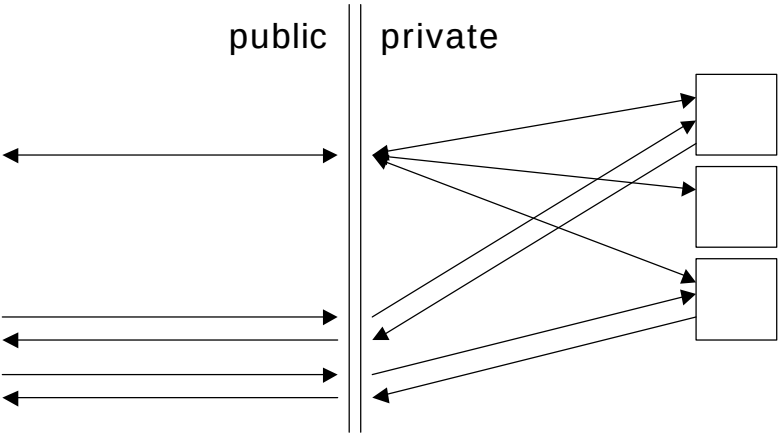
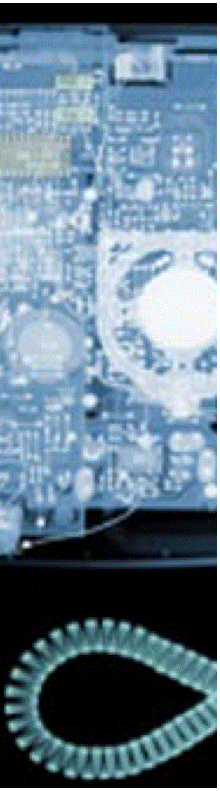
snom 4S SIP NAT gateway

User Manual

Version 0.97



snom VoIP-telephones



IMPORTANT:

snom reserves the right to make changes without further notice to any products herein. snom makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does snom assume any liability arising out of the application or use of any product, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. All operating parameters must be validated for each customer application by customer's technical experts. snom does not convey any license under its patent rights nor the rights of others. snom products are not designed, intended, or authorized for use as components in systems intended for applications intended to support or sustain life, or for any other application in which the failure of the snom product could create a situation where personal injury or death may occur. Should the user purchase or use snom products for any such unintended or unauthorized application, the user shall indemnify and hold snom and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that snom was negligent regarding the design or manufacture of the part. snom and are registered trademarks of snom technology Aktiengesellschaft.

For more information, mail info@snom.de, Pascalstr. 10E, 10587 Berlin, Germany, sip:info@snomag.de.

Preface

SIP is becoming more and more accepted in the VoIP area. Many companies are working on SIP solutions and prepare great products that will make telephony much easier and better. However, in many installations NAT is used and SIP messages and the associated RTP cannot flow through the NAT gateway without additional overhead. This was the reason why we decided to add a complementary product to our SIP phones: a SIP NAT gateway.

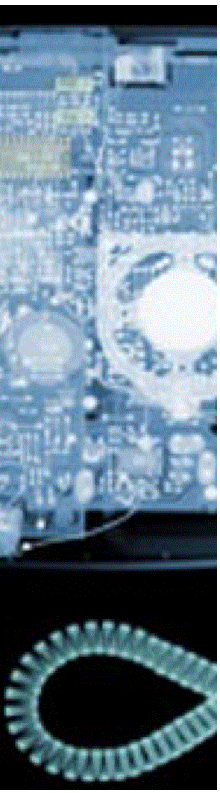
With our experience in VoIP technology, adding this soft product was easy. However, we implemented only those features that we think are most useful and simple in the current VoIP environments.

Interoperability is important for us. We tried to stick to the SIP standard as good as possible and tested with phones of others vendors. We hope that this helps building up a flourishing VoIP telephony where the products of the different vendors work together like the products in the computer industry do today. We believe that having a choice is good for you as a customer and therefore it is good for us.

Let's get VoIP up and running!

Dr. Christian Stredicke
snom technology AG

Contents



Theory of Operation

The NAT Problem

When the Internet was defined, only few computers were connected to the network. The designers used 32 bit addresses for identifying the network elements and introduced different classes for networks. Address areas were assigned to important institutions and regions.

Over time, the Internet community ran our of addresses. That lead to the development of “IPv6”, Version 6 of the Internet Protocol. Instead of using 32 bits, 128 bits are used for addressing — more than the number of atoms of the planet and surely enough for the near and far future. However, the installations in place is mostly not able to deal with the new protocol.

That was the reason why a trick was used to increase the number of computers that can be attached to the Internet: Network Address Translation (NAT).

