobile terminal, access points, an access network manager, communication partners of the MT (the other end(s) of the communication association).
From a technical point of view, terminal mobility means changing the access point of a mobile terminal (MT). However, technologies might change in various directions (access technology, QoS technology, administrative domain). If the access points are within one specific QoS technology (independent of access technology) we call this intra-QoS technology handoff. In the case of an inter-QoS technology handoff, one changes from e.g. a DiffServ to an IntServ domain, however still using the same access technology. Finally, if the access points are using different access technologies we call it inter-technology hand-off.

The following issues are of special importance in this scenario:

1) Handoff decision

- The QoS management requests handoff. The QoS management can decide to change the access point, since the traffic conditions of the new access point are better supporting the QoS requirements. The metric may be different (optimized towards a single or a group/class of users). Note that the MT or the network (see below) might trigger the handoff.

- The mobility management forces handoff. This can have several reasons. The operator optimizes his network, admission is no longer granted (e.g. emptied prepaid condition). Or another example is when the MT is reaching the focus of another base station. However, this might be detected via measurements of QoS on the physical layer and is therefore out of scope of QoS signaling in IP. Note again that the MT or the network (see below) might trigger the handoff.

- This scenario shows that local decisions might not be enough. The rest of the path to the other end of the communication needs to be considered as well. Hand-off decisions in a QoS domain, does not only depend on the local resource availability, e.g., the wireless

part, but involves the rest of the path as well. Additionally, decomposition of an end-to-end reservation might be needed, in order to change only parts of it.

2) Trigger sources

- Mobile terminal: If the end-system QoS management identifies another (better-suited) access point, it will request the handoff from the terminal itself. This will be especially likely in the case that two different provider networks are involved. Another important example is when the current access point bearer disappears (e.g. removing the Ethernet cable). In this case, the QoS initiator is basically located on the mobile terminal.

- Network (access network manager): Sometimes, the handoff trigger will be issued from the network management to optimize the overall load situation. Most likely this will result in changing the base-station of a single providers network. Most likely the QoS initiator is located on a system within the network.

3) Integration with other protocols

- Interworking with other protocol must be considered in one or the other form. E.g., it might be worth combining QoS signaling between different QoS domains with mobility signaling at hand-over.

3.2 Scenario 2: Cellular Networks

In this scenario, the user is using the packet service of a 3rd generation cellular system, e.g. UMTS. The region between the End Host and the edge node connecting the cellular network to another QoS domain (e.g. the GGSN in UMTS or the PDSN in 3GPP2) is considered to be a single QoS domain [4][5].

The issues in such an environment regarding QoS include:

1) Cellular systems provide their own QoS technology with specialized parameters to co-ordinate the QoS provided by both the radio access and wired access network. For example, in a UMTS network, one aspect of GPRS is that it can be considered as a QoS technology; provisioning of QoS within GPRS is described mainly in terms of calling UMTS bearer classes. This QoS technology needs to be invoked with suitable parameters when a request for QoS is triggered by higher layers, and this therefore involves mapping the requested IP QoS onto these UMTS bearer classes. This request for resources might be triggered by IP signaling messages that pass across the cellular system, and possibly other QoS domains, to negotiate for network resources. Typically, cellular system specific messages invoke the underlying cellular system QoS technology in parallel with the IP QoS negotiation, to allocate the resources

within the cellular system.

2) The placement of QoS initiators and QoS controllers (terminology in the framework given here). The QoS initiator could be located at

the End Host (triggered by applications), the GGSN/PDSN, or at a node not directly on the data path, such as a bandwidth broker. In the second case, the GGSN/PDSN could either be acting as a proxy on behalf of an End Host with little capabilities, and/or managing aggregate resources within its QoS domain (the UMTS core network). The IP signaling messages are interpreted by the QoS controllers, which may be located at the GGSN/PDSN, and in any QoS sub-domains within the cellular system.

3) Initiation of IP-level QoS negotiation. IP-level QoS re-negotiation may be initiated by either the End Host, or by the network, based on current network loads, which might change depending on the location of the end host.

4) The networks are designed and mainly used for speech communication (at least so far).

Note that in comparison to the former scenario, the emphasis is much less on the mobility aspects, because mobility is mainly handled on the lower layer.

3.3 Scenario 3: Session Mobility

In this scenario, a session is moved from one end-system to another. Ongoing sessions are kept and QoS parameters need to be adapted, since it is very likely that the new device provides different capabilities. Note that it is open which entity initiates the move, which implies that the QoS initiator might be triggered by different entities.

User mobility (i.e., a user changing the device and therefore moving the sessions to the new device) is considered to be a special case within the session mobility scenario.

Note that this scenario is different from terminal mobility. Not the terminal (end-system) has moved to a different access point. Both terminals are still connected to an IP network at their original points.

The issues include:

1) Keeping the QoS guarantees negotiated implies that the end-point(s) of communication are changed without changing the reservations.

2) The trigger of the session move might be the user or any other party involved in the session.

3.4 Scenario 4: QoS reservations/negotiation from access to core network

The scenario includes the signaling between access networks and core networks in order to setup and change reservations together with potential negotiation.

The issues to be solved in this scenario are different from previous ones.

1) The entity of reservation is most likely an aggregate.

2) The time scales of reservations might be different (long living reservations of aggregates, rarer re-negotiation).

3) The specification of the traffic (amount of traffic), a particular QoS is guaranteed for, needs to be changed. E.g., in case additional flows are added to the aggregate, the traffic specification of the flow needs to be added if it is not already included in the aggregates specification.

4) The flow specification is more complex including network addresses and sets of different address for the source as well as for the destination of the flow.

3.5 Scenarios 5: QoS reservation/negotiation over administrative boundaries

Signaling between two or more core networks to provide QoS is handled in this scenario. This might also include access to core signaling over administrative boundaries. Compared to the previous one it adds the case, where the two networks are not in the same administrative domain. Basically, it is the inter-domain/inter provider signaling which is handled in here.

The domain boundary is the critical issue to be resolved. Which as various flavors of issues a QoS signaling protocol has to be concerned with.

1) Competing administrations: Normally, only basic information should be exchanged, if the signaling is between competing administrations. Specifically information about core network internals (e.g., topology, technology, etc.) should not be exchanged. Some information exchange about the "access points" of the core networks (which is topology information as well) may need to be exchanged, because it is needed for proper signaling.

2) Additionally, as in scenario 4, signaling most likely is based on aggregates, with all the issues raise there.

3) Authorization: It is critical that the QoS initiator is authorized to perform a QoS path setup.

4) Accountability: It is important to notice that signaling might be used as an entity to charge money for, therefore the interoperation with accounting needs to be available.

4 Problem Statement and Scope

We provide in the following a preliminary architectural picture as a basis for discussion. We will refer to it in the following requirement section.

