HIP Extensions for the Traversal of Network Address Translators

draft-schmitt-hip-nat-traversal-01

Miika Komu <miika@iki.fi>
Abhinav Pathak <abhinav.pathak@hiit.fi>
Lars Eggert <lars.eggert@netlab.nec.de>
Martin Stiemerling <stiemerling@netlab.nec.de>
Vivien Schmitt <vivien.schmitt@gmx.de>

Table of Contents

- Summary of changes
- Issues and their current solutions or solution proposals in the draft
- Next steps
- References

Summary of Changes from 00 to 01

- Resolved 10 Issues
  - Server behind NAT
  - Port numbering
  - NAT keepalives after handovers
  - Mobility and privacy
  - Hairpin translation
  - Editorial suggestions
- 3 Unresolved Issues
  - Multihoming, protection of keepalives, split draft

Issue 1: Reuse of IKE Ports

- Problem: reuse the same UDP port as IKE
  - Benefit: no extra holes to firewalls
    - NAT+firewall combinations are out of scope of the draft
  - Drawback: requires software modifications when a HIP and IKE implementation are running on the same host
- Solution in draft version 01:
  - The initiator can use the IKE ports when the ports are unoccupied
  - Responder listen only to the HIP-NAT port

Issue 2: Random Source Port

- Problem:
  - Cone NAT (does not change port number, only IP)
  - Multiple hosts behind the NAT
  - Fixed port number for UDP
  - Result: only one host can be traverse the NAT
- Solution in draft-01:
  - Initiator can select a random UDP source port

Issue 3: Responder Behind NAT

- Problem:
  - Responder is behind a NAT
  - NAT drops the I1
- Solution in draft-01:
  - Responder registers to rendezvous server to open a hole in the responder NAT
  - Initiator sends I1 through rendezvous server that relays the packet to the responder using the hole in the NAT
  - Does not work with symmetric NATs

Issue 4: Rendezvous and NAT

- The rendezvous server description was not present in the earlier draft => added text
- Limitation: does not work with symmetric NATs

Issue 5: Mobility, NATs and Privacy

- Problem: draft-00 specified that a mobile node communicates even private addresses to its peer after it relocates to a NATted network
  - benefit: easy to implement; NAT implementation extensions only add or remove UDP headers
  - drawback: negative privacy implications
- Solution in draft-01:
  - Solved in favour of privacy
  - Implementation has to filter all private addresses from UPDATE LOCATORs

Issue 6: Inner Addresses

- Problem: draft-00 only was referring only to HIT type of inner addresses, not LSI
- Solution: removed most of the text referring to the type of inner addresses because it is not related to the draft

Issue 7: Editorial Comments

- Various editorial comments from several people
- The text has been modified based on the feedback

Issue 8: Data Channel Reactivation after a Handover

- Problem:
  - draft-00 define separate channels (=UDP ports) for control and data traffic
  - After mobile node moved to a NATted network, it had to reactivate the data channel using a keepalive, or otherwise NAT just drop the UDP encapsulated ESP traffic.
  - ESP keepalive packet does not contain an SPI, so it is not possible to determine unambiguously the corresponding host association
- Solution in draft-01:
  - Joined the control and data channels (single port)
  - UPDATE message activates the shared channel

Issue 9: Hairpin translation

- Hairpin translation = two hosts are behind the same NAT but were not able to detect it using e.g. STUN
- Problem: the hosts communicate through the NAT even though they could communicate with each other directly => unnecessary network traffic for the NAT
- Solution:
  - The host tries to send I1 first without UDP encapsulation
  - If no R1 was received within a small time period, the host assumes the presence of NAT and starts to encapsulate the I1 retransmissions within UDP

Issue 10: NAT and Multihoming

- A host can, at the same time, have interfaces both behind NATs and in publicly addressable networks
- We need to define how the details work in the draft

Issue 11: Responder is NAT

- Problem: experimentation showed us that HIP implementations may optimize routes when responder = NAT device:
  - I1(10.0.0.123, 130.233.53.72)
  - R1(10.0.0.254, 10.0.0.123)
- Solution in draft-01: added some hints for implementors regarding to this

Issue 12: Keepalives with HMAC and Signatures

- Problem: should we include HMACs and signatures in HIP keepalive messages?
  - Benefit: protected keepalives??
  - Drawback: keepalives consume CPU cycles
- Solution
  - Decided to exclude HMACs and signatures from HIP keepalives in favour of efficiency

Issue 13: Split Mobility and Multihoming to a Separate Draft?

- Problem: the draft is getting lengthy – should we separate mobility and multihoming to separate draft
- Solution: no?

Next Steps

- Should we split mobility and multihoming to a separate draft (no)?
- Define multihoming NAT extensions
- Accept as an official WG item?

References

- Contact e.g. Miika Komu <miika@iki.fi> or any of the authors
- The draft:
  - http://www.ietf.org/internet-drafts/draft-schmitt-hip-nat-traversal-01.txt
- Issue tracker:
  - http://hip4inter.net/cgi-bin/roundup.cgi/hip-nat
- NAT enabled HIP implementations:
  - HIPL: http://infrahip.hiit.fi/hipl
  - OpenHIP: http://www.openhip.org/