Progetto di rete aziendale

RETI DI CALCOLATORI: PROTOCOLLI

ANNO ACCADEMICO 2021/2022

Docente: Sergio Tasso

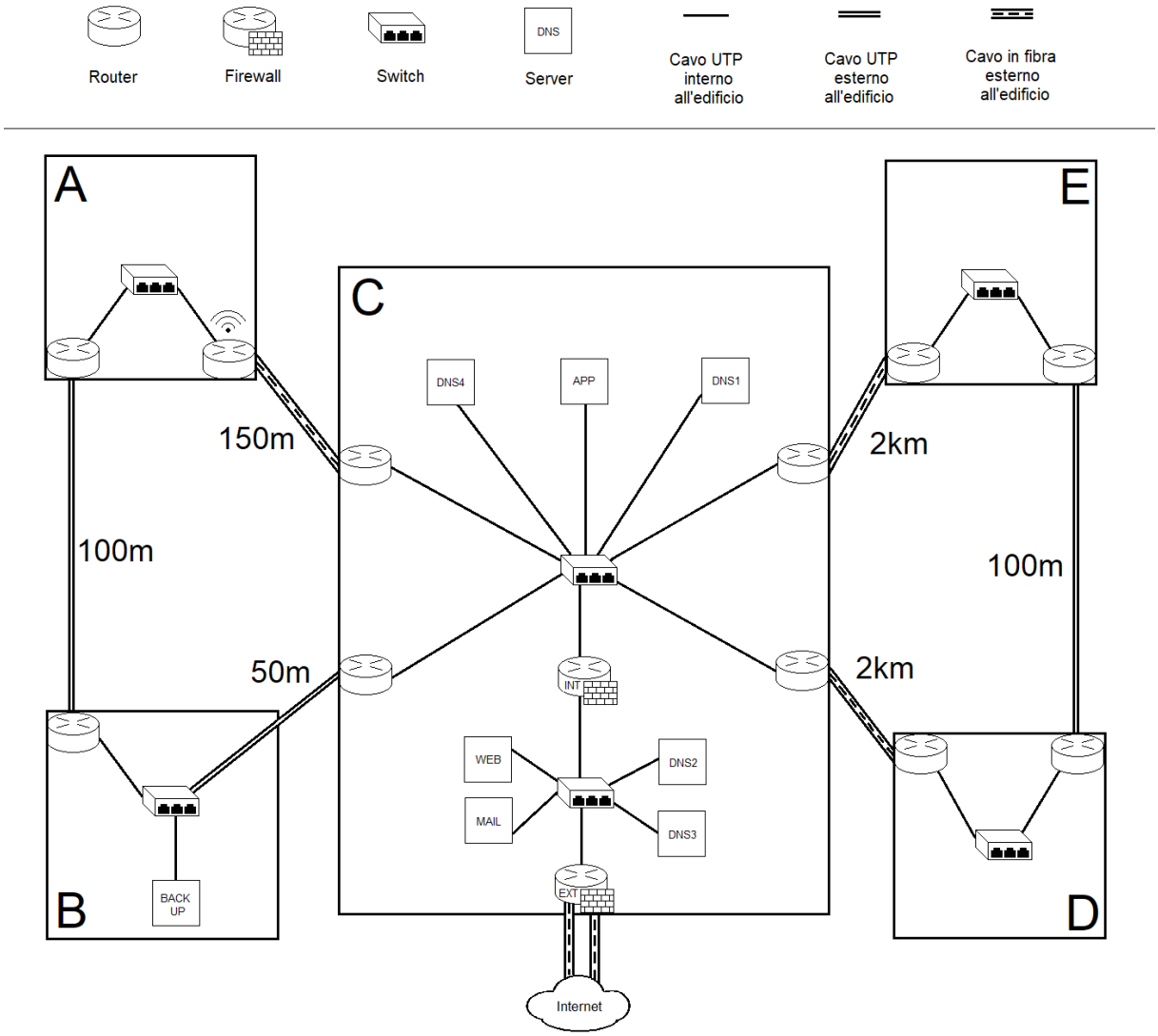
A cura di:

DEBORA DAMIANI, 325436

ANDREA IMPARATO, 323840

Indice:

1. SCHEMA FISICO DELLA RETE



Il mezzo trasmissivo più versatile e conveniente in termini di rapporto qualità/prezzo per un cablaggio aziendale è il cavo UTP, che costituisce quindi l'opzione predominante in questo schema. Sia i cavi che si trovano all'interno degli edifici (linea singola) che quelli esterni di interconnessione tra i vari edifici (linea doppia) devono essere di categoria Cat5 o superiore, ovvero cavi di lunghezza predeterminata prodotti industrialmente ad hoc e già crimpati. Un cablaggio del genere permetterebbe di raggiungere una velocità massima di 100Mbps, limite che in teoria può essere ulteriormente esteso qualora le schede di rete degli host aziendali fossero compatibili anche con lo standard 1000baseT e qualora l'azienda decidesse di dotarsi di cavi di qualità ancora più alta (Cat5e o superiore). Di contro, un importante limite del cavo

UTP come mezzo trasmissivo è che la lunghezza massima di un qualsiasi cavo non deve mai superare i 100m (in accordo con gli standard 10/100/1000baseT di IEEE802.3). Per collegare gli edifici A, D ed E all'edificio C abbiamo quindi optato per un cablaggio in fibra.

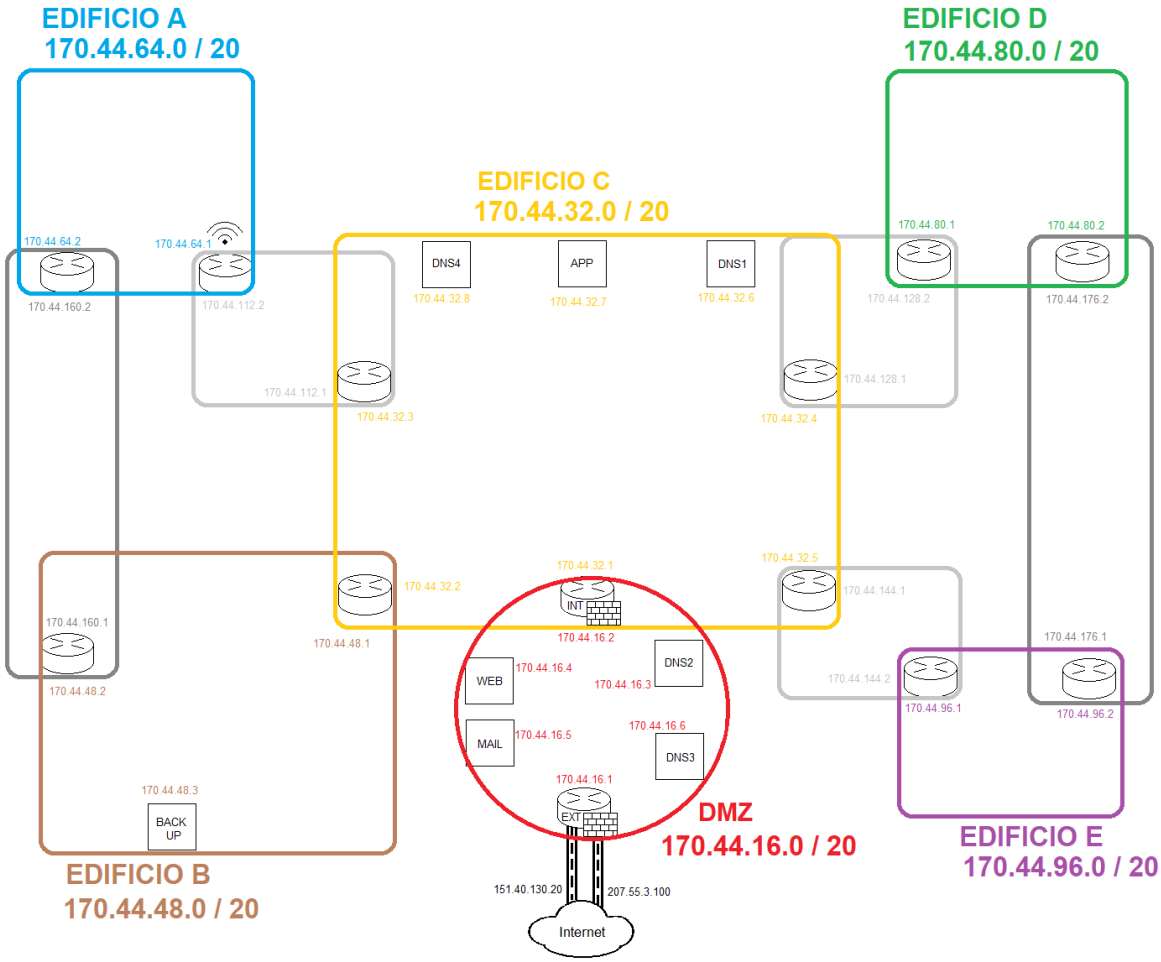
Dato che ogni cavo UTP permette al più di creare un singolo collegamento tra due stazioni, ogni edificio deve avere ovviamente uno Switch dedicato volto a permettere l'aggancio a tutti i suoi host (l'edificio C ne ha a disposizione uno aggiuntivo per la DMZ).

