Performance of NAT64 versus NAT44 in the Context of IPv6 Migration

Kenneth Joachim O. Llanto and William Emmanuel S. Yu

Abstract—**IPv4 address exhaustion is now upon us. Yet, we still live with uncertainty on what to do with the situation. This paper aims to pave the way for Internet Protocol version 6 (IPv6) migration in the Philippines and similar countries using Network Address Translation (NAT) technology. This is achieved by showing service providers that there is already a clear and easy path to migration by using a similar NAT technology, NAT Ipv6 to IPv4 (NAT64). This was done through experimental and live laboratory network tests using NAT44 and NAT64 technologies to show that NAT64s performance is comparable to current NAT44 networks. The setup was implemented using Linux based software and systems to be cost-effective in the translation. Throughput and Round Trip Time (RTT) were used to compare the experimental networks. Results showed that NAT64 has a slight improvement over NAT44 networks in terms of throughput. The Live laboratory networks focused on RTT, CPU and Memory utilization. These results validated the slight improvement brought about by NAT64 in terms of RTT and showed that NAT64 has a lower CPU and Memory utilization. In conclusion, NAT64 as a transition mechanism can be implemented easily and clearly has benefits for current NAT44 networks. Full IPv6 implementation is still in its early stages and more research will be beneficial to the future Internet. This makes prolongation strategies like these more important.**

Index Terms—**IPv6 migration, NAT64, IPv6 transition mechanisms, IPv6 translation strategies**

I. Introduction

IPv4 exhaustion is now at hand. IANA has just recently allocated the last five /8 blocks of unassigned IPv4 addresses to the world's five RIR. This is where IPv6, the next generation protocol designed to replace IPv4, comes into play. The question now is are we ready for it? Is it safe to say that we are prepared for the future? Or are we still hiding inside our NAT44 networks? This research came about because of the need to provide a clean path to IPv6 migration. This paper aims to encourage NAT44 Networks to take the leap or even the small step of preparing and transitioning to IPv6 by presenting the idea of using NAT64 as a suitable replacement to current NAT44 networks. Now is the time to seize the future, the time for the Next Generation IP Protocol, IPv6 migration.

IPv6 is here mainly to solve the problem of IPv4 exhaustion. But, do we know what IPv6 is? Are we ready to replace the dotted decimal IP addresses like 192.168.254.254 with something like 2001:df0:23b::0? To the normal user it would just seem that it's a different addressing scheme. The complexities and constraints that come with IPv6 are concealed from them. This burden will fall upon the network engineers and administrators. But the rewards are even better, aside from a virtually limitless address space (32-bit address to 128 bit address), it has the following features as an improvement from IPv4: a New header format designed to minimize overhead, an efficient and hierarchical addressing and routing infrastructure with the use of IPv6 global addresses, stateless and stateful address configuration for easy host configuration, support for IPSec, better support for prioritized delivery with the use of flow labels in the IPv6 header and a new protocol for neighboring node interaction and extensibility for new features by adding extension headers [4]. With all these new features and a huge amount of IP addresses IPv6 will be the internet protocol for years to come.

II. Methodology

A. Procedures

The following experimental network set-ups were created: NAT44, native IPv6 and NAT64. To evaluate the experimental networks ping and apachebench were used. Ping[3] uses an Internet control message protocol(ICMP) to send datagram packets to a destination computer and waits for echo responses. Ping was used to check for network route connectivity and packet RTTs. Apachebench[1] is a Hypertext Transfer Protocol (HTTP) server benchmarking tool used to test the experimental networks deployed for the research. This was used to compare data in terms of throughput.

For the next phase of the research, RTT, CPU utilization and Available memory measurements were taken from the existing NAT44 network. These measurements were taken using ping and SNMP [15] applications and commands. The data gathered were displayed in easily viewable graphs using MRTG [12]. After data tabulation and validation, the NAT44 network was migrated to the NAT64 network. During this process the researcher noted the steps, costs incurred, problems encountered and challenges faced during this phase. The information obtained during this phase can be useful considerations for other NAT44 networks with the intention of using NAT64. This is the vital step that hopes to open doors and pave the way to IPv6 migration in the Philippines. After completion of the NAT64 network, measurements were gathered from this network. These data was then compared to the data taken gathered from the NAT44 network.

B. Software Tools and Daemons

This section provides a discussion of the software tools and daemons used for the experimental and laboratory networks.

Manuscript received December 02, 2011; revised January 12, 2012. This work was supported in part by the Department of Science and Technology under Engineering Research and Development for Technology Scholarship grant in coordination with the Ateneo de Manila University, Philippines.

K. O. Llanto is a M.S. student of the Department of Information Systems and Computer Studies of the Ateneo de Manila University (e-mail:kennllanto@yahoo.com.ph).

W. S. Yu is with the Department of Information Systems and Computer Studies of the Ateneo de Manila University(e-mail:wyu@ateneo.edu).

1) IPtables: IPtables is the command line program used in configuring packet filtering rules [13]. This provided the masquerading techniques for NAT44 and performed packet filtering services for NAT44 and the IPv4 network of the NAT64 set-up.

2) Berkeley Internet Name Daemon (BIND): BIND is open source software that implements the DNS protocols for the Internet [2]. The purpose of BIND was to provide DNS service to the NAT44, IPv6 and NAT64 clients when accessing the webserver.

3) Router Advertisement Daemon (RADVD): RADVD is run by Linux or Berkeley Software Distribution (BSD) systems acting as IPv6 routers. It sends Router Advertisement messages to a local Ethernet Local Area Network (LAN) periodically and when requested by a node sending a Router Solicitation message [14]. RADVD provided the IPv6 clients auto-configured IPv6 addresses.

4) Trick Or Treat Daemon (TOTD): TOTD is a small Domain Name System (DNS) proxy nameserver that supports IPv6 only hosts/networks that communicate with the IPv4 world using some translation mechanism [7]. This was used to provide IPv6 addresses for external IPv4 servers not having IPv6 addresses so that it can accessed by the IPv6 clients.

5) TAYGA: TAYGA is an out-of-kernel stateless NAT64 implementation for Linux that uses the TUNnnel (TUN) driver to exchange IPv4 and IPv6 packets with the kernel. It is intended to provide production-quality NAT64 service for networks where dedicated NAT64 hardware would be overkill [10]. This provided the NAT64 service for the IPv6 network so that access to the IPv4 webserver was made possible from the IPv6 network. Being out of kernel relative to IPtables that provide NAT44 functionality will probably have consequences on performance.

