33.1 Indirizzi e nomi

La gestione diretta degli indirizzi IP in forma numerica può essere utile in fase di progetto di una rete, ma a livello di utente è una pretesa praticamente inaccettabile. Per questo, agli indirizzi IP numerici si affiancano quasi sempre dei nomi che teoricamente potrebbero anche essere puramente fantastici e senza alcuna logica. Ogni volta che

si fa riferimento a un nome, il sistema è (o dovrebbe essere) in grado di convertirlo nel numero IP corrispondente. In pratica, si usa di solito la convenzione dei nomi a dominio, come descritto in parte nella sezione (32.4.10).

Ci sono due metodi per trasformare un nome in un indirizzo IP e viceversa: un elenco contenuto nel file '`/etc/hosts`' oppure l'uso di un servente DNS.

Qui si analizzano inizialmente '`/etc/hosts`' e gli altri file di configurazione legati alla traduzione dei nomi; successivamente si passa alla trattazione della gestione di un servente DNS con il quale si ottiene un servizio di risoluzione dei nomi (*name server*).

33.1.1 Configurazione del tipo di conversione: file «/etc/host.conf»

Prima di procedere con la trasformazione di un nome in un indirizzo IP, occorre definire in che modo si vuole che il sistema esegua questa operazione. Il file di configurazione attraverso il quale si definisce ciò è '`/etc/host.conf`', ma anche attraverso l'uso di variabili di ambiente si può intervenire in questa configurazione.

Il file '`/etc/host.conf`' viene usato per determinare quali servizi usare per risolvere i nomi a dominio. Ogni riga rappresenta un'opzione di funzionamento, inoltre il simbolo '**#**' rappresenta l'inizio di un commento. Solitamente vengono specificate solo due direttive: '**order**' e '**multi**', come nell'esempio seguente:

```
order hosts,bind
multi on
```

Nella prima riga, l'opzione '**order**' indica l'ordine dei servizi. In questo caso si utilizza prima il file '`/etc/hosts`' (33.1.2.1) e quindi si interpella il servizio di risoluzione dei nomi. Nella seconda riga, '**multi on**', abilita la possibilità di trovare all'interno del file '`/etc/hosts`' l'indicazione di più indirizzi IP per lo stesso nome. Un evento del genere può verificarsi quando uno stesso elaboratore ha due o più connessioni per la rete e per ognuna di queste ha un indirizzo IP diverso.

Tabella 33.2. Alcune direttive.

Direttiva	Descrizione
`order {hosts|bind|nis}[,…[,…]]`	L'opzione '**order**' richiede uno o più argomenti (separati da spazio, virgola, punto e virgola o due punti) indicanti la sequenza di servizi attraverso cui si deve tentare di risolvere un nome.
`multi {on|off}`	L'opzione '**multi**' attiva o disattiva la possibilità di trovare all'interno del file '`/etc/hosts`' l'indicazione di più indirizzi IP per lo stesso nome.

Attraverso l'uso delle variabili di ambiente ***RESOLV_HOST_CONF***, ***RESOLV_SERV_ORDER*** e ***RESOLV_SERV_MULTI***, è possibile interferire con la configurazione del file '`/etc/host.conf`', come descritto nella tabella successiva.

Tabella 33.3. Alcune variabili di ambiente.

Variabile	Descrizione
RESOLV_HOST_CONF	Se esiste e non è vuota, definisce il nome di un file alternativo a '`/etc/host.conf`'.
RESOLV_SERV_ORDER	Definisce l'ordine dei servizi di risoluzione dei nomi, senza tenere conto di quanto eventualmente già definito attraverso l'opzione '**order**' nel file '`/etc/host.conf`'.

Variabile	Descrizione
RESOLV_SERV_MULTI	Può contenere la stringa '**on**' oppure '**off**', con lo stesso significato dell'opzione '**multi**' del file '`/etc/host.conf`' e serve a sostituirsi all'eventuale dichiarazione fatta nel file stesso.

33.1.2 File per la conversione

Prima che esistessero i serventi DNS si dovevano risolvere i nomi attraverso l'uso di un file unico, contenente un elenco di indirizzi IP associato ai nomi rispettivi. Teoricamente, utilizzando un servente DNS questo file potrebbe non essere più necessario. In pratica conviene utilizzare ugualmente questo vecchio metodo per garantirsi l'accessibilità alla rete locale anche quando l'eventuale servente DNS non dovesse funzionare.

33.1.2.1 File «/etc/hosts»

Il file '`/etc/hosts`' viene usato per convertire i nomi degli elaboratori in numeri IP e viceversa. È particolarmente utile la sua compilazione all'interno di piccole reti che non dispongono di un servente DNS. Nell'ambito di una rete locale può essere predisposto uguale per tutti gli elaboratori connessi, così da facilitare per quanto possibile l'aggiornamento all'interno di questi. Segue un estratto di esempio di questo file.[1]

```
# Necessario per il "loopback" IPv4.
127.0.0.1                   localhost.localdomain  localhost

# Indirizzi IPv4.
192.168.1.1                 dinkel.brot.dg  dinkel
192.168.1.2                 roggen.brot.dg  roggen
192.168.2.1                 weizen.mehl.dg  weizen

# Necessario per il loopback IPv6.
::1                         ip6-localhost   ip6-loopback

# Necessari per il multicast IPv6.
fe00::0                     ip6-localnet
ff00::0                     ip6-mcastprefix
ff02::1                     ip6-allnodes
ff02::2                     ip6-allrouters
ff02::3                     ip6-allhosts

# Indirizzi IPv6.
fec0::1:2a0:24ff:fe77:4997  dinkel.brot.dg  dinkel
fec0::1:280:5fff:fea6:6d3d  roggen.brot.dg  roggen
fec0::2:280:adff:fec8:a981  weizen.mehl.dg  weizen
```

In pratica, il file può contenere righe vuote o commenti (le righe che iniziano con il simbolo '**#**') e righe che iniziano con un indirizzo IP (sia IPv4, sia IPv6). Dopo l'indirizzo IP, separato da spazi o caratteri di tabulazione, inizia l'elenco dei nomi a esso abbinati, anche questo può essere separato da spazi o da caratteri di tabulazione.

Di solito, si indica il nome a dominio completo (FQDN o *Fully qualified domain name*), seguito eventualmente da possibili abbreviazioni o soprannomi.

Come già accennato, è possibile creare un file '`/etc/hosts`' identico per tutti gli elaboratori della propria rete locale. Ma se la rete locale si articola in sottoreti, è normale che il dominio di appartenenza di ogni sottorete cambi. Nell'esempio visto, si fa riferimento a due sottoreti IPv4 e IPv6: 192.168.1.0 e fec0::1::/64 denominata *brot.dg*; 192.168.2.0 e fec0::2::/64 denominata *mehl.dg*. In questa situazione, potrebbe capitare che un elaboratore nella rete *mehl.dg* abbia lo stesso nome locale di un altro collocato nelle rete *brot.dg*. Per questo, l'attribuzione di soprannomi, o semplicemente di abbreviazioni, deve essere deve essere tale da non creare ambiguità, oppure deve essere evitata. A questo fa eccezione il caso dell'indirizzo di *loopback*: ogni elaboratore è bene che si chiami *localhost*.

33.1.2.2 File «/etc/networks»

Il file '`/etc/networks`' viene usato per convertire i nomi delle sottoreti in codici IPv4. Come nel caso del file '`/etc/hosts`', può essere predisposto in forma unificata per tutti i nodi di una stessa rete, così da facilitare per quanto possibile l'aggiornamento all'interno di questi. Segue un estratto di esempio di questo file:

```
localdomain     127.0.0.0

brot.dg         192.168.1.0
mehl.dg         192.168.2.0
```

La presenza di questo file non è indispensabile; in effetti, la gestione delle sottoreti attraverso l'uso diretto degli indirizzi IP non dovrebbe essere un problema. Il vantaggio di avere questo file, sta nell'utilizzo del programma '**`route`**' per visualizzare la tabella di instradamento: gli indirizzi di rete vengono trasformati nei nomi ottenuti dal file '`/etc/networks`'.

È bene chiarire che normalmente non si utilizza il servente DNS per risolvere i nomi della rete; quindi, di solito, la gestione dei nomi si attua solo attraverso la predisposizione di questo file.

33.1.2.3 File «/etc/resolv.conf»

Quando il file '`/etc/hosts`' non basta, si deve poter accedere a un servizio di risoluzione dei nomi, ovvero a un servente DNS. Viene usato il file '`/etc/resolv.conf`' per conoscere l'indirizzo o gli indirizzi dei servizi di risoluzione dei nomi di competenza della rete cui si appartiene. Se non si intende utilizzare il sistema DNS per risolvere i nomi della propria rete, oppure si dispone di un solo elaboratore, ma si vuole accedere alla rete Internet, devono essere indicati gli indirizzi dei servizi di risoluzione dei nomi forniti dall'ISP (*Internet service provider*), ovvero dal fornitore di accesso a Internet.

Questo file può contenere righe vuote o commenti (le righe che iniziano con il simbolo '`#`') e righe che iniziano con un nome di opzione seguite normalmente da un argomento. Le opzioni utilizzabili sono descritte nella tabella successiva.

Tabella 33.6. Alcune direttive.

Direttiva	Descrizione
`nameserver` ***indirizzo_ip_servente_dns***	L'opzione '**`nameserver`**' è la più importante e permette di definire l'indirizzo IP di un servizio di risoluzione dei nomi. Se questa opzione non viene utilizzata, si fa riferimento a un servizio locale, raggiungibile precisamente all'indirizzo 127.0.0.1. Il file '`/etc/resolv.conf`' può contenere più righe con questa opzione, in modo da poter fare riferimento a servizi di risoluzione dei nomi alternativi quando quello principale non risponde.
`domain` ***nome_a_dominio***	Stabilisce il dominio predefinito per le interrogazioni del servizio di risoluzione dei nomi.
`search` ***nome_a_dominio*`...`	Definisce un elenco di domini possibili (l'elenco è separato da spazi o caratteri di tabulazione) per le interrogazioni del servizio di risoluzione dei nomi.

Una configurazione normale non ha bisogno dell'indicazione delle opzioni '**`domain`**' e '**`search`**'. Se il file '`/etc/resolv.conf`' si limita a contenere opzioni '**`nameserver`**', questo può essere standardizzato su tutta la rete locale.

Segue un esempio in cui si utilizza il servizio di risoluzione dei nomi offerto dall'indirizzo IP 8.8.8.8 ed eventualmente, in sua mancanza, dall'indirizzo 8.8.4.4.

```
nameserver 8.8.8.8
nameserver 8.8.4.4
```

33.2 DNS come base di dati distribuita

Prima di descrivere in pratica l'allestimento di un sistema DNS per la risoluzione dei nomi, è necessario comprendere, almeno a grandi linee, i concetti di partenza: domini, zone, record di risorsa.

