FACOLTÀ DI INGEGNERIA

CORSO DI LAUREA TRIENNALE IN INGEGNERIA INFORMATICA E DELL'AUTOMAZIONE

Configurazione e verifica di apparati di rete IPv4 ed IPv6 in ambito enterprise

Enterprise network: configuring and verifying network devices for Ipv4 and IPv6 connectivity

Relatore:
Prof. Ennio Gambi

Candidato:
Marco Barbarella

Correlatore:
Ing. Adelmo De Santis

ANNO ACCADEMICO 2022/2023

Indice

Introduzione

La tesi si basa sullo sviluppo di un progetto basato sull'uso di apparati di rete su di una piattaforma software in grado di simulare le funzionalità di questi dispositivi e permette la configurazione e la verifica del funzionamento. Il simulatore in questione è eNSP sviluppato dall'azienda "Huawei Technologies Co., Ltd" e gli apparati in questione sono switch, router, server e pc. Lo scopo del lavoro di tesi è di creare una topologia sul simulatore, connettendo tra loro gli apparati e configurandoli in modo che rispettino le specifiche di progetto.
La tesi è stata scelta dopo aver frequentato il corso di preparazione alla certificazione Huawei HCIA Routing e Switching, nel quale si sono approfonditi gli argomenti principali di Networking (in particolar modo routing e switching). Gli argomenti principali affrontati durante lo sviluppo del progetto sono essenzialmente gli algoritmi di routing per l'instradamento dei pacchetti scambiati dai router.

Il Simulatore eNSP

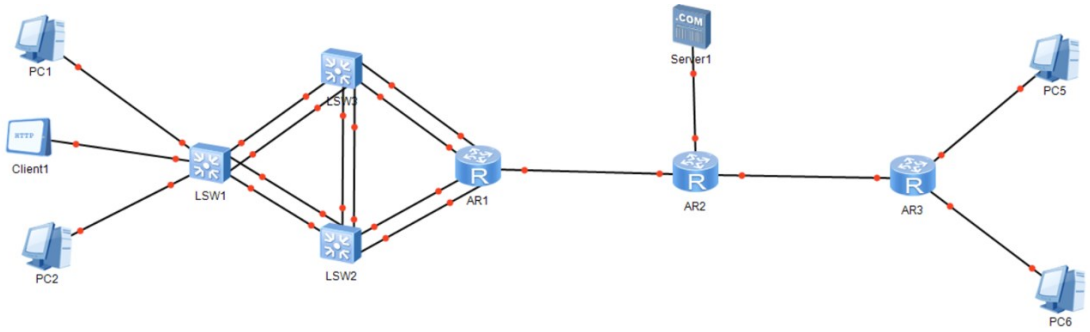
eNSP (Enterprise Network Simulation Platform) sviluppato da “Huawei Technologies Co., Ltd”, uno dei principali fornitori globali di soluzioni ICT, è una piattaforma di simulazione di reti grafiche ed aziendali. Un simulatore di reti, in generale, è un potente software in grado in grado di creare e simulare scenari di reti a scopo didattico o di test. Questi sono in grado di simulare le funzioni di base dei dispositivi che si vogliono testare ma non forniscono tutte le lore funzionalità. eNSP permette la simulazione di apparati reali come router switch e tecnologie di rete tramite la configurazione da parte di un utente. Il software funziona grazie alla virtualizzazione, quindi, permette l’emulazione di reti complesse, anche a livello enterprise, su di un singolo hardware. La sua interfaccia grafica è così ben costruita che permette a qualsiasi utente di testare questi apparati con facilità e rapida intuizione.

Gli apparati implementati sul simulatore utilizzano come sistema operativo VRP (Versatile Routing Platform). La configurazione, la gestione e il monitoraggio dei dispositivi che utilizzano VRP si basa su un sistema standardizzato e gerarchico a riga di comando. eNSP per poter funzionare ha bisogno di un software di supporto: VirtualBox il quale è un software per la virtualizzazione di sistemi x86. Rilasciato come progetto open source, adesso VirtualBox è supportato da Oracle. VirtualBox crea su un computer con un sistema operativo (definito 'host') una macchina virtuale (VM, virtual machine) su cui può essere eseguito un sistema operativo differente (definito 'guest'). In fase di configurazione, si può scegliere quanti core della CPU, quanta RAM e quanto spazio su disco devono essere dedicati alla VM. eNSP utilizza VirtualBox per poter emulare il funzionamento dei vari apparati di rete come router e switch. Oltre a VirtualBox è necessario installare Wireshark software in gradi analizzare il traffico dei pacchetti sulle varie interfacce e WinPcap software in grado di acquisire, filtrare e generare dati statistici per i pacchetti di rete una interna. Senza l’ausilio di questi software aggiuntivi il funzionamento di eNSP sarebbe limitato.

Durante l’utilizzo del simulare è possibile riscontrare diverse criticità dovute in primis alla vecchiaia e mancanza di aggiramento del software e anche alla difficile compatibilità con diversi sistemi operativi moderni come Windows 11. Oltre a questo, sono presenti anche diversi bug ancora irrisolti dalla casa produttrice che alcune volte impediscono il funzionamento complessivo di tutta la configurazione in atto, creando disagio (es. DHCPv6). Tutto questo è dovuto al fatto

che eNSP è end of life da ormai molti anni ed è compatibile al massimo con VirtualBox versione 5.2.X.

Descrizione della topologia



La topologia è divisa in due macroaree: la parte sinistra comprende un router (AR1) tre switch (LSW1, LSW2, LSW3) due pc (PC1, PC2) e un client (Client1), mentre la parte destra è composta da un router (AR3) e due pc (PC5 e PC6); nella topologia è presente un terzo router (AR2) che si trova al centro e verrà utilizzato per mettere in collegamento le due parti, a quest'ultimo è connesso anche un server (Server1). Nella topologia coesistono due tecnologie a L3: IPv6 di cui fanno parte PC1 e PC5 e IPv4 di cui fanno parte PC2 e PC6. Lo scopo della configurazione della topologia è quello di creare vlan in modo che i due "mondi" IPv4 e IPv6 siano separati anche a L2. Per effettuare questo passaggio è necessario assegnare a ogni dispositivo un indirizzo IP (IPv4 o IPv6) mediante un'assegnazione dinamica tramite i DHCP server. Una volta effettuato questo passaggio per poter mettere in collegamento vi è la necessità di configurare un tunnel GRE tra i due router più esterni delle due parti. Infine, per poter rendere raggiungibile il SERVER 1 avente indirizzo IP pubblico si utilizza la tecnica del NAT per renderlo "pingabile" dai dispositivi facenti parte della vlan IPv4.

RSTP

Il primo protocollo implementato nella topologia è RSTP (RAPID SPANNING TREE PROTOCOL) variante di STP il quale è un protocollo di comunicazione, utilizzato all'interno di reti complesse, che opera a livello fisico (L2) ed ha lo scopo di coordinare il funzionamento di switch che si trovino su percorsi ridondati. Questo protocollo è stato implementato negli switch e nel router facenti parte della sezione di sinistra, la quale presenta percorsi multipli. STP garantisce una rete senza loop ma la sua bassa velocità di convergenza può essere un problema in alcuni ambiti operativi. Per questo successivamente è stato introdotto RSTP che utilizza un processo di proposal agreement per definire il ruolo delle porte di uno switch e consente l'immediata negoziazione dei collegamenti, diminuendo drasticamente il tempo di convergenza della topologia. Quindi i ruoli delle porte sono dichiarati e non devono essere desunti dalla topologia. In questo protocollo abbiamo diversi ruoli di porte:

-Designeted (porta downstream);
-Root (porta upstream);
-Altrenate (backup della porta root);
-Backup (backup della porta designeted);
-Edge (porta di connessione tra switch e end point).

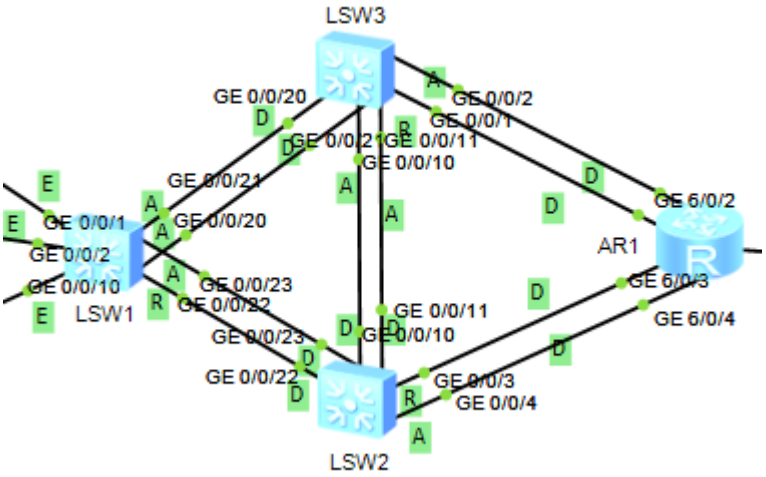
Per poter applicare il protocollo RSTP su tutti gli apparati è necessario entrare nell'interfaccia di configurazione di ogni dispositivo, entrare in modalità "SYSTEM VIEW" e digitare il comando "stp mode rstp". Il protocollo RSTP richiede che uno switch sia impostato come primary e per poter fare questo è necessario digitare il comando nello switch prescelto: "stp priority 4096". La priorità se non impostata manualmente viene calcolata tramite un algoritmo che prevede l'elezione del root bridge, attraverso il confronto dei "bridge id" dei dispositivi coinvolti nella topologia.

LACP

LAG (LINK AGGREGATION GROUP) è una tecnica con la quale si possono aggregare più interfacce fisiche per creare una interfaccia logica avente banda pari alla somma delle velocità delle singole interfacce fisiche. LACP è un protocollo che consente a due switch di negoziare automaticamente i parametri LAG. L'interfaccia logica che si andrà a formare prende il nome di Ehternet-Trunk. Questo protocollo viene utilizzato per creare aggregazioni di collegamenti Ethernet tra dispositivi di rete al fine di aumentare la capacità e la ridondanza della connessione e quindi l'affidabilità della connessione. Questa tecnica viene utilizzata soprattutto in ambito enterprise a livello Core/Aggregation dove è richiesta una maggiore affidabilità e capacità. Nella topologia in questione è stata scelta l'opzione che fa uso del protocollo LACP (Lacp-mode static), la quale prevede che ci sia uno scambio di BPDU (bridge protocol data unit) tra gli switch in moda tale da coordinare su quali link saranno attivi e quali link saranno di backup. In modalità static-lacp i dispositivi possono avere due tipi di ruolo: "actor" ovvero il dispositivo a priorità maggiore che comanda e "partner" il dispositivo restante che esegui gli ordini. La priorità di default è 32768 (ha maggiore priorità il numero più piccolo o in caso di parità colui che ha indirizzo MAC minore). È possibile creare un massimo di 64 Eth-Trunk con un massimo di 8 interfacce fisiche per ogni trunk. Per creare un link aggregation bisogna digitare:

-"interface Eth-Trunk 1" (interfaccia logica eht-trunk numero 1);
-"mode lacp-static" (tipologia lacp static);
-"trunkport gig 0/0/1" (serve per aggiungere una qualsiasi interfaccia al trunk);
-"max active-linknumber 1" (numero massimo di interfacce attive);
-"bpdu enable" (consente il passaggio delle bpdu tra gli switch).

Questa procedura è stata applicata nel lato sinistro della configurazione a tutti gli switch presenti e al router AR1 per creare 5 eth-trunk.



VLAN

Per far sì che i mondi ipv4 e ipv6 siano separati anche a L2 è necessario l'utilizzo di vlan. Una VLAN è una tecnologia che viene utilizzata per suddividere una LAN fisica in più domini di broadcast al fine di isolare i servizi e, di migliorare la sicurezza e la gestione della rete. Gli host all'interno di una VLAN possono comunicare direttamente solo con altri host nella stessa VLAN e devono utilizzare un router per comunicare con gli host in altre VLAN. È possibile creare una VLAN solo tramite apparati di livello L2, ovvero degli switch. La tecnologia VLAN ha il vantaggio aggiuntivo dell'isolamento del traffico senza limitazioni di confini fisici. Gli utenti possono essere fisicamente dispersi in varie zone ma essere comunque associati come parte di un singolo dominio di broadcast, isolando a livello logico gli utenti da altri gruppi di utenti a livello L2. Gli switch riconoscono che un frame fa parte di una precisa vlan tramite l'attributo TAG (posizione che sostituisce il campo type). Nelle VLAN esistono due tipi di port link type: access e trunk. Il port link type di tipo access si utilizza quando vi è un collegamento tra uno switch e un end point; invece, il tipo trunk, si utilizza per il collegamento tra switch, essendo sempre un collegamento punto-punto. Il port link type trunk ha una proprietà molto importante: consente di veicolare su una sola connessione fisica, frames appartenenti a tante vlan differenti distinguendoli

attraverso il loro Vlan ID (attributo del campo type). [1] Entrambe le tipologie di collegamento sono proprietà di interfaccia. Nella topologia sono state implementate due vlan: vlan4 e vlan6, alle quali verranno assegnati rispettivamente tutti i dispositivi ipv4 e ipv6. La creazione di un vlan si esegue tramite il comando "vlan |X|", dove X rappresenta il numero della vlan. Per associare le porte alle vlan, una volta entrati nell'interfaccia di interesse è necessario digitare: "port link-type access" e successivamente "port default vlan |X|" dove X sta a indicare il numero della vlan.

Per poter permettere la comunicazione tra diverse vlan è necessario utilizzare lo scambio di pacchetti ip e per questo vi è la necessità di utilizzare apparati di livello L3 (come switch L3). Gli switch L3 mantengono tutte le interfacce a L2, ma consentono di creare delle interfacce virtuali chiamate VLANIF (VLAN INTERFACE) che sono a tutti gli effetti delle interfacce di livello 3 e per questo possono ricevere un indirizzo IP. Per poter configurare e assegnare indirizzi ip alle vlanif è necessario utilizzare i comandi: "interface vlanif |X|" per entrare nell'interfaccia della vlan X (X = numero della vlan) e "ip address |indirizzo ip| |subnet mask|" per assegnargli un indirizzo ipv4 o ipv6. Nella topologia sono state create due vlan interface: una per il mondo ipv4 (vlanif4) e una per il mondo ipv6 (vlanif6).

In sintesi, una VLAN rappresenta un segmento di rete virtuale, mentre una VLANIF è un'interfaccia logica che consente il routing o la gestione dei pacchetti tra le VLAN e i livelli di rete superiore.

.

[1] Vi è differenza tra l'interfaccia Eth-Trunk del link aggregation e l'interfaccia di tipo trunk nelle vlan, nel primo caso si tratta del nome mentre nel secondo del tipo.

DHCP

Per poter assegnare un indirizzo ip a un'interfaccia di un dispositivo esistono sostanzialmente due metodi: ip statico, dove l'utente assegna manualmente un indirizzo ip ad un'interfaccia o tramite un server DHCP. La scelta effettuata nel nostro caso è quella di utilizzare server DHCP in maniera dinamica. Nella topologia in questione coesistono i due mondi ipv4 e ipv6 e per questo avremmo sia DHCPv4 sia DHCPv6. DHCP (dynamic host configuration protocol) è un protocollo che funziona con un paradigma client-server che assegna a ogni host un indirizzo ip, un default gateway, un server dns e altri parametri necessari per consentire l'accesso alla rete. Le risorse che vengono assegnate ai client si trovano all'interno dei pool che possono essere di due tipologie: global o interface. Nella configurazione viene utilizzata la configurazione dhcp global per i server DHCPv4, il quale prevede di abilitare il server anche per segmenti di rete che non sono direttamente connessi al router. Questa configurazione permette appunto di avere una versatilità maggiore poiché si possono aggiungere maggiori opzioni. Il server DHCPv4 per la parte sinistra è stato configurato su AR1 e per fare questo prima è stato necessario creare un pool di indirizzi che poi il server assegnerà ai vari host tramite i comandi:

- "dhcp enable" (abilita il server dhcp)
- "ip pool pool4" (nome del pool scelto dall'utente);
- "network 192.168.4.0 mask 24" (rete e subnetmask dal quale prendere gli indirizzi);
- "gateway-list 192.168.4.254" (gateway predefinito);
- "dns-list 8.8.4.4." (server dns di Google);
- "lease day 1" (leasy di 1 giorno).

Una volta creato il pool si entra nell'interfaccia prescelta per abilitare il dhcp server e si digita il comando "dhcp select global". Ovviamente se fossero presenti più pool il server andrebbe a prendere il pool al quale fa parte l'indirizzo ip dell'interfaccia stessa. Una volta effettuati questi passaggi si può entrare nel pannello di controllo di PC2 e nella command line, digitare il comando "ipconfig" e verificare che il pc abbia effettivamente preso un indirizzo ipv4. La nostra configurazione presenta più host richiedenti ipv4 e per questo è stato svolto lo stesso processo di configurazione di server dhcpv4 anche nella parte destra della topologia, con i comandi sopra indicati, ma modificando l'indirizzo ip della rete per il pool, portandolo a "192.168.44.0". Ora sia PC2 che PC6 hanno indirizzo ipv4.

```
PC>ipconfig

Link local IPv6 address.....: fe80::5689:98ff:fe5a:7b2b
IPv6 address.....: :: / 128
IPv6 gateway.....: ::
IPv4 address.....: 192.168.4.253
Subnet mask.....: 255.255.255.0
Gateway.....: 192.168.4.254
Physical address.....: 54-89-98-5A-7B-2B
DNS server.....: 8.8.4.4
```

PC2

```
PC>ipconfig

Link local IPv6 address.....: fe80::5689:98ff:fe11:4d8a
IPv6 address.....: :: / 128
IPv6 gateway.....: ::
IPv4 address.....: 192.168.44.253
Subnet mask.....: 255.255.255.0
Gateway.....: 192.168.44.254
Physical address.....: 54-89-98-11-4D-8A
DNS server.....: 8.8.4.4
```

PC6

Oltre agli indirizzi ipv4 allo stesso tempo coesistono gli indirizzi ipv6. Per fare ciò è necessario configurare anche un server DHCPv6 che assegni ai vari host gli indirizzi. Il funzionamento è simile a quello del DHCPv4 ovvero vi è sempre un paradigma client-server, nel quale abbiamo un client che fa una richiesta UDP, alla porta 547 del server, di informazioni in merito alla sua configurazione e il server risponde alla porta 546 del client. Come nel caso precedente dobbiamo definire prima di tutto un address pool dal quale il server preleva e assegna gli indirizzi ip agli host, tramite i comandi:

- “dhcpv6 pool pool6” (nome del pool scelto dall’utente);
- “address prefix 3000::/64 ” (prefix dell’indirizzo che verrà assegnato all’host);
- “excluded-address 3000::1 ” (indirizzi non assegnabili agli host perché statici);
- “dns-server 3001::1” (server dns);
- “dns-domain-name huawei.com” (nome del dns).

Una volta creato il pool, questo deve essere associato ad un’interfaccia tramite l’esecuzione di questa procedura:

- “ipv6” (abilita ipv6);
- “dhcp enable ” (si abilita il server dhcp);
- “interface |X|” (si entra in un’interfaccia X)
- “ipv6 enable” (si abilita ipv6);
- “ipv6 address 3000::1/64 (si assegna un indirizzo ipv6 all’interfaccia);
- “dhcpv6 server pool6 (si assegna il pool al dhcp).

Una volta completata tutta la configurazione, va replicata anche nell’altro lato della topologia sempre con indirizzi diversi, per far si che sia PC1 che PC5 abbiano indirizzi ipv6.

```
Link local IPv6 address.....: fe80::5689:98ff:fefc:653a
IPv6 address................: 3000::2 / 128
IPv6 gateway................: fe80::2e0:fcff:fe1f:35f
IPv4 address................: 0.0.0.0
Subnet mask.................: 0.0.0.0
Gateway.....................: 0.0.0.0
Physical address............: 54-89-98-FC-65-3A
DNS server..................:
```

PC1

```
Link local IPv6 address.....: fe80::5689:98ff:fe1d:20e1
IPv6 address................: 4000::2 / 128
IPv6 gateway................: fe80::2e0:fcff:fe1f:35f
IPv4 address................: 0.0.0.0
Subnet mask.................: 0.0.0.0
Gateway.....................: 0.0.0.0
Physical address............: 54-89-98-1D-20-E1
DNS server..................:
```

PC5

GRE

Una volta assegnati indirizzi IP a tutti gli host, sia della parte destra sia della parte sinistra della topologia, è necessario poter mettere in comunicazione, per esempio, le due LAN che fanno capo a AR1 e AR3. Per fare questo la soluzione migliore è utilizzare la tecnica del GRE: GRE (Generic Routing Encapsulation), è un protocollo di tunneling utilizzato per incapsulare pacchetti di diversi protocolli, all'interno di pacchetti IP. GRE consente una maggiore versatilità rispetto ad IPSec (altro protocollo di tunneling, con una attenzione marcata per la sicurezza), visto che quest'ultimo consente di incapsulare solo pacchetti di tipo IP. Tramite GRE è possibile quindi incapsulare un protocollo all'interno di un altro in maniere abbastanza libera. Il funzionamento si basa essenzialmente, sulla costruzione di un Tunnel tra due end point in modo tale che le reti LAN di questi due end point si possano vedere direttamente connessi tra di loro, a prescindere dalla complessità topologica che le separa. Uno dei difetti di utilizzare GRE è la mancanza di sicurezza al suo interno, con la conseguenza che tutti i dati passano in chiaro; come soluzione si può creare un primo tunnel IPSec e creare un secondo tunnel GRE all'interno del primo, così da risolvere il problema e con l'unico svantaggio di aumentare dell'overhead. Sostanzialmente creare un tunnel GRE significa inserire all'interno di un pacchetto un header che ci consente di gestire il tunnel che si va a formare. Tutto questo prevede l'abilitazione all'interno del router di un particolare modulo che gestisce l'incapsulamento di questi pacchetti e fare in modo che questo modulo aggiunga l'header GRE a quello che si vuole traportare.

Nella topologia in questione per permettere la comunicazione tra le LAN ipv6 di AR1 e AR3 è stato necessario creare un tunnel GRE. Preliminarmente sono stati assegnati indirizzi ip alle interfacce dei router AR1, AR2, AR3 che entreranno a far parte del processo di tunneling. Una volta fatto questo, l'interfaccia di partenza e di destinazione devono essere per lo meno raggiungibili altrimenti il tunnel non è in grado di mantenere la comunicazione. La scelta più semplice in questo caso è stata quella di creare rotte statiche tra i vari end point, essendo poche le interfacce in questione. In AR1 imposto "ip route-static 20.0.0.0 255.255.255.0 10.0.0.2" ovvero che la rete 20.0.0.0/24 è raggiungibile tramite 10.0.0.2 (nexthop). Su AR3 al contrario imposto "ip route-static 10.0.0.0 255.255.255.0 20.0.0.1", ovvero che la rete 10.0.0.0/24 è raggiungibile tramite 20.0.0.1(nexthop). Una volta che i due endpoint sono raggiungibili è possibile creare il tunnel tramite l'esecuzione dei comandi:

- “interface Tunnel 0/0/1” (nome del tunnel);
- “ip address 40.0.0.1/24” (ip dell’interfaccia);
-“tunnel-protocol gre” (tipo di incapsulamento utilizzato);
-“source 10.0.0.1” (ip di partenza);
-“destination 20.0.0.2” (ip di destinazione);
-“quit” (uscita dall’interfaccia);
-“ ipv6 route-static 4000:: 64 Tunnel0/0/1” (rotta statica che permette di raggiungere la rete ipv6 di PC5 tramite il tunnel).

Ora tutti i dispositivi di AR1 e AR3 sono in comunicazione e sono mutualmente raggiungibili: il tunnel GRE è configurato. Tramite richieste di ping è possibile verificare che tutti gli host sono raggiungibili e tramite richieste di tracing è possibile verificare che i pacchetti passano realmente all’interno del tunnel GRE.

NAT

NAT (Network Address Traslation) è una tecnica che permette di tradurre un indirizzo ip privato in un indirizzo ip pubblico. È fortemente utilizzato in quanto al giorno d’oggi ipv4 è il protocollo maggiormente utilizzato e gli indirizzi ip pubblici assegnati dagli enti internazionali sono già terminati; dunque, è necessario utilizzare il NAT per poter navigare in internet. Il dispositivo che tipicamente effettua il NAT è il router, il quale effettua due operazioni:

1. modifica il contenuto del campo “ip sorgente” quando il pacchetto ip lascia la LAN;
2. modifica il contenuto del campo “ip destinatario” quando il pacchetto ip entra nella LAN.

Nella topologia è presente un Server, dunque, è necessario configurare un NAT per la rete a cui fanno parte PC2 e Client1 e per la rete di PC6 per far si che il server sia raggiungibile. Come operazione preliminare è necessario assegnare un indirizzo IP pubblico al Server1 tramite la sua finestra di impostazioni e poi assegnare un indirizzo ip, facente parte della stessa LAN, all’interfaccia gig 0/0/2 del router AR2 connessa con il nostro server. Successivamente, è poi

indispensabile creare due rotte statiche per poter rendere raggiungibile la rete a cui fa parte il Server1 (1.1.1.0/24): quindi in AR1 digito: “ip route-static 1.1.1.0 255.255.255.0 10.0.0.2” per fare in modo che la rete in questione è raggiungibile tramite 10.0.0.2 (interfaccia di AR2) e digito su AR3 “ip route-static 1.1.1.0 255.255.255.0 20.0.0.1” ovvero che la rete in questione è raggiungibile tramite 20.0.0.1 (interfaccia di AR2). Il tipo di NAT che è stato scelto per questa configurazione è “Easy IP”. Questa tipologia prevede che uno o più spazi di indirizzi ip privati possano essere trasformati in un solo indirizzo ip pubblico che è quello che l’ISP assegna all’interfaccia pubblica del router. L’unico strumento di cui ho bisogno per configurare un easy ip è un classificatore del traffico, cioè devo selezionare quale tipo di traffico può accedere al NAT senza la necessità di configurare un pool degli indirizzi, perché questa informazione viene direttamente evinta dall’interfaccia; quindi, basta configurare un acl. Su AR1 digito:

-“acl 2000”;
-“ rule 5 permit source 192.168.4.0 0.0.0.255”.

poi una volta entrato nell’interfaccia WAN digito il comando “nat outbound 2000”. Invece su AR3:

-“acl 2000”;
-“ rule 5 permit source 192.168.4.0 0.0.0.255”.

e digito lo stesso comando sull’interfaccia WAN. Ora il mio NAT è configurato e posso raggiungere il Server1, il quale ha indirizzo ip pubblico, sia dalla rete di PC2 sia dalla rete di PC6.

Test

Per poter verificare che tutte le configurazioni realizzate, siano state applicate correttamente è necessario eseguire dei test sul simulatore eNSP. Il test più utilizzato è il ping che ci permette di capire istantaneamente se due dispositivi sono raggiungibili mutuamente. [2] Questo si esegue tramite riga di comando digitando: "ping " e di seguito l'indirizzo ip dell'host che si vuole raggiungere (una volta a conoscenza dell'indirizzo). Nel caso sia un host avente indirizzo ipv6 è necessario aggiungere la sigla "ipv6" dopo ping. Un primo test può essere quello di verificare se PC2 e PC6 risultino raggiungibili:

```
PC>ping 192.168.44.253

Ping 192.168.44.253: 32 data bytes, Press Ctrl_C to break
From 192.168.44.253: bytes=32 seq=1 ttl=125 time=78 ms
From 192.168.44.253: bytes=32 seq=2 ttl=125 time=78 ms
From 192.168.44.253: bytes=32 seq=3 ttl=125 time=78 ms
From 192.168.44.253: bytes=32 seq=4 ttl=125 time=62 ms
From 192.168.44.253: bytes=32 seq=5 ttl=125 time=47 ms

--- 192.168.44.253 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
```

Da PC2 eseguo il ping verso PC6

```
PC>ping 192.168.4.253

Ping 192.168.4.253: 32 data bytes, Press Ctrl_C to break
From 192.168.4.253: bytes=32 seq=1 ttl=125 time=78 ms
From 192.168.4.253: bytes=32 seq=2 ttl=125 time=109 ms
From 192.168.4.253: bytes=32 seq=3 ttl=125 time=94 ms
From 192.168.4.253: bytes=32 seq=4 ttl=125 time=94 ms
From 192.168.4.253: bytes=32 seq=5 ttl=125 time=78 ms

--- 192.168.4.253 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
```

Da PC6 eseguo il ping verso PC2

[2] Ping (Packet internet groper) è un'utility usata per misurare il tempo, espresso in millisecondi, impiegato da uno o più pacchetti ICMP a raggiungere un dispositivo di rete (attraverso una qualsiasi rete basata su IP) e a ritornare indietro all'origine.

È possibile eseguire test di ping anche da un router verso un qualsiasi host e viceversa, ad esempio è possibile verificare che dal router AR1, facente parte del lato sinistro, PC5 e PC6 siano raggiungibili.

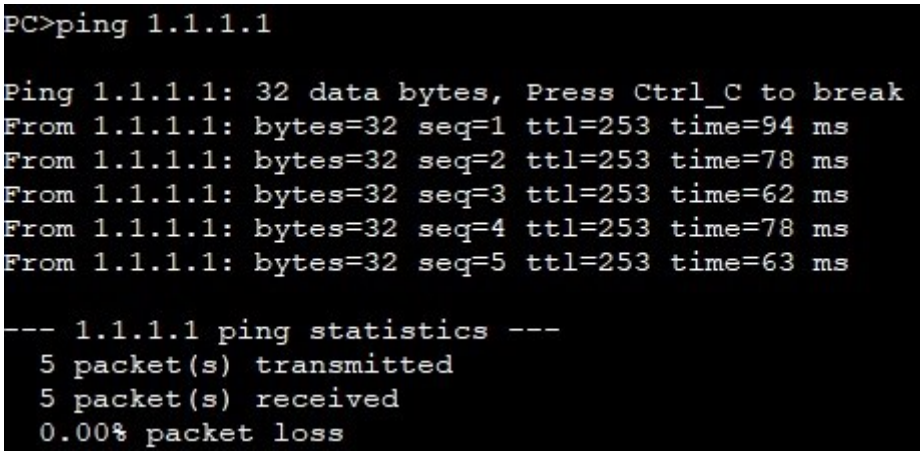
```
<AR1>ping ipv6 4000::2
  PING 4000::2 : 56  data bytes, press CTRL_C to break
    Reply from 4000::2
    bytes=56 Sequence=1 hop limit=254  time = 60 ms
    Reply from 4000::2
    bytes=56 Sequence=2 hop limit=254  time = 40 ms
    Reply from 4000::2
    bytes=56 Sequence=3 hop limit=254  time = 50 ms
    Reply from 4000::2
    bytes=56 Sequence=4 hop limit=254  time = 40 ms
    Reply from 4000::2
    bytes=56 Sequence=5 hop limit=254  time = 30 ms

  --- 4000::2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 30/44/60 ms
```

Da AR1 eseguo il ping ipv6 verso PC5

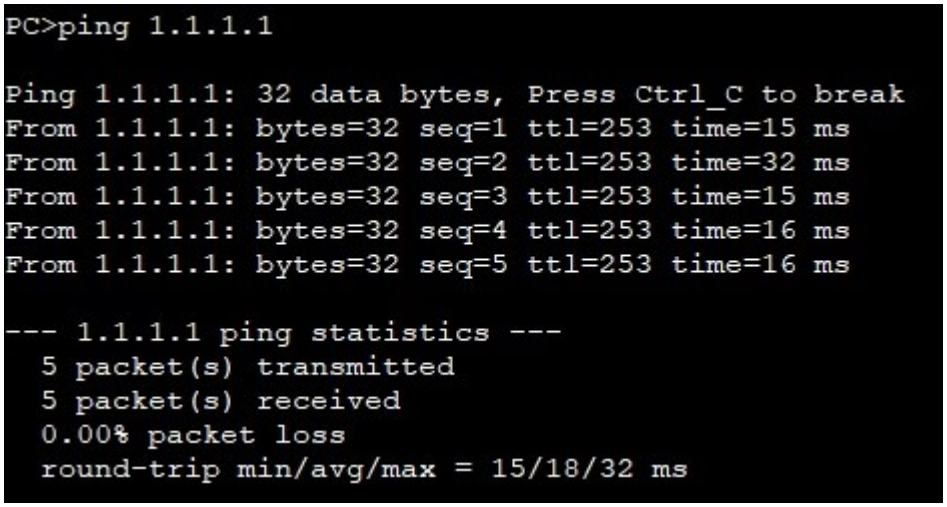
```
<AR1>ping 192.168.44.253
  PING 192.168.44.253: 56  data bytes, press CTRL_C to break
    Reply from 192.168.44.253: bytes=56 Sequence=1 ttl=126 time=40 ms
    Reply from 192.168.44.253: bytes=56 Sequence=2 ttl=126 time=30 ms
    Reply from 192.168.44.253: bytes=56 Sequence=3 ttl=126 time=30 ms
    Reply from 192.168.44.253: bytes=56 Sequence=4 ttl=126 time=30 ms
    Reply from 192.168.44.253: bytes=56 Sequence=5 ttl=126 time=30 ms

  --- 192.168.44.253 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 30/32/40 ms
```

Da AR1 eseguo il ping verso PC6

Infine, per verificare che il Server1 sia raggiungibile sia dai dispositivi del lato destro, che del lato sinistri e per verificare che il NAT sia stato configurato correttamente possiamo eseguire test di ping da PC2 (lato sinistro) e PC6 (lato destro) verso l'indirizzo ip pubblico del Server1 (1.1.1.1).

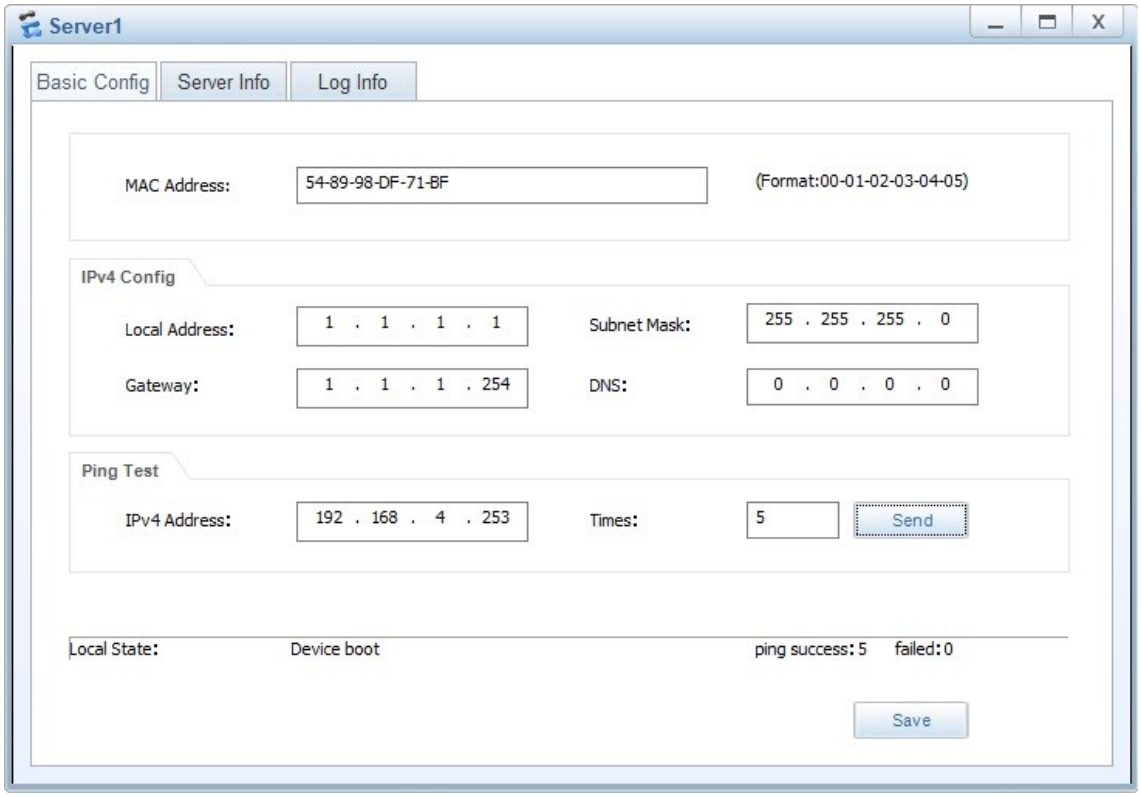


Da PC2 eseguo il ping verso SERVER1



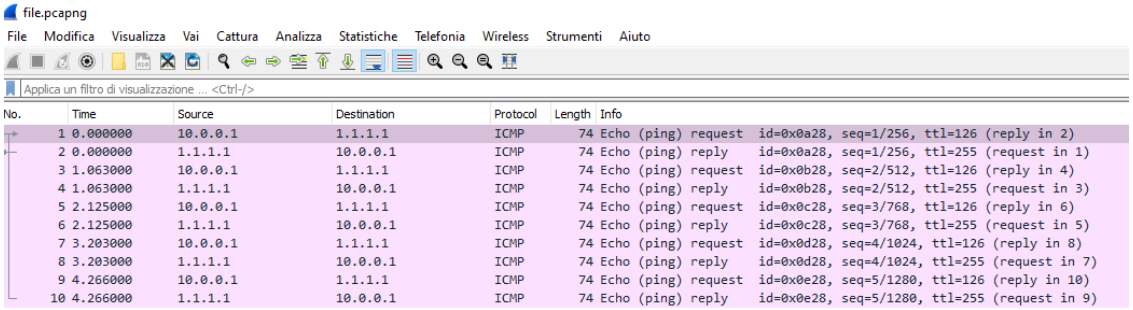
Da PC6 eseguo il ping verso SERVER1

Per eseguire un test di ping da SERVER1 verso un qualsiasi host è sufficiente entrare nel pannello di configurazione dove è presente una voce “PING TEST” e digitare un qualsiasi indirizzo ip.

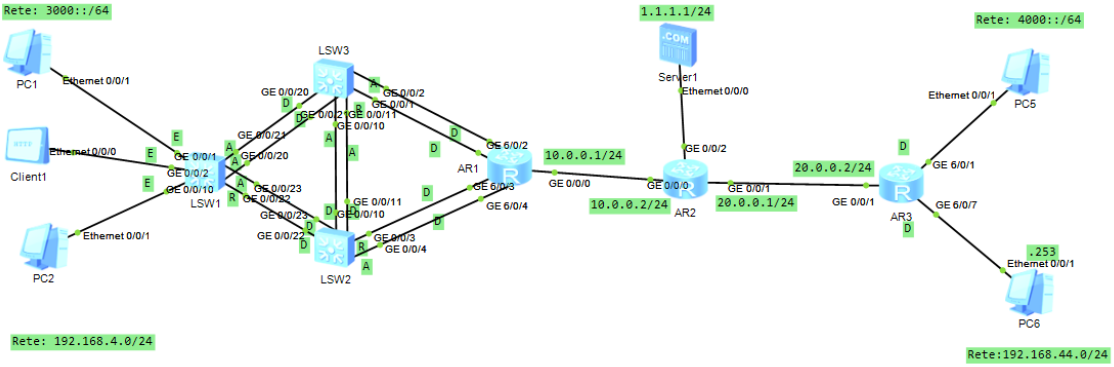


Da SERVER1 eseguo il ping verso PC2.

Inoltre, per mostrare il funzionamento del NAT è possibile utilizzare il software Wireshark per poter analizzare il traffico dei pacchetti su un’interfaccia in particolare; in questo caso è stata eseguita una cattura su AR1 nell’interfaccia gig 0/0/0 dopo che il router ha eseguito il NAT.



Conclusioni



Tramite tutti gli algoritmi e i protocolli implementati nella topologia è stato possibile effettuare una configurazione di apparati di rete IPv4 e IPv6 i quali sono in grado di coesistere e funzionare. In particolar modo è stato utilizzato RSTP per evitare loop tra i vari collegamenti degli switch, e in aggiunta LACP per gestire i ruoli dei collegamenti. Per assegnare gli indirizzi ipv4 e ipv6 ai vari host sono stati utilizzati rispettivamente server DHCPv4 e server DHCPv6. Mediante l'utilizzo di vlan è stato possibile separare i mondi ipv4 e ipv6 anche a L2. Un tunnel GRE ha reso possibile la creazione di una comunicazione tra le LAN della parte destra e della parte sinistra della topologia. Infine, è stato indispensabile l'utilizzo del NAT per poter comunicare con il server avente indirizzo ip pubblico. Tutto questo ha permesso di simulare una rete complessa di livello enterprise.

Bibliografia e Sitografia

- Basic Enterprise Network Architectures di HUAWEI TECHNOLOGIES CO., LTD.
- Intermediate Enterprise Network Architectures di HUAWEI TECHNOLOGIES CO., LTD.
- eNSP Help
- https://support.huawei.com/enterprise/it/data-communication/ensp-pid-9017384

Ringraziamenti

Parto con il ringraziare la mia famiglia per avermi dato la possibilità di intraprendere questo percorso, per il sostegno e per aver permesso tutto questo. Ringrazio tutti i miei amici, veramente tutti, che mi hanno sempre ricordato quanto valessi e quanto sarei potuto arrivare lontano. Ringrazio tutto lo staff e tutti gli amici del Samp che mi hanno permesso di maturare, imparare a prendermi le mie responsabilità e a lavorare in gruppo, come leader e come componente attivo, senza mai sminuire nessuno, anzi valorizzando ogni membro. Infine, ringrazio tutto lo staff del MamaTeam/Celeste, con me da ormai 10 anni, che ha sempre creduto in me e mi ha valorizzato. Grazie a tutti, vi voglio bene.