The overall problems to be solved have been given at a top level by the use cases/scenarios of section 3. However, the problem of QoS has an extremely wide scope and there is a great deal of work already done to provide different components of the solution, such as QoS technologies for example. A basic goal should be to re-use these wherever possible, and to focus requirements work at an early stage on those areas where a new solution is needed (e.g. an especially simple one). We also try to avoid defining requirements related to internal implementation aspects.

In this section, we present a simple conceptual model of the overall QoS problem in order to identify the applicability to NSIS of requirements derived from the use cases, and to clarify the scope of the work, including any open issues. This model also identifies further sources of requirements from external interactions with other parts of an overall QoS solution, clarifies the terminology used, and allows the statement of design goals about the nature of the solution (see section 5).

Note that this model is intended not to constrain the technical approach taken subsequently, simply to allow concrete phrasing of requirements (e.g. requirements about placement of the QoS initiator, or ability to 'drive' particular QoS technologies.)

4.1 Problem Discussion Model

A simple layer model covering a single path segment is shown in figure 1, using the terminology from Section 2.

Roughly, the scope of NSIS within the context of this diagram is assumed to be the interaction between the initiator and controller(s), including selection of signaling protocols to carry the QoS information, and the syntax/semantics of the information that is exchanged. Further statements on assumptions/exclusions are given below. The main elements shown are:

1. Something that starts the request for QoS, the QoS Initiator. This might be in the end system or within some part of the network. The distinguishing feature of the QoS initiator is that it acts on triggers coming (directly or indirectly) from the higher layers in the end systems, mapping the QoS requested by them, and also provides feedback information to the higher layers which might be used by transport layer rate management or adaptive applications.

2. Something that assists in managing QoS further along the path the

QoS controller. The QoS controller does not interact with higher layers, but interacts with the QoS initiator and possibly more QoS controllers on the path, edge to edge or possibly end to end.

3. The QoS initiator and controller(s) interact with each other, path segment by path segment. This interaction involves the exchange of data (QoS control information) over some signaling protocol.

4. The path segment traverses an underlying network (QoS domain or subdomain) covering one or more IP hops. The underlying network uses some local QoS technology. This QoS technology has to be provisioned appropriately for the flow, and this is done by the QoS initiator and controller(s), mapping their QoS control information to technology-related QoS parameters and receiving indications about success or failure in response.

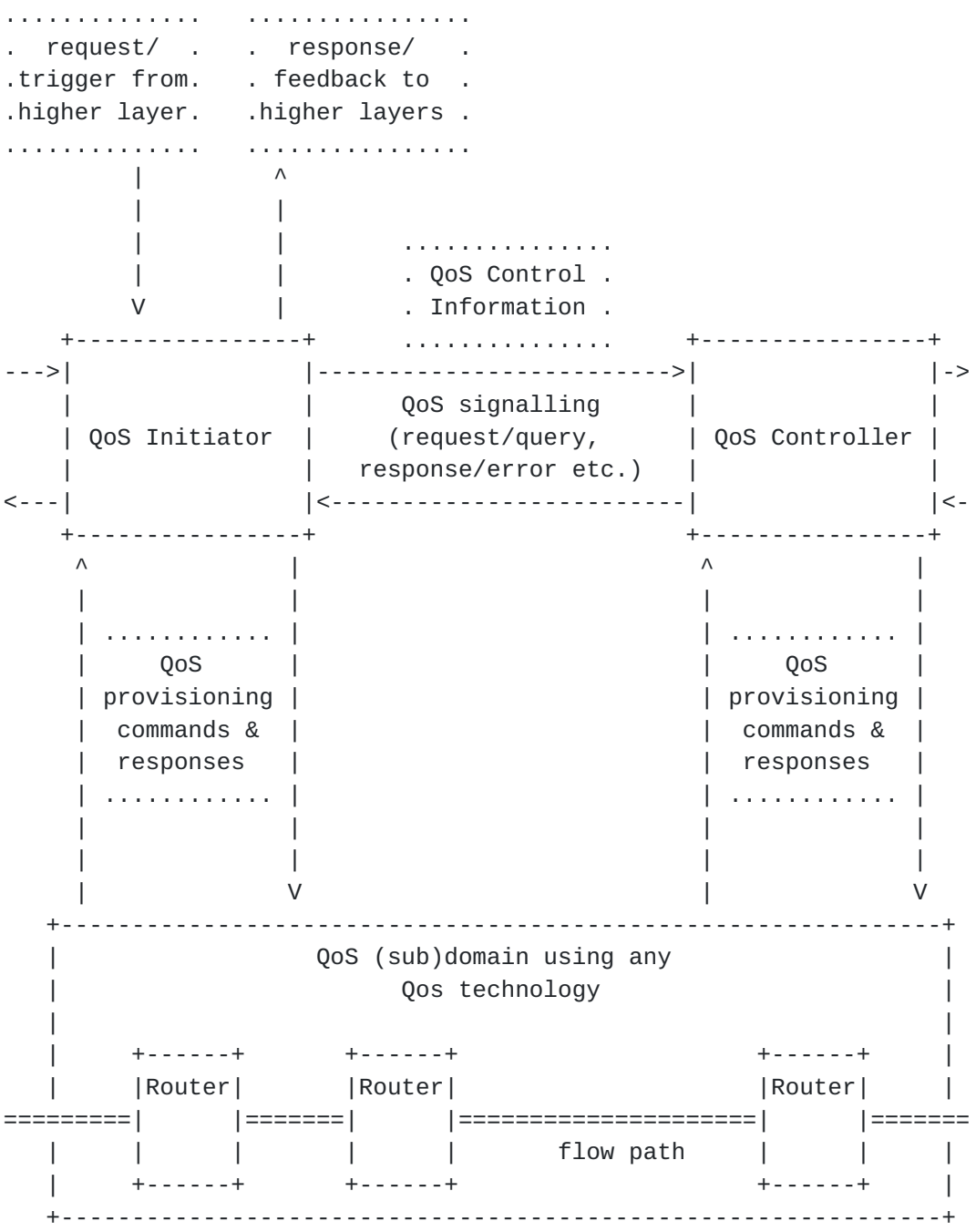

Figure 1: Generic scope of signaling

A second diagram, figure 2, concentrates more on the overall end to end (multiple QoS domains) aspects, in particular:

1. The QoS initiator need not be located at an end system, and the QoS controllers are not assumed to be located on the flow's data path. However, they must be able to identify the ingress and egress points for the flow path as it traverses the domain/subdomain. Any

signaling protocol must be able to find the appropriate QoS controller and carry this ingress/egress point information.

