

Detecting CGN in the ISP

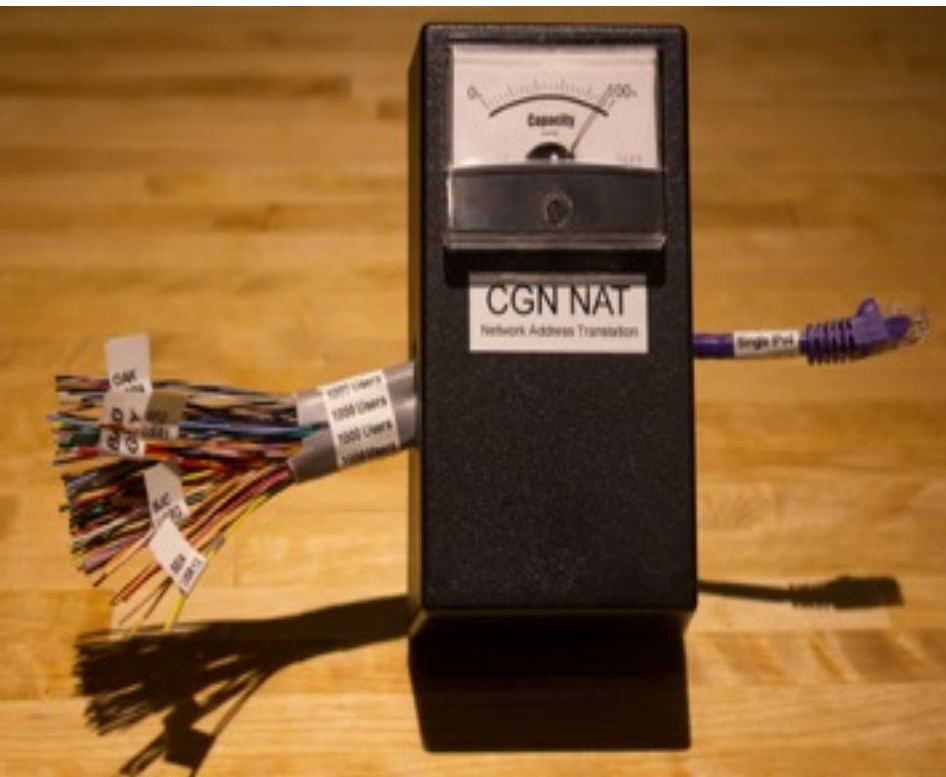
Andra Lutu, Marcelo Bagnulo,
Amogh Dhamdhere, kc claffy



Network Address Translation (NAT)

- We are out of IPv4 address space
- IPv6 adoption is slow, though accelerating in recent times
- Network Address Translation prolongs the life of IPv4 by enabling address sharing
- NATs can be performance bottlenecks, break certain applications, or inhibit IPv6 adoption in the near term

NAT444 / Carrier Grade NAT/ Large Scale NAT



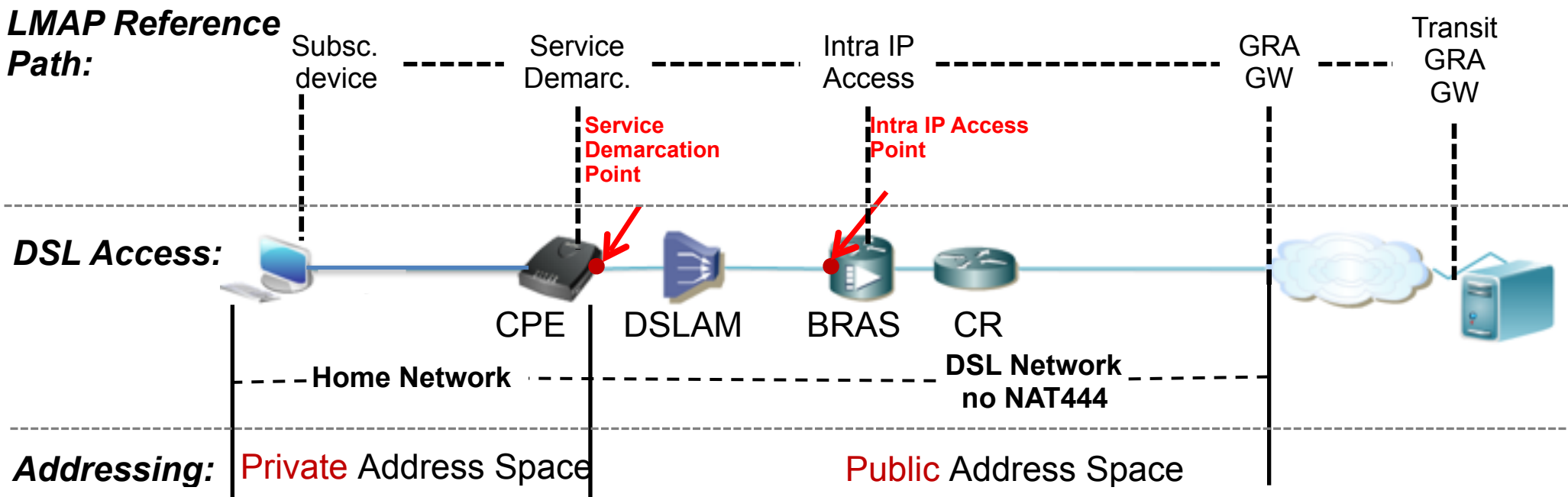
RFC7021: Assessing the Impact of Carrier-Grade NAT on Network Applications

- On-line gaming
- Video streaming
- BitTorrent
- VPN & Encryption
- VoIP
- ...



Traditional NAT (NAT44)

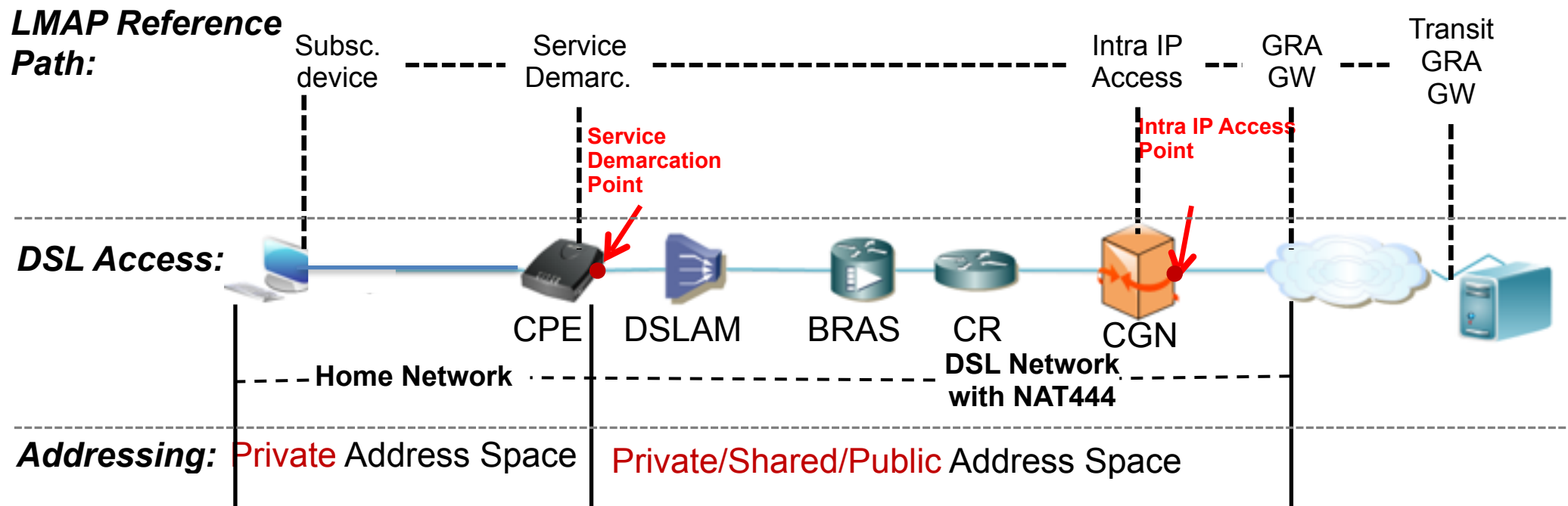
DSL Access Network mapped to the LMAP Reference Path



