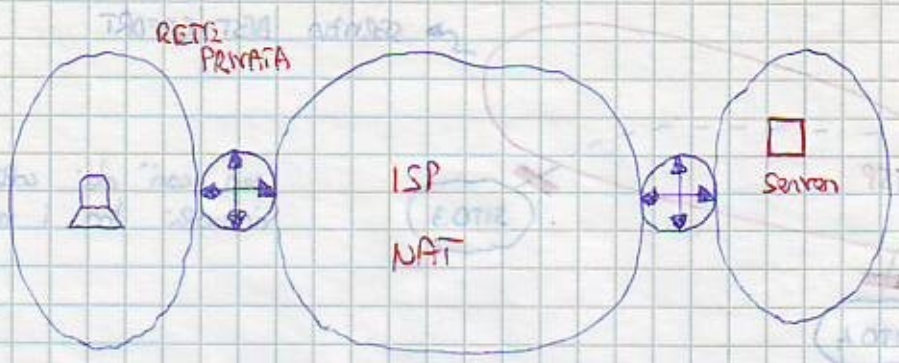
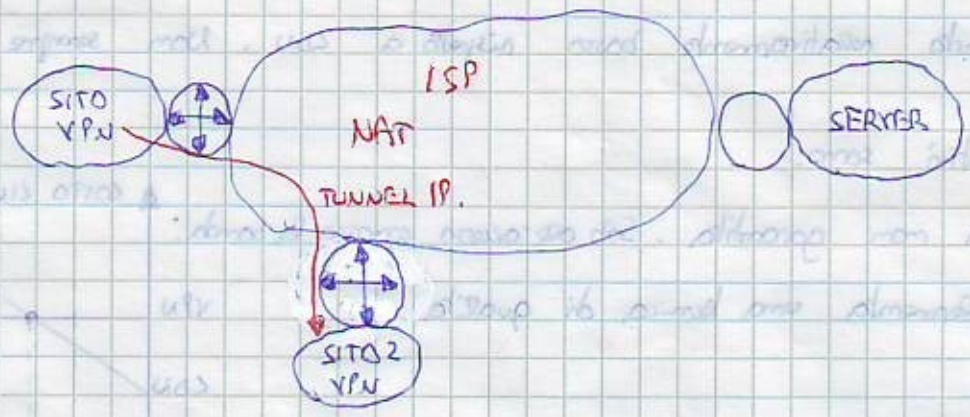
56



Spesso ci sono degli indirizzi non usabili in Internet e spesso le reti private vituali hanno degli "indirizzi locali". Quindi in questa situazione come si gestisce una comunicazione su Internet? Qui subentra la NAT Network Address Translation.



Si nota che tra le sito 1 e le sito 2 non ci sono problemi. Per le sito PRN e le server si usa NAT che ha due versioni:

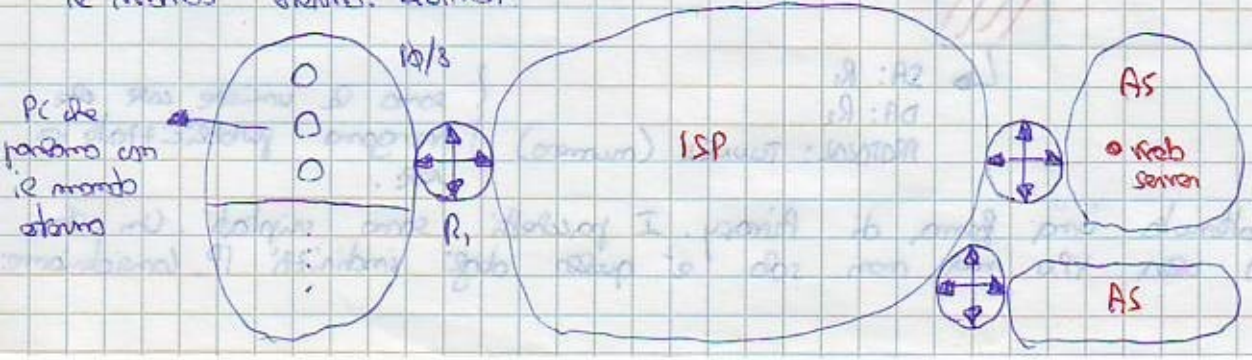
1) NAT

2) NAPT = Network Address Port Translation ed è più potente di NAT.

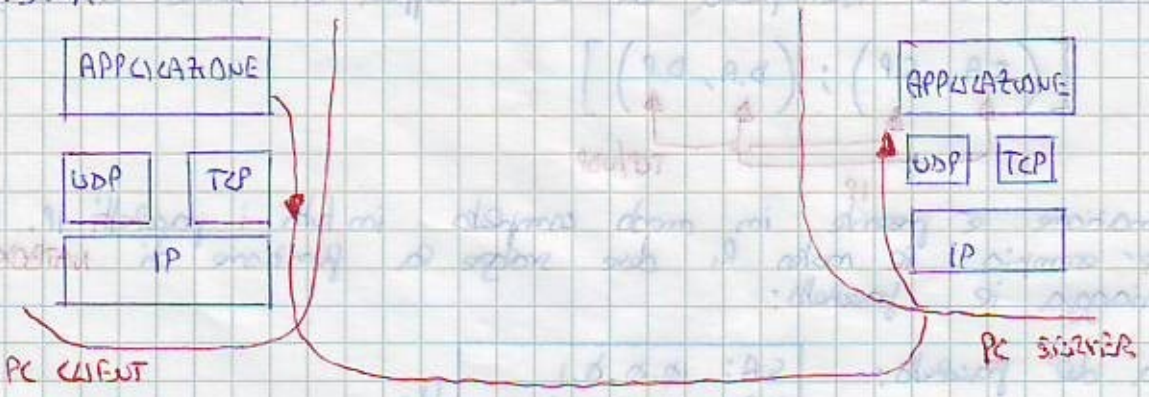
Consideriamo: → set limitato di indirizzi

Se A vuole parlare con il mondo esterno usa indirizzi buoni. Se non ci sono indirizzi buoni non si può attuare la comunicazione.

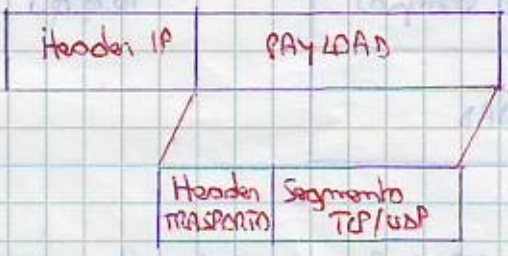
Partizioniamo ora le sito in una parte contenente il mondo dei PC che "parlano" con l'esterno, e in un'altra parte contenente PC che non parlano con il mondo esterno. Quindi:



Ipotizziamo che R1 abbia a disposizione solo un indirizzo buono. Per esempio: 199.10.3.5.



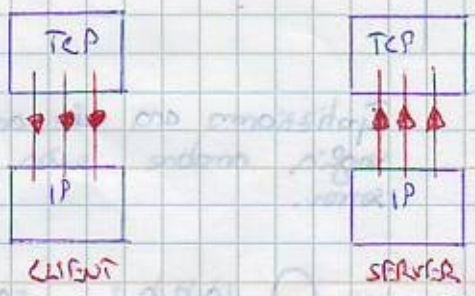
Le informazioni necessarie per NAPT nell'header e payload sono:



Sia per TCP che per UDP l'header è:

{SP: Source Port
{DP: Destination Port

Le informazioni di interesse sono sia le S.A e le D.A dell'IP che le S.P e D.P dell'header TCP.
Vediamo ora cosa sono i socket. Consideriamo:



Queste connessioni possono essere molteplici, e quindi devono avere degli identificativi. Questi sono gli SP. Questo processo è la MULTIPLAZIONE E DEMULTIPLAZIONE A LIVELLO TCP. Quindi ogni connessione è descritta dalla coppia (S.A, S.P).