In ogni edificio sono inoltre presenti router che gestiscono il passaggio a cavallo delle varie sottoreti ed edifici. Ai due estremi di ogni collegamento in fibra figurano sempre due router poiché ha luogo il passaggio a un mezzo trasmissivo diverso. A livello logico ciò si tradurrà nell'istanziazione di una sottorete di servizio per ogni tratto in fibra.

L'edificio C è il più centrale di tutti nonché quello in grado di ospitare il maggior numero di host aziendali, ragion per cui l'allaccio con Internet ha luogo proprio in questo edificio. Ne consegue che anche la DMZ si trova qui, adiacente a un External Firewall dotato di tre interfacce: una verso la DMZ stessa e due verso l'esterno. Il motivo di ciò è che abbiamo scelto di dotare l'azienda di due contratti con due ISP ben distinti, un investimento il cui fine è sia garantire una maggiore robustezza in caso di guasti di cui è responsabile il Provider sia rendere l'azienda candidabile a diventare un Autonomous System (dato che il possesso di due indirizzi IP è un prerequisito). Come vedremo più in dettaglio nel prossimo capitolo abbiamo inoltre immaginato che all'azienda sia stato conferito il controllo di un'intera rete di classe B (rimaniamo ovviamente consapevoli che uno scenario del genere nella vita reale sarebbe ormai improbabile).

I Server dedicati ai servizi che devono risultare raggiungibili dagli utenti esterni (Web, Mail, Dns2, Dns3) sono stati collocati nella DMZ. Il Server delle App aziendali e i Server Dns1 e Dns4 si trovano invece nella rete interna, protetti dall'Internal Firewall. Il Server di Backup si trova anch'esso nella rete interna ma addirittura in un edificio completamente diverso, in modo che possa agire come strumento di disaster recovery anche in caso di distruzione fisica dell'edificio C. L'edificio A deve permettere anche la connessione Wireless, ragion per cui il suo Router è anche un Access Point protetto da WPA2 che fa da Server DHCP.

2. SCHEMA LOGICO DELLA RETE



All'azienda è stato conferito il controllo della rete di classe B 170.44.0.0. La Subnet Mask è stata estesa globalmente a /20 per permettere un opportuno partizionamento: avendo noi istanziato 11 sottoreti principali si sono resi necessari almeno 4 bit per l'indirizzo di sottorete. A ogni edificio è stata assegnata la sua sottorete dedicata, mentre un'ulteriore sottorete serve a implementare la DMZ. Altre sono invece semplici sottoreti di servizio che fanno da ponte, dovute alla presenza di un mezzo trasmissivo diverso (grigio chiaro) o di cavi la cui lunghezza tocca il limite massimo consentito (grigio scuro): in queste sottoreti non sono in teoria previsti ulteriori host oltre ai due router localizzati alle due estremità. Le due sottoreti in grigio scuro sono le uniche a creare collegamenti ridondanti. In tutte le altre è sempre identificabile un singolo router passando per il quale si raggiunge l'esterno dalla rete aziendale nel minor numero possibile di hop, a cui viene quindi sempre assegnato "1" come indirizzo host per enfatizzare il fatto che è il candidato ideale a fare da default gateway per quella sottorete.

3. CONFIGURAZIONE INTERFACCE

3.1 Interfacce degli host

Quattro elementi sono imprescindibili affinché un generico host sia in grado di navigare in rete: IP Address, Subnet Mask, Default gateway e NameServer Address. L'impostazione dei primi due ha luogo tramite il comando **ifconfig**, l'impostazione del default gateway tramite il comando **route** e l'impostazione del Nameserver Address tramite la modifica manuale del file **/etc/resolv.conf** . Supponiamo ad esempio di voler collegare in rete la primissima postazione dipendenti nella sottorete appena installata nell'edificio D, il cui indirizzo è 170.44.80.0 / 20 . Siccome gli indirizzi host "1" e "2" sono già riservati dai due router ivi presenti, la nostra postazione assumerà come indirizzo "3". Dallo schema logico visto nel Capitolo 2 sappiamo inoltre che in ogni sottorete il router per cui conviene sempre passare per raggiungere l' esterno della rete aziendale il più rapidamente possibile è contraddistinto dall'indirizzo "1". I comandi da lanciare per configurare la postazione saranno quindi:

ifconfig eth0 170.44.80.3 netmask 255.255.240.0 broadcast 170.44.95.255
route add -net 0.0.0.0 gateway 170.44.80.1

Non va inoltre dimenticato che questi comandi andranno replicati a ogni nuovo avvio della macchina, ragion per cui è opportuno aggiungerli al file **/etc/rc.local** al fine di non doverli battere manualmente ogni volta (alla linea relativa al comando route va inoltre aggiunto anche **> /dev/console**). Il file **/etc/resolv.conf** dovrà invece contenere le seguenti righe, che specificano i Nameserver Address da usare per la risoluzione degli hostname sconosciuti:

```
domain identerprises.it
nameserver 170.44.32.8
nameserver 85.37.17.51
nameserver 85.38.28.97
```

3.2 Interfacce dei Router

Quanto appena detto in modo generico per gli host continua a valere anche per i Router, che per svolgere il proprio compito devono però avere come minimo due interfacce configurate. E' inoltre fondamentale assicurarsi che sia impostato a "1" il valore contenuto nel file **/proc/sys/net/ipv4/ip_forward**: se così non fosse il router non consentirebbe il transito di pacchetti a cavallo delle sue interfacce e non sarebbe quindi in grado di svolgere la sua funzione. E' altresì importante controllare che sia impostato a "1" anche il valore contenuto nel file **/proc/sys/net/ipv4/conf/all/send_redirects** in modo che il Router possa inviare eventuali Redirect ICMP, ma in questo caso non è necessario intervenire in /etc/rc.local perché normalmente questo valore è già "1" di default. Come esempio prendiamo in esame il router che connette la sottorete dell'edificio B alla sottorete dell'edificio C. Alla luce di quanto appena detto, i comandi relativi a Interfacce, Default Gateway e Ip forwarding inclusi nel file /etc/rc.local di suddetto router dovranno essere:

ifconfig eth0 170.44.32.2 netmask 255.255.240.0 broadcast 170.44.47.255
ifconfig eth1 170.44.48.1 netmask 255.255.240.0 broadcast 170.44.63.255
route add -net 0.0.0.0 gateway 170.44.32.1 > /dev/console
echo "1" > /proc/sys/net/ipv4/ip_forward

La configurazione di un qualsiasi altro Router della rete aziendale avverrà in modo identico salvo che per le eventuali differenze nel valore assunto dagli indirizzi e nel nome dell'interfaccia qualora il Router fosse collegato a un tratto in fibra. Ad esempio il file /etc/rc.local del Router che connette la sottorete dell'edificio D alla sottorete di servizio corrispondente al tratto in fibra che permette di raggiungere l'edificio C conterrà:

ifconfig fib0 170.44.128.2 netmask 255.255.240.0 broadcast 170.44.143.255
ifconfig eth0 170.44.80.1 netmask 255.255.240.0 broadcast 170.44.95.255
route add -net 0.0.0.0 gateway 170.44.128.1 > /dev/console
echo "1" > /proc/sys/net/ipv4/ip_forward

La configurazione delle due interfacce esterne del Router che fa da External Firewall non può ovviamente contraddire le regole stabilite dai due ISP associati.