2. We see the network at the level of domains/subdomains rather than individual routers (except in the special case that the domain

contains one link). Domains are assumed to be administrative entities, so security requirements apply to the signaling between them. Subdomains are introduced to allow the fact a given QoS provisioning mechanism may only be used within a part of a domain, typically for a particular subnetwork technology boundary. Aggregation can also take place at subdomain boundaries.

3. Only a unicast flow is shown, with the QoS initiator at or near one end. However, we do not exclude bi-directional flows with the QoS requested by either end. Further QoS initiators may exist on the path. Multicast or anycast flows or flows with variable paths within a subdomain (e.g. to a mobile end system) are also logically possible.

4. Any domain may contain QoS administration functions (e.g. to do with traffic engineering, admission control, policy and so on). These are assumed to interact with the QoS initiator and controllers (and end systems) using standard mechanisms.

Note that Figure 2 does show a generic picture. Specifically, the placement of the QoS initiators and QoS controllers is not fixed. Actually, there are two extreme cases:

- Each router on the data path implements a QoS controller and QoS initiator.

- Only the end systems incorporate a QoS controller and QoS initiator, which means the end systems need to have QoS provisioning capabilities. However this case does not seam to be realistic but shows the flexible allocation of the controller and initiator function.

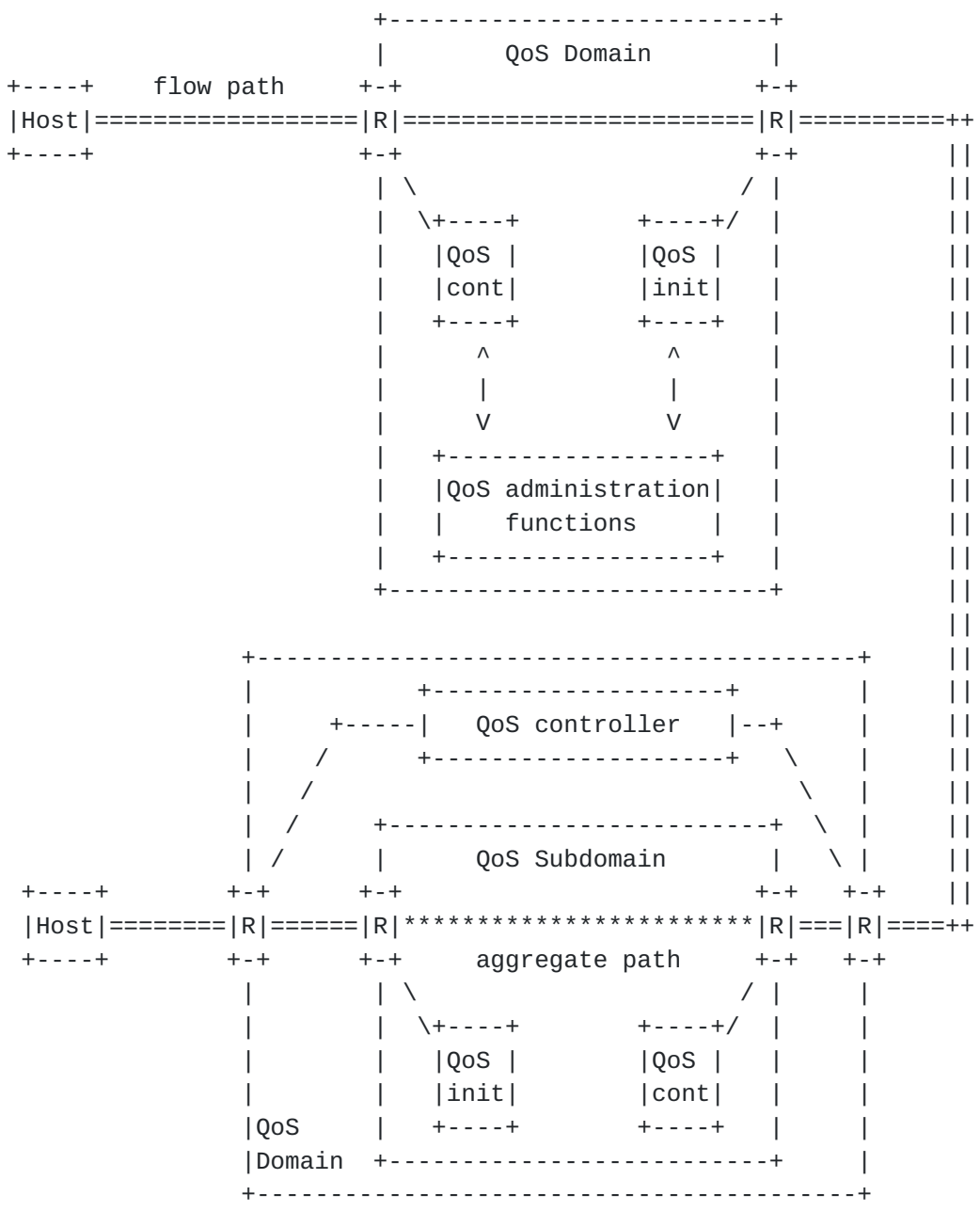

Figure 2: Signaling in a multiple (QoS)domains

4.2 Assumptions and Non-Assumptions

1. The NSIS signaling could run end to end, end to edge, or edge to edge, or network-to-network ((between providers), depending on what point in the network acts as the initiator, and how far towards the other end of the network the signaling propagates. Although the figures show QoS controllers at a very limited number of locations in the network (e.g. at domain or subdomain borders, or even

controlling a complete domain), this is only one possible case. In general, we could expect QoS controllers to become more 'dense' towards the edges of the network, but this is not a requirement. An overprovisioned domain might contain no QoS controllers at all (and be NSIS transparent); at the other extreme, QoS controllers might be

placed at every router. In the latter case, QoS provisioning can be carried out in a local implementation-dependent way without further signalling, whereas in the case of remote QoS controllers, a provisioning protocol might be needed to control the routers along the path. This provisioning protocol is then independent of the end to end NSIS signalling.

2. We do not consider 'pure' end-to-end QoS signaling that is not interpreted anywhere within the network. Such signaling is an application-layer issue and IETF protocols such as SIP etc. can be used.

3. Where the signaling does cover several QoS domains or subdomains, we do not exclude that different signaling protocols are used in each path segment. We only place requirements on the universality of the QoS control information that is being transported. (The goals here would be to allow the use of signaling protocols which are matched to the characteristics of the portion of the network being traversed.) Note that the outcome of NSIS work might result in various protocols or various flavors of the same protocol. This implies the need for the translation of information into QoS domain specific format as well.

4.3 Exclusions

1. Development of specific mechanisms and algorithms for application and transport layer adaptation are not considered, nor are the protocols that would support it.

2. Specific mechanisms (APIs and so on) for interaction between transport/applications and the network layer are not considered, except to clarify the requirements on the negotiation capabilities and information semantics that would be needed of the signaling protocol. The same applies to application adaptation mechanisms.

