UNIVERSITA' POLITECNICA DELLE MARCHE

Facoltà di ingegneria

Corso di laurea in ingegneria informatica e dell'automazione

SIMULAZIONE DI UNA ARCHITETTURA DI RETE ENTERPRISE BASATA SUL SOFTWARE HUAWEI ENSP

Simulation of an enterprise network architecture
based on Huawei eNSP software

Relatore:
Prof. Ennio Gambi

Correlatore:
Prof. Adelmo De Santis

Candidato:
Cristian Cingolani

Anno Accademico 2019-2020

RINGRAZIAMENTI

Prima di procedere con la trattazione, vorrei ringraziare tutte le persone che hanno contribuito al raggiungimento di questo importante obiettivo.

Le prime persone che tengo a ringraziare sono il prof. Ennio Gambi e il prof. Adelmo De Santis le quali mi hanno fatto appassionare a tutto il mondo delle telecomunicazioni. Da ammirare la loro capacità di trasmettere tutto questo nonostante la situazione particolare che stiamo affrontando. Sono stati sempre disponibili per chiarimenti e per il superamento di problematiche che si sono presentate nello svolgimento del progetto e non solo.

Vorrei ringraziare i miei genitori perché mi hanno accompagnato e sostenuto in questo percorso. Con mio padre ho condiviso e continuo a condividere passioni come la tecnologia e forse questo è anche il motivo per cui sono arrivato a questo traguardo. Mia madre poi si è sempre impegnata per permettermi di raggiungere qualsiasi tipo di obiettivo.

Infine, un ringraziamento va ai miei amici e a tutte le persone con le quali ho passato bellissimi momenti riguardanti la vita universitaria e non, grazie a loro sono riuscito ad affrontare le giornate in modo migliore e con serenità. In particolare, un grazie va ad Alessandro Ricci con il quale ho condiviso ogni momento della mia vita da quando l'ho conosciuto.

INDICE

1 INTRODUZIONE

In questo lavoro di tesi si vuole proporre lo sviluppo di una architettura di rete enterprise andando ad esaminare puntualmente le tecnologie ed i protocolli utilizzati al fine di ottenere una rete funzionante e conforme alle specifiche indicate.

In particolare, viene utilizzato il software eNSP, sviluppato da Huawei, il quale fornisce mediante interfaccia grafica un laboratorio virtuale con tutti gli strumenti di routing e switching necessari alla pianificazione, costruzione, messa in funzione e manutenzione di reti ICT.

Questo elaborato è stato suddiviso in quattro aree principali:

- Nel capitolo due si introduce l'argomento relativo alle telecomunicazioni e ad internet, in particolare lo sviluppo negli anni delle tecnologie ed i principali problemi da affrontare come ad esempio il crescente numero di dispostivi connessi alla rete e la crescente richiesta del mercato di reti veloci e stabili. Si vuole mettere in risalto lo sviluppo tecnologico che ha portato le reti, anche domestiche, a passare da collegamenti su rame a collegamenti ottici. Vengono trattati brevemente TCP/IP e OSI.
- Nel capitolo tre si fa una rapida panoramica sul software utilizzato per lo sviluppo del progetto, eNSP
- Nel capitolo quattro si analizzano in dettaglio le varie fasi dello sviluppo del progetto, ponendo l'accento di volta in volta sul protocollo/tecnologia utilizzata.
- Infine, nel capitolo cinque, vengono svolti dei test sull'effettivo soddisfacimento delle specifiche facendo uso di Wireshark, ping fra dispositivi, ecc…

2 INTERNET

Di seguito viene ripercorsa la storia che ha portato lo sviluppo delle reti internet a rivestire un ruolo fondamentale in ogni ambito della società. Una così larga diffusione ha portato con sé numerosi problemi, molti dei quali sono stati risolti mentre altri non hanno visto soluzioni definitive in attesa dello sviluppo e della diffusione di nuove tecnologie e protocolli.

Per comprendere meglio cos'è effettivamente internet e quali sono alcune sue caratteristiche si può ricorrere ad una citazione:

"Multivac non aveva una sede precisa da monto tempo. Era una presenza globale […]. Possedeva un cervello suddiviso in centinaia di sussidiari ma che funzionava come un'entità unica. Aveva sbocchi ovunque […]"

Icaac Asimov 'Vita ai tempi di Multivac' Times, 1975

2.1 NASCITA E SVILUPPO DI INTERNET

Internet nasce negli Stati Uniti per conto di ARPA, l'ente di ricerca dell' esercito statunitense. L'obiettivo era di creare una rete che mettesse in comunicazione i punti nevralgici del sistema difensivo. Lo scambio di informazioni inizialmente riguardava 4 università e si basava solo su connessioni Client-Server che avevano velocità di circa 50kbps. Un passo in avanti venne fatto nel 1971 con l'e-mail.

Ben presto anche altre nazioni aderirono al progetto e la prima rete in Italia fu sviluppata a Pisa nel 1986. Si poteva dunque definire internet come una "rete di reti" ed il tutto fu possibile anche grazie alle caratteristiche del protocollo TCP/IP, come vedremo in seguito.

Con la standardizzazione del WWW, World Wide Web, negli anni 90 ci fu una vera e propria esplosione di internet. In questo periodo si svilupparono i primi browser, alcuni linguaggi di markup come HTML, protocolli come http, strutture come URL e molto altro.

2.2 CRESCITA ESPONENZIALE DEI DISPOSITIVI CONNESSI

Il rapido sviluppo di dispositivi in grado di connettersi ad Internet mise rapidamente in crisi protocolli come IPV4.

IPV4 è un protocollo pubblicato nel 1981 da IETF che ha lo scopo di individuare qualsiasi dispositivo connesso a global internet tramite un identificatore univoco, composto da 32 bit. Gli aspetti tecnici relativi a tale protocollo vengono spiegati nel capitolo 4.1.1.

Nel 1998 venne dunque sviluppato IPV6 il quale riserva 128 bit per gli indirizzi IP, una quantità ampiamente superiore alle esigenze attuali. Oltre a questo aspetto IPV6 introduce nuovi servizi e semplifica molto la configurazione e la gestione delle reti IP.

Ancora la transizione non è avvenuta completamente e il 3 febbraio 2011 IANA ha assegnato gli ultimi indirizzi IPV4. La soluzione provvisoria in attesa del passaggio definitivo a IPV6 risiede nel protocollo NAT, approfondito nel capitolo 4.3.1.

Una delle motivazioni principali di questa rapida crescita è da ricercare nello sviluppo di dispositivi IoT, ovvero oggetti che si rendono riconoscibili e acquistano “intelligenza” grazie alla possibilità di ricevere ed inviare informazioni ad altri oggetti. Per comprendere bene di cosa si sta parlando possono essere utili alcuni esempi: smart home (casa domotica), smart city (uso di sensori per parcheggi, illuminazione…), auto connesse, uso di sensori e droni in agricoltura ecc…

Dalle ultime previsioni Cisco, comunicate in occasione del Cisco Annual Internet Report, si passerà da 18.4 bilioni di dispositivi connessi del 2018 a 29.3 bilioni nel 2023. Risalta l’incremento delle connessioni M2M, ovvero di connessioni fra dispostivi in grado di comunicare e prendere decisioni senza un intervento dell’uomo, che passa dal 33% al 50%.

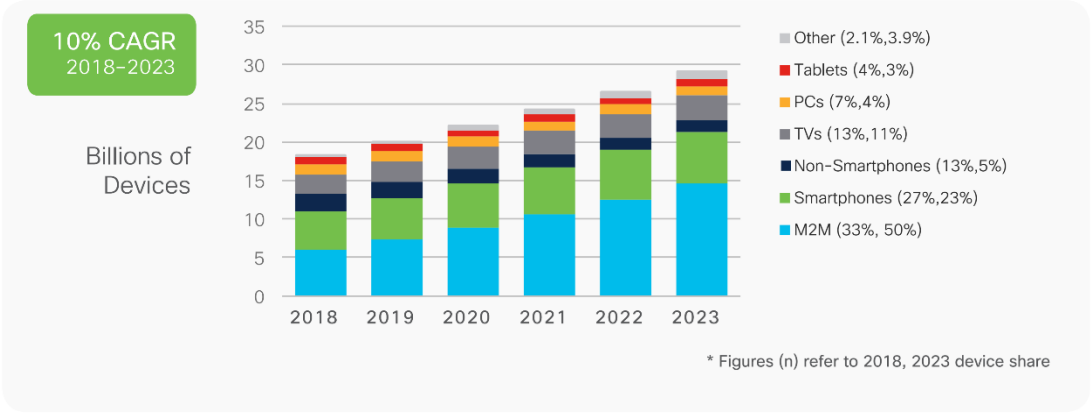


Figura 2.1

2.3 CRESCITA DELLA VELOCITÀ MEDIA DELLE CONNESSIONI

Il crescente uso di internet in molti campi lavorativi, e non, ha reso necessario lo sviluppo di reti sempre più performanti, specialmente durante il 2020 in cui a causa della pandemia di Covid-19 molti lavori e attività didattiche non possono essere svolte in presenza.

Interessanti sono le proiezioni di Cisco che riguardano la velocità media delle reti Wi-Fi nei vari continenti. A livello globale si presuppone di passare da 30.3 Mbps del 2018 a 91.6 Mbps nel 2023

Region	2018	2019	2020	2021	2022	2023	CAGR (2018–2023)
Global	30.3	36.3	50.8	58.9	72.9	91.6	25%
Asia Pacific	34.5	42.2	62.3	80.2	98.5	116.1	27%
Latin America	10.6	12.1	25.1	27.3	30.4	34.6	27%
North America	46.9	56.8	70.7	87.3	98.4	109.5	18%
Western Europe	30.8	36.3	53.4	64.7	79.4	97.4	26%
Central and Eastern Europe	22.6	24.1	30.0	35.4	42.9	52.7	18%
Middle East and Africa	7.0	7.9	16.3	18.6	21.9	25.7	30%

Figura 2.2

Questa crescita è anche da attribuire al miglioramento delle tecnologie legate alla parte fisica in cui si ha un passaggio da connessioni elettriche su rame a connessioni ottiche.

Di seguito un breve riepilogo di queste tecnologie.

Cavo Coassiale

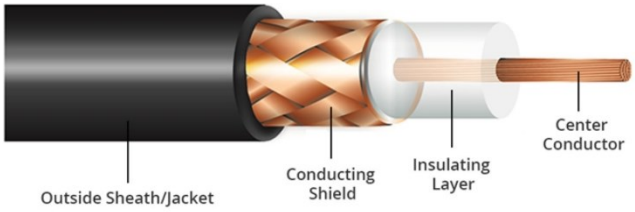


Figura 2.3

È composto da un conduttore in rame posto al centro, anima, e da un dielettrico, in polietilene o PTFE, che separa l'anima da uno schermo esterno costituito da fili metallici intrecciati, maglia, o da una lamina avvolta a spirale, treccia, che garantisce l'isolamento tra i due conduttori. Lo schermo di metallo aiuta a bloccare le interferenze ed il cavo è munito di connettori ai suoi estremi. Gli standard più comunemente usati sono il 10Base2 e il 10Base5 che raggiungono rispettivamente distanze massime di 185 e 500m e velocità di 10Mbps. Il cavo coassiale è un sistema legacy, non essendo più in uso da almeno venti anni.

Cavo Ethernet

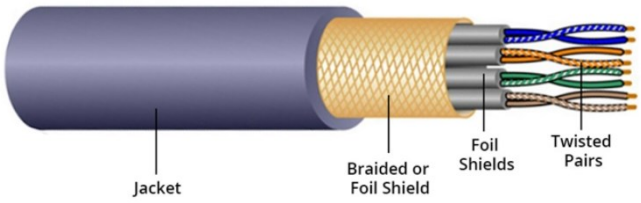


Figura 2.4

Le reti ethernet sono tipicamente realizzate utilizzando un cablaggio costituito da un cavo multicoppia. I connettori usati sono tipicamente Rj45. All'interno della guaina troviamo quattro doppini responsabili della trasmissione dei dati da un dispositivo di rete all'altro, spesso schermati. I doppini si distinguono l'uno dall'altro grazie a colori identificativi: blu, arancio, verde e marrone. Si possono realizzare cavi di tipo “straight” o “crossed”. I cavi straight hanno uno schema di cablaggio uguale ad ogni estremo. I cavi crossed adottano uno schema EIA/TIA 568° ad un estremo ed EIA/TIA 568B all’altro.

Generalmente la distanza massima di connessione è di 100m, mentre la velocità massima è in funzione dello standard adottato come ad esempio 40Gbps per la Cat. 8, 8.1 e 8.2

Fibra Ottica

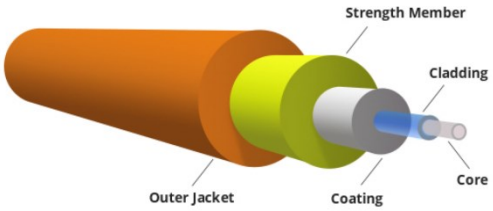


Figura 2.5

Ogni singola fibra ottica è composta da un nucleo cilindrico centrale, core, e un mantello o cladding attorno a esso. Il nucleo presenta un diametro di circa 10 µm per le monomodali e 50 µm per le multimodali, mentre il mantello ha un diametro di circa 125 µm. I due strati sono realizzati con materiali con indice di rifrazione leggermente diversi in modo da riflettere completamente, o quasi, la luce che passa nel core. Si basa sulla legge di Snell che ammette un angolo limite di incidenza oltre il quale la luce viene riflessa completamente e non più rifratta. Le connessioni su fibra ottica possono raggiungere svariati chilometri di lunghezza e velocità di decine di Gbps.

2.4 TCP/IP E OSI

TCP/IP è uno dei protocolli fondamentali per lo sviluppo ed il funzionamento di una rete internet in quanto stabilisce come i dati vengono trasmessi e processati. Per queste ragioni viene trattato in modo non esaustivo ma che permetta di comprendere meglio le operazioni che verranno svolte nei capitoli successivi.

Il modello su cui si basa TCP/IP è caratterizzato da un livello *network interface,* che spesso si considera diviso in *livello fisico* e *livello datalink*, un livello *network,* che stabilisce come il traffico è gestito dal punto di vista logico, un livello *transport*, che assicura affidabilità del traffico fra sorgente e destinatario, ed un livello *application*, che rappresenta l'interfaccia con il quale si offrono servizi all'utente finale.

Di seguito un'immagine che rappresenta i livelli di TCP/IP.

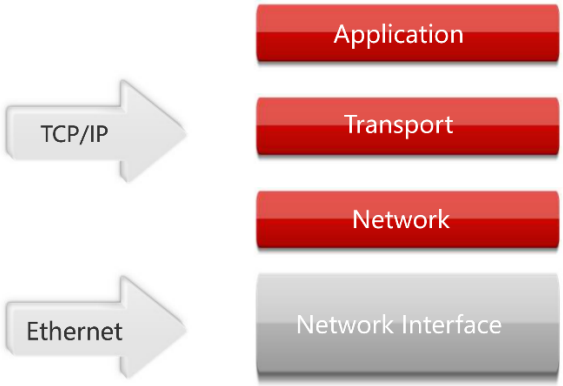


Figura 2.6

Il modello OSI è stato introdotto per creare maggiore chiarezza nella divisione dei livelli, soprattutto per quanto riguarda quelli più bassi.

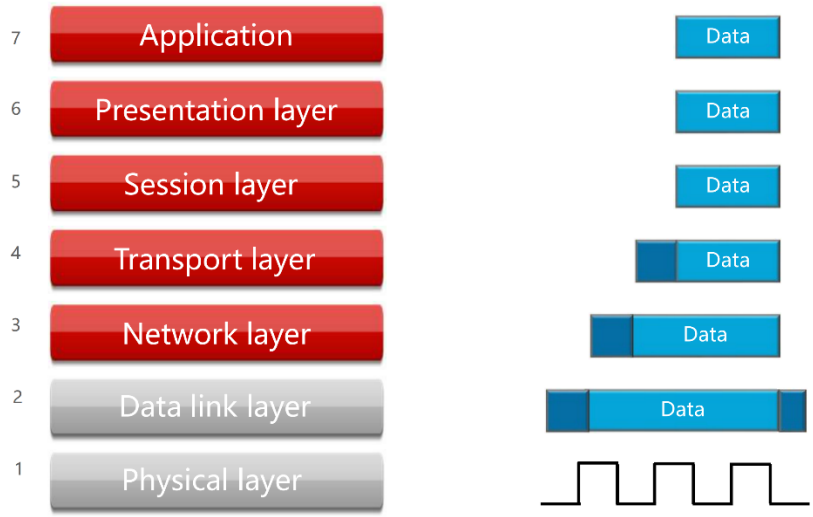


Figura 2.7

1 - Trasmissione dei dati sul mezzo
2 - Protocolli che consentono ai dati di essere trasferiti su un mezzo fisico
3 - Logical addressing
4 - Data delivery between hosts
5 - Raggruppamento di messaggi bidirezionali in un workflow per poterli meglio gestire
6 - Formattazione dei dati e cifratura/decifratura
7 - Fornisce una interfaccia alle applicazioni verso la rete

La figura 2.7 rappresenta il modello ISO/OSI e si può notare come nel passaggio dai livelli alti ai livelli bassi vengano introdotte informazioni aggiuntive. Viene tipicamente aggiunto un header o un trailer che forniscono informazioni, al ricevente, sul modo in cui i dati dovranno essere processati. Questo processo viene chiamato incapsulamento ed introduce un certo overhead.

Le unità di dati impiegate nei vari livelli sono le seguenti:

- Segmento : PDU(Protocol Data Unit) del livello Transport
- Pacchetto : PDU del livello Network
- Frame : PDU del livello Datalink

La mappatura fra indirizzi Ip e MAC viene fatta dal protocollo ARP.

3 SOFTWARE eNSP

Come anticipato, il software utilizzato nel progetto è eNSP ed è stato sviluppato da Huawei. Si basa principalmente su interfaccia grafica, ed è possibile aggiungere dispositivi come server, router, switch, client ecc… e simulare collegamenti fisici. La configurazione dei dispositivi avviene principalmente da riga di comando. Di seguito due screenshot in cui viene mostrata l'interfaccia del programma. Nella sezione a sinistra sono presenti i dispostivi ed i collegamenti che possono essere utilizzati (nella figura 3.2 vengono mostrati tutti). In alto vi è una barra grazie alla quale possono essere svolte operazioni di salvataggio, avvio di dispostivi, uso di tag, ecc… Infine la regione bianca è adibita alla progettazione della rete.

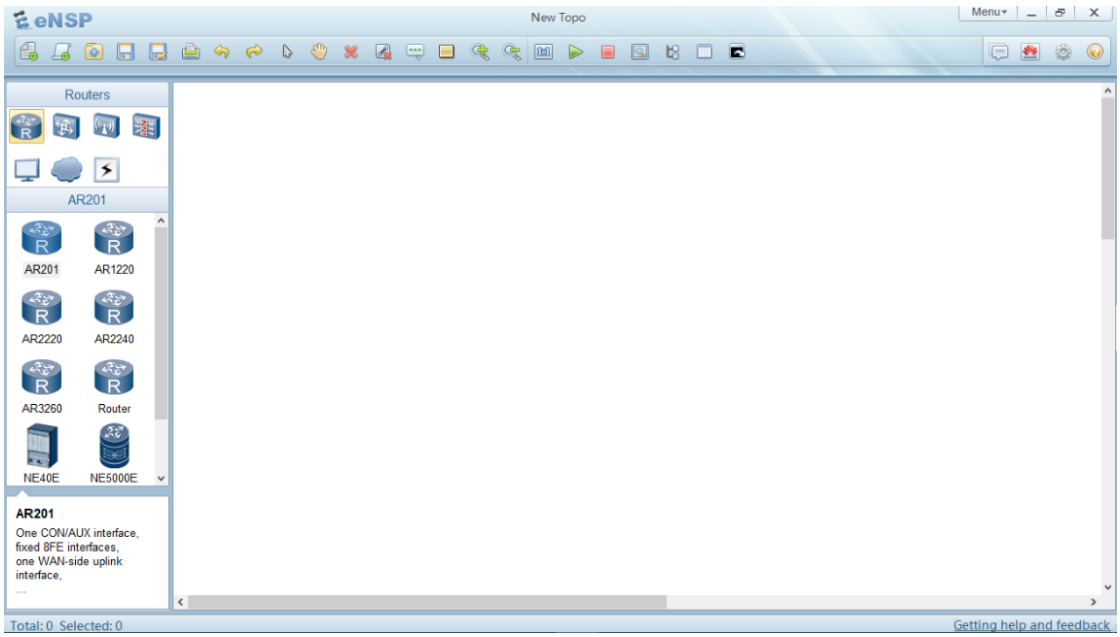


Figura 3.1

Figura 3.2

4 SVOLGIMENTO DEL PROGETTO

In questo capitolo verranno analizzati tutti i passi che porteranno alla realizzazione del progetto. Prima dell'applicazione di un preciso protocollo ci si concentrerà sugli aspetti teorici più rilevanti al fine di comprendere le ragioni che stanno dietro a determinate scelte.

Ogni comando eseguito verrà argomentato e accompagnato da screenshot di finestre eNSP.

L'immagine mostra la struttura iniziale della rete con le relative specifiche da rispettare (riquadro verde).

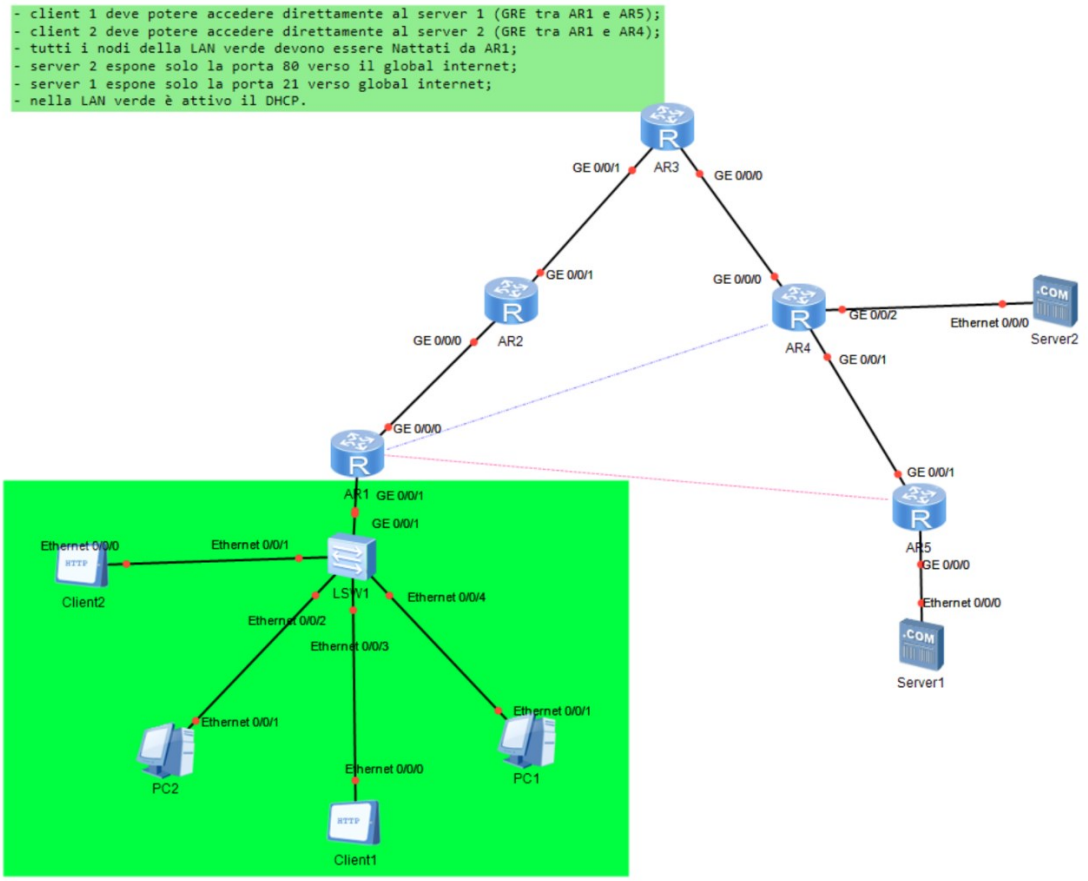


Figura 4.1

4.1 INDIRIZZI IP

Uno degli aspetti fondamentali, senza il quale risulta impossibile riconoscere dispostivi ed interfacce all'interno di una rete, è la configurazione di indirizzi IP.

Di seguito si analizzerà la struttura di un indirizzo ipv4 e poi si proseguirà con la configurazione sulle interfacce dei router e dei server 1 e 2 (lan).

4.1.1 INTRODUZIONE AD IPV4

Definiamo un indirizzo IP come un numero che identifica univocamente un dispositivo, host, collegato a una rete informatica che utilizza l'Internet Protocol come protocollo per l'instradamento/indirizzamento.

Un indirizzo ipv4 è composto da 32 bit di cui una parte, *Network,* che individua la rete ed una parte, *Host,* che individua il dispositivo. All' interno dello spazio di indirizzi definito per una rete, ci sono alcuni indirizzi riservati, come, ad esempio, l' indirizzo di broadcast. Questo si ottiene ponendo ad 1 tutti i bit del campo host ed è utilizzato per inviare un pacchetto a tutti gli host che fanno capo allo stesso spazio di indirizzamento.

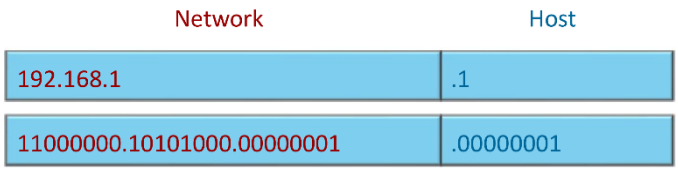


Figura 4.2

Per soddisfare l'esigenza di avere reti di piccole o grandi dimensioni è stato introdotto il concetto di classe. Es: le reti di classe A sono "poche", avendo a disposizione 8 bit per il campo network, ma di grandissime dimensioni, avendo 24 bit riservati al campo host.

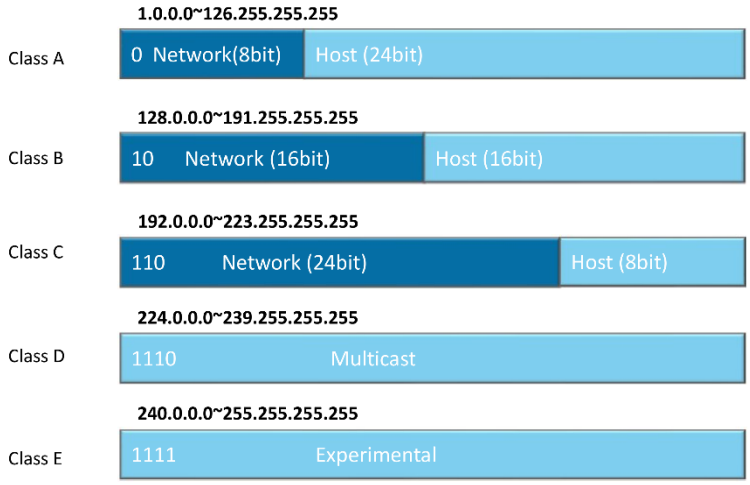


Figura 4.3

La *subnet mask*, che si ottiene ponendo ad 1 i bit del campo network e a 0 quelli della parte host, riveste un ruolo fondamentale. È utile soprattutto laddove è necessario suddividere una rete in più sottoreti, *subnetting*, o accorpare segmenti di rete sotto un'unica rete, *supernetting*.

4.1.2 CONFIGURAZIONE INTERFACCE ROUTER ED IP PRIVATI DEI SERVER

Come prima operazione sono stati assegnati degli indirizzi IP alle interfacce fra i router, ricordando che interfacce collegate dal punto di vista fisico devono avere indirizzi con la stessa parte network.

Per prima cosa bisogna accedere al router corrispondente, entrare in *system-view* (permette di configurare il dispositivo) ed accedere all'interfaccia, a cui si vuole associare un indirizzo IP, tramite il comando *interface GigabitEthernet x/x/x.* In questo caso si presuppone di avere interfacce GigabitEthernet ed a x/x/x deve essere sostituito il numero che identifica l'interfaccia.

L'assegnazione dell'indirizzo avviene tramite il comando *ip address <ip-address > { mask | mask-length }.*

Di seguito due screenshot estratti dalla configurazione.

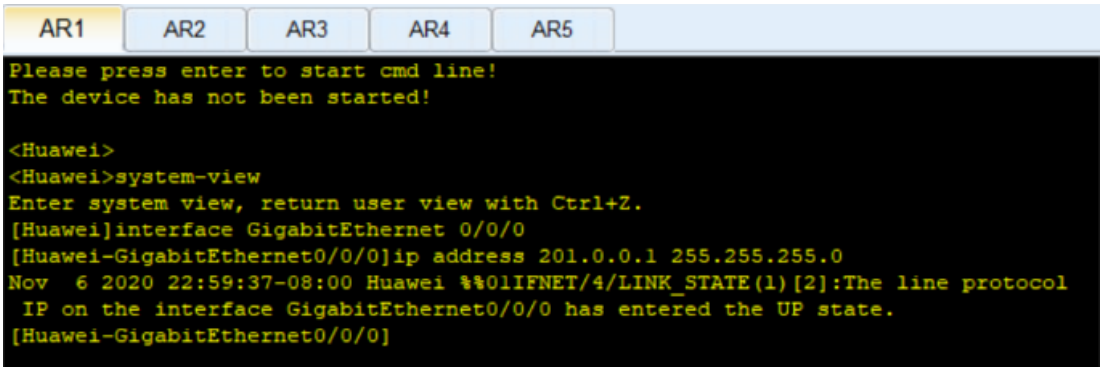


Figura 4.4

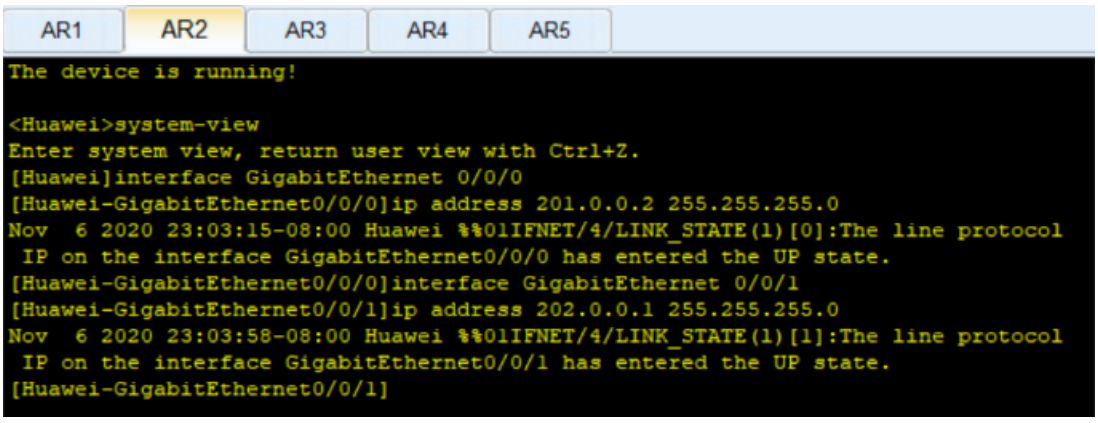


Figura 4.5

Questa operazione è stata svolta per tutti e cinque i router.

È poi necessario configurare gli indirizzi privati della lan contenente il server 1 e della lan contenente il server 2. Questo in previsione dell'implementazione del NAT internal server.

Il server implementa un'interfaccia grafica in cui è sufficiente inserire i valori nei campi Local Address, Subnet Mask, Gateway e DNS.

Di seguito uno screenshot relativo alla configurazione su server 1.

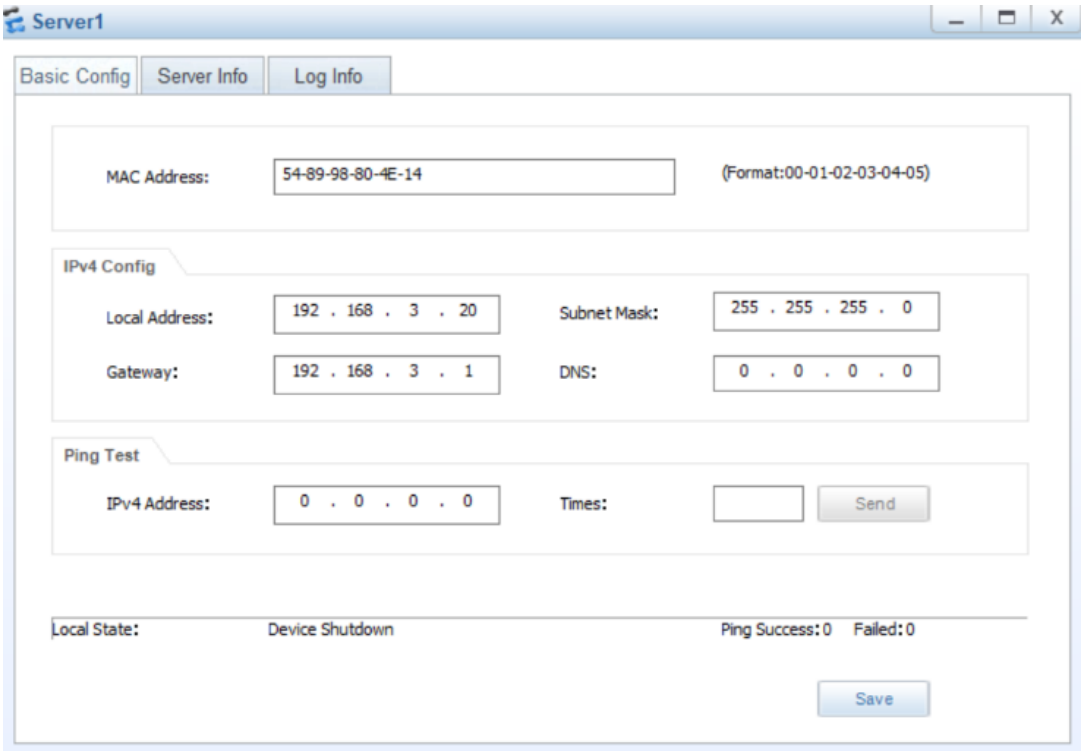


Figura 4.6

Il risultato ottenuto è sintetizzato nella seguente immagine in cui le tag con sfondo rosso rappresentano gli indirizzi pubblici delle interfacce dei router mentre le tag azzurre rappresentano gli indirizzi privati delle rispettive lan.

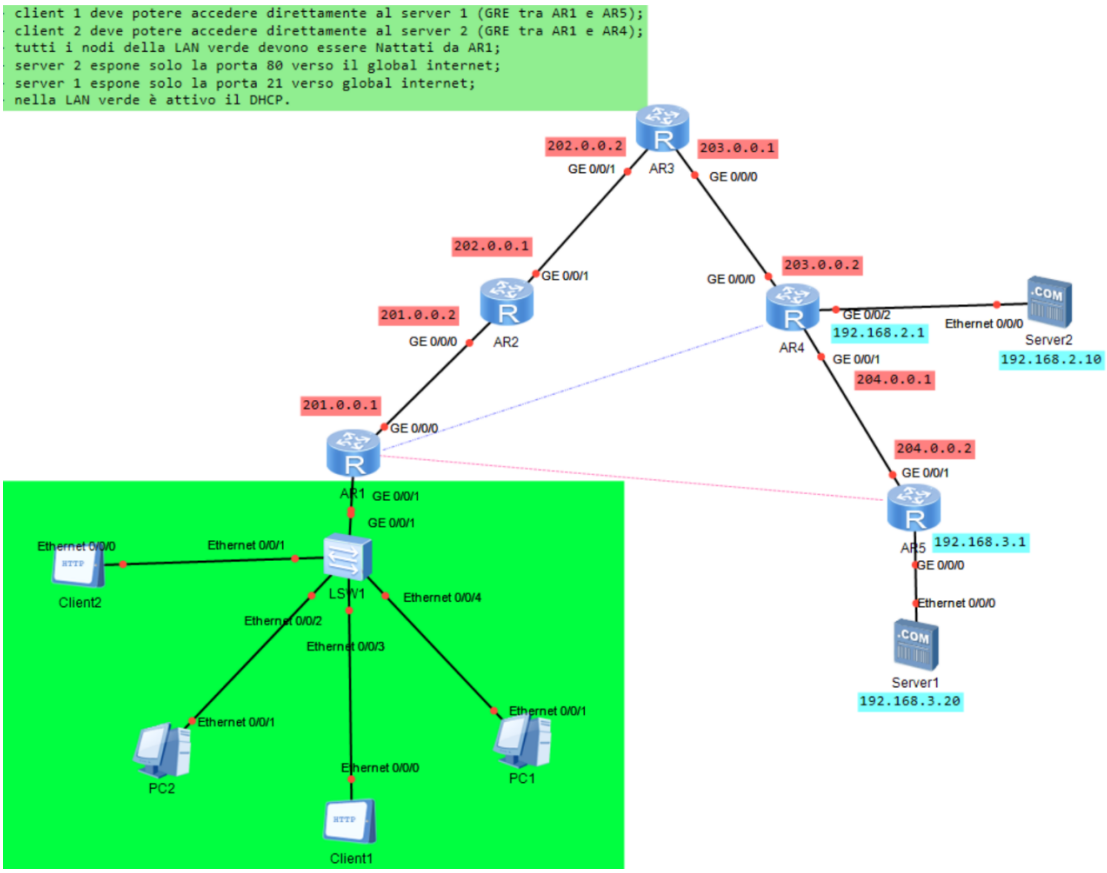


Figura 4.7

4.2 DHCP

L'assegnazione manuale di indirizzi IP richiede un certo tempo, DHCP ha lo scopo di risolvere questa problematica.

Verrà analizzato DHCP nel dettaglio e verrà applicato alla lan verde.

4.2.1 INTRODUZIONE AL PROTOCOLLO DHCP

Come visto in precedenza ad ogni dispositivo deve essere associato un indirizzo IP. Inizialmente, quando si sono sviluppate le reti private, questo processo veniva svolto manualmente e richiedeva molto tempo, specialmente laddove si doveva gestire un numero di dispositivi molto elevato.

DHCP supporta tre meccanismi per l'allocazione di indirizzi:

- Manuale: non offre particolari vantaggi, viene sfruttato solo quando si ha la necessità di imporre un certo indirizzo
- Automatico: la scelta dell'indirizzo è automatica ma viene assegnato un indirizzo permanente
- Dinamico: l'indirizzo assegnato ha validità per un periodo di tempo

Questo processo prevede alcune fasi:

Il client invia un messaggio di *DHCP Discover* ad un server DHCP che ricevuta questa richiesta invia un *DHCP Offer* in cui è presente l'offerta di un indirizzo non ancora assegnato. Il client accetta inviando un *DHCP Request* contenente sempre l'indirizzo proposto e il Server invia un *DHCP ACK* in cui, oltre all'indirizzo IP, sono contenute altre informazioni. Terminato questa fase il client invia un pacchetto ARP gratuito per verificare che all'interno della rete non vi siano dispositivi con lo stesso indirizzo. Se tutto procede correttamente il client usa l'indirizzo IP assegnato.

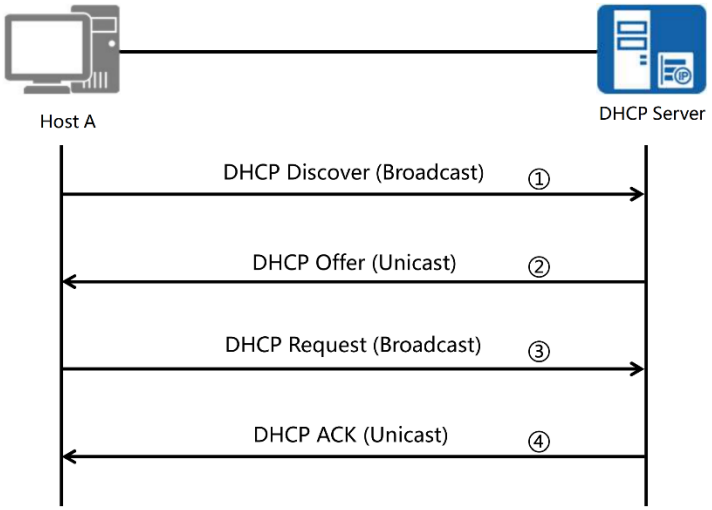


Figura 4.8

La configurazione dell' *address poll* può essere globale o basato su interfaccia. Nel primo caso la configurazione si basa su tutti gli indirizzi relativi ad un server DHCP, nel secondo caso invece si basa sul solo segmento di rete relativo ad un'interfaccia.

Nel progetto si è utilizzato il secondo approccio.

4.2.2 CONFIGURAZIONE DHCP SU LAN VERDE

Si deve abilitare DHCP nella lan verde dunque le parti fondamentali della configurazione devono essere svolte su AR1 (router 1).

Per prima cosa viene configurato l'Ip privato dell'interfaccia GE 0/0/1 con la procedura indicata nel sotto capitolo 4.1.2.

Dopo esser entrati in *system-view* viene abilitato il DHCP nel router tramite il comando *DHCP enable.* Ricordando che la configurazione viene fatta sulla base dell'interfaccia su cui si applica, si entra in modalità interfaccia e si digita *dhcp select interface.* Si termina la configurazione con i comandi *dhcp dns-list <ip address>* e *dhcp server excluded-ip-address <ip address>* che configurano il DNS-Server ed escludono il relativo indirizzo Ip durante l'assegnazione dinamica degli indirizzi.

AR1

```
<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
[Huawei]dhcp enable
Info: The operation may take a few seconds. Please wait for a moment.done.
[Huawei]interface GigabitEthernet 0/0/1
[Huawei-GigabitEthernet0/0/1]dhcp select interface
[Huawei-GigabitEthernet0/0/1]dhcp server dns-list 192.168.1.2
[Huawei-GigabitEthernet0/0/1]dhcp server excluded-ip-address 192.168.1.2
[Huawei-GigabitEthernet0/0/1]quit
[Huawei]
```

Figura 4.9

Terminata la configurazione sul router è necessario assegnare gli indirizzi a PC1, PC2, Client 1 e Client 2. Per i secondi l'assegnazione viene fatta in modo manuale compilando i campi Local Address, Subnet Mask, Gateway e DNS. Per i primi l'assegnazione viene fatta in modo dinamico scrivendo nelle relative sezioni Command il comando *ipconfig.*

Di seguito due screenshot relativi a PC1 e Client1.

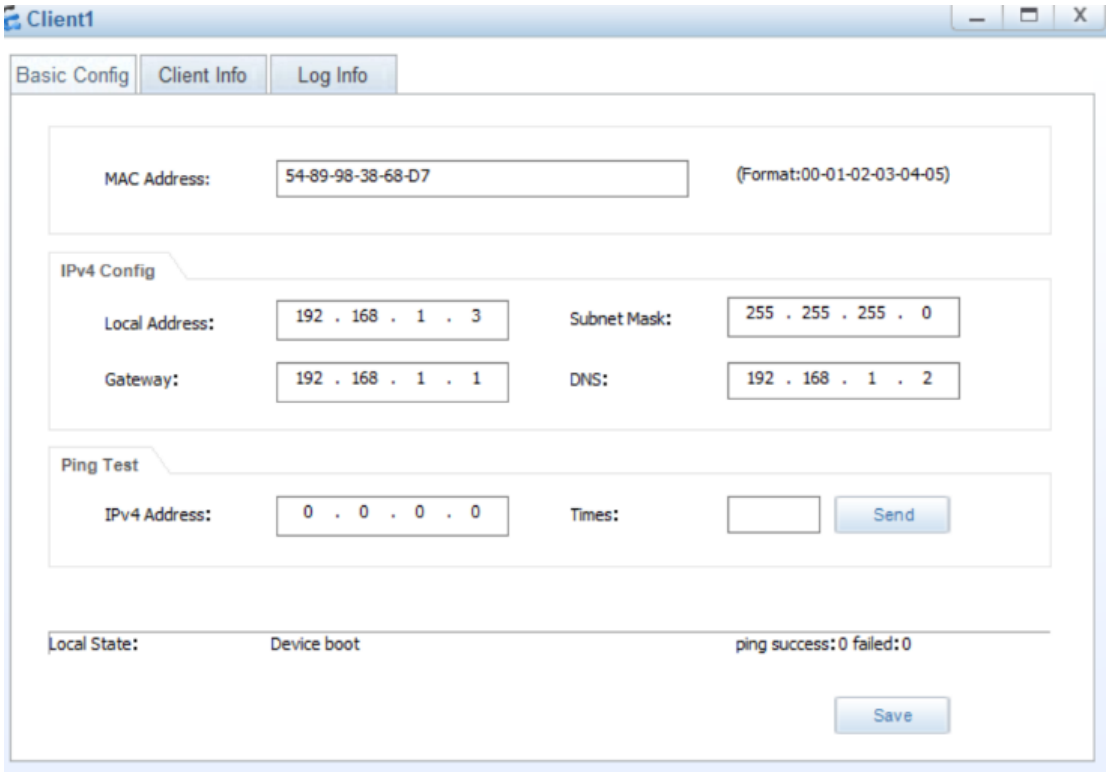


Figura 4.10

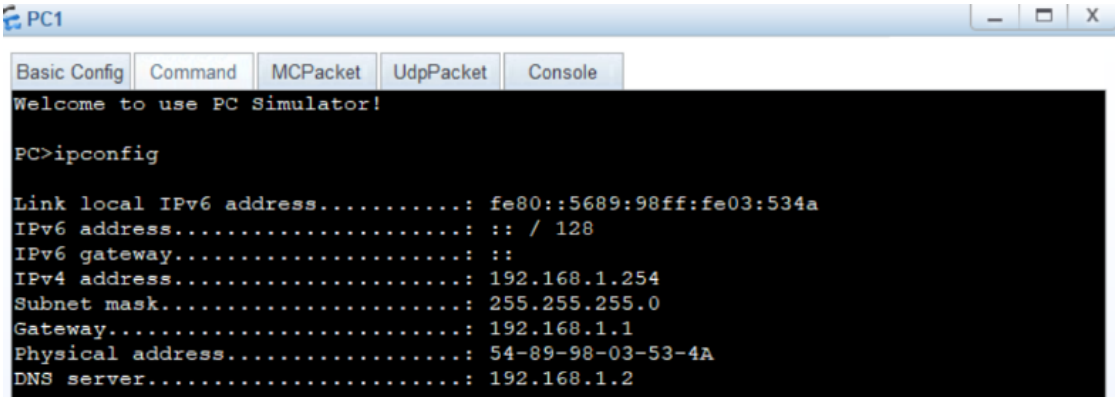


Figura 4.11

4.3 NAT DINAMICO

NAT è il protocollo che ha lo scopo di evitare lo spreco di indirizzi IPV4 che come spiegato in precedenza sono già finiti, dunque si cerca di riciclare i pochi indirizzi rimasti in attesa di IPV6. In questa sezione analizzeremo in particolare il NAT dinamico.

Si proseguirà con la configurazione di questa tecnica su AR1.

4.3.1 INTRODUZIONE AL PROTOCOLLO NAT DINAMICO

NAT consente ad uno o più host che hanno un indirizzo IP privato, pertanto che non hanno il vincolo di unicità a livello globale, di comunicare con altri host attraverso la rete internet. Vi è dunque una conversione fra indirizzi pubblici ed indirizzi privati e Huawei distingue queste due regioni come *global* e *inside*.

Il dispositivo, di livello 3, che effettua il NAT compie due operazioni:

- Modifica il contenuto del campo Ip source del pacchetto quando questo lascia la LAN;
- Modifica il contenuto del campo Ip destination del pacchetto quando questo entra nella LAN.

Di seguito un' immagine che permette di comprendere meglio quanto spiegato fino ad ora.

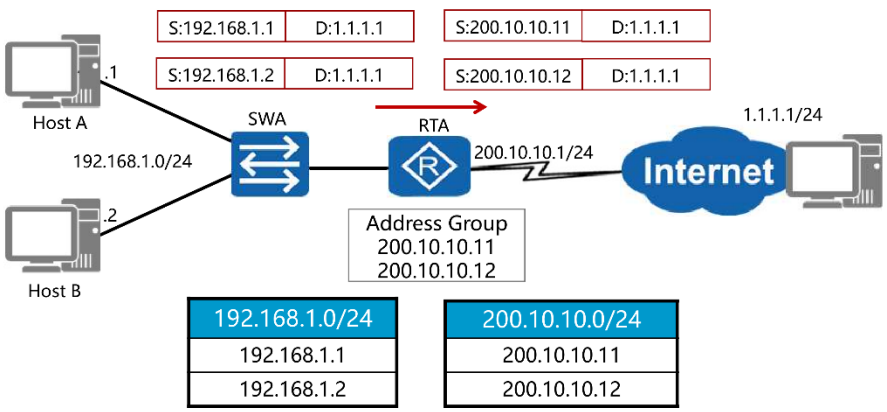


Figura 4.12

A differenza del NAT statico, in cui c'è una corrispondenza uno a uno fra indirizzi pubblici e privati, nel NAT dinamico la mappatura avviene tramite *address pools*. La conversione viene fatta all'occorrenza quindi non c'è una relazione uno-uno duratura nel tempo e generalmente il numero di indirizzi dell'address pool è inferiore al numero degli indirizzi privati.

4.3.2 CONFIGURAZIONE NAT DINAMICO SU LAN VERDE

Lo scopo è quello di creare un NAT dinamico che converta gli indirizzi privati della lan verde in indirizzi pubblici che verranno usati fuori dall'interfaccia GE 0/0/0 di AR1.

Dopo essere entrati in *system-view,* con il comando *nat address-group 2 201.0.0.10 201.0.0.11* si identifica l'access-group ed i corrispondenti indirizzi pubblici (pool). Viene poi associato un *access control lists*, con identificativo 2000, che mi abilita il traffico per gli indirizzi privati della lan verde 192.168.1.0 0.0.0.255, i rispettivi comandi sono *acl 2000* e *rule 5 permit source 192.168.1.0 0.0.0.255.* Dopo essere usciti dall' acl tramite il comando *quit* si accede all'interfaccia GE 0/0/0 tramite il comando *interface GigabitEthernet 0/0/0* e si associa ad essa l'address-group e l'acl desiderato con *nat outbound 2000 address-group 2 no-pat*. L'aggiunta del parametro no-pat sta ad indicare che non viene fatta alcuna traslazione relativa ad una porta in particolare ma ogni host viene convertito in un unico indirizzo pubblico.

Di seguito lo screen relativo all'intera configurazione.

AR1

```
<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
[Huawei]nat address-group 2 201.0.0.10 201.0.0.11
[Huawei]acl 2000
[Huawei-acl-basic-2000]rule 5 permit source 192.168.1.0 0.0.0.255
[Huawei-acl-basic-2000]quit
[Huawei]interface GigabitEthernet 0/0/0
[Huawei-GigabitEthernet0/0/0]nat outbound 2000 address-group 2 no-pat
```

Figura 4.13

Per avere una visione globale della configurazione DHCP e NAT sulla lan verde si può far riferimento all'immagine seguente. Le tag azzurre della lan verde rappresentano l'assegnazione degli indirizzi privati grazie a DHCP, mentre la tag rossa rappresenta il pool per la conversione, tramite NAT dinamico, di indirizzi privati in pubblici in uscita da GE 0/0/0 di AR1.

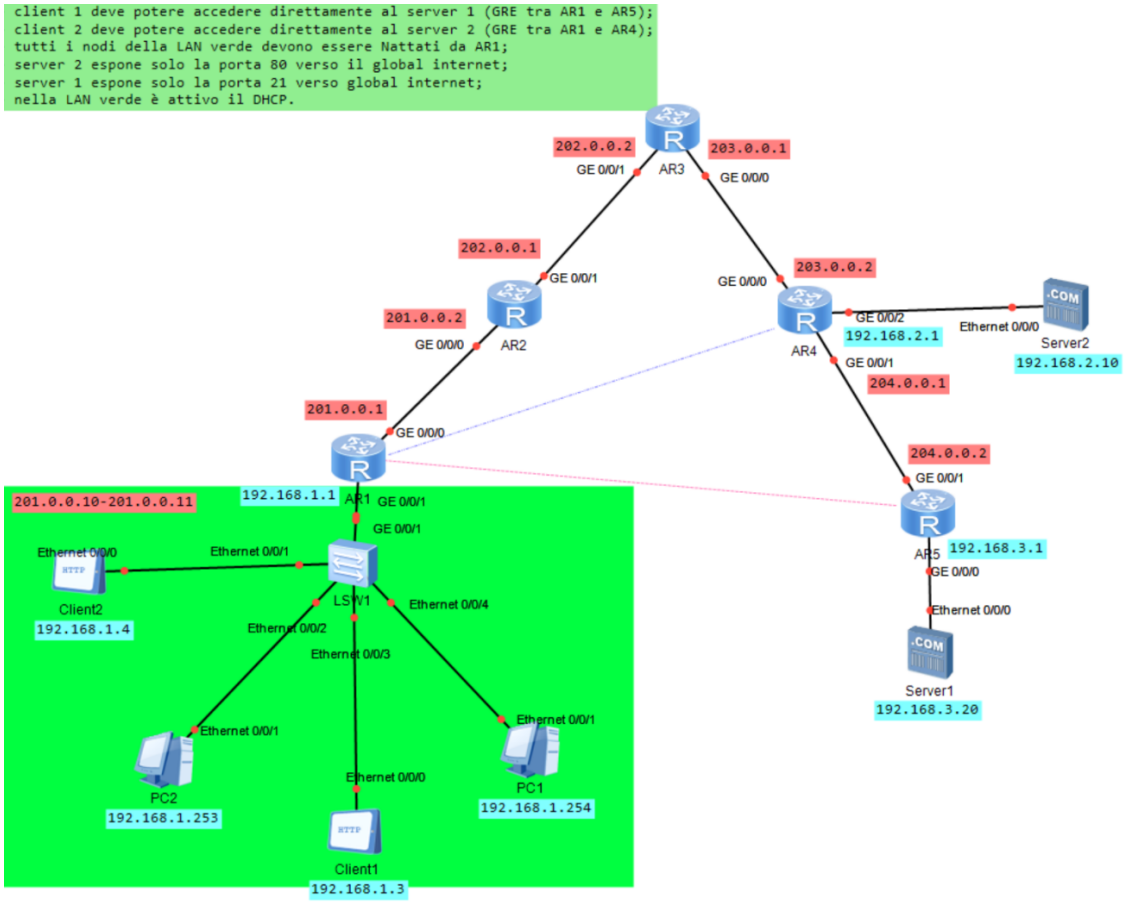


Figura 4.14

4.4 NAT INTERNAL SERVER

Oltre al NAT statico e dinamico è possibile sfruttare il NAT internal server per esporre solo particolari porte di un server al global internet. In questa sezione analizzeremo quest'ultimo aspetto e lo applicheremo a server 1 e server 2.

4.4.1 INTRODUZIONE AL PROTOCOLLO NAT INTERNAL SERVER

Quando una richiesta arriva al router in cui è implementato il NAT internal server viene fatta una conversione dell'indirizzo e della porta di destinazione nel corrispettivo indirizzo privato e nella relativa porta.

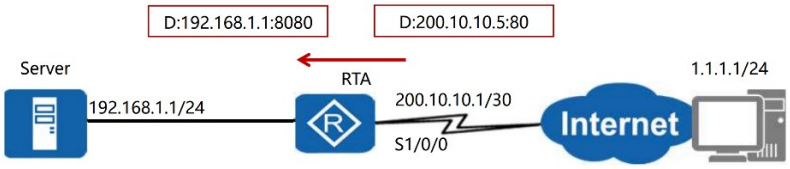


Figura 4.15

Nell'immagine precedente si può osservare come inizialmente l'indirizzo di destinazione del pacchetto fosse il 200.10.10.5 e la porta di destinazione la 80. Dopo la conversione l'indirizzo di destinazione è diventato il 192.168.1.1 e la porta la 8080.

4.4.2 CONFIGURAZIONE NAT INTERNAL SERVER RELATIVA A SERVER 1 E 2

Lo scopo della configurazione di NAT internal server su AR4 e AR5 è quello di esporre solo la porta 21 di server 1 e solo la porta 80 di server 2 al global internet. L'implementazione deve avvenire su ogni interfaccia verso il global internet quindi nel caso di AR4 è stata necessaria una configurazione sia per GE 0/0/0 che per GE 0/0/1.

Per AR4 si entra in *system-view* e sull'interfaccia GE 0/0/0, tramite il comando *interface GigabitEthernet 0/0/0.* Con il comando *nat server protocol tcp global 203.0.0.10 80 inside 192.168.2.10 80* si specifica il protocollo usato tra tcp e udp e si specificano l'indirizzo globale con la relativa porta e l'indirizzo privato corrispondente a server 2 con

la relativa porta. Un' operazione analoga avviene per l'interfaccia GE 0/0/1 ma in questo caso l'indirizzo globale è il 204.0.0.10 con porta 80.

In AR5 viene configurata l' interfaccia GE 0/0/1 con il comando *nat server protocol tcp global 204.0.0.20 21 inside 192.168.3.20 21*.

Una precisazione doverosa riguarda la scelta delle porte. Il numero di porta è formato da 16 bit che permettono un intervallo che va da 0 a 65535. I numeri di porta sono classificati in tre gruppi:

- Porte conosciute/ben note che vanno da 0 a 1023 (assegnate da IANA)
- Porte registrate che vanno da 1024 a 49151
- Porte dinamiche che vanno da 16152 a 65535

Fra le più conosciute vi sono sicuramente le porte 21 e 80 che sono assegnate rispettivamente a File Transfer Protocol (FTP) e Hypertext Transfer Protocol (HTTP). Il secondo viene ampiamente usato nel World Wide Web.

Di seguito le immagini relative alla configurazione su AR4 e AR5.

AR4 AR5

```
<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
[Huawei]interface GigabitEthernet 0/0/0
[Huawei-GigabitEthernet0/0/0]nat server protocol tcp global 203.0.0.10 80 inside
 192.168.2.10 80
[Huawei-GigabitEthernet0/0/0]interface GigabitEthernet 0/0/1
[Huawei-GigabitEthernet0/0/1]nat server protocol tcp global 204.0.0.10 80 inside
 192.168.2.10 80
```

Figura 4.16

AR4 AR5

```
The device is running!

<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
[Huawei]interface GigabitEthernet 0/0/1
[Huawei-GigabitEthernet0/0/1]nat server protocol tcp global 204.0.0.20 21 inside
 192.168.3.20 21
```

Figura 4.17

4.5 TUNNEL GRE

Spesso nasce il problema di mettere in comunicazione diretta due dispositivi appartenenti a reti diverse e la soluzione risiede nel protocollo GRE.

Analizzeremo più nel dettaglio tale protocollo e lo applicheremo alle comunicazioni fra AR1-AR4 e AR1-AR5.

4.5.1 INTRODUZIONE AL PROTOCOLLO GRE

Generic Routing Encapsulation, o GRE, è un protocollo in grado di incapsulare una grande varietà di protocolli.

GRE è un modo per impostare una connessione diretta punto-punto attraverso una rete, allo scopo di semplificare le connessioni tra reti separate.

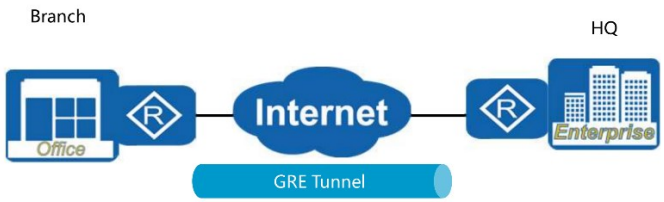


Figura 4.18

Consente inoltre l'utilizzo di protocolli che normalmente non sono supportati da una rete e risolve problemi legati ai più comuni protocolli IGP come OSPF e RIP. Nel caso di RIP infatti si supera il problema dei 15 hops massimi con l'introduzione di un Tunnel GRE.

Vediamo ora nel dettaglio il processo di *incaplulamento* e *decapsulamento* partendo dal presupposto che il GRE header viene inserito all'interno del pacchetto per la creazione del tunnel grazie al quale viene messa in piedi una rete *virtuale* sulla rete fisica esistente (si ha la percezione di passare per il tunnel ma in realtà dal punto di vista fisico i pacchetti passano comunque per tutti i router intermedi della rete).

Figura 4.19

Quando arriva un pacchetto proveniente dall'interfaccia relativa alla rete privata viene verificato l'indirizzo di destinazione e cercata l'interfaccia, nella routing table, al quale inviare il pacchetto. Se l'interfaccia in questione è il tunnel viene inviato al modulo tunnel. Dopo aver ricevuto il pacchetto, il modulo tunnel incapsula il pacchetto sulla base del tipo di protocollo utilizzato e configura i parametri relativi al tunnel GRE, aggiungendo anche un header GRE al pacchetto. Poi viene aggiunto un IP header in cui il source address è l'estremo locale del tunnel, mentre il destination address è l'indirizzo IP dell'interfaccia remota del tunnel. A questo punto il modulo IP cerca l'interfaccia pubblica su cui inoltrare il pacchetto, ricordando che il pacchetto incapsulato viaggia attraverso tutti i router intermedi. Quando il pacchetto viene ricevuto su una data interfaccia si verifica il campo Protocol Type, se è uguale a 47 è in uso il protocollo GRE. Vengono rimossi l' IP header ed il GRE header e consegnato poi il pacchetto al protocollo, indicato nel GRE header, che è di fatto quello attivo sulla rete privata.

Opzionalmente il GRE prevede un campo per l'*autenticazione* ma non verrà sfruttata questa caratteristica nel progetto.

Ultimo aspetto da approfondire riguarda la funzione di *GRE keepalive* la quale viene utilizzata per rilevare se il collegamento del tunnel è nello stato keepalive, cioè se il peer del tunnel è raggiungibile. Dopo che la funzione keepalive è stata abilitata, l'estremità locale del tunnel GRE invia periodicamente un pacchetto di rilevamento keepalive al peer. Se il peer è raggiungibile l'estremità locale riceve un pacchetto di risposta dal peer, altrimenti se dopo un certo intervallo di tempo non rileva una risposta viene chiusa la connessione tramite tunnel.

4.5.2 CONFIGURAZIONE DI DUE TUNNEL GRE

L'obiettivo all'interno del progetto è di creare un tunnel GRE fra AR1 e AR4 in modo da permettere a client 1 di accedere direttamente a server 1 ed un tunnel GRE fra AR1 e AR5 in modo da permettere a client 2 di accedere direttamente a server 2.

Dopo essere entrati nell'interfaccia del router ed in *system-view* si crea e si accede all'interfaccia del tunnel tramite *interface Tunnel x/x/x* in cui x/x/x è l'identificativo del tunnel. Poi tramite *tunnel-protocol gre* viene specificato il protocollo usato per l'incapsulamento ovvero GRE. Vengono dunque specificati gli indirizzi della sorgente e

di destinazione con i comandi *source <tunnel source address>* e *destination <tunnel destination address>*. Le rotte che il tunnel è in grado di supportare, quindi è possibile inoltrare pacchetti da e verso questi indirizzi, sono specificate dal comando *ip route-static <addresses> <subnet mask> Tunnel x/x/x*. La rotta può essere statica o dinamica.

Una configurazione simile deve essere fatta sull'altro router ma invertendo il source ed il destination address.

Di seguito le configurazioni su AR4 e AR5.

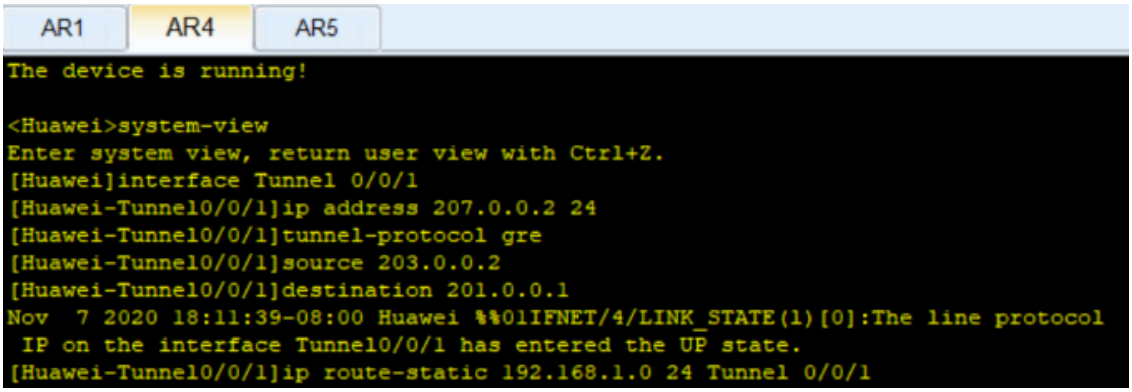


Figura 4.20

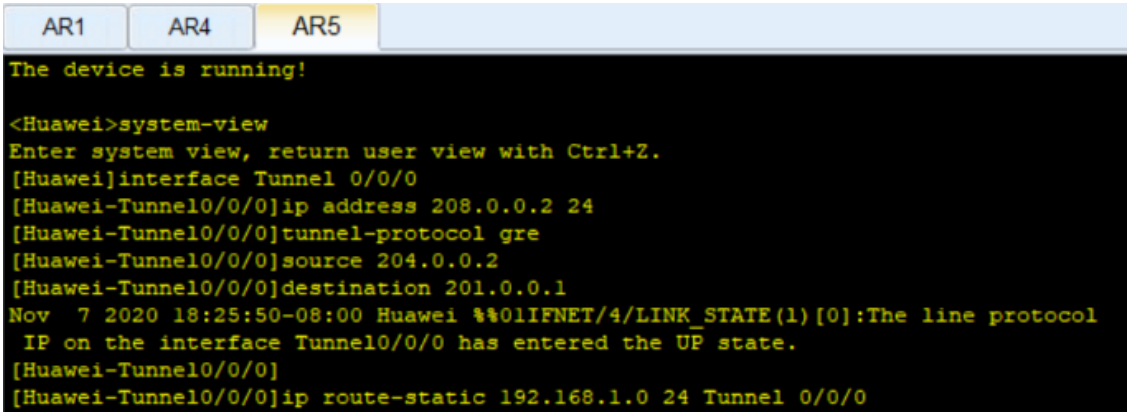


Figura 4.21

Per AR1 la configurazione dei due Tunnel è simile:

- per Tunnel 0/0/0 con AR5 l'ip address risulta pari a 208.0.0.1 24, si invertono gli indirizzi di source e destination e la rotta statica è la 192.168.3.0 24
- per Tunnel 0/0/1 con AR4 l'ip address risulta pari a 207.0.0.1 24, si invertono gli indirizzi di source e destination e la rotta statica è la 192.168.2.0 24

Di seguito si può apprezzare una visione globale del progetto ponendo particolare attenzione sulle tag rosse di server 1 e 2 che rappresentano gli indirizzi pubblici e le porte esposte al global internet ottenute grazie a NAT internal server. Poi grazie alle tag verdi si possono osservare gli indirizzi delle interfacce Tunnel.

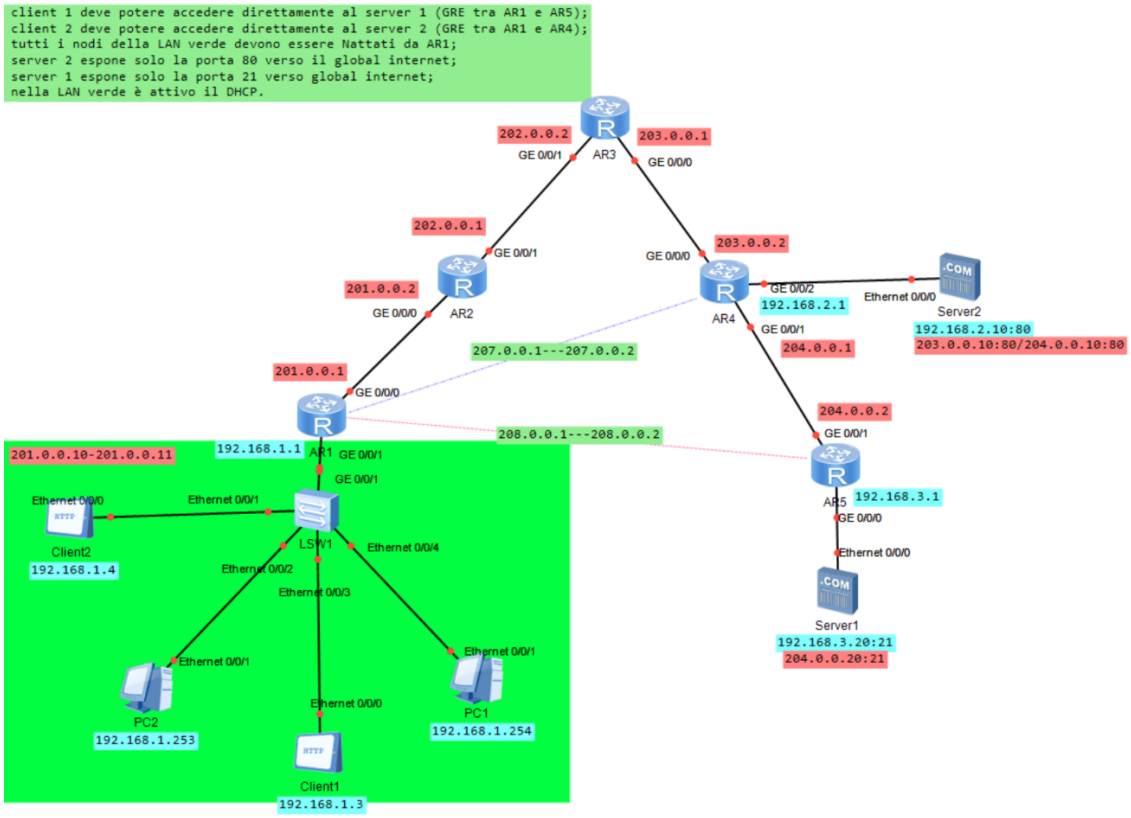


Figura 4.22

4.6 OSPF

Rimane da trattare OSPF, uno dei protocolli più importanti per l'instradamento dei pacchetti e per il calcolo del percorso a costo minore.

Il protocollo OSPF sarà implementato in due diverse situazioni: per garantire la raggiungibilità tra i router e per veicolare i pacchetti tra le LAN che fanno parte dei tunnel.

4.6.1 INTRODUZIONE AL PROTOCOLLO OSPF

I router sono i principali dispositivo a livello di rete utilizzati per definire il gateway di ciascuna rete locale e abilitare la segmentazione della rete IP. Funzionano come mezzi per instradare i pacchetti da una rete locale all'altra, di fondamentale importanza è il concetto di indirizzamento IP. Una politica di instradamento può essere intesa come un insieme di regole che determinano la modalità di gestione del traffico all'interno di un sistema autonomo. Quando un router ha più percorsi verso una determinata destinazione, grazie ad algoritmi, viene determinato il percorso migliore, dunque anche il router successivo a cui inoltrare il pacchetto. Le decisioni che governano il percorso da prendere possono variare a seconda del protocollo di instradamento in uso.

Il dispositivo riceve un frame su una delle sue interfacce. Viene controllato se FCS è corretto e viene controllato il destination mac. Viene poi analizzato il campo protocol in cui troviamo l'indicazione 0x0800 (IP). Perciò il dispositivo capisce che il contenuto del campo data deve essere inviato al layer 3 IP per potere essere correttamente processato. Il campo data viene decapsulato e processato con le funzioni IP. Il layer 3 riceve quindi una stringa di bit e identifica l'header. All'interno dell'header viene individuato il destination ip sulla base del quale saranno prese le scelte di inoltro. Il destination ip viene confrontato con la tabella di routing in cui sono presenti gli spazi di indirizzi raggiungibili in modo diretto o attraverso un ulteriore nodo intermediario. Se il pacchetto è indirizzato ad uno spazio di indirizzi sconosciuto dal router, può essere scartato o inoltrato ad un altro router sperando che quest'ultimo lo contenga nella tabella quello spazio di indirizzi.

Di seguito un screenshot ricavato da AR1 che mostra la tabella di routing

AR1

```
[Huawei]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 30        Routes : 30

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        1.1.1.0/24  Direct  0    0           D   1.1.1.1         LoopBack0
        1.1.1.1/32  Direct  0    0           D   127.0.0.1       LoopBack0
      1.1.1.255/32  Direct  0    0           D   127.0.0.1       LoopBack0
        2.2.2.2/32  OSPF    10   1           D   201.0.0.2       GigabitEthernet
0/0/0
        3.3.3.3/32  OSPF    10   2           D   201.0.0.2       GigabitEthernet
0/0/0
        4.4.4.4/32  OSPF    10   3           D   201.0.0.2       GigabitEthernet
0/0/0
        5.5.5.5/32  OSPF    10   4           D   201.0.0.2       GigabitEthernet
0/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
    192.168.1.0/24  Direct  0    0           D   192.168.1.1     GigabitEthernet
0/0/1
    192.168.1.1/32  Direct  0    0           D   127.0.0.1       GigabitEthernet
0/0/1
  192.168.1.255/32  Direct  0    0           D   127.0.0.1       GigabitEthernet
0/0/1
    192.168.2.0/24  Static  60   0           D   207.0.0.1       Tunnel0/0/1
```

Figura 4.23

- *Proto* indica il protocollo grazie al quale si è venuti a conoscenza di un indirizzo.
- *Pre* indica l'affidabilità del protocollo che ha comportato l'inserimento della rotta nella routing table
- *Cost* è la metrica usata per decidere il percorso quando ho più alternative di pari preferenza
- *Interface* indica l'interfaccia da usare per raggiungere la destinazione

Tutti i percorsi possibili per raggiungere una rete di destinazione sono analizzati e confrontati tra loro, il migliore viene inserito nella tabella di routing sulla base di alcuni fattori come la lunghezza del segmento o la larghezza di banda. Un collegamento con una velocità maggiore rappresenta un valore di costo inferiore, consentendo di preferire un percorso rispetto a un altro.

Vista la complessità che una rete può raggiungere i router scambiano tra loro solo le informazioni necessarie, permettendo di risparmiare tempo e risorse. Dopodiché in base

alle informazioni raccolte e alla metrica utilizzata viene presa una decisione di instradamento.

Alcuni punti di forza di OSPF sono:

- Traffico di informazioni minimo fra router
- Convergenza rapida
- Scalabilità
- Metrica accurata

La convergenza di OSPF richiede che ogni router che esegue attivamente il protocollo OSPF abbia conoscenza dello stato di tutte le interfacce e le relazioni tra i router a cui sono collegati, al fine di stabilire il percorso migliore per ogni rete. Inizialmente si sfruttano le Link State Advertising (LSA) che sono unità di dati contenenti informazioni come reti note e stati dei collegamenti per ciascuna interfaccia all'interno di un dominio di routing. Ogni router utilizzerà l'LSA ricevuto per creare un database di stato dei collegamenti (LSDB) che fornisce le basi per stabilire l'albero dei percorsi più brevi per ciascuna rete, i cui percorsi vengono infine incorporati nella tabella di instradamento IP.

Un router-id è un valore a 32 bit utilizzato per identificare ogni router che esegue il protocollo OSPF.

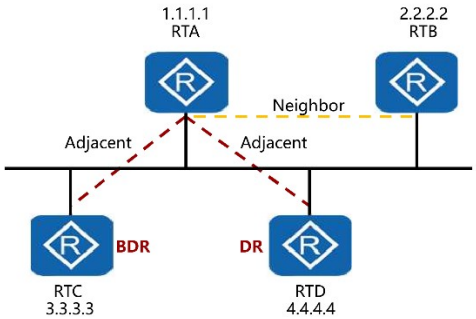


Figura 4.24

Al fine di indirizzare e ottimizzare la comunicazione di OSPF su reti di tipo broadcast o NBMA, OSPF implementa un router designato (DR) che funge da punto centrale di comunicazione per tutti gli altri router. Senza l'utilizzo di un DR si avrebbero n*(n-1)/2 interazioni. Viene implementato anche un BDR ovvero un router che dovrà sostituire il DR in caso di guasto, dunque tutti i router vicini devono comunicare anche con esso.

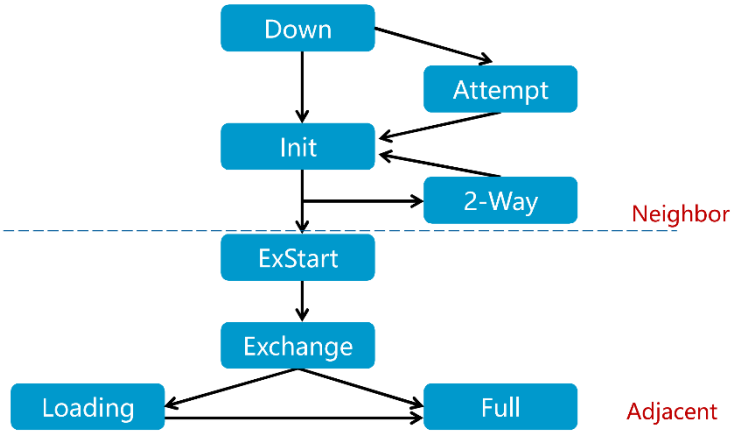
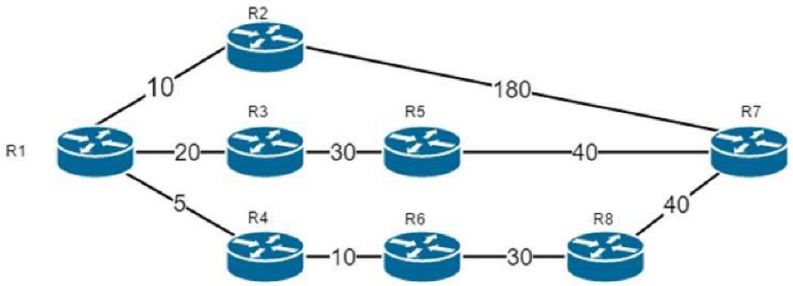


Figura 4.25

Ogni router che partecipa a OSPF passerà attraverso un numero di stati di collegamento per ottenere uno stato *neighbor* o uno stato *adjacent*. Tutti i router iniziano nello stato *down* al momento dell'inizializzazione e passano attraverso un processo di rilevamento dei vicini, che prevede innanzitutto di rendere nota la presenza di un router nella rete OSPF tramite un pacchetto Hello. Eseguendo questa azione il router passerà a uno stato *init*. Una volta che il router riceve una risposta sotto forma di un pacchetto Hello contenente l'ID router del router che a sua volta riceve la risposta, verrà raggiunto uno stato *2-Way* e verrà creata una relazione di vicinato. I router che non raggiungono una relazione adjacent rimarranno in uno stato neighbor con uno stato di comunicazione *2-Way*. Router come DR e BDR creeranno uno stato neighbor adjacent con tutti gli altri router adiacenti e pertanto scambiano informazioni per stabilire un database completo. I router di peering che stabiliscono un'adiacenza negoziano lo scambio di informazioni sullo stato del collegamento (*ExStart*). Una relazione completamente sincronizzata tra vicini è determinata dallo stato *full* in cui entrambi i router di peering possono essere considerati adiacenti.

In questa sede non approfondiremo molti altri aspetti di OSPF però è interessante comprendere la metrica adottata. Il costo di un'interfaccia viene calcolata in base alla larghezza di banda dell'interfaccia stessa attraverso la formula: costo = valore di riferimento della larghezza di banda / larghezza di banda. Il valore di riferimento è impostato a 100Mbps ma si può cambiare, in particolare si consiglia di aumentarlo laddove siano supportate velocità superiori a quella standard di riferimento. Questo perché il costo varia da 1 a 65535.

Nella figura successiva è rappresentato un esempio in cui viene calcolato il percorso a costo minore.



Path : R1 -> R7	Costo
Via R2	190
Via R3	90
Via R4	85

Figura 4.26

Infine, si contrappone il concetto di *Single Area* a quello di *Multi Area*. Nel primo caso avremo una sola area, dunque un singolo database per tutto il dominio. Nel secondo caso si hanno LSDB diversi per ciascuna area, questa implementazione è necessaria quando la complessità della rete diventa importante.

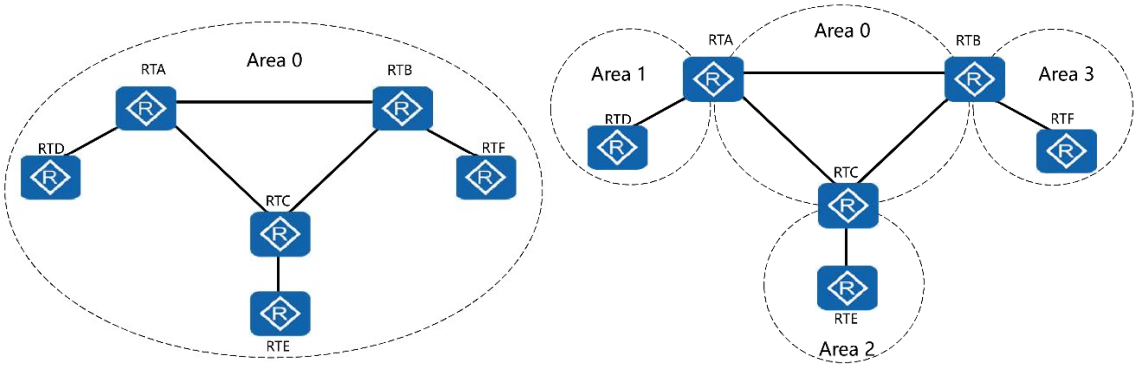


Figura 4.27

4.6.2 CONFIGURAZIONE DI OSPF1 E OSPF2

Come anticipato in precedenza l' obiettivo è di creare OSPF 1 che lavori nei "normali" collegamenti fra router ed OSPF 2 che si occupi dell'instradamento quando è richiesto il passaggio per i Tunnel.

Entrati in *system-view* si configura il router id tramite il comando *router id <router-id>*. Dunque, si entra in uno specifico process-id, che di fatto identifica una precisa logica grazie alla quale avvengono associazioni fra router, tramite il comando ospf <process-id>. Nel progetto viene sfruttato questo aspetto per creare un OSPF 1 ed un OSPF 2, il primo serve garantire la raggiungibilità tra i router ed il secondo per veicolare i pacchetti tra le LAN che fanno parte dei tunnel. Con *area 0* si specifica semplicemente l'area di appartenenza dei networks che verranno precisati in seguito con il comando *network <network-ip> <wildcare-mask>*. La wildcard mask utilizza i bit 0 e 1 per identificare i singoli indirizzi IP o un gruppo di essi in un modo molto più flessibile rispetto alla rigidità della subnet mask. Se nella wildcare mask il bit è a 0 si controlla il bit corrispondente dell'indirizzo IP, se è a 1 si ignora.

Di seguito uno screen della configurazione di ospf 1 area 0 su AR1, nell'immagine viene anche configurata l'interfaccia di loopback0.

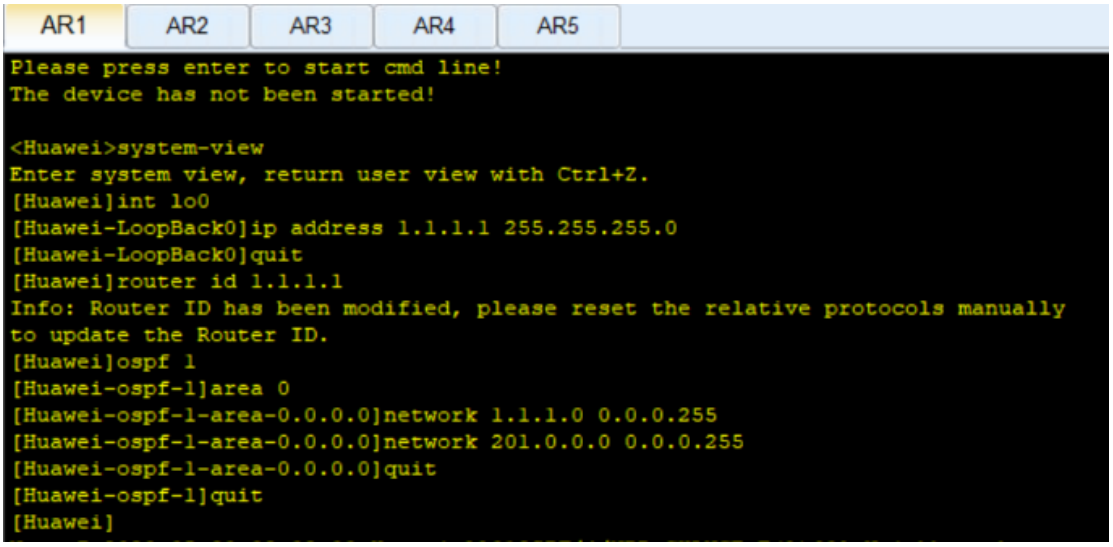


Figura 4.28

Un breve riassunto delle reti aggiunte a ciascun process-id.

OSPF 1 AREA 0

- AR1 : 201.0.0.0 0.0.0.255
- AR2 : 201.0.0.0 0.0.0.255 | 202.0.0.0 0.0.0.255
- AR3 : 202.0.0.0 0.0.0.255 | 203.0.0.0 0.0.0.255
- AR4 : 203.0.0.0 0.0.0.255 | 204.0.0.0 0.0.0.255
- AR5 : 204.0.0.0 0.0.0.255

OSPF 2 AREA 0

- AR1 : 192.168.1.0 0.0.0.255 | 208.0.0.0 0.0.0.255 | 207.0.0.0 0.0.0.255
- AR4 : 192.168.2.0 0.0.0.255 | 207.0.0.0 0.0.0.255
- AR5 : 192.168.3.0 0.0.0.255 | 208.0.0.0 0.0.0.255

5 TEST

In questo capitolo vedremo alcuni comandi e alcuni screenshot di Wireshark che permettono di verificare se le varie configurazioni sono avvenute come previsto.

Il primo comando che viene preso in esame è *display interface GigabitEthernet x/x/x* che permette di visualizzare i parametri di un' interfaccia, in particolare sono presenti l'indirizzo Ip e lo stato del protocollo usato.

Prendiamo come esempio la porta GigabitEthernet 0/0/0 di AR1, ma è possibile eseguire questo comando per tutte le porte dei Router.

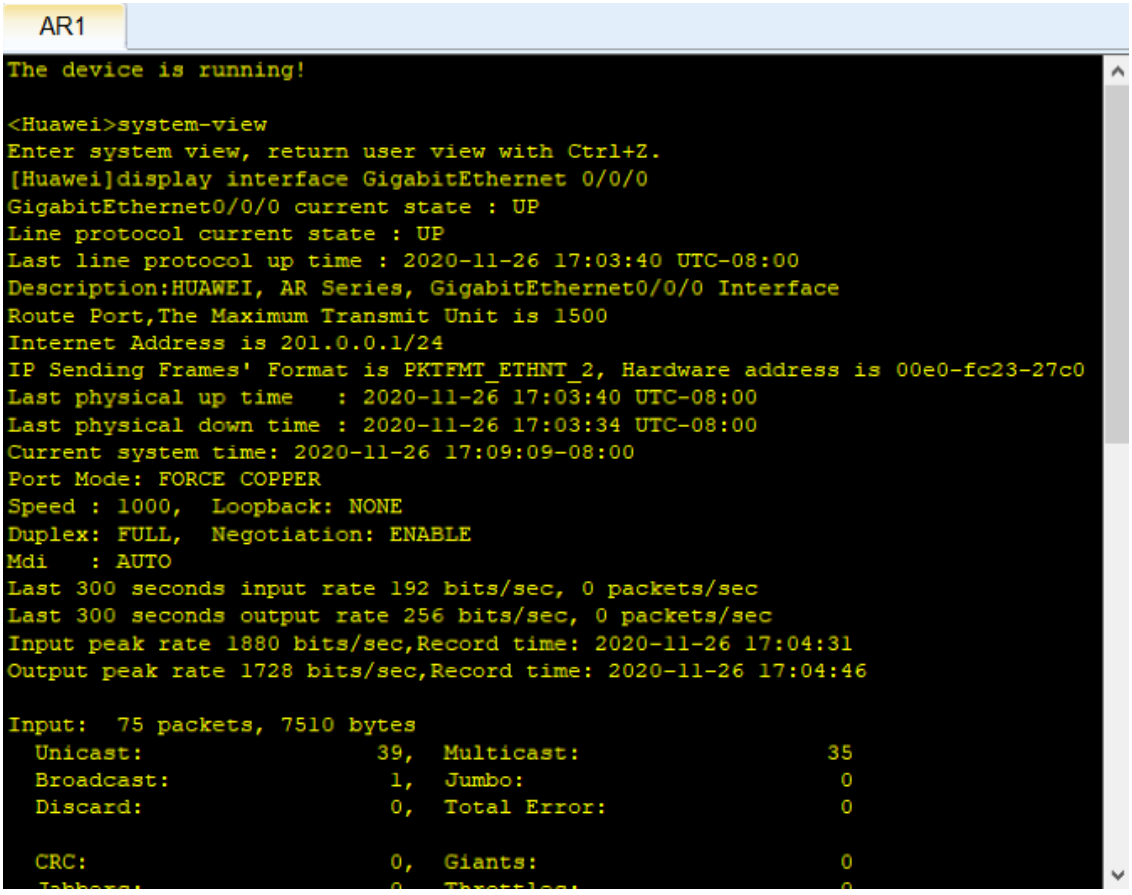


Figura 5.1

Un altro comando riguardante le interfacce è *display interface brief*. Si ha in questo modo una visione globale dello stato delle interfacce di un router. Nell'immagine di seguito (ricavata da AR1) possiamo osservare i campi PHY, che indica lo stato fisico dell'interfaccia, Protocol, che indica lo stato del protocollo riguardante il collegamento, e molti altri. Si può osservare che il risultato ottenuto è coerente con le specifiche del

progetto in quanto sono presenti le tre interfacce GigabitEthernet, una Loopback e i due Tunnels.

AR1

```
<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
[Huawei]display interface brief
PHY: Physical
*down: administratively down
(l): loopback
(s): spoofing
(b): BFD down
^down: standby
(e): ETHOAM down
(d): Dampening Suppressed
InUti/OutUti: input utility/output utility
Interface                   PHY   Protocol InUti OutUti   inErrors  outErrors
GigabitEthernet0/0/0        up    up          0%     0%          0          0
GigabitEthernet0/0/1        up    up          0%     0%          0          0
GigabitEthernet0/0/2        down  down        0%     0%          0          0
LoopBack0                   up    up(s)       0%     0%          0          0
NULL0                       up    up(s)       0%     0%          0          0
Tunnel0/0/0                 up    up          --     --          0          0
Tunnel0/0/1                 up    up          --     --          0          0
[Huawei]
```

Figura 5.2

Per verificare la configurazione di DHCP sulla lan verde si fa uso del comando *display ip pool interface GigabitEthernet0/0/1*. Si possono verificare parametri come Lease, se settato ad un valore diverso da quello di default, l'indirizzo del DNS-server, l'indirizzo del Gateway ed il range di indirizzi utilizzabili. La figura rappresenta la configurazione DHCP su AR1.

AR1

```
[Huawei]display ip pool interface GigabitEthernet0/0/1
  Pool-name      : GigabitEthernet0/0/1
  Pool-No        : 0
  Lease          : 1 Days 0 Hours 0 Minutes
  Domain-name    : -
  DNS-server0    : 192.168.1.2
  NBNS-server0   : -
  Netbios-type   : -
  Position       : Interface       Status          : Unlocked
  Gateway-0      : 192.168.1.1
  Mask           : 255.255.255.0
  VPN instance   : --
 ------------------------------------------------------------------------------
        Start           End     Total  Used  Idle(Expired)  Conflict  Disable
 ------------------------------------------------------------------------------
    192.168.1.1   192.168.1.254   253     2        250(0)         0         1
 ------------------------------------------------------------------------------
```

Figura 5.3

Con il comando *display nat address-group 2* si attengono gli indirizzi del pool del NAT Dinamico nella lan verde. Come previsto gli indirizzi del pool sono solo due.

Utile è anche il comando *display nat outbound* grazie al quale è possibile ottenere ulteriori dettagli della configurazione dinamica di NAT. Nel caso specifico abbiamo che per l'interfaccia GigabitEthernet 0/0/0 è stata applicata l' acl 2000, l' address-group è il 2 ed è specificato il parametro no-pat.

AR1

```
<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
[Huawei]display nat address-group 2

 NAT Address-Group Information:
 --------------------------------------
 Index   Start-address      End-address
 --------------------------------------
 2          201.0.0.10       201.0.0.11
 --------------------------------------
  Total : 1
[Huawei]display nat outbound
 NAT Outbound Information:
 --------------------------------------------------------------------------
 Interface                    Acl     Address-group/IP/Interface      Type
 --------------------------------------------------------------------------
 GigabitEthernet0/0/0         2000                             2    no-pat
 --------------------------------------------------------------------------
  Total : 1
```

Figura 5.4

Per il NAT internal server esiste un comando specifico ovvero *display nat server*. I parametri di interesse che possiamo visualizzare sono Global Ip/Port e Inside Ip/Port che ci permettono di comprendere come avviene la conversione di indirizzi e porte.

Nel progetto può essere applicato ad AR4 e AR5 in modo da verificare la conversione che avviene per Server 1 e 2.

Di seguito uno sceenshot da AR4 in cui è possibile osservare come vi sia una configurazione per ogni interfaccia verso il global internet. Per AR5 il discorso è analogo ma con una sola interfaccia verso il global internet.

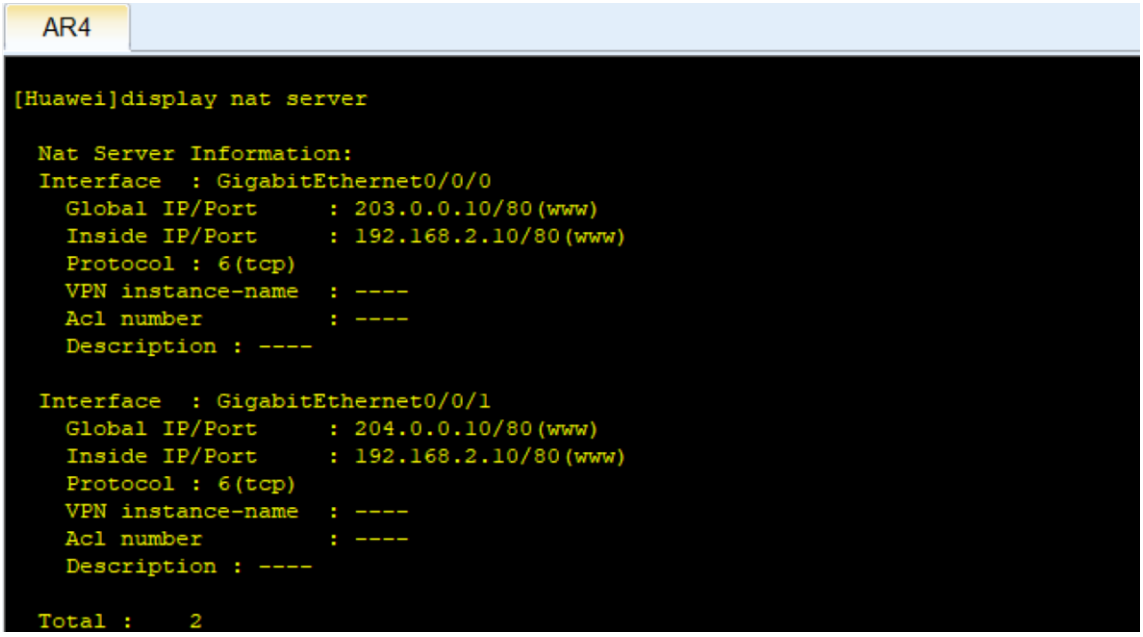


Figura 5.5

Una verifica può essere fatta anche sulla configurazione dei Tunnels GRE tramite *interface Tunnel x/x/x*. È possibile visualizzare lo stato del Tunnel, lo stato del protocollo usato sulla linea, l'indirizzo Ip dell' interfaccia Tunnel, la source ed il destination del tunnel e molti altri parametri. La figura mostra la configurazione di Tunnel 0/0/1 su AR1, come previsto l'indirizzo dell'interfaccia Tunnel è la 207.0.0.1/24, la source 201.0.0.1 ed il destination 203.0.0.2. Questa verifica viene fatta per tutte le altre interfacce dei router coinvolti.

AR1

```
[Huawei]display interface Tunnel 0/0/1
Tunnel0/0/1 current state : UP
Line protocol current state : UP
Last line protocol up time : 2020-11-26 17:04:28 UTC-08:00
Description:HUAWEI, AR Series, Tunnel0/0/1 Interface
Route Port,The Maximum Transmit Unit is 1500
Internet Address is 207.0.0.1/24
Encapsulation is TUNNEL, loopback not set
Tunnel source 201.0.0.1 (GigabitEthernet0/0/0), destination 203.0.0.2
Tunnel protocol/transport GRE/IP, key disabled
keepalive disabled
Checksumming of packets disabled
Current system time: 2020-11-27 05:16:20-08:00
    300 seconds input rate 0 bits/sec, 0 packets/sec
    300 seconds output rate 56 bits/sec, 0 packets/sec
    0 seconds input rate 0 bits/sec, 0 packets/sec
    0 seconds output rate 0 bits/sec, 0 packets/sec
    0 packets input,  0 bytes
    0 input error
    3992 packets output,  351296 bytes
    0 output error
    Input bandwidth utilization  : --
    Output bandwidth utilization : --
```

Figura 5.6

Per OSPF con il comando *display ospf peer* vengono visualizzati tutti i processi OSPF attivi su un router, dunque anche tutti i vicini. Nello screenshot si può vedere come AR2 abbia un Router id pari a 2.2.2.2 e comunichi con AR1 tramite l'interfaccia GE 0/0/0 e con AR3 tramite l'interfaccia GE 0/0/1. In modo analogo si possono vedere le configurazioni fatte su tutti i router.

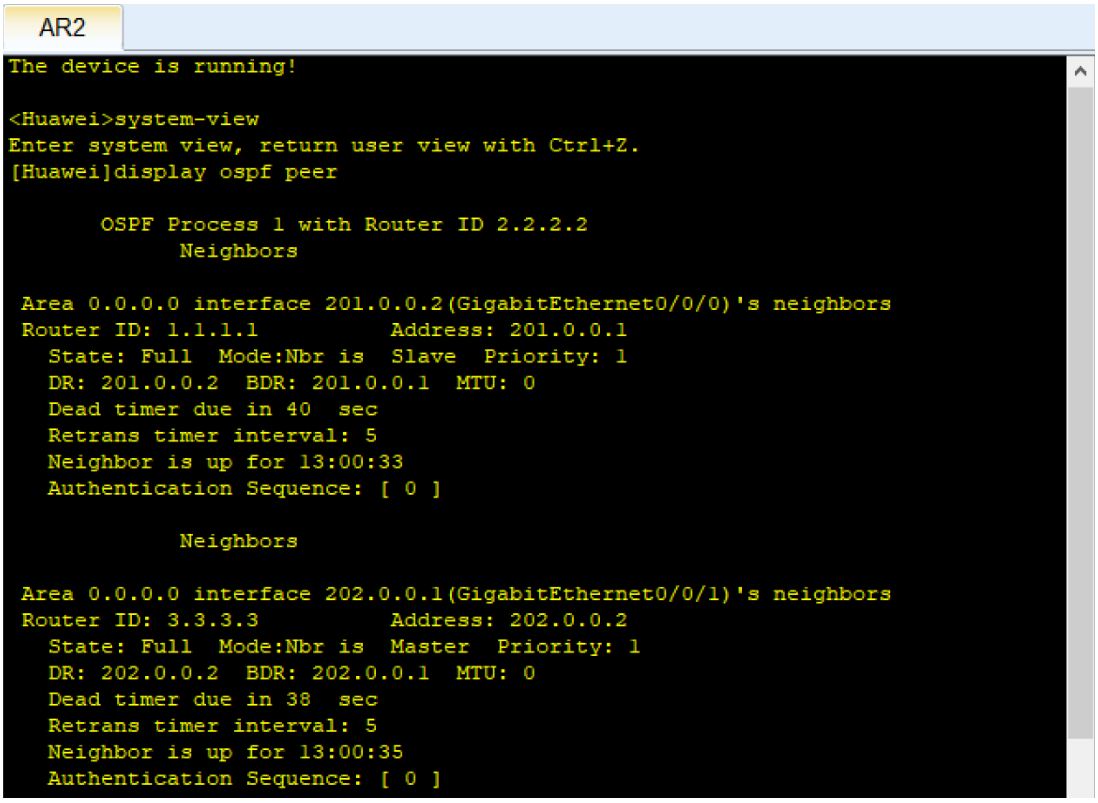


Figura 5.7

Ora si vuole verificare che operazioni come ping e tracert o servizi come ftp e http funzionino correttamente.

Come prima prova si esegue un ping da parte di PC1 verso GE 0/0/1 di AR5 che ha indirizzo 204.0.0.2. Come possiamo vedere nell'immagine seguente è stato sfruttato il comando *ping <address>*, vediamo come è stato possibile raggiungere la destinazione. Inoltre, il ttl (Time To Live) è decrementato di tante unità quanti sono i router che sono stati attraversati.

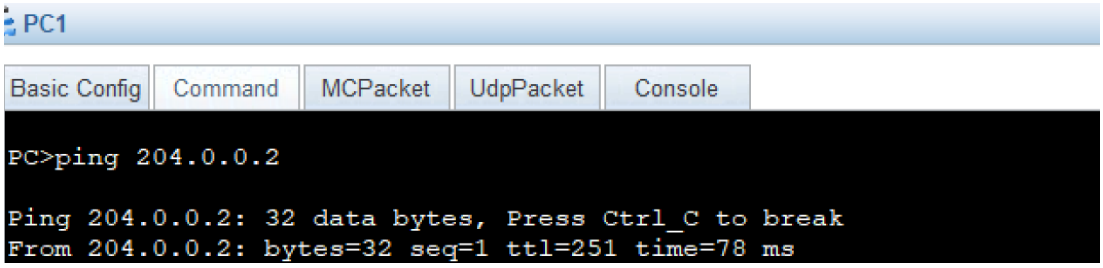


Figura 5.8

Con una cattura di Wildshark sulle interfacce GE 0/0/0 e GE 0/0/1 di AR1 si può apprezzare come il router converta l'indirizzo privato di PC1 in un indirizzo pubblico da 192.168.1.254 a 201.0.0.10, prova del corretto funzionamento del NAT dinamico. La prima immagine riguarda la cattura su GE 0/0/1, la seconda su GE 0/0/0.

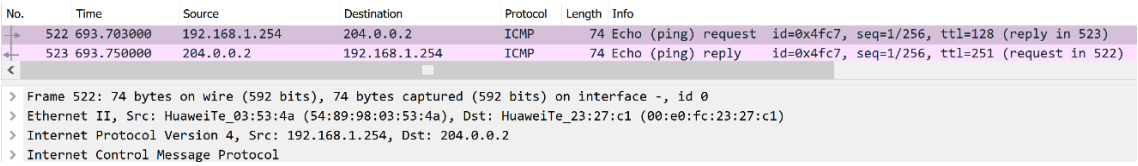


Figura 5.9

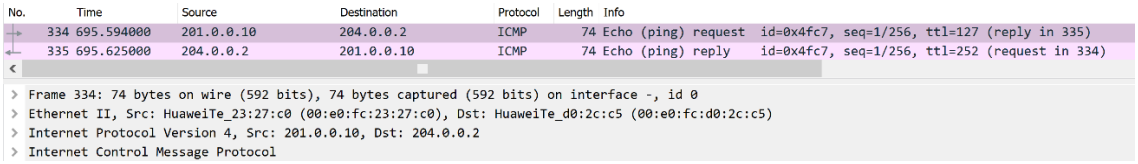


Figura 5.10

Una prova più accurata sul percorso che viene compiuto può essere fatta con il comando *tracert <address>*. Lanciando tracert 204.0.0.2 su AR1 si ha la conferma che il percorso effettivo corrisponde con quello previsto.

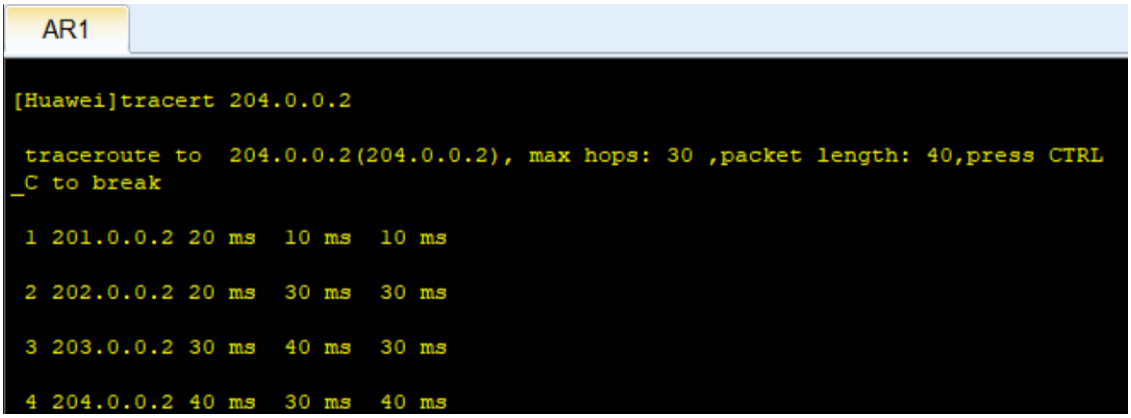


Figura 5.11

Si può applicare anche questo comando per verificare il corretto funzionamento del Tunnel GRE. Per semplicità viene eseguito su PC1 verso Server 1 ma con il NAT internal server di AR4 non configurato. Si noti il passaggio per 207.0.0.2, indirizzo dell'interfaccia tunnel di AR4. Considerazioni analoghe sono valide per il Tunnel con AR5.

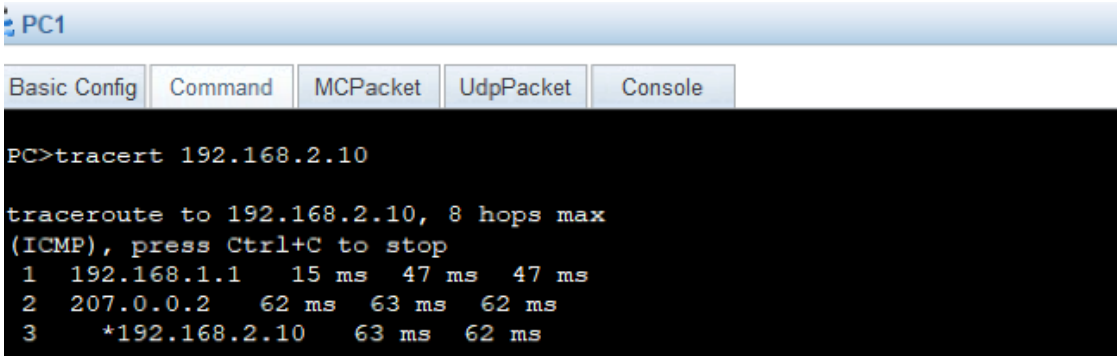


Figura 5.12

Rimane da verificare il corretto funzionamento dei servizi offerti da Server 1 e 2 ovvero ftp e http. Prendiamo in esame l'interazione fra Client 1 e Server 1, l'accesso a quest'ultimo dovrebbe essere possibile sono da porta 21 dal global internet. Di seguito sono presenti due screenshot, il primo rappresenta i file eventualmente da trasferire e l'attivazione del servizio ftp sulla porta 21 del server, il secondo la configurazione del client per poter trasferire file con il server. In particolare, nella seconda immagine viene specificato l'indirizzo Ip del server e la porta in modo da eseguire il login. Il risultato è l'accesso ai file leggibili di Server 1. Una prova simile può essere fatta con Client 2 e Server 2, è possibile accedere al server tramite un url in cui deve essere specificato l'indirizzo del server ed il nome del file.

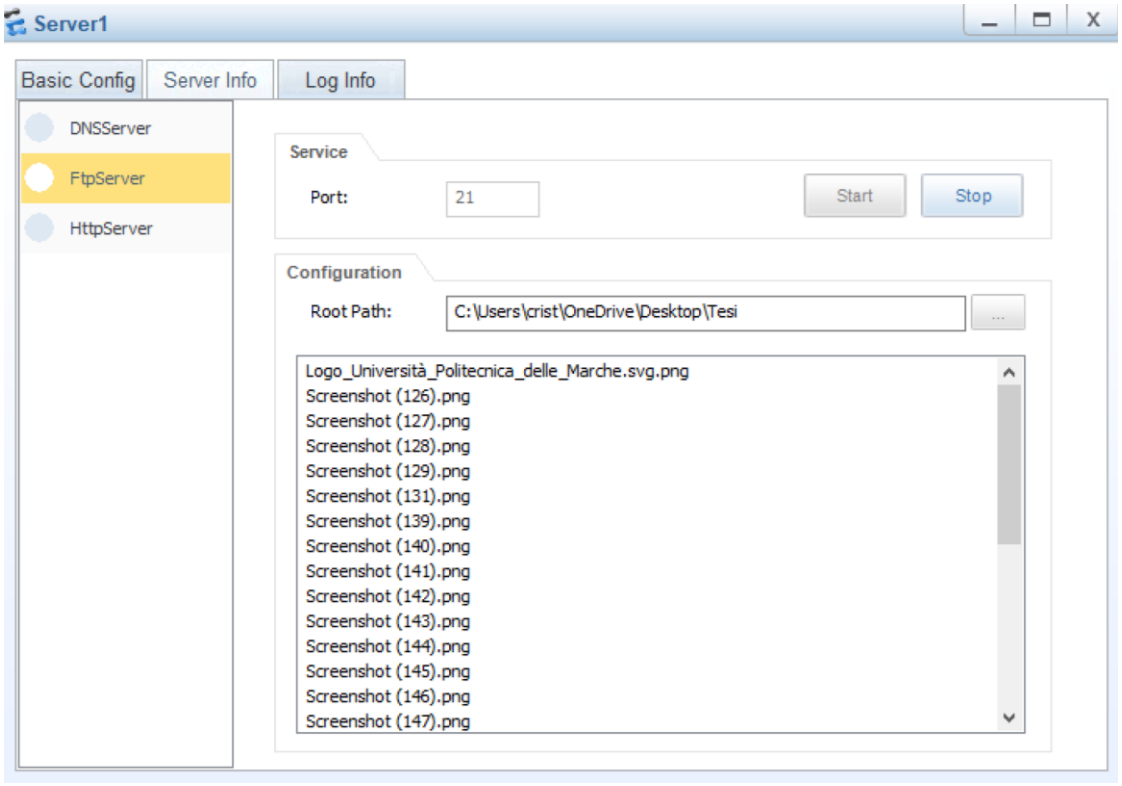


Figura 5.13

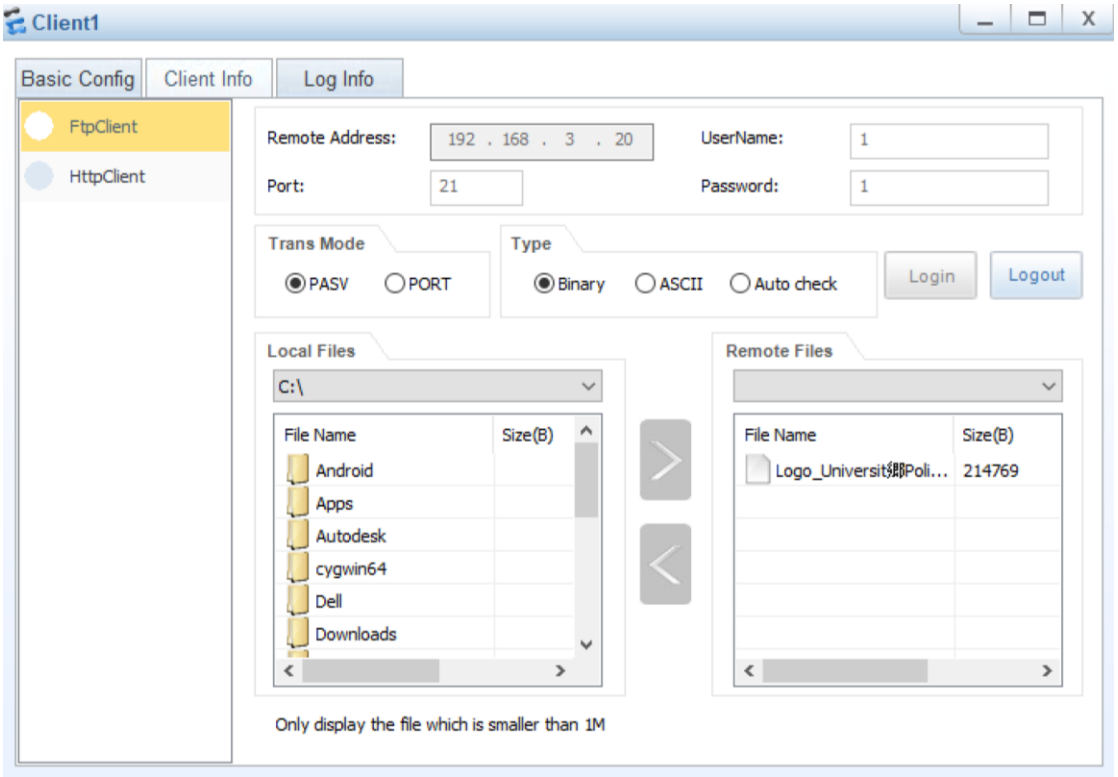


Figura 5.14

6 INDICE DELLE FIGURE

7 BIBLIOGRAFIA

Fastweb (2017): Internet, la "rete delle reti" tra passato e futuro

Link: https://www.fastweb.it/web-e-digital/internet-la-rete-delle-reti-tra-passato-e-futuro/

Cisco (2018): Cisco Annual Internet Report (2018–2023) White Paper

Link: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

FS (2013): Fiber Optic Cable vs Twisted Pair Cable vs Coaxial Cable

Link: https://community.fs.com/blog/the-difference-between-fiber-optic-cable-twisted-pair-and-cable.html

Wikipedia (2016): Port (computer networking)

Link: https://en.wikipedia.org/wiki/Port_(computer_networking)

Huawei Technologies Co. (2019), Basic_Training_Material

Huawei Technologies Co. (2019), Intermediate_Training_Material

Alessandro Cucchiarelli (2020), Corso tecnologie Web