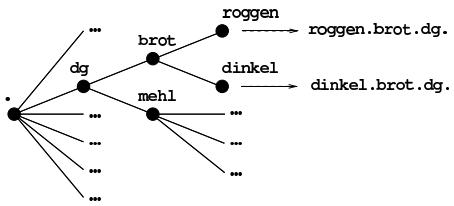
33.2.1 Nome a dominio

Alla base del sistema esiste il nome a dominio, che è la forma con cui si rappresenta un indirizzo attraverso una denominazione strutturata. Per esempio, *dinkel.brot.dg* potrebbe essere il nome a dominio che corrisponde a un nodo preciso nella rete (in tal caso di parla di FQDN), nome che si può scomporre secondo una sequenza gerarchica, come si vede nella figura 33.8.

Figura 33.8. Scomposizione del nome a dominio *dinkel.brot.dg*.

I nomi a dominio, nel loro insieme, costituiscono una struttura ad albero, in cui la radice è il dominio principale, rappresentato con un punto singolo oppure lasciato sottinteso. Ogni nodo di questo albero è un dominio, rappresentato attraverso l'unione dei nomi dei nodi attraversati a partire dalla radice, indicandoli da destra verso sinistra, separati con un punto uno dall'altro, come si intende meglio dalla figura 33.9.

Figura 33.9. Struttura ad albero dei nomi a dominio.



In linea di principio, le «foglie» di questo albero, ovvero i nodi terminali, dovrebbero corrispondere a dei nodi di rete; tuttavia, benché sconsigliabile, è possibile che un nodo non terminale nell'albero dei nomi a dominio, corrisponda a un nodo di rete. Seguendo l'esempio della figura 33.9, *dinkel.brot.dg* e *roggen.brot.dg* sono intesi come nodi di rete, ma non si può escludere che lo siano anche *brot.dg* e *dg* stesso.

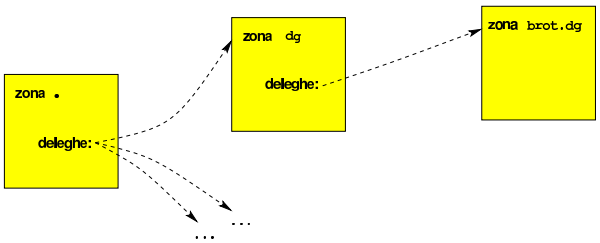
La lunghezza di un nome a dominio si esprime in ***livelli***, intesi come quantità di nodi che si devono attraversare, esclusa la radice. Per esempio, il nome *dinkel.brot.dg* ha tre livelli. In particolare, si fa riferimento al primo nodo successivo alla radice come al dominio di primo livello, noto in generale come TLD, ovvero *top level domain*. Pertanto, il nome *dinkel.brot.dg* appartiene quindi al dominio di primo livello *dg*.

33.2.2 Zone

Secondo il DNS, i livelli gerarchici di suddivisione delle competenze sono le ***zone***, le quali si sovrappongono all'albero dei domini. Una zona riguarda un ramo dell'albero dei domini, a partire da un

certo nodo in poi, ma al suo interno, questa zona può demandare la competenza per dei rami inferiori ad altre zone.

Figura 33.10. Suddivisione in zone.



L'esempio della figura 33.10 dovrebbe aiutare a comprendere il meccanismo: la zona principale è competente per tutto l'albero dei domini, ma demanda ad altre zone la competenza per il dominio *dg* e per altri domini che dipendono direttamente da quello principale. La zona '**dg**' è competente per il dominio *dg* e per tutti i suoi sottodomini, tranne *brot.dg* che viene demandato a un'altra zona (con lo stesso nome); infine, la zona '**brot.dg**' è competente per tutti i suoi sottodomini.

Da questo esempio si dovrebbe comprendere che le zone seguono la struttura dei domini, ma non hanno necessariamente la stessa frequenza di suddivisione.

33.2.3 Record di risorsa

Ogni zona organizza le informazioni di sua competenza in quelli che sono chiamati record di risorsa. Questi record definiscono l'associazione tra un nome a dominio e un'altra informazione, in base al tipo di record. Per esempio, per cercare l'indirizzo IPv4 associato a un certo nome a dominio, si consultano i record di tipo «A»; per conoscere il servizio di risoluzione dei nomi competente per un certo nome a dominio (in questo caso inteso come zona), si consultano i record di tipo «NS».

L'interrogazione di un servizio DNS corrisponde all'interrogazione di una base di dati, in cui, il risultato è il record desiderato. Naturalmente, tutto questo avviene generalmente in modo trasparente, per opera dei programmi che ne hanno bisogno, senza disturbare l'utente.

33.2.4 Risoluzione inversa

La base di dati che costituisce il sistema DNS serve principalmente per due cose: trovare l'indirizzo numerico corrispondente a un nome a dominio e trovare il nome a dominio a partire dall'indirizzo numerico (ammesso che sia disponibile un nome). Tuttavia, il sistema DNS gestisce **solo** nomi a dominio, pertanto la risoluzione da indirizzo a nome avviene attraverso un meccanismo un po' strano.

Infatti, alcuni domini sono speciali, perché servono a rappresentare, in qualche modo, un indirizzo numerico. Per esempio, *4.3.2.1.in-addr.arpa* è uno di questi domini speciali, che fa riferimento implicito all'indirizzo IPv4 1.2.3.4 (in questo caso, trattandosi di IPV4, l'inversione delle cifre è voluta).

I domini più importanti che servono a rappresentare in qualche modo un indirizzo numerico sono *in-addr.arpa* per gli indirizzi IPv4 e *ip6.arpa* per gli indirizzi IPv6.

33.2.5 Registrazione di un nome a dominio

I nomi a dominio utilizzati all'interno di Internet si ottengono attraverso una fase chiamata ***registrazione***. Intuitivamente si può comprendere che la registrazione di un nome avvenga facendo una richiesta a chi è competente per la zona a cui questo nome appartiene. Per esempio, se si vuole registrare il nome *rosso.marrone.nero*, si deve chiedere la cosa a chi gestisce la zona *marrone.nero*.

Generalmente, si registrano nomi a dominio di secondo livello, pertanto ci si rivolge a quella che viene chiamata ***autorità di registrazione*** (nota anche con la sigla RA, per *Registration authority*), competente per il dominio di primo livello a cui si vuole fare riferimento. Per esempio, se si volesse registrare il nome *prova.it*, occorrerebbe rivolgersi all'autorità di registrazione italiana: *http://www.nic.it* . In questo contesto particolare, il dominio di primo livello è noto come TLD, ovvero *Top level domain*; inoltre, nell'ambito della normativa italiana, si parla preferibilmente di ***nomi a dominio***.

La registrazione di un nome a dominio è paragonabile alla registrazione di un marchio, con la differenza fondamentale che, per essere usato, richiede l'aggiornamento del DNS.

La procedura per la registrazione di un nome a dominio attraverso un'autorità di registrazione, può essere complessa, ma soprattutto, la procedura cambia da un'autorità all'altra. Per questo e anche per sollevare dall'incombenza legata alla gestione tecnica del DNS, esistono diverse aziende che offrono la loro assistenza per la registrazione e la cura del DNS. Generalmente, è conveniente rivolgersi a intermediari di questo tipo, purché siano chiari i servizi che vengono offerti e le condizioni relative; soprattutto è indispensabile verificare che la registrazione venga effettuata a nome del cliente (persona o ente) che vuole ottenere tale registrazione.

Normalmente, le autorità di registrazione pubblicano le informazioni sui domini di loro competenza. Queste notizie dovrebbero essere accessibili attraverso il protocollo NICNAME, noto anche con il nome WHOIS, descritto nei documenti RFC 812 e RFC 954. In un sistema GNU si ottengono queste informazioni con il programma Whois,[2] il quale è in grado di decidere da solo quale servente interpellare, a meno di indicare qualcosa di diverso attraverso le opzioni della riga di comando:

```
whois [opzioni] oggetto
```

Generalmente, si utilizza il programma indicando semplicemente il nome a dominio a cui si è interessati. L'esempio seguente ottiene le informazioni disponibili sul dominio *linuxdidattica.org*:

```
$ whois informaticalibera.net [Invio]

 Whois Server Version 2.0
 …
    Domain Name: INFORMATICALIBERA.NET
    Registrar: KEY-SYSTEMS GMBH
    Whois Server: whois.rrpproxy.net
    Referral URL: http://www.key-systems.net
    Name Server: NS1.NICE.NET
    Name Server: NS2.NICE.NET
    Name Server: NS3.NICE.NET
    Status: ok
    Updated Date: 18-nov-2009
    Creation Date: 12-apr-2007
    Expiration Date: 12-apr-2014
 …
 DOMAIN: INFORMATICALIBERA.NET

 RSP: NICE S.r.l.
 URL: http://www.niceweb.eu

 owner-contact: P-DCG606
 owner-organization: daniele giacomini
 owner-fname: daniele
 owner-lname: giacomini
 owner-street: via Morganella Est, 21
 owner-city: Ponzano Veneto (TV)
 owner-zip: I-31050
 owner-country: IT
 owner-phone: +39.04221835202
 owner-email: appunti2@gmail.com

 admin-contact: P-DCG606
 admin-organization: daniele giacomini
 admin-fname: daniele
 admin-lname: giacomini
 admin-street: via Morganella Est, 21
 admin-city: Ponzano Veneto (TV)
 admin-zip: I-31050
 admin-country: IT
 admin-phone: +39.04221835202
```

```
admin-email: appunti2@gmail.com

tech-contact: P-NOO151
tech-organization: NICE S.r.l.
tech-fname: NICE
tech-lname: Operations
tech-street: business unit niceweb.it Via Nomentana 186
tech-city: Roma
tech-state: RM
tech-zip: 00162
tech-country: IT
tech-phone: +39.06874461
tech-email: support@niceweb.it

billing-contact: P-NCB327
billing-organization: NICE S.r.l.
billing-fname: NICE
billing-lname: Billing
billing-street: business unit niceweb.it Via Nomentana 186
billing-city: Roma
billing-state: RM
billing-zip: 00162
billing-country: IT
billing-phone: +39.06874461
billing-email: billing@niceweb.it

nameserver: ns1.nice.net
nameserver: ns2.nice.net
nameserver: ns3.nice.net
```

33.3 Esempio di configurazione del DNS

Per la gestione di un servizio DNS si fa riferimento generalmente al pacchetto BIND,[3] rappresentato concretamente dal '**named**'; tuttavia è bene evitare di fare confusione: '**named**' è il nome del demone che compie il lavoro; BIND è il nome del pacchetto che racchiude tutto il necessario alla gestione del DNS, compreso '**named**'.

Si dispone di una piccola rete locale composta da due elaboratori con indirizzi IPv4 e IPv6:

IPv4	IPv6	Nome
192.168.1.1	fec0:0:0:1::1	*dinkel.brot.dg*
192.168.1.2	fec0:0:0:1::2	*roggen.brot.dg*

Il primo di questi due elaboratori è connesso a Internet (con un'altra coppia di indirizzi) e viene predisposto per gestire un servizio di risoluzione dei nomi attraverso il demone '**named**'. La connessione esterna serve solo all'elaboratore '**dinkel**' e non permette all'altro elaboratore di accedere a Internet.

33.3.1 Prima di gestire un servente DNS

Quando non si gestisce localmente un servizio di risoluzione dei nomi e si vuole accedere a Internet, è necessario almeno fare uso di un servizio esterno, di solito messo a disposizione dallo stesso fornitore di accesso.