(S.A, S.P) = identificano la macchina e la applicazione.

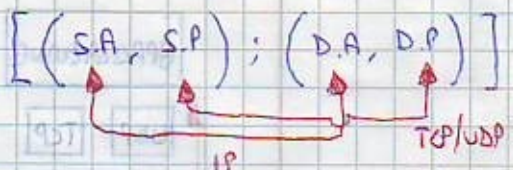
Questa coppia
va sotto il nome di SOCKET. Nel caso server avviene la medesima cosa. TCP può avere tante connessioni in parallelo. In questo caso si ha:

(D.A, D.P)

Ma tutto questo non basta per identificare una connessione.

(58)

Una singola connessione è identificata da una coppia di socket. Quindi:

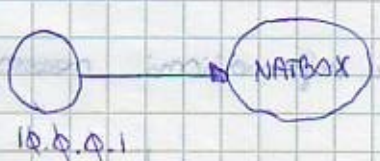


Questa informazione è presente in modo completo in tutti i pacchetti IP. Quindi tornando ad esempio in cui router R_1 deve svolgere la funzione di NATBOX, vediamo come viaggia il pacchetto:

1) Uscita del pacchetto:

S.A: 10.0.0.1 D.A: 131.175.21.1 PROTOCOL: TCP
D.P: 80 S.P: 5821 (esempio)
PAYLOAD

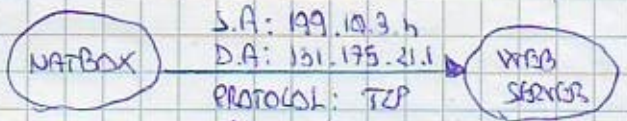
IP



2) Il pacchetto viene trasmesso verso NATBOX la quale ha la seguente tabella:

PRIV ADD	PRIV PORT	EXT ADD	EXT PORT	NAT PORT	PROTOCOL
10.0.0.1	5821	131.175.21.1	80	12323	TCP
10.0.0.2	5821	131.175.21.1	80	12824	TCP

→ Tuttavia ha SP.



Ipotizziamo ora che anche 10.0.0.2 voglia andare sullo stesso web server.

Quindi NATBOX istanzia una riga nella tabella per il nuovo pacchetto. Quindi il pacchetto arriverà al web server con i seguenti parametri:

S.A: 199.10.3.4
D.A: 131.175.21.1
PROTOCOL: TCP
D.P: 80
S.P: 12824

Si nota che l'unica differenza è la SOURCE PORT.

10.0.0.2
SP: 5821
DP: 80
DA: 131.175.21.1
SA: 10.0.0.2

Il web server risponde. Figli invia la seguente di pacchetti:

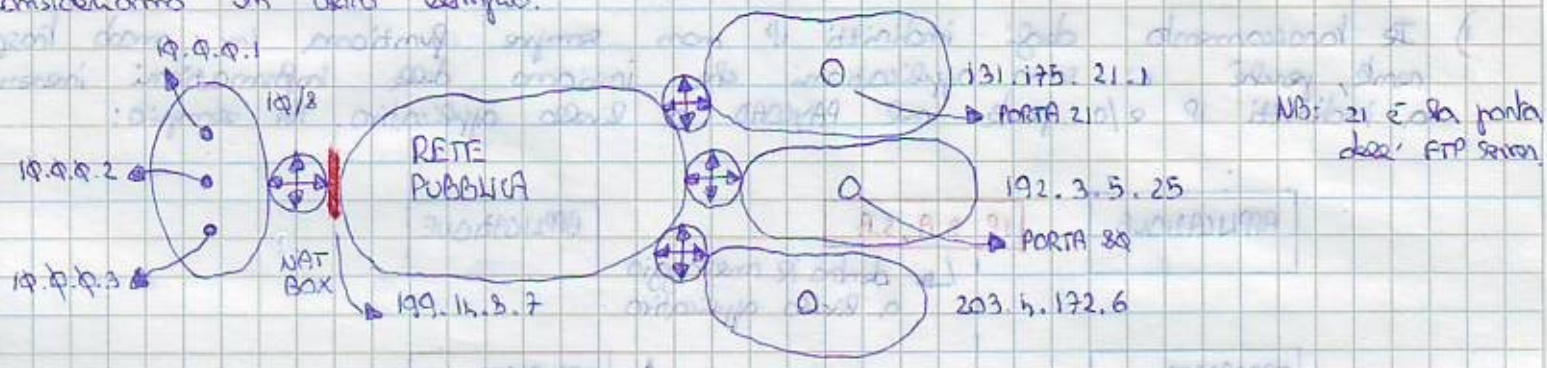
S.A: 131.175.21.1
D.A: 199.1h.3.h
PROTOCOL: TCP
SP: 80
DP: 12026h

