Application Note

Implementing Network Address Translation and Port Redirection in ePipe

Contents

1 Introduction

Many SMEs (Small to Medium Enterprises) want to have their own web or email server that is accessible to Internet users without having to go to the expense and complexity of building a DMZ (De-Militarized Zone) or other network for the sole purpose of hosting these services. At the same time they want to host their own server instead of out-sourcing to an ISP (Internet Service Provider) or hosting company.

Assuming such a SME uses an ePipe to connect their LAN to the Internet, the ePipe can use a feature of NAT (Network Address Translation) to direct inbound traffic that matches pre-defined types (protocols) to a specific server on the internal or private LAN. This allows the SME to have a single fixed public IP address on the ePipe's link to the Internet and use private IP addresses for the web and email server on the LAN.

This capability is implemented in the ePipe as *NAT Rule Sets*. A NAT Rule Set is a set of NAT redirection rules that define what type of traffic is to be redirected to one or mote internal hosts or IP addresses.

NOTE	Although only web (HTTP) and email (SMTP) traffic has been mentioned, NAT redirection can be used with any TCP or UDP based protocol that uses a single TCP or UDP port for all traffic to the host on the LAN.

This application note will focus on:

- NAT Theory
  What are NAT, NAT Redirection and Bimap?
- Implementation
  How to implement NAT redirection and Bimap using ePipe.
- Testing
  How to test and troubleshoot NAT redirection and Bimap.

2 Network Address Translation

2.1 What is NAT?

Network Address Translation, put simply, translates the internal or LAN IP addresses of IP packets into one or more external or public addresses as the packet passes through a NAT enabled router. This router usually connects the private LAN to a public network such as the Internet. This is sometimes referred to as Dynamic NAT, Outbound NAT, Basic NAT, or Network Address and Port Translation (NAPT).

There are a couple of reasons for using NAT, the first being that NAT hides the internal addresses of machines on the LAN from prying eyes on the Internet. All IP packets sent to the Internet appear to be coming from the one public address (of the NAT router or ePipe), not the private internal addresses on the LAN. This makes it much more difficult for computers on the LAN to be attacked.

The second reason for using NAT has to do with the finite number of IP addresses available using IPv4 (the current version of IP used on the Internet). Basically the number of unique IP addresses available for allocation to computers on the Internet is diminishing. One solution to this problem is to only allocate a unique IP address to the router that connects an organization's LAN to the Internet, while the LAN can be allocated addresses that are unique for that organization but not globally unique. NAT allows these non-globally unique IP addresses to be used on LANs because NAT hides these addresses from the Internet by translating the internal LAN IP addresses into one or more public (globally unique) IP addresses, usually the IP address of the router on the Internet. Certain ranges of IP addresses were designated for private use and these are displayed in Table 1 below.

Table 1 – Private IP Address Ranges

Class	Reserved IP Address Range	IP Addresses in this range
A	10.x.x.x	10.0.0.1 – 10.255.255.254
B	172.16.x.x – 172.31.x.x	172.16.0.1 – 172.31.255.254
C	192.168.x.x	192.168.0.1 – 192.168.255.254

These addresses are reserved for private use and should not be used in public networks such as the Internet. LANs may be assigned an address range from these private ranges.

2.2 NAT Redirection

NAT redirection (or NAPT – Network Address and Port Translation) is a feature of NAT in the ePipe that redirects traffic sent to the external public IP address and arriving on a given TCP/UDP port number to a pre-defined IP address. For example, if a company has a web server on the LAN and the LAN is connected to the Internet by an ePipe with NAT enabled, then the web server cannot be seen from the Internet, as the only IP address visible from the Internet is the public IP address of the connection from the ePipe to the Internet. In the case of a web server, a NAT redirect rule would be implemented for HTTP (TCP port 80) that would redirect all traffic arriving at the fixed public IP address of the ePipe on TCP port 80 through to the web server on the private network. Figure 1 (below) illustrates this network configuration as well as redirection of SMTP (TCP port 25) traffic to a different server on the LAN.

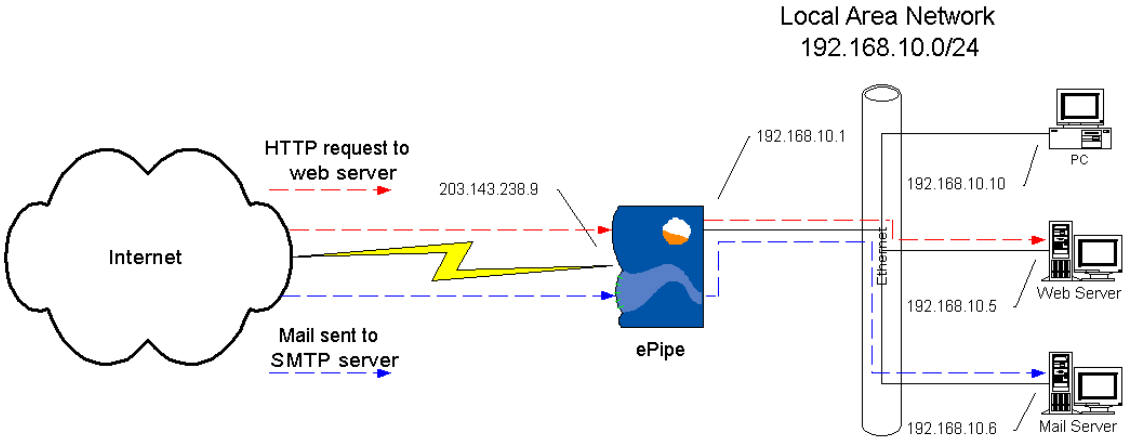

Figure 1 - Physical Network Design with IP Addressing

As shown in Figure 1 above, all packets to a server arrive at the public IP address of the ePipe – 203.143.238.9 in this case. Using a NAT redirection rule in the ePipe, the HTTP request arriving on TCP port 80 is redirected through to the web server on the network, addressed 192.168.10.5. Similarly, email is being sent to the email server on the network. Inbound email (SMTP traffic) would be sent to the public IP address of the ePipe – again 203.143.238.9. SMTP traffic is redirected through to the mail server, which has the internal address 192.168.10.6.

NOTE By redirecting traffic through the ePipe firewall, the responsibility for security of traffic on that particular port number belongs with the application receiving the traffic. In the above example, the web and email servers are responsible for traffic on ports 80 and 25 respectively. It is not advisable to redirect traffic through a firewall unless the server receiving the redirected traffic is secured against any vulnerability in the software receiving the traffic. Please ensure that all patches are applied and other security recommendations adhered to before allowing access to internal servers.

The network designer will need to find out which applications and, therefore, which TCP/UDP ports are required to be accessed from the public network.

Common applications for this include hosting a web (HTTP), email (SMTP) or DNS server. A complete and up-to-date list of assigned or well known service names and their corresponding port numbers can be found at:

http://www.iana.org/assignments/port-numbers

2.3 Bimap

Bi-directional mapping (Bimap) is a way of mapping or directing all traffic destined for one IP address to another IP address. This is frequently used to implement a virtual DMZ, such that packets with a range of unique public destination IP addresses are individually mapped to a number of private addresses on the internal LAN or other network, on a one-to-one basis.

Bimap solves a problem that arises when using Port Address Translation - how to redirect traffic to two servers listening on the same TCP/UDP port number. As an example, Company A requires that its two web servers can be accessed from a public network with separate names/IP addresses. This would be difficult to achieve using NAT redirect rules, as HTTP (TCP port 80) packets sent to the public IP address of the ePipe would need to be redirected to two different servers, which cannot be achieved with NAT redirection. The way to solve this is to map different external unique IP addresses to the separate web servers, thus giving each web server its own identity on the public network (Internet).