3. Specific mechanisms for QoS provisioning within a domain/subdomain are not considered. It should be possible to exploit these mechanisms optimally within the end to end context. Consideration of how to do this might generate new requirements for NSIS however. For example, the information needed by an QoS controller to manage a radio subnetwork needs to be provided by the NSIS solution.

4. Specific mechanisms (APIs and so on) for interaction between the network layer and underlying QoS provisioning mechanisms are not considered.

5. Interaction with QoS administration capabilities is not considered. Standard protocols should be used for this (e.g. COPS).

This may imply requirements for the sort of information that should be exchanged between the NSIS network QoS entities.

6. Security issues related to multicasting are outside the scope of the QoS signaling protocol.

Since multicasting is currently not an issue for the QoS protocol, security issues related to multicast are outside the scope. Multicast security may additionally be an application issue that is also outside the scope of the QoS protocol.

7. Protection of non-QoS signaling messages is outside the scope of the QoS protocol

Security protection of data messages transmitted along the established QoS path are outside the scope of the QoS protocol. These security properties are likely to be application specific and may be provided by the corresponding application layer protocol.

5 Requirements

This section defines more detailed requirements for a QoS signaling solution, derived from consideration of the use cases/scenarios, and respecting the framework, scoping assumptions, and terminology considered earlier. The requirements are in subsections, grouped roughly according to general technical aspects: architecture and design goals, topology issues, QoS parameters, performance, security, information, and flexibility.

Two general (and potentially contradictory) goals for the solution are that it should be applicable in a very wide range of scenarios, and at the same time lightweight in implementation complexity and resource requirements in nodes. One approach to this is that the solution could deal with certain requirements via modular components or capabilities, which are optional to implement in individual nodes.

Some of the requirements are technically contradictory. Depending on the scenarios a solution applies to, one or the other requirement is applicable.

Find in Section 6 the MUSTs, SHOULDs, and MAYs

5.1 Architecture and Design Goals

This section contains requirements related to desirable overall characteristics of a solution, e.g. enabling flexibility, or independence of parts of the framework.

5.1.1 Applicability for different QoS technologies.

The QoS signaling protocol must work with various QoS technologies. The information exchanged over the signaling protocol must be in such detail and quantity that it is useful for various QoS technologies.

5.1.2 Resource availability information on request

In some scenarios, e.g., the mobile terminal scenario, it is required to query, whether resources are available, without performing a reservation on the resource. One solution might be a feedback mechanism based on which a QoS inferred handover can take place.

5.1.3 Modularity

A modular design allows for more lightweight implementations, if fewer features are needed. Mutually exclusive solutions are supported. Examples for modularity:

- Work over any kind of network (narrowband / broadband, error-prone / reliable...) - This implies low bandwidth signaling and redundant information must be supported if necessary.

- In case QoS requirements are soft (e.g. banking transactions, gaming), fast and lightweight signaling (e.g., not more than one round-trip time)

- Uni- and bi-directional reservations are possible

5.1.4 Decoupling of protocol and information it is carrying

The signaling protocol(s) used must be clearly separated from the QoS control information being transported. This provides for the independent development of these two aspects of the solution, and allows for this control information to be carried within other protocols, including application layer ones, existing ones or those being developed in the future. The gained flexibility in the information transported allows for the applicability of the same protocol in various scenarios.
However, note that the information carried needs to be the same. Otherwise interoperability is difficult to achieve.

5.1.5 Reuse of existing QoS provisioning

Reuse existing QoS functions and protocols for QoS provisioning within a domain/subdomain unchanged. (Motivation: 'Don't re-invent the wheel'.)

5.1.6 Avoid duplication of [sub]domain signaling functions

The specification of the NSIS signaling protocol should be optimized to avoid duplication of existing [sub]domain QoS signaling and to minimize the overall complexity. (Motivation: we don't want to introduce duplicate feedback or negotiation mechanisms, or complicate the work by including all possible existing QoS signaling

in some form. The function will be placed in the new part if it has to be end-to-end, universal to all network types ('simple/lightweight'), or if it has to be protected by upper layer security mechanisms.)

The point here is that the QoS technology (lower layer stuff) gets re-used unchanged, and we have new signaling above it. But, in many cases the local QoS technology will contain equivalent functions to the NSIS-required ones, just in a technology specific form. Examples of these functions would be error/QoS violation notifications, ability to query for resources and so on. So, there is a danger that our 'lightweight' signaling ends up trying to carry all this information all over again, and (even worse) that the initiator/controller functions have to weigh up nearly equivalent information coming from two directions. However, the basic problem here is that the boundary between new and re-used stuff is pretty shaky. The requirement is trying to scope our problem (a) to eliminate the potential overlap, and (b) to keep the new NSIS stuff simple.

5.1.7 Avoid modularity with large overhead (in various dimensions)

The protocols used for transporting signaling information over various path segments do not need to be the same. Only the QoS control information needs to be exchanged and therefore must interwork between each segment. (Motivation: the protocol can be chosen optimally for the characteristics of the QoS domain being traversed. Also, we allow a choice of protocols in end systems and networks without forcing everyone to implement all choices; the network implementers choice of protocol can be local.)

5.1.8 Possibility to use the signaling protocol for existing local technologies

It needs to be possible to use the new signaling as another local QoS technology in its own right. For example, the treatment of aggregates but possibly for other reasons also. Note that figure 2 shows precisely this case, it is being used there to support signaling QoS for the aggregates.

5.1.9 Independence of signaling and provisioning paradigm

The QoS signaling should be independent of the paradigm and mechanism of QoS provisioning. The independence allows for using the NSIS protocol together with various QoS technologies.

5.2 Signaling Flows

This section contains requirements related to the possible signaling flows that should be supported, e.g. over what parts of the flow path, between what entities (end-systems, routers, middleboxes, management systems), in which direction.

5.2.1 Free placement of QoS Initiator and QoS Controllers functions

The protocol(s) must work in various scenarios such as end-to-end, edge-to-edge, (e.g., just within one providers domain), user-to-network (from end system into the network, ending, e.g., at the

entry to the network and vice versa), network-to-network (e.g., between providers).

Placing the QoS controller and initiator functions at different locations allows for various scenarios to work with the same or similar protocols.

5.2.2 No constraint of the QoS signaling and QoS Controllers to be in the data path.

There is a set of scenarios, where QoS signaling is not on the data path. The QoS Controller being in the data path is one extreme case and useful in certain cases.

There are going to be cases where a centralized entity will take a decision about QoS requests. In this case, there's no question there is no need to have data follow the signalling path.

