NAT44 Address translation

Best Practice Document

Produced by the UNINETT-led working group
on campus infrastructure

Authors: Svein Ove Undal (UNINETT), Tom Myren
(UNINETT), Harald Terkelsen, Gunnar Bøe (UNINETT)

April 2015

Document No:	UFS 144
Version / date:	Version 1.0; April 2015
Original language	English
Original title:	“NAT44 Address Translation”
Original version / date:	Version 1.0; June 2014
Contact:	svein.undal@uninett.no

UNINETT is responsible for the contents of this document. The document was developed by a UNINETT-led working group on campus infrastructure.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 605243, relating to the project 'Multi-Gigabit European Research and Education Network and Associated Services (GN3plus)'.

Table of Contents

Table of Figures

Executive Summary

Until recently, the HE sector has had good access to IPv4 addresses. Today, however, the HE sector is also running out of addresses. We therefore make a recommendation on how Network Address Translation (NAT) can be used in a simple but effective way.

We recommend using a Linux server for NAT44. It has been shown that a virtual server is easily capable of being the NAT44 gw for a /20 network (~4000 clients).

One important security aspect is the continued traceability of online clients. Logging via Netflow/IPFIX is used for this purpose.

We expect organisations to start using IPv6 as soon as possible, and therefore also include information on how NAT44 can be used in conjunction with IPv6.

1 Introduction

NAT has long been available, both as an integrated router function and as dedicated hardware. With IPv6, more versions of NAT have appeared, such as NAT64 and NAT46. Address translation between different IPv4 addresses is therefore in this document named NAT44. Native IPv6 should be implemented in addition to NAT44 to offload as much traffic as possible from the NAT44 gw.

One has to use private addresses from one or more of the following RFC1918 prefixes:

* 10.0.0.0/8 (255.0.0.0)

* 172.16.0.0/12 (255.240.0.0)

* 192.168.0.0/16 (255.255.0.0)

It is important not to assign too large an RFC1918 network to 1 public address, in order to avoid port starvation. In our model, we use 1 public address for 255 addresses (/24).

UNINETT has explored and tested various possibilities for NAT44. One solution may be to use a dedicated router with NAT support, but they are often expensive and logging has turned out to be a challenge in terms of CPU usage.

2 Recommended Solution

This solution is based on a Debian installation on a virtual machine, here using iptables for NAT44 and a netflow module for logging data. For security and operating reasons, we recommend using Debian.

The virtual machine should have minimum of 2 gigabytes of RAM, a 2–4-core CPU, access to minimum of 30 gigabytes of storage and two network cards. We recommend that log data is sent to another machine via the netflow module. However, log data may also be stored locally.

The network size should not exceed a /20 network; if it is larger, there will be too much broadcast traffic. For larger networks, it is better to run multiple instances of the NAT44 solution.

Note that the virtual network card e1000 used in older VMware versions may cause performance problems under very heavy network traffic. The recommended virtual network card is vmxnet3 or newer.

The NAT44 solution can run on both virtual and physical hardware. Even though a physical server could theoretically provide better network flow and make better use of the hardware, tests show that there is little difference between the two alternatives.

The NAT44 part of the solution is SNAT[1](Source NAT) with 1 public IPv4-address for each 256 clients (/24). You therefore need 4 public IPv4 addresses to use NAT on a network with about 1,000 clients (/22). Here is one example:

```
10.0.0.0/24 -> 203.0.113.120
10.0.1.0/24 -> 203.0.113.121
10.0.2.0/24 -> 203.0.113.122
10.0.3.0/24 -> 203.0.113.123
```

1 http://www.netfilter.org/documentation/HOWTO/NAT-HOWTO-6.html

3 Installing and Configuring NAT44

You will find the solution in the apt repo of software.uninett.no, and as a VMware image. It was created to be fast and simple. No extensive or special Linux skills are needed to get it running.

The solution requires minimal administration, like an ordinary Linux server with updates to the operating system and possibly configuration backups.

The NAT44 gw requires two network cards: eth1 connected to a VLAN with private IPv4 addresses (RFC 1918), where the NAT44 gw will be the gateway for IPv4 clients, and eth0 connected to a VLAN with public routed IPv4 addresses.

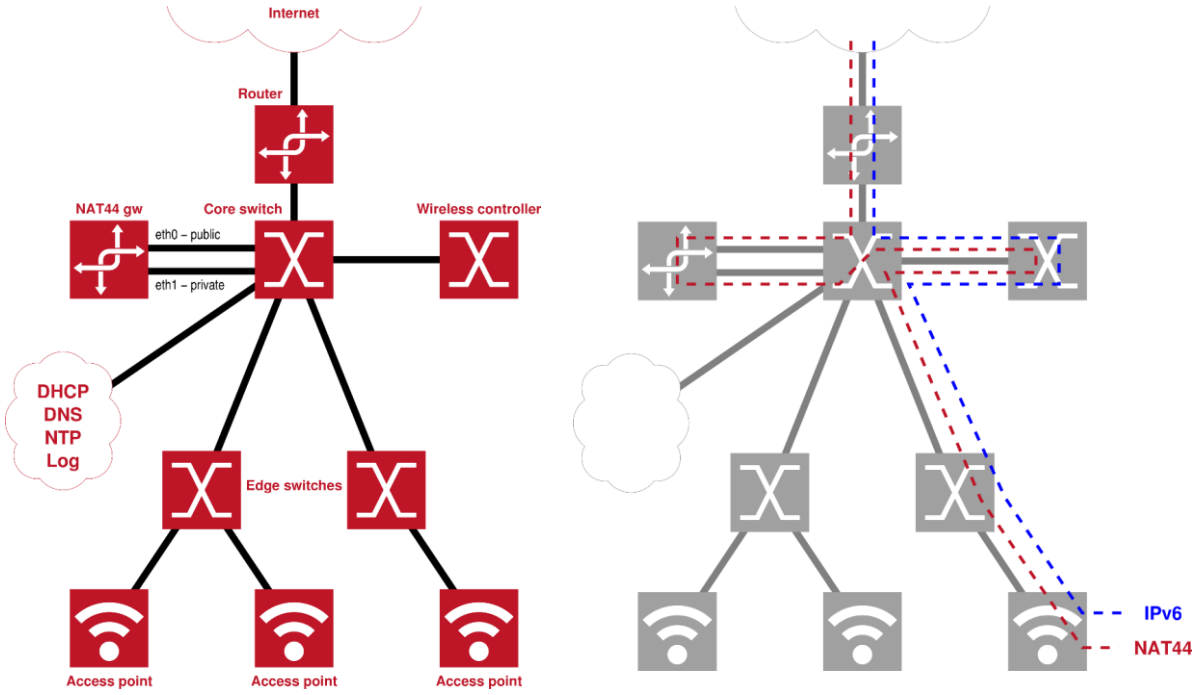

Figure 3.1: Network for NAT44 gw

The advantage of this approach is that all IPv6 traffic will go directly to the core switch and not via the NAT44 gw, resulting in less load and log data on the NAT44 gw.

3.1 General Information on the Installation

This guide assumes that Debian 7 is already installed on the machine designated as the NAT44 gw. There should be network access on the network interface eth0, which is on a public network.

All installation is through the Linux terminal interface. It is assumed that you have a new Debian 7 installation without a graphic interface.

The "#" defines a shell as root, and the "$" is an ordinary user shell.

Most of the configuration requires a superuser (root), which you get by entering the shell command:

```
$ su root    (root password)
```

Or

```
$ sudo -s     (own password)
```

You will see the $ changing to # in your terminal, meaning that you are now logged in as root.

3.2 Installing NAT44 from ''apt.uninett.no''

As an alternative to the following procedure, a VMware image can be found at ftp://ftp.uninett.no/pub/network/NAT44/. It is installed with Debian 7 and the packages NAT44 and NTP. If you use this image, you can go directly to section 4: Configuration.