File '`/etc/host.conf`' (sezione 33.1.1)

È il file di configurazione principale dei servizi di rete. Serve in particolare per determinare in che modo si intendono risolvere i nomi a dominio. L'esempio seguente è quello classico, utilizzato quasi sempre.

```
order hosts,bind
multi on
```

L'opzione '**order**' indica l'ordine dei servizi. In questo caso si utilizza prima il file '`/etc/hosts`' e quindi si interpella il servizio di risoluzione dei nomi.

File '`/etc/hosts`' (sezione 33.1.2.1)

Questo file permette di definire i nomi degli elaboratori abbinati al loro indirizzo IP, senza fare uso di un servente DNS. Per entrambi gli elaboratori dell'esempio, va bene il contenuto seguente:


```
127.0.0.1       localhost.localdomain   localhost
::1             ip6-localhost           ip6-loopback

fe00::0         ip6-localnet
ff00::0         ip6-mcastprefix
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters
ff02::3         ip6-allhosts

192.168.1.1     dinkel.brot.dg          dinkel
fec0:0:0:1::1   dinkel.brot.dg          dinkel
192.168.1.2     roggen.brot.dg          roggen
fec0:0:0:1::2   roggen.brot.dg          roggen
```

File '`/etc/networks`' (sezione 33.1.2.2)

Questo file attribuisce i nomi agli indirizzi di rete (solo IPv4). Per entrambi gli elaboratori dell'esempio va bene il contenuto seguente:

```
localdomain     127.0.0.0
brot.dg         192.168.1.0
```

File '`/etc/resolv.conf`' (sezione 33.1.2.3)

Viene usato per conoscere l'indirizzo o gli indirizzi dei servizi di risoluzione dei nomi di competenza della rete cui si appartiene. Se non si vuole gestire questo servizio nella propria rete locale, se ne deve indicare almeno uno esterno per accedere a Internet. Nell'esempio seguente, si fa riferimento a quelli di Google:

```
nameserver 8.8.8.8
nameserver 8.8.4.4
```

33.3.2 Predisposizione di un servente DNS elementare

Il tipo di servizio di risoluzione dei nomi più semplice è quello che si occupa solo di accumulare in una memoria cache gli ultimi indirizzi richiesti, senza avere alcuna competenza di zona. Il servizio viene allestito all'interno dell'elaboratore '**dinkel**'.

File '`/etc/resolv.conf`' (33.1.2.3)

Viene modificato in modo da fare riferimento all'indirizzo locale (*localhost*), dal momento che si intende usare il proprio elaboratore per la gestione del servizio di risoluzione dei nomi.

```
nameserver 127.0.0.1
```

File '`/etc/named.conf`' o '`/etc/bind/named.conf`'

Viene utilizzato da '**named**' come punto di partenza della configurazione del servizio DNS.

```
options {
        directory "/etc/bind";
        listen-on-v6 { any; };
};
zone "." {
        type hint;
        file "named.root";
};
zone "0.0.127.in-addr.arpa" {
        type master;
        file "zone/127.0.0";
};
```

La prima direttiva, che occupa le prime quattro righe, definisce in particolare la directory predefinita per contenere gli altri file di configurazione del servizio di risoluzione dei nomi.

La seconda direttiva indica il file '`named.root`', contenuto in '`/etc/bind/`', che serve come fonte per gli indirizzi necessari a raggiungere i servizi di risoluzione dei nomi del dominio principale (ciò è rappresentato simbolicamente dal punto isolato).

La terza direttiva indica il file '`127.0.0`' contenuto in '`/etc/bind/zone/`', utilizzato come configurazione per la rete dell'elaboratore locale (*localhost*).

in-addr.arpa è un dominio speciale attraverso il quale si definisce che le cifre precedenti rappresentano un indirizzo IPv4 rovesciato.

File '`/etc/bind/named.root`', '`/etc/bind/named.ca`'

Si tratta del file contenente le indicazioni necessarie a raggiungere i servizi di risoluzione dei nomi del dominio principale. Nella consuetudine può avere diversi nomi, tra cui i più importanti sono '`named.root`' e '`named.rc`'. Questo file viene realizzato da un'autorità esterna e viene quindi semplicemente utilizzato così com'è. Segue un esempio di questo.

```
.                       3600000  IN  NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.     3600000      A     198.41.0.4
.                       3600000      NS    B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.     3600000      A     128.9.0.107
.                       3600000      NS    C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET.     3600000      A     192.33.4.12
.                       3600000      NS    D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET.     3600000      A     128.8.10.90
.                       3600000      NS    E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET.     3600000      A     192.203.230.10
.                       3600000      NS    F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET.     3600000      A     192.5.5.241
.                       3600000      NS    G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET.     3600000      A     192.112.36.4
.                       3600000      NS    H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET.     3600000      A     128.63.2.53
.                       3600000      NS    I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET.     3600000      A     192.36.148.17
.                       3600000      NS    J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET.     3600000      A     198.41.0.10
.                       3600000      NS    K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET.     3600000      A     193.0.14.129
.                       3600000      NS    L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET.     3600000      A     198.32.64.12
.                       3600000      NS    M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET.     3600000      A     198.32.65.12
```

File '`/etc/bind/zone/127.0.0`'

Definisce la configurazione per la rete 127.0.0.∗, cioè quella a cui appartiene il nome *localhost*.

```
@  IN  SOA   localhost.localdomain. root.localhost.localdomain. (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                           NS    localhost.localdomain.
1.0.0.127.in-addr.arpa.    PTR   localhost.localdomain.
```

La prima riga, '**`SOA`**' (*Start of authority*), è il preambolo del file. Si riferisce all'origine rappresentata dal simbolo '`@`' (in questo caso '`@`' rappresenta *0.0.127.in-addr.arpa*) e definisce in particolare i dati seguenti:

- l'elaboratore di provenienza, *localhost.localdomain*, indicato in modo assoluto e per questo terminato con un punto;
- l'indirizzo di posta elettronica della persona o del gruppo che mantiene il servizio di risoluzione dei nomi (in questo caso, la notazione '**`root.localhost.localdomain.`**' si riferisce all'utente '**`root@localhost.localdomain`**' e l'indirizzo è assoluto perché termina con un punto);
- il numero di serie, rappresentato in modo da comprendere la data (anno, mese, giorno), seguita da due cifre che permettono di esprimere la versione del giorno.

La seconda riga, NS (*Name server*) indica il nome dell'elaboratore che offre il servizio di risoluzione dei nomi.

La terza riga, PTR, indica che il nome a dominio *1.0.0.127.in-addr.arpa* (ovvero l'indirizzo 127.0.0.1) corrisponde a *localhost.localdomain*.

In pratica, tutto questo definisce un servizio di risoluzione dei nomi che è in grado esclusivamente di interrogare i servizi del livello principale e di tradurre l'indirizzo 127.0.0.1 in *localhost.localdomain*.

33.3.3 Gestire anche la rete locale

Perché il servizio di risoluzione dei nomi sia in grado di gestire anche la rete locale, occorre che possa tradurre i nomi utilizzati nella rete locale in indirizzi IP e viceversa.

File '`/etc/named.conf`' o '`/etc/bind/named.conf`'

Il file viene modificato in modo da fare riferimento ad altri quattro file:

- '`/etc/bind/zone/dg`'
  per la trasformazione dei nomi a dominio appartenenti al dominio principale della rete locale (*dg*) in indirizzi numerici;
- '`/etc/bind/zone/brot.dg`'
  per la trasformazione dei nomi a dominio appartenenti alla rete locale *brot.dg* in indirizzi numerici;
- '`/etc/bind/zone/192.168.1`'
  per la trasformazione degli indirizzi IPv4 appartenenti alla rete locale (192.168.1.∗) in nomi a dominio;
- '`/etc/bind/zone/fec0:0:0:1`'
  per la trasformazione degli indirizzi IPv6 appartenenti alla rete locale (fec0:0:0:1:∗) in nomi a dominio.

```
options {
        directory "/etc/bind";
        listen-on-v6 { any; };
};
//
zone "." {
        type hint;
        file "named.root";
};
//
zone "0.0.127.in-addr.arpa" {
        type master;
        file "zone/127.0.0";
};
zone "1.168.192.in-addr.arpa" {
        type master;
        file "zone/192.168.1";
};
zone "\[xfec0000000000001/64].ip6.arpa" {
        type master;
        file "zone/fec0:0:0:1";
};
zone "dg" {
        type master;
        file "zone/dg";
};
zone "brot.dg" {
        type master;
        file "zone/brot.dg";
};
```

File '`/etc/bind/zone/192.168.1`'

Definisce la configurazione per la rete locale 192.168.1.∗.

```
@  IN  SOA   dinkel.brot.dg. root.dinkel.brot.dg.  (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                            NS    dinkel.brot.dg.

1.1.168.192.in-addr.arpa.   PTR   dinkel.brot.dg.
2.1.168.192.in-addr.arpa.   PTR   roggen.brot.dg.
```

In tal modo è possibile determinare che l'indirizzo 192.168.1.1 corrisponde a *dinkel.brot.dg* e che 192.168.1.2 corrisponde a *roggen.brot.dg*.[4]

File '/etc/bind/zone/dg'

Definisce la configurazione per la rete locale *dg*.

```
@  IN           SOA  dinkel.brot.dg. root.dinkel.brot.dg.  (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                NS   dinkel.brot.dg.
```

In tal modo è possibile determinare non ci sono nomi corrispondenti a nodi, che dipendono direttamente dalla zona *dg*.

File '/etc/bind/zone/brot.dg'

Definisce la configurazione per la rete locale della zona *brot.dg*.

```
@  IN           SOA  dinkel.brot.dg. root.dinkel.brot.dg.  (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                NS   dinkel.brot.dg.

dinkel.brot.dg. A      192.168.1.1
dinkel.brot.dg. A6   0 fec0:0:0:1:0:0:0:1
roggen.brot.dg. A      192.168.1.2
roggen.brot.dg. A6   0 fec0:0:0:1:0:0:0:2
```

In tal modo è possibile determinare che l'indirizzo *dinkel.brot.dg* corrisponde a 192.168.1.1 per IPv4 e a fec0:0:0:1:0:0:0:1 per IPv6; inoltre, *roggen.brot.dg* corrisponde a 192.168.1.2 per IPv4 e a fec0:0:0:1:0:0:0:2 per IPv6.

File '/etc/bind/zone/127.0.0'

Dal momento che adesso l'elaboratore locale può essere identificato con un nome più significativo del semplice *localhost*, conviene modificare anche il file '/etc/bind/zone/127.0.0', benché ciò non sia strettamente necessario.

```
@  IN  SOA   dinkel.brot.dg. root.dinkel.brot.dg.  (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                            NS    dinkel.brot.dg.

1.0.0.127.in-addr.arpa.     PTR   localhost.localdomain.
```

File '/etc/bind/zone/fec0:0:0:1'

Definisce la trasformazione degli indirizzi IPv6 appartenenti alla rete locale (fec0:0:0:1:∗) in nomi a dominio.


```
@       IN     SOA   dinkel.brot.dg. root.dinkel.brot.dg.  (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                              NS      dinkel.brot.dg.

\[x0000000000000001/64]         PTR     dinkel.brot.dg.
\[x0000000000000002/64]         PTR     roggen.brot.dg.
```

Si osservi il fatto che è possibile avere indirizzi IPv4 e indirizzi IPv6 che si risolvono in un nome in comune.