There are going to be cases wiout a centralized entity managing resources and the signaling will be used as a tool for resource management. For various reasons (such as efficient use of expensive bandwidth), one will want to have fine-grained, fast, and very dynamic control of the resources in the network. -

There are going to be cases where there will be neither signaling nor a centralized entity (overprovisioning). Nothing has to be done anyway.

One can capture the requirement with the following wording: If one views the domain with a QoS technology as a virtual router then NSIS signaling used between those virtual routers must follow the same path as the data.

Routing the signaling protocol along an independent path is desired by network operators/designers. Ideally, the capability to route the protocol along an independent path would give the network designer/operator the option to manage bandwidth utilization through the topology.

There are other possibilities as well. An NSIS protocol must accept all of these possibilities.

5.2.3 Concealment of topology and technology information

The QoS protocol should allow hiding the internal structure of a QoS domain from end-nodes and from other networks. Hence an adversary should not be able to learn the internal structure of a network with the help of the QoS protocol.

In various scenarios, topology information should be hidden for various reasons. From a business point of view, some administrations don't want to reveal the topology and technology used.

5.2.4 Optional transparency of QoS signaling to network

It should be possible that the QoS signaling for some flows traverse path segments transparently, i.e., without interpretation at QoS controllers within the network. An example would be a subdomain within a core network, which only interpreted signaling for aggregates established at the domain edge, with the flow-related signaling passing transparently through it.

5.2.5 Deal with IP fragmentation gracefully

5.3 Additional information beyond signaling of QoS information

This section contains the desired signaling (messages) for other purposes other than that for conveying QoS parameters.

5.3.1 Explicit release of resources

When a QoS reservation is no longer necessary, e.g. because the application terminates, or because a mobile host experienced a hand-off, it must be possible to explicitly release resources.

5.3.2 Ability to signal life-time of a reservation

Information about the lifetime of a reservation allows reducing the reservation update frequency in case of soft state based signaling. Note however, that we do not require in advance reservation, only the expected duration of the reservation should be included.

5.3.3 Possibility for automatic release of resources after failure

When the QoS Initiator goes down, the resources it requested should be released, since they will no longer be necessary.

5.3.4 Possibility for automatic re-setup of resources after recovery

In case of a failure, the reservation can get setup again automatically. It enables sort of a persistent reservation, if the QoS Initiator requests it. In scenarios where the reservations are on a longer time scale, this could make sense to reduce the signaling load in case of failure and recovery.

5.3.5 Prompt notification of QoS violation in case of error / failure to QoS Initiator and QoS Controllers

5.3.6 Feedback about the actually received level of QoS guarantees

The feedback must be independent of streaming technology used.
In some scenarios it might be requested to receive statistics about

the QoS received. E.g., feedback information might be used as input to adaptation mechanisms.

5.3.7 Automatic notification on available resources not been granted before

In many cases, a QoS initiator does want to get a notification when the resource, he requested for some time ago, gets free. In order to keep it simple, information on how long a request is kept and notified. It implies keeping state about requests, which have been rejected.

5.4 Layering

This section contains requirements related to the way the signaling being considered interacts with upper layer functions (users, applications, and QoS administration), and lower layer QoS technologies.

5.4.1 The signaling protocol and QoS control information should be application independent.

However, opaque application information might get transported in the signaling message, without being handled in the network. Development and deployment of new applications should be possible without impacting the network infrastructure. Additionally, QoS protocols are expected to conform to the Internet principles.

5.5 QoS Control Information

This section contains requirements related to the QoS control information that needs to be exchanged.

5.5.1 Mutability information on parameters

It should be possible for the initiator to control the mutability of the QSC information. This prevents from being changed in a non-recoverable way. The initiator should be able to control what is requested end to end, without the request being gradually mutated as it passes through a sequence of domains. This implies that in case of changes made on the parameters, the original requested ones must still be available.

5.5.2 Possibility to add and remove local domain information

It should be possible for the QoS control functions to add and remove local scope elements. E.g., at the entrance to a QoS domain domain-specific information is added, which is used in this domain only, and the information is removed again when a signaling message leaves the domain. The motivation is in the economy of re-use the protocol for domain internal signaling of various information. Where additional information is needed for QoS control within a particular domain, it should be possible to carry this at the same time as the

'end to end' information.)

5.5.3 Simple mapping to lower-layer QoS provisioning parameters

The QoS service classes should be defined taking into account how they will be mapped to QoS provisioning or upper layer parameters. (Motivation: the simpler and more direct this mapping, the more faithful the overall QoS provided to the application.)

5.5.4 Aggregation method specification

The QoS initiator should be able to specify the aggregation method that will be applied to the flow. Since the aggregation method implicitly affects the QoS that applies to the flows, the initiator must be able to influence this.

The point in this requirement is that a reservation for a flow may make sense in isolation, but for scalability we need to aggregate flows together (as we all know). The treatment of the flow within the aggregate won't match the original reservation exactly - there will most likely be an information loss - but the user (QoS initiator) should be able to at least indicate how the aggregation takes place.

As an example, we use a controlled load service request for NRT traffic as an example. The initiator is happy to have just some sort of fair sharing with other flows within the aggregate rather than precise matching of the leaky bucket parameters at every hop along the aggregate path. A second more direct aspect is that a user might want to make a set of reservations but indicate the way they get aggregated together (e.g. set of reservations which are all intended to share a common resource).

As another example, say a user has multiple web sessions running and wants anything sent to him on port 80 to be aggregated onto a single reservation where possible (so that he doesn't have to pay for individual reservations for each session). The requirement is to allow the user to specify a minimum aggregation that he would like for his flows, but without preventing each individual domain from further aggregating flows according to their own QoS technology.

5.5.5 Multiple levels of detail

The QSC should allow for multiple levels of detail in description. (Motivation: someone interpreting the request can tune its own level of complexity by going down to more or less levels of detail. A lightweight implementation within the core could consider only the coarsest level.)

5.5.6 Ranges in specification

The QSC should allow for specification of minimum required QoS and/or desirable QoS. (Motivation: The QoS Service Classes should

allow for ranges to be indicated, to minimize negotiation latency and suppress error notifications during handover events.)

5.5.7 Independence of reservation identifier

A reservation identifier must be used, which is independent of the flow identifier, the IP address of the QoS Initiator, and the flow end-points. Various scenarios in the mobility area require this independence because flows resulting from handoff might have changed end-points etc. but still have the same QoS requirement.

5.5.8 Seamless modification of already reserved QoS

In many case, the reservation needs to be updated (up or downgrade). This must happen seamlessly without service interruption. At least the signaling protocol must allow for it, even if some data path elements might not be capable of doing so.

5.5.9 Signaling must support quantitative, qualitative, and relative QoS specifications

5.5.10 QoS conformance specification

The initiator should be able to indicate how faithfully the QoS provided by the network should conform to that requested. (Motivation: this allows for some flexibility in the level of QoS fulfilled by the network compared to that requested by the initiator deep inside the network.)

