NAT66
draft-mrw-behave-nat-02.txt

Margaret Wasserman

mrw@sandstorm.net

Why Do People Deploy NAT?

- Many home/small business users deploy NAT to amplify limited IPv4 address space
  - Won't be needed with IPv6
- Some deploy NAT as a "simple security" solution
  - Better provided by more secure, more flexible firewalls
- However, many enterprises that have firewalls and plenty of IPv4 "swamp space" use NAT for...
  - Address Independence
  - Topology Hiding

Address Independence

- The IP addresses used inside the local network (for nodes, ACLs, logs) do not need to be renumbered if the ISP changes an enterprise's global address prefix
- The IP addresses used inside the local network (for nodes, ACLs, logs) do not need to be renumbered when a site changes ISPs
- It is not necessary for an administrator to convince an ISP to route his or her provider-independent addresses

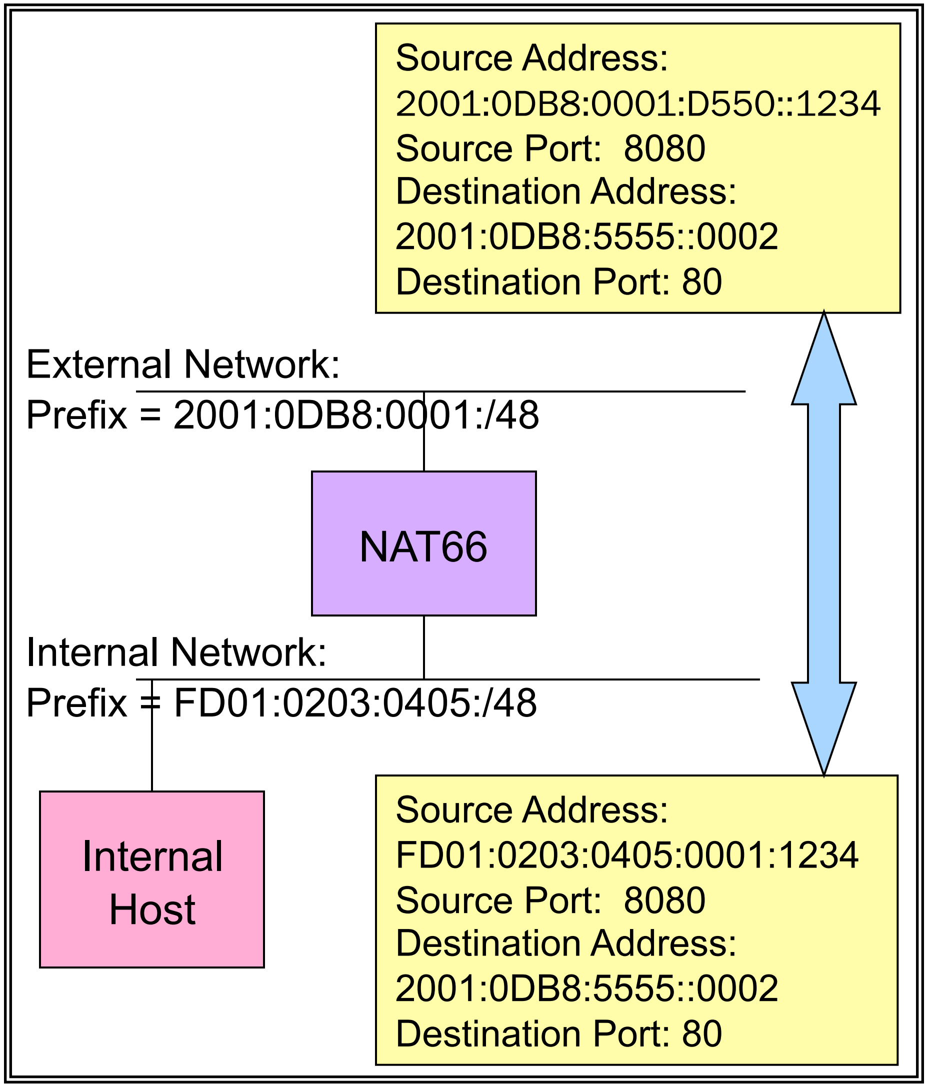
Topology Hiding

- Topology hiding is a poorly-defined and poorly-understood concept in the IETF
  - Before we could define a solution for topology hiding, we'd have to define the problem
- Topology hiding is also out-of-scope for this BOF

So, what is NAT66?

- A stateless, transport-neutral IPv6-to-IPv6 Network Address Translation (NAT66) function that provides the address independence benefit associated with IPv4 NAT while minimizing, but not completely eliminating, the problems associated with IPv4 NAT

Simple NAT66 Example



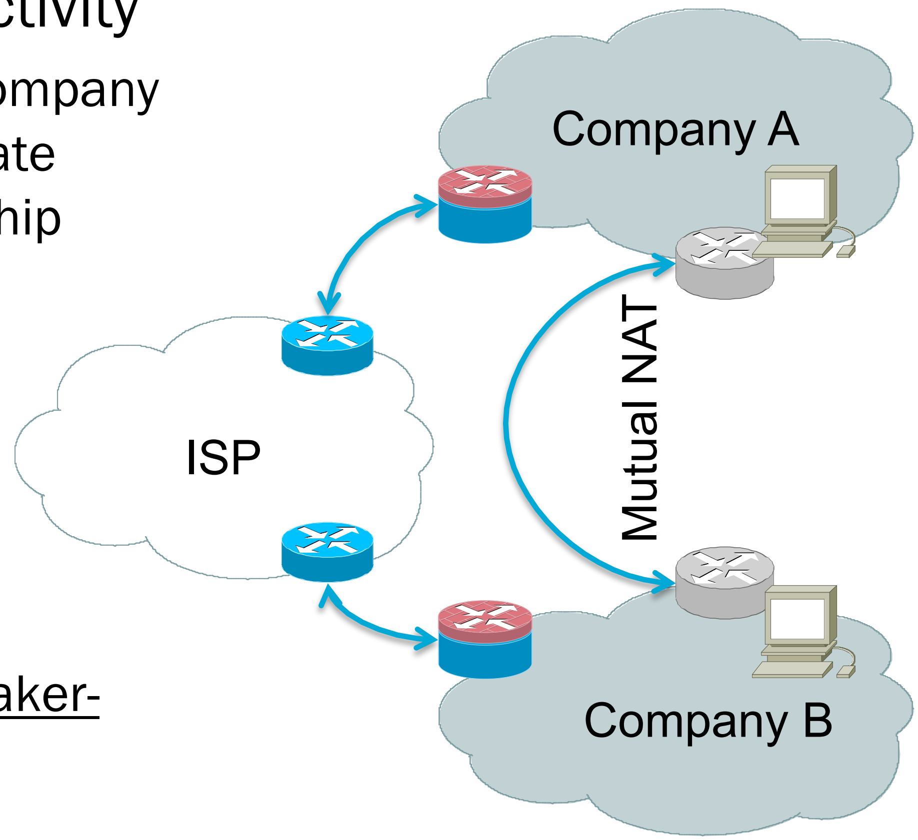
- Only the IP address prefixes are mapped
  - Source prefix on outbound traffic
  - Destination prefix on inbound traffic
- No per-host/connection state on NAT66 device
  - Prefixes configured
- Port numbers and transport checksum are not changed

NAT66 Scenarios

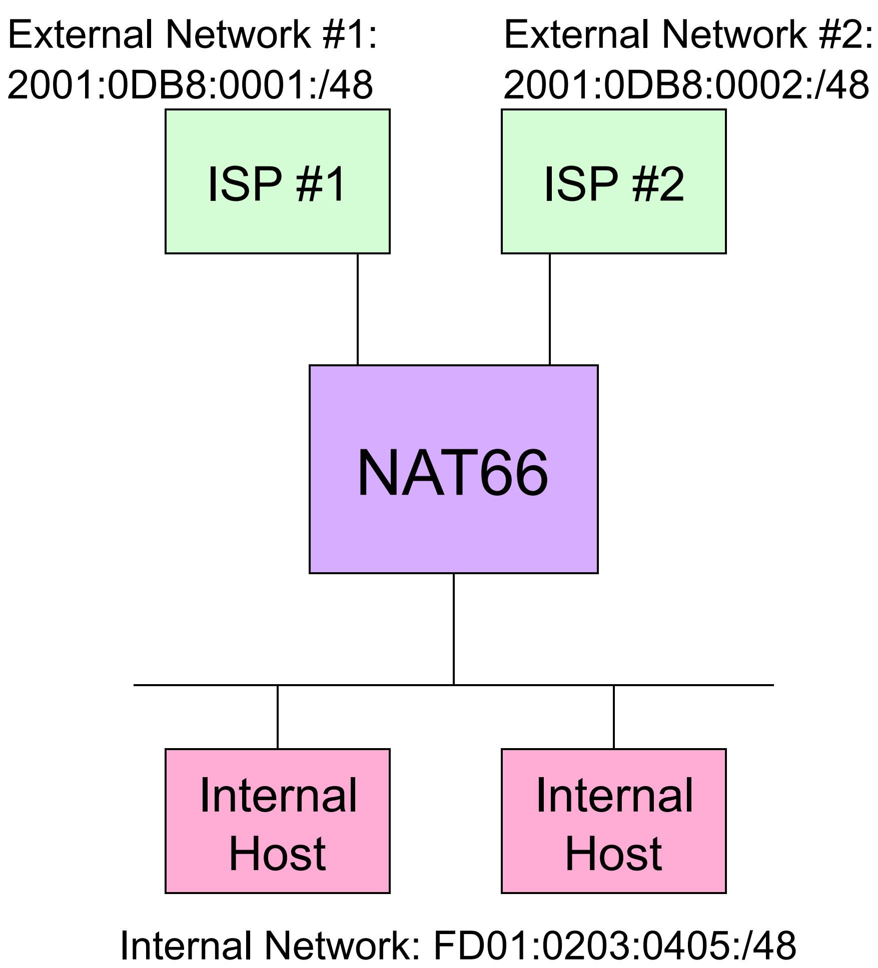
- The draft describes 3 scenarios for NAT66 deployment
  - Leaf network connected to the Internet via a single NAT66 device
  - More than one NAT66 device attached to a single network
    - Algorithmic mapping removes necessity for state sharing
  - NAT66 device between two private networks

Business-to-Business VPN

- Business-to-business connectivity
  - Company A uses services of company B under contract and has private security/connectivity relationship
- Issues:
  - Connectivity management
  - Mutual exposure – limiting information revealed
- Problem discussed in
  - http://tools.ietf.org/id/draft-baker-v6ops-b2b-private-routing



Simple Multihoming



- NAT66 allows for a simple multihoming solution
- Internal nodes use a single address prefix
- NAT66 translates into appropriate outbound prefix
  - One preferred, one fallback interface
  - Per-flow load balancing
- Two (external) addresses in global DNS for each node

Two-Way Algorithmic Mapping

- On outbound packets:
  - The source address prefix is overwritten with the external prefix
  - Checksum correction is performed as follows:
    - Calculate checksum of the old prefix (cP)
    - Calculate checksum of the new prefix(cP')
    - Take the ones complement difference (cP' + ~cP)
    - The difference is subtracted (using ones complement addition) to 16 non-prefix bits in the address
      - Bytes 49-64 if the prefixes are /48 or shorter
      - Bytes 113-128 if the prefixes are /49 or longer

Two-Way Mapping Example

Internal Prefix:	FD01:0203:0405:/48	Configured on NAT66 Device
External Prefix:	2001:0DB8:0001:/48	

Outbound Example:

ORIGINAL SOURCE ADDRESS: FD01:0203:0405:0001::1234

cP = 0xFCF5

External prefix is copied into the address, cP’ = 0xD245

~cP’ = ~0xD245 = 0x2DBA

Diff = cP + ~cP’ = 0xFCF5 + 0x2DBA = 0x2AB0

~Diff = ~0x2AB0 = 0xD54F

Bits 49 - 64 => 0x0001 + 0xD54F = 0xD550

0x0000 != 0xFFFF, so not changed to 0x0000

MAPPED ADDRESS = 2001:0DB8:0001:D550::1234

Two-Way Mapping Example (Cont.)

Internal Prefix: FD01:0203:0405:/48
External Prefix: 2001:0DB8:0001:/48
} Configured on NAT66 Device

Inbound Example:

ORIGINAL DESTINATION ADDRESS: 2001:0DB8:0001:D550::1234

cP = 0xD245

External prefix is copied into the address, cP' = 0xFCF5

~cP' = ~0xD245 = 0x030A

Diff = cP + ~cP' = 0xD245 + 0x030A = 0xD54F

~Diff = ~0xD54F = 0x2AB0

Bits 49 - 64 => 0xD550 + 0x2AB0 = 0x0001

0x0001 != 0xFFFF, so not changed to 0x0000

MAPPED ADDRESS = FD01:0203:0405:0001::1234

IPv4 NA(P)T vs. NAT66

- There are substantial differences between IPv4 port-mapping NATs and NAT66
- The following slides outline the elements of a typical IPv4 NA(P)T
  - Each element has associated advantages and disadvantages
  - Red text marks things that are different in NAT66
  - ✓ Checks mark things that are the same in NAT66