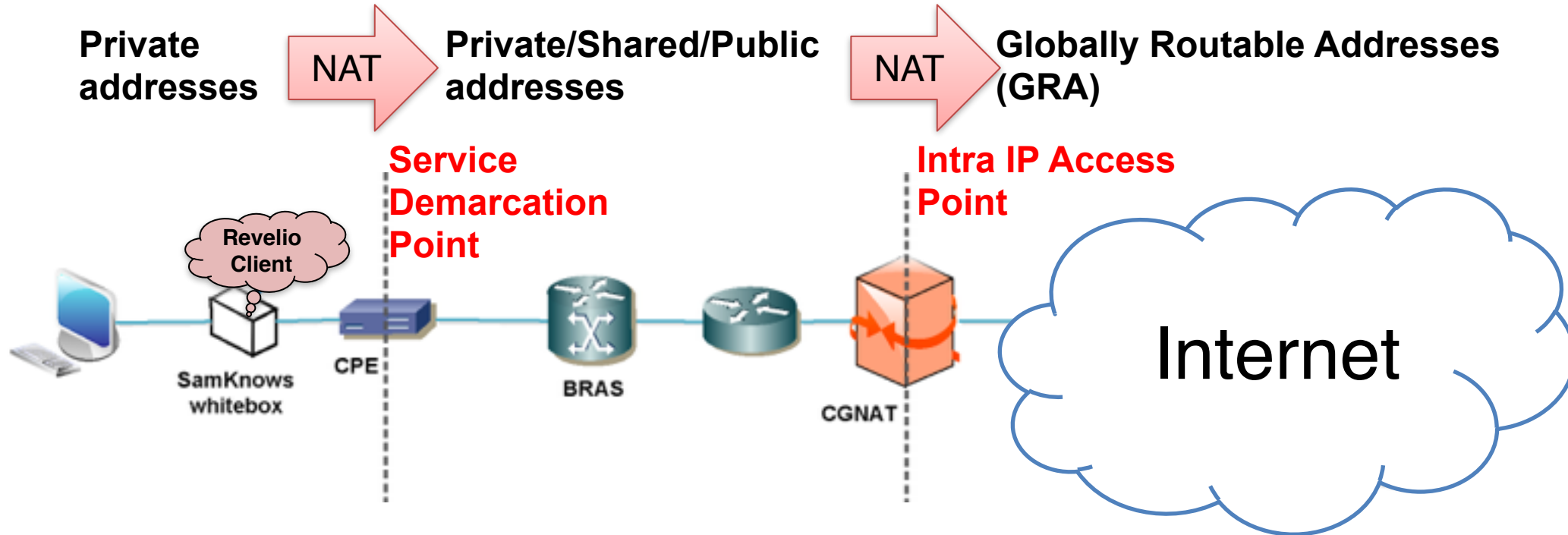


Large Scale NAT (NAT444)

DSL Access Network *with NAT444 deployment*



NAT Revelio



- Detect the usage of private/shared address space beyond the CPE, in the ISP access network
- Detect the location (home network or ISP access network) of the device doing the translation to the GRA

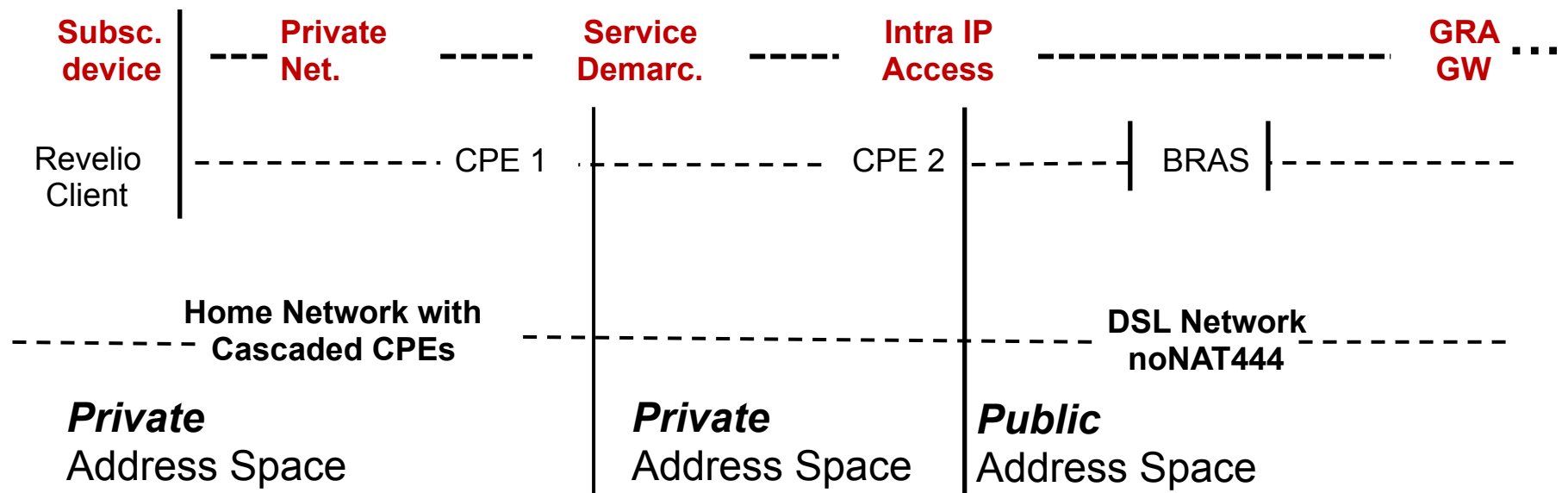
Client-side detection

- Two approaches to CGN detection: using measurements from the client or from “outside”
- NAT Revelio is a client-side approach
- Specific use scenario: from the user CPE (e.g., SamKnows or Bismark router)
- Pro: more control over measurements
- Con: coverage limited to networks with VPs

NAT Revelio: Design Challenges

- Diverse home network configurations, e.g. in-home cascaded NAT, with probe **NOT** connected directly to the CPE, misconfiguration in setting up SamKnows box
- Diverse ISP configurations and deployments, e.g. use of private IP addresses internally even if they don't do NAT444