33.3.4 Gli altri elaboratori della rete

Gli altri elaboratori della rete locale, in questo caso solo *roggen.brot.dg*, fanno uso del servizio di risoluzione dei nomi offerto da *dinkel.brot.dg*, cioè 192.168.1.1, quindi il loro file '/etc/resolv.conf' deve contenere il riferimento a questo:

```
nameserver 192.168.1.1
```

33.3.5 Gestire la posta elettronica locale

Per inserire anche l'indicazione di un servente di posta elettronica, basta modificare il file '/etc/bind/zone/brot.dg' contenuto nell'elaboratore *dinkel.brot.dg*, aggiungendo la riga '**MX**':

```
@  IN           SOA  dinkel.brot.dg. root.dinkel.brot.dg.  (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                NS   dinkel.brot.dg.
                MX   10 dinkel.brot.dg.

dinkel.brot.dg. A      192.168.1.1
dinkel.brot.dg. A6   0 fec0:0:0:1:0:0:0:1
roggen.brot.dg. A      192.168.1.2
roggen.brot.dg. A6   0 fec0:0:0:1:0:0:0:2
```

33.3.6 Gestire gli alias

Spesso è conveniente definire dei nomi fittizi riferiti a elaboratori che ne hanno già uno. Viene modificato il file '/etc/bind/zone/brot.dg' in modo da aggiungere gli alias *www.brot.dg* e *ftp.brot.dg*, che fanno riferimento sempre al solito *dinkel.brot.dg* che però svolge anche le funzioni di servente HTTP e FTP:

```
@  IN           SOA   dinkel.brot.dg. root.dinkel.brot.dg. (
                                1998031800 ; Serial
                                28800      ; Refresh
                                7200       ; Retry
                                604800     ; Expire
                                86400 )    ; Minimum
                NS     dinkel.brot.dg.

                MX     10 dinkel.brot.dg.

www.brot.dg. CNAME  dinkel.brot.dg.
ftp.brot.dg. CNAME  dinkel.brot.dg.

dinkel.brot.dg. A      192.168.1.1
dinkel.brot.dg. A6   0 fec0:0:0:1:0:0:0:1
roggen.brot.dg. A      192.168.1.2
roggen.brot.dg. A6   0 fec0:0:0:1:0:0:0:2
```

33.3.7 Isolamento dall'esterno

Se la rete locale funziona senza poter accedere alla rete Internet esterna, conviene evitare che si tenti di interrogare i servizi di risoluzione dei nomi del dominio principale: basta commentare la direttiva che attiva questa ricerca nel file 'named.conf'.

File ‘/etc/named.conf’ o ‘/etc/bind/named.conf’

I commenti possono iniziare con una doppia barra obliqua (‘//’), terminando così alla fine della riga, oppure possono essere inseriti tra ‘/*’ e ‘*/’.

```
options {
        directory "/etc/bind";
        listen-on-v6 { any; };
};
//
// La zona root viene esclusa attraverso dei commenti
//zone "." {
//      type hint;
//      file "named.root";
//};
//
zone "0.0.127.in-addr.arpa" {
        type master;
        file "zone/127.0.0";
};
zone "1.168.192.in-addr.arpa" {
        type master;
        file "zone/192.168.1";
};
zone "\[xfec0000000000001/64].ip6.arpa" {
        type master;
        file "zone/fec0:0:0:1";
};
zone "dg" {
        type master;
        file "zone/dg";
};
zone "brot.dg" {
        type master;
        file "zone/brot.dg";
};
```

33.4 Gestione del servizio di risoluzione dei nomi

In un sistema Unix il servizio di risoluzione dei nomi viene offerto generalmente dal programma ‘**named**’. Per verificarne il funzionamento si possono usare dei programmi specializzati nella sua interrogazione.

33.4.1 Utilizzo di «named»

Il programma ‘**named**’ è il demone che compie in pratica il servizio di risoluzione dei nomi del pacchetto BIND. Si avvale di un file di avvio (o di configurazione) che in passato è stato ‘/etc/named.boot’ e attualmente è invece ‘/etc/named.conf’, oppure ‘/etc/bind/named.conf’. Eventualmente, se viene indicato un nome di file negli argomenti, viene utilizzato quel file invece di quello predefinito.

```
named [opzioni] [[-b] file_di_avvio]
```

Nei sistemi in cui si attiva la gestione di un servizio di risoluzione dei nomi, ‘**named**’ viene avviato dalla procedura di inizializzazione del sistema (Init), ma può anche essere avviato manualmente.

A ogni modo, se la propria distribuzione GNU non mette a disposizione uno script specifico (per esempio il file ‘/etc/init.d/bind’), si può controllare il funzionamento o il riavvio di questo demone attraverso il programma ‘**rndc**’, che fa sempre parte di BIND. Quello che segue è solo una semplificazione dello schema sintattico complessivo:

```
rndc {start|stop|restart}
```

Il significato dell’argomento è intuitivo: avvia, ferma o riavvia il servizio. Evidentemente, è necessario riavviare il servizio ogni volta che si modifica la configurazione.

Il DNS utilizza una serie di protocolli, tra cui anche UDP. Se ci si trova a essere protetti da un firewall che esclude il transito dei pacchetti UDP, per poter interpellare gli altri servizi di risoluzione dei nomi delle zone che sono al di fuori della propria competenza locale, occorre aggiungere una direttiva che rinvia le richieste a un servizio esterno. Questa situazione può verificarsi quando la propria connessione a Internet avviene attraverso un ISP attento ai problemi di sicurezza e che usa questa politica di protezione.

33.4.2 Nslookup

Nslookup[5] è il programma tradizionale per l’interrogazione del servizio di risoluzione dei nomi. Esistono delle alternative a questo programma, forse più semplici da usare, ma conviene comunque conoscerne almeno l’uso elementare.

L’eseguibile che svolge il lavoro è ‘**nslookup**’ e si utilizza secondo il modello sintattico seguente:

```
nslookup [opzioni] [nodo_da_trovare | - servente ]
```

```
nslookup [opzioni] nodo_da_trovare [servente]
```

Nslookup offre due modalità di funzionamento: interattiva e non interattiva. Nel primo caso, il programma offre un invito attraverso il quale inserire dei comandi, nel secondo tutto si conclude con l’uso di argomenti nella riga di comando.

Si entra nella modalità interattiva quando non vengono forniti argomenti e di conseguenza viene utilizzato il servizio di risoluzione dei nomi predefinito attraverso il file ‘/etc/resolv.conf’, oppure quando il primo argomento è un trattino (‘-’) e il secondo è il nome o l’indirizzo necessario a raggiungere un servente per la risoluzione dei nomi. In tal caso, Nslookup mostra un invito costituito da un semplice simbolo di maggiore:

```
$ nslookup [Invio]

>
```

Per uscire dalla modalità interattiva, si deve usare il comando ‘**exit**’:

```
> exit
```

La modalità non interattiva viene utilizzata quando il nome o l’indirizzo di un nodo di rete da cercare viene indicato come primo argomento. In tal caso, il secondo argomento opzionale è il nome o l’indirizzo per raggiungere un servizio di risoluzione dei nomi.

Nelle situazioni più comuni, ci si limita a usare il programma per tradurre un indirizzo in nome o viceversa. Segue la descrizione di alcuni esempi:

- `$ nslookup 192.168.1.2` [Invio]

  restituisce il nome e l’indirizzo Internet corrispondente al nodo di rete indicato attraverso il numero IP;

- `$ nslookup roggen.brot.dg.` [Invio]

  restituisce il nome e l’indirizzo Internet corrispondente al nodo di rete indicato attraverso il nome a dominio completo;

- `$ nslookup roggen.brot.dg. ns2.brot.dg` [Invio]

  interpella il servizio di risoluzione dei nomi offerto dall’elaboratore *ns2.brot.dg* per ottenere le informazioni su *roggen.brot.dg*.

33.4.3 Host

Host[6] è un programma alternativo a Nslookup, il cui utilizzo è, per certi versi, un po’ più semplice. L’eseguibile che compie il lavoro è ‘**host**’:

```
host [opzioni] {nodo | -l zona} [servente_dns]
```

Le opzioni e le relative funzionalità a disposizione sono molte. Per lo studio dettagliato delle possibilità di questo programma conviene consultare la sua pagina di manuale: *host(1)*.

Dal modello sintattico presentato si può osservare che il primo argomento dopo le opzioni, è il nome o l'indirizzo di un nodo di rete, oppure il nome di una zona, espressa attraverso il nome a dominio relativo. Eventualmente, si può aggiungere un secondo argomento che permette di specificare un servente DNS alternativo a quello predefinito. La tabella seguente riepiloga le opzioni più comuni.

Tabella 33.31. Alcune opzioni.

Opzione	Descrizione
`-v`	Permette di ottenere maggiori informazioni.
`-t` *tipo*	Elenca i record del tipo specificato. Per fare riferimento a tutti i tipi di record, si può usare la parola chiave '**ANY**', oppure l'asterisco (opportunamente protetto, se necessario, dall'interpretazione della shell).
`-l` *zona*	Permette di indicare una zona nel primo argomento, al posto di un nodo di rete particolare, ma non è detto che il servizio interpellato sia disposto a dare tutte queste informazioni.

Seguono alcuni esempi:

- `$ host dinkel.brot.dg` [*Invio*]

  mostra il nome e l'indirizzo corrispondente;

- `$ host 192.168.1.1` [*Invio*]

  mostra l'indirizzo e il nome corrispondente;

- `$ host -l brot.dg` [*Invio*]

  richiede la lista completa dei nodi di rete nella zona *brot.dg*, ma la risposta potrebbe essere omessa dal servente;

- `$ host -l dg` [*Invio*]

  mostra la lista completa dei nodi di rete nella zona *dg*, ma la risposta potrebbe essere omessa dal servente;

- `$ host -l 1.168.192.in-addr.arpa` [*Invio*]

  mostra la lista completa dei nodi di rete nella zona *1.168.192.in-addr.arpa*, ovvero della rete 192.168.1.*, ma la risposta potrebbe essere omessa dal servente;

- `$ host -t AAAA www.aerasec.de` [*Invio*]

  mostra l'indirizzo IPv6, ottenuto da un record AAAA (ammesso che sia disponibile, essendo stato sostituito dai record A6).

33.4.4 Dig

Dig,[7] ovvero *Domain information groper* è un sistema di interrogazione dei servizi DNS, flessibile e complesso nel contempo. Si compone dell'eseguibile '**dig**' che si utilizza secondo lo schema seguente, il quale appare qui semplificato rispetto alla sintassi completa:

```
dig [@servente_dns] [opzioni] [nome_risorsa] [tipo_richiesta] ↩
↪   [opzione…]
```

Un utilizzo comune di questo eseguibile, si traduce nella sintassi seguente:

```
dig [@servente_dns] nome_risorsa [tipo_richiesta]
```

L'esempio seguente restituisce il record «A» della risorsa *dinkel.brot.dg*, assieme ad altre informazioni di contorno:

```
$ dig @127.0.0.1 dinkel.brot.dg A [Invio]
```

Il listato è interrotto per motivi tipografici:


```
; <<>> DiG 9.2.0 <<>> @127.0.0.1 dinkel.brot.dg A
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4122
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; QUESTION SECTION:
;dinkel.brot.dg.                        IN      A

;; ANSWER SECTION:
dinkel.brot.dg.         86400   IN      A       192.168.1.1

;; AUTHORITY SECTION:
brot.dg.                86400   IN      NS      dinkel.brot.dg.
…
```

In pratica si ottiene l'indirizzo IPv4 associato al nome *dinkel.brot.dg*, dal servente DNS raggiungibile all'indirizzo 127.0.0.1. Ma per fare la ricerca opposta (il nome a partire dall'indirizzo), occorre indicare il nome a dominio appartenente a *in-addr.arpa*:

```
$ dig @127.0.0.1 1.1.168.192.in-addr.arpa PTR [Invio]
```

Ecco un piccolo estratto di ciò che Dig può restituire:

```
…
;; ANSWER SECTION:
1.1.168.192.in-addr.arpa. 86400 IN      PTR   dinkel.brot.dg.
…
```

Prima di andare oltre questi esempi elementari, è bene chiarire che se si omette l'indicazione del servente da interrogare, Dig utilizza il primo che riesce a raggiungere dall'elenco contenuto nel file '`/etc/resolv.conf`'; inoltre, se manca l'indicazione del tipo di record da cercare, si intende il tipo «A», ovvero quello che abbina nomi a dominio a indirizzi IPv4.

Appare subito la difficoltà dell'utilizzo di questo strumento, che richiede un conoscenza approfondita del modo in cui si descrivono i file di zona di un servizio DNS.

Per ottenere la risoluzione inversa da un indirizzo al nome corrispondente, si può usare una forma alternativa del comando:

```
dig [@servente_dns] -x indirizzo_numerico
```

Per esempio, per trovare il nome corrispondente al numero 192.168.1.1 si può usare il comando seguente:

```
$ dig @127.0.0.1 -x 192.168.1.1 [Invio]
```

Il risultato è lo stesso già visto per l'interrogazione di un record PTR.

Alla fine degli argomenti normali della riga di comando, si possono aggiungere delle opzioni speciali, che iniziano con il segno '**+**', con le quali si modifica il comportamento di Dig. Tra tutte, merita attenzione l'opzione '**+short**', che consente di ridurre al minimo le informazioni restituite da Dig. Per esempio, il comando seguente interroga il record «A» della risorsa *dinkel.brot.dg*, restituendo semplicemente il numero dell'indirizzo IPv4 corrispondente:

```
$ dig dinkel.brot.dg +short [Invio]

192.168.1.1
```

Come ultima considerazione su Dig, si vuole mostrare cosa succede se si utilizza senza alcun argomento:

```
$ dig [Invio]
```

Se è disponibile l'accesso alla rete esterna, si ottiene il file contenente l'elenco dei serventi DNS competenti per il dominio principale ('`.`'), come ottenuto dall'interrogazione del servente DNS predefinito ('`/etc/resolv.conf`'):

```
; <<>> DiG 9.2.0 <<>>
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19406
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0,
;;        ADDITIONAL: 13

;; QUESTION SECTION:
;.                              IN    NS
```

```
;; ANSWER SECTION:
.                       3430    IN      NS      F.ROOT-SERVERS.NET.
.                       3430    IN      NS      G.ROOT-SERVERS.NET.
.                       3430    IN      NS      H.ROOT-SERVERS.NET.
.                       3430    IN      NS      I.ROOT-SERVERS.NET.
.                       3430    IN      NS      J.ROOT-SERVERS.NET.
.                       3430    IN      NS      K.ROOT-SERVERS.NET.
.                       3430    IN      NS      L.ROOT-SERVERS.NET.
.                       3430    IN      NS      M.ROOT-SERVERS.NET.
.                       3430    IN      NS      A.ROOT-SERVERS.NET.
.                       3430    IN      NS      B.ROOT-SERVERS.NET.
.                       3430    IN      NS      C.ROOT-SERVERS.NET.
.                       3430    IN      NS      D.ROOT-SERVERS.NET.
.                       3430    IN      NS      E.ROOT-SERVERS.NET.

;; ADDITIONAL SECTION:
F.ROOT-SERVERS.NET.   604659  IN      A       192.5.5.241
G.ROOT-SERVERS.NET.   604659  IN      A       192.112.36.4
H.ROOT-SERVERS.NET.   604659  IN      A       128.63.2.53
I.ROOT-SERVERS.NET.   604659  IN      A       192.36.148.17
J.ROOT-SERVERS.NET.   604659  IN      A       198.41.0.10
K.ROOT-SERVERS.NET.   604659  IN      A       193.0.14.129
L.ROOT-SERVERS.NET.   604629  IN      A       198.32.64.12
M.ROOT-SERVERS.NET.   604629  IN      A       202.12.27.33
A.ROOT-SERVERS.NET.   604637  IN      A       198.41.0.4
B.ROOT-SERVERS.NET.   604657  IN      A       128.9.0.107
C.ROOT-SERVERS.NET.   604658  IN      A       192.33.4.12
D.ROOT-SERVERS.NET.   604659  IN      A       128.8.10.90
E.ROOT-SERVERS.NET.   604659  IN      A       192.203.230.10

;; Query time: 4 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Wed May 22 15:28:57 2002
;; MSG SIZE  rcvd: 436
```

Se non ci si fida del servente DNS predefinito, si può richiedere espressamente l'informazione a un nodo di fiducia; per esempio:

```
$ dig @rs.internic.net . ns [Invio]
```

33.4.5 Verifica del funzionamento del servizio

Se è appena stato configurato il servizio di risoluzione dei nomi, si può riavviare (o semplicemente avviare) il servizio utilizzando il programma '**rndc**', oppure un altro messo a disposizione dalla propria distribuzione GNU.

```
# rndc stop [Invio]

# rndc start [Invio]
```

Il demone '**named**' emette alcuni messaggi che vengono annotati nel registro del sistema, generalmente nel file '`/var/log/messages`' (oppure un altro collocato sempre sotto '`/var/log/`', a seconda della configurazione del sistema operativo). È utile consultare il suo contenuto per verificare che la configurazione sia corretta. Trattandosi dell'ultima cosa avviata, i messaggi si trovano alla fine del file.

```
# tail /var/log/messages [Invio]
```

Il listato seguente si riferisce a un esempio di configurazione già apparso in precedenza:

```
May 31 15:20:56 dinkel named[2778]: starting BIND 9.2.0
May 31 15:20:56 dinkel named[2778]: using 1 CPU
May 31 15:20:56 dinkel named[2780]: loading configuration from ↩
↪'/etc/bind/named.conf'
May 31 15:20:56 dinkel named[2780]: listening on IPv6 interfaces, port 53
May 31 15:20:56 dinkel named[2780]: binding TCP socket: address in use
May 31 15:20:56 dinkel named[2780]: listening on IPv4 interface lo, 127.0.0.1#53
May 31 15:20:56 dinkel named[2780]: binding TCP socket: address in use
May 31 15:20:56 dinkel named[2780]: listening on IPv4 interface eth0, ↩
↪192.168.1.1#53
May 31 15:20:56 dinkel named[2780]: binding TCP socket: address in use
May 31 15:20:56 dinkel named[2780]: zone 127.0.0.in-addr.arpa/IN: loaded ↩
↪serial 1
May 31 15:20:56 dinkel named[2780]: 192.168.1:1: no TTL specified; using SOA ↩
↪MINTTL instead
May 31 15:20:56 dinkel named[2780]: zone 1.168.192.in-addr.arpa/IN: loaded ↩
↪serial 1998031800
May 31 15:20:56 dinkel named[2780]: 192.168.2:1: no TTL specified; using SOA ↩
↪MINTTL instead
May 31 15:20:56 dinkel named[2780]: zone 2.168.192.in-addr.arpa/IN: loaded ↩
↪serial 1998031800
May 31 15:20:56 dinkel named[2780]: fec0:0:0:1:1: no TTL specified; using SOA ↩
↪MINTTL instead
May 31 15:20:56 dinkel named[2780]: zone \[xFEC0000000000001/64].ip6.arpa/IN: ↩
↪loaded serial 1998031800
```



```
May 31 15:20:56 dinkel named[2780]: dg:1: no TTL specified; using SOA MINTTL ↩
↪instead
May 31 15:20:56 dinkel named[2780]: zone dg/IN: loaded serial 1998031800
May 31 15:20:56 dinkel named[2780]: brot.dg:1: no TTL specified; using SOA ↩
↪MINTTL instead
May 31 15:20:56 dinkel named[2780]: zone brot.dg/IN: loaded serial 1998031800
May 31 15:20:56 dinkel named[2780]: zone localhost/IN: loaded serial 1
May 31 15:20:56 dinkel named[2780]: running
```

Se qualcosa non va, è lo stesso '**named**' ad avvisare attraverso questi messaggi. Se è andato tutto bene si può provare a vedere cosa accade avviando l'eseguibile '**dig**' senza argomenti:

```
$ dig [Invio]
```

Se il servente DNS è appena stato riavviato e non è disponibile una connessione con l'esterno, si ottiene un responso nullo, dal quale si vede comunque chi ha risposto:

```
; <<>> DiG 9.2.0 <<>>
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 52215
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0,
;;        ADDITIONAL: 0

;; QUESTION SECTION:
;.                               IN      NS

;; Query time: 4 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Wed May 22 16:37:30 2002
;; MSG SIZE  rcvd: 17
```

Alla fine c'è l'indicazione di chi ha risposto e in questo caso si tratta dell'indirizzo 127.0.0.1, ovvero l'elaboratore locale.

Se si è connessi alla rete esterna, si può provare a interrogare il servente per la risoluzione di un nome, per esempio *informaticalibera.net*.[8]

```
$ dig informaticalibera.net [Invio]
```

```
; <<>> DiG 9.5.0-P1 <<>> informaticalibera.net
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12849
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 1

;; QUESTION SECTION:
;informaticalibera.net.         IN      A

;; ANSWER SECTION:
informaticalibera.net.  3600    IN      A       87.24.59.9

;; AUTHORITY SECTION:
informaticalibera.net.  3600    IN      NS      ns2.nice.net.
informaticalibera.net.  3600    IN      NS      ns1.nice.net.
informaticalibera.net.  3600    IN      NS      ns3.nice.net.

;; ADDITIONAL SECTION:
ns2.nice.net.           543     IN      A       87.233.133.47

;; Query time: 416 msec
;; SERVER: 212.216.172.62#53(212.216.172.62)
;; WHEN: Fri Mar 19 15:28:54 2010
;; MSG SIZE  rcvd: 130
```

Dal momento che il servizio di risoluzione dei nomi locale non dispone di tale informazione, per ottenerla ha dovuto interpellare i vari servizi DNS a partire dal dominio principale ('`.`'), fino a quando ha potuto ricevere la risposta. Per evitare di appesantire la rete in caso di richieste analoghe, il nome e l'indirizzo corrispondente vengono memorizzati in modo temporaneo, nella memoria cache.