There are a couple of requirements for using Bimap rules. The first is that the addresses to be mapped to (i.e. the public IP addresses which will be mapped to the private IP addresses) must be from a different subnet to that used in the public IP address of the ePipe's external interface or link. For example, if the ePipe already had a single link to the Internet and its fixed public IP address, assigned by the ISP, was 203.143.238.9, then a second subnet of IP addresses would be needed for mapping to/from the web servers. These IP addresses could be 202.130.130.10/29[†], for example. This would give eight IP addresses, the first of which is the network address and the last being the broadcast address. There are then six IP addresses that could be used in bimap rules.

The second requirement is that the ISP must route all traffic for the 202.130.130.10/29 subnet to the fixed public IP address of the ePipe, which in this example is 203.148.238.9. Figure 3 shows how the mapping of addresses will work.

2.4 NAT in ePipe

ePipe implements NAT independently on each link via a check box that is checked by default when the SIA Setup Wizard is run. This turns on dynamic NAPT for that interface, which translates the source IP address of packets leaving the ePipe interface into the public IP address assigned to that interface.

ePipe implements NAT Redirection and Bimap via NAT Rule Sets. NAT Rule Sets are sets of NAT rules. Each rule defines a specific map, bimap or redirection. As map rules are handled automatically by the ePipe (as mentioned above), this document only looks at how to implement redirection and bimap rules.

3 Design and Implementation

3.1 NAT Redirection

NAT redirection rules may be configured either when using the SIA Setup Wizard or by clicking on Advanced, then NAT, within the ePipe Management Assistant

[†] This range of addresses is written in CIDR (Classless Inter-Domain Routing) form and is randomly chosen. CIDR Addresses take the form of IP_Address/Number_of_subnet_bits. For example, a subnet mask of 255.255.255.0 is 11111111 11111111 11111111 00000000 in binary. The ones correspond to the network or subnet part of the IP address so the network address 192.168.1.0 with subnet mask 255.255.255.0 is written 192.168.1.0/24 in CIDR form.

(web-based user interface of the ePipe). On the IP Network Address Translation (NAT) Manager page, a new NAT rule set may be created or, if one or more rule sets already exist, the NAT rule set(s) may be modified.

Click on the *Add Rule* button to add a traffic type from the list (see Figure 2). Select a traffic type and enter the LAN or internal IP address of the server that this traffic type is to be redirected to, then click *Next*. There are a limited number of traffic types in the drop down list, so if the traffic type you require isn't in the list, the NAT rule will need to be added through the CLI (Command Line Interface; see below).

All of the rules in the rule set should now be visible on the screen. Add rules in the above manner until all of the required rules have been added, then click *Configure* to save the changes.

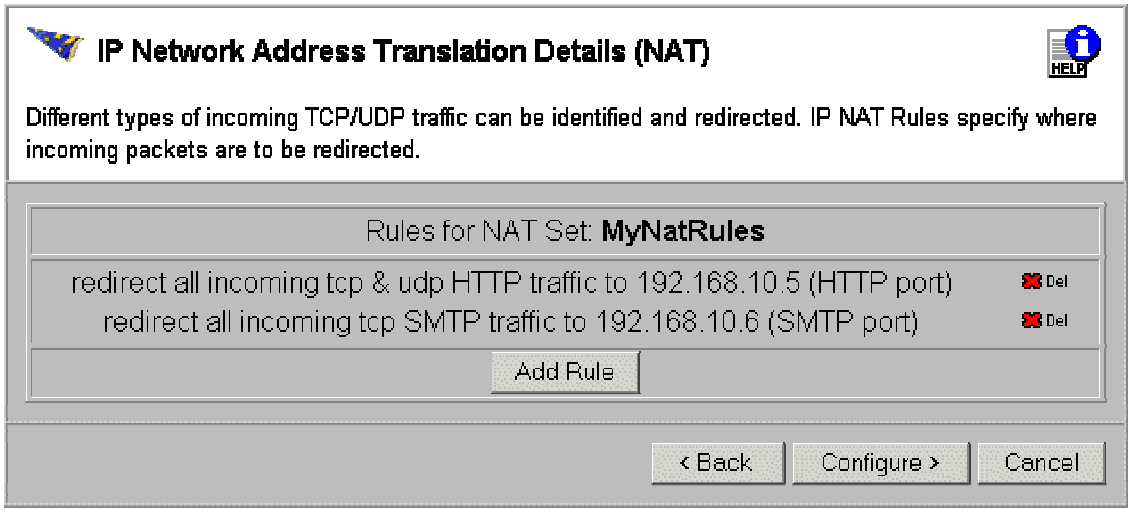

Figure 2 – IP Network Address Translation Rules from Figure 1

If some of the traffic types weren't present in the drop down menu, you will now have to use the CLI to enter the additional rules. To do this, use the console port or telnet to the ePipe and log in as the root user. Type in the command:

SHOW INTERNET NAT ALL

Current NAT rule sets will be displayed. It is important that you take note of the number of the last rule in the NAT rule set being configured, as new NAT rules entered via the CLI will need to be numbered sequentially following the last number.

To enter a NAT rule, type in the following command (broken over two lines for readability):

CHANGE INTERNET NAT "*ruleset_name*" ENTRY *x* RULE
"rdr %s 0.0.0.0/0 port *y* -> 192.168.10.10 port *z* tcpudp"

Where:

- *ruleset_name* is the name of the NAT rule set. These are listed at the top of each rule set in the SHOW INTERNET NAT ALL command.
- *x* is the number of the NAT entry. If the NAT rule set already had 3 rules then this should be number 4 to create the fourth rule.
- %s is a parameter that matches the interface that the NAT rules are being applied to. Leave this as is.
- 0.0.0.0/0 matches any source IP address.
- *y* and *z* are the TCP/UDP port numbers. E.g. TCP port 80 for HTTP.
- tcpudp is a code to indicate what type of traffic to match. The types can be "tcpudp", "tcp/udp", "tcp" or "udp". The first two mean either TCP or UDP.

An example rule:

CHANGE INTERNET NAT "My_NAT" ENTRY 5 RULE
"rdr %s 0.0.0.0/0 port 25 -> 192.168.10.10 port 8025 tcpudp"

This rule would redirect packets sent from any IP address with a destination TCP or UDP port number of 25 to host 192.168.10.10 on port 8025. This would only happen on a bundle (group of links in the ePipe) that had the "My_NAT" NAT rule set applied to it. Note that this example redirects the packet to a different destination port number as well as IP address. This can only be done using the CLI.

3.2 Bimap

Bimap rules can currently only be configured using the ePipe CLI. The order of the rules is important and it is possible to have NAT and Bimap rules which conflict with one another. As with entering NAT redirect rules in the CLI, you will need to be aware of the number of the last rule in the rule set to be configured. Once again, use the SHOW INTERNET NAT ALL command to see the currently configured NAT rule sets.

To configure bimap rules, open a console or telnet session and log in as the root user. Type in the following command (broken over 2 lines for readability):

CHANGE INTERNET NAT "*nat_name*" ENTRY *x* RULE
"bimap %s *private_address*/32 -> *public_address*/32"

Where:

- *nat_name* is the name of the NAT rule set.
- *x* is the number of the rule being added to the NAT rule set.
- *private_address* is the LAN IP address of the server that packets are being mapped to.
- *public_address* is the public IP address by which the server will be known on the public network.

NOTE:	As a bimap rule translates <u>all</u> traffic to a specific public address into a specific private address, no TCP or UDP ports numbers are required. Therefore bimap rules will also work with non-TCP/UDP traffic (e.g. ICMP).

Using the network as depicted in Figure 3, HTTP requests would require bimap rules to get to their respective servers. Figure 3 is a diagrammatic explanation of how this would work.

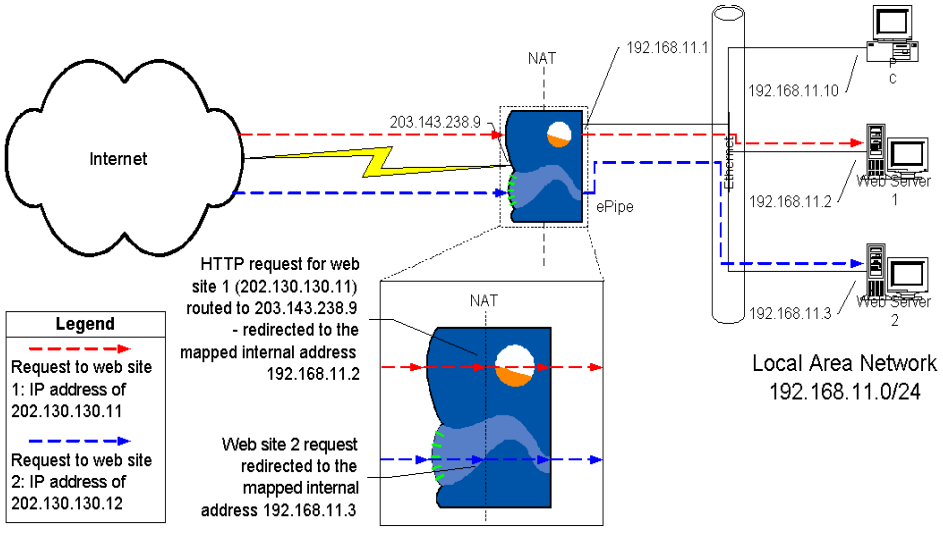

Figure 3 - Physical Network Design with IP Addressing

3.3 Bundles

Bundles are groups of links to the Internet. Each bundle can have a separate NAT rule set and filter applied to it. Once a NAT rule set has been created it will need to be applied to the appropriate bundle. Simply use the ePipe Management Assistant to browse to *Advanced* > *Bundle* and select the bundle to assign the NAT rule set to. You will now be able to edit the bundle and change the *NAT* drop down list to the name of the NAT rule set. If the NAT rule set is not displayed in the drop down list then it has not been created. Click *Configure* to save the configuration.

Once the NAT rule set has been assigned to a bundle, stop and restart the bundle to activate the NAT rules. The easiest way to do this is to browse to *Advanced* > *Summary* and toggle the ON/OFF button off then on.

3.4 Filters

When using NAT to redirect or map packets, the intention is to let them through the ePipe firewall. The filter on the bundle that the NAT rule set is applied to will also need to have the filter on that bundle modified to allow the traffic from the Internet to be redirected or mapped to the internal server. Therefore, the traffic that is being redirected will need to be allowed through the filter using a filter rule that "allows the traffic in". To check or modify the filter, use the ePipe Management Assistant to browse to Advanced > Filter and then select the filter name to view and modify it.

NOTE	Traffic that is inbound to the ePipe from the Internet will be NAT'd before being filtered. Conversely, traffic that is outbound to the Internet will be filtered and then NAT'd.

4 Testing and Troubleshooting

Testing a NAT rule set is fairly simple although it must be remembered that NAT rule sets redirect incoming requests from an external source and, as such, do not redirect internal requests. In other words, the packet needs to arrive on the interface that the NAT rule set is active on.

To test a NAT rule and confirm that it is working, from the external or public network, try to use the redirect that was created. For example, if a HTTP redirect has been implemented, open a web browser and try to browse to the external IP address of the ePipe. This should result in the home page of the web server being displayed. If the ePipe Management Assistant is displayed then the redirect is not working.

If a bimap rule was created, talk to the mapped host on its external IP address using whatever application or protocol the host can be contacted on. Remember to allow this traffic through the filter on the bundle in the ePipe.

The most common problem encountered when implementing a NAT rule set is the rule set has not been applied to the bundle. To check this, browse to the ePipe, then click on *Advanced* > *Bundles*. This will display the Bundle Manager screen. Check that a bundle is configured with the appropriate NAT rule set in the NAT column. This bundle should be the one containing the link to the Internet down which the redirected traffic will be received. Also remember to restart the bundle after it is modified. Changes to a bundle do not become active until the bundle is disabled and then re-enabled.

The NAT table may be viewed by using the ePipe CLI (Command Line Interface) to run the command:

SHOW INTERNET NAT

An example output from this command is shown in Figure 4 below.

```
List of active MAP/Redirect filters:
rdr lxpkt0     0.0.0.0/0 port 80 via default  -> 127.0.0.1 port 8080 tcp
rdr pppoe0 0.0.0.0/0 port 80 -> 192.168.100.141 port 80 tcp/udp
rdr pppoe0 0.0.0.0/0 port 25 -> 192.168.100.141 port 25 tcp
map pppoe0 203.143.238.9/32  -> 203.143.238.9/32
map pppoe0 0.0.0.0/0  -> 203.143.238.9/32  proxy port ftp ftp/tcp
map pppoe0 0.0.0.0/0  -> 203.143.238.9/32  portmap tcp/udp 10000:60000
map pppoe0 0.0.0.0/0  -> 203.143.238.9/32
rdr pppoe1     0.0.0.0/0 port 80 -> 192.168.100.141 port 80 tcp/udp
rdr pppoe1     0.0.0.0/0 port 25 -> 192.168.100.141 port 25 tcp
map pppoe1 144.137.136.120/32  -> 144.137.136.120/32
map pppoe1 0.0.0.0/0  -> 144.137.136.120/32  proxy port ftp ftp/tcp
map pppoe1 0.0.0.0/0  -> 144.137.136.120/32  portmap tcp/udp10000:60000
map pppoe1 0.0.0.0/0  -> 144.137.136.120/32

List of active sessions:
RDR 192.168.100.141   80    <- -> 203.143.238.9   80    [61.9.138.237 10332]
RDR 192.168.100.141   25    <- -> 203.143.238.9   25    [200.47.135.251 3834]
MAP 192.168.100.141   4187  <- -> 203.143.238.9   21834 [64.23.81.137 25]
MAP 192.168.100.111   4183  <- -> 203.143.238.9   21833 [202.101.172.4 80]
RDR 192.168.100.141   21    <- -> 203.143.238.9   21    [212.142.193.158 3713]
MAP 192.168.100.55    1842  <- -> 203.143.238.9   1842  [203.16.234.20 21]
RDR 192.168.100.141   53    <- -> 203.143.238.9   53    [192.216.91.3 53]
MAP 203.143.238.9     2000  <- -> 203.143.238.9   2000  [203.164.99.210 32187]
MAP 203.143.238.9     0     <- -> 203.143.238.9   0     [203.45.128.247 0]
MAP 192.168.100.141   25    <- -> 203.143.238.9   10022 [203.56.233.124 32781]
MAP 192.168.100.141   80    <- -> 203.143.238.9   10018 [212.240.172.111 5207]
```

Figure 4 – Example NAT Table

The device this particular NAPT table came from has two PPPoE (ADSL) Interfaces currently connected. An understanding of a NAPT table will allow rules to be checked and ensure that the rule has been implemented effectively. The output of the above command is broken into two sections, the list of active NAT rules or mappings as defined by NAT Rule Sets and by enabling IP NAT on a link, followed by the list of active sessions currently being NAT'd. Some of the aspects of this output are discussed in Figure 5 below.

```
rdr lxpkt0 0.0.0.0/0 port 80 via default  -> 127.0.0.1 port 8080 tcp
```

The rule above is an internal redirect within ePipe used by I2B.

```
rdr pppoe0 0.0.0.0/0 port 80 -> 192.168.100.141 port 80 tcp/udp
rdr pppoe0 0.0.0.0/0 port 25 -> 192.168.100.141 port 25 tcp
```

These rules are NAT redirection rules showing traffic arriving on the first PPPoE interface (pppoe0), matching any IP address (0.0.0.0/0) and sent to ports 80 (either TCP or UDP) or 25 (TCP) is to be redirected to 192.168.100.141 on the same port numbers.

```
map pppoe0 203.143.238.9/32  -> 203.143.238.9/32
```

The above rule is present to prevent packets with a source address of 203.143.238.9 from being mapped by NAT.

```
map pppoe0 0.0.0.0/0  -> 203.143.238.9/32  proxy port ftp ftp/tcp
map pppoe0 0.0.0.0/0  -> 203.143.238.9/32  portmap tcp/udp 10000:60000
map pppoe0 0.0.0.0/0  -> 203.143.238.9/32
```

The next three rules are added when IP NAT is enabled on a link. The first rule maps the IP address of FTP traffic to the links IP address for transmission. FTP needs a special rule because it creates two TCP connections (one for control and one for data) and the source and destination TCP port numbers of the data channel are often unpredictable (due to FTPs active/passive nature). The second rule provides for the mapping of both the IP address and TCP/UDP port number and is referred to as NAPT (Network Address and Port Translation), which allows the ePipe to map many outbound sessions through a single external IP address. The final rule simply maps all other traffic through pppoe0 to that link's IP address.

```
rdr pppoe1 0.0.0.0/0 port 80 -> 192.168.100.141 port 80 tcp/udp
rdr pppoe1 0.0.0.0/0 port 25 -> 192.168.100.141 port 25 tcp
map pppoe1 144.137.136.120/32  -> 144.137.136.120/32
map pppoe1 0.0.0.0/0  -> 144.137.136.120/32  proxy port ftp ftp/tcp
map pppoe1 0.0.0.0/0  -> 144.137.136.120/32  portmap tcp/udp 10000:60000
```

```
map pppoe1 0.0.0.0/0  -> 144.137.136.120/32
```

The above entries are the same as those explained above, except they are for the second PPPoE interface (pppoe1).

The next section shows active mapped sessions based on traffic being routed through the ePipe. Some examples with explanations are listed below.

```
RDR 192.168.100.141   80    <- -> 203.143.238.9   80    [61.9.138.237 10332]
```

This shows a request from 61.9.138.237 port 10332 to 203.143.238.9 port 80 (HTTP) that was redirected to the internal address 192.168.100.141 on port 80. In other words the packet had its destination address mapped from 203.143.238.9 to 192.168.100.141 inbound and the opposite outbound.

```
RDR 192.168.100.141   25    <- -> 203.143.238.9   25    [200.47.135.251 3834]
```

This shows email being redirected to the mail server (192.168.100.141) through the ePipe (on pppoe0 - 203.143.238.9) from 64.23.81.137.

```
MAP 192.168.100.111   4183  <- -> 203.143.238.9   21833 [202.101.172.4 80]
```

This entry shows an outgoing TCP connection to 202.101.172.4 on port 80 (HTTP) from the private address 192.168.100.111 being mapped to 203.143.238.9, the IP address of the link the packet is transmitted on.

```
MAP 192.168.100.55    1842  <- -> 203.143.238.9   1842  [203.16.234.20 21]
```

This entry shows an FTP request from a PC on the LAN (192.168.100.55) to a server of the Internet (203.16.234.20).

```
MAP 203.143.238.9     0     <- -> 203.143.238.9   0     [203.45.128.247 0]
```

This entry shows a non TCP or UDP packet going through NAT, thus the port numbers of 0. This could be due to a ping (ICMP echo/echo reply) or some other protocol.

Figure 5 – Example NAT Table explanation

5 Conclusion

NAT has several options for translating IP addresses that allow packets to be redirected to specific internal hosts based on service or protocol as well as IP address. It must be understood, however, that doing so may compromise the security of the network. Using a redirect or bimap rule in NAT essentially sends traffic "as is" to an internal host and that host is then responsible for whatever happens. Thus, while NAT redirection in particular is a useful way of being able to host a web server, it can never replace a true DMZ.

Bimap allows all traffic sent to a specific IP address to be forwarded to some other address. This is useful if all traffic is to be checked by some 3rd party security device or where a virtual DMZ is required.

For further information, contact ePipe by sending email to support@ml-ip.com, quoting the name of this paper in the subject header.

Document Number: AN-EP-004 First Edition: January, 2002
Keywords: Network Address Translation NAT PAT NAPT Firewall Redirection Bimap This revision: September, 2002