## Domain Name Service

TCP/IP usa un indirizzo a 32 bit per instradare un datagramma verso una destinazione. È utile dimenticare questi indirizzi a 32 bit e usare invece nomi comuni.

Vi sono diversi metodi usati per questo. Il più comune è stato già esaminato e consiste nell'impiegare un file ASCII sulla macchina mittente che ha i nomi e i corrispondenti indirizzi (/etc/hosts su Unix). La principale limitazione di questo sistema è che la macchina può instradare solo verso le altre macchine a cui corrisponde una voce in questo file, che è impossibile da gestire quando vi sono molte macchine target o si vuole accedere a tutti i dispositivi nella rete.

Un altro approccio è quello di affidare la risoluzione degli indirizzi ad un altro processo che agisce come un servizio di directory. Ci sono due di tali schemi in uso oggi: Domain Name Service (DNS) e Network Information Service , che è ora parte di NFS. Guardiamo ora il DNS.

### *Domain Name Service (DNS)*

Un nome simbolico è una stringa di caratteri usata per identificare una macchina .

Quando si inviano informazioni ad una macchina remota , si devono usare indirizzi IP o Internet. Invece di richiedere che l'utente memorizzi i numeri della macchina remota è cosa comune usare un nome simbolico.

La conversione fra indirizzo IP e nome simbolico è usualmente realizzata nella macchina mittente usando un file come /etc/hosts per Unix. Questo tipo di approccio lavora bene su piccole reti , dove è coinvolto un piccolo numero di destinatari. Quando si ha a che fare con l'intera Internet , comunque, è irragionevole aspettarsi che un unico file ASCII contenga tutti i possibili nomi simbolici e i loro indirizzi.

La sola grandezza del file non è il solo problema. Ampie reti tendono a cambiare costantemente . centinaia di aggiunte o modifiche devono essere effettuate ogni giorno. Il tempo impiegato per aggiornare ogni macchina ( o anche soltanto gateway socializzati sulla rete) sarebbe enorme.

La soluzione del problema è di offrire un metodo per allontanare la gestione delle tabelle di lookup dal Network Information center (NIC) , che governa Internet , e verso i partecipanti e le loro reti autonome in modo che il carico di lavoro sulla rete sia piccolo ma la flessibilità non sia compromessa. Questo è quello che fa il DNS.

Unix implementa DNS attraverso un daemon chiamato named[1] che gira su un server di nome[2] , una macchina che gestisce la risoluzione dei nomi simbolici[3] usando metodi DNS. Parte del sistema è una libreria di funzioni che possono esser usate in applicazioni per realizzare query sul name server. Questa routine di query è chiamata revolver[4] o name revolver[5] e può risiedere su un'altra macchina.

---

[1] named

[2] name server

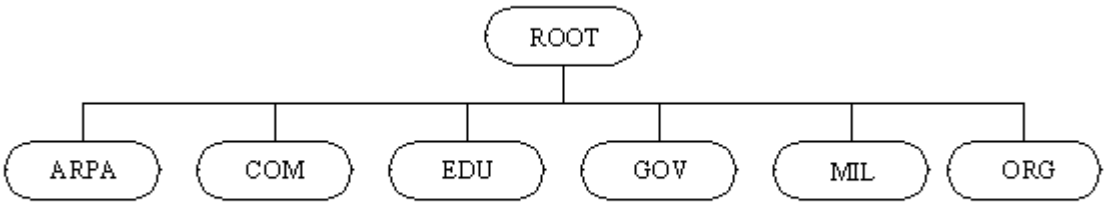
[3] symbolic names

[4] resolver

[5] name resolver

### Struttura DNS

Il DNS lavora suddividendo Internet in un set di domini[6] che possono ulteriormente divisi in sottodomini[7]. Questa struttura assomiglia ad un albero come nella figura seguente




il primo set di domini si chiama top-level domains[8]. Ci sono sei domini top-level in uso regolare:

- ARPA[9]: per organizzazioni specifiche di Internet
- COM[10]: per imprese commerciali
- EDU[11]: per organizzazioni educative
- GOV[12]: per organismi governativi
- MIL[13]: per organizzazioni militari