Add uninett as an apt repository:

```
# echo ''deb http://apt.uninett.no/debian wheezy nat'' >>
/etc/apt/sources.list
```

Add a key:

```
# wget http://apt.uninett.no/uninett_apt.key; apt-key add
uninett_apt.key; rm uninett_apt.key
```

Update apt:

```
# apt-get update
```

Install NAT44:

```
# apt-get install nat44
```

Now everything should be installed and you should be ready to configure the NAT44 gw.

This may also work on other Debian-based distributions, e.g. Ubuntu, following the same procedure.

Note that this is developed and tested on Debian so for security and operating reasons, we recommend using Debian.

4 Configuring NAT44

4.1 NAT44 Configuration

This section configures your network for NAT44. Here is a brief explanation of the various settings.

The file that should be edited is /etc/nat44/nat44.conf and you can edit the file with nano, or your preferred editor.

```
# nano /etc/nat44/nat44.conf
```

```
internal-if = eth1
```

internal-if is the internal network interface. This is the interface that will serve as a gateway for your clients.

```
external-if = eth0
```

External interface: This is the interface that should be connected to the Internet.

```
admin-address = nnn.nnn.nnn.nnn #
```
(for example, 204.0.113.120)

This is the external public network address of your NAT44 gw.

```
internal-network = 10.0.0.0/22
```

This is the internal network. It is not recommended to have a larger network than /20 (~4000 clients).

```
internal-gw = 10.0.0.1
```

This will be the address of the internal interface “internal-if”, the gateway for your clients.

```
external-network = nnn.nnn.nnn.nnn/nn #
```
(for example, 204.0.113.0/24)

This is where you define the external network.

```
external-addresses = 204.0.113.120 204.0.113.121 204.0.113.123
204.0.113.124
```

These are the external addresses used to translate private addresses to public addresses. You need 1 public IP address per /24 (~ 254) clients, meaning that you need 4 public addresses on a /22 internal network.

Note that the admin address may also appear in the list of external addresses.

```
default-gateway = 204.0.113.1
```

The gateway from the server to the network.

```
dns = 8.8.4.4 8.8.8.8
```

DNS. It is important to change the Domain Name Server to your own DNS.

```
log = netflow
```

IPT netflow module for logging. Set it to «None» if, for particular reasons, you don't want logging.

```
log_ip = 127.0.0.1
```

The address the log is sent to. If it is set to 127.0.0.1, it will install flow-tools and create a folder in /var/flow/self/ for flow data.

```
log_port = 2055
```

The port where log data is sent.

Getting your flow log onto a central log server is a major advantage. To use such a server, you set log_ip to the server's IP address and log_port to the port it is listening on. Check that this port is not blocked by a firewall or router filter.

When configuration is correct, you are ready to run NAT44.

4.2 Further Configuration of NAT44

Run the code (script) that performs the entire setup:

```
# nat44
```

It is important to run this code as a superuser (with sudo or as root). Also, be aware that the code includes a number of tests that will stop execution if there are any logic errors in your network configuration.

When finished, it will have created the following files:

/etc/nat44/ipt.conf	(iptables script)
/etc/networking/if-up.d/network	(network file with settings based on /etc/nat44.conf)
/etc/resolv.conf	(overwrites this with settings in nat44.conf)
/etc/networking/interfaces	(not yet in use; will eventually take over functions from the above files)

If everything has gone well, you will have working address translation with logging.

4.3 Logging

It is recommended to place the NAT log on your central log server, since this gives better control and overview of the traffic.

If you are using standard settings for logging locally on your NAT44 gw, you can show flow data with flow-print.

```
 # flow-print < /var/flows/self/2014/2014-03/2014-03-27/ft-
v05.2014-03-27.000001+0100
```

You will see something like this:

```
srcIP dstIP prot srcPort dstPort octets packets
10.0.0.13 158.38.130.7 1 1334 1334 0 0
10.0.0.13 158.38.130.7 1 1334 1334 0 0
```

Ipt-netflow with NetFlow v5 logs post NAT source and pre NAT destination under NetFlow “srcAS” (post-nat-source) and “dstAS” (pre-nat-dport). With NetFlow v9/IPFIX NAT translations will be sent as standard NetFlow Event Login (NEL). Having control on post and pre NAT-ports is important because with iptables SNAT they might not always be the same.