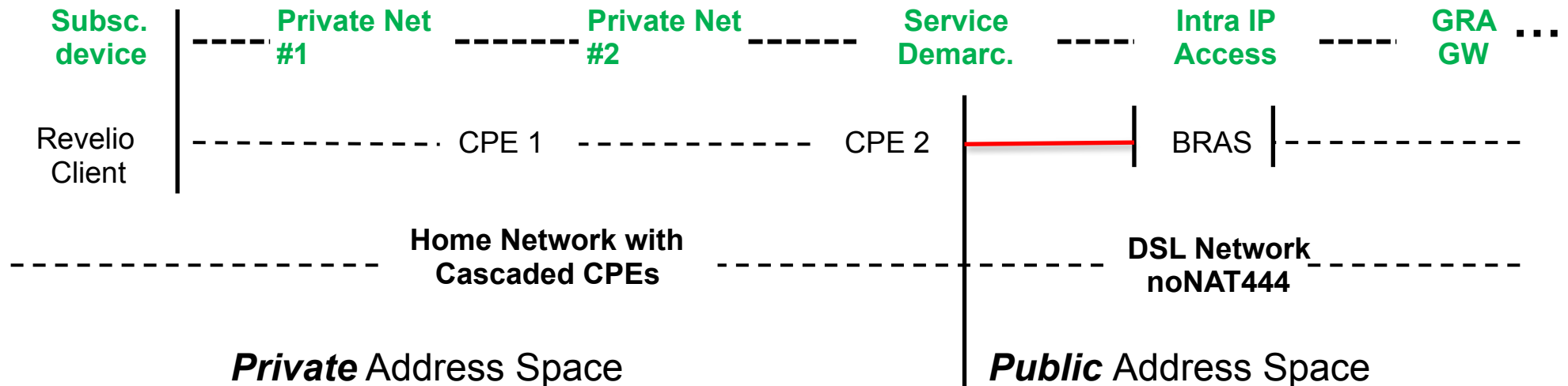
Incorrect Mapping with the LMAP Reference Path:



NAT Revelio: Design Challenges

- Need to detect the access link to delimit the access network and the home network
- Eliminates some false positives

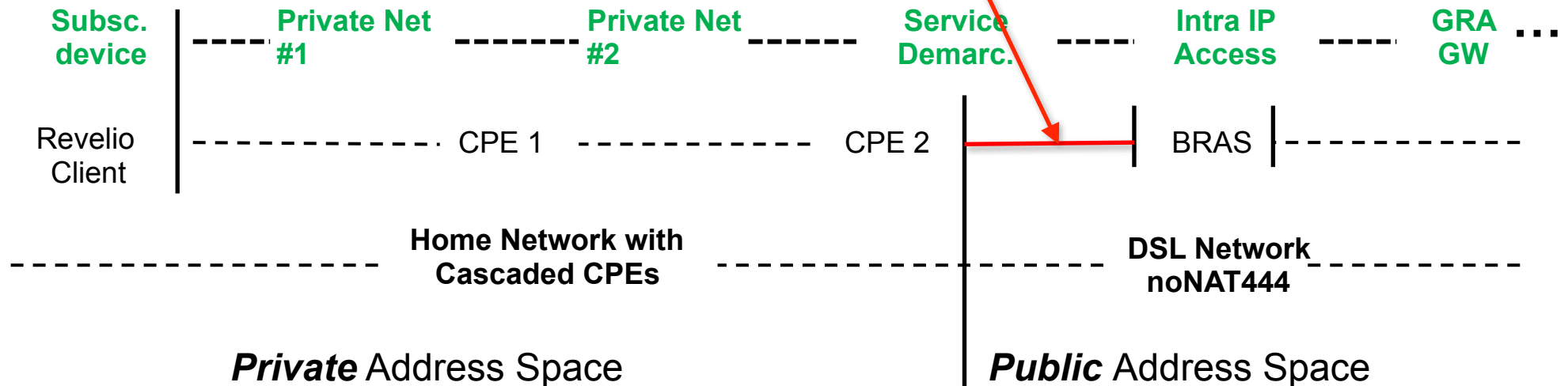
Correct Mapping with the LMAP Reference Path



NAT Revelio: Design Challenges

- Need to detect the access link to delimit the access network and the home network
- Eliminates some false positives

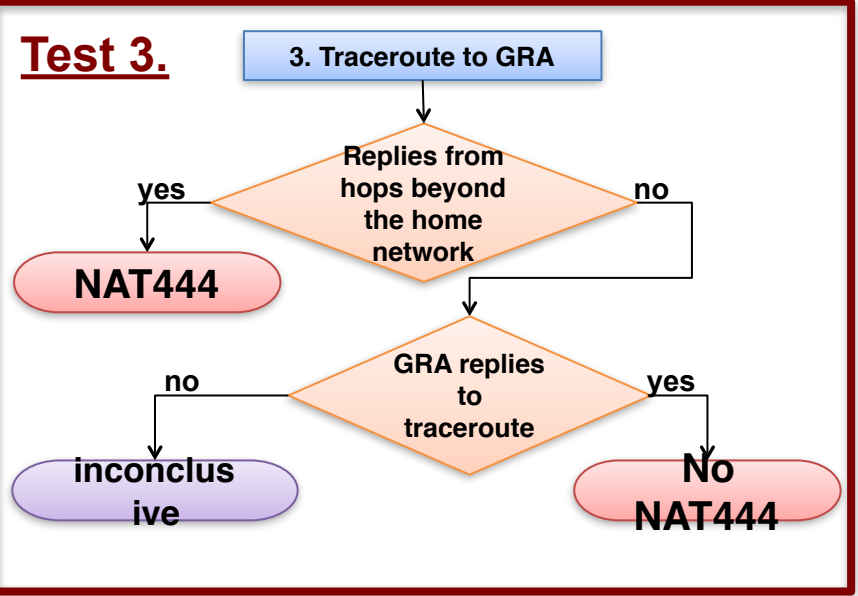
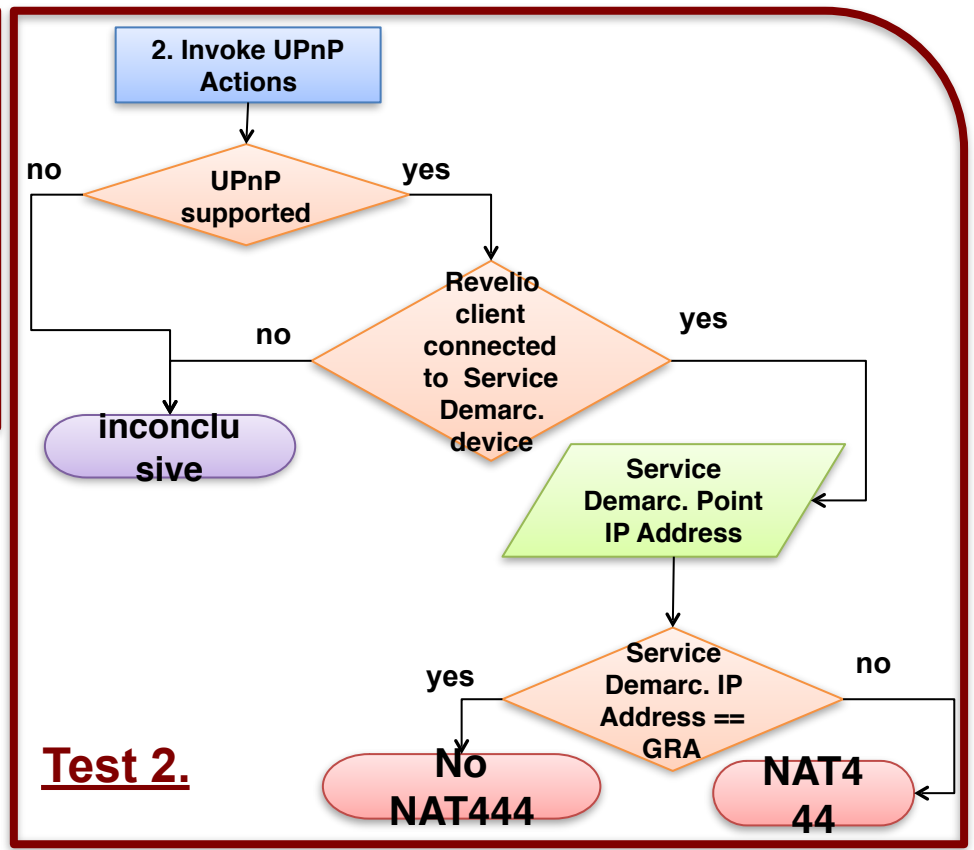
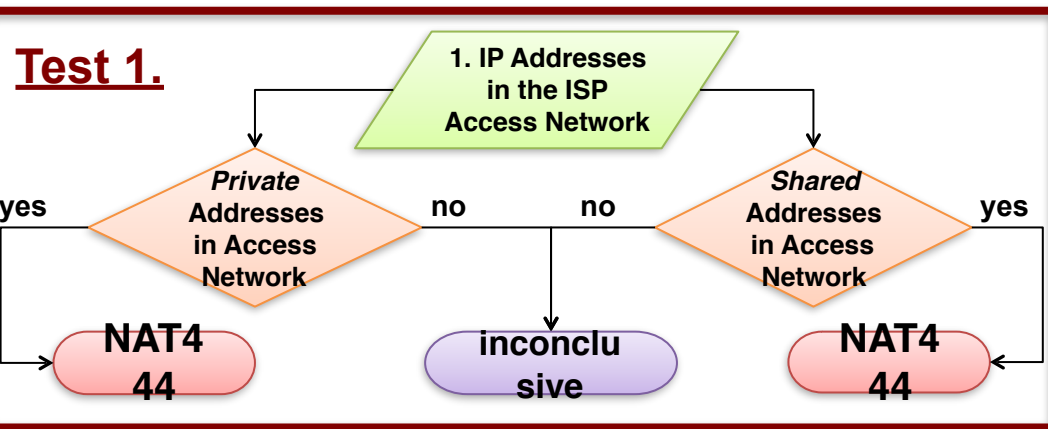
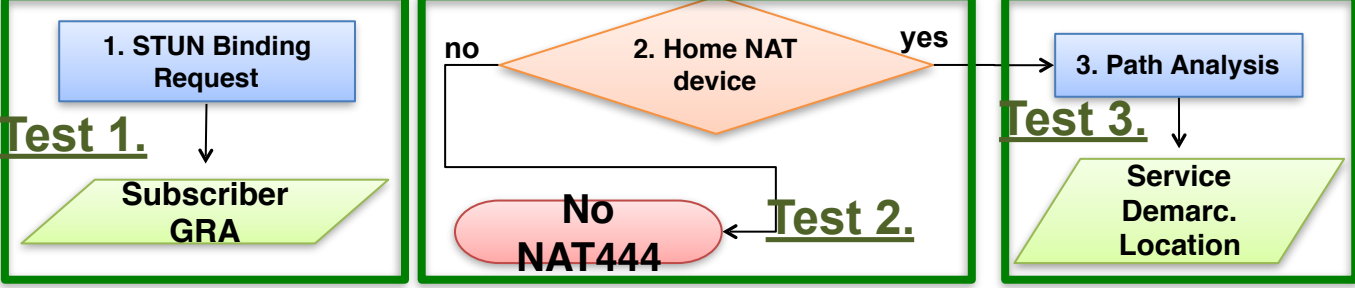
Correct Mapping with the LMAP Reference Path

