

# FREQUENTLY ASKED QUESTION

## Operating phones behind NAT

[ FREQUENTLY ASKED QUESTION ]

Network Address Translation (NAT) is a reality in today's networks. Many countries save IP addresses by providing only one IP address for complete companies. Firewall manufacturers make NAT a feature by performing inspection of packets that go through NAT. Even for IPv6 networks, the fundamental problem will remain as there will also be a need for firewalls and private networks.

The Session Initiation Protocol (SIP) has neglected this problem in the beginning. However, in some recent RFC there have been useful proposals how to deal with the problem. This document shows how the snom 4S can be used to solve the problems and how the snom phones help making the problems as small as possible.

### NAT

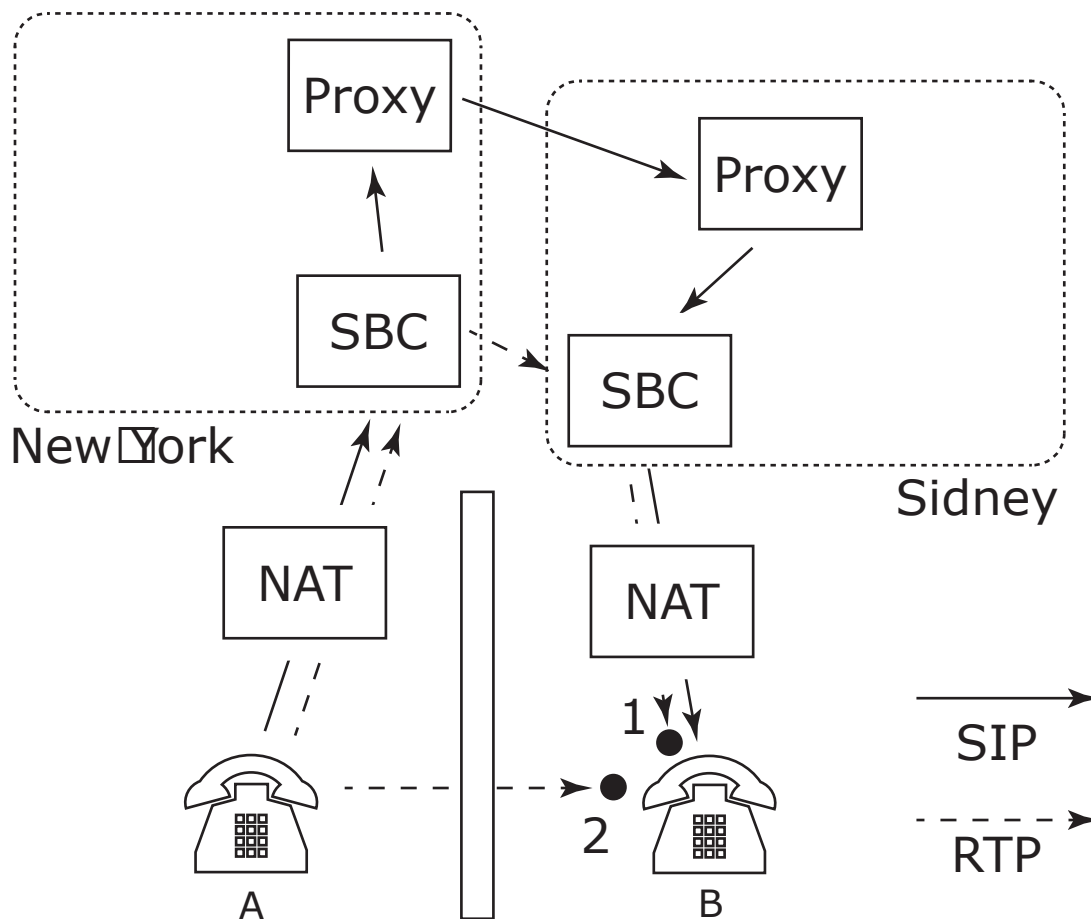
NAT is essentially a translation table that maps public IP address and ports combinations to private IP address and port combinations.

The translation table is implicitly set up when a packet is sent from the private network to the public network. The association is kept alive for a certain time and is refreshed every time a new packet is sent from the same origin. This fact is used by STUN (RFC3489) to set up an association between a public IP address and a private IP address.

In symmetrical NAT, the router stores the address where the packet was sent. Only packets coming from this address are forwarded back to the private address. This algorithm increases the security as it is harder to guess the source IP and port for attackers. Full cone NAT does not perform this check.

There are some mixed variants between full cone NAT and symmetrical NAT. Restricted port NAT works similar like symmetrical NAT, but uses only one port association.