⇒ NATBOX guarda la tabella di traspazione e fa "matching". Qui è unica differenza è la DESTINATION PORT.

≈≈≈ PACCHETTO FINALE.

DP: 5821

Consideriamo un altro esempio:



NB: 21 è la porta del FTP server

Vediamo l'evoluzione della NATBOX:

PRIVATE ADDR.	PRIVATE PORT	EXT ADDR	EXT PORT	NAT PORT	PROTOCOL
10.0.0.1	587h	131.175.21.1	21	3000	TCP
10.0.0.1	5875	131.175.21.1	21	3001	TCP
10.0.0.2	8720	131.175.21.1	21	3002	TCP
10.0.0.3	22000	192.3.5.25	80	3000	TCP
10.0.0.3	23000	192.3.5.25	80	3000	UDP

Si noti che se si hanno reti della stessa organizzazione si usa il tunneling altrimenti si usa NATBOX. Osservando la tabella si nota che quando EXT ADDR è diverso si può usare una NATPORT che è identica per più reti; tanto è ambiguità viene "tolta" da EXT ADDR. Gli indirizzi di NATPORT si ricordi che sono a 16 bit. Quindi si possono avere 2^{16} indirizzi diversi, ma tali indirizzi sono comunque limitati. Si noti inoltre che si può creare la stessa porta NAT cambia il protocollo. Vediamo ora come sono fatti i pacchetti usando TCP.

10.0.0.3 → NATBOX

S.A: 10.0.0.9
D.A: 192.3.5.25
SP: 22000
D.P: 80
PROTOCOL: TCP

NATBOX → 192.3.5.25

S.A: 199.1h.3.7
D.A: 192.3.5.25
S.P: 3000
D.P: 80
PROTOCOL: TCP

192.3.5.25 → NATBOX

S.A: 192.3.5.25
D.A: 199.1h.3.7
S.P: 80
D.P: 3000
PROTOCOL: TCP

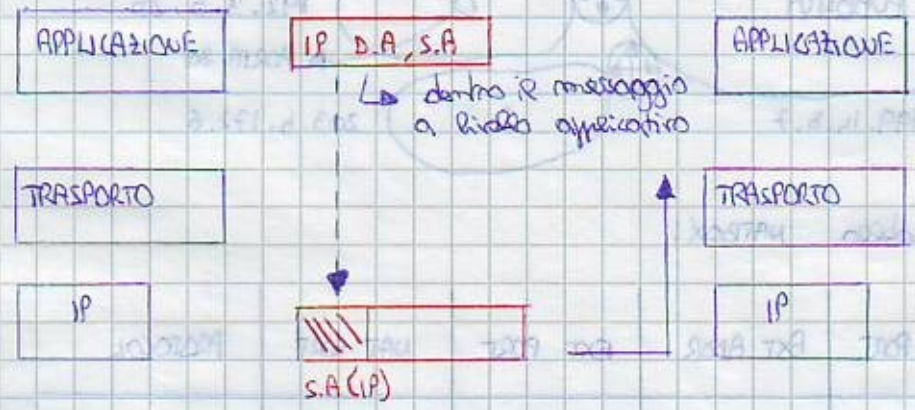
(69)