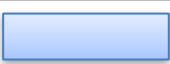





NAT Revelio

- The NAT Revelio test suite includes 2 phases
- Environmental Characterization
 - Understand the environment hosting the device running the Revelio Client
- NAT444 Discovery
 - Detection of signals that the ISP might deploy a NAT444 solution in the ISP access network

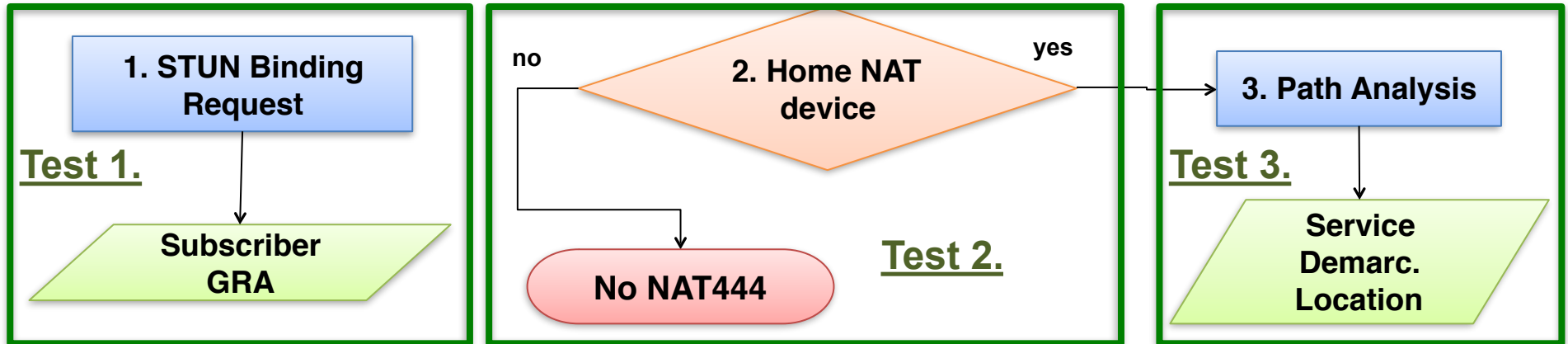
Phase 1) Environment Characterization



-  **Action** performed (e.g., send STUN request to retrieve the subscriber GRA)
-  **Data** retrieved (e.g., the subscriber GRA)
-  **Test** performed (e.g., is the GRA configured on the Service Demarcation point)
-  **Conclusion** stop block (i.e., NAT444 in the ISP, no NAT444 in the ISP or inconclusive)

Phase 2) NAT444 Discovery

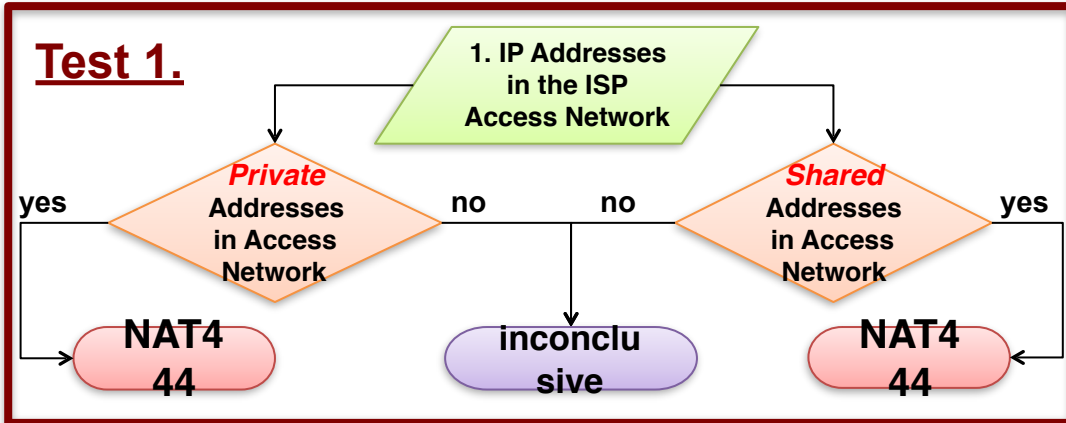
Environment Characterization



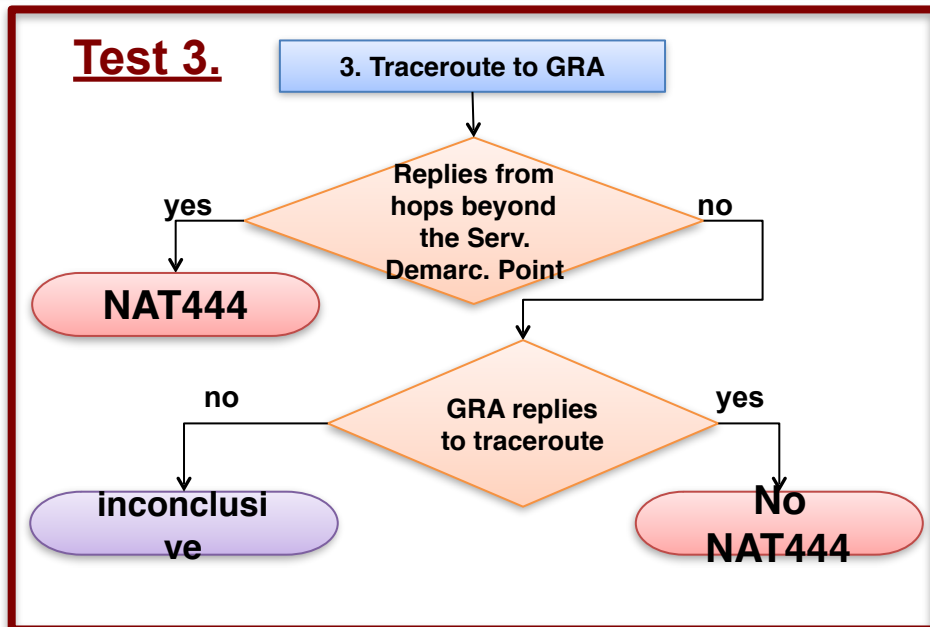
- Test 1: The GRA of the subscriber running the Revelio client
- Test 2: Whether the subscriber is behind at least one level of NAT (i.e., the CPE performs the NAT function)
- Test 3: Position of the Revelio client related to the Service Demarc. Device (i.e., the position of the access link relative to the Revelio client)

NAT444 Discovery

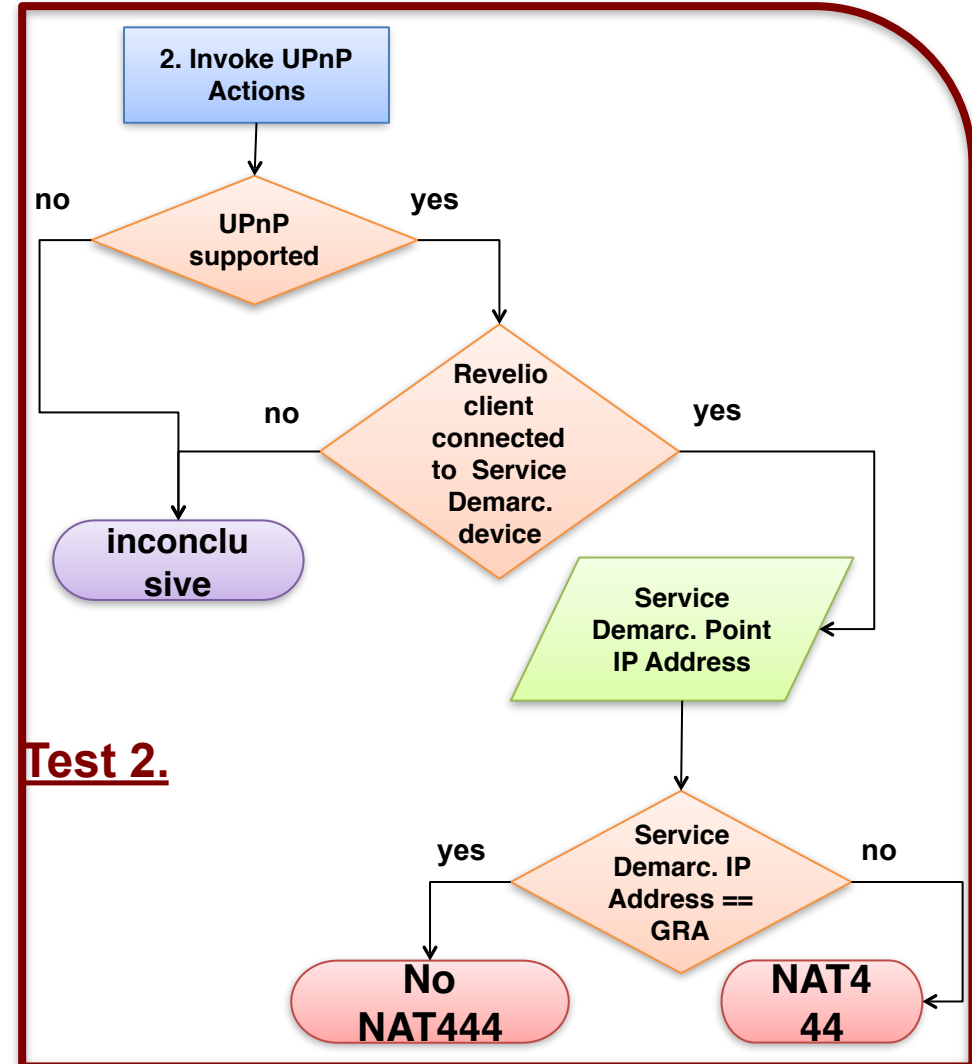
Test 1.



Test 3.



Test 2.



Experimental Results

- *NAT Revelio deployment on a large scale*
- 1,954 SamKnows Whiteboxes in 26 ISPs across the UK
- We found that 10 end-users are connected behind a NAT444 deployment
 - 5 different ISPs
- Repeated test 6 months later, with consistent results

