Network Working Group
Internet-Draft
Expires: July 1, 2004

S. Vaarala
A. Nuopponen
Netseal
F. Adrangi
Intel
January 2004

Optimized Mobile IPv4 UDP Encapsulation
draft-vaarala-mip4-optudp-00

Status of this Memo

Copyright Notice

Abstract

This document specifies an extension to Mobile IPv4 UDP encapsulation (RFC 3519) which enables optimization of overhead when UDP encapsulation is used, and most of the mobile node's data traffic is destined to one particular correspondent node.

Table of Contents

1. Introduction

1.1 Overview

In some cases the majority of a Mobile IPv4 mobile node's traffic is destined to a single correspondent node. For instance, this is the case when Mobile IPv4 is used to provide mobility for IPsec (i.e. IPsec runs over Mobile IPv4); similarly, when using VoIP, packet size is small (and overhead thus relatively large), while almost all traffic is sent to a single CN.

The IP-IP [2] or IP-over-UDP [4] encapsulation ordinarily used by Mobile IPv4 is for the most part unnecessary in such cases, and consequently encapsulation overhead can be minimized in a straightforward manner.

This document specifies an extension to the Mobile IPv4 UDP encapsulation specification [4] which enables optimization of overhead when UDP encapsulation is used. The following things are specified:

- a new extension to request optimized UDP encapsulation for a certain "preferred CN";
- a new extension to acknowledge and enable use of optimized UDP encapsulation;
- a new flag to MIP Tunnel Data message header, indicating that optimized UDP encapsulation was used; and
- a processing model for inbound and outbound packets at the MN and the HA.

It is assumed that peers supporting this extension also support RFC 3519; the MIP Tunnel Data message type defined in RFC 3519 is used as a basis for the extension.

Use of optimized encapsulation together with foreign agent care-of address is not supported. The problem is that without a home address, an FA cannot easily demultiplex traffic correctly. (It may be possible to overcome this problem by e.g. creative use UDP source port by the FA; these approaches are not described in this document.)

2. Negotiation

2.1 Overview

Optimized UDP encapsulation adds a new field into the mobility binding state: Preferred-CN. Preferred-CN of 0.0.0.0 (default) implies that optimized encapsulation is disabled, while a non-zero value implies that optimized encapsulation is enabled. The latter case requires a successful negotiation, using the two MIPv4 options

described in this section.

The negotiation process is depicted in the figure below.

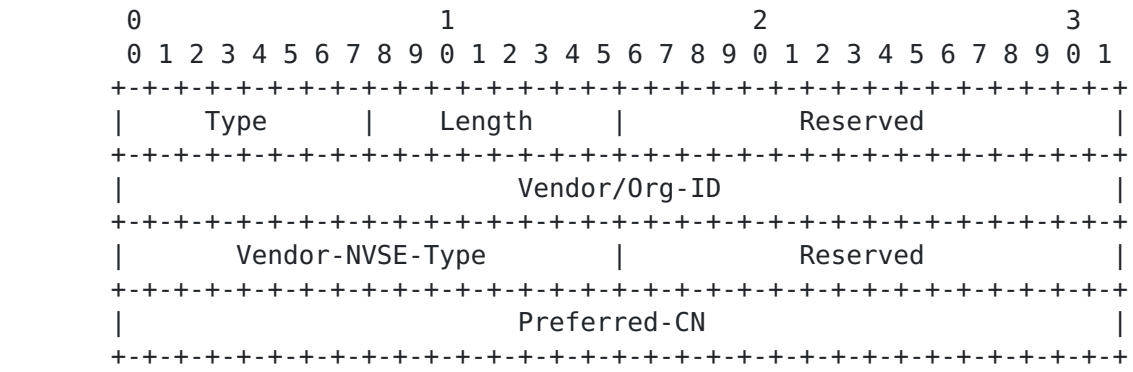
```
    MN                          HA
    ==                          ==
    ---------------------------->
     rrq w/ Optimized UDP Tunnel Request Extension (skippable)
                    Preferred-CN: a.b.c.d
            UDP Tunnel Request Extension (skippable)
                    Flags: F=<any> R=<any>
                    Encapsulation: 0x00 or 0x04
            <other extensions>

    <----------------------------
     rrp w/ Optimized UDP Tunnel Reply Extension (non-skippable)
            UDP Tunnel Reply Extension (non-skippable)
                    Reply Code: 0
                    Flags: F=<any>
                    Keepalive Interval: <any>
            <other extensions>
```

Optimized encapsulation is enabled if both (1) UDP tunneling is established using a successful exchange of UDP Tunnel Request and UDP Tunnel Reply extensions, and (2) a successful exchange of Optimized UDP Tunnel Request and Optimized UDP Tunnel Reply extensions takes place. If optimized encapsulation is desired even when NAT devices are not detected, the F-flag (force encapsulation) of the UDP Tunnel Request Extension should be used.