Quando il servizio di risoluzione dei nomi interpellato è competente per la zona richiesta e non deve rivolgersi altrove per ottenere la risposta, si ha una risposta «autorevole»; diversamente, la risposta generata dalle informazioni accumulate in una memoria provvisoria, non è autorevole.

Per controllare se i file di zona di competenza del servizio di risoluzione dei nomi locale sono corretti, conviene cambiare il tipo di interrogazione, facendo riferimento a tutti i tipi di record della zona che interessa (in questo caso *brot.dg*), attraverso la parola chiave

‘**any**’:

```
$ dig brot.dg any [Invio]

; <<>> DiG 9.2.0 <<>> brot.dg any
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60850
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 2

;; QUESTION SECTION:
;brot.dg.                       IN      ANY

;; ANSWER SECTION:
brot.dg.                86400   IN      SOA     ↩
↪dinkel.brot.dg. root.dinkel.brot.dg. 1998031800 28800 7200 604800 86400
brot.dg.                86400   IN      NS      dinkel.brot.dg.
brot.dg.                86400   IN      MX      10 dinkel.brot.dg.

;; ADDITIONAL SECTION:
dinkel.brot.dg.         86400   IN      A       192.168.1.1

;; Query time: 4 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Wed May 22 17:05:12 2002
;; MSG SIZE  rcvd: 147
```

33.5 File di configurazione più in dettaglio

A questo punto è necessario analizzare un po’ meglio la sintassi del contenuto dei vari file di configurazione utilizzati da ‘**named**’. Il loro significato può essere apprezzato solo dopo il conforto di alcuni esperimenti riusciti con il sistema di risoluzione dei nomi.

Nei file di definizione delle zone i commenti vanno preceduti da un punto e virgola; per quanto riguarda invece il file ‘`named.boot`’, i commenti si realizzano come nel linguaggio C: ‘`/*…*/`’ oppure ‘`//…`’.

33.5.1 File «/etc/named.conf» o «/etc/bind/named.conf»

Il file ‘`named.conf`’ appare già in altre sezioni precedenti. Si riprende qui il solito esempio, con la differenza che la directory predefinita per i file è quella comune.

```
options {
        directory "/var/cache/bind";
        listen-on-v6 { any; };
};
zone "." {
        type hint;
        file "/etc/bind/named.root";
};
zone "0.0.127.in-addr.arpa" {
        type master;
        file "/etc/bind/zone/127.0.0";
};
zone "1.168.192.in-addr.arpa" {
        type master;
        file "/etc/bind/zone/192.168.1";
};
zone "\[xfec0000000000001/64].ip6.arpa" {
        type master;
        file "/etc/bind/zone/fec0:0:0:1";
};
zone "dg" {
        type master;
        file "/etc/bind/zone/dg";
};
zone "brot.dg" {
        type master;
        file "/etc/bind/zone/brot.dg";
};
```

Segue l’elenco e la descrizione delle direttive e delle opzioni più importanti di questo file.

Tabella 33.41. Alcune direttive e opzioni. Si osservi che le parentesi graffe fanno parte delle direttive e sono da intendersi in senso letterale.

Opzione	Descrizione

`options {`
`  `***opzione***`;`
`  …`
`};`
	La direttiva ‘**options**’ serve a definire una serie di opzioni generali. La più comune è ‘**directory**’, con cui si dichiara la directory predefinita a cui fanno riferimento le direttive sulla definizione dei file di zona.

`options {`
`  …`
`  directory `***directory_di_partenza***`;`
`  …`
`};`
	L’opzione ‘**directory**’ definisce la collocazione predefinita dei file di zona, in modo da permetterne successivamente l’indicazione in modo relativo a questa directory.

`options {`
`  …`
`  forwards {`
`  `***indirizzo_numerico***`;`
`  …`
`  };`
`  …`
`};`
	L’opzione ‘**forwarders**’ dichiara che il servizio di risoluzione dei nomi locale può interpellare a sua volta altri servizi, indicati da indirizzi numerici, per le richieste che non dovesse riuscire a risolvere.
	Si osservi che è indispensabile utilizzare questa opzione se il proprio elaboratore è difeso da un firewall che impedisce il transito di pacchetti UDP.

`options {`
`  …`
`  forward only;`
`  …`
`};`
	L’opzione ‘**forward only**’ serve a specificare che si tratta di un servizio di risoluzione dei nomi che rinvia sistematicamente ogni richiesta agli indirizzi indicati nell’opzione ‘**forwarders**’.

`options {`
`  …`
`  listen-on-v6 [port `***n***`] { any; | none;};`
`  …`
`};`
	Consente o esclude l’ascolto per le interrogazioni IPv6 dalla porta indicata. Se la porta non viene indicata, si fa riferimento implicitamente al numero 53.

`zone "`***dominio***`" {`
`  …`
`};`
	La direttiva ‘**zone**’ serve a fare riferimento a una zona; ma ciò può avvenire in modi diversi, descritti nelle sezioni successive.

È importante sottolineare che in questo file non si usa il punto finale per indicare domini assoluti. I domini sono sempre indicati esattamente come sono, senza sottintendere alcunché, pertanto il punto finale sarebbe solo un errore.

33.5.2 Memoria cache del dominio principale

```
zone "." {
        type hint;
        file file_di_zona;
};
```

In questo modo si indica il file contenente le informazioni necessarie a raggiungere i DNS del dominio principale. Il DNS locale conser-

va una memoria cache delle informazioni ottenute, per non dover interrogare ogni volta tutti i DNS esterni necessari.

Senza una direttiva '**zone**' che faccia riferimento al dominio principale, '**named**' non ha modo di accedere ad altri servizi di risoluzione dei nomi al di fuori del suo stretto ambito di competenza.

Si fa a meno della specificazione di questa zona quando si gestisce un servizio di risoluzione dei nomi a uso esclusivo di una rete locale chiusa, senza accesso all'esterno. Si può fare a meno di questa indicazione quando si utilizzano serventi di inoltro, ovvero i *forwarder*.

33.5.3 Gestione delle zone su cui si ha autorità

```
zone "dominio" {
        type master;
        file file_di_zona;
};
```

Quando la direttiva '**zone**' serve a indicare una zona su cui si ha autorità, attraverso l'opzione '**type master**' si stabilisce che le informazioni su questa devono essere tratte dal file indicato.

La zona può essere riferita a un dominio normale, oppure a domini *in-addr.arpa* e *ip6.arpa* (*ip6.int* è superato). Nel primo caso, le informazioni del file servono a tradurre i nomi a dominio in indirizzi numerici; nel secondo, dal momento che i domini *in-addr.arpa* e *ip6.arpa* contengono nel nome l'informazione dell'indirizzo numerico, i file servono a tradurre gli indirizzi numerici in nomi a dominio normali.

Convenzionalmente, è sempre presente una direttiva '**zone**' riferita al dominio *0.0.127.in-addr.arpa* che indica il file in grado di tradurre gli indirizzi di *loopback* per IPv4.[9]

33.5.4 Riproduzione delle informazioni di un altro DNS

```
zone "dominio" {
    type slave;
    file file_di_zona;
    masters {
        indirizzo_ip_master;
        …
    };
};
```

Il DNS locale può servire a fornire informazioni per cui è autorevole assieme ad altri, da cui trae periodicamente le informazioni. In pratica, l'opzione '**type slave**' definisce che il file specificato deve essere generato automaticamente e aggiornato, in base a quanto fornito per quel dominio da altri DNS elencati nell'opzione '**masters**'.

In questi casi è bene che il file di zona sia collocato al di sotto di '`/var/cache/bind/`', proprio per la sua dinamicità. Diversamente, è conveniente che i file di zona sui quali si ha il controllo si trovino a partire dalla directory '`/etc/bind/`'.

Se i servizi di risoluzione dei nomi esterni dovessero risultare inaccessibili per qualche tempo, quello locale può continuare a fornire le informazioni, fino a quando queste raggiungono il periodo di scadenza.

33.5.5 File di zona

I file di zona costituiscono in pratica la base di dati DNS dell'ambito in cui il sistema è autorevole. Sono costituiti da una serie di record di tipo diverso, detti RR (*Resource record*) o record di risorsa, ma con una sintassi comune.

```
[dominio] [durata_vitale] [classe] tipo dati_della_risorsa
```


I campi sono separati da spazi o caratteri di tabulazione; inoltre, un record può essere suddiviso in più righe reali, come si fa solitamente con il tipo SOA.

Ogni file di zona è associato a un dominio di origine definito all'interno del file '`named.conf`' nella direttiva che nomina il file di zona in questione. All'interno dei file di zona, il simbolo '`@`' rappresenta questo dominio di origine. Questo simbolo viene utilizzato comunemente **solo** nel record SOA.

Segue l'elenco dei vari campi dei record di risorsa contenuti nei file di zona.

1. Il primo campo indica il dominio a cui gli altri elementi del record fanno riferimento. Se non viene specificato, si intende che si tratti di quello dichiarato nel record precedente. Il dominio può essere indicato in modo assoluto, quando termina con un punto, o relativo al dominio di origine.
2. Il secondo campo indica il tempo di validità dell'informazione, espressa in secondi. Serve solo per i serventi secondari (*slave*) che hanno la necessità di sapere per quanto tempo deve essere considerata valida un'informazione, prima di eliminarla in mancanza di riscontri dal servente primario (*master*). Generalmente, questa informazione non viene indicata, perché così si utilizza implicitamente quanto indicato nel record SOA, nell'ultimo campo numerico (*minimum*). Questa informazione viene definita TTL (*Time to live*) e non va confusa con altri tipi di TTL esistenti e riferiti a contesti diversi.[10]
3. Il terzo campo rappresenta la classe di indirizzamento. Con le reti TCP/IP si usa la sigla '**IN**' (*Internet*). Se non viene indicata la classe, si intende fare riferimento implicitamente alla stessa classe del record precedente. Generalmente si mette solo nel primo: il record SOA.
4. Il quarto campo rappresenta il tipo di record indicato con le sigle già descritte in sezioni precedenti.
5. Dopo il quarto campo seguono i dati particolari del tipo specifico di record. Questi sono già descritti in parte nel capitolo.

Nei record di risorsa può apparire il simbolo '`@`' che rappresenta il ***dominio di origine***, cioè quello indicato nella direttiva del file '`named.conf`' corrispondente alla zona in questione.

Nelle sezioni seguenti vengono descritti i record di risorsa più importanti.