Current status

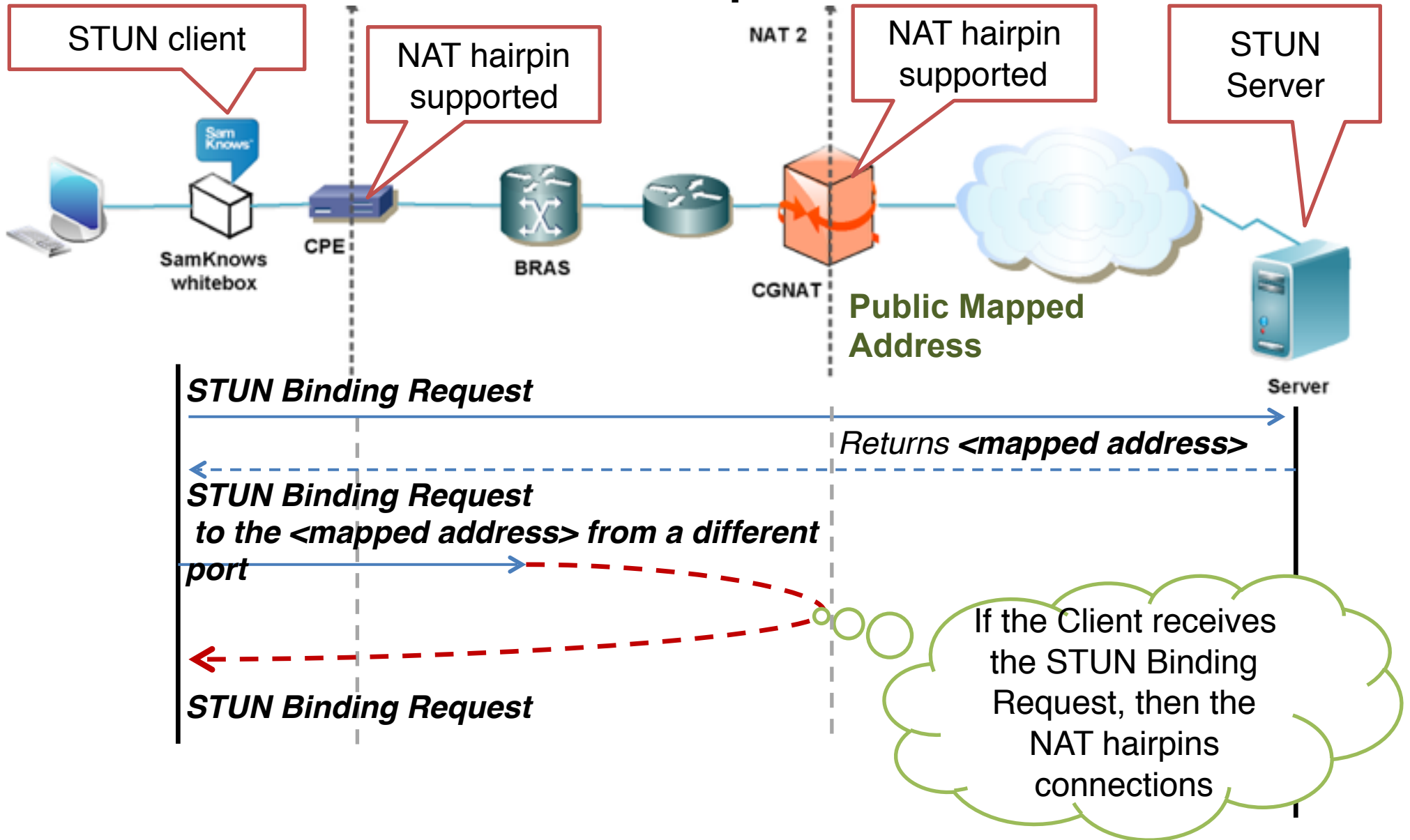
- Working with the FCC to deploy on the FCC/SamKnows infrastructure in the US
- Estimated deployment soon (ish). maybe