The principle is simple. A computer may have up to 65535 ports for each protocol family, usually only a few of them are actually used. By associating ports with computers the number of computers associated with one Internet address can be easily multiplied. From the Internet, it seems that there is only one computer, however this device just dispatches the packets to the network behind.

The principle can easily be used for firewalls. The NAT computer checks the packets for permission to traverse the NAT firewall. A whole industry grew around this important problem.

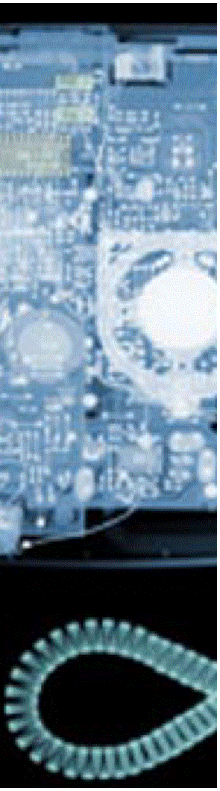
For connection oriented protocols, the NAT principle can easily be managed by the NAT gateway itself. It keeps an internal list of the open connections and can forward packets accordingly.

However, voice over IP mandates connectionless protocols. The voice packets need to be transported over UDP, so that packets can be transported in real time. For the NAT gateway, there is no way to find out

where the packets should be sent without prior knowledge about what is going on.

Some vendors are working on a firewall control protocol that tells the NAT gateway, which ports it should associate with what computer. However, the standard is not finalized yet.

Therefore, the approach for solving this problem is to use a application layer gateway, that looks at the SIP packets and changes them in a way, that the destination can use them. If media channels are proposed, the gateway opens tunnels accordingly, so that the RTP packets are forwarded to the right destination.



Message Flow

SIP uses proxies for forwarding messages. Stateless proxies just forward packets without knowing the complete context the messages are in. By putting a stateless proxy on the NAT gateway, the messages can easily be modified to meet the requirements on an application layer SIP NAT gateway.

SIP URLs contain a user and a host part, e.g. sip:user@host.com. The user name can be used to encapsulate the private address which is not visible from the outside world. For example, if the NAT gateway is located on the computer nat.company.com, the private SIP address usera@192.168.1.2 becomes usera%40192.168.1.2@nat.company.com (the @ becomes %40, so that there is no ambiguity). If the NAT gateway receives a message for the encapsuled address, it just strips its hostname and uses the address given in the user name.

The SIP NAT gateway just checks if the message is supposed to be sent to the Internet side or the NAT side. It then patches the messages headers according to the encapsulation rules and sends the message.

It also looks at the session descriptions (SDP) which may be part of the SIP messages. If a port is advertised, it opens a tunnel for the RTP connection which forwards the RTP packets on arrival to the right destination.

TIP

Routing calls through the NAT gateway requires “Record-Route” header. This way it is ensured that subsequent messages pass the NAT gateway next time as well. Make sure your equipment is able to handle these headers, they are mandatory according to the latest SIP draft.

Replies will automatically pass the NAT gateway, as it always inserts “Via” headers in the requests.

The SIP NAT gateway inserts Record-Route headers into the messages other than REGISTER. This ensures that all messages within the transaction flow the NAT gateway.

For REGISTER, the SIP community has developed a special “Record-Routing” mechanism. This method inserts a Path-header into the REGISTER messages.[1] So that the proxy can set up an initial route when a message comes in for the registered user agent.

By requiring SIP timer refreshes (or using a default timeout), the gateway knows when the RTP tunnels can be closed, so that the number of involved resources is kept at a reasonable level.

PPPoE

Many users use telephone lines for transmitting Internet packets. Telephone lines are just pipes that take their input on one side and transport them to the other end (point to point). The Internet society invented another protocol to bridge this distance: The point to point protocol (PPP). This protocol uses the given bandwidth with less overhead than the plain Internet Protocol. This is the reason why many DSL providers use this protocol.

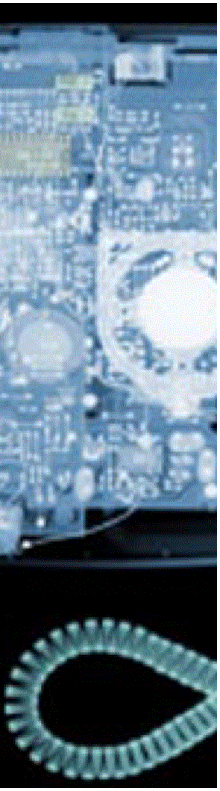
The physical layer used for accessing the DSL modem today is Ethernet in most cases. Therefore, to trans-

[1] See http://www.ietf.org/internet-drafts/draft-willis-sip-path-02.txt for details

port the data from a more intelligent router to the modem, PPP over Ethernet (PPPoE) is used.