Hairpinning is the ability of the NAT to route packets coming from the private network addressed towards a public IP address binding back to the private network. Not all routers



are supporting this feature.

## Symmetrical RTP

Symmetrical RTP is a trick to extend the number of cases when communication can be established. A SIP user agent that supports symmetrical RTP waits for the first RTP packet coming in and then sends its media stream back to the IP address from which it received that packet. Symmetrical RTP works always if the user agent that does symmetrical RTP is on a globally routable address. However, this algorithm can easily be cheated (port spraying) and therefore implies a certain security risk.

## Signalling SIP

SIP traffic is relatively unproblematic because SIP typically is not as time critical as media. Usually, it is ok to route SIP packets through a longer path than media.

In SIP it is legal to send from a different port than the receiving port. When this is being done, there is no way of supporting these devices behind NAT. However, some phones offer an option that disables this mechanism so that the sending port is the same as the receiving port.

Typically, the SIP proxy will run on a public IP address where it is possible to deal with all kinds of NAT. Keep-Alive messages may keep the NAT binding open (for example, short registration periods or non-SIP messages).

## Media RTP

Media is much more problematic than SIP because users are sensitive to delay in a voice conversation. When the delay is too long, the speakers need to be disciplined not to interrupt the other person when starting to speak. Also, the ear is much more sensitive to echo when the media delay becomes too long. The effect is known from intercontinental calls where the speed of light increases the delay for voice transmission.

SIP was designed for peer-to-peer communication. That means the user agents

(telephones) send the media directly to the other user agent. This approach is the best way to minimize the delay; however it becomes a problem when NAT is involved.

## Classification of User Agents

From a proxy point of view, available user agents can be classified into the following categories:

- Public IP devices. These devices operate on public IP addresses and don't need any specific support regarding NAT. The true location of these devices may be in a private network, as they might have allocated a public identity using mechanisms like UPnP™ [3]. These devices are most welcome as they don't cause any additional requirements.
- STUN devices. Phones that operate behind full cone NAT and allocate public IP addresses themselves fall into this category. The only support that the proxy needs to give is a STUN server. Apart from that they act like public IP devices.
- Stupid devices. These devices don't attempt to check the NAT type or allocate a public IP address. Often, they are "legacy" devices that have been designed without having NAT in mind. These devices can register only for a short period of time, so that the REGISTER messages keep the port association open (the SIP messages are used to keep the port association). Also, these devices need a NAT-aware media server or other device that forward the RTP packets of these devices.
- Symmetrical NAT devices. These devices may be NAT-aware; however, because they operate behind symmetrical NAT, there is little that they can do. They essentially behave like stupid SIP devices and hope for the support of the proxy.

## Probing Media Paths

Interactive Communications Establishment (ICE) is a method that has been proposed recently in the IETF [4]. The algorithm is simple: A user agent that

supports ICE lists the possible addresses where it could possibly be reached. These addresses may include the private address, an address allocated via STUN, one or more addresses allocated with the TURN protocol or an address allocated with UPnP. Because practically it is hard to predict which of these addresses are visible to the other user agent, all of the possible addresses are proposed to the other user agent.

The other user agent sends test packets to the possible addresses. Picking the first reply on the test packet will establish a working media path and it will also probably be the fastest connection. STUN is being used for these test packets.

## The Role of the NAT Filter

When a user agent is not able to allocate a globally routable address or it is not sure if it found enough possible addresses, a NAT Filter (NATF) can help out.

Some vendors call this component session border controller. However, we prefer the term "NAT Filter" because this component is not the end of a session, from a SIP perspective it is just a proxy in the signalling path.

Again, the way a NAT Filter works, is simple. For the signalling, the NAT Filter keeps the NAT alive with bogus messages (which can be SIP messages or other non-SIP message). It patches the messages in such a way, that other user agents will address the NAT Filter instead of the user agent when they want to deliver a message. The NAT Filter then forwards the message to the user agent using the connection which is kept open with the keep-alive messages.

When the NAT Filter sees a message that contains information about sending media (session description protocol, SDP), it opens a local globally routable port on behalf of the user agent and patches these messages in a way that the destination will send media via this port. The NAT Filter will relay the media to the user agent like it relays SIP messages. Using symmetrical RTP, it can detect the user

agents public media identity and reroute the packets to this destination.

While this approach has the huge advantage that it does work with all kinds of NAT, it has the disadvantage that it increases the media path significantly. For example, when a user A in Tokyo is registered with an operator in New York and wants to call his colleague B (which is registered to a service provider in Sydney and who is sitting in the same office in a private network), the media would have to flow first from Tokyo via New York then via Sydney and then back to Tokyo. Considering the speed of light, the delay would at least be around one second; practically it would be much higher although the user agents are located in the same network.