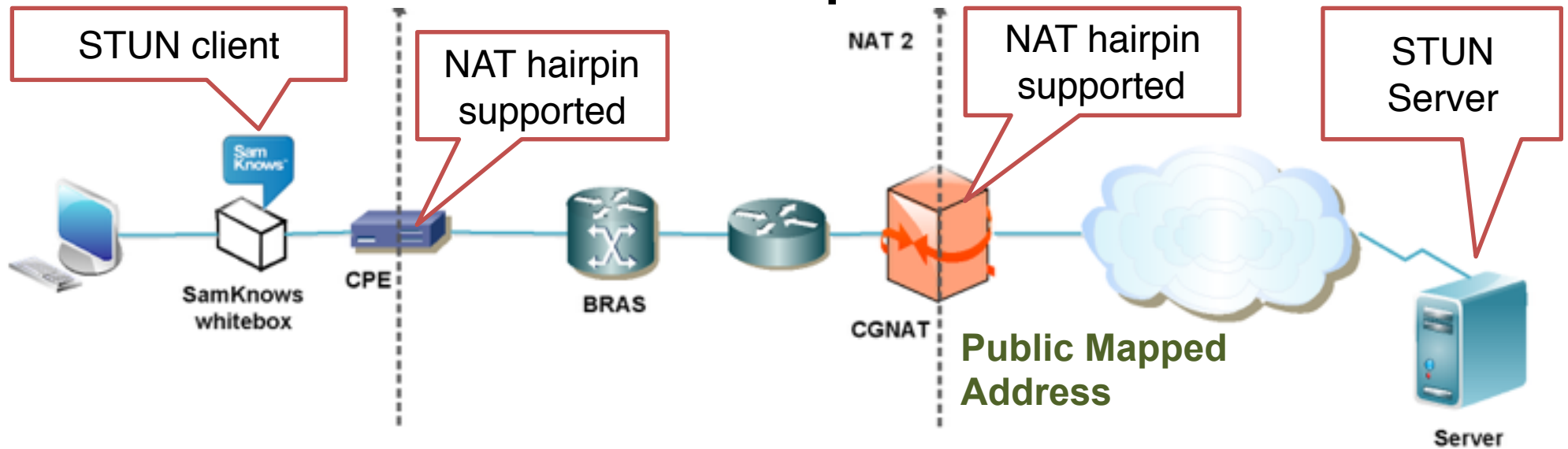
NAT Revelio

- Other tests
 - Hairpin test
 - Port preservation test
 - Multi-client test

NAT Hairpin Test

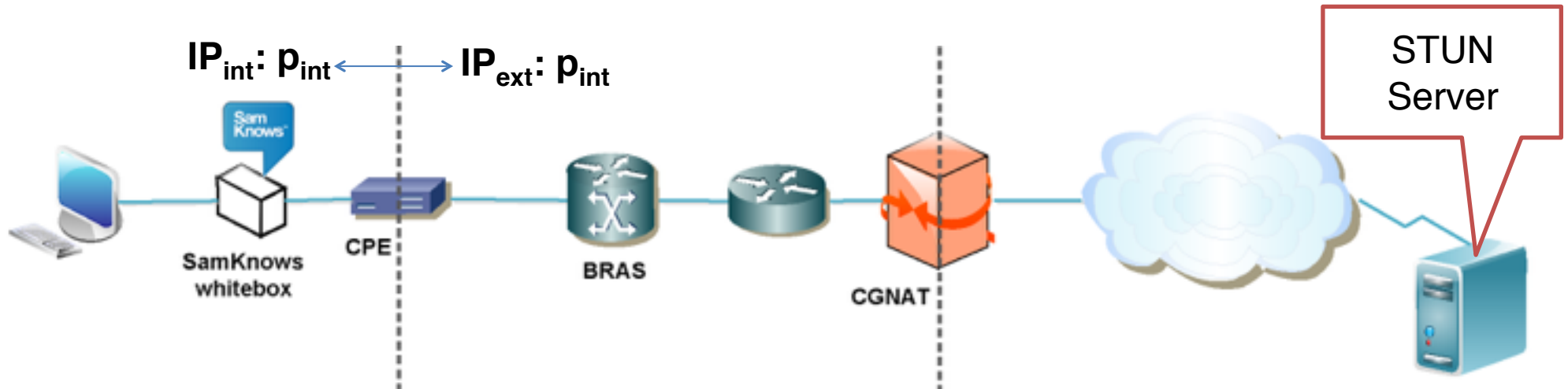


NAT Hairpin Test



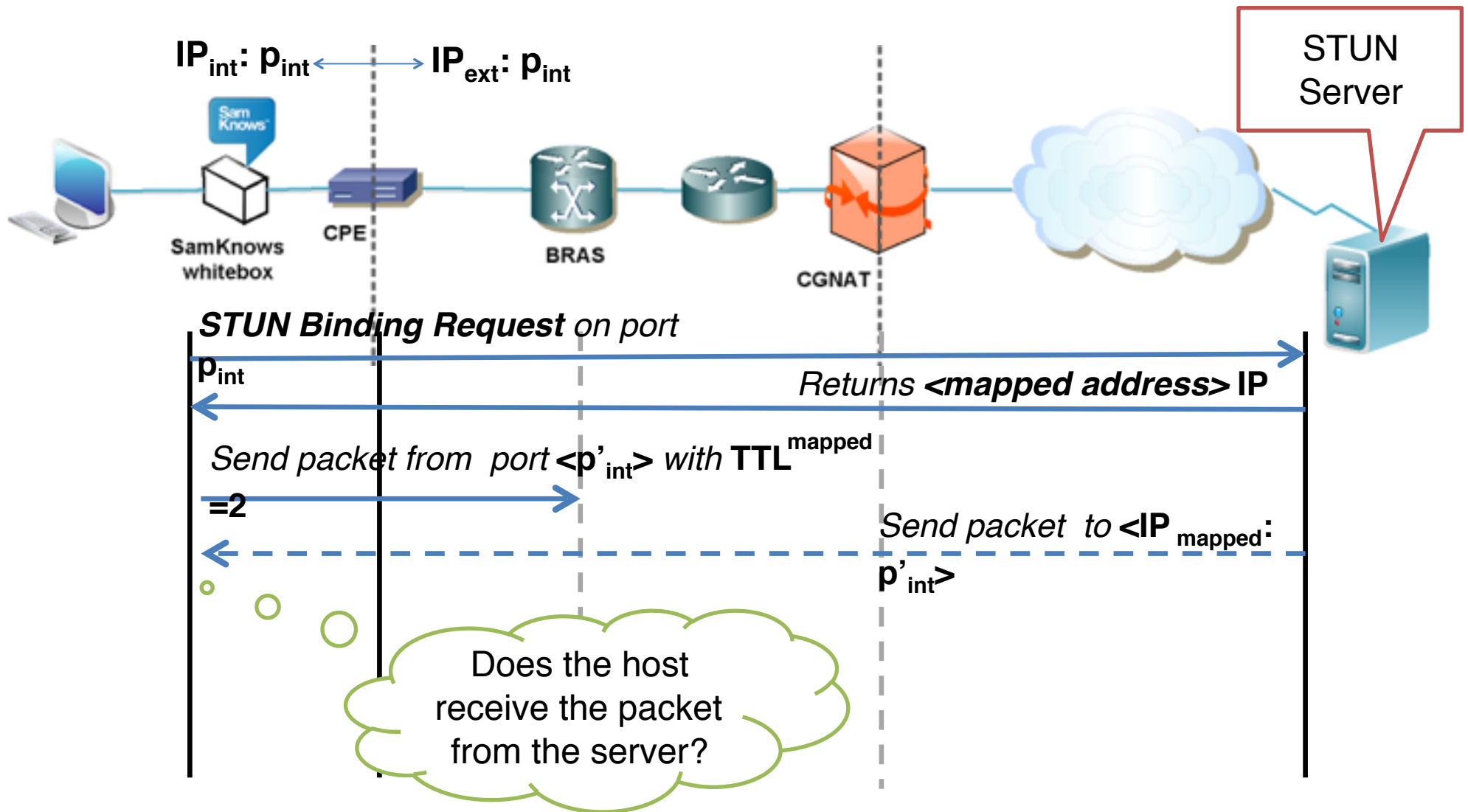
- If the NAT hairpins connections, the client verifies the received STUN Binding Request received to check the TTL value
 - E.g., if $TTL < 254$, the $\langle \text{mapped IP} \rangle$ is not the external IP of the CPE \Rightarrow CGN detected

Port Preservation test

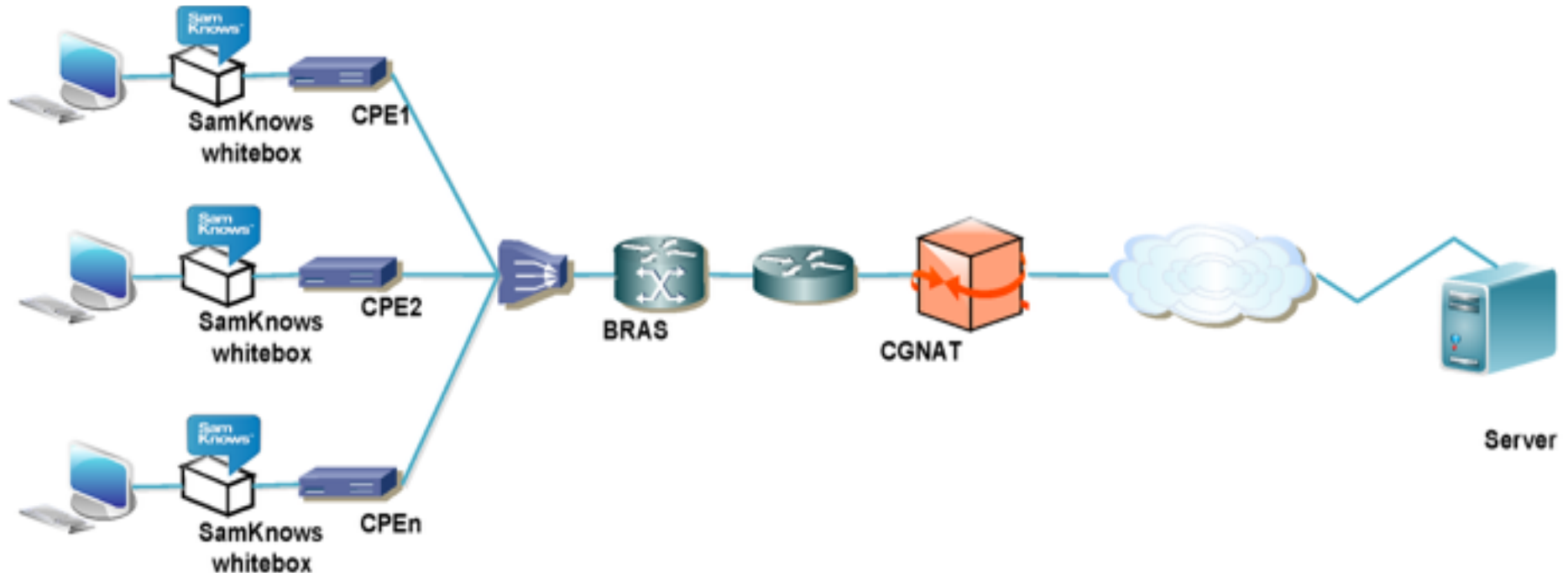


- Some NATs implement the port assignment behaviour known as **port preservation**
 - Attempt to preserve the port number used internally when assigning a mapping to an external IP address and port
- Send a Binding Request to the STUN Server from port p_{int}
- Learn the **<mapped address>**
- Create a new mapping for port p'_{int} in the CPE (send packet from port p'_{int} with **TTL = 2**)
- Send a packet from the MS to $IP_{mapped}: p'_{int}$
- If the host does not receive the packet => CGN detected

Port Preservation test

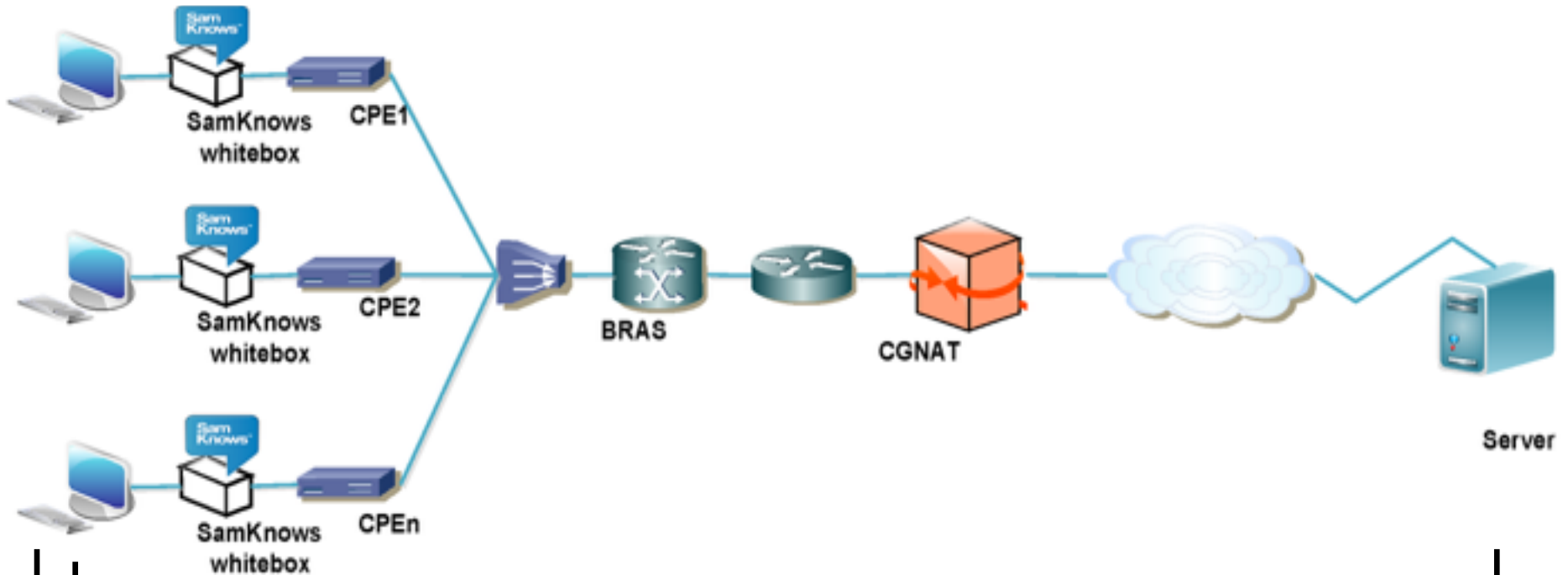


Multi-client test



- Retrieve the Mapped Public Address for each probe
- If any two probes have the same mapped public address => CGN detected
- Cannot detect all the clients that are behind the same CGN, but it can tell if the ISP is using a CGN

Multi-client test



Send STUN Binding Request

Send STUN Binding Request

Send STUN Binding Request

Which is the Source Address seen by the Server for each packet?

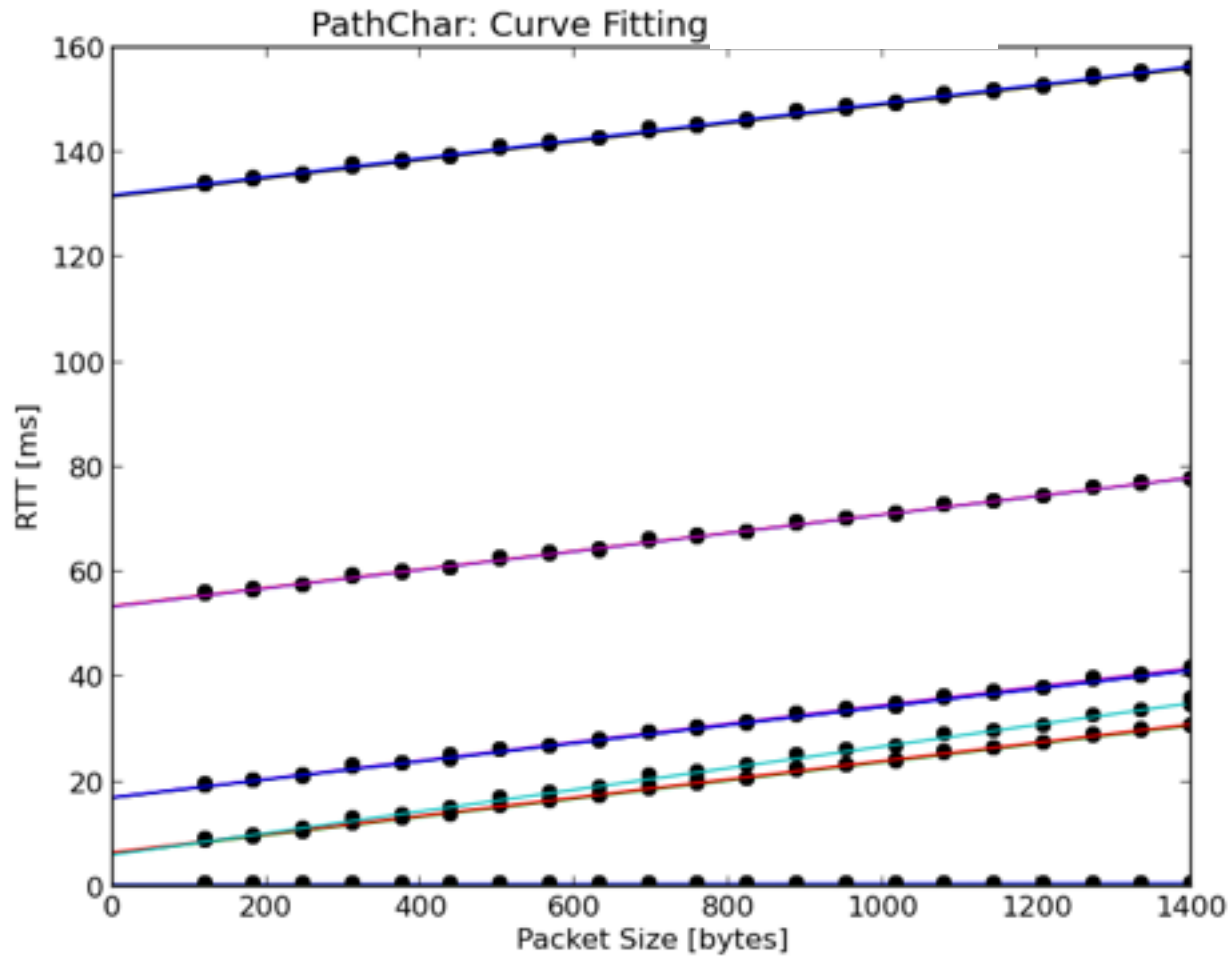
pathchar to detect the access link

- Run UDP traceroute to a fixed target (router inside Level3 network with no rate limiting)
 - Used the well-known traceroute port range
 - 21 different packet sizes (from 120 to 1400 bytes)
 - One traceroute probe per TTL, max TTL of 30
- Run every hour, over 4 days => collected **96 *RTT samples per TTL*** and for each packet size

pathchar to detect the access link

- For each TTL:
 - 1) Minimum Filtering:
 - For each packet size, choose the minimum value of the RTT
 - Capture only the transmission delay and the propagation delay
 - $RTT = \text{packet_size}/BW + LAT$
 - 2) Line fitting
 - Using the 21 different points, fit a regression line for the RTT and determine the **slope [1/BW]** and the **intercept [LAT]**

pathchar to detect the access link



pathchar to detect the access link

3) Differencing

- Given the estimated cumulative parameters above, *pathchar* determines the per-link parameters (slope and intercept, i.e., $1/BW$ and LAT) by subtracting the consecutive fitted lines parameters