5.6 Performance

This section discusses performance requirements and evaluation criteria and the way in which these could and should be traded off against each other in various parts of the solution.

Scalability is a must anyway. However, depending on the scenario the question to which extends the protocol must be scalable.

5.6.1 Scalability in the number of messages received by a signaling communication partner (QoS initiator and controller)

5.6.2 Scalability in number of hand-offs

5.6.3 Scalability in the number of interactions for setting up a reservation

5.6.4 Scalability in the number of state per entity (QoS initiators and QoS controllers)

5.6.5 Scalability in CPU use (end terminal and intermediate nodes)

5.6.6 Low latency

Low latency is only needed in scenarios, where reservations are in a short time scale (e.g. mobile environments), or where human interaction is immediately concerned (e.g., voice communication setup delay)

5.6.7 Low bandwidth consumption

Again only small sets of scenarios call for low bandwidth, mainly those where wireless links are involved.

Note that many of the performance issues are heavily dependent on the scenario assumed and are normally a trade-off between speed, reliability, complexity, and scalability. The trade-off varies in different parts of the network. For example, in radio access networks low bandwidth consumption will overweight the low latency requirement, while in core networks it may be reverse.

5.7 Flexibility

This section lists the various ways the protocol can flexibly be employed.

5.7.1 Aggregation capability, including the capability to select and change the level of aggregation.

5.7.2 Flexibility in the placement of the QoS initiator

It might be the sender or the receiver of content. But also network-initiated reservations are required in various scenarios.

5.7.3 Flexibility in the initiation of re-negotiation (QoS change requests)

Again the sender or the receiver of content might initiate a re-negotiation due to various reasons, such as local resource shortage (CPU, memory on end-system) or a user changed application preference/profiles. But also network-initiated re-negotiation is required in cases, where the network is not able to further guarantee resources etc.

5.7.4 Uni / bi-directional reservation

Both uni-directonal as well as bi-direction reservations must be possible.

5.8 Security

This section discusses security-related requirements.

5.8.1 The QoS protocol must provide strong authentication

A QoS protocol must make provision for enabling various entities to be authenticated against each other using data origin and/or entity authentication. The QoS protocol must enable mutual authentication

between the two communicating entities. The term strong authentication points to the fact that weak plain-text password mechanisms must not be used for authentication.

5.8.2 The QoS protocol must provide means to authorize resource requests

This requirement demands a hook to interact with a policy entity to request authorization data. This allows an authenticated entity to be associated with authorization data and to verify the resource request. Authorization prevents reservations by unauthorized entities, reservations violating policies, theft of service and additionally limits denial of service attacks against parts of the network or the entire network. Additionally it might be helpful to provide some means to inform other protocols of participating nodes within the same administrative domain about a previous successful authorization event.

5.8.3 The QoS signaling messages must provide integrity protection.

The integrity protection of the transmitted signaling messages prevent an adversary from modifying parts of the QoS signaling message and from mounting denial of service attacks against network elements participating in the QoS protocol.

5.8.4 The QoS signaling messages must be replay protected.

To prevent replay of previous signaling messages the QoS protocol must provide means to detect old messages. A solution must cover issues of synchronization problems in the case of a restart or a crash of a participating network element. The use of replay mechanism apart from sequence numbers should be investigated.

5.8.5 The QoS signaling protocol must allow for hop-by-hop security.

Hop-by-Hop security is a well known and proven concept in QoS protocols that allows intermediate nodes that actively participate in the QoS protocol to modify the messages as required by the QoS processing. Note that this requirement does not exclude end-to-end or network-to-network security of a QoS reservation request. End-to-end security between the initiator and the responder may be used to provide protection of non-mutable data fields. Network-to-network security refers to the protection of messages over various hops but not in an end-to-end manner i.e. protected over a particular network.

5.8.6 The QoS protocol should allow identity confidentiality and location privacy.

Identity confidentiality enables privacy and avoids profiling of entities by adversary eavesdropping the signaling traffic along the path. The identity used in the process of authentication may also be hidden to a limited extent from a network to which the initiator is attached. It is however required that the identity provide enough

information for the access network to collect accounting data. Location privacy is an issue for the initiator who triggers the QoS protocol. In some scenarios the initiator may not be willing to reveal location information to the responder.

5.8.7 The QoS protocol should prevent denial-of-service attacks against signaling entities.

To effectively prevent denial-of-service attacks the QoS protocol and the used security mechanisms should not force to do heavy computation to verify a resource request prior authenticating the requesting entity. Additionally the QoS protocol and the used security mechanisms should not require large resource consumption (for example main memory or other additional message exchanges) before a successful authentication was done.

5.8.8 The QoS protocol should support confidentiality of signaling messages.

Based on the signaling information exchanged between nodes participating in the QoS protocol an adversary may learn both the identities and the content of the QoS messages. To prevent this from happening, confidentiality of the QoS requests in a hop-by-hop manner should be provided. Note that hop-by-hop is always required whenever entities actively participating in the protocol must be able to read and eventually modify the content of the QoS messages. This does not exclude the case where one or more network elements are not required to read the information of the transmitted QoS messages.

5.8.9 The QoS protocol should provide hooks to interact with protocols that allow the negotiation of authentication and key management protocols.

The negotiation of an authentication and key management protocols within the QoS protocol is outside the scope of the QoS protocol. This requirement originates from the fact that more than one key management protocol may be used to provide security associations. So both entities must be capable to use the same protocol which may be difficult in a mobile environment with different requirements and different protocols. The goal of such a negotiation step is to determine which authentication and key management protocol to use is executed prior to the execution of the chosen key management protocol. The used key management protocol must however be able to create a security association that matches with the one used in the QoS protocol. A QoS protocol should however provide a way to interact with these negotiation protocols.

5.8.10 The QoS protocol should provide means to interact with key management protocols

Key management protocols typically require a larger number of messages to be transmitted to allow a session key and the corresponding security association to be derived. To avoid the

complex issue of mapping individual authentication and key management protocols to a QoS protocol such a protocol is outside the scope of the QoS protocol. Although the key management protocol may be independent there must be a way for the QoS protocol to

exploit existing security associations to avoid executing a separate key management protocol (or instance of the same protocol) for protocols that closely operate together. If no such security association exists then there should be means for the QoS protocol to trigger a key management protocol to dynamically create the required security associations.

5.9 Mobility

Mobility related requirements are already covered in [2], and are not repeated here.

5.10 Interworking with other protocols and techniques

Hooks must be provided to enable efficient interworking between various protocols and techniques including:

5.10.1 Interworking with IP tunneling

IP tunneling for various applications must be supported. More specifically tunneling for IPSec tunnels are of importance. This mainly impacts the identification of flows. Additionally, care needs to be taken using IPSec for signaling message.