2.2 Optimized UDP Tunnel Request Extension

This extension is a (skippable) Normal Vendor Specific Extension (NVSE) [3]. The extension MUST be placed before the Mobile-Home Authentication Extension. The format of the extension is described below.



Type NVSE-TYPE-NUMBER 134

Length 14

Reserved Reserved for future use, MUST be set to 0 when sending and ignored when receiving

Vendor/Org-ID
9213 (allocated to Netseal by IANA)

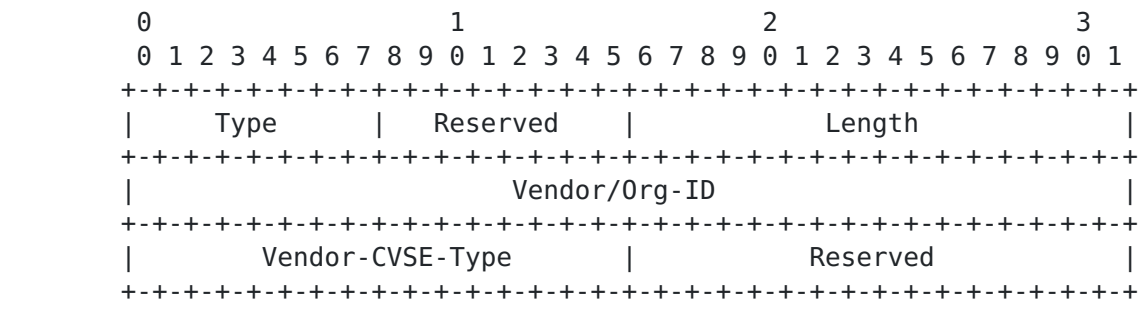
Vendor-NVSE-Type
9

Preferred-CN
IPv4 address of the preferred correspondent node.

Preferred-CN MUST be non-zero. The HA MUST ignore the extension if Preferred-CN is zero.

2.3 Optimized UDP Tunnel Reply Extension

This extension is a (non-skippable) Critical Vendor Specific Extension (CVSE) [3]. The extension MUST be placed before the Mobile-Home Authentication Extension. The format of the extension is described below.



Type CVSE-TYPE-NUMBER 38

Length 8

Vendor/Org-ID
  9213 (allocated to Netseal by IANA)

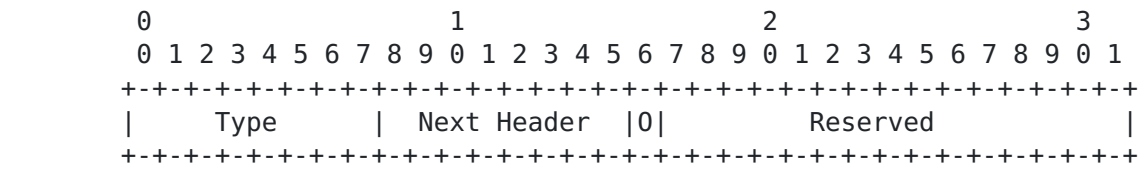
Vendor-CVSE-Type
  10

Reserved Reserved for future use, MUST be set to 0 when sending and ignored when receiving

The extension has no content. The purpose of the extension is to acknowledge and accept support for the optimized encapsulation mechanism for the binding in question.

3. MIP Tunnel Data flag

The O-flag is added to the MIP Tunnel Data header (specified in RFC 3519 [4] Section 3.3) as follows:



Type 4

Next Header Indicates the type of tunnelled data, using the same numbering as the IP Header Protocol Field.

O If 1, optimized encapsulation has been used in encapsulating the packet.

Reserved	Reserved for future use. MUST be set to 0 on sending, MUST be ignored on reception.

4. Packet processing

4.1 MN outbound packet processing

The optimized encapsulation applies only to data packets (i.e. not Mobile IPv4 signaling packets). The following additional conditions must be fulfilled:

- reverse tunneling is enabled;
- the mobility binding used for processing the packet has UDP encapsulation enabled, and the protocol type being encapsulated is IPv4;
- the mobility binding used for processing the packet has optimized encapsulation enabled;
- the destination address of the packet is Preferred-CN;
- the packet is sufficiently small to guarantee that it will not be fragmented, taking into account the Mobile IPv4 UDP encapsulation overhead;
- there are no IPv4 options.

When these conditions are met, the optimized encapsulation may be used. The conditions essentially guarantee that the receiver (HA) will be able to reconstruct the inner IPv4 header which is omitted in the optimized encapsulation.

Note that although the conditions are quite strict, they apply to almost all data packets sent using Mobile IPv4 in practice (assuming they are sent to Preferred-CN, of course). For instance, VoIP voice data packets fulfill these conditions, and can thus use the optimization.