Unfortunately, it is not trivial to make the media path shorter. There have been some attempts to reduce the problem, but it is much easier to address the problem from the user agent. If the user agent uses ICE, it will try all addresses listed in the SDP attachment, including the port allocated by the NAT Filter. If there should be a shorter path, it will switch to this shorter path. If there is no other way or if the other side does not support ICE, it will fall back to the NAT Filter-allocated port which will work in all cases.

## Optimizing the Media Path for Symmetrical NAT

In the case when both user agents are behind symmetrical NAT the NAT Filter approach will ensure that media will flow between the user agents. However, the Tokyo example shows that this might result in intolerable media delay.

To address this problem, TURN [5] comes into play. The idea behind this approach is to allocate identities on several places in the Internet and to propose all of the allocated ports to the other user agent. If the ports are allocated on all continents, the other user agent will automatically pick the TURN server with the shortest delay. In the Tokyo example, a TURN server located in Japan will reduce the

delay to a tolerable level (if there is not even a direct path between the user agents).

## Operation of the 4S

Because of the importance of the NAT Filter, snom offers a snom NAT Filter (ssbc) as part of the snom 4S software package. There is a separate documentation available for the operation of the ssbc.

This approach obsoletes the old approach of the snom 4S proxy to keep the SIP connection alive using short registration periods and to use the snom 4S media server for media relay.

## Conclusion

Using a NAT Filter and ICE-capable user agents, customers will enjoy two-way audio and short media delay in most cases.

snom phones support ICE since version 2.04. The snom NAT Filter is part of the snom 4S SIP solution.

## Questions and Answers

**Question:** *Does the solution require any user side software or user side firewall/NAT reconfiguration?*

**Answer:** No. The NATF automatically detects if a user agent behaves correctly behind NAT and then decides if it needs the support of the NATF. Of course, this algorithm is independent from the user agent type (snom, other vendors).

**Question:** *Does the solution require anything beyond an outbound proxy setting from the UA perspective?*

**Answer:** We currently support only UDP transport layers. But as far as I know only the new MS messenger has problems with that (when authentication is required).

**Question:** *Does the solution include SIP proxy/registrar?*

**Answer:** No, you still need a snom 4S proxy/registrar or any other SIP-compliant proxy/registrar.

**Question:** *With what SIP proxy/registrars has the solution been tested for interoperability?*

**Answer:** We are using the standard SIP stack from snom for the NATF. That means we should have touched all equipment that is available today.

**Question:** *Does the solution support all types of NATs including full cone, restricted cone, port restricted cone and symmetric NATs?*

**Answer:** Yes. The assumption of the NATF is that it's symmetrical NAT (pessimistic approach). If there should be something less strict, it will also work. The built-in ICE will find more efficient paths automatically if they exist.

**Question:** *Does your solution work through multiple levels of NAT?*

Yes. There is no problem because of the pessimistic approach.

**Question:** *Does your solution require that all RTP media traverse through the server or is there a mode where media can flow from UA to UA rather than through the server? Is this configurable? Under what conditions if any can media flow from peer to peer?*

**Answer:** We do this with the ICE approach. Essentially, the NATF adds "just" another ICE contact. For non-ICE capable devices, this contact will be the primary contact – so that communication works all the time, even if not very efficient.

**Question:** *Does your solution support audio and video? Has it been tested with video? Is it codec dependent? If so, what codecs are supported?*

**Answer:** The NATF does not look at the media lines of the SDP. Therefore, there is no problem with video, text, online gaming or whatever media types people would like to use.

**Question:** *How scalable is the solution? (e.g. multiple servers, load balancing, failover, etc). How many simultaneous audio and audio/video sessions can be supported per unit?*

**Answer:** The maximum per host of course depends on the CPU. However, as we

are just relaying media it should be able to deal with several hundred calls at the same time with a modern CPU. If that's not enough you can use several boxes and link them via DNS SRV.

**Question:** *Does your solution support connection oriented media (comedia) as per <http://www.ietf.org/internet-drafts/draft-ietf-mmusic-sdp-comedia-06.txt>?*

**Answer:** Yes, but only UDP. Actually, it is necessary to get the NATF working properly.

**Question:** *What platform requirements are necessary (e.g. CPU, Operating System, etc)?*

**Answer:** The NATF will be available for Linux (SuSE/RedHat 7/8/9) and Windows 32.

**Question:** *Does the solution provide usage accounting/CDR?*

**Answer:** No. This must be done with the proxy/registrar. An outbound proxy inside the NATF makes sure that no one can escape the authentication procedure.

**Question:** *It appears the snom 4S proxy/registrar and the NATF can coexist on the same platform. Is this correct? Does this present any concurrent call limitations or scalability concerns that should be taken into consideration for deploying in such a fashion?*

**Answer:** Yes they can coexist. The NATF is a "real time" task while the proxy is not so time critical. I would say on a network with less than 1000 users it should be ok to run it on the same box. With more you have to keep an eye on the CPU load. In a large network, you would choose several boxes for the NATF anyway.

**Question:** *Can you describe the conditions under which "media path optimization" can take place where the media can flow from UA to UA rather than through the media relay? Your previous response indicated that NATF uses the ICE approach. Can you summarize what UA characteristics/behavior is required to support media path optimization (e.g. is the UA required to support REINVITES for media path redirection or are there other specific UA requirements)?*

**Answer:** No we don't do re-invites and there are no specific requirements to the user agent. The NATF handles the worst case

scenario while the user agents may handle more optimistic scenarios. The idea of ICE is to just try it out — and take the best path. The NATF is just one of these paths. Sounds simple, is simple but effective.

Other NATF have much more logic that try to optimize other scenarios as well. However, no NATF will be able to safely make sure that two user agents in the same private network talk to each other (the "call the colleague case") – which is a very obvious case that must work properly. The only solution to this today is ICE support on the user agent.

**Question:** *Is UA required to support DNS SRV?*

No. If you run the NATF on port 5060 you can also use DNS A.

**Question:** *If the UA is behind a NAT (let's say it is not a symmetric NAT), and the UA is configured with NATF as outbound proxy, can the RTP media traverse from peer to peer if the UA has no STUN functionality or other „network intelligence" functionality or will the RTP media always traverse the NATF in such a case. What I am trying to get at are the minimum requirements from a UA perspective that will allow media to flow from UA to UA (not for a worse case symmetric NAT scenario but for a non symmetric NAT case).*

**Answer:** The minimum requirement is that the UA does NOT try to allocate a „false" identity (for example, allocate and use a STUN port when behind symmetrical NAT). I think most of the UA have this behavior. Unfortunately, some of the snom software releases do it (but there is a flag that turns this off).

As another requirement, the UA must send media as soon as possible. This is because of the symmetrical RTP algorithm. Unfortunately, MS messenger does not start sending media until VAD kicks in. But most of the other devices don't perform VAD or immediately start sending media.

**Question:** *Does NATF modify (shorten) UA registration expiration intervals? I did not notice short expiration intervals in the 200 OK returned by NATF. I requested 1800 sec and 200 OK contained 1800 sec. How are NAT bindings to remain in effect in such case?*

**Answer:** The NATF does not interfere with the registration intervals. The registrar should choose a reasonable short time (maybe ten minutes). The refreshing is done by keep-alive packets which are sent from the NATF to the UA (every 15 seconds; configurable parameter).

However, I realize it does make sense that the NATF proposes a shorter time so that the registrar can accept shorter refresh intervals for those UA that need it.

**Question:** *Also I do not notice a modified contact address in the 200 OK response to the REGISTER request. If UA behind NAT is registering, shouldn't the contact address be modified in the location server so future unsolicited INVITES can reach the UA?*

**Answer:** Well, that's because some UA have problems if they see something different! But if you look at the other side of the NATF, you will see that the registration identity is changed to another value.

## References

[1] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and E. Schooler, "SIP: Session Initiation Protocol", IETF, RFC 3261, June 2002, <http://ietf.org>.

[2] Rosenberg, J., Weinberger, J., Huitema, C., Mahy, R., "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", IETF, RFC3489, March 2003, <http://ietf.org>.

[3] UPnP™ Forum, <http://www.upnp.org/>

[4] Rosenberg, J.: "Interactive Connectivity Establishment (ICE): A Methodology for Network Address Translator (NAT) Traversal for Multimedia Session Establishment Protocols" (Internet draft), <http://ietf.org/internet-drafts/draft-ietf-mmusic-ice-01.txt>

[5] TURN, <http://www.ietf.org/internet-drafts/draft-rosenberg-midcom-turn-04.txt>

snom technology Aktiengesellschaft  
Pascalstr. 10B, 10587 Berlin, Germany  
Phone: +49 (30) 39833-0  
mailto: [info@snom.com](mailto:info@snom.com)  
http: [www.snom.com](http://www.snom.com)  
sip: [info@snom.com](sip:info@snom.com)

© 2003-2004 snom technology AG  
All rights reserved.

**snom**  
VoIP phones