Due to the shortage on IP addresses, some DSL operators change the IP addresses of their clients on a periodical basis (e.g. daily). This should prevent that an IP address is blocked by a user that is not active. That means, that the IP address changed sporadically and that connections get lost during this reassignment of IP addresses.

Therefore, SIP NAT gateway has to poll for address changes and reflect these changes in future messages. A static set-up of the public IP address is not enough.



Domain Proxy behind NAT

Many installations don't have only one public address available that is used by the NAT gateway. However, they can be reached via DNS from the public network as well as from the private network.

Example: Company X has the domain smallcomp.com associated with 123.123.123.123 and would like to be reachable at e.g. sip:info@smallcomp.com.

To make this possible, all incoming packets on the SIP NAT gateway must be forwarded to the private address where the proxy is running. You can achieve this by starting the gateway with the option --default <adr>, where the address is expressed as adr:port, e.g. 192.168.3.4:5062.

Starting

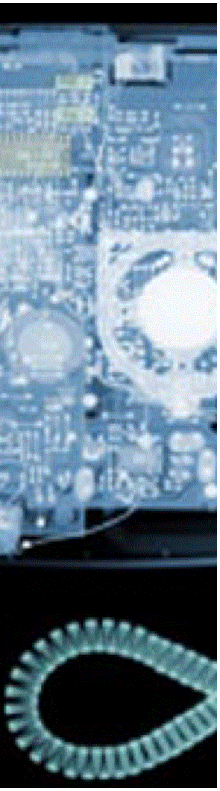
Manual Start

You can start the NAT gateway on your Linux NAT router with the command sipnat. The following program options are available.

- --log <n>: Set the log level to a value between 0 and 9. 0 means only the most urgent messages get through, 9 means the sipnat is verbose.

- --logfile <file>: Set the log file.
- --no-daemon: Don't start the gateway as daemon, keep attached to the terminal and don't fork child processes.
- --no-path: Don't insert Path headers, use patched user names instead. This mode allows to use proxies that don't support the Path header. Ask your proxy vendor if he supports Path registrations.
- --rtp-forward-out: Normally, RTP packets leaving the private network are not forwarded. This saves performance, however may cause trouble with some firewalls. This option turns explicit RTP forwarding on.
- --private <adr>: Explicitly define the private address. This is useful when the private address cannot be determined automatically.
- --public <adr>: Explicitly define the public address.
- --sip-port <port>: Define which SIP port to use. Defaults to 5060. Don't change this parameter unless you know what you are doing.
- --connect-port <port>: Open another input port for SIP traffic. This port can be attached to the firewall, that forwards packets to this port. This way the gateway can work as application layer gateway.
- --default <adr>: Forward all packets that don't have an explicit route to the given address. This way proxies inside a private network can serve as domain servers.
- --rtp-port <start> <end>: Define which port should be tried for passing RTP data through the NAT gateway. Default values are 10000 and 11000 (exclusively).
- --pppoe-dev: Define the PPPoE device, defaults to ppp0. See ifconfig for the available devices.
- --version: Show the version number.

sipnat will try to guess the public and private address on its own. The public address is polled automatically if it has not been entered manually. This is useful because some DSL internet provider change the public IP address on a periodical basis.



Automatic Start

If you want the gateway to be stated automatically after a reboot, you need to set up some files as root. We describe here the procedure for SuSe Linux.

1. Copy or link the proxy in the binary to /usr/sbin/sip-natgw
2. Copy or link the startup script sip-natgw.sh to /etc/init.d/sip-natgw
3. Link the startup script to /etc/init.d/rc[23].d/[SK]20sip-natgw (in total 4 links).
4. Link /usr/sbin/rcsip-natgw to /etc/init.d/sip-natgw
5. Set up the variable START_SIP_NATGW to “yes” in the /etc/rc.config
6. Set the necessary options that you would use in manual startup in the SIP_NATGW_OPTS variable.

You can then try start the gateway with the command “rcsip-natgw start”. Check with the ps command if you can see the process. Reboot the system and check if after the reboot the sip NAT gateway was started automatically.

Proxy Chain

When a phone want to initiate a phone call to a destination that is outside the NAT, sipnat needs to be in the proxy chain. This can be achieved in different ways: By using an outbound proxy and by using the IP gateway.

Outbound Proxy

SIP defines a model of the “outbound proxy” similar to a http proxy, that handles all messages that come from a phone. By pointing the outbound proxy to the sipnat gateway, it can be ensured that all messages go through the application layer gateway, even if they don’t traverse the NAT.

To avoid unnecessary traffic, the phones can be pointed to another proxy that is within the private network which devices if messages have to traverse the SIP NAT gateway. This keeps traffic away from the NAT gateway and avoid unnecessary opening of RTP tunnels. The snom 4S proxy supports this feature.