---

[6] domain domains

[7] subdomains subdomain

[8] top level domains

[9] ARPA

[10] COM

[11] EDU

[12] GOV

- ORG[14]: per organizzazioni non commerciali

In aggiunta per questi domini top-level vi sono domini top-level dedicati per ogni paese connesso. Essi sono usualmente identificati con una forma corta del nome del paese come .ca[15] per il Canada e .uk[16] per Regno Unito, .it[17] per Italia. Questi domini top-level legati al paese[18] sono usualmente lasciati fuori dal diagramma della struttura Internet per questioni di praticità. Ecco un elenco completo

| | |
|---|---|
| **.aero** | Aviation |
| **.biz** | Business Organizations |
| **.com** | Commercial |
| **.coop** | Co-Operative Organizations |
| **.edu** | Educational |
| **.gov** | US Government |
| **.info** | Open TLD |

---

[13] MIL

[14] ORG

[15] .ca

[16] .uk

[17] .it

[18] country top-level domains

| | |
|---|---|
| **.int** | International Organizations |
| **.mil** | US Dept of Defense |
| **.museum** | Museums |
| **.name** | Personal |
| **.net** | Networks |
| **.org** | Organizations |
| **.ac** | Ascension Island |
| **.ad** | Andorra |
| **.ae** | United Arab Emirates |
| **.af** | Afghanistan |
| **.ag** | Antigua and Barbuda |
| **.ai** | Anguilla |
| **.al** | Albania |
| **.am** | Armenia |
| **.an** | Netherlands Antilles |

| | |
|---|---|
| **.ao** | Angola |
| **.aq** | Antarctica |
| **.ar** | Argentina |
| **.as** | American Samoa |
| **.at** | Austria |
| **.au** | Australia |
| **.aw** | Aruba |
| **.az** | Azerbaijan |
| **ba** | Bosnia and Herzegovina |
| **.bb** | Barbados |
| **.bd** | Bangladesh |
| **.be** | Belgium |
| **.bf** | Burkina Faso |
| **.bg** | Bulgaria |
| **.bh** | Bahrain |
| **.bi** | Burundi |
| **.bj** | Benin |
| **.bm** | Bermuda |

| | |
|---|---|
| **.bn** | Brunei Darussalam |
| **.bo** | Bolivia |
| **.br**[19] | Brazil |
| **.bs** | Bahamas |
| **.bt** | Bhutan |
| **.bv** | Bouvet Island |
| **.bw** | Botswana |
| **.by** | Belarus |
| **.bz** | Belize |
| **.ca** | Canada |
| **.cc** | Cocos (Keeling) Islands |
| **.cd** | Congo, Democratic republic of the (former Zaire) |
| **.cf** | Central African Republic |
| **.cg** | Congo, Republic of |
| **.ch** | Switzerland |
| **.ci** | Côte d'Ivoire |

[19] .br

| | |
|---|---|
| **.ck** | Cook Islands |
| **.cl** | Chile |
| **.cm** | Cameroon |
| **.cn** | China |
| **.co** | Colombia |
| **.cr** | Costa Rica |
| **.cs** | Czechoslovakia (former - non-existing) |
| **.cu** | Cuba |
| **.cv** | Cape Verde |
| **.cx** | Christmas Island |
| **.cy** | Cyprus |
| **.cz** | Czech Republic |
| **.de**[20] | Germany |
| **.dj** | Djibouti |
| **.dk** | Denmark |
| **.dm** | Dominica |
| **.do** | nican Republic |
| **.dz** | Algeria |

[20] .de

| | |
|---|---|
| **.ec** | Ecuador |
| **.ee** | Estonia |
| **.eg** | Egypt |
| **.eh** | Western Sahara |
| **.er** | Eritrea |
| **.es** | Spain |
| **.et** | Ethiopia |
| **.eu**[21] | European Union |
| **.ec** | Ecuador |
| **.ee** | Estonia |
| **.eg** | Egypt |
| **.eh** | Western Sahara |
| **.er** | Eritrea |
| **.es** | Spain |
| **.et** | Ethiopia |
| **.eu**[22] | European |

[21] .eu

| | |
|---|---|
| | Union |
| **.fi** | Finland |
| **.fj** | Fiji |
| **.fk** | Falkland |
| | Islands |
| **.fm** | Micronesia |
| **.fo** | Faroe |
| | Islands |
| **.fr**[23] | France |
| **.ga** | Gabon |
| **.gb**[24] | United |
| | Kingdom |
| **.gd** | Grenada |
| **.ge** | Georgia |
| **.gf** | French |
| | Guiana |
| **.gg** | Guernsey |
| **.gh** | Ghana |

---

[22] .eu

[23] .fr

[24] .gb

| | |
|---|---|
| **.gi** | Gibraltar |
| **.gl** | Greenland |
| **.gm** | Gambia |
| **.gn** | Guinea |
| **.gp** | Guadeloupe |
| **.gq** | Equatorial Guinea |
| **.gr** | Greece |
| **.gs** | South Georgia and the South Sandwich Islands |
| **.gt** | Guatemala |
| **.gu** | Guam |
| **.gw** | Guinea-Bissau |
| **.gy** | Guyana |
| **.hk** | Hong Kong |
| **.hm** | Heard and McDonald |

| | |
|---|---|
| | Islands |
| **.hn** | Honduras |
| **.hr** | Croatia |
| **.ht** | Haiti |
| **.hu** | Hungary |
| **.id** | Indonesia |
| **.ie** | Ireland |
| **.il** | Israel |
| **.im** | Isle of Man |
| **.in** | India |
| **.io** | British Indian |
| | Ocean |
| | Territory |
| **.iq** | Iraq |
| **.ir** | Iran |
| **.is** | Iceland |
| **.it**[25] | Italy |
| **.je** | Jersey |
| **.jm** | Jamaica |

[25] .it

| | |
|---|---|
| **.jo** | Jordan |
| **.jp**[26] | Japan |
| **.ke** | Kenya |
| **.kg** | Kyrgyzstan |
| **.kh** | Cambodia |
| **.ki** | Kiribati |
| **.km** | Comoros |
| **.kn** | Saint Kitts and Nevis |
| **.kp** | Korea, Democratic People's Republic of |
| **.kr** | Korea, Republic of |
| **.kw** | Kuwait |
| **.ky** | Cayman Islands |
| **.kz** | Kazakhstan |

[26] .jp

| | |
|---|---|
| **.la** | Lao People's Democratic Republic |
| **.lb** | Lebanon |
| **.lc** | Saint Lucia |
| **.li** | Liechtenstein |
| **.lk** | Sri Lanka |
| **.lr** | Liberia |
| **.ls** | Lesotho |
| **.lt** | Lithuania |
| **.lu** | Luxembourg |
| **.lv** | Latvia |
| **.ly** | Libyan Arab Jamahiriya |
| **ma** | Morocco |
| **.mc** | Monaco |
| **.md** | Moldova |
| **.mg** | Madagascar |
| **.mh** | Marshall Islands |
| **.mk** | Macedonia |

| | |
|---|---|
| **.ml** | Mali |
| **.mm** | Myanmar |
| **.mn** | Mongolia |
| **.mo** | Macau |
| **.mp** | Northern Mariana Islands |
| **.mq** | Martinique |
| **.mr** | Mauritania |
| **.ms** | Montserrat |
| **.mt** | Malta |
| **.mu** | Mauritius |
| **.mv** | Maldives |
| **.mw** | Malawi |
| **.mx**[27] | Mexico |
| **.my** | Malaysia |
| **.mz** | Mozambique |
| **.na** | Namibia |
| **.nc** | New |

[27] .mx

| | |
|---|---|
| | Caledonia |
| **.ne** | Niger |
| **.nf** | Norfolk |
| | Island |
| **.ng** | Nigeria |
| **.ni** | Nicaragua |
| **.nl** | The |
| | Netherlands |
| **.no** | Norway |
| **.np** | Nepal |
| **.nr** | Nauru |
| **.nu** | Niue |
| **.nz** | New Zealand |
| **.om** | Oman |
| **.pa** | Panama |
| **.pe** | Peru |
| **.pf** | French |
| | Polynesia |
| **.pg** | Papua New |
| | Guinea |
| **.ph** | Philippines |

| | |
|---|---|
| **.pk** | Pakistan |
| **.pl** | Poland |
| **.pm** | St. Pierre and Miquelon |
| **.pn** | Pitcairn |
| **.pr** | Puerto Rico |
| **.ps** | Palestine |
| **.pt** | Portugal |
| **.pw** | Palau |
| **.py** | Paraguay |
| **.qa** | Qatar |
| **.re** | Reunion |
| **.ro** | Romania |
| **.ru**[28] | Russia |
| **.rw** | Rwanda |
| **.sa**[29] | Saudi Arabia |
| **.sb** | Solomon |

[28] .ru

[29] .sa

| | |
|---|---|
| | Islands |
| **.sc** | Seychelles |
| **.sd** | Sudan |
| **.se** | Sweden |
| **.sg** | Singapore |
| **.sh** | St. Helena |
| **.si** | Slovenia |
| **.sj** | Svalbard and Jan Mayen Islands |
| **.sk** | Slovakia |
| **.sl** | Sierra Leone |
| **.sm** | San Marino |
| **.sn** | Senegal |
| **.so** | Somalia |
| **.sr** | Surinam |
| **.st** | Sao Tome and Principe |
| **.su** | USSR (former) |
| **.sv** | El Salvador |

**.sy** Syrian Arab Republic

**.sz** Swaziland

**.tc** The Turks & Caicos Islands

**.td** Chad

**.tf** French Southern Territories

**.tg** Togo

**.th** Thailand

**.tj** Tajikistan

**.tk** Tokelau

**.tm** Turkmenistan

**.tn** Tunisia

**.to** Tonga

**.tp** East Timor

**.tr** Turkey

**.tt** Trinidad and Tobago

**.tv** Tuvalu

| | |
|---|---|
| **.tw** | Taiwan |
| **.tz** | Tanzania |
| **ua** | Ukraine |
| **.ug** | Uganda |
| **.uk** | United Kingdom |
| **.um** | United States Minor Outlying Islands |
| **.us**[30] | United States |
| **.uy** | Uruguay |
| **.uz** | Uzbekistan |
| **.va**[31] | Holy See (Vatican City State) |

[30] .us

[31] va

| | |
|---|---|
| **.vc** | Saint Vincent and the Grenadines |
| **.ve** | Venezuela |
| **.vg** | Virgin Islands British |
| **.vi** | Virgin Islands U.S |
| **.vn** | Vietnam |
| **.vu** | Vanuatu |
| **.wf** | Wallis and Futuna Islands |
| **.ws** | Samoa |

La suddivisione del dominio è talvolta ripetuta all'interno del dominio del paese, così vi può essere un'estensione .com accoppiata con .ca per indicare un dominio

commerciale canadese o un .edu con .uk per indicare un'organizzazione educativa britannica.

Sotto i domini top-level vi è un altro livello per le organizzazioni individuali all'interno di un dominio di top level. I nomi di dominio sono tutti registrati con il NIC e sono univoci per la rete.

Ci sono due modi per denominare un target . se il target è nell'internetwork , viene usato il nome assoluto[32]. Il nome assoluto è unico e non ambiguo , e specifica il domino del target. Un nome relativo può essere usato sia dentro il dominio locale , dove il name server sa che il target è dentro il dominio e quindi non ha bisogno di instradare il datagramma sull'internet, o quando il nome relativo è conosciuto dal name server e può essere espanso e instradato correttamente.

### Il name server

Ogni Name Sever DNS gestisce una differente area di una rete (o un intero dominio se la rete è piccola). Il set di macchine gestite dal name server è chiamato zone. Diverse zone possono essere gestite dallo stesso name server . praticamente ogni zona ha un name server secondario o di backup, con i due scevre che contengono informazioni duplicate. I name server dentro una zona comunicano usando un zone transfer protocol[33].

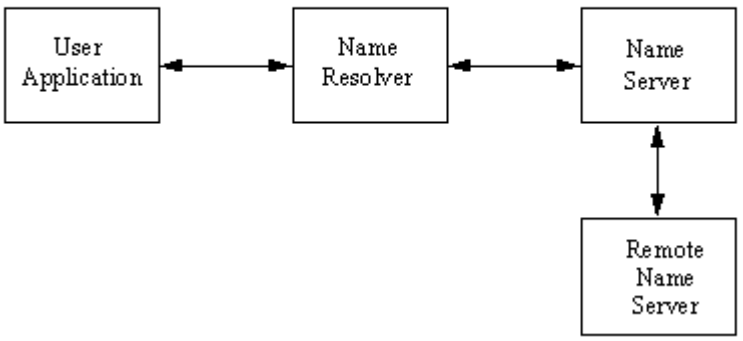
---

[32] absolute name

[33] zone transfer protocol

I DNS operano avendo un set di zone innestate. Ogni name server comunica con quello sopra di lui (e, se c'è, con quello sotto). Ogni zona ha almeno un name server responsabile di conoscere le informazioni di indirizzo per ogni macchina in quella zona. Ogni name server inoltre conosce l'indirizzo di almeno un altro name server. Messaggi fra name server usualmente usano UDