Network Working Group
Request for Comments: 3102
Category: Experimental

Editors:
M. Borella
CommWorks
J. Lo
Candlestick Networks
Contributors:
D. Grabelsky
CommWorks
G. Montenegro
Sun Microsystems
October 2001

Realm Specific IP: Framework

Status of this Memo

Copyright Notice

IESG Note

The IESG notes that the set of documents describing the RSIP technology imply significant host and gateway changes for a complete implementation. In addition, the floating of port numbers can cause problems for some applications, preventing an RSIP-enabled host from interoperating transparently with existing applications in some cases (e.g., IPsec). Finally, there may be significant operational complexities associated with using RSIP. Some of these and other complications are outlined in section 6 of RFC 3102, as well as in the Appendices of RFC 3104. Accordingly, the costs and benefits of using RSIP should be carefully weighed against other means of relieving address shortage.

Abstract

This document examines the general framework of Realm Specific IP (RSIP). RSIP is intended as a alternative to NAT in which the end-to-end integrity of packets is maintained. We focus on implementation issues, deployment scenarios, and interaction with other layer-three protocols.

Table of Contents

1. Introduction

Network Address Translation (NAT) has become a popular mechanism of enabling the separation of addressing spaces. A NAT router must examine and change the network layer, and possibly the transport layer, header of each packet crossing the addressing domains that the NAT router is connecting. This causes the mechanism of NAT to violate the end-to-end nature of the Internet connectivity, and disrupts protocols requiring or enforcing end-to-end integrity of packets.

While NAT does not require a host to be aware of its presence, it requires the presence of an application layer gateway (ALG) within the NAT router for each application that embeds addressing information within the packet payload. For example, most NATs ship with an ALG for FTP, which transmits IP addresses and port numbers on its control channel. RSIP (Realm Specific IP) provides an alternative to remedy these limitations.

RSIP is based on the concept of granting a host from one addressing realm a presence in another addressing realm by allowing it to use resources (e.g., addresses and other routing parameters) from the second addressing realm. An RSIP gateway replaces the NAT router, and RSIP-aware hosts on the private network are referred to as RSIP hosts. RSIP requires ability of the RSIP gateway to grant such resources to RSIP hosts. ALGs are not required on the RSIP gateway for communications between an RSIP host and a host in a different addressing realm.

RSIP can be viewed as a "fix", of sorts, to NAT. It may ameliorate some IP address shortage problems in some scenarios without some of the limitations of NAT. However, it is not a long-term solution to the IP address shortage problem. RSIP allows a degree of address realm transparency to be achieve between two differently-scoped, or completely different addressing realms. This makes it a useful architecture for enabling end-to-end packet transparency between addressing realms. RSIP is expected to be deployed on privately addresses IPv4 networks and used to grant access to publically addressed IPv4 networks. However, in place of the private IPv4 network, there may be an IPv6 network, or a non-IP network. Thus, RSIP allows IP connectivity to a host with an IP stack and IP applications but no native IP access. As such, RSIP can be used, in conjunction with DNS and tunneling, to bridge IPv4 and IPv6 networks, such that dual-stack hosts can communicate with local or remote IPv4 or IPv6 hosts.

It is important to note that, as it is defined here, RSIP does NOT require modification of applications. All RSIP-related modifications to an RSIP host can occur at layers 3 and 4. However, while RSIP does allow end-to-end packet transparency, it may not be transparent to all applications. More details can be found in the section "RSIP complications", below.

1.1. Document Scope

This document provides a framework for RSIP by focusing on four particular areas:

- Requirements of an RSIP host and RSIP gateway.
- Likely initial deployment scenarios.
- Interaction with other layer-three protocols.
- Complications that RSIP may introduce.

The interaction sections will be at an overview level. Detailed modifications that would need to be made to RSIP and/or the interacting protocol are left for separate documents to discuss in detail.

Beyond the scope of this document is discussion of RSIP in large, multiple-gateway networks, or in environments where RSIP state would need to be distributed and maintained across multiple redundant entities.

Discussion of RSIP solutions that do not use some form of tunnel between the RSIP host and RSIP gateway are also not considered in this document.

This document focuses on scenarios that allow privately-addressed IPv4 hosts or IPv6 hosts access to publically-addressed IPv4 networks.

1.2. Terminology

Private Realm

A routing realm that uses private IP addresses from the ranges (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16) specified in [RFC1918], or addresses that are non-routable from the Internet.

Public Realm

A routing realm with globally unique network addresses.

RSIP Host

A host within an addressing realm that uses RSIP to acquire addressing parameters from another addressing realm via an RSIP gateway.

RSIP Gateway

A router or gateway situated on the boundary between two addressing realms that is assigned one or more IP addresses in at least one of the realms. An RSIP gateway is responsible for parameter management and assignment from one realm to RSIP hosts in the other realm. An RSIP gateway may act as a normal NAT router for hosts within the a realm that are not RSIP enabled.

RSIP Client

An application program that performs the client portion of the RSIP client/server protocol. An RSIP client application MUST exist on all RSIP hosts, and MAY exist on RSIP gateways.

RSIP Server

An application program that performs the server portion of the RSIP client/server protocol. An RSIP server application MUST exist on all RSIP gateways.

RSA-IP: Realm Specific Address IP

An RSIP method in which each RSIP host is allocated a unique IP address from the public realm.

RSAP-IP: Realm Specific Address and Port IP

An RSIP method in which each RSIP host is allocated an IP address (possibly shared with other RSIP hosts) and some number of per-address unique ports from the public realm.

Demultiplexing Fields

Any set of packet header or payload fields that an RSIP gateway uses to route an incoming packet to an RSIP host.

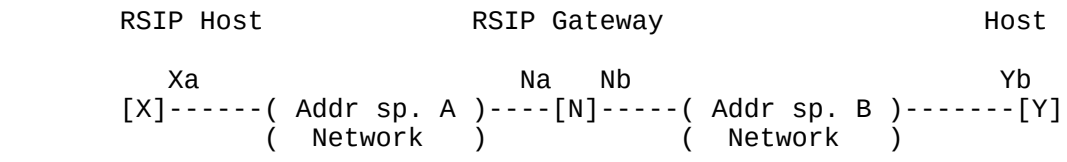
All other terminology found in this document is consistent with that of [RFC2663].

1.3. Specification of Requirements

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this documents are to be interpreted as described in [RFC2119].

2. Architecture

In a typical scenario where RSIP is deployed, there are some number of hosts within one addressing realm connected to another addressing realm by an RSIP gateway. This model is diagrammatically represented as follows:



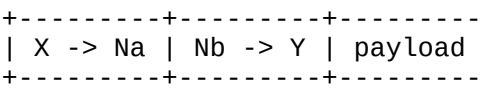
Hosts X and Y belong to different addressing realms A and B, respectively, and N is an RSIP gateway (which may also perform NAT functions). N has two interfaces: Na on address space A, and Nb on address space B. N may have a pool of addresses in address space B which it can assign to or lend to X and other hosts in address space A. These addresses are not shown above, but they can be denoted as Nb1, Nb2, Nb3 and so on.

As is often the case, the hosts within address space A are likely to use private addresses while the RSIP gateway is multi-homed with one or more private addresses from address space A in addition to its public addresses from address space B. Thus, we typically refer to the realm in which the RSIP host resides as "private" and the realm from which the RSIP host borrows addressing parameters as the "public" realm. However, these realms may both be public or private - our notation is for convenience. In fact, address space A may be an IPv6 realm or a non-IP address space.

Host X, wishing to establish an end-to-end connection to a network entity Y situated within address space B, first negotiates and obtains assignment of the resources (e.g., addresses and other routing parameters of address space B) from the RSIP gateway. Upon assignment of these parameters, the RSIP gateway creates a mapping, referred as a "bind", of X’s addressing information and the assigned resources. This binding enables the RSIP gateway to correctly de-multiplex and forward inbound traffic generated by Y for X. If permitted by the RSIP gateway, X may create multiple such bindings on the same RSIP gateway, or across several RSIP gateways. A lease time SHOULD be associated with each bind.

Using the public parameters assigned by the RSIP gateway, RSIP hosts tunnel data packets across address space A to the RSIP gateway. The RSIP gateway acts as the end point of such tunnels, stripping off the outer headers and routing the inner packets onto the public realm. As mentioned above, an RSIP gateway maintains a mapping of the

assigned public parameters as demultiplexing fields for uniquely mapping them to RSIP host private addresses. When a packet from the public realm arrives at the RSIP gateway and it matches a given set of demultiplexing fields, then the RSIP gateway will tunnel it to the appropriate RSIP host. The tunnel headers of outbound packets from X to Y, given that X has been assigned Nb, are as follows:



There are two basic flavors of RSIP: RSA-IP and RSAP-IP. RSIP hosts and gateways MAY support RSA-IP, RSAP-IP, or both.

When using RSA-IP, an RSIP gateway maintains a pool of IP addresses to be leased by RSIP hosts. Upon host request, the RSIP gateway allocates an IP address to the host. Once an address is allocated to a particular host, only that host may use the address until the address is returned to the pool. Hosts MAY NOT use addresses that have not been specifically assigned to them. The hosts may use any TCP/UDP port in combination with their assigned address. Hosts may also run gateway applications at any port and these applications will be available to the public network without assistance from the RSIP gateway. A host MAY lease more than one address from the same or different RSIP gateways. The demultiplexing fields of an RSA-IP session MUST include the IP address leased to the host.

When using RSAP-IP, an RSIP gateway maintains a pool of IP addresses as well as pools of port numbers per address. RSIP hosts lease an IP address and one or more ports to use with it. Once an address / port tuple has been allocated to a particular host, only that host may use the tuple until it is returned to the pool(s). Hosts MAY NOT use address / port combinations that have not been specifically assigned to them. Hosts may run gateway applications bound to an allocated tuple, but their applications will not be available to the public network unless the RSIP gateway has agreed to route all traffic destined to the tuple to the host. A host MAY lease more than one tuple from the same or different RSIP gateways. The demultiplexing fields of an RSAP-IP session MUST include the tuple(s) leased to the host.

3. Requirements

3.1. Host and Gateway Requirements

An RSIP host MUST be able to maintain one or more virtual interfaces for the IP address(es) that it leases from an RSIP gateway. The host MUST also support tunneling and be able to serve as an end-point for

one or more tunnels to RSIP gateways. An RSIP host MUST NOT respond to ARPs for a public realm address that it leases.

An RSIP host supporting RSAP-IP MUST be able to maintain a set of one or more ports assigned by an RSIP gateway from which choose ephemeral source ports. If the host's pool does not have any free ports and the host needs to open a new communication session with a public host, it MUST be able to dynamically request one or more additional ports via its RSIP mechanism.

An RSIP gateway is a multi-homed host that routes packets between two or more realms. Often, an RSIP gateway is a boundary router between two or more administrative domains. It MUST also support tunneling and be able to serve as an end-point for tunnels to RSIP hosts. The RSIP gateway MAY be a policy enforcement point, which in turn may require it to perform firewall and packet filtering duties in addition to RSIP. The RSIP gateway MUST reassemble all incoming packet fragments from the public network in order to be able to route and tunnel them to the proper host. As is necessary for fragment reassembly, an RSIP gateway MUST timeout fragments that are never fully reassembled.

An RSIP gateway MAY include NAT functionality so that hosts on the private network that are not RSIP-enabled can still communicate with the public network. An RSIP gateway MUST manage all resources that are assigned to RSIP hosts. This management MAY be done according to local policy.

3.2. Processing of Demultiplexing Fields

Each active RSIP host must have a unique set of demultiplexing fields assigned to it so that an RSIP gateway can route incoming packets appropriately. Depending on the type of mapping used by the RSIP gateway, demultiplexing fields have been defined to be one or more of the following:

- destination IP address
- IP protocol
- destination TCP or UDP port
- IPSEC SPI present in ESP or AH header (see [RFC3104])
- others

Note that these fields may be augmented by source IP address and source TCP or UDP port.

Demultiplexing of incoming traffic can be based on a decision tree. The process begins with the examination of the IP header of the incoming packet, and proceeds to subsequent headers and then the payload.

- In the case where a public IP address is assigned for each host, a unique public IP address is mapped to each RSIP host.

- If the same IP address is used for more than one RSIP host, then subsequent headers must have at least one field that will be assigned a unique value per host so that it is usable as a demultiplexing field. The IP protocol field SHOULD be used to determine what in the subsequent headers these demultiplexing fields ought to be.

- If the subsequent header is TCP or UDP, then destination port number can be used. However, if the TCP/UDP port number is the same for more than one RSIP host, the payload section of the packet must contain a demultiplexing field that is guaranteed to be different for each RSIP host. Typically this requires negotiation of said fields between the RSIP host and gateway so that the RSIP gateway can guarantee that the fields are unique per-host

- If the subsequent header is anything other than TCP or UDP, there must exist other fields within the IP payload usable as demultiplexing fields. In other words, these fields must be able to be set such that they are guaranteed to be unique per-host. Typically this requires negotiation of said fields between the RSIP host and gateway so that the RSIP gateway can guarantee that the fields are unique per-host.

It is desirable for all demultiplexing fields to occur in well-known fixed locations so that an RSIP gateway can mask out and examine the appropriate fields on incoming packets. Demultiplexing fields that are encrypted MUST NOT be used for routing.

3.3. RSIP Protocol Requirements and Recommendations

RSIP gateways and hosts MUST be able to negotiate IP addresses when using RSA-IP, IP address / port tuples when using RSAP-IP, and possibly other demultiplexing fields for use in other modes.

In this section we discuss the requirements and implementation issues of an RSIP negotiation protocol.

For each required demultiplexing field, an RSIP protocol MUST, at the very least, allow for:

- RSIP hosts to request assignments of demultiplexing fields

- RSIP gateways to assign demultiplexing fields with an associated lease time

- RSIP gateways to reclaim assigned demultiplexing fields

Additionally, it is desirable, though not mandatory, for an RSIP protocol to negotiate an RSIP method (RSA-IP or RSAP-IP) and the type of tunnel to be used across the private network. The protocol SHOULD be extensible and facilitate vendor-specific extensions.

If an RSIP negotiation protocol is implemented at the application layer, a choice of transport protocol MUST be made. RSIP hosts and gateways may communicate via TCP or UDP. TCP support is required in all RSIP gateways, while UDP support is optional. In RSIP hosts, TCP, UDP, or both may be supported. However, once an RSIP host and gateway have begun communicating using either TCP or UDP, they MAY NOT switch to the other transport protocol. For RSIP implementations and deployments considered in this document, TCP is the recommended transport protocol, because TCP is known to be robust across a wide range of physical media types and traffic loads.

It is recommended that all communication between an RSIP host and gateway be authenticated. Authentication, in the form of a message hash appended to the end of each RSIP protocol packet, can serve to authenticate the RSIP host and gateway to one another, provide message integrity, and (with an anti-replay counter) avoid replay attacks. In order for authentication to be supported, each RSIP host and the RSIP gateway MUST either share a secret key (distributed, for example, by Kerberos) or have a private/public key pair. In the latter case, an entity's public key can be computed over each message and a hash function applied to the result to form the message hash.

3.4. Interaction with DNS

An RSIP-enabled network has three uses for DNS: (1) public DNS services to map its static public IP addresses (i.e., the public address of the RSIP gateway) and for lookups of public hosts, (2) private DNS services for use only on the private network, and (3) dynamic DNS services for RSIP hosts.

With respect to (1), public DNS information MUST be propagated onto the private network. With respect to (2), private DNS information MUST NOT be propagated into the public network.

With respect to (3), an RSIP-enabled network MAY allow for RSIP hosts with FQDNs to have their A and PTR records updated in the public DNS. These updates are based on address assignment facilitated by RSIP, and should be performed in a fashion similar to DHCP updates to dynamic DNS [DHCP-DNS]. In particular, RSIP hosts should be allowed to update their A records but not PTR records, while RSIP gateways can update both. In order for the RSIP gateway to update DNS records on behalf on an RSIP host, the host must provide the gateway with its FQDN.

Note that when using RSA-IP, the interaction with DNS is completely analogous to that of DHCP because the RSIP host "owns" an IP address for a period of time. In the case of RSAP-IP, the claim that an RSIP host has to an address is only with respect to the port(s) that it has leased along with an address. Thus, two or more RSIP hosts' FQDNs may map to the same IP address. However, a public host may expect that all of the applications running at a particular address are owned by the same logical host, which would not be the case. It is recommended that RSAP-IP and dynamic DNS be integrated with some caution, if at all.

3.5. Locating RSIP Gateways

When an RSIP host initializes, it requires (among other things) two critical pieces of information. One is a local (private) IP address to use as its own, and the other is the private IP address of an RSIP gateway. This information can be statically configured or dynamically assigned.

In the dynamic case, the host's private address is typically supplied by DHCP. A DHCP option could provide the IP address of an RSIP gateway in DHCPOFFER messages. Thus, the host's startup procedure would be as follows: (1) perform DHCP, (2) if an RSIP gateway option is present in the DHCPOFFER, record the IP address therein as the RSIP gateway.

Alternatively, the RSIP gateway can be discovered via SLP (Service Location Protocol) as specified in [SLP-RSIP]. The SLP template defined allows for RSIP service provisioning and load balancing.

3.6. Implementation Considerations

RSIP can be accomplished by any one of a wide range of implementation schemes. For example, it can be built into an existing configuration protocol such as DHCP or SOCKS, or it can exist as a separate protocol. This section discusses implementation issues of RSIP in general, regardless of how the RSIP mechanism is implemented.

Note that on a host, RSIP is associated with a TCP/IP stack implementation. Modifications to IP tunneling and routing code, as well as driver interfaces may need to be made to support RSA-IP. Support for RSAP-IP requires modifications to ephemeral port selection code as well. If a host has multiple TCP/IP stacks or TCP/IP stacks and other communication stacks, RSIP will only operate on the packets / sessions that are associated with the TCP/IP stack(s) that use RSIP. RSIP is not application specific, and if it is implemented in a stack, it will operate beneath all applications that use the stack.

4. Deployment

When RSIP is deployed in certain scenarios, the network characteristics of these scenarios will determine the scope of the RSIP solution, and therefore impact the requirements of RSIP. In this section, we examine deployment scenarios, and the impact that RSIP may have on existing networks.

4.1. Possible Deployment Scenarios

In this section we discuss a number of potential RSIP deployment scenarios. The selection below are not comprehensive and other scenarios may emerge.

4.1.1. Small / Medium Enterprise

Up to several hundred hosts will reside behind an RSIP-enabled router. It is likely that there will be only one gateway to the public network and therefore only one RSIP gateway. This RSIP gateway may control only one, or perhaps several, public IP addresses. The RSIP gateway may also perform firewall functions, as well as routing inbound traffic to particular destination ports on to a small number of dedicated gateways on the private network.

4.1.2. Residential Networks

This category includes both networking within just one residence, as well as within multiple-dwelling units. At most several hundred hosts will share the gateway’s resources. In particular, many of these devices may be thin hosts or so-called "network appliances" and therefore not require access to the public Internet frequently. The RSIP gateway is likely to be implemented as part of a residential firewall, and it may be called upon to route traffic to particular destination ports on to a small number of dedicated gateways on the private network. It is likely that only one gateway to the public

network will be present and that this gateway's RSIP gateway will control only one IP address. Support for secure end-to-end VPN access to corporate intranets will be important.

4.1.3. Hospitality Networks

A hospitality network is a general type of "hosting" network that a traveler will use for a short period of time (a few minutes or a few hours). Examples scenarios include hotels, conference centers and airports and train stations. At most several hundred hosts will share the gateway's resources. The RSIP gateway may be implemented as part of a firewall, and it will probably not be used to route traffic to particular destination ports on to dedicated gateways on the private network. It is likely that only one gateway to the public network will be present and that this gateway's RSIP gateway will control only one IP address. Support for secure end-to-end VPN access to corporate intranets will be important.

4.1.4. Dialup Remote Access

RSIP gateways may be placed in dialup remote access concentrators in order to multiplex IP addresses across dialup users. At most several hundred hosts will share the gateway's resources. The RSIP gateway may or may not be implemented as part of a firewall, and it will probably not be used to route traffic to particular destination ports on to dedicated gateways on the private network. Only one gateway to the public network will be present (the remote access concentrator itself) and that this gateway's RSIP gateway will control a small number of IP addresses. Support for secure end-to-end VPN access to corporate intranets will be important.

4.1.5. Wireless Remote Access Networks

Wireless remote access will become very prevalent as more PDA and IP / cellular devices are deployed. In these scenarios, hosts may be changing physical location very rapidly - therefore Mobile IP will play a role. Hosts typically will register with an RSIP gateway for a short period of time. At most several hundred hosts will share the gateway's resources. The RSIP gateway may be implemented as part of a firewall, and it will probably not be used to route traffic to particular destination ports on to dedicated gateways on the private network. It is likely that only one gateway to the public network will be present and that this gateway's RSIP gateway will control a small number of IP addresses. Support for secure end-to-end VPN access to corporate intranets will be important.

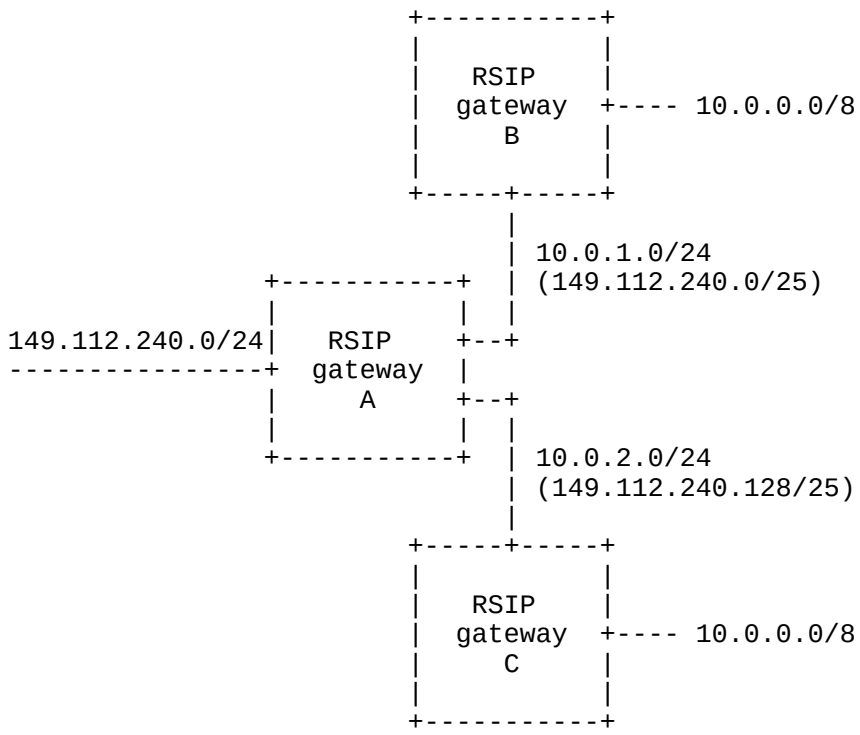
4.2. Cascaded RSIP and NAT

It is possible for RSIP to allow for cascading of RSIP gateways as well as cascading of RSIP gateways with NAT boxes. For example, consider an ISP that uses RSIP for address sharing amongst its customers. It might assign resources (e.g., IP addresses and ports) to a particular customer. This customer may use RSIP to further subdivide the port ranges and address(es) amongst individual end hosts. No matter how many levels of RSIP assignment exists, RSIP MUST only assign public IP addresses.

Note that some of the architectures discussed below may not be useful or desirable. The goal of this section is to explore the interactions between NAT and RSIP as RSIP is incrementally deployed on systems that already support NAT.

4.2.1. RSIP Behind RSIP

A reference architecture is depicted below.



RSIP gateway A is in charge of the IP addresses of subnet 149.112.240.0/24. It distributes these addresses to RSIP hosts and RSIP gateways. In the given configuration, it distributes addresses 149.112.240.0 - 149.112.240.127 to RSIP gateway B, and addresses 149.112.240.128 - 149.112.240.254 to RSIP gateway C. Note that the subnet broadcast address, 149.112.240.255, must remain unclaimed, so that broadcast packets can be distributed to arbitrary hosts behind RSIP gateway A. Also, the subnets between RSIP gateway A and RSIP gateways B and C will use private addresses.

Due to the tree-like fashion in which addresses will be cascaded, we will refer to RSIP gateways A as the 'parent' of RSIP gateways B and C, and RSIP gateways B and C as 'children' of RSIP gateways A. An arbitrary number of levels of children may exist under a parent RSIP gateway.

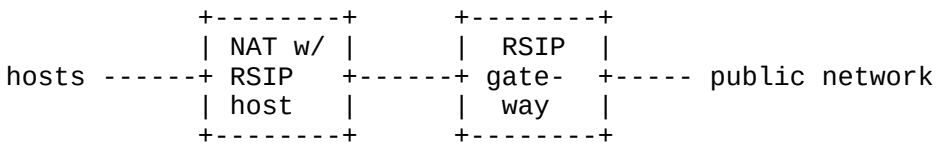
A parent RSIP gateway will not necessarily be aware that the address(es) and port blocks that it distributes to a child RSIP gateway will be further distributed. Thus, the RSIP hosts MUST tunnel their outgoing packets to the nearest RSIP gateway. This gateway will then verify that the sending host has used the proper address and port block, and then tunnel the packet on to its parent RSIP gateway.

For example, in the context of the diagram above, host 10.0.0.1, behind RSIP gateway C will use its assigned external IP address (say, 149.112.240.130) and tunnel its packets over the 10.0.0.0/8 subnet to RSIP gateway C. RSIP gateway C strips off the outer IP header. After verifying that the source public IP address and source port number is valid, RSIP gateway C will tunnel the packets over the 10.0.2.0/8 subnet to RSIP gateway A. RSIP gateway A strips off the outer IP header. After verifying that the source public IP address and source port number is valid, RSIP gateway A transmits the packet on the public network.

While it may be more efficient in terms of computation to have a RSIP host tunnel directly to the overall parent of an RSIP gateway tree, this would introduce significant state and administrative difficulties.

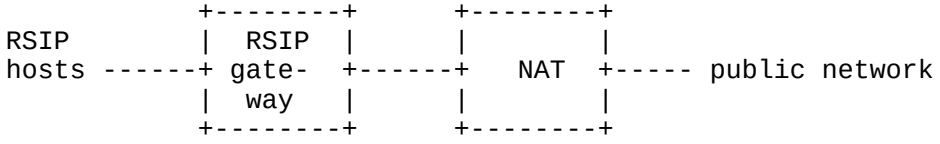
A RSIP gateway that is a child MUST take into consideration the parameter assignment constraints that it inherits from its parent when it assigns parameters to its children. For example, if a child RSIP gateway is given a lease time of 3600 seconds on an IP address, it MUST compare the current time to the lease time and the time that the lease was assigned to compute the maximum allowable lease time on the address if it is to assign the address to a RSIP host or child RSIP gateway.

4.2.2. NAT Behind RSIP



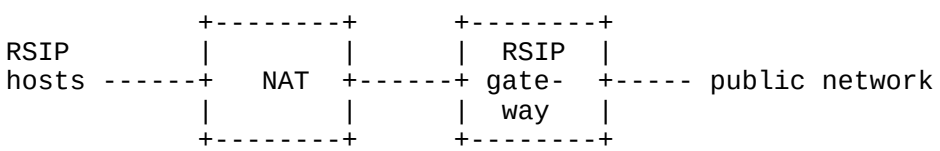
In this architecture, an RSIP gateway is between a NAT box and the public network. The NAT is also equipped with an RSIP host. The NAT dynamically requests resources from the RSIP gateway as the hosts establish sessions to the public network. The hosts are not aware of the RSIP manipulation. This configuration does not enable the hosts to have end-to-end transparency and thus the NAT still requires ALGs and the architecture cannot support IPSEC.

4.2.3. RSIP Behind NAT



In this architecture, the RSIP hosts and gateway reside behind a NAT. This configuration does not enable the hosts to have end-to-end transparency and thus the NAT still requires ALGs and the architecture cannot support IPSEC. The hosts may have transparency if there is another gateway to the public network besides the NAT box, and this gateway supports cascaded RSIP behind RSIP.

4.2.4. RSIP Through NAT



In this architecture, the RSIP hosts are separated from the RSIP gateway by a NAT. RSIP signaling may be able to pass through the NAT if an RSIP ALG is installed. The RSIP data flow, however, will have its outer IP address translated by the NAT. The NAT must not translate the port numbers in order for RSIP to work properly. Therefore, only traditional NAT will make sense in this context.

5. Interaction with Layer-Three Protocols

Since RSIP affects layer-three objects, it has an impact on other layer three protocols. In this section, we outline the impact of RSIP on these protocols, and in each case, how RSIP, the protocol, or both, can be extended to support interaction.

Each of these sections is an overview and not a complete technical specification. If a full technical specification of how RSIP interacts with a layer-three protocol is necessary, a separate document will contain it.

5.1. IPSEC

RSIP is a mechanism for allowing end-to-end IPSEC with sharing of IP addresses. Full specification of RSIP/IPSEC details are in [RSIP-IPSEC]. This section provides a brief summary. Since IPSEC may encrypt TCP/UDP port numbers, these objects cannot be used as demultiplexing fields. However, IPSEC inserts an AH or ESP header following the IP header in all IPSEC-protected packets (packets that are transmitted on an IPSEC Security Association (SA)). These headers contain a 32-bit Security Parameter Index (SPI) field, the value of which is determined by the receiving side. The SPI field is always in the clear. Thus, during SA negotiation, an RSIP host can instruct their public peer to use a particular SPI value. This SPI value, along with the assigned IP address, can be used by an RSIP gateway to uniquely identify and route packets to an RSIP host. In order to guarantee that RSIP hosts use SPIs that are unique per address, it is necessary for the RSIP gateway to allocate unique SPIs to hosts along with their address/port tuple.

IPSEC SA negotiation takes place using the Internet Key Exchange (IKE) protocol. IKE is designated to use port 500 on at least the destination side. Some host IKE implementations will use source port 500 as well, but this behavior is not mandatory. If two or more RSIP hosts are running IKE at source port 500, they MUST use different initiator cookies (the first eight bytes of the IKE payload) per assigned IP address. The RSIP gateway will be able to route incoming IKE packets to the proper host based on initiator cookie value. Initiator cookies can be negotiated, like ports and SPIs. However, since the likelihood of two hosts assigned the same IP address attempting to simultaneously use the same initiator cookie is very small, the RSIP gateway can guarantee cookie uniqueness by dropping IKE packets with a cookie value that is already in use.

5.2. Mobile IP

Mobile IP allows a mobile host to maintain an IP address as it moves from network to network. For Mobile IP foreign networks that use private IP addresses, RSIP may be applicable. In particular, RSIP would allow a mobile host to bind to a local private address, while maintaining a global home address and a global care-of address. The global care-of address could, in principle, be shared with other mobile nodes.

The exact behavior of Mobile IP with respect to private IP addresses has not be settled. Until it is, a proposal to adapt RSIP to such a scenario is premature. Also, such an adaptation may be considerably complex. Thus, integration of RSIP and Mobile IP is a topic of ongoing consideration.

5.3. Differentiated and Integrated Services

To attain the capability of providing quality of service between two communicating hosts in different realms, it is important to consider the interaction of RSIP with different quality of service provisioning models and mechanisms. In the section, RSIP interaction with the integrated service and differentiated service frameworks is discussed.

5.3.1. Differentiated Services

The differentiated services architecture defined in [RFC2475] allows networks to support multiple levels of best-effort service through the use of "markings" of the IP Type-of-Service (now DS) byte. Each value of the DS byte is termed a differentiated services code point (DSCP) and represents a particular per-hop behavior. This behavior may not be the same in all administrative domains. No explicit signaling is necessary to support differentiated services.

For outbound packets from an edge network, DSCP marking is typically performed and/or enforced on a boundary router. The marked packet is then forwarded onto the public network. In an RSIP-enabled network, a natural place for DSCP marking is the RSIP gateway. In the case of RSAP-IP, the RSIP gateway can apply its micro-flow (address/port tuple) knowledge of RSIP assignments in order to provide different service levels to different RSIP host. For RSA-IP, the RSIP gateway will not necessarily have knowledge of micro-flows, so it must rely on markings made by the RSIP hosts (if any) or apply a default policy to the packets.

When differentiated services is to be performed between RSIP hosts and gateways, it must be done over the tunnel between these entities. Differentiated services over a tunnel is considered in detail in [DS-TUNN], the key points that need to be addressed here are the behaviors of tunnel ingress and egress for both incoming and going packets.

For incoming packets arriving at an RSIP gateway tunnel ingress, the RSIP gateway may either copy the DSCP from the inner header to the outer header, leave the inner header DSCP untouched, but place a different DSCP in the outer header, or change the inner header DSCP while applying either the same or a different DSCP to the outer header.

For incoming packets arriving at an RSIP host tunnel egress, behavior with respect to the DSCP is not necessarily important if the RSIP host not only terminates the tunnel, but consumes the packet as well. If this is not the case, as per some cascaded RSIP scenarios, the RSIP host must apply local policy to determine whether to leave the inner header DSCP as is, overwrite it with the outer header DSCP, or overwrite it with a different value.

For outgoing packets arriving at an RSIP host tunnel ingress, the host may either copy the DSCP from the inner header to the outer header, leave the inner header DSCP untouched, but place a different DSCP in the outer header, or change the inner header DSCP while applying either the same or a different DSCP to the outer header.

For outgoing packets arriving at an RSIP gateway tunnel egress, the RSIP gateway must apply local policy to determine whether to leave the inner header DSCP as is, overwrite it with the outer header DSCP, or overwrite it with a different value.

It is reasonable to assume that in most cases, the diffserv policy applicable on a site will be the same for RSIP and non-RSIP hosts. For this reason, a likely policy is that the DSCP will always be copied between the outer and inner headers in all of the above cases. However, implementations should allow for the more general case.

5.3.2. Integrated Services

The integrated services model as defined by [RFC2205] requires signalling using RSVP to setup a resource reservation in intermediate nodes between the communicating endpoints. In the most common scenario in which RSIP is deployed, receivers located within the private realm initiate communication sessions with senders located within the public realm. In this section, we discuss the interaction of RSIP architecture and RSVP in such a scenario. The less common

case of having senders within the private realm and receivers within the public realm is not discussed although concepts mentioned here may be applicable.

With senders in the public realm, RSVP PATH messages flow downstream from sender to receiver, inbound with respect to the RSIP gateway, while RSVP RESV messages flow in the opposite direction. Since RSIP uses tunneling between the RSIP host and gateway within the private realm, how the RSVP messages are handled within the RSIP tunnel depends on situations elaborated in [RFC2746].

Following the terminology of [RFC2476], if Type 1 tunnels exist between the RSIP host and gateway, all intermediate nodes inclusive of the RSIP gateway will be treated as a non-RSVP aware cloud without QoS reserved on these nodes. The tunnel will be viewed as a single (logical) link on the path between the source and destination. End-to-end RSVP messages will be forwarded through the tunnel encapsulated in the same way as normal IP packets. We see this as the most common and applicable deployment scenario.

However, should Type 2 or 3 tunnels be deployed between the tunneling endpoints , end-to-end RSVP session has to be statically mapped (Type 2) or dynamically mapped (Type 3) into the tunnel sessions. While the end-to-end RSVP messages will be forwarded through the tunnel encapsulated in the same way as normal IP packets, a tunnel session is established between the tunnel endpoints to ensure QoS reservation within the tunnel for the end-to-end session. Data traffic needing special QoS assurance will be encapsulated in a UDP/IP header while normal traffic will be encapsulated using the normal IP-IP encapsulation. In the type 2 deployment scenario where all data traffic flowing to the RSIP host receiver are given QoS treatment, UDP/IP encapsulation will be rendered in the RSIP gateway for all data flows. The tunnel between the RSIP host and gateway could be seen as a "hard pipe". Traffic exceeding the QoS guarantee of the "hard pipe" would fall back to the best effort IP-IP tunneling.

In the type 2 deployment scenario where data traffic could be selectively channeled into the UDP/IP or normal IP-IP tunnel, or for type 3 deployment where end-to-end sessions could be dynamically mapped into tunnel sessions, integration with the RSIP model could be complicated and tricky. (Note that these are the cases where the tunnel link could be seen as a expandable soft pipe.) Two main issues are worth considering.

- For RSIP gateway implementations that does encapsulation of the incoming stream before passing to the IP layer for forwarding, the RSVP daemon has to be explicitly signaled upon reception of incoming RSVP PATH messages. The RSIP implementation has to

recognize RSVP PATH messages and pass them to the RSVP daemon instead of doing the default tunneling. Handling of other RSVP messages would be as described in [RFC2746].

- RSIP enables an RSIP host to have a temporary presence at the RSIP gateway by assuming one of the RSIP gateway's global interfaces. As a result, the RSVP PATH messages would be addressed to the RSIP gateway. Also, the RSVP SESSION object within an incoming RSVP PATH would carry the global destination address, destination port (and protocol) tuples that were leased by the RSIP gateway to the RSIP host. Hence the realm unaware RSVP daemon running on the RSIP gateway has to be presented with a translated version of the RSVP messages. Other approaches are possible, for example making the RSVP daemon realm aware.

A simple mechanism would be to have the RSIP module handle the necessary RSVP message translation. For an incoming RSVP signalling flow, the RSIP module does a packet translation of the IP header and RSVP SESSION object before handling the packet over to RSVP. The global address leased to the host is translated to the true private address of the host. (Note that this mechanism works with both RSA-IP and RSAP-IP.) The RSIP module also has to do an opposite translation from private to global parameter (plus tunneling) for end-to-end PATH messages generated by the RSVP daemon towards the RSIP host receiver. A translation on the SESSION object also has to be done for RSVP outbound control messages. Once the RSVP daemon gets the message, it maps them to an appropriate tunnel sessions.

Encapsulation of the inbound data traffic needing QoS treatment would be done using UDP-IP encapsulation designated by the tunnel session. For this reason, the RSIP module has to be aware of the UDP-IP encapsulation to use for a particular end-to-end session. Classification and scheduling of the QoS guaranteed end-to-end flow on the output interface of the RSIP gateway would be based on the UDP/IP encapsulation. Mapping between the tunnel session and end-to-end session could continue to use the mechanisms proposed in [RFC2746]. Although [RFC2746] proposes a number of approaches for this purpose, we propose using the SESSION_ASSOC object introduced because of its simplicity.

5.4. IP Multicast

The amount of specific RSIP/multicast support that is required in RSIP hosts and gateways is dependent on the scope of multicasting in the RSIP-enabled network, and the roles that the RSIP hosts will play. In this section, we discuss RSIP and multicast interactions in a number of scenarios.

Note that in all cases, the RSIP gateway MUST be multicast aware because it is on an administrative boundary between two domains that will not be sharing their all of their routing information. The RSIP gateway MUST NOT allow private IP addresses to be propagated on the public network as part of any multicast message or as part of a routing table.

5.4.1. Receiving-Only Private Hosts, No Multicast Routing on Private Network

In this scenario, private hosts will not source multicast traffic, but they may join multicast groups as recipients. In the private network, there are no multicast-aware routers, except for the RSIP gateway.

Private hosts may join and leave multicast groups by sending the appropriate IGMP messages to an RSIP gateway (there may be IGMP proxy routers between RSIP hosts and gateways). The RSIP gateway will coalesce these requests and perform the appropriate actions, whether they be to perform a multicast WAN routing protocol, such as PIM, or to proxy the IGMP messages to a WAN multicast router. In other words, if one or more private hosts request to join a multicast group, the RSIP gateway MUST join in their stead, using one of its own public IP addresses.

Note that private hosts do not need to acquire demultiplexing fields and use RSIP to receive multicasts. They may receive all multicasts using their private addresses, and by private address is how the RSIP gateway will keep track of their group membership.

5.4.2. Sending and Receiving Private Hosts, No Multicast Routing on Private Network

This scenarios operates identically to the previous scenario, except that when a private host becomes a multicast source, it MUST use RSIP and acquire a public IP address (note that it will still receive on its private address). A private host sending a multicast will use a public source address and tunnel the packets to the RSIP gateway. The RSIP gateway will then perform typical RSIP functionality, and route the resulting packets onto the public network, as well as back to the private network, if there are any listeners on the private network.

If there is more than one sender on the private network, then, to the public network it will seem as if all of these senders share the same IP address. If a downstream multicasting protocol identifies sources

based on IP address alone and not port numbers, then it is possible that these protocols will not be able to distinguish between the senders.

6. RSIP Complications

In this section we document the know complications that RSIP may cause. While none of these complications should be considered "show stoppers" for the majority of applications, they may cause unexpected or undefined behavior. Where it is appropriate, we discuss potential remedial procedures that may reduce or eliminate the deleterious impact of a complication.

6.1. Unnecessary TCP TIME_WAIT

When TCP disconnects a socket, it enters the TCP TIME_WAIT state for a period of time. While it is in this state it will refuse to accept new connections using the same socket (i.e., the same source address/port and destination address/port). Consider the case in which an RSIP host (using RSAP-IP) is leased an address/port tuple and uses this tuple to contact a public address/port tuple. Suppose that the host terminates the session with the public tuple and immediately returns its leased tuple to the RSIP gateway. If the RSIP gateway immediately allocates this tuple to another RSIP host (or to the same host), and this second host uses the tuple to contact the same public tuple while the socket is still in the TIME_WAIT phase, then the host's connection may be rejected by the public host.

In order to mitigate this problem, it is recommended that RSIP gateways hold recently deallocated tuples for at least two minutes, which is the greatest duration of TIME_WAIT that is commonly implemented. In situations where port space is scarce, the RSIP gateway MAY choose to allocate ports in a FIFO fashion from the pool of recently deallocated ports.

6.2. ICMP State in RSIP Gateway

Like NAT, RSIP gateways providing RSAP-IP must process ICMP responses from the public network in order to determine the RSIP host (if any) that is the proper recipient. We distinguish between ICMP error packets, which are transmitted in response to an error with an associated IP packet, and ICMP response packets, which are transmitted in response to an ICMP request packet.

ICMP request packets originating on the private network will typically consist of echo request, timestamp request and address mask request. These packets and their responses can be identified by the tuple of source IP address, ICMP identifier, ICMP sequence number,

and destination IP address. An RSIP host sending an ICMP request packet tunnels it to the RSIP gateway, just as it does TCP and UDP packets. The RSIP gateway must use this tuple to map incoming ICMP responses to the private address of the appropriate RSIP host. Once it has done so, it will tunnel the ICMP response to the host. Note that it is possible for two RSIP hosts to use the same values for the tuples listed above, and thus create an ambiguity. However, this occurrence is likely to be quite rare, and is not addressed further in this document.

Incoming ICMP error response messages can be forwarded to the appropriate RSIP host by examining the IP header and port numbers embedded within the ICMP packet. If these fields are not present, the packet should be silently discarded.

Occasionally, an RSIP host will have to send an ICMP response (e.g., port unreachable). These responses are tunneled to the RSIP gateway, as is done for TCP and UDP packets. All ICMP requests (e.g., echo request) arriving at the RSIP gateway MUST be processed by the RSIP gateway and MUST NOT be forwarded to an RSIP host.

6.3. Fragmentation and IP Identification Field Collision

If two or more RSIP hosts on the same private network transmit outbound packets that get fragmented to the same public gateway, the public gateway may experience a reassembly ambiguity if the IP header ID fields of these packets are identical.

For TCP packets, a reasonably small MTU can be set so that fragmentation is guaranteed not to happen, or the likelihood or fragmentation is extremely small. If path MTU discovery works properly, the problem is mitigated. For UDP, applications control the size of packets, and the RSIP host stack may have to fragment UDP packets that exceed the local MTU. These packets may be fragmented by an intermediate router as well.

The only completely robust solution to this problem is to assign all RSIP hosts that are sharing the same public IP address disjoint blocks of numbers to use in their IP identification fields. However, whether this modification is worth the effort of implementing is currently unknown.

6.4. Application Servers on RSAP-IP Hosts

RSAP-IP hosts are limited by the same constraints as NAT with respect to hosting servers that use a well-known port. Since destination port numbers are used as routing information to uniquely identify an RSAP-IP host, typically no two RSAP-IP hosts sharing the same public

IP address can simultaneously operate publically-available gateways on the same port. For protocols that operate on well-known ports, this implies that only one public gateway per RSAP-IP IP address / port tuple is used simultaneously. However, more than one gateway per RSAP-IP IP address / port tuple may be used simultaneously if and only if there is a demultiplexing field within the payload of all packets that will uniquely determine the identity of the RSAP-IP host, and this field is known by the RSIP gateway.

In order for an RSAP-IP host to operate a publically-available gateway, the host must inform the RSIP gateway that it wishes to receive all traffic destined to that port number, per its IP address. Such a request MUST be denied if the port in question is already in use by another host.