Decomposition of an IPv4 NAT

- Address mapping
  - √ Maps between internal/local and external/global realms
  - – Entire address is replaced (prefix & host portion)
  - – Mapping is many:1
    - multiple internal hosts share an external address
- Advantage(s):
  - √ Address Independence
  - – Superficially hides number and organization of internal hosts
    - comes from many:1 many to one
- Disadvantage(s):
  - – Internal nodes cannot be addressed from external nodes
    - Because they are not identified by separate addresses
  - √ Inconsistent with security that encrypts/protects IP headers
  - √ Loss of end-to-end address transparency

Decomposition of an IPv4 NAT (2)

- Port mapping
  - Maps local port number to an available external port
  - Required due to many:1 mapping
    - Original local port may be in use
- Advantage(s):
  - Obscures original port selected by the host
    - Makes it slightly harder to infer number/organization of internal hosts
  - Provides opportunity to introduce port randomization if the host does not
- Disadvantage(s):
  - Requires modification of transport layer header
    - Inconsistent with security that encrypts/protects transport headers
    - Complicates or blocks innovation at the transport layer

Decomposition of an IPv4 NAT (3)

- Maintenance of mapping state
  - Maintains dynamic address/port mappings for active flows
  - Required due to many:1 address mapping
- Advantage(s): None
- Disadvantage(s):
  - Introduces single point of failure
    - Connections are lost if the NAT device goes down/loses state
  - Undermines dynamic routing
    - Connections are lost if they are no longer routed through the same NAT device
  - Requires keep-alive packets to maintain NAT state for idle connections
    - Reduces battery life of mobile nodes
    - Increases overhead traffic in the network

Decomposition of an IPv4 NAT (4)

- Checksum modification
  - Updates IPv4 header checksum
  - Updates checksum in UDP/TCP headers
    - Required due to IP pseudo-header checksum
- Advantages: None
- Disadvantages:
  - Incompatible with security that encrypts/protects transport layer headers
  - Complicates/blocks innovation at the transport layer

Decomposition of an IPv4 NAT (5)

- Application-layer IP address and port mapping
  - √ AKA Application Layer Gateway (ALGs)
  - √ Maps between internal and external IP addresses and ports that appear in application-layer headers
  - – Even if FQDNs are used instead of IP Addresses, still may need to map between internal and external ports

√ Advantage(s): None

√ Disadvantage(s):

- – Incompatible with security mechanisms that encrypt, or provide integrity checking for, the application layer headers/payload
- – Requires application-specific code in the NAT device
  - Complicates/blocks innovation at the application layer
  - Partially mitigated by use of NAT traversal tools (STUN in IPv4, something lighter in IPv6) in new application layer protocols

Side-by-side Comparison

Typical IPv4 NAT

- Address mapping
  - Many:1, one-way, stateful
- Port mapping
  - Maps local port number to an available local port
- Mapping state maintenance
  - Maintains dynamic address/port mappings for active flows
- IPv4 & TCP/UDP Checksum modification
- Application-layer IP address and port mapping (ALGs)
  - Needed for IP addresses and ports in some application-layer headers

NAT66

- Address mapping
  - 1:1, reversible, stateless
  - Includes UDP/TCP checksum correction
- No port mapping
- No state maintenance
- No transport checksum modification
- Application-layer IP address mapping (ALGs)
  - Still needed for IP addresses in some application layer headers

Why publish NAT66?

- A few facts..
  - There is demand from enterprise network operators for IPv6 NAT
  - Vendors are implementing IPv6 NAT products to meet that demand
  - There will be IPv6 NAT, and the IETF cannot do anything to prevent it
- Therefore, we have two choices…
  - Refuse to document IPv6 NAT
    - Some vendors will simply build IPv4 NA(P)Ts with longer addresses
    - Others will try to make improvements, causing inconsistency
  - Document an IPv6 NAT mechanism (such as NAT66)
    - Share our understanding of how to build a less problematic IPv6 NAT
    - Minimize negative impacts of IPv6 NAT
    - Promote consistency in how IPv6 NATs will work