4. CONFIGURAZIONE DEL ROUTING

4.1 Cenni generali

Per facilitare la configurazione delle route l'opzione migliore è affidarsi a protocolli di routing dinamici dal momento che il numero di sottoreti istanziate renderebbe onerosa (seppure teoricamente possibile) una configurazione statica basata sul semplice uso ripetuto del comando **route**. Visto il numero tutto sommato contenuto di sottoreti una prima opzione possibile sarebbe quella di usare il semplice protocollo **RIP**: essendo le sottoreti solo 11 sarebbe sempre garantito il raggiungimento dell'esterno della rete aziendale senza mai toccare il limite massimo di 15 hop. Ciò nonostante, dopo attenta riflessione abbiamo concluso che sarebbe più opportuno affidare i calcoli al protocollo **OSPF** giacché la sua capacità di bilanciare il carico su diversi Link permetterebbe di sfruttare meglio le route ridondanti offerte dalle sottoreti di servizio che nel Capitolo 2 abbiamo evidenziato in grigio. Il protocollo RIP verrà quindi usato solo per annunciare le route all'interno delle singole sottoreti, di modo che gli host possano ricevere le informazioni di routing ricorrendo a un solo protocollo. L'opzione migliore a livello di software diventa quindi **gated** in quanto capace di gestire simultaneamente non solo RIP e OSPF ma anche **EGP**, che giocherebbe a sua volta un ruolo importante qualora l'azienda riuscisse effettivamente ad acquisire lo status di Autonomous System come ipotizzato nel Capitolo 1. Indipendentemente dal se sono Host semplici, Server o Router tutti gli host connessi a qualsiasi sottorete dovranno quindi configurare in modo opportuno il file **/etc/gated.conf** e includere nel file /etc/rc.local i seguenti comandi, necessari ad assicurare il lancio di gated all'avvio della macchina:

```
if [ -f /etc/gated -a -f /etc/gated.conf ] then
        gated;
        echo -n 'gated' > /dev/console
fi
```

Nei prossimi paragrafi esamineremo più in dettaglio il contenuto del file di configurazione /etc/gated.conf in diversi host, giacché il suo contenuto varia a seconda del ruolo che l'host stesso gioca di volta in volta.

4.2 Configurazione di gated.conf per gli Host semplici

In ogni host semplice è sufficiente abilitare il solo protocollo RIP in modalità di solo ascolto in modo che possa ricevere le informazioni di routing. E' inoltre bene impostare come "passive" l'unica interfaccia attiva per proteggerla da eventuali tentativi di modifica da parte di gated, che comprometterebbero la connettività dell'host. Ad esempio nell'host "3" connesso alla sottorete installata nell'edificio D il file /etc/gated.conf sarà così impostato:

```
interfaces {
        interface 170.44.80.3 passive;
};

rip yes {
        nobroadcast;
        version 2;
        multicast;
        authentication simple "RIPauth";
};
```

4.3 Configurazione di gated.conf per i Router interni

Nei Router interni devono essere abilitati sia OSPF che RIP, ed è bene assegnare esplicitamente un valore di "preference" minore (ovvero più "buono") a OSPF in modo da assicurarsi che RIP non possa mai sovrascrivere le informazioni calcolate con OSPF. Come ulteriore misura di sicurezza disabilitiamo in toto l'ascolto sul protocollo RIP aggiungendo "noripin" a tutte le interfacce: questa scelta riflette il fatto che RIP è relegato al semplice ruolo di strumento di annuncio all'interno delle varie sottoreti, ragion per cui i Router devono parlare senza udire e gli host ascoltare senza parlare. Gli statement di export sono organizzati in modo tale che ogni Router fornisca a OSPF tutte le informazioni relative ai suoi Link diretti, mentre al protocollo RIP vengono passate sia le informazioni relative ai Link diretti che alle route calcolate tramite OSPF. Relativamente a OSPF, tutte le sottoreti tranne la DMZ faranno parte di un'unica stub area, mentre la DMZ stessa farà da backbone. Fintanto che ci si sposta all'interno delle sottoreti aziendali esiste sempre più di una strada percorribile (tranne che per la DMZ), pertanto la protezione delle interfacce tramite "passive" non è fondamentale

come nel caso dei singoli host. Per fare un esempio prendiamo ora in esame il file /etc/gated.conf del Router che collega la sottorete dell'edificio C al quella dell'edificio B:

```
routerid 170.44.32.2;


rip yes {
        preference 100;
        broadcast;
        version 2;
        multicast;
        interface 170.44.32.2 {
                noripin;
                authentication simple "RIPauth";
        };
        interface 170.44.48.1 {
                noripin;
                authentication simple "RIPauth";
        };
};


ospf yes {
        defaults {
                preference 90;
        };
        area 1 {
                stub;
                authtype simple;
                interface 170.44.32.2 {
                        authkey "OSPFauth";
                        priority 5;
                };
                networks {
                        170.44.64.0;
                        170.44.48.0;
                        170.44.32.0;
                        170.44.80.0;
                        170.44.96.0;
                        170.44.112.0;
                        170.44.160.0;
                        170.44.128.0;
                        170.44.176.0;
                        170.44.144.0;
                };
        };
```

```
};

export proto ospf metric 0 {
        proto direct 170.44.32.2 {
                network 170.44.32.0;
        };
        proto direct 170.44.48.1 {
                network 170.44.48.0;
        };
};

export proto rip {
        interface all;
        proto ospf;
        proto direct;
};
```

4.4 Configurazione di gated.conf per l'Internal Firewall

La configurazione del Router che fa da Internal Firewall è del tutto analoga a quella degli altri router salvo che per quanto concerne OSPF: questo router comunica infatti sia con la stub area che con la backbone, facendo uso di due routerid ben distinti corrispondenti alle sue due interfacce. Queste due interfacce vanno protette con "passive" similmente a come è stato già fatto negli host semplici, giacché la disattivazione accidentale di anche solo una di esse comprometterebbe la connettività dell'intera rete aziendale. Il file /etc/gated.conf sarà quindi così configurato:

```
interfaces {
        interface 170.44.32.1 passive;
        interface 170.44.16.2 passive;
};
```

```
routerid 170.44.32.1;
routerid 170.44.16.2;


rip yes {
        preference 100;
        broadcast;
        version 2;
        multicast;
        interface 170.44.32.1 {
                noripin;
                authentication simple "RIPauth";
        };
        interface 170.44.16.2 {
                noripin;
                authentication simple "RIPauth";
        };
};


ospf yes {
        defaults {
                preference 90;
        };
        area 1 {
                stub;
                authtype simple;
                interface 170.44.32.1 {
                        authkey "OSPFauth";
                        priority 5;
                };
                networks {
                        170.44.64.0;
                        170.44.48.0;
                        170.44.32.0;
                        170.44.80.0;
                        170.44.96.0;
                        170.44.112.0;
                        170.44.160.0;
                        170.44.128.0;
                        170.44.176.0;
                        170.44.144.0;
                };
        };
        backbone {
                authtype simple;
```

```
                interface 170.44.16.2 {
                        authkey "OSPFauth";
                        priority 5;
                };
                networks {
                        170.44.16.0;
                };
        };
};

export proto ospf metric 0 {
        proto direct 170.44.32.1 {
                network 170.44.32.0;
        };
        proto direct 170.44.16.2 {
                network 170.44.16.0;
        };
};

export proto rip {
        interface all;
        proto ospf;
        proto direct;
};
```

4.5 Configurazione di gated.conf per l'External Firewall

L'External Firewall è agganciato verso l'interno alla backbone OSPF e verso l'esterno a due Autonomous System differenti corrispondenti ai due ISP i cui contratti permettono l'esistenza delle due interfacce esterne. Nel nostro esempio supporremo che l'ASN della nostra azienda sia 147, mentre quelli dei due ISP rispettivamente 169 e 200. Supporremo altresì che gli indirizzi IP degli egp neighbours con cui parla l'External Firewall siano quelli che figurano nella configurazione di EGP. Il protocollo RIP va disattivato esplicitamente, giacché l'opzione di default di gated ne prevede l'attivazione automatica se non diversamente specificato. Delle

interfacce disponibili solo quella verso l'interno va protetta con "passive" giacché il senso di avere a disposizione due interfacce esterne è proprio che la connettività continui ad essere garantita anche in caso di disattivazione di una delle due per cause di forza maggiore. Gli statement di export sono volti a far sì che le informazioni relative alle route esterne vengano passate dal protocollo EGP al protocollo OSPF affinché circolino anche all'interno. Il protocollo EGP stesso riceverà invece come informazioni da esportare all'esterno solo quelle relative alla DMZ: il senso di ciò è che siccome si suppone che nessuno provi mai a contattare dall'esterno gli host protetti dall'Internal Firewall non vogliamo neanche che vengano pubblicizzate eventuali route che permetterebbero di raggiungerli. Alla luce di quanto appena detto, il file /etc/gated.conf dell'External Router dovrà essere così configurato:

```
interfaces {
        interface 170.44.16.1 passive;
};


routerid  170.44.16.1;
autonomoussystem 147;
options gendefault;

rip no;

ospf yes {
        defaults {
                preference 90;
        };
        backbone {
                authtype simple;
                interface 170.44.16.1 {
                        authkey "OSPFauth";
                        priority 5;
                };
                networks {
                        170.44.16.0;
                };
        };
};

egp yes {
        packetsize 12288;
        group minhello 2:30 minpoll 10:00 {
                neighbor 151.30.20.3;
```

```
                neighbor 207.55.3.168;
        };
};

export proto ospf {
        proto egp as 147 {all};
        proto direct 170.44.16.1 {
                network 170.44.16.0;
        };
};

export proto egp as 169 {
        proto direct;
        proto ospf {
                network  170.44.16.0;
        };
};

export proto egp as 200 {
        proto direct;
        proto ospf {
                network  170.44.16.0;
        };
};
```