In general, contacting devices behind an RSIP gateway may be difficult. A potential solution to the general problem would be an architecture that allows an application on an RSIP host to register a public IP address / port pair in a public database. Simultaneously, the RSIP gateway would initiate a mapping from this address / port tuple to the RSIP host. A peer application would then be required to contact the database to determine the proper address / port at which to contact the RSIP host’s application.

6.5. Determining Locality of Destinations from an RSIP Host

In general, an RSIP host must know, for a particular IP address, whether it should address the packet for local delivery on the private network, or if it has to use an RSIP interface to tunnel to an RSIP gateway (assuming that it has such an interface available).

If the RSIP hosts are all on a single subnet, one hop from an RSIP gateway, then examination of the local network and subnet mask will provide the appropriate information. However, this is not always the case.

An alternative that will work in general for statically addressed private networks is to store a list of the network and subnet masks of every private subnet at the RSIP gateway. RSIP hosts may query the gateway with a particular target IP address, or for the entire list.

If the subnets on the local side of the network are changing more rapidly than the lifetime of a typical RSIP session, the RSIP host may have to query the location of every destination that it tries to communicate with.

If an RSIP host transmits a packet addressed to a public host without using RSIP, then the RSIP gateway will apply NAT to the packet (if it supports NAT) or it may discard the packet and respond with and appropriate ICMP message.

A robust solution to this problem has proven difficult to develop. Currently, it is not known how severe this problem is. It is likely that it will be more severe on networks where the routing information is changing rapidly that on networks with relatively static routes.

6.6. Implementing RSIP Host Deallocation

An RSIP host MAY free resources that it has determined it no longer requires. For example, on an RSAP-IP subnet with a limited number of public IP addresses, port numbers may become scarce. Thus, if RSIP hosts are able to dynamically deallocate ports that they no longer need, more hosts can be supported.

However, this functionality may require significant modifications to a vanilla TCP/IP stack in order to implement properly. The RSIP host must be able to determine which TCP or UDP sessions are using RSIP resources. If those resources are unused for a period of time, then the RSIP host may deallocate them. When an open socket’s resources are deallocated, it will cause some associated applications to fail. An analogous case would be TCP and UDP sessions that must terminate when an interface that they are using loses connectivity.

On the other hand, this issue can be considered a resource allocation problem. It is not recommended that a large number (hundreds) of hosts share the same IP address, for performance purposes. Even if, say, 100 hosts each are allocated 100 ports, the total number of ports in use by RSIP would be still less than one-sixth the total port space for an IP address. If more hosts or more ports are needed, more IP addresses should be used. Thus, it is reasonable, that in many cases, RSIP hosts will not have to deallocate ports for the lifetime of their activity.

Since RSIP demultiplexing fields are leased to hosts, an appropriately chosen lease time can alleviate some port space scarcity issues.

6.7. Multi-Party Applications

Multi-party applications are defined to have at least one of the following characteristics:

- A third party sets up sessions or connections between two hosts.

- Computation is distributed over a number of hosts such that the individual hosts may communicate with each other directly.

RSIP has a fundamental problem with multi-party applications. If some of the parties are within the private addressing realm and others are within the public addressing realm, an RSIP host may not know when to use private addresses versus public addresses. In particular, IP addresses may be passed from party to party under the assumption that they are global endpoint identifiers. This may cause multi-party applications to fail.

There is currently no known solution to this general problem. Remedial measures are available, such as forcing all RSIP hosts to always use public IP addresses, even when communicating only on to other RSIP hosts. However, this can result in a socket set up between two RSIP hosts having the same source and destination IP addresses, which most TCP/IP stacks will consider as intra-host communication.

6.8. Scalability

The scalability of RSIP is currently not well understood. While it is conceivable that a single RSIP gateway could support hundreds of RSIP hosts, scalability depends on the specific deployment scenario and applications used. In particular, three major constraints on scalability will be (1) RSIP gateway processing requirements, (2) RSIP gateway memory requirements, and (3) RSIP negotiation protocol traffic requirements. It is advisable that all RSIP negotiation protocol implementations attempt to minimize these requirements.

7. Security Considerations

RSIP, in and of itself, does not provide security. It may provide the illusion of security or privacy by hiding a private address space, but security can only be ensured by the proper use of security protocols and cryptographic techniques.

8. Acknowledgements

The authors would like to thank Pyda Srisuresh, Dan Nessett, Gary Jaszewski, Ajay Bakre, Cyndi Jung, and Rick Cobb for their input. The IETF NAT working group as a whole has been extremely helpful in the ongoing development of RSIP.

9. References

[DHCP-DNS] Stapp, M. and Y. Rekhter, "Interaction Between DHCP and DNS", Work in Progress.

[RFC2983] Black, D., "Differentiated Services and Tunnels", RFC 2983, October 2000.

[RFC3104] Montenegro, G. and M. Borella, "RSIP Support for End-to-End IPSEC", RFC 3104, October 2001.

[RFC3103] Borella, M., Grabelsky, D., Lo, J. and K. Taniguchi, "Realm Specific IP: Protocol Specification", RFC 3103, October 2001.

[RFC2746] Terzis, A., Krawczyk, J., Wroclawski, J. and L. Zhang, "RSVP Operation Over IP Tunnels", RFC 2746, January 2000.

[RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G.J. and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC2002] Perkins, C., "IP Mobility Support", RFC 2002, October 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to indicate requirement levels", BCP 14, RFC 2119, March 1997.

[RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.

[RFC2205] Braden, R., Zhang, L., Berson, S., Herzog, S. and S. Jamin, "Resource Reservation Protocol (RSVP)", RFC 2205, September 1997.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z. and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, December 1998.

[RFC3105] Kempf, J. and G. Montenegro, "Finding an RSIP Server with SLP", RFC 3105, October 2001.

10. Authors' Addresses

Michael Borella
CommWorks
3800 Golf Rd.
Rolling Meadows IL 60008

Phone: (847) 262-3083
EMail: mike_borella@commworks.com

Jeffrey Lo
Candlestick Networks, Inc
70 Las Colinas Lane,
San Jose, CA 95119

Phone: (408) 284 4132
EMail: yidarlo@yahoo.com

David Grabelsky
CommWorks
3800 Golf Rd.
Rolling Meadows IL 60008

Phone: (847) 222-2483
EMail: david_grabelsky@commworks.com

Gabriel E. Montenegro
Sun Microsystems
Laboratories, Europe
29, chemin du Vieux Chene
38240 Meylan
FRANCE

Phone: +33 476 18 80 45
EMail: gab@sun.com

11. Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.