The optimized encapsulation process is as follows:

1. Receive from stack: IP1 ! TCP ! data
2. Check preconditions
3. Strip IP header: TCP ! data
4. Encapsulate: IP2 ! UDP ! MIP-TD(next=TCP, O=1) ! TCP ! data
5. Send to network

Note that the processing of any other Mobile IPv4 packets (signaling, unoptimized UDP tunneling, etc) is not changed.

4.2 MN inbound packet processing

Packets using optimized UDP encapsulation can be uniquely identified based on the MIP Tunnel Data header O-flag. If the O-flag is set, the MN MUST check whether the current binding (for that care-of address) has optimized encapsulation enabled. If not, the incoming packet MUST be dropped.

If fragments are received, the fragments MUST be reassembled into a complete packet before the packet processing described here takes place. (Even though fragments are not sent by MN or HA, the packets may still be fragmented on the route.)

The process is as follows:

1. Receive from n/w: IP2 ! UDP ! MIP-TD(next=TCP, O=1) ! TCP ! data
2. Check preconditions
3. Decapsulate: TCP ! data
4. Reconstruct: IP1 ! TCP ! data
5. Send to stack

Note that the processing of any other Mobile IPv4 packets (signaling, unoptimized UDP tunneling, etc) is not changed.

The packet is first decapsulated: the outermost IP header, the UDP header, and the MIP Tunnel Data header are removed. Then, a new IPv4 header is constructed based on the Preferred-CN stored in the binding. If Preferred-CN is 0.0.0.0, the packet MUST be dropped as IPv4 header reconstruction cannot be done.

The new IPv4 header is reconstructed as follows:

- Version: 4
- IHL: 5
- Type of Service: Copied from received IP header
- Total Length: Computed based on received IP header Total Length field
- Identification: Copied from received IP header
- Flags: DF=0, MF=0, reserved=0
- Fragment Offset: 0
- Time to Live: Copied from received IP header
- Protocol: Copied from MIP Tunnel Data header "Next header" field
- Header Checksum: As specified in RFC 791 [1]
- Source Address: Preferred-CN
- Destination Address: Home address
- Options and Padding: Not used

4.3 HA outbound packet processing

HA outbound processing is the same as MN outbound processing, except that reverse tunneling does not need to be checked.

4.4 HA inbound packet processing

HA inbound processing is the same as MN inbound processing, except that addresses in the reconstructed IPv4 header are reversed.

4.5 Transparency issues

Optimized encapsulation is not completely transparent. An application using the IPv4 service provided by Mobile IPv4 can detect the difference between normal and optimized encapsulation.

For instance, the Time-to-Live (TTL) field of the application IPv4 header (before Mobile IPv4 encapsulation) behaves differently depending on whether optimized encapsulation is used or not.

(The differences should be documented here in more detail.)

5. Security considerations

Establishing the Preferred-CN is protected using ordinary Mobile IPv4 integrity protection. Tampering with a data packet on the route to a Mobile IPv4 peer has little effect on security compared to Mobile IPv4 without optimization: in both cases fields of the "inner" IPv4 header can easily be tampered with.

6. Alternatives and other issues

New message type instead of flag in MIP Tunnel Data.
- Benefits and drawbacks?

Add option to describe multiple preferred CNs in the registration, and use multiple bits to select destination.
- Adds complexity - which practical scenarios benefit?

Add FA support
- Negotiation protocol can be easily extended.
- Incoming packets from HA: what algorithm should FA use to demultiplex packets? Or leave out of scope?

7. Acknowledgements

Several people provided feedback on sketches of how optimization could be done most comfortably. In particular, we would like to

thank Nuutti Kotivuori for implementor input.

References

[1] Postel, J., "INTERNET PROTOCOL", RFC 791, September 1981.

[2] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

[3] Dommety, G. and K. Leung, "Mobile IP Vendor/ Organization-Specific Extensions", RFC 3115, April 2001.

[4] Levkowetz, H. and S. Vaarala, "Mobile IP Traversal of Network Address Translation (NAT) Devices", RFC 3519, April 2003.

Authors' Addresses

Sami Vaarala
Netseal
Niittykatu 6
Espoo 02600
Finland

EMail: sami.vaarala@iki.fi
URI: http://www.netseal.com/

Antti Nuopponen
Netseal
Niittykatu 6
Espoo 02600
Finland

EMail: antti.nuopponen@iki.fi
URI: http://www.netseal.com/

Farid Adrangi
Intel
2111 N.E. 25th Avenue
Hillsboro OR 97124
USA

Phone: 503-712-1791
EMail: farid.adrangi@intel.com
URI: http://www.intel.com/

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

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