NATBOX → 10.0.0.3

S.A: 192.3.5.25 → ci va l'indirizzo del punto esterno della connessione.
D.A: 10.0.0.3
S.P: 80
D.P: 22000
PROTOCOL: TCP

Purtroppo ci sono dei problemi in NAT:

1) Il funzionamento degli indirizzi IP non sempre funziona in modo trasparente, perché ci sono applicazioni che inviano delle informazioni inerenti a indirizzi IP e/o porte nel PAYLOAD a livello applicativo. Per esempio:



L'applicazione vede il S.A e D.A dell'altra applicazione invece del S.A e D.A dell'IP. È ovvio che NAT non funziona perché non avviene la traslazione. NAT purtroppo in questi casi deve cambiare gli indirizzi IP dentro i messaggi a livello applicativo (spostamento del messaggio e ricerca dell'indirizzo IP da cambiare.) Si ha quindi una doppia traslazione. Questo deve essere fatto in modo diverso per ogni applicazione. Quindi è un'ulteriore traslazione dipende dalla specifica applicazione. Per ogni applicazione ci vuole una specifica procedura. Però qui nasce un ulteriore problema:

2) È un problema legato ai livelli di trasporto (TCP). Il TCP diventa critico perché se per esempio abbiamo:

S.A: 10.0.0.8 → S.A: 192.15.202.17 (codifica ASCII)

noi passiamo da 8 caratteri a 13 caratteri nella codifica. Questo è un problema grave. Infatti il TCP, per eseguire le sue operazioni, da un numero di sequenza ai byte. Quindi noi andiamo a inserire o togliere a livello di NATBOX, dei byte alterando così la numerazione dei byte da parte del TCP. Quindi la NATBOX deve gestire anche la numerazione di sequenza dei byte a livello TCP "impacchettando". Questa però è una operazione abbastanza complessa.

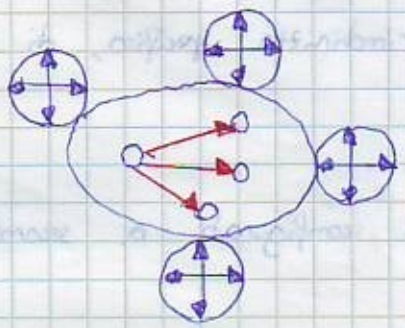
Il router, per quanto riguarda il software, ha già una funzione di NAT.

Vediamo ora i tipi di collegamenti disponibili:

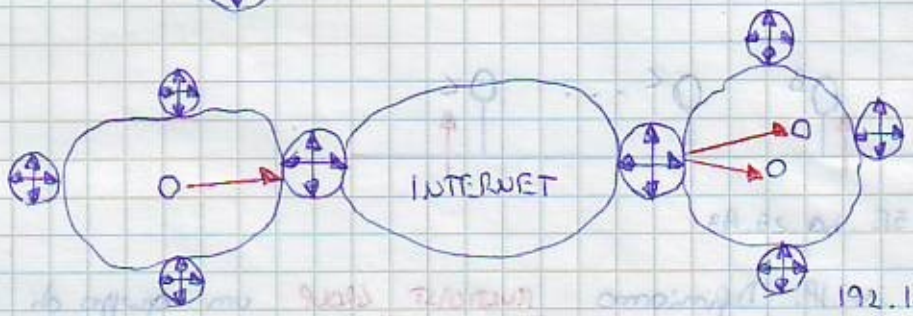
1) COLLEGAMENTO PUNTO - PUNTO

Per esempio: CLIENT → SERVER

2) COLLEGAMENTO BROADCAST



D.A: 255. 255. 255. 255 Broadcast locale



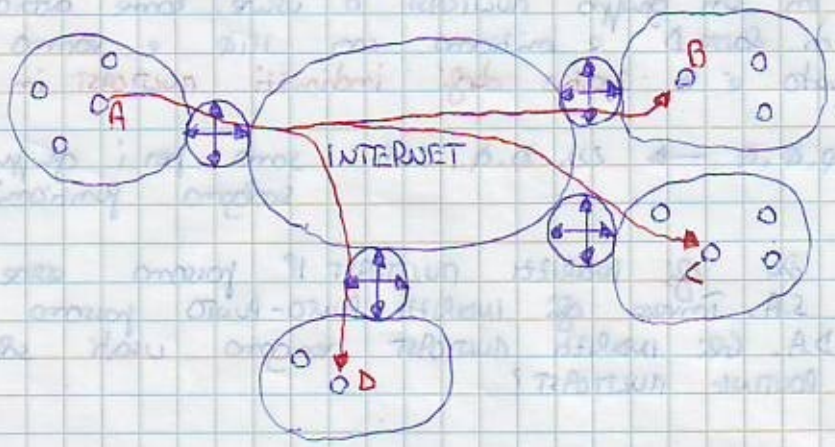
Broadcast limitato

Limito così il traffico.

192.16.3.0 /24

D.A: 192.16.3.255

3) COLLEGAMENTO MULTICASTING: dove invio il pacchetto ad un gruppo specifico di macchine. Consideriamo per esempio:



I pacchetti giungono a quelle specifiche macchine ed non alle altre.

PUNTO → MULTIPUNTO

Quest'ultimo tipo di connessione è una connessione che sta crescendo di importanza. Ma come avviene il MULTICASTING? Vediamo il multicasting locale sulla tecnologia ETHERNET:

A livello IP abbiamo: 255.255.255.255
A livello Hardware abbiamo: FF.FF.FF.FF.FF.FF → indirizzo HW di broadcast

Una scheda ETHERNET riconosce e' indirizzo specifico (quello della scheda) e riconosce e' indirizzo broadcast. Ma esiste anche le MULTICAST:

INDIRIZZO MULTICAST ETHERNET 01.00.5E.XX.XX.XX

01.00.5E: parte fissa
XX.XX.XX: 24 bit per la specifica macchina (ind. MULTICAST)

In realtà si usano 23 e non 24 bit.
Quindi una scheda ETHERNET riconosce e' indirizzo specifico, di broadcast, e uno o più indirizzi MULTICAST.

INDIRIZZO SPECIFICO } STATICI
INDIRIZZO BROADCAST }
INDIRIZZO MULTICAST → vanno configurati a seconda dell'applicazione.

Ad esempio:

01.00.5E.0A.25.A3

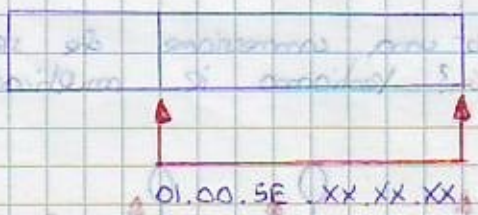
Vediamo le MULTICAST in IP. Definiamo MULTICAST GROUP un gruppo di destinazione coinvolto in una comunicazione MULTICAST. Quindi si ha:

GROUP ADDRESS, CLASSFULL, CLASSE D → classe di indirizzi tipo MULTICAST.

Esistono 2^{28} gruppi MULTICAST ed esistono procedure tramite le quali un host può richiedere di entrare in un gruppo MULTICAST o uscire. Come abbiamo detto gli indirizzi MULTICAST sono di classe D e iniziano con 1110 e vanno da 224.0.0.0 a 239.255.255.255 (questo è lo spazio degli indirizzi MULTICAST in IP).

224.0.0.0 → 224.0.0.255 sono per i gruppi predefiniti che svolgono funzioni speciali.

Si noti in particolare che gli INDIRIZZI MULTICAST IP possono essere usati solo come D.A, e non come S.A. Invece gli INDIRIZZI PUNTO-PUNTO possono essere usati sia come S.A che come D.A. Gli INDIRIZZI MULTICAST vengono usati solo come D.A.
Ma come avviene le ROUTING MULTICAST?



Prendo i 23 bit meno significativi dell'indirizzo MULTICAST e li copio nei 23 bit meno significativi dell'indirizzo hardware MULTICAST.