33.5.6 SOA -- Start of authority

Il primo record di ogni file di zona inizia con la dichiarazione standard dell'origine. Ciò avviene generalmente attraverso il simbolo '`@`' che rappresenta il dominio di origine, come già accennato in precedenza. Per esempio, nel file '`named.conf`', la direttiva seguente fa riferimento al file di zona '`/etc/bind/zone/brot.dg`'.

```
zone "brot.dg" {
        type master;
        file "/etc/bind/zone/brot.dg";
};
```

In tal caso, il simbolo '`@`' del primo record del file '`/etc/bind/zone/brot.dg`' rappresenta precisamente il dominio *brot.dg*.

```
@       IN      SOA   dinkel.brot.dg. root.dinkel.brot.dg. (
                                1998031800
                                28800
                                7200
                                604800
                                86400 )
```

Sarebbe quindi come se fosse stato scritto nel modo seguente:

```
brot.dg.        IN      SOA     …
```

Tutti i nomi a dominio che dovessero essere indicati senza il punto finale sono considerati relativi al dominio di origine. Per esempio, nello stesso record appare il nome '**dinkel.brot.dg.**' che rappresenta un dominio assoluto. Al suo posto sarebbe stato possibile scri-

vere solo '**dinkel**', senza punto finale, perché verrebbe completato correttamente dal dominio di origine.[11]

La sintassi completa del record SOA potrebbe essere espressa nel modo seguente:

```
dominio  classe  SOA  servente_primario  contatto  (
                      numero_seriale
                      refresh
                      retry
                      expire
                      minimum  )
```

Nell'esempio visto, la parola chiave '**IN**' rappresenta la classe di indirizzamento, *Internet*, ed è praticamente obbligatorio il suo utilizzo, almeno nel record SOA.

La parola chiave SOA definisce il tipo di record, *Start of authority*; inoltre deve trattarsi del primo record di un file di zona. Segue la descrizione dei dati specifici di questo tipo di record, precisamente ciò che segue la parola chiave SOA.

- Il ***nome canonico*** dell'elaboratore che svolge la funzione di servente DNS primario per il dominio indicato all'inizio del record. Convenzionalmente, si indica un nome a dominio assoluto.
- L'indirizzo di posta elettronica della persona responsabile per la gestione del servizio. Dal momento che il simbolo '**@**' ha un significato speciale per questi record, lo si sostituisce con un punto. Il nome '**root.dinkel.brot.dg.**' deve essere interpretato come *root@dinkel.brot.dg*.[12]
- Il numero di serie serve ai serventi DNS secondari per sapere quando i dati sono stati modificati. Il numero **deve** essere progressivo. È consentito l'uso di 10 cifre numeriche, pertanto, generalmente si indica la data (in formato ***aaaammgg***) seguita da due cifre aggiuntive. Ogni volta che si modifica il file di zona, questo numero deve essere incrementato; utilizzando la data come in questo esempio si hanno a disposizione le ultime due cifre per indicare diverse versioni riferite allo stesso giorno.
- Il numero definito come *refresh* rappresenta l'intervallo in secondi tra una verifica e la successiva da parte di un servente DNS secondario per determinare se i dati sono stati modificati. Come già specificato, questa verifica si basa sul confronto del numero di serie: se è aumentato, il servente DNS deve rileggere i dati di questo file.
- Il numero definito come *retry* rappresenta l'intervallo in secondi tra una tentativo fallito di accedere al servente DNS e il successivo. In pratica, quando il servente DNS primario è inattivo, i serventi secondari continuano a funzionare e fornire il loro servizio, tuttavia, a intervalli regolari tentano di contattare il servente primario. Questo intervallo è generalmente più corto del tempo di *refresh*, ma non troppo breve, per non sovraccaricare inutilmente la rete con richieste eccessive.
- Il numero definito come *expire* rappresenta la durata massima di validità dei dati quando il servente DNS secondario non riesce più a raggiungere quello primario. In situazioni normali può trattarsi di un valore molto grande, per esempio un mese, anche se negli esempi mostrati è stato usato un valore molto inferiore.
- Il numero definito come *minimum* rappresenta il tempo predefinito di validità per gli altri record di risorsa. Anche questo valore, se ciò è conveniente, può essere piuttosto grande.

33.5.7 NS -- Name Server

Il secondo record è generalmente quello che indica il nome del nodo che offre il servizio di risoluzione dei nomi, ovvero il servente DNS, come nell'esempio seguente:

```
                        NS    dinkel.brot.dg.
```


La parola chiave '**NS**' sta appunto a indicare di che record si tratta. In un file di zona possono apparire più record NS, quando si vuole demandare parte della risoluzione di quella zona ad altri serventi DNS, oppure quando si vogliono semplicemente affiancare.

Questo record viene usato generalmente senza l'indicazione esplicita del dominio e della classe, dal momento che può fare riferimento a quelli già dichiarati nel record SOA. Sotto questo punto di vista, l'esempio appena mostrato corrisponde alla trasformazione seguente:

```
@               IN      NS    dinkel.brot.dg.
```

Il nome del servente DNS dovrebbe essere un nome canonico, cioè un nome per il quale esiste un record di tipo '**A**' corrispondente.

33.5.8 MX -- Mail Exchanger

Nei file di zona utilizzati per tradurre i nomi a dominio in indirizzi numerici, dopo l'indicazione dei record NS, si possono trovare uno o più record che rappresentano i servizi per lo scambio della posta elettronica (serventi SMTP). La sintassi precisa è la seguente:

```
dominio  classe  MX  precedenza  nodo
```

Si osservi l'esempio seguente:

```
                        MX      10      dinkel.brot.dg.
                        MX      20      roggen.brot.dg.
```

Qui appaiono due record di questo tipo. La parola chiave MX indica il tipo di record; il numero che segue rappresenta il livello di precedenza; il nome finale rappresenta il nodo che offre il servizio di scambio di posta elettronica. Nell'esempio, si vuole fare in modo che il primo servizio a essere interpellato sia quello dell'elaboratore *dinkel.brot.dg* e se questo non risponde si presenta l'alternativa data da *roggen.brot.dg*.

Anche qui sono state omesse le indicazioni del dominio e della classe di indirizzamento, in modo da utilizzare implicitamente quelle della dichiarazione precedente. Anche in questo caso, l'intenzione è quella di fare riferimento al dominio di origine e alla classe '**IN**'.

```
@               IN      MX      10      dinkel.brot.dg.
@               IN      MX      20      roggen.brot.dg.
```

33.5.9 A, AAAA, A6 -- Address

I file di zona utilizzati per tradurre i nomi a dominio in indirizzi numerici sono fatti essenzialmente per contenere record di tipo A, AAAA e A6, ovvero record di indirizzo, che permettono di definire le corrispondenze tra nomi e indirizzi numerici.

```
dinkel.brot.dg.     A       192.168.1.1
dinkel.brot.dg.     A6    0 fec0:0:0:1:0:0:0:1
roggen.brot.dg.     A       192.168.1.2
roggen.brot.dg.     A6    0 fec0:0:0:1:0:0:0:2
```

Nell'esempio si mostrano quattro di questi record. Il primo, in particolare, indica che il nome *dinkel.brot.dg* corrisponde all'indirizzo numerico 192.168.1.1, IPv4, mentre il secondo indica che lo stesso nome corrisponde all'indirizzo fec0:0:0:1:0:0:0:1 per IPv6.

Da questo si comprende che i record A riguardano indirizzi IPv4, mentre i record A6 riguardano indirizzi IPv6. I record AAAA sono superati e servivano anche questi per ottenere gli indirizzi IPv6. L'esempio seguente riguarda l'uso di un record AAAA:

```
dinkel.brot.dg.         AAAA    fec0:0:0:1:0:0:0:1
```

Come già accennato in precedenza, i nomi possono essere indicati in forma abbreviata, relativi al dominio di origine per cui è stato definito il file di zona; in questo caso si tratta di *brot.dg*. Per cui, i quattro record appena mostrati avrebbero potuto essere rappresentati nella forma seguente:

```
dinkel          A         192.168.1.1
dinkel          A6      0 fec0:0:0:1:0:0:0:1
roggen          A         192.168.1.2
roggen          A6      0 fec0:0:0:1:0:0:0:2
```

È possibile attribuire nomi diversi allo stesso indirizzo numerico, come nell'esempio seguente. Non si tratta di alias, ma di nomi diversi che vengono tradotti nello stesso indirizzo reale.

```
dinkel.brot.dg.         A       192.168.1.1
roggen.brot.dg.         A       192.168.1.2

farro.brot.dg.          A       192.168.1.1
segale.brot.dg.         A       192.168.1.2
```

Questo tipo di record prevede anche la possibilità di utilizzare l'indicazione della durata di validità (TTL) e della classe. Come al solito, se la classe non viene utilizzata, si fa riferimento alla classe del record precedente, mentre per la durata di validità vale quanto definito come *minimum* nel record SOA. Dagli esempi già mostrati, i quattro record di questa sezione potrebbero essere scritti nel modo seguente:

```
dinkel.brot.dg.     86400   IN   A       192.168.1.1
dinkel.brot.dg.     86400   IN   A6    0 fec0:0:0:1:0:0:0:1
roggen.brot.dg.     86400   IN   A       192.168.1.2
roggen.brot.dg.     86400   IN   A6    0 fec0:0:0:1:0:0:0:2
```

33.5.10 PTR -- Pointer

Nei file di zona utilizzati per tradurre i nomi a dominio che appartengono a *.arpa* in nomi a dominio normali, cioè quelli che servono a ottenere il nome a partire dall'indirizzo numerico, si utilizzano i record PTR (o record puntatori) con questo scopo.

```
1                       PTR     dinkel.brot.dg.
2                       PTR     roggen.brot.dg.
```

L'esempio dei due record che appaiono sopra si riferisce a indirizzi IPv4, con un significato intuitivo, ma non necessariamente chiaro. Il numero che appare all'inizio è un nome a dominio abbreviato, riferito all'origine *1.168.192.in-addr.arpa*, per cui, volendo indicare nomi a dominio completi, si dovrebbe fare come nell'esempio seguente:

```
1.1.168.192.in-addr.arpa.       PTR     dinkel.brot.dg.
2.1.168.192.in-addr.arpa.       PTR     roggen.brot.dg.
```

Dovrebbe essere più chiaro adesso che i record PTR rappresentano un collegamento tra un nome a dominio e un altro. È comunque solo attraverso questo meccanismo che si può ottenere una traduzione degli indirizzi numerici in nomi a dominio.

È il caso di considerare il fatto che attraverso i record A e A6 possono essere abbinati più nomi a dominio allo stesso indirizzo numerico, ma con i record PTR si può abbinare un indirizzo numerico a un solo nome a dominio. Cioè a dire che quando si chiede il nome corrispondente a un indirizzo numerico se ne ottiene uno solo. Anche per questo, è necessario che il nome a dominio indicato corrisponda a un nome canonico.

Con indirizzi IPv6 si usa una notazione particolare:

```
\[x0000000000000001/64]         PTR     dinkel.brot.dg.
\[x0000000000000002/64]         PTR     roggen.brot.dg.
```

Qui la stringa '**`\[x0000000000000001/64]`**' fa riferimento esplicito a un numero esadecimale, 0000000000000001_{16}, in cui vanno presi in considerazione gli ultimi 64 bit. Questa stringa va attaccata alla stringa corrispondente che rappresenta il dominio di origine, come indicato nel file '`named.conf`':

```
zone "\[xfec0000000000001/64].ip6.arpa" {
        type master;
        file "/etc/bind/zone/fec0:0:0:1";
};
```

Pertanto, si intende fare riferimento all'indirizzo $fec0000000000001 0000000000000001_{16}$, ovvero fec0:0000:0000:0001:0000:0000:0000:0001, ovvero fec0:0:0:1:0:0:0:1.

