SIPNAT (source_IP NAT)

November 13, 2009
IETF 76 [behave] WG
charliep@wichorus.com

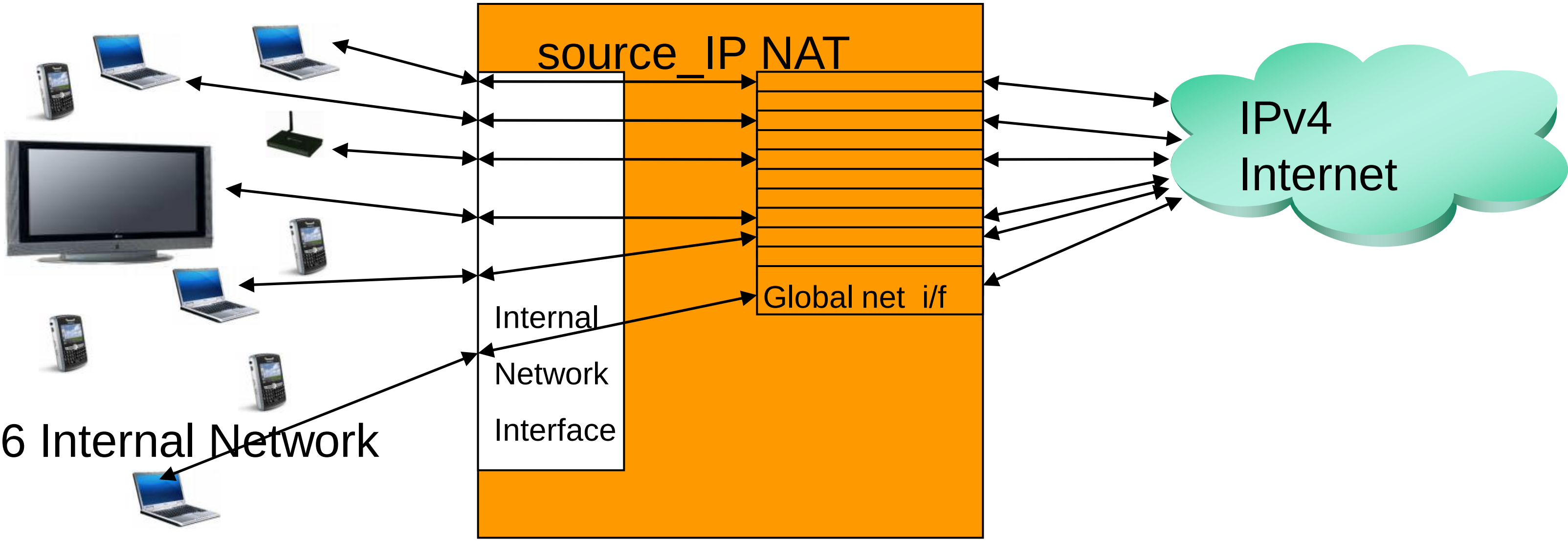
WICHORUS

Business pitfalls of moving to IPv6 today

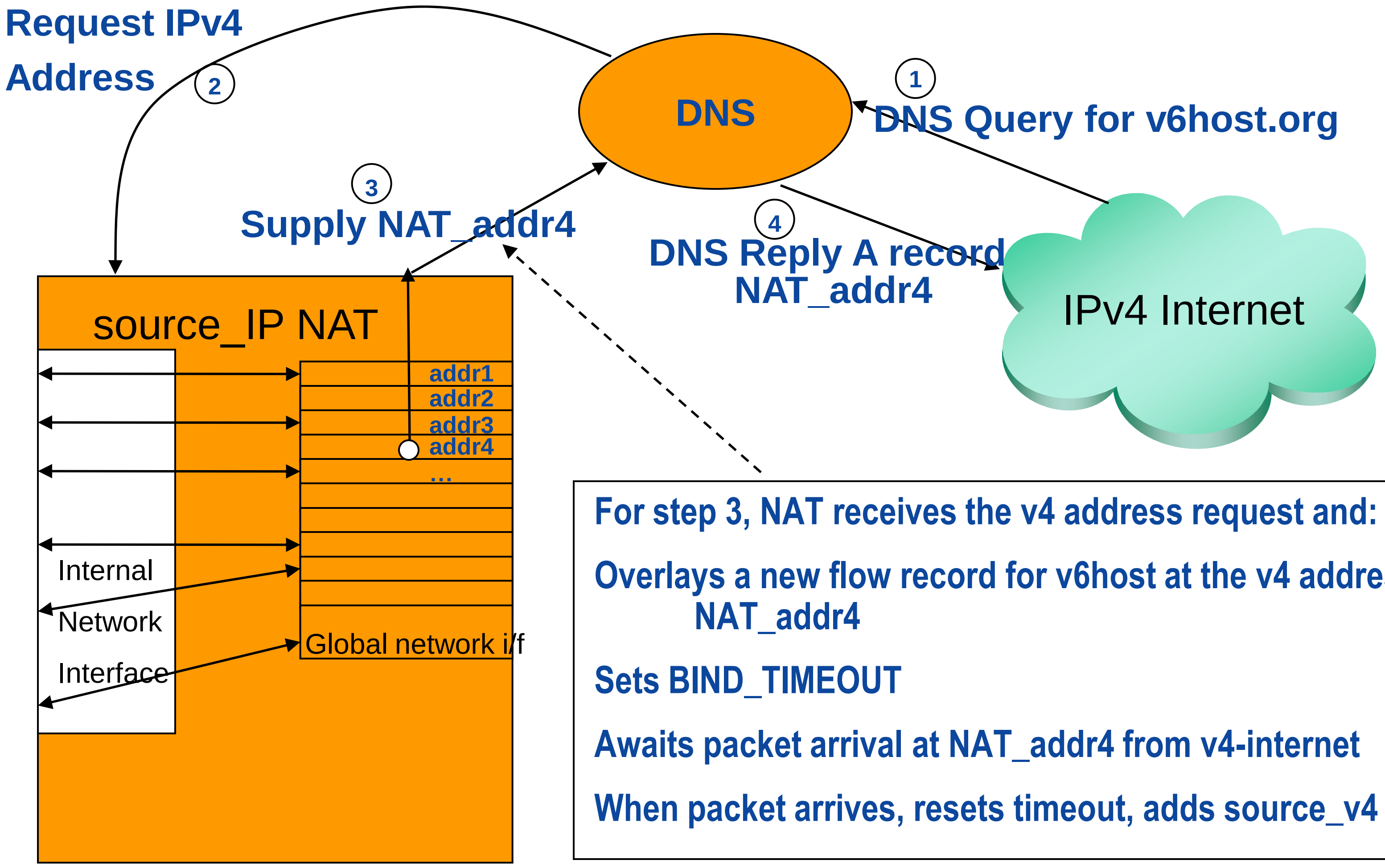
- **Practically all of the customers are using IPv4**
- **So, business must serve IPv4 web accesses**
- **Web presence is required 24 x 7 x 52 x …**
- **This is not compatible with today's NAT solutions, or today's IPv6 solutions**
  - ➢ **Customers need to be able to contact business**
  - ➢ **Not the other way around!**
- **Needed: "always on" NAT for v4→v6 translation**
  - ➢ NOTE: NAT is needed ***for sure*** ["evil" or not]

Bidirectional NAT v4 ←→ v6 (uses DNS)

- No changes to IPv6-only hosts or IPv4-only hosts
- No dual-stack
- No tunneling
- Can delegate special domain to NAT box if desired
- Modeled as a flow-management problem



Operation of system…



For step 3, NAT receives the v4 address request and:

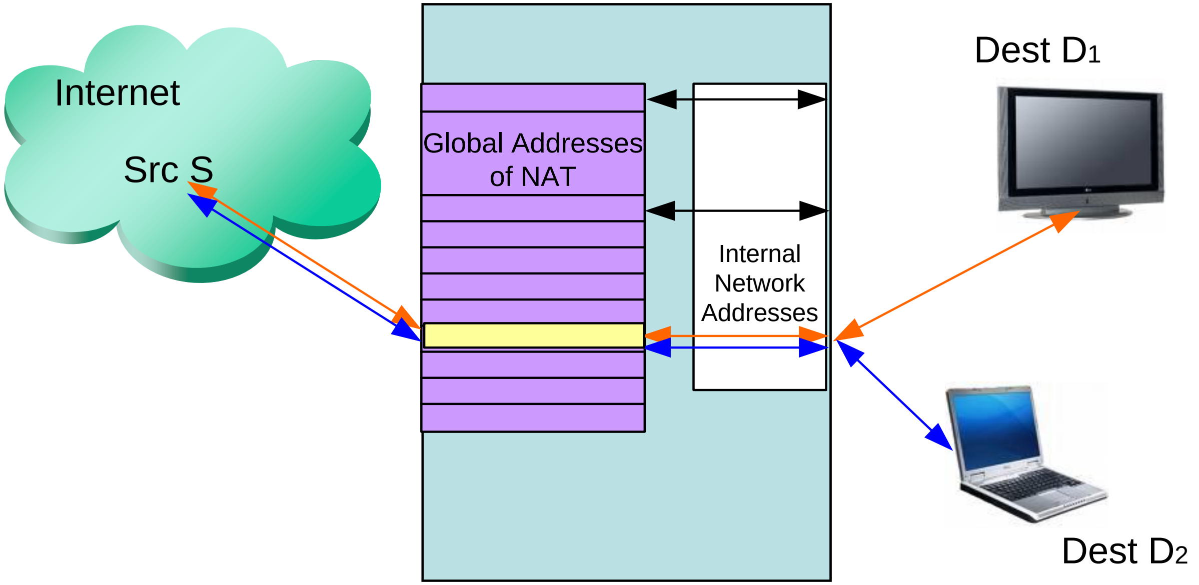
Overlays a new flow record for v6host at the v4 address NAT_addr4

Sets BIND_TIMEOUT

Awaits packet arrival at NAT_addr4 from v4-internet

When packet arrives, resets timeout, adds source_v4

Unassisted mode failure: one source ←→ two dests



- **The system will fail if a specific source tries to access too many destinations**
  - **At each IPv4 address of the NAT, a source IP address (and, possibly, source port) _identifies_ the flow**
  - **Can have one flow per source per NATv4 address, if lucky**

Unassisted mode: failure scenario B

The system will fail if there are too many new flow requests at about the same time

- Have to keep the request “pending” until a packet arrives to provide the exact source IP address
- Thus, each flow request temporarily (WAIT_TIME) consumes a NATv4 interface address
- Since the DNS Request does not have the source IP address, the allocated flow will go to the source of the first packet to arrive that is not already deliverable
- May need also to keep “pending” address open just a little longer to mitigate DoS

Is it really like flow management?

- Incoming <v4dev, sport, NATaddr, dport, TOS> → <v4mapped, sport, v6dev, dport, TOS>
- Use DPI to figure out which ALG to use
- Gradually move more functions to hardware?
  - Checksums
  - Pattern recognition
- Have to search overlapping flow records per v4addr
  - Determine maximum degree of overlap?
  - This is what provides scalability for the solution

Payload assist for higher scalability / robustness

- Base v4→v6 NAT system works well
- Can improve scalability and robustness using known payload fields (for certain protocols)
- Good example: http GET contains “http.host” field, identifying the destination
- Also: works for SIP (e.g., VoIP, presence, instant messaging, …)
- Additional techniques to enable peer-to-peer

Pattern Matching techniques

- A large majority of website pathnames are unique to specific destinations
- For HTTP: pattern matching machine could identify the correct destination *based only* on payload
- Can _assure_ delivery for aware customers
  - ➢ For example: http://www.wichorus.com/wichoruspages/...
- Similar techniques work for other protocols

Recent analytical results

- “Queueing Theoretic Analysis of Source IP NAT” with Cedric Westphal submitted to ICC 2010
- Wait time W → 0 as # of translation interfaces grows
- Analytical expressions for W are derived under various conditions
  - Single interface
  - Multiple interfaces, random assignment
  - Multiple interfaces, round-robin assignment

Conclusions

- Offering content and services on IPv6 requires access *from* IPv4
- SIPNAT enables scalable, bidirectional, transparent communication between IPv4 ←→ IPv6 Internets
    - ➢ No tunneling, no host upgrades, no dual-stack
    - ➢ Can run at line speed using flow management
- Basic system offers high reliability
- Using additional DPI-related techniques, SIPNAT can provide 100% packet delivery accuracy