5.10.2 The solution should not constrain either to IPv4 or IPv6

5.10.3 Combination with Mobility management

Combining mobility and QoS signaling should be supported for economic signaling behavior (e.g., negotiation with the new access network: Mobile IP message to acquire new care-of address and query for QoS information could be combined, in order to preserve bandwidth and reduce latency).

5.10.4 Independence from charging model

Signaling must not be constrained by charging models or the charging infrastructure used. However, the end-system should be able to query current pay statistics and to specify user cost functions.

5.10.5 The QoS protocol should provide hooks for AAA protocols

The security mechanism should be developed with respect to be able to collect usage records from one or more network elements.

6 The MUSTs, SHOULDs, and MAYs

In order to prioritize the various requirements from Section 5 in different scenarios (Section 3), we have chosen a table based approach. Each requirement can have different priorities depending

on the scenario given.

Note that the scenario and requirement titles are listed for better reading.

Scenarios
S1: Terminal Mobility
S2: Cellular Networks
S3: Session Mobility
S4: QoS reservations/negotiation from access to core network
S5: QoS reservation/negotiation over administrative boundaries

5.1 Architecture and Design Goals
5.1.1 Applicability for different QoS technologies.
5.1.2 Resource availability information on request
5.1.3 Modularity
5.1.4 Decoupling of protocol and information it is carrying
5.1.5 Reuse of existing QoS provisioning
5.1.6 Avoid duplication of [sub]domain signaling functions
5.1.7 Avoid modularity with large overhead (in various dimensions)
5.1.8 Possibility to use the signaling protocol for existing local technologies
5.1.9 Independence of signaling and provisioning paradigm

	S1	S2	S3	S4	S5
5.1.1	MUST	MUST	MUST	SHOULD	SHOULD
5.1.2	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD
5.1.3	MUST	MUST	MUST	MUST	MUST
5.1.4	MUST	MUST	MUST	MUST	MUST
5.1.5	MUST	MUST	MUST	MUST	MUST
5.1.6	MUST	MUST	MUST	MAY	MAY
5.1.7	MUST	MUST	MUST	MUST	MUST
5.1.8	MUST	MUST	MUST	MUST	MUST
5.1.9	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD

5.2 Signaling Flows
5.2.1 Free placement of QoS Initiator and QoS Controllers functions
5.2.2 No constraint of the QoS signaling and QoS Controllers to be in the data path.

	S1	S2	S3	S4	S5
5.2.1	MUST	MUST	MUST	MUST	MUST
5.2.2	MUST	MUST	MUST	MUST	MUST
5.2.3	MAY	MAY	MAY	SHOULD	MUST
5.2.4	SHOULD	SHOULD	SHOULD	MUST	MUST
5.2.5	MUST	MUST	MUST	MUST	MUST

5.3 Additional information beyond signaling of QoS information
5.3.1 Explicit release of resources
5.3.2 Ability to signal life-time of a reservation
5.3.3 Possibility for automatic release of resources after failure
5.3.4 Possibility for automatic re-setup of resources after recovery
5.3.5 Prompt notification of QoS violation in case of error / failure to QoS Initiator and QoS Controllers
5.3.6 Feedback about the actually received level of QoS guarantees
5.3.7 Automatic notification on available resources not been granted before

	S1	S2	S3	S4	S5
5.3.1	MUST	MUST	MUST	MUST	MUST
5.3.2	SHOULD	SHOULD	SHOULD	MUST	MUST
5.3.3	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD
5.3.4	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD
5.3.5	MUST	MUST	MUST	MUST	MUST
5.3.6	MUST	MUST	MUST	MUST	MUST
5.3.7	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD

5.4 Layering
5.4.1 The signaling protocol and QoS control information should be application independent.

	S1	S2	S3	S4	S5

5.4.1	MUST	MUST	MUST	MUST	MUST

5.5 QoS Control Information

5.5.1 Mutability information on parameters
5.5.2 Possibility to add and remove local domain information
5.5.3 Simple mapping to lower-layer QoS provisioning parameters
5.5.4 Aggregation method specification
5.5.5 Multiple levels of detail
5.5.6 Ranges in specification
5.5.7 Independence of reservation identifier
5.5.8 Seamless modification of already reserved QoS
5.5.9 Signaling must support quantitative, qualitative, and relative QoS specifications
5.5.10 QoS conformance specification

	S1	S2	S3	S4	S5
5.5.1	MUST	MUST	MUST	MUST	MUST
5.5.2	MAY	MAY	MAY	SHOULD	MUST
5.5.3	MAY	MAY	MAY	MUST	MUST
5.5.4	MUST	SHOULD	MUST	SHOULD	SHOULD
5.5.5	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD
5.5.6	MUST	MUST	MUST	MUST	MUST
5.5.7	MUST	MUST	MUST	MUST	MUST
5.5.8	MUST	MUST	MUST	MUST	MUST
5.5.9	MUST	MUST	MUST	MUST	MUST
5.5.10	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD

5.6 Performance
5.6.1 Scalability in the number of messages received by a signaling communication partner (QoS initiator and controller)
5.6.2 Scalability in number of hand-offs
5.6.3 Scalability in the number of interactions for setting up a reservation
5.6.4 Scalability in the number of state per entity (QoS initiators and QoS controllers)
5.6.5 Scalability in CPU use (end terminal and intermediate nodes)
5.6.6 Low latency
5.6.7 Low bandwidth consumption

	S1	S2	S3	S4	S5
5.6.1	MAY	MUST	MAY	MUST	MUST
5.6.2	MUST	MUST	MAY	MAY	MAY

5.6.3	MUST	MUST	MUST	MUST	MUST
5.6.4	MAY	MAY	MAY	MUST	MUST
5.6.5	MUST	MUST	MUST	MUST	MUST
5.6.6	MUST	MUST	MAY	MAY	MAY
5.6.7	MUST	MUST	MUST	MAY	MAY

5.7 Flexibility

5.7.1 Aggregation capability, including the capability to select and change the level of aggregation.

5.7.2 Flexibility in the placement of the QoS initiator

5.7.3 Flexibility in the initiation of re-negotiation (QoS change requests)

5.7.4 Uni / bi-directional reservation

	S1	S2	S3	S4	S5
5.7.1	MAY	MAY	MAY	MUST	MUST
5.7.2	MUST	MUST	MAY	SHOULD	SHOULD
5.7.3	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD
5.7.4	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD

5.8 Security

5.8.1 The QoS protocol must provide strong authentication

5.8.2 The QoS protocol must provide means to authorize resource requests

5.8.3 The QoS signaling messages must provide integrity protection.

5.8.4 The QoS signaling messages must be replay protected.

5.8.5 The QoS signaling protocol must allow for hop-by-hop security.

5.8.6 The QoS protocol should allow identity confidentiality and location privacy.

5.8.7 The QoS protocol must prevent denial-of-service attacks against signaling entities.

5.8.8 The QoS protocol may support confidentiality of signaling messages.

5.8.9 The QoS protocol should provide hooks to interact with

protocols that allow the negotiation of authentication and key management protocols.

5.8.10 The QoS protocol should provide means to interact with key management protocols

	S1	S2	S3	S4	S5
5.8.1	MUST	MUST	MUST	MUST	MUST
5.8.2	MUST	MUST	MUST	MUST	MUST
5.8.3	MUST	MUST	MUST	MUST	MUST
5.8.4	MUST	MUST	MUST	MUST	MUST
5.8.5	MUST	MUST	MUST	MUST	MUST
5.8.6	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD
5.8.7	MUST	MUST	MUST	MUST	MUST
5.8.8	MUST	MUST	MUST	MUST	MUST
5.8.9	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD
5.8.10	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD

5.10 Interworking with other protocols and techniques
5.10.1 Interworking with IP tunneling
5.10.2 The solution should not constrain either to IPv4 or IPv6
5.10.3 Combination with Mobility management
5.10.4 Independence from charging model
5.10.5 The QoS protocol should provide hooks for AAA protocols

	S1	S2	S3	S4	S5
5.10.1	MUST	MUST	MUST	MUST	MAY
5.10.2	MUST	MUST	MUST	MUST	MUST
5.10.3	MUST	MUST	MAY	MAY	MAY
5.10.4	MUST	MUST	MUST	MUST	MUST
5.10.5	SHOULD	SHOULD	SHOULD	SHOULD	SHOULD

7 References

[1] Kempf, J., "Dormant Mode Host Alerting ("IP Paging") Problem

Statement", RFC 3132, June 2001.

[2] Chaskar, H., "Requirements of a QoS Solution for Mobile IP", draft-ietf-mobileip-qos-requirements-01.txt, Work in Progress, August 2001

[3] Manner. J., et al, "Mobility Related Terminology", draft-manner-seamoby-terms-02.txt, Work In Progress, July 2001.

[4] 3GPP, "General Packet Radio Service (GPRS); Service Description Stage 2 v 7.7.0", TS 03.60, June 2001

[5] 3GPP2, "Network Reference Model for cdma2000 Spread Spectrum System, revision B", S.R0005-B, May 2001

[6] Bradner, S., Mankin, A., "Report of the Next Steps in Signaling BOF", draft-bradner-nsis-bof-00.txt, Work in Progress, July 2001

[7] Partain, D., et al, "Resource Reservation Issues in Cellular Radio Access Networks", draft-westberg-rmd-cellular-issues-00.txt, Work in Progress, June 2001

8 Acknowledgments

Quite a number of people have been involved in the discussion of the draft, adding some ideas, requirements, etc. We list them without a guarantee on completeness: Changpeng Fan (Siemens), Krishna Paul (NEC), Maurizio Molina (NEC), Mirko Schramm (Siemens), Andreas Schrader (NEC), Hannes Hartenstein (NEC), Ralf Schmitz (NEC), Juergen Quittek (NEC), Morihisa Momona (NEC), Holger Karl (Technical University Berlin), Xiaoming Fu (Technical University Berlin), Hans-Peter Schwefel (Siemens), Mathias Rautenberg (Siemens), Christoph Niedermeier (Siemens), Andreas Kassler (University of Ulm), Ilya Freytsis.

Some text and/or ideas for text, requirements, scenarios have been taken from a draft written by the following authors: David Partain (Ericsson), Anders Bersten (Telia Research), Marc Greis (Nokia), Georgios Karagiannis (Ericsson), Jukka Manner (University of Helsinki), Ping Pan (Juniper), Vlora Rexhepi (Ericsson), Lars Westberg (Ericsson), Haihong Zheng (Nokia).

9 Author's Addresses

Marcus Brunner (Editor)
NEC Europe Ltd.
Network Laboratories
Adenauerplatz 6
D-69115 Heidelberg
Germany
E-Mail: brunner@ccrle.nec.de (contact)

Robert Hancock, Eleanor Hepworth
Roke Manor Research Ltd

Romsey, Hants, SO51 0ZN
United Kingdom
E-Mail: [robert.hancock|eleanor.hepworth]@roke.co.uk

Cornelia Kappler
Siemens AG
Berlin 13623
Germany
Phone: +49-30-386-32894
E-Mail: cornelia.kappler@icn.siemens.de

Hannes Tschofenig
Siemens AG
Otto-Hahn-Ring 6
81739 Munchen
Germany
Email: Hannes.Tschofenig@mchp.siemens.de

Open Issues/To Dos

1) Scenario
Do we need to add, remove, or change the scenarios?

-I heard the list and in draft-partain-... is one concerning Voice over IP (SIP signaled calls, PSTN to PSTN tunnels over IP)

-What has been missed so far (or was implicit) is the most simple

one (host requests QoS from the network without mobility, wireless etc.)

2) Sender/receiver initiation

What is the requirement concerning data sender or data receiver or both to initiate a QoS request

3) Draft organization
- it might make sense to put the MUST tables together with the requirements for better understanding
- there might be quite a number of requirements, which have the same priority in all the scenarios, because there are pretty generic. We might choose to delete that parts of the table later.

4) MUSTs, SHOULDs, MAY needs discussion

5) Framework text: I assume that we remove this as soon as we have a more stable framework draft.

6) The requirement organization
I heard some voices on the list that the grouping should be more along the lines of host-to-edge, edge to edge etc.
So far I have not changed it, because I though that the requirements heavily depend on the scenario we are looking at.

7) Hemant Chaskar: Section 3.1, items 1) Handoff decision and 2) Trigger sources: The handoff decision and trigger sources should be out of scope of NSIS. NSIS should rather focus only on "establishing" QoS along the packet path after handoff.

I assume this needs more group discussion

8) bi-directional data path setup with one QoS request
I have not seen consensus on whether to require bi-directional data path setup with QoS.

- How can we actually perform bi-directional reservations when the forward and reverse paths are not reciprocal, with respect to routing topology and routing policy of network domains between sender and receiver?

- The need to ensure that the return path is the same as the forwarding path is one of the problems with RSVP, particularly in a mobile environment.

9) Potential requirement: must be implementable in user space (on end hosts)

10) Potential requirement: must provide support for globally defined services as well as private services (Ruediger)

11) Potential requirement: Flexibility in the granularity of reservation (I don't remember who brought it up, but I assume it refers to the flexibility in terms of what size the flow has. Where

size can be bandwidth etc.)

And many more I am sure. But I have them not captured in my bookkeeping.