5. CONFIGURAZIONE DEI SERVER:

5.1 Server DHCP

L'accesso Wireless nell'edificio A è garantito dalla presenza di un Router capace di essere anche Access Point dotato di WPA2 (a livello hardware) e Server DHCP (a livello software). In aggiunta alla configurazione dei parametri già menzionati per i router interni su questo host andrà predisposto anche il lancio automatico del daemon **dhcpd** all'avvio della macchina tramite l'aggiunta delle seguenti linee al file **/etc/rc.local**:

```
if [ -f /usr/sbin/dhcpd -a -f /etc/dhcp/dhcpd.conf ] then
        /usr/sbin/dhcpd;
        echo -n '/usr/sbin/dhcpd' > /dev/console
fi
```

Dovrà inoltre essere stato preventivamente configurato anche il file **/etc/dhcp/dhcpd.conf** usando i seguenti parametri:

```
default-lease-time 6000;
max-lease-time 72000;
option subnet-mask 255.255.240.0;
option routers 170.44.64.1;
option domain-name-servers 170.44.32.8 , 85.37.17.51 , 85.38.28.97 ;
option domain-name "identerprises.it";
subnet 170.44.64.0 netmask 255.255.240.0 {
        range 170.44.64.53 170.44.64.103;
}
```

Il range di indirizzi host scelto per la dhcp pool va dal "53" al "103" per garantire la possibilità di avere fino a 50 host connessi in modalità Wireless (in accordo con quanto previsto dalle specifiche del progetto). Quelli dal "3" al "52" non vengono invece mai offerti in modo da tenerli riservati per eventuali host che vogliano connettersi via cavo.

5.2 Considerazioni generali sui Server DNS

Il numero di Server DNS dislocati nella rete aziendale è stato aumentato da 2 a 4 in modo da poter distribuire meglio le loro funzionalità e definirne meglio il ruolo. Al fine di offrire all'esterno una quantità di NameServer conforme alla soglia minima stabilita da RFC 1034 per ogni dominio, due Server DNS sono stati posizionati nella DMZ. Il compito di questi Server è rendere noti agli utenti esterni gli indirizzi IP di Server associati a servizi offerti all'esterno (Web, Mail, gli stessi NameServer): di questi, il Server DNS2 (170.44.16.3) è il "primario" e il Server DNS3 (170.44.16.6) è il "secondario". E' presente però anche un terzo Server localizzato all'interno della rete aziendale (e quindi mai visibile dall'esterno) che fa da "Hidden Master": è in realtà lui a farsi carico della produzione e del mantenimento degli zone file usati dal Server DNS che viene presentato pubblicamente come "primario". Questa architettura è resa possibile dal fatto che quanto dichiarato all'interno dei singoli zone files dal record SOA non fa testo relativamente ai rapporti master-slave che sussistono tra i Server DNS e al come si scambiano tra loro gli zone files: l'unico dato rilevante in questo senso è quanto essi dichiarano reciprocamente nel file named.conf. Pertanto il Server DNS2 può presentarsi sulla carta come "primario" sfruttando il record SOA dello zone file associato al dominio aziendale anche se in realtà scarica lui stesso suddetto zone file dall'Hidden Master. Scopo di questa scelta è nascondere il più possibile l'esistenza dell'Hidden Master, che non figurerà nemmeno nella lista di NameServer pubblicizzati dal top level domain .it per mezzo dei record NS. Un quarto Server DNS protetto anch'esso dall'Internal Firewall è invece un Caching-only al servizio della rete aziendale (DNS4). Su tutti questi Server dovrà essere stato preventivamente installato il software **bind9** e dovranno essere state aggiunte al file /etc/rc.local le seguenti linee volte ad assicurare il lancio del daemon **named** all'avvio della macchina:

```
if [ -f /usr/sbin/named ] then
        /usr/sbin/named;
        echo -n '/usr/sbin/named' > /dev/console
fi
```

Dovranno inoltre essere stati opportunamente configurati sia il file **/etc/resolv.conf** che il file **/etc/bind/named.conf** , oltre che una serie di zone file secondari che dipendono da quest'ultimo. Specifichiamo tuttavia che nelle versioni più recenti di bind il file named.conf

contiene solo linee volte a includere altri file secondari, ragion per cui è opportuno intervenire direttamente su questi: nello specifico si tratta dei file **named.conf.default-zones**, **named.conf.options** e infine **named.conf.local**. Il primo di questi tre file contiene le dichiarazioni relative alle zone associate all'interfaccia di loopback (che a loro volta fanno riferimento ai due zone files **/etc/bind/db.local** e **/etc/bind/db.127**) e specifica anche il file contenente i record dei tredici Root Server da usare in caso di risoluzione ricorsiva (solitamente è **/usr/share/dns/root.hints**). Questi file sono in ultima analisi uguali in tutti i Server DNS e dovrebbero risultare già configurati in automatico una volta completata l'installazione. Se così non fosse, le zone da dichiarare nel file named.conf.default-zones (o in alternativa direttamente in named.conf) sono:

```
zone "localhost" {
        type master;
        file "/etc/bind/db.local";
        allow-transfer {none;};
};

zone "127.in-addr.arpa" {
        type master;
        file "/etc/bind/db.127";
        allow-transfer {none;};
};

zone "." {
        type hint;
        file "/usr/share/dns/root.hints";
        allow-transfer {none;};
};
```

Mentre il file **/etc/bind/db.local** può essere così configurato:

```
$TTL  86400
@     IN     SOA   localhost.      root.localhost. (
                   2022011201     ; serial, numero di versione dello zone file
```

```
                        43200       ; refresh, dopo quanto controllare se è cambiato
                        3600        ; retry, dopo quanto riprovare sul server primario
                        3600000     ; expire, dopo quanto scade definitivamente
                        2592000     ; minimum ttl, applicato alle risposte nxdomain
)
;
@       IN      NS      localhost.
@       IN      A       127.0.0.1
```

Questa può invece essere una configurazione valida per **/etc/bind/db.127** :

```
$TTL    86400
@       IN      SOA     localhost.      root.localhost. (
                        2022011201      ; serial
                        43200           ; refresh
                        3600            ; retry
                        3600000         ; expire
                        2592000         ; minimum ttl
)
;
@       IN      NS      localhost.
1.0.0   IN      PTR     localhost.
```

Per configurare il file **/usr/share/dns/root.hints** è invece sufficiente far riferimento alla lista dei tredici Root Server, che per ovvi motivi è facilmente reperibile in rete. In alternativa si può omettere il file nella zona e dare direttamente il valore "mirror" all'attributo type per richiedere il download della lista direttamente da Internet.

I file **named.conf.options** e **named.conf.local** contengono invece opzioni e zone che variano a seconda del Server DNS specifico sulla base del suo ruolo. Nei prossimi paragrafi li esamineremo quindi nel dettaglio.