In passato è esistito anche un altro modo per rappresentare un indirizzo IPv6, attraverso il dominio superato *ip6.int*. Anche se si tratta di un sistema superato, vale la pena di annotare il meccanismo. Nel file '`named.conf`' si indicava il dominio come:


```
zone "1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.c.e.f.IP6.INT" {
        type master;
        file "fec0:0:0:1";
};
```

Come si intuisce, si tratta di un dominio ottenuto da tutte le cifre esadecimali che compongono la prima parte dell'indirizzo. Nel file di zona, si continuava il dominio:

```
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0         PTR     dinkel.brot.dg.
2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0         PTR     roggen.brot.dg.
```

oppure lo si scriveva per esteso, come già si può fare per *in-addr.arpa*:

```
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.c.e.f.IP6.INT ↩
↪PTR   dinkel.brot.dg.
2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.c.e.f.IP6.INT ↩
↪PTR   roggen.brot.dg.
```

Nella documentazione originale, questa notazione è nota con il termine *nibble* (usato come aggettivo), perché questo è il nome che un tempo veniva dato ai gruppetti di 4 bit (mezzo byte), dal momento che i domini *ip6.int* si scompongono seguendo le cifre esadecimali, ognuna delle quali occupa 4 bit.

Naturalmente, anche per il record PTR valgono le considerazioni fatte per il tipo A e A6, riguardo all'indicazione della durata di validità e alla classe di indirizzamento.

33.5.11 CNAME -- Canonical Name

Nei file di zona utilizzati per tradurre i nomi a dominio in indirizzi numerici, possono apparire dei record CNAME che permettono di definire degli alias a nomi a dominio già definiti (i nomi canonici).

```
www.dinkel.brot.dg.     CNAME   dinkel.brot.dg.
ftp.dinkel.brot.dg.     CNAME   dinkel.brot.dg.
```

L'esempio dei due record appena mostrati, indica che i nomi *www.dinkel.brot.dg* e *ftp.dinkel.brot.dg* sono alias del nome canonico *dinkel.brot.dg*.

Teoricamente si può fare la stessa cosa utilizzando record di tipo A e di tipo A6 con la differenza che i nomi vanno abbinati a un indirizzo numerico. L'utilità del record CNAME sta nella facilità con cui possono essere cambiati gli indirizzi: in questo caso, basta modificare l'indirizzo numerico di *dinkel.brot.dg* e gli alias non hanno bisogno di altre modifiche.

Tuttavia, l'uso di alias definiti attraverso record CNAME è altamente sconsigliabile nella maggior parte delle situazioni. Questo significa che nei record SOA, NS, MX e CNAME, è meglio indicare sempre solo nomi a dominio per cui esiste la definizione di corrispondenza attraverso un record A o A6. In pratica, i record CNAME andrebbero usati solo per mostrare all'esterno nomi alternativi esteticamente più adatti alle varie circostanze, come nell'esempio mostrato in cui si aggiunge il prefisso '**`www`**' e '**`ftp`**'.

In particolare, nel record SOA è assolutamente vietato utilizzare nomi definiti come alias.

33.5.12 File dei serventi principali

Nelle sezioni precedenti sono stati descritti i vari record di risorsa e il loro utilizzo nei file di zona. Il file utilizzato per elencare i serventi DNS principali contiene esclusivamente due tipi di record: NS e A.

I record NS servono a indicare i nomi dei vari serventi DNS competenti per il dominio principale; i record A forniscono la traduzione di questi nomi in indirizzi numerici. Ciò è esattamente quanto serve in questo tipo di file.

```
.                       3600000  IN  NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.     3600000      A     198.41.0.4
```

33.6 Serventi DNS secondari

Un servente DNS secondario, o *slave*, è quello che riproduce le informazioni di altri serventi, controllando la validità a intervalli regolari, aggiornando i dati quando necessario.

Supponendo di volere realizzare un servente DNS secondario nell'elaboratore *roggen.brot.dg*, per seguire gli esempi già mostrati, si può semplicemente definire il file '`named.conf`' come nell'esempio seguente:

```
options {
        directory "/var/cache/bind";
};
//
zone "." {
        type hint;
        file "/etc/bind/named.root";
};
//
zone "0.0.127.in-addr.arpa" {
        type master;
        file "/etc/bind/zone/127.0.0";
};
//
zone "1.168.192.in-addr.arpa" {
        type slave;
        file "zone/192.168.1";
        masters {
                192.168.1.1;
        };
};
zone "\[xfec0000000000001/64].ip6.arpa" {
        type slave;
        file "zone/fec0:0:0:1";
        masters {
                192.168.1.1;
        };
};
zone "dg" {
        type slave;
        file "zone/dg";
        masters {
                192.168.1.1;
        };
};
zone "brot.dg" {
        type slave;
        file "zone/brot.dg";
        masters {
                192.168.1.1;
        };
};
```

I file '`/etc/bind/named.root`' e '`/etc/bind/zone/127.0.0`' sono i soliti già visti per il caso del servente primario. In questo modo, il servente DNS secondario è in grado di risolvere da solo le richieste al di fuori delle zone di competenza.

Le direttive di dichiarazione di zona che contengono l'opzione '**`type slave`**' servono a fare in modo che il DNS locale risponda alle richieste riferite a queste, anche se poi a sua volta deve aggiornare i file relativi in base a quanto ottenuto dai DNS indicati nell'opzione '**`masters`**'.

Si osservi che in questo caso, le zone copiate dal DNS primario sono inserite in file collocati al di sotto di '`/var/cache/bind/`', dal momento che sono stati usati percorsi relativi. Per esempio, il file '`/var/cache/bind/zone/192.168.1`' serve a contenere la zona relativa agli indirizzi 192.168.1.*.

33.7 Servente DNS di inoltro

Un servente DNS di inoltro, o *forwarder*, è quello che rinvia le richieste a un altro servizio di risoluzione dei nomi.

Il DNS utilizza una serie di protocolli, tra cui anche UDP. Se ci si trova a essere protetti da un firewall che esclude il transito dei pacchetti UDP, per poter interpellare gli altri servizi di risoluzione dei nomi delle zone che sono al di fuori della propria competenza locale, occorre rinviare le richieste a un servizio esterno. Questa situazione può verificarsi quando la propria connessione a Internet avviene attraverso un ISP attento ai problemi di sicurezza e che usa questa politica di protezione.

Supponendo di volere realizzare un servente DNS di inoltro nell'elaboratore *roggen.brot.dg*, per seguire gli esempi già mostrati, si può semplicemente definire il file '`named.conf`' come nell'esempio seguente:

```
options {
        directory "/var/cache/bind";
        forward only;
        forwarders {
                192.168.1.1;
        };
};
//
zone "0.0.127.in-addr.arpa" {
        type master;
        file "/etc/bind/zone/127.0.0";
};
```

Si può osservare l'assenza della dichiarazione della zona del dominio principale. Solo il dominio *0.0.127.in-addr.arpa* viene risolto localmente, tutto il resto viene richiesto al DNS corrispondente all'indirizzo 192.168.1.1. L'opzione '**`forward only`**' sottolinea questo fatto.

33.8 Esercitazione: individuazione dei nomi a dominio disponibili e occupati

Con l'ausilio del programma '**`whois`**', si cerchino le informazioni utili a contattare chi ha registrato dei nomi a dominio che potrebbero essere di proprio interesse. I nomi a dominio in questione devono essere di secondo livello (del tipo *tizio.it*). Il nome a dominio da cercare può essere scelto liberamente, in base a un proprio interesse ragionevole, oppure può essere costituito dal proprio cognome o dal proprio nome. La ricerca va fatta sui domini di primo livello per i quali è possibile eseguire la registrazione, come nell'esempio seguente:

Dominio di secondo livello	Ente di registrazione (*registrar*)	Organizzazione o persona per la quale è fatta la registrazione (*registrant*)	Scadenza della registrazione	Utilizzo del nome a dominio
tizio.it	IT-INC	Primo Tizio srl	17 ottobre 2012	No
tizio.com	REGISTER.COM NETWORK SOLUTIONS	Tizio Tizi spa	3 maggio 2013	No
tizio.net		Caio Cai	22 gennaio 2013	Sì
tizio.org	Register-it	Mevio Mary	22 novembre 2012	No
tizio.info	Register-it	Sempronio Sesto	25 ottobre 2012	No
tizio.name	--	--	--	--
tizio.ws	--	--	--	--
tizio.biz	--	--	--	--
tizio.tv	--	--	--	--
tizio.cc	--	--	--	--
tizio.tk	--	--	--	--

Per scoprire se un dominio registrato è utilizzato, si può usare un navigatore per provare se esiste effettivamente un sito con quel nome, magari con l'aggiunta del prefisso '**`www`**' (come per esempio potrebbe

essere *www.tizio.ws*).

33.9 Riferimenti

- *Bind 9 administrator reference manual*, 2001, Internet Software Consortium, *http://unbound.sourceforge.net/manual/Bv9ARM.html*
- K. Harrenstien, M. Stahl, E. Feinler, *RFC 954: NICNAME/WHOIS*, 1985, *http://www.ietf.org/rfc/rfc954.txt*
- Autorità di registrazione italiana, *it-nic*, *http://www.nic.it/*
- *Dynamic DNS (DDNS) Providers*, *http://dnslookup.me/dynamic-dns/*
- DMOZ, *Dynamic DNS*, *http://www.dmoz.org/Computers/Internet/Protocols/DNS/DNS_Providers/Dynamic_DNS/*
- Olaf Kirch, *NAG, The Linux Network Administrators' Guide*
- *Internet Systems Consortium*, *http://www.isc.org/*
- *named(8)*
- *Bind 9 administrator reference manual*, 2001, Internet Software Consortium, *http://unbound.sourceforge.net/manual/Bv9ARM.html*

[1] Nel file '`/etc/hosts`' è possibile indicare sia gli indirizzi IPv4, sia quelli IPv6 (site-local), mantenendo gli stessi nomi a dominio. In pratica, quello che si vede nell'esempio funziona.

[2] **Whois** GNU GPL

[3] **BIND** software libero con licenza speciale e restrizioni per quanto riguarda l'algoritmo RSA

[4] I nomi a dominio sono completi (FQDN) perché sono indicati con un punto finale.

[5] **Nslookup** software libero con licenza speciale

[6] **Host** software libero con licenza speciale

[7] **Dig** software libero con licenza speciale

[8] L'esempio proposto riguarda la situazione di un certo momento. Se si tenta di ripetere l'esempio, è probabile che il risultato sia differente, soprattutto per ciò che riguarda i numeri IP attribuiti ai vari nodi che si incontrano.

[9] Eventualmente, potrebbe essere conveniente anche la presenza di una direttiva '**zone**' riferita al dominio *\[x00000000000000000000000000000001/128].ip6.arpa*, per la traduzione dell'indirizzo ::1 IPv6.

[10] Per esempio, si parla di TTL anche a proposito di pacchetti IP, ma in quel caso si intende indicare il numero massimo di salti (attraverso i router) che questi possono fare.

[11] Tuttavia, in un record SOA è preferibile indicare solo nomi a dominio assoluti.

[12] Di conseguenza, indirizzi di posta elettronica del tipo *mario.rossi@brot.dg* non si possono usare, perché contengono il punto prima della chiocciola.