4.4 NTP

In order to trace a client via the log, it is very important that everything be logged with the correct time. The clock of the NAT44 gw must be exactly synchronised with the clock on the other servers.

Use NTP (Network Time Protocol), to synchronise the clock on the NAT44 gw with one or more selected NTP servers.

Install NTP with the apt-get install:

```
# apt-get install ntp
```

Edit the file /etc/ntp.conf using vi or nano

```
# nano /etc/ntp.conf
```

Locate the section that says something like:

```
server 0.debian.pool.ntp.org iburst
server 1.debian.pool.ntp.org iburst
server 2.debian.pool.ntp.org iburst
server 3.debian.pool.ntp.org iburst
```

Remove the debian.pool.ntp.org servers, then add your usual NTP the example shows UNINETT NTP servers.

```
server ntp1.uninett.no
server oliven.uninett.no
```

Then save the file (ctrl+x, y, enter).

Next, issue the command needed for NTP to load the new NTP servers:

```
# dpkg-reconfigure ntp
```

Test NTP:

```
# ntpq -p
```

If you see the output displaying oliven.uninett.no and ntp1.uninett.no NTP info, it's working.

5 NAT44 Failover with isc-dhcp-server

It is not unusual to suffer a periodic lack of public IPv4 addresses i.e. too few addresses in the DHCP pool. To handle this, an alternative has been developed to assign RFC1918 addresses if no more public IP addresses are available. This is made possible by using an isc-dhcp-server. It is found in the Debian repository and the installation is explained below.

5.1 Topology for NAT44 Failover

The topology for NAT44 failover is very similar to the topology we have already explained. You still need 2 VLANS; one with minimum 1 public IPv4 address on the nat44 gw (eth0) and the second VLAN that will have both IPv6, public and private IPv4 addresses. The public networks should have a default route to the core switch and the private networks (RFC 1918) to the NAT44 gw eth1. This solution demands a dhcp service locally on the NAT44 gw handing out both public and private IPv4 addresses.

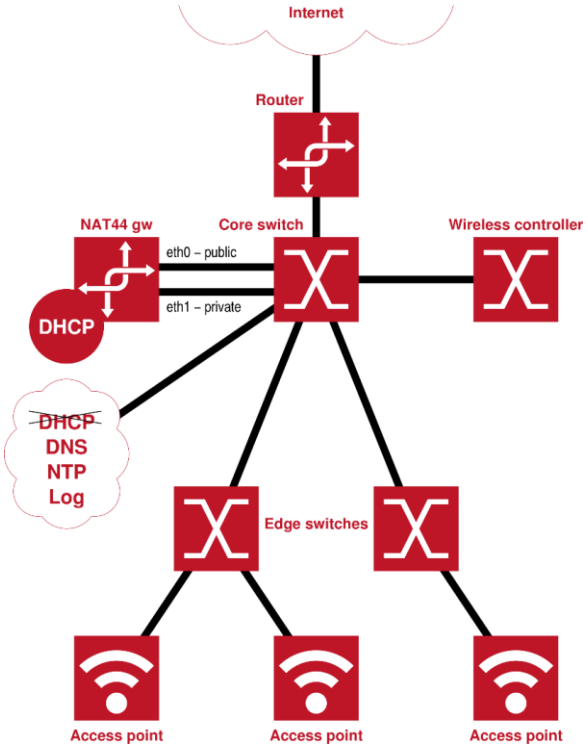

Figure 5.1: NAT44 with its own DHCP

Routing public networks via NAT44 is entirely possible, but that is a more complex configuration and is not covered here.

5.2 Installing and configuring the isc-dhcp-server

To install this software, issue the following command (as root):

```
# apt-get install isc-dhcp-server
```

Edit /etc/default/isc-dhcp-server. Change the part that says INTERFACES="" to INTERFACES="eth1"

Then change /etc/dhcp/dhcpd.conf so that it resembles the configuration below:

```
ddns-update-style none;
option domain-name-servers 8.8.8.8, 8.8.4.4;
default-lease-time 300; #TIMEOUT TIME 5 min
max-lease-time 480; #TIMEOUT TIME 8 min
authoritative;
```

```
log-facility local7;
shared-network backupNAT {
subnet 203.0.113.0 netmask 255.255.255.0 {
option routers 203.0.113.1;
}
pool {
range 203.0.113.130 203.0.113.230; # Here you should use your
public IP addresses for NAT
}
subnet 10.0.0.0 netmask 255.255.254.0 {
option routers 10.0.0.1;
}
pool {
range 10.0.0.10 10.0.2.254;
}
}
```

Here, addresses will be allocated from the first pool until it is full, then it goes on to the next one. Note that there will be two subnets on the same VLAN. This will work fine, but you will need to make changes depending on your local setup. The VLAN that you use must have access to the NAT gw and public gw.

Next, start the isc-dhcp-server:

```
# service isc-dhcp-server start
```

6 Troubleshooting NAT44

The most likely points where problems may arise are covered in this section.

6.1 NAT44

When errors occur, the problem may often be solved by updating the system and then running the NAT44 code again:

```
# apt-get update; apt-get upgrade
```

Next, check that the settings in /etc/nat44.conf is in accordance with the information in the chapter 5.1 before running

```
# nat44
```

6.2 ipt-netflow

This module should be automatically launched by NAT44 and ipt_NETFLOW is also added to the file /etc/modules, so that the module is launched on startup. The settings for the module can be found at /etc/modprobe.d/ipt_NETFLOW.conf

You can see ipt-netflow statistics by typing:

```
# cat /proc/net/stats/ipt_netflow
```

To check the settings of the netflow module, you can type>

```
# sysctl net.netflow
```

If this does not result in any output or an error message, the module has not been launched or is not installed. It can be manually launched with:

```
# modprobe ipt_NETFLOW
```

If this does not lead to any error message, the module has been launched. You can retrieve information with the sysctl command.

If you run NAT44, the script will try to fix ipt-netflow; the alternative is to compile from source.

6.3 Advanced Troubleshooting

There are several tools you can use to get a better overview. For example you can check the firewall rules with:

```
# iptables -t nat -L
```

The command will show the NAT table rules. You should see SNAT rules with RFC1918 addresses as source and a public address that will be the translated address.

You can also view the state table by typing:

```
# cat /etc/net/ip_conntrack
```

This will show the active connections via the NAT44 gw.

To check the status of resource usage on the NAT44 gw, you can use top:

```
# top
```

7 Installing ipt-netflow from source

ipt-netflow may also be installed separately. The best way is to compile the module from the source at https://github.com/aabc/ipt-netflow.

Everything must be run as superuser (as root or with sudo)

First, you have to install the required packages:

```
# apt-get install module-assistant conntrack iptables-dev pkg-
config git
```

Use module-assistant to download the linux source:

```
# m-a prepare
```

You now need the source code:

```
# git clone https://github.com/aabc/ipt-netflow
```

Configure, compile, install:

```
# cd ipt-netflow; ./configure; make; make install
```

You can now test with

```
# modprobe ipt_NETFLOW
```

If there is no error message, everything went fine. You can then continue configuring NAT44 as explained above.

Glossary

GSW	Term used for the router/switch at the boundary of your network
NAT	Network Address Translation
NTP	Network Time Protocol
SNAT	Source NAT
VLAN	Virtual Local Area Network
NAT44	ipv4 to ipv4 port NAT
HE	Higher Education

Complete BPDs are available at http://services.geant.net/cbp/Pages/Home.aspx

campus-bp-announcements@geant.org