5.3 Server DNS4 (Caching-only aziendale)

L'host di indirizzo IP 170.44.32.8 localizzato all'interno dell'edificio C è il Server DNS Caching-only messo a disposizione degli host aziendali, nonché il primo che viene contattato quando essi devono risolvere un hostname sconosciuto in accordo con quanto stabilito nella configurazione del loro file /etc/resolv.conf. Questo Server si farà carico della risoluzione avvalendosi anche del supporto dei Server DNS pubblici inclusi nella lista "forw" (appartenenti a Google), per poi salvare il risultato localmente in cache in modo da rendere più rapide le risoluzioni degli hostname consultati più di frequente. In questo Server il file **/etc/resolv.conf** è configurato in modo leggermente diverso rispetto agli altri host, giacché non trattandosi di un Resolver only esso può risolvere gli hostname sconosciuti ricorrendo innanzitutto a se stesso:

```
domain identerprises.it
nameserver 127.0.0.1
nameserver 85.37.17.51
nameserver 85.38.28.97
```

Questo Server non ha autorità su nessuna zona, per cui il file **/etc/bind/named.conf.local** non conterrà nulla. Questa invece è la configurazione del file **/etc/bind/named.conf.options** :

```
acl "trusted-networks" {
        localhost;
        170.44.0.0/16;
};

acl "forw" {
        8.8.8.8;
        8.8.4.4;
}

options {
        directory "/var/cache/bind";
```

```
        version "Not disclosed";
        allow-query { trusted-networks; };
        forwarders {  forw; };
        recursion yes;
};
```

Il parametro "allow-query" impostato su "trusted-networks" preclude a qualsiasi host che non faccia parte della lista "trusted-networks" la possibilità di richiedere risoluzioni, mentre il parametro "forwarders" impostato su "forw" specifica che il nostro Server DNS inoltrerà per prima cosa ai server inclusi nella lista "forw" il nome da risolvere al fine di farsi aiutare da loro nella risoluzione. Infine, il parametro "recursion" impostato su "yes" specifica che il Server farà di tutto per risolvere un hostname, incluso risalire la gerarchia dei domini tramite query ricorsive. Va qui fatto un appunto a parte relativo al Server delle Applicazioni Aziendali. Essendo localizzato dietro l'Internal Firewall della rete aziendale il suo indirizzo IP non viene mai pubblicizzato all'esterno per motivi di sicurezza. Di conseguenza, nessuno dei NameServer a cui si affidano gli host aziendali potrà mai essere d'aiuto nello scoprire il suo indirizzo IP. Affinché gli host della rete aziendale riescano a risolverne l'hostname raccomandiamo quindi la risoluzione statica tramite l'inclusione nel loro file **/etc/hosts** della seguente linea:

```
170.44.32.7     app.identerprises.it
```

Analogamente si può considerare la possibilità di aggiungere a questo file anche altre linee relative ai Server aziendali i cui indirizzi IP sono noti pubblicamente, in modo da garantire maggiore robustezza in caso di guasto ai Server DNS aziendali. Raccomandiamo però di non mostrare mai l'indirizzo del Server di Backup, nemmeno in questo file.

5.4 Server DNS1 (Hidden Master)

L'host di indirizzo IP 170.44.32.6 localizzato all'interno dell'edificio C è il Server DNS responsabile della creazione degli zone file associati al dominio identerprises.it , il cui scaricamento è consentito solo al Server DNS2 (170.44.16.3). Questo Server non si fa mai carico di nessuna query di risoluzione, ragion per cui il suo file **/etc/bind/named.conf.options** sarà così configurato:

```
options {
        directory "/var/cache/bind";
        version "Not disclosed";
        allow-transfer { 170.44.16.3; };
        allow-query { none; };
        recursion no;
};
```

Il file **/etc/bind/named.conf.local** conterrà invece le seguenti zone:

```
zone "identerprises.it" {
        type master;
        file "/etc/bind/identerprises.it.db";
        notify explicit;
        also-notify { 170.44.16.3; };
};

zone "44.170.in-addr.arpa" {
        type master;
        file "/etc/bind/44.170.in-addr.arpa.db";
        notify explicit;
        also-notify { 170.44.16.3; };
};
```

La combinazione di "notify explicit" e "also-notify { 170.44.16.3; }" è necessaria perché la

semplice opzione "notify yes" in questo caso non avrebbe effetto, giacché il suo funzionamento prevede l'invio della notifica a tutti i Server di cui esiste un record NS nello zone file ad eccezione di quello identificato come primario tramite il record SOA: peccato che in questo caso il destinatario sia proprio lui. Lo zone file **identerprises.it.db** (relativo alla risoluzione diretta per il dominio identerprises.it) sarà così configurato:

```
$TTL  86400
$ORIGIN identerprises.it.
@       IN      SOA   dns2.identerprises.it.  root.identerprises.it. (
                      2022011201   ; serial
                      43200        ; refresh
                      3600         ; retry
                      3600000      ; expire
                      2592000      ; minimum ttl
)
;
        IN      NS    dns2.identerprises.it.
        IN      NS    dns3.identerprises.it.
        IN      MX    10      mail.identerprises.it.
;
mail    IN      A     170.44.16.5
dns3    IN      A     170.44.16.6
dns2    IN      A     170.44.16.3
web     IN      A     170.44.16.4
www     IN      CNAME       web
@       IN      CNAME       web
```

Il file **44.170.in-addr.arpa.db** (relativo alla risoluzione inversa) conterrà invece:

```
$TTL  86400
$ORIGIN 44.170.in-addr.arpa.
@       IN      SOA   dns2.identerprises.it.  root.identerprises.it. (
                      2022011201   ; serial
                      43200        ; refresh
```

```
                        3600            ; retry
                        3600000         ; expire
                        2592000         ; minimum ttl
)
;
        IN      NS      dns2.identerprises.it.
        IN      NS      dns3.identerprises.it.
        IN      MX      10      mail.identerprises.it.
;
5.16    IN      PTR     mail.identerprises.it.
6.16    IN      PTR     dns3.identerprises.it.
3.16    IN      PTR     dns2.identerprises.it.
4.16    IN      PTR     web.identerprises.it.
```

L'assenza in entrambi i file di record di tipo "A" e "PTR" relativi ai Server che si trovano dietro l'Internal Firewall non è casuale, perché questi Server non sono fatti per essere contattati dall'esterno e per motivi di sicurezza non vogliamo quindi mai divulgare i loro indirizzi IP. Analogamente, l'Hidden Master non include mai se stesso nella lista ufficiale dei Server DNS associati al dominio (implementata tramite i record "NS" dello zone file).

5.5 Server DNS2 (Primario)

L'host di indirizzo 170.44.16.3 localizzato nella DMZ costituisce il Server DNS che all'esterno viene presentato come "primario" nonostante esso scarichi segretamente gli zone file dall'Hidden Master. Insieme al suo compagno secondario (170.44.16.6) il suo scopo è permettere agli utenti esterni di reperire gli indirizzi IP dei Server i cui servizi sono fruibili anche all'esterno della rete aziendale. In virtù di ciò questo Server verrà predisposto affinché accetti query da chiunque, ma siccome non vogliamo inavvertitamente offrirlo agli utenti esterni come Caching-only gratuito avremo anche cura di assegnare al parametro "recursion" il valore "no", in modo che il Server si limiti a rispondere solo alle richieste relative agli zone file su cui ha autorità. Alla luce di quanto appena detto, il file **/etc/bind/named.conf.options**

sarà così configurato:

```
options {
        directory "/var/cache/bind";
        version "Not disclosed";
        allow-query { any; };
        recursion no;
};
```

Mentre il file **/etc/bind/named.conf.local** conterrà invece:

```
zone "identerprises.it" {
        type slave;
        file "/etc/bind/identerprises.it.db";
        masters { 170.44.32.6; };
        allow-transfer { 170.44.16.6; };
        notify yes;
};

zone "44.170.in-addr.arpa" {
        type slave;
        file "/etc/bind/44.170.in-addr.arpa.db";
        masters { 170.44.32.6; };
        allow-transfer { 170.44.16.6; };
        notify yes;
};
```

I file **identerprises.it.db** e **44.170.in-addr.arpa.db** non vanno ovviamente configurati perché questo Server li riceve direttamente dall'Hidden Master.

5.6 Server DNS3 (Secondario)

L'host di indirizzo 170.44.16.6 localizzato nella DMZ è il Server DNS3: esso è un semplice Server DNS secondario che scarica gli zone files relativi al dominio dell'azienda dal Server DNS2 (il primario), che in ultima analisi rimane l'unico a essere consapevole dell'esistenza di un Hidden Master. Per il Server DNS3 valgono considerazioni simili a quelle già fatte per il Server DNS2: anch'esso accetterà query da chiunque ma mai con la ricorsione attiva. Il file **/etc/bind/named.conf.options** sarà quindi così configurato:

```
options {
        directory "/var/cache/bind";
        version "Not disclosed";
        allow-query { any; };
        recursion no;
};
```

Mentre il file **/etc/bind/named.conf.local** conterrà invece:

```
zone "identerprises.it" {
        type slave;
        file "/etc/bind/identerprises.it.db";
        masters { 170.44.16.3; };
};

zone "44.170.in-addr.arpa" {
        type slave;
        file "/etc/bind/44.170.in-addr.arpa.db";
        masters { 170.44.16.3; };
};
```

5.7 Server Web

Il Server Web si trova anch'esso nella DMZ in modo da risultare raggiungibile dagli host esterni alla rete aziendale: nello specifico è l'host di indirizzo 170.44.16.4. Oltre alla configurazione già vista per gli host generici, su questa macchina devono essere installati sia **php** che il software **apache2**. Una volta configurato, il comando da usare per lanciarlo è **systemctl start apache2.service**. La configurazione di apache2 inizia dal file **/etc/apache2/apache2.conf** e prevede molteplici aspetti tra cui la predisposizione all'ascolto sulle porte 80 e 443 e l'associazione di un certificato al dominio identerprises.it (l'uso del software open source CERTBOT può semplificare il lavoro). Altri passaggi riguardano invece la creazione di un virtual host, che inizia dalla creazione di una directory in cui mettere i file html relativi al sito web. Essa dovrà trovarsi in **/var/www** e avere permessi e owner configurati in modo opportuno. Per farlo basta lanciare da root i seguenti comandi:

mkdir /var/www/identerprises.it
chown -R www-data:www-data /var/www/identerprises.it
chmod -R 755 /var/www/identerprises.it

Nella directory **/etc/apache2/sites-available** sono contenuti tutti i file di configurazione associati ai vari virtual host. In questa cartella bisogna creare un file **identerprises.it.conf** e inserirvi le seguenti linee:

```
<VirtualHost *:80>
        ServerAdmin admin@identerprises.it
        ServerName identerprises.it
        ServerAlias www.identerprises.it web.identerprises.it
        DocumentRoot /var/www/identerprises.it
        ErrorLog ${APACHE_LOG_DIR}/error_identerprises.log
        CustomLog ${APACHE_LOG_DIR}/error_identerprises.log combined
</VirtualHost>
```

In seguito, bisogna aver cura di attivare il file di configurazione appena scritto e di disattivare quello di default. Per farlo vanno inseriti i comandi:

a2dissite 000-default.conf
a2ensite identerprises.it.conf

Va infine aggiunta la riga "ServerName identerprises.it" al file **/etc/apache2/conf-available/servername.conf** in modo che possa essere risolto correttamente il nome dell'host. A seconda del bisogno si può inoltre intervenire sul file **/etc/apache2/conf-enabled/security.conf** per modificare ulteriori impostazioni di sicurezza relative al sito.

5.8 Server Mail

Anche il Server Mail è stato posizionato nella DMZ in modo che gli host esterni alla rete aziendale possano contattarlo: è l'host di indirizzo 170.44.16.5. In aggiunta a quanto già visto per gli host generici su questa macchina deve essere installato il software **sendmail**, la cui configurazione implica il lavorare su diversi file. Il primo di questi è **/etc/mail/access**, i cui parametri permettono di specificare:
-Quali host possono affidare al server della mail che non verrà ulteriormente instradata (OK)
-Quali host possono affidargli della mail da instradare a destinatari terzi (RELAY)
-Quali host vanno rifiutati in toto (REJECT o in alternativa codici di errore personalizzati).
-Il delay specifico che riceve un host quando si connette al server (GreetPause)
-Il limite di connessioni simultanee che può stabilire un host (ClientConn)
-Il limite di connessioni simultanee che può stabilire un host in un certo intervallo (ClientRate)
Alla luce di quanto appena detto aggiungeremo quindi queste linee al file:

```
Connect:170.44        RELAY
ClientConn:170.44     2
ClientRate:170.44     2
GreetPause:170.44     5
170.44                RELAY
identerprises.it      RELAY
```

Nel file **/etc/mail/local-host-names** devono invece essere dichiarati esplicitamente i domini per conto dei quali il Server Mail riceve la posta. Aggiungeremo quindi a "localhost" le seguenti linee:

identerprises.it
mail.identerprises.it

Va poi modificato leggermente il file **/etc/mail/sendmail.mc**, da cui deriva direttamente il principale file di configurazione di sendmail (sendmail.conf). In particolare, alla riga DAEMON_OPTIONS('Family=inet, Name=MTA-v4, Port=smtp, Addr=127.0.0.1')dnl va rimosso il "Addr=127.0.0.1" finale in modo che sendmail processi le mail di tutto il dominio. Va inoltre aggiunta alla fine del file la riga FEATURE('relay_entire_domain') in modo da abilitare l'inoltro per qualsiasi host. A questo punto si può passare alla creazione delle mailbox, sfruttando il comando di Linux adibito alla creazione di nuovi utenti:

useradd --create-home -s /sbin/nologin andrea
passwd andrea
useradd --create-home -s /sbin/nologin debora
passwd debora
useradd --create-home -s /sbin/nologin postmaster
passwd postmaster
useradd --create-home -s /sbin/nologin admin
passwd admin
useradd --create-home -s /sbin/nologin generic
passwd generic

Ogni volta che viene inserito un comando del tipo "passwd <utente>" va inserita due volte la password da impostare per suddetta mailbox. Affinché la posta possa arrivare correttamente nelle mailbox appena create deve prima essere definita una mappatura che le associ agli indirizzi mail corrispondenti. Questo viene fatto tramite il file **/etc/mail/virtusertable** , in cui vanno quindi incluse le linee:

root@identerprises.it root
andrea@identerprises.it andrea

debora@identerprises.it debora
postmaster@identerprises.it postmaster
admin@identerprises.it admin
@identerprises.it generic

Il file **/etc/mail/aliases** può invece essere usato per definire aliases il cui scopo è assegnare ruoli predefiniti a utenti specifici. Come semplice esempio possiamo includere le linee:

postmaster: andrea, debora
admin: andrea, debora

Ricordiamo infine che affinché tutte le modifiche apportate entrino in vigore bisogna aggiornare i database lanciando il comando **make** all'interno della directory /etc/mail prima di avviare sendmail. Per assicurarne il lancio automatico a ogni avvio della macchina è sufficiente inserire il comando **service sendmail start** nel file /etc/rc.local.

5.9 Server Applicazioni Aziendali

La configurazione di questo Server non verrà esaminata nel dettaglio perché le sue caratteristiche specifiche dipendono dalla volontà dell'azienda. Vogliamo comunque sottolineare che il Server (il cui indirizzo IP è 170.44.32.7) si trova all'interno della rete aziendale perché non deve risultare accessibile agli utenti esterni. Ai fini della definizione di politiche di sicurezza (aspetto che verrà esaminato nel dettaglio nel prossimo Capitolo) supporremo che questo Server usi tcp sulla porta **4777**.

5.10 Server di Backup

Il Server di Backup si trova anch'esso all'interno della rete aziendale affinché non risulti raggiungibile dagli utenti esterni, nello specifico all'indirizzo 170.44.48.3. Il compito di questo Server è copiare in modo automatico determinati file da tutti gli altri Server a intervalli regolari di una settimana. Supporremo in questo senso che in ognuno degli altri Server sia stato creato ad hoc un utente di nome "backup" nella cui home directory sarà presente una sottodirectory "dati" che al momento della copia ospiterà tutti i dati da trasferire. Sul Server di Backup sarà invece stata creata la directory /Backup in cui conservare le copie dei file. Per la copia verrà usato il protocollo **sftp**, che si appoggia a sua volta a **ssh**. Sul Server di Backup dovrà quindi essere stato lanciato il comando **ssh-keygen** per generare la coppia di chiavi (pubblica e privata) da usare durante il login ssh. Raccomandiamo di cambiare i permessi di lettura, scrittura ed esecuzione dei due file relativi alle chiavi appena generate in modo da precludere l'accesso ad eventuali altri utenti (per farlo è sufficiente il comando **chmod 700**). Sui Server che ospitano i dati da copiare dovrà invece risultare attivo al momento del login il daemon **sshd**, che a sua volta invocherà **sftp-server**. Vogliamo inoltre che il Server di Backup venga autenticato senza richiesta di password. Per far ciò la sua chiave pubblica dovrà figurare nel file **/home/backup/.ssh/authorized_keys** in ognuno dei Server a cui vuole connettersi: il comando **ssh-copy-id** può essere usato per esportarla agevolmente. Raccomandiamo inoltre di impostare i permessi della directory .ssh a 700 e quelli del file authorized_keys a 640 (lettura e scrittura per l'Owner, sola lettura per il Group). Sempre sui Server destinatari del login raccomandiamo inoltre di intervenire sul file **/etc/ssh/sshd_config** rimuovendo il commento alla linea "PasswordAuthentication" e impostandone a "no" il valore, in modo da disattivare in toto i login ssh tramite password e prevenire così eventuali tentativi di accesso abusivi basati sull'uso di brute force. Vanno infine specificati degli opportuni comandi per automatizzare il processo di copia e predisporne l'esecuzione a delle ore precise. Nello specifico, vogliamo che i Server principali si rendano disponibili alla copia solo in una breve finestra (dalle 0:00 alle 6:00 di Domenica). Il Server di Backup sfrutterà quindi l'apertura di questa finestra per lanciare lo script preconfigurato /Backup/beginCopy.sh in cui sono contenuti i comandi che avviano le operazioni di copia. Il file **/etc/crontab** di ognuno dei Server destinatari dovrà quindi contenere le linee:

```
0 0     * * 0   root    service sshd start
```

```
0 6     * * 0   root    service sshd stop
```

Mentre il file **/etc/crontab** del Server di Backup dovrà contenere la linea:

```
1 0     * * 0   backup          /Backup/beginCopy.sh
```

Lo script deve essere stato configurato con permessi tali da permetterne l'esecuzione all'utente "backup" creato ad hoc. Le istruzioni contenute in esso aprono in sequenza sessioni sftp verso ognuno dei Server per poi copiarne i file. Un esempio può essere:

```
#!/bin/bash

sftp backup@170.44.16.3 && sftp> get /home/backup/dati /Backup/DNS2 ; sftp> exit
sftp backup@170.44.16.4 && sftp> get /home/backup/dati /Backup/Web ; sftp> exit
sftp backup@170.44.16.5 && sftp> get /home/backup/dati /Backup/Mail ; sftp> exit
sftp backup@170.44.16.6 && sftp> get /home/backup/dati /Backup/DNS3 ; sftp> exit
sftp backup@170.44.32.6 && sftp> get /home/backup/dati /Backup/DNS1 ; sftp> exit
sftp backup@170.44.32.7 && sftp> get /home/backup/dati /Backup/App ; sftp> exit
sftp backup@170.44.32.8 && sftp> get /home/backup/dati /Backup/DNS4 ; sftp> exit
exit 0
```

Presumiamo ovviamente che lo spazio su disco a disposizione del Server di Backup sia notevolmente maggiore rispetto agli altri Server e che le sei ore in cui i Server mantengono attivo il daemon sshd bastino a portare a compimento le operazioni. Se così non dovesse essere, la finestra temporale può essere progressivamente estesa riconfigurando il file /etc/crontab.

6. PROTEZIONE DELLA RETE

6.1 Disattivazione di Telnet, rlogin, rsh, rcp

I comandi Telnet, rlogin, rsh, rcp sono generalmente deprecati in virtù di vari rischi che comportano dal punto di vista della sicurezza, perciò ne raccomandiamo la disattivazione su tutti gli host della rete aziendale. Le funzionalità offerte da questi comandi andrebbero invece fruite tramite **ssh**, che ne costituisce una controparte più sicura. Per disattivare i comandi è sufficiente accedere al file **/etc/inetd.conf** e commentare le righe associate ad essi. In alcune versioni di Linux il file /etc/inetd.conf potrebbe non essere più disponibile, nel qual caso la disattivazione dovrà essere svolta intervenendo sui file secondari associati a xinetd, che risiedono nella directory /etc/xinet.d .

6.2 Configurazione del TCP Wrapper

Il TCP Wrapper è un utile strumento di host hardening che permette di definire regole volte a precludere l'accesso ai servizi attivi su una macchina sulla base di determinate condizioni. Per attivarlo è necessario configurare i file /etc/hosts.deny e /etc/hosts.allow, linkare xinetd, portmap ed ssh alla libreria **/usr/lib/libwrap.a** e lanciare il daemon **tcpd**. Siccome le regole immesse nel file /etc/hosts.allow hanno precedenza su quelle immesse nel file /etc/hosts.deny , raccomandiamo di inserire preventivamente nel file **/etc/hosts.deny** di tutti i Server aziendali la seguente linea:

```
ALL : ALL : spawn /bin/date %c >> /var/log/intrusion.log
```

Questa impostazione predispone come comportamento di default il rifiuto in toto di qualsiasi accesso a qualsiasi servizio: eventuali tentativi di intrusione verranno inoltre registrati in un file di log creato ad hoc. Gli unici servizi a cui verrà invece consentito l'accesso sono quelli specificati nel file **/etc/hosts.allow**, che dovrà quindi riflettere i servizi specifici offerti dal singolo Server. Ad esempio il file hosts.allow del Server Mail conterrà le seguenti linee:

in.sendmail: ALL
in.imapd: ALL
in.ipop3d: ALL
in.sshd: 170.44.48.3

Abbiamo qui permesso l'accesso a tutti gli host (sia interni che esterni all'azienda) ai daemon sendmail, imapd e ipop3d perché tentativi di accesso esterni a sendmail potrebbero essere di Server Mail che stanno cercando di inoltrare posta destinata all'azienda mentre tentativi di accesso esterni a imapd e ipop3d potrebbero provenire da dipendenti dell'azienda che vogliono accedere alla loro casella di posta da un host remoto. L'accesso a ssh viene invece permesso solo al Server di Backup affinché possa copiare periodicamente il file system. Altri Server presenteranno una configurazione diversa, ad esempio nel Server di Backup il file /etc/hosts.allow può essere lasciato completamente vuoto dato che il lavoro di questo Server prevede solo l'aprire connessioni con gli altri Server mentre esso stesso non deve mai offrire alcun servizio. Specifichiamo inoltre che i daemon monitorati da tcpd purtroppo possono variare a seconda del sistema operativo specifico, ragion per cui il TCP Wrapper potrebbe non essere in grado di monitorare determinati servizi: per questi servizi si renderà quindi necessario intervenire con contromisure alternative, come ad esempio il filtraggio dei pacchetti in input. Si ricorda in questo senso che **iptables** è di default sempre disponibile su qualsiasi host Linux e può quindi essere sfruttato anche come strumento di host hardening oltre che per il filtraggio dei pacchetti in transito nei Router. Nel Server Web ad esempio si possono usare i seguenti comandi per predisporre il rifiuto di tutti i pacchetti ad eccezione di quelli associati alle porte 80 e 443 (per http) e 22 (per ssh, ma solo se a connettersi è il server di backup):

iptables -F INPUT
iptables -P INPUT DROP
iptables -A INPUT -p tcp -m multiport --dports 80,443 -j ACCEPT
iptables -A INPUT -p tcp -s 170.44.48.3 --dport 22 -j ACCEPT

6.3 Configurazione di Xinetd

Xinetd può essere usato come ulteriore strumento di host hardening in virtù della sua capacità di monitorare i servizi di un host e definire opzioni personalizzate per ognuno di essi. Per abilitarlo, il primo file da configurare è **/etc/xinetd.conf** , in cui risiedono opzioni di default che vengono applicate globalmente. Un esempio di configurazione può essere il seguente:

```
defaults
{
        instances = 60
        log_type = SYSLOG authpriv
        log_on_success = HOST PID
        log_on_failure = HOST
        cps = 25 30
}

includedir /etc/xinetd.d
```

Con questa configurazione xinetd accetterà un massimo di 60 richieste attive in contemporanea con un cap di 25 connessioni al secondo per ogni singolo servizio (superate le quali suddetto servizio verrà negato per 30 minuti). Sia i tentativi di accesso riusciti che quelli falliti verranno salvati in un log (quello specificato tramite l'attributo "log_type"). L'ultima linea serve invece a includere in toto il contenuto della directory **/etc/xinetd.d**, in cui risiedono i file secondari associati ai singoli servizi: è proprio in questi file che andrebbero definite le opzioni specifiche di ognuno di essi. Per predisporre l'accettazione delle connessioni ssh solo quando provengono dal Server di Backup possiamo ad esempio creare un file /etc/xinetd.d/ssh con all'interno:

```
service ssh
{
        disable = no
        socket_type = stream
        wait = no
        user = root
        server = /usr/sbin/sshd
```

```
        only_from = 170.44.48.3
        access_times = 00:00-06:00
}
```

I file nella directory /etc/xinetd.d possono inoltre essere usati anche per predisporre la completa disattivazione di un servizio, nello specifico impostando a "yes" il valore dell'attributo "disable". Se ad esempio si volesse disattivare Telnet sarebbe sufficiente creare un file /etc/xinetd.d/telnet e inserirvi all'interno:

```
service telnet
{
        socket_type = stream
        wait = no
        user = root
        server = /usr/sbin/in.telnetd
        disable = yes
}
```

In alternativa si può lasciare attivo un servizio che non si vuole usare al fine di sfruttarlo come sensore volto a rilevare eventuali tentativi di port scan. Per farlo si imposta un nuovo valore all'attributo "flags" e si specifica un "deny_time", usando la seguente configurazione:

```
service telnet
{
        flags = SENSOR
        socket_type = stream
        wait = no
        user = nobody
        deny_time = 1440
        disable = no
}
```

I file in /etc/xinetd.d risulteranno ovviamente diversi da Server a Server sulla base dei servizi offerti. Una volta ultimata la loro creazione e configurazione, per avviare xinetd sarà sufficiente usare il comando **service xinetd start** .

6.4 Configurazione di iptables nell'Internal Firewall

L'Internal Firewall si occupa del filtraggio dei pacchetti che transitano dalla DMZ verso l'interno della rete aziendale e viceversa. Come principio generale proponiamo di lasciare completamente libero il flusso verso la DMZ (diretto verso l'interfaccia "eth0") e di limitare solo quello verso l'interno della rete aziendale (diretto verso l'interfaccia "eth1"). Per implementare il filtraggio useremo ovviamente **iptables** essendo di default sempre presente nei sistemi Linux. Ricordiamo però che le regole di filtraggio andrebbero reimpostate a ogni nuovo avvio della macchina, ragion per cui è bene includere i seguenti comandi direttamente nel file /etc/rc.local (o in alternativa in un opportuno script da lanciare sempre in rc.local):

```
iptables -F FORWARD
iptables -P FORWARD DROP
iptables -A FORWARD -i eth1 -j ACCEPT
iptables -A FORWARD -o eth1 -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -o eth1 -s 170.44.16.3 -d 170.44.32.6 -p tcp --dport 53 -j ACCEPT
iptables -A FORWARD -o eth1 -s 170.44.16.4 -d 170.44.32.7 -p tcp --dport 4777 -j ACCEPT
iptables -A FORWARD -o eth1 -s 170.44.16.0/20 -d 170.44.32.8 -p udp --dport 53 -j ACCEPT
iptables -A FORWARD -o eth1 -s 170.44.16.1 -p ospf -j ACCEPT
```

Le prime due regole servono rispettivamente a svuotare la chain FORWARD e predisporre come policy di base il DROP di tutti i pacchetti. La terza regola predispone l'accettazione dei pacchetti che transitano dall'interno verso la DMZ. La quarta regola predispone l'accettazione dei pacchetti che transitano verso l'interno della rete aziendale a patto che siano associati a connessioni già stabilite (potrebbero ad esempio essere risposte di Server che sono stati contattati dagli host aziendali). La quinta regola stabilisce che il Server DNS che si presenta pubblicamente come primario può contattare l'Hidden master usando TCP sulla porta 53 (per richiedere uno zone transfer). La sesta regola stabilisce che il Server Web può inviare richieste al Server delle Applicazioni aziendali (stiamo qui supponendo che il Server Web possa aver bisogno di alcune informazioni prodotte dall'App aziendale per fornire dati accurati sulla pagina web dell'azienda). La settima regola permette agli host della DMZ di inviare query al Server DNS Caching-only messo a disposizione degli host aziendali. L'ultima regola serve a far sì che eventuali pacchetti ospf generati dall'External Firewall possano transitare verso l'interno della rete aziendale. Questa regola potrebbe invero risultare superflua perché prevediamo che il trasferimento di informazioni ospf dalla Backbone area alla Stub area verrà

in realtà gestito proprio dall'Internal Firewall, essendo lui stesso il Backbone Router di raccordo nella gerarchia ospf.

6.5 Configurazione di iptables nell'External Firewall

L'External Firewall si occupa del filtraggio dei pacchetti che transitano dall'esterno della rete aziendale verso la DMZ e viceversa. Specifichiamo che l'interfaccia tramite cui l'External Firewall è connesso alla DMZ è "eth0" mentre le due interfacce esterne sono "fib0" e "fib1". Similmente a come abbiamo già visto per l'Internal Firewall la configurazione delle regole di filtraggio avverrà tramite comandi **iptables** che vanno salvati in /etc/rc.local, nello specifico i seguenti:

```
iptables -F FORWARD
iptables -P FORWARD DROP
iptables -A FORWARD -i eth0 -j ACCEPT
iptables -A FORWARD -o eth0 -m state --state ESTABLISHED, RELATED -j ACCEPT
iptables -A FORWARD -o eth0 -d 170.44.16.4 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -o eth0 -d 170.44.16.4 -p tcp --dport 443 -j ACCEPT
iptables -A FORWARD -o eth0 -d 170.44.16.3 -p udp --dport 53 -j ACCEPT
iptables -A FORWARD -o eth0 -d 170.44.16.6 -p udp --dport 53 -j ACCEPT
iptables -A FORWARD -o eth0 -d 170.44.16.5 -p tcp -m multiport --dports 25,587,143,993,110,995 -j ACCEPT
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.32.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.48.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.64.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.80.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.96.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.112.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.160.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.128.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.176.0/20 -j MASQUERADE
iptables -t nat -A POSTROUTING -o fib0,fib1 -s 170.44.144.0/20 -j MASQUERADE
```

I principi dietro queste regole sono simili a quelli già visti per l'Internal Firewall. Per prima cosa si svuota la chain FORWARD e si predispone come policy di default il DROP di tutti i pacchetti. Poi si predispone l'accettazione dei pacchetti che provengono dall'interno della DMZ (-i eth0). Quelli diretti verso la DMZ (-o eth0) vengono invece accettati solo a patto che

siano relativi a connessioni già aperte in precedenza. Altre regole specifiche predispongono infine l'accettazione di pacchetti provenienti da host esterni e diretti a Server specifici che risiedono nella DMZ (-o eth0) sulla base del numero di porta. Se la destinazione è il Server Web (170.44.16.4) verranno quindi accettati i pacchetti TCP relativi alle porte 80 e 443 (http). Se la destinazione è il Server DNS2 (170.44.16.3) o il Server DNS3 (170.44.16.6) verranno accettati i pacchetti UDP relativi alla porta 53 (dns). Se la destinazione è il Server Mail (170.44.16.5) verranno accetti i pacchetti TCP relativi alle porte 25, 587 (smtp), 143, 993 (imap), 110, 995 (pop3). Le ultima righe sono invece regole relative alla tabella NAT il cui scopo è mascherare gli indirizzi IP degli host che risiedono nelle varie sottoreti aziendali interne quando aprono connessioni in uscita, in modo da ridurre le informazioni relative all'organizzazione logica della rete aziendale che vengono divulgate all'esterno. Abbiamo optato per il target MASQUERADE in luogo di SNAT per evitare problemi qualora l'indirizzo IP delle interfacce esterne cambiasse improvvisamente a seguito di decisioni prese dagli ISP. Qualora l'azienda lo ritenesse opportuno può valutare anche l'aggiunta di Intrusion Detection Systems (IDS) come ulteriore misura di controllo del traffico di rete.

7. PREVENTIVO DI SPESA

Il seguente preventivo non comprende gli host, che supponiamo essere già disponibili presso l'azienda. Per la quantità di Cavo UTP necessaria abbiamo fatto una stima grossolana in cui supponiamo che gli host all'interno dei vari edifici si trovino mediamente a una distanza di 15 metri dai loro Switch. Il numero di host aziendali da connettere (460 host generici a cui si aggiungono i 20 host specifici che figurano nello schema fisico) è stato quindi moltiplicato per la distanza media stimata (15m) e a questo numero sono stati aggiunti i 250m di cavo necessari a interconnettere gli edifici esternamente, per una lunghezza totale di 7450m. Il cablaggio in fibra comprende solo i tratti che connettono i vari edifici, per una lunghezza totale di 4150m. Relativamente agli Switch buona parte dei prodotti in commercio arriva fino a un massimo di 48 porte, oltre le quali si sfrutta lo stacking di dispositivi multipli. Per fornire agli edifici A, B, D ed E un numero adeguato di porte RJ45 abbiamo quindi optato per lo stacking in ognuno di essi di tre Switch da 24 porte, mentre nell'edificio C abbiamo optato per lo stacking di sei Switch da 48 porte. Per la DMZ è invece sufficiente un singolo Switch da 24 porte in virtù del numero limitato di host previsti.

Elemento	**Quantità**	**Prezzo**	**Totale**
1. Cavo UTP Cat 5	7450 m	1 € / m	7450 €
2. Cavo Fibra	4150 m	1,5 € / m	6225 €
3. Switch S3910-24TF	13	380 €	4940 €
4. Switch S5810-48TS-P	6	1880 €	15040 €
5. Progettazione	90 h	25 € / h	2250 €
6. Installazione	700 h	40 € / h	28000 €

Il costo complessivo stimato è quindi di 63905 €.