IP Gateway

Unfortunately, not all phones on the market support the feature of an outbound proxy.

In Linux, the kernel can be configured to forward all packets to an application, e.g. the NAT gateway. You need to set up a filter rule like this (using –sip-port 5069):

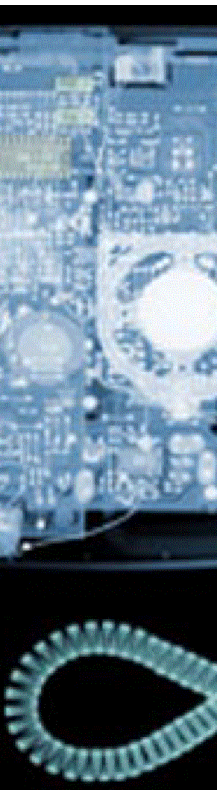
```
iptables –t nat –A PREROUTING –i eth0 –p UDP -
-dport 5060 –j REDIRECT –to-port 5069
```

In many installations, the NAT gateway is used as firewall as well. For SuSe installations, setting up the iptables does not work in these cases, because the startup script for the firewall clears the iptable setup. In this case you’ll need to set up the ports in /etc/rc.config.d/firewall2.rc.config like this:

```
…
FW_SERVICES_EXT_UDP="5060 10000:10999"
FW_REDIRECT="192.168.0.0/16,0/0,udp,5060,5061"
…
```

TIP

For the snom 100 VoIP phone, you find the outbound proxy in the menu Settings, SIP, Outbound Proxy.



The first line allows to access port 5060 (SIP port) and the ports 10000 to 10999 (RTP ports, see command line argument to the SIP NAT gateway). The second line forwards packet arriving at port 5060 to the port 5061, on which the SIP NAT gateway is expecting packets (using –connect-port 5061).

However, the method of using the IP gateway has the disadvantage that SIP messages that go to other port that 5060 (which is perfectly legal) do not pass the SIP NAT gateway properly. Therefore, we recommend using outbound proxy in the phones instead.

Quality of Service

Telephone calls require a good quality of service (QoS), meaning that the associated media UDP packets are transported with a higher priority than other packets.

Reliable protocols like TCP ensure by packet repeating that the content does not get lost. This can become a problem, when the same line is used for receiving emails and VoIP. The underlying transport mechanisms need to know about the priority of the packets.

Many mechanisms have been proposed for controlling and ensuring the QoS ranging from RSVP, MPLS, ATM etc. Practically, most installations support DiffSrv and can deliver good quality with this mechanism.

Linux supports the different mechanisms. The SIP NAT gateway uses DiffSrv marking of RTP packets. However, it is important that the kernel was compiled with the corresponding configuration.

The SIP NAT gateway does not check if the underlying bandwidth is enough for transporting the media stream. If you try to make two phone calls over a 128 kBit/s line (occupying more than kBit/s per media stream with ulaw encoding), will result in a breaking phone call.

Release Notes

Versions

Version 0.94

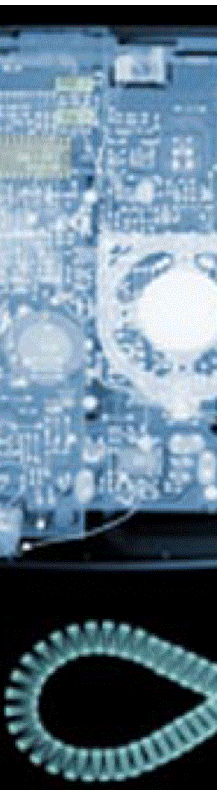
- First release as a standalone package

Version 0.97

- Path headers
- Improved resource management. Ports were not closed properly in previous revisions
- Support for domain proxy behind NAT
- Outgoing RTP packets don't traverse the gateway by default

Open Issues

- TCP traffic not supported. Workaround: Use UDP.
- No automatic selection for low rate codecs. Workaround: Have enough bandwidth or set up low rate codecs in the user agent.
- No transcoding of RTP streams. No work-around.
- Automatic detection and signalling of STUN messages is missing (no proposal in STUN available). Workaround: Ignore this problem and accept the additional traffic or manually set up the user agents.
- Bandwidth overflow. If too many channels are open, the traffic will cause a jam. This cannot be controlled by the gateway, no workaround.
- Currently, only Linux is supported. Code compiles with Windows as well, but set up procedure is not clear (any suggestions?).



snom technology Aktiengesellschaft · Pascalstr. 10 · D-10587 Berlin
phone: +49 30 39833-0 · mailto:sip@snom.de · sip:info@snomag.de