6) Apachebench: Apachebench is a tool used for benchmarking an Apache Hypertext Transfer Protocol (HTTP) server. It is designed to provide information on how the Apache server performs. It allows simulation of varying amounts of requests and concurrency levels [1].

7) Simple Network Management Protocol (SNMP): SNMP is the standard operations and maintenance protocol for the Internet [15]. It will be used to monitor the traffic that passes through the interfaces of the server and other metrics such as memory and Central Processing Unit (CPU) utilization of the router.

8) Multi Router Traffic Grapher (MRTG): MRTG is a tool to monitor the traffic load on network links [12]. MRTG generates html pages being used to monitor the results of SNMP and convert them to graphical information so that data can be easily interpreted.

C. Experimental Networks

For the Experimental networks, ping/ping6 was used to send 21 packets to the webserver to test for connectivity and RTT while the apachebench tool was used to test using 1000, 2000 and 3000 requests along 10, 100, 200 and 300 concurrency levels. This was chosen based on the capacity of the webserver used during the experiments. The Figure 1 shows the general set-up for the experimental networks.

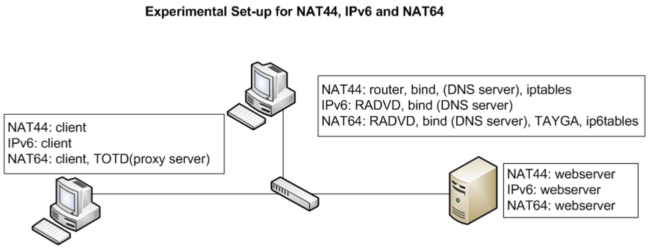

Experimental Set-up for NAT44, IPv6 and NAT64
NAT44: router, bind, (DNS server), iptables
IPv6: RADVD, bind (DNS server)
NAT64: RADVD, bind (DNS server), TAYGA, ip6tables
NAT44: client
IPv6: client
NAT64: client, TOTD(proxy server)
NAT44: webserver
IPv6: webserver
NAT64: webserver

Fig. 1. Experimental Set-up for NAT44, native IPv6 and NAT64

D. NAT44

In the experimental NAT44 network, IPtables were used to perform NAT and routing. BIND [2] was used to perform the Domain Name System (DNS) service to provide access to the webserver. Ping was used to check for successful connectivity between the devices and RTT. Apachebench was used to evaluate the performance of the network in terms of Throughput. Figure 2 shows the logical diagram of the NAT44 experimental network.

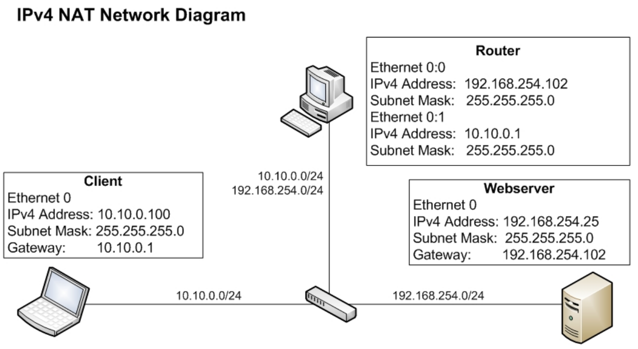


Fig. 2. Logical Diagram of the NAT44 experimental network

In this set-up, IPv4 clients were provided with non-routable IPv4 addresses inside their local networks. These clients use NAT44 as a means to connect to the IPv4 internet via the gateway's single routable address. Figure 3 shows the flow diagram of the NAT44 experimental network.

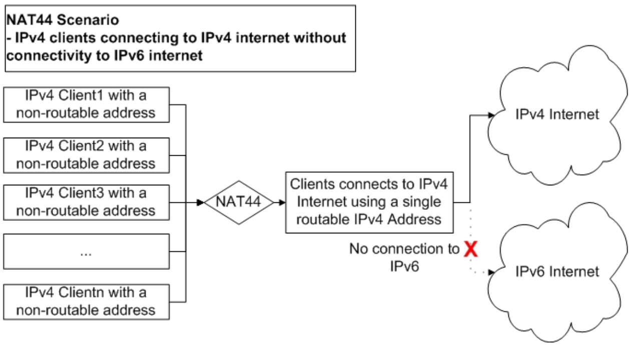

NAT44 Scenario
- IPv4 clients connecting to IPv4 internet without connectivity to IPv6 internet
IPv4 Client1 with a non-routable address
IPv4 Client2 with a non-routable address
IPv4 Client3 with a non-routable address
...
IPv4 Clientn with a non-routable address
NAT44
Clients connects to IPv4 Internet using a single routable IPv4 Address
IPv4 Internet
No connection to IPv6
IPv6 Internet

Fig. 3. Flow Diagram of the NAT44 experimental network

E. Native IPv6

Router Advertisement Daemon (RADVD) [14] was used to provide IPv6 addresses to the clients. Since all clients would fall under the same network by using RADVD, routing was no longer needed. BIND was still used to perform the DNS service to access the webserver. Ping was used to check for successful connectivity between the devices and RTT. Apachebench was used to evaluate the performance of the network. Figure 4 shows the logical diagram of the Native IPv6 experimental network.

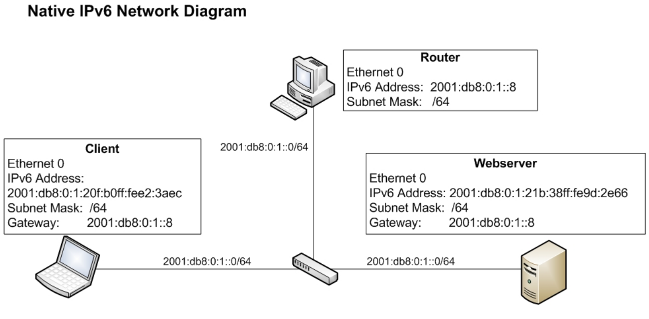


Fig. 4. Logical Diagram of the Native IPv6 experimental network

In this set-up, native IPv6 clients were issued routable IPv6 addresses inside their local networks. These clients are able to directly connect to IPv6 Internet. This is the ideal set-up once migration to IPv6 has been completed. Figure 5 shows the flow diagram of the native IPv6 experimental network.

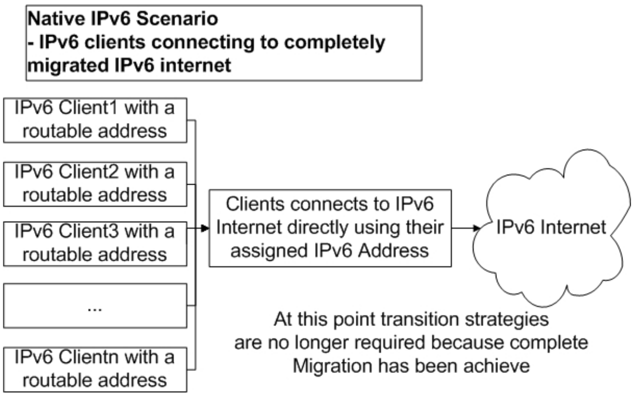

Native IPv6 Scenario
- IPv6 clients connecting to completely migrated IPv6 internet
IPv6 Client1 with a routable address
IPv6 Client2 with a routable address
IPv6 Client3 with a routable address
...
IPv6 Clientn with a routable address
Clients connects to IPv6 Internet directly using their assigned IPv6 Address
IPv6 Internet
At this point transition strategies are no longer required because complete Migration has been achieve

Fig. 5. Flow Diagram of the Native IPv6 experimental network

1) NAT64: Setting-up IPv6 NAT64 requires IPtables, RADVD, TOTD, BIND, TAYGA to be used. The IPtables were used to connect the IPv4 external network to the IP addresses to be used by TAYGA, the NAT64 daemon. RADVD was used to provide IPv6 addresses for the IPv6 clients. The purpose of TOTD is to serve as a proxy server performing DNS64 so that IPv4 servers can be accessed by the IPv6 network. BIND provided the actual DNS service. To test the experimental set-up apachebench was used to access an external IPv4 webserver and ping was also used to test RTT and connectivity between these networks. The Figure 6 shows the logical diagram of the IPv6 NAT64 experimental network. This design was used to allow direct IPv6 routing on the event that the destination supports IPv6. In this case, the use of NAT64 decreases as IPv6 networks increases.

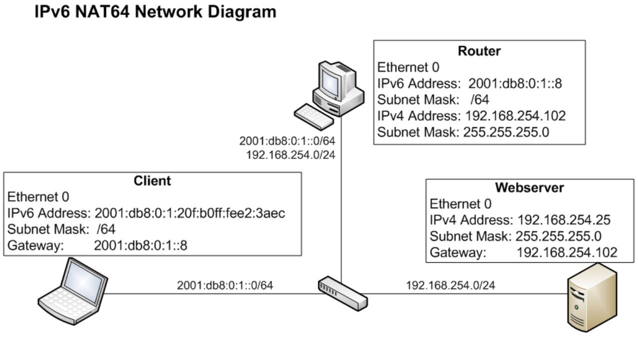

IPv6 NAT64 Network Diagram
Router
Ethernet 0
IPv6 Address: 2001:db8:0:1::8
Subnet Mask: /64
IPv4 Address: 192.168.254.102
Subnet Mask: 255.255.255.0
2001:db8:0:1::0/64
192.168.254.0/24
Client
Ethernet 0
IPv6 Address: 2001:db8:0:1:20f:b0ff:fee2:3aec
Subnet Mask: /64
Gateway: 2001:db8:0:1::8
Webserver
Ethernet 0
IPv4 Address: 192.168.254.25
Subnet Mask: 255.255.255.0
Gateway: 192.168.254.102
2001:db8:0:1::0/64
192.168.254.0/24

Fig. 6. Logical Diagram of the IPv6 NAT64 experimental network

In this set-up, IPv6 clients were issued routable IPv6 addresses inside their local networks. These clients are able to directly connect to IPv6 Internet using their routable address. They are also able to connect to the IPv4 internet using NAT64. This set-up prepares networks for IPv6 migration and allows connection to the IPv4 and IPv6 internet. Figure 7 shows the flow diagram of the IPv6 NAT64 experimental network.

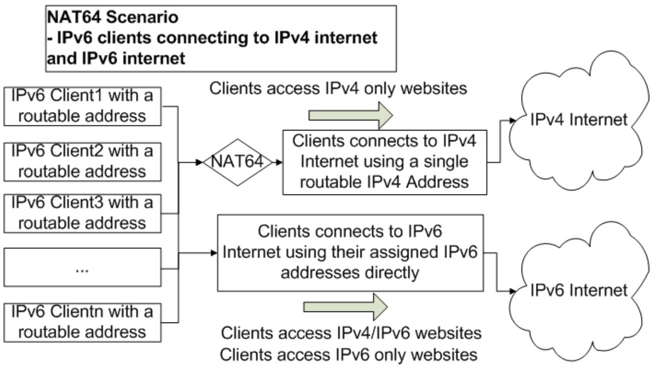

NAT64 Scenario
- IPv6 clients connecting to IPv4 internet and IPv6 internet
IPv6 Client1 with a routable address
IPv6 Client2 with a routable address
IPv6 Client3 with a routable address
...
IPv6 Clientn with a routable address
Clients access IPv4 only websites
NAT64
Clients connects to IPv4 Internet using a single routable IPv4 Address
IPv4 Internet
Clients connects to IPv6 Internet using their assigned IPv6 addresses directly
IPv6 Internet
Clients access IPv4/IPv6 websites
Clients access IPv6 only websites

Fig. 7. Flow Diagram of the NAT64 experimental network

F. Laboratory Networks

For the laboratory networks, ping and the mrtg-ping probe will be used to send packets to servers outside the local area network to test for connectivity and RTT. SNMP data will be gathered to check the CPU utilization and Free Memory. MRTG will graph the SNMP and RTT results.

1) NAT44 Laboratory Network: Setting-up the Ateneo NAT44 network has a similar requirement for the experimental NAT64 network and required IPtables, Ateneo's local DNS server and a Dynamic Host Configuration Protocol (DHCP) server. IPtables was used to connect the IPv4 external network to the internal IPv4 Network using NAT44. The local DNS server provides DNS resolution for the laboratory clients. DHCP was be used to provide dynamic IPv4 addresses to the clients. Ping was also used to test connectivity between these networks. SNMP was used to monitor data traffic and MRTG was used to display the gathered data. No additional configuration was done on the clients as they are all IPv4 enabled. Figure 8 shows a logical set-up of the IPv6 NAT44 network.

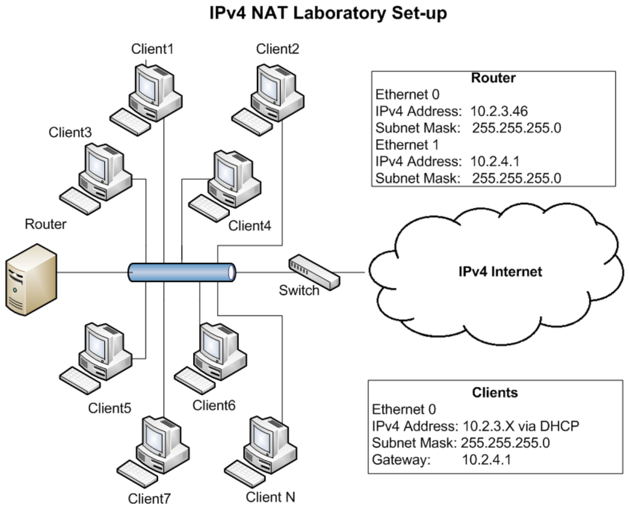


Fig. 8. Laboratory Set-up for NAT44

2) IPv6 NAT64: Setting-up the Ateneo IPv6 NAT64 network required IPtables, RADVD, TOTD, BIND, TAYGA to be used. IPtables was used to connect the IPv4 external network to the IP addresses to be used by TAYGA, the NAT64 daemon. RADVD was used to provide IPv6 addresses for the IPv6 clients. TOTD served as a proxy server performing DNS64 so that IPv4 servers can be accessed by the IPv6 network. An actual DNS was used to provide the DNS service. TOTD intercepts the DNS response to add a corresponding IPv6 address if one is not present. TAYGA served as the NAT64 daemon. The main difference this has on the experimental network is that it has actual connection to the IPv4 Internet. In this matter, BIND can be replaced by actual DNS servers instead. DHCPv6 was also installed to give clients the IPv6 DNS servers. To test the experimental set-up ping was used to test connectivity between these networks. SNMP was used to monitor data traffic and MRTG was used to display the gathered data. Linux clients already come with IPv6 support but required installation of Recursive DNS Server Daemon(rdnssd) [16] to be able to obtain the correct IPv6 DNS server. For the windows XP clients, the IPv6 protocol must first be enabled as to it is disabled by default. Additionally, dibbler [11] was installed to store an IPv6 DNS server to the clients. Also note that the Windows XP clients require DNS resolution using IPv4, thus IPv4 DNS must be done to allow DNS service. This would not hinder IPv6 connectivity because Windows gives priority to IPv6 DNS records over IPv4 records when both records are returned by the DNS server. Figure 9 shows a logical set-up of the IPv6 NAT64 network.

III. Results

A. Experimental Testbed Results

This section is further subdivided into three parts: RTT results, Throughput results and t-Test results

1) RTT Results: This section shows the RTT results using ping and ping6 commands for the NAT44, native IPv6 and NAT64 experimental networks for 21 packets.

a. RTT Summary : Figure 10 shows a summary of the results of the ping command. It can be seen that the overall performance of IPv6 is better in all aspects compared to NAT44 and NAT64. This gives as a preview of our future networks when IPv6 migration has been completely implemented.

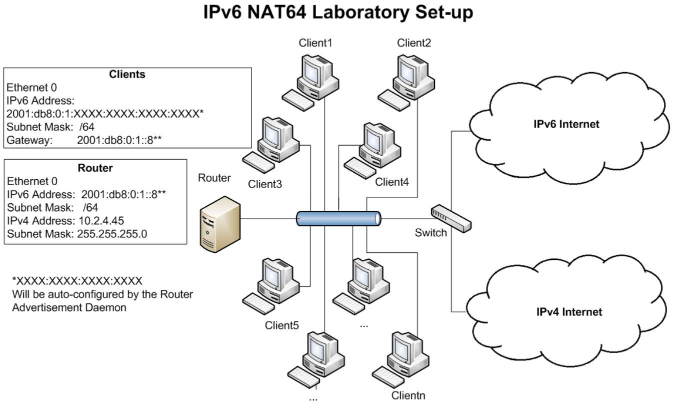


Fig. 9. Laboratory Set-up for NAT64

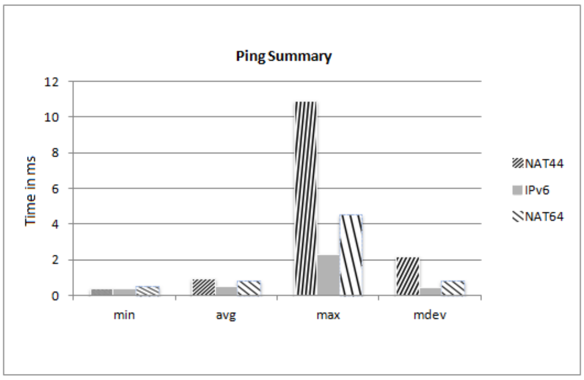


Fig. 10. Summary of RTT results

2) Throughput Results: This section shows the throughput results using the apachebench tool when accessing a 173,225 byte webpage. The experiments were done for 1000, 2000 and 3000 requests. Each of these was done for concurrency levels 10, 100, 200 and 300. The results for Total time, Total Bytes transferred, Successful keep alive packets, requests per second, time per requests and transfer rates are shown in the following subsections. Since the purpose of the paper is to evaluate the NAT64, the results discussed are all in comparison with NAT64.

1) NAT64 vs. NAT44 This section shows the throughput results when NAT64 is compared with NAT44
    a) Total Time Difference : The time difference between NAT64 and NAT44 are almost insignificant aside from 2 spikes of NAT64 at 1000 requests with 300 concurrency and 3000 requests with 200 concurrency. It can also be noted that at 2000 requests NAT64 completes all the requests in a shorter amount of time for all concurrency levels.
    b) Total Bytes Transferred Difference : The difference in total bytes transferred between NAT64 and NAT44 are almost insignificant. It can also be seen for NAT64 at 1000 requests with 300 concurrency and 3000 requests with 200 concurrency that there is a large difference in the total bytes transferred by NAT64.

This is the probable reason why there are spikes at the total time of NAT64 at these levels.

c) Successful Keep Alive Packets Difference : The difference in successful keep alive packets between NAT64 and NAT44 are almost insignificant especially for 1000 requests. It can also be seen for NAT64 at 1000 requests with 300 concurrency and 3000 requests having 200 concurrency that there is a large difference in the number of successful keep alive packets by NAT64. This is the probable explanation why there is a huge difference in the total bytes for NAT64 at these levels.

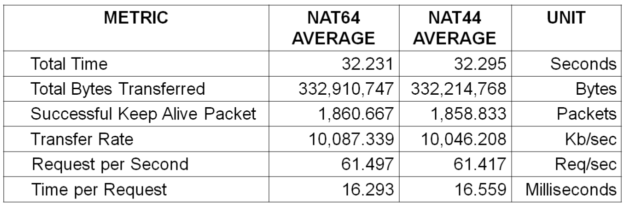
d) Time per Request Difference : Aside from 1000 request with concurrency level of 10, the performance of NAT64 and NAT44 are at par with each other.

e) Request per Second Difference : For most cases, the difference is within 1 second between NAT64 and NAT44. But for almost all instances NAT64 performance is better except for 1000 requests with 300 concurrency level and 3000 request with 200 concurrency level. This probably caused the huge difference in the total bytes for NAT64 at these levels.

f) Transfer Rate Difference : For almost all instances below NAT64 performs better than NAT44, but this performance difference becomes almost insignificant at the 300 concurrency level.

g) NAT64 vs. NAT44 Throughput Summary : The table I below shows the summary of the throughput results between NAT64 and NAT44. It can be seen that there is only a very small difference in the values of their averages.

TABLE I
NAT64 VS. NAT44 THROUGHPUT SUMMARY



METRIC	NAT64 AVERAGE	NAT44 AVERAGE	UNIT
Total Time	32.231	32.295	Seconds
Total Bytes Transferred	332,910,747	332,214,768	Bytes
Successful Keep Alive Packet	1,860.667	1,858.833	Packets
Transfer Rate	10,087.339	10,046.208	Kb/sec
Request per Second	61.497	61.417	Req/sec
Time per Request	16.293	16.559	Milliseconds

2) NAT64 vs. Native IPv6 This section discusses the throughput results when IPv6 NAT64 is compared with Native IPv6

a) Total Time Difference : For the time difference between NAT64 and IPv6, IPv6 performs better for all instances. It can also be noted that the total time difference increases with the number of requests.

b) Total Bytes Transferred Difference : The difference in total bytes transferred between NAT64 and NAT44 is almost insignificant for concurrency levels 10 and 100. For higher concurrency levels there are fluctuations between the total bytes transferred of NAT64 and native IPv6.

c) Successful Keep Alive Packets Difference : The difference in successful keep alive packets between NAT64 and native IPv6 is almost insignificant for concurrency levels 10 and 100. Similar to the total bytes transferred the number of successful keep alive packets fluctuates for higher concurrency levels.

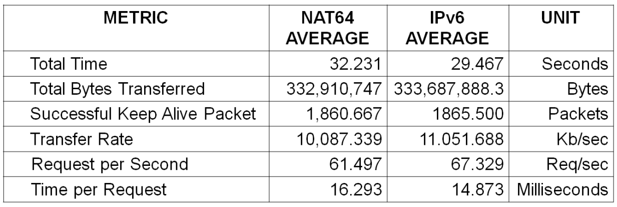
d) Time per Request Difference : For all instances below, it can be seen that the performance of IPv6 is better compared to NAT64.

e) Request per Second Difference : For all instances in the figure below, it can be seen that the IPv6 performance is a lot better than NAT64.

f) Transfer Rate Difference : The IPv6 transfer rate is better for all concurrency levels and number of requests. The value of the difference seems to stabilize close to 950 kbps at different concurrency levels.

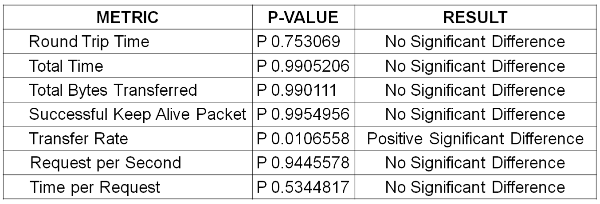
g) NAT64 vs. IPv6 Throughput Summary : The table II below shows the summary of the throughput results between NAT64 and IPv6. It can be seen that for all instances native IPv6 performs better compared to NAT64.

TABLE II
NAT64 VS. NATIVE IPV6 THROUGHPUT SUMMARY



METRIC	NAT64 AVERAGE	IPv6 AVERAGE	UNIT
Total Time	32.231	29.467	Seconds
Total Bytes Transferred	332,910,747	333,687,888.3	Bytes
Successful Keep Alive Packet	1,860.667	1865.500	Packets
Transfer Rate	10,087.339	11.051.688	Kb/sec
Request per Second	61.497	67.329	Req/sec
Time per Request	16.293	14.873	Milliseconds

TABLE III
SUMMARY OF THE EXPERIMENTAL T-TEST RESULTS FOR NAT64 VS. NAT44



METRIC	P-VALUE	RESULT
Round Trip Time	P 0.753069	No Significant Difference
Total Time	P 0.9905206	No Significant Difference
Total Bytes Transferred	P 0.990111	No Significant Difference
Successful Keep Alive Packet	P 0.9954956	No Significant Difference
Transfer Rate	P 0.0106558	Positive Significant Difference
Request per Second	P 0.9445578	No Significant Difference
Time per Request	P 0.5344817	No Significant Difference

3) Summary of t-Test Results: The Table III shows that for Round Trip Time, Total Time, Total Bytes Transferred, Successful Keep Alive Packets, Request per Second and Time per Request there is no significant difference for those metrics. Transfer Rate on the other hand showed a positive significant difference wherein NAT64 performed better by an average of 42 kb/s.

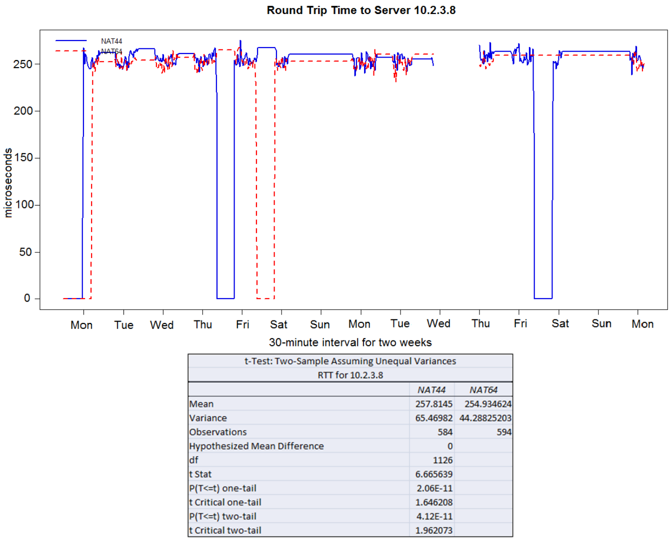
B. Laboratory Results

This section shows the results of the live laboratory testing done for more than 30 computers from July 4, 2011 to July 18, 2011 for NAT44 and August 8, 2011 to August 22, 2011 for NAT64 at the F204 network laboratory in the Faura building of the Ateneo de Manila University. The data gathered were in terms of RTT, CPU Utilization and available memory. Measurements were taken using SNMP and ping commands and displayed using MRTG.

1) RTT: This section shows the summary of the RTT results during the two (2) 15-day periods wherein the NAT44 and NAT64 networks were measured placed side-by-side. RTT was tested on two Ateneo servers, 10.2.3.8 server a server within the WAN of the router and the 10.0.1.254, the Ateneo router at the edge of the Ateneo network.

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

a. RTT for 10.2.3.8 Graph 11 shows the RTT of ping packets originating for our NAT gateway located at F204 during the periods of July 4 to July 18 where NAT44 traffic was measured and August 8 to August 22 where NAT64 traffic was measured. Additionally, the t-Test results for the NAT44 and NAT64 experiments were included. Note that there was an abnormally in the behavior of RTT during Wednesday to Thursday during the second week of the test, the data both from NAT44 and NAT64 were removed along with the zero values to provide a more accurate t-test result. The t-test shows that during the time of the testing RTT is a little better for NAT64 compared to NAT44. It can also be noted that this closely matches the results of the experimental networks.



t-Test: Two-Sample Assuming Unequal Variances
RTT for 10.2.3.8

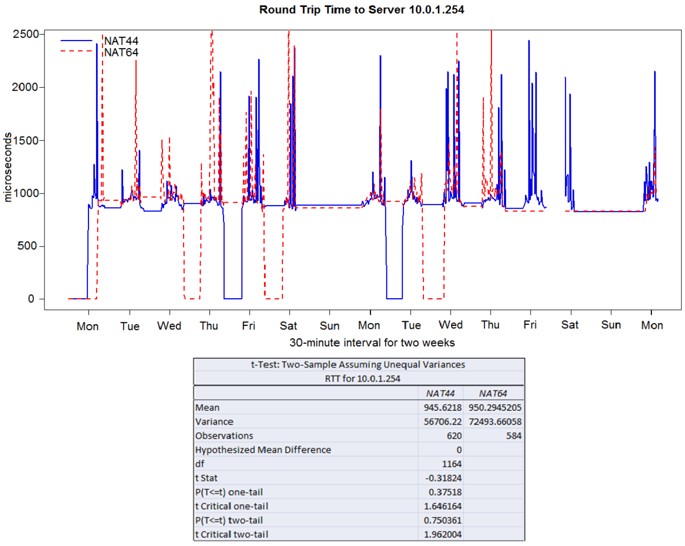
	NAT44	NAT64
Mean	257.8145	254.934624
Variance	65.46982	44.28825203
Observations	584	594
Hypothesized Mean Difference	0	
df	1126	
t Stat	6.665639	
P(T<=t) one-tail	2.06E-11	
t Critical one-tail	1.646208	
P(T<=t) two-tail	4.12E-11	
t Critical two-tail	1.962073	

Fig. 11. NAT44 and NAT64 RTT for server 10.2.3.8 in microseconds

b. RTT for 10.0.1.254 Graph 12 shows the RTT of ping packets originating for our NAT gateway located at F204 during the periods of July 4 to July 18 where NAT44 traffic was measured and August 8 to August 22 where NAT64 traffic was measured. Additionally, the t-Test the results for the NAT44 and NAT64 experiments were included. Note that there was an abnormality in the behavior of RTT during Friday to Saturday during the second week of the test, the data both from NAT44 and NAT64 were removed along with the zero values to provide a more accurate t-Test result. Although NAT44 performed a little better, the t-Test shows that during the time of the testing RTT there is no significant difference between NAT44 and NAT64 at a p = 0.75 which is below the 0.05 alpha. It can also be noted that this validates the results of the experimental networks which show very little discrepancy between them.

2) CPU Utilization: This section shows the summary of the CPU Utilization box plot and t-Test results during the two(2) 15-day periods wherein the NAT44 and NAT64 networks were measured.

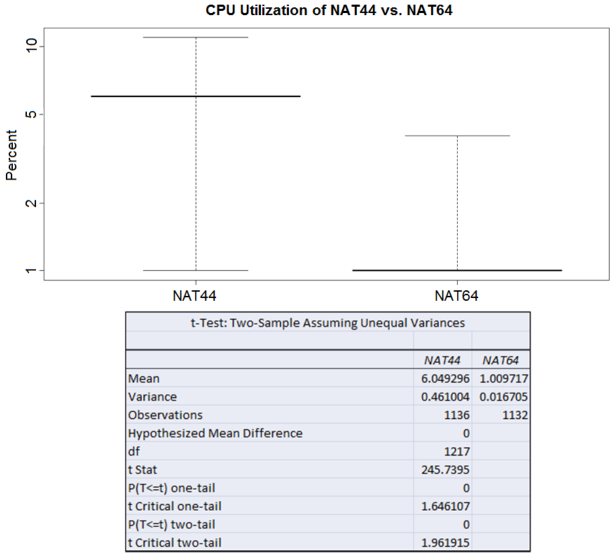
a. CPU Utilization in percentage The box plot in Figure 13 shows the CPU Utilization of the NAT44 and the NAT64 networks. It can be seen that the utilization of the router during NAT64 implementation is very low compared to NAT44 with a median of only 1 percent for NAT64 vs. about 6 percent for NAT44. The probable reason for this is because IPv6 packets are simpler compared to the complex IPv4 packets and that NAT44 performs IP address mappings to ports while the NAT64 implementation does plain IPv6 address to IPv4 address mapping. Also, it could be noteworthy that on the durations of the experiments NAT44 throughput was greater than that of NAT64. Despite this CPU utilization for NAT64 was lower.



t-Test: Two-Sample Assuming Unequal Variances
RTT for 10.0.1.254

	NAT44	NAT64
Mean	945.6218	950.2945205
Variance	56706.22	72493.66058
Observations	620	584
Hypothesized Mean Difference	0	
df	1164	
t Stat	-0.31824	
P(T<=t) one-tail	0.37518	
t Critical one-tail	1.646164	
P(T<=t) two-tail	0.750361	
t Critical two-tail	1.962004	

Fig. 12. NAT44 and NAT64 RTT for server 10.0.1.254 in microseconds



t-Test: Two-Sample Assuming Unequal Variances

	NAT44	NAT64
Mean	6.049296	1.009717
Variance	0.461004	0.016705
Observations	1136	1132
Hypothesized Mean Difference	0	
df	1217	
t Stat	245.7395	
P(T<=t) one-tail	0	
t Critical one-tail	1.646107	
P(T<=t) two-tail	0	
t Critical two-tail	1.961915	

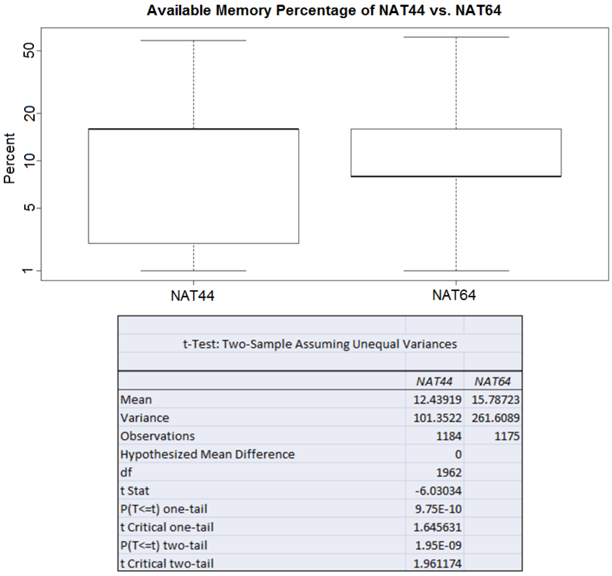
Fig. 13. NAT44 and NAT64 CPU Utilization in percentage

3) Available Memory: This section shows the summary of the available memory in terms of percentage using box plots and the t-Test results during the two (2) 15-day periods wherein the NAT44 and NAT64 networks were measured.

a. Available Memory in percentage The box plot in Figure 14 shows the Available memory in percentage for NAT44 and the NAT64 networks and the t-Test results for the two networks. It can be seen that the median for NAT44 higher

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

compared to NAT64. But in terms of the average, NAT64 has a lower mean compared to NAT44. This higher median for NAT44 was probably caused by the additional daemons installed like DHCPv6, TOTD, RADVD and TAYGA for NAT64. The higher average for NAT64 on the other hand was probably because these daemons only use more memory during start-up and initial requests, once the networks have converged these daemons no longer consume that much memory.



t-Test: Two-Sample Assuming Unequal Variances

	NAT44	NAT64
Mean	12.43919	15.78723
Variance	101.3522	261.6089
Observations	1184	1175
Hypothesized Mean Difference	0	
df	1962	
t Stat	-6.03034	
P(T<=t) one-tail	9.75E-10	
t Critical one-tail	1.645631	
P(T<=t) two-tail	1.95E-09	
t Critical two-tail	1.961174	

Fig. 14. NAT44 and NAT64 Available Memory in percentage

4) Summary of Laboratory Results: This section shows the summary of the live testing results during the two (2) 15-day periods wherein the NAT44 and NAT64 networks were measured.

a. Summary of Laboratory Tests The Table IV shows the summary of the live laboratory results of NAT64 and NAT44. It can be seen that the average RTT for 10.2.3.8 for NAT64 is slightly lower while for the RTT for 10.0.1.254 NAT64 is slightly higher at that time. In terms of CPU utilization NAT64 uses less and available memory the NAT64 performs better compared to NAT44.

TABLE IV
SUMMARY OF LABORATORY TESTS

METRIC	NAT64 AVERAGE	NAT44 AVERAGE	UNIT
Round Trip Time to 10.2.3.8	254.934624	257.8145	microseconds
Round Trip Time to 10.0.1.254	950.2945205	945.6218	microseconds
CPU Utilization	1.009717	6.049296	percent
Available Memory	15.78723	12.43919	percent

b. Summary of t-Test Results The Table V shows the summary of the t-Test results for the live laboratory testing for an Alpha of 0.05. The RTT for 10.2.3.8, CPU utilization and available memory in percentage and bytes all showed a significant difference between NAT64 and NAT44 while RTT for 10.01.254 showed no significant difference between the two protocols.

TABLE V
SUMMARY OF T-TEST RESULTS

METRIC	P-VALUE	RESULT
Round Trip Time to 10.2.3.8	P 4.12E-11	Positive Significant Difference
Round Trip Time to 10.0.1.254	P 0.750361	No Significant Difference
CPU Utilization	P 0	Positive Significant Difference
Available Memory	P 1.95E-09	Positive Significant Difference

IV. SUMMARY AND CONCLUSION

This section discusses insights gained from the behavior of NAT with respect to IPv6 traffic.

A. Summary

In terms of RTT, NAT64 and NAT44 had comparable performance both for the experimental and laboratory results. This shows that no additional latency was caused by using NAT64 as a transition mechanism for NAT44 networks. In terms of throughput, there was no significant difference in the between NAT64 and NAT44 aside from the slight advantage of NAT64 in terms of transfer rate for the experiments.

CPU utilization of NAT64 fared a lot better despite of the higher throughput during that period compared to NAT44, this is probably caused by the simpler IPv6 packet structure compared to the IPv4 packets.

For available memory, NAT44 fared well against NAT64 in terms of the median, this is probably caused by the additional processes and daemons installed in the NAT64 environment. But in terms of the average available memory NAT64 had a higher average available memory, this is probably because the additional daemons and processes only run during start-up and initializations. Ones those requests have been completed and the network has converged NAT64 stabilizes to consume lower amount of memory.

Overall, we were able to validate the RTT results and the CPU results showed that NAT64 needs less CPU resource. However, further validation needs to be done to get conclusive results on throughput for the live network test due to the difficulty in control level.

B. Conclusion

IP networks can be compared using many available technologies. Ping is obviously one of the most important tool for testing connectivity and RTT. Other tools such as apachebench and SNMP are good choices to measure throughput and other metrics. Apachebench is a viable tool for testing test beds mostly because of its capability to simulate multiple concurrent devices and multiple requests. SNMP, along with MRTG is very useful in providing real time graphs for live testing and provides the users with live statistics that are readily available via Hypertext Markup Language (HTML). Migration from NAT44 to NAT64 is an easy process as shown by the research. Planning the interfaces and address allocation is always the first step. Setting-up a Router to provide router advertisements and dhcpv6 is the 2nd step. This is followed by the set-up of NAT64 using TAYGA and DNS64 using TOTD. Finally, configuring the IPtables completes the tasks of migrating your network using NAT64. For these migrations to be smooth, some considerations such as the operating system used by clients may have certain dependencies as shown by Windows XP and Ubuntu. The presence of a connection to IPv6 Internet would also be helpful, must be observed. The results of the research show that NAT64 fared well against NAT44 in terms of RTT for both experimental and live networks.

In terms of throughput, the experimental network showed almost no significant difference between NAT64 and NAT44 with a positive significant difference in transfer rate. For CPU utilization and available memory, NAT64 also showed positive significant difference compared to NAT44.

Clearly, there are benefits of NAT64 as a transition mechanism of choice for open-source based NAT44 networks. Aside from being a low cost solution and faring well in terms of RTT, CPU Utilization and Memory, it is a simple and easy way to turn your existing NAT44 networks to IPv6 and allowing direct routing of IPv6 packets to IPv6 networks. Still, there are certain concerns, implementation on a larger scale and carrier grade set-ups are yet to be explored.

IPv6 is the Internet of the next generation and we must prepare for it by taking the first step towards migration. The benefits of IPv6 far outweigh the detriments that hinder us from using any of the translation techniques. Additionally, aside from the obvious difference in addressing format, this change will not be felt by normal user. Delaying the implementation will not only be harmful to providers, but will prove to be more costly, this is brought about by buying additional IPv4 addresses now treated as a commodity or buying expensive equipments to port your IPv4 networks to IPv6 Internet. Rather than wait for this to happen, the researcher suggests that the first step towards IPv6 should be taken now and that NAT64 can be a good choice for networks with the similar situation in the Philippines or in other Asian countries that are already aided and believing in NAT. Now is the time to seize the future, the time for the Next Generation IP Protocol, IPv6 migration.

V. Recommendations

NAT64 as well as other IPv6 transition technologies is still a very new and rich topic that could be an integral part of the next generation IP networks. To the future researchers interested in pursuing NAT64 as a transition mechanism, following recommendations are suggested:

1) Implementation on a larger scale. A live network on a bigger classroom or an open lab is recommended or even outside the University. This is to perform and learn more about the limitations of larger scale or even carrier grade NAT64 and will push the technology to the limits and further show a better comparison of NAT44 and NAT64 networks using networks that offer higher traffic. Allowing experiments outside the University compounds also allow the removal of filters that prevent certain applications and websites. This also will provide further validation for the use of NAT64 as a viable technology in the larger scale. A good option for this would be to enforce this to an existing facility with NAT44.
2) Longer duration and simultaneous testing. A longer duration of the live testing is highly recommended along with simultaneous testing of NAT44 and NAT64, this is to eliminate the variations in the network traffic trends brought about by examination weeks and class suspensions. A possible opting is to divide hosts within a room or use two different rooms. At the same time, actual throughput is measured.
3) Tests using many IP addresses for the experimental set-up. Implement tests that use various applications, different load conditions and measuring techniques [6], [9] as suggested during the presentation of the initial paper at APNIC32 Conference [5]. This type of testing offers different scenarios for the experiments done. Many different IPs may cause a change in the behavior of NAT44 and NAT64 once they perform different mapping techniques for experimental set-ups. Various applications can be used to determine the compatibility and support of these NAT technologies to commonly used applications and services. Varying kinds of load would have different impacts on the behavior of the networks. Additionally, using PATHLOAD [8] or other means to gather metrics can be used to measure network data to further validate the results of the research. This tries to determine if there are implementation-related issues that are exploited during the tests.

References

[1] apachebench. http://httpd.apache.org/docs/2.0/programs/ab.html Internet. Last accessed Sept. 30, 2011.

[2] Bind. http://www.isc.org/software/bind. Internet. Last accessed Sept. 30, 2011.

[3] ping/ping6. http://linux.die.net/man/8/ping. Internet. Last accessed Sept. 30, 2011.

[4] Microsoft server 2008: Introduction to ip version 6, 2003. http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=21536. Internet. Last accessed Oct. 17, 2011.

[5] Apnic32 conference destination::ipv6. 2011. http://meetings.apnic.net/32. Internet. Last accessed October 2, 2011.

[6] Asia pacific operators forum. 2011. http://meetings.apnic.net/32/program/apops/transcript. Internet. Last accessed September 9, 2011.

[7] Feike W. Dillema. Totd. http://www.dillema.net/software/totd.html. Internet. Last accessed Sept. 30, 2011.

[8] Manish Jain and Constantinos Dovrolis. Pathload: A measurement tool for end-to-end available bandwidth. *In Proceedings of Passive and Active Measurements (PAM) Workshop*, pages 14–25, 2002.

[9] Kenneth Llanto and William Yu. Performance of nat64 versus nat44 in the context of ipv6 migration. August 2011. http://meetings.apnic.net/__data/assets/file/0018/38232/kenneth-APNIC_Presentation_2.21.pdf. Internet. Last accessed October 2, 2011.

[10] Nathan Luchansky. Tayga, 2010. http://www.litech.org/tayga/. Internet. Last accessed September 30, 2011.

[11] Tomasz Mrugalski and Marek Senderski. dibbler. http://klub.com.pl/dhcpv6/. Internet. Last accessed September 30, 2011.

[12] Tobias Oetiker. Multi router traffic grapher (mrtg). *First Published Proceedings of the Twelfth Systems Administration Conference*, pages 141–148, December 1998. http://www.usenix.org/event/lisa98/full_papers/oetiker/oetiker.pdf. Internet. Last accessed Oct. 16, 2011.

[13] Paul "Rusty" Russell. Iptables. http://www.netfilter.org/projects/iptables/index.html. Internet. Last accessed Sept. 30, 2011.

[14] Pekka Savola. Radvd. http://www.litech.org/radvd/. Internet. Last accessed Sept. 30, 2011.

[15] Pierrick SIMIER. http://www.snmplink.org/snmpresource/snmpv3/#2. internet. last accessed september 30, 2011.

[16] Pierre Ynard. rdnssd. http://rdnssd.linkfanel.net/. Internet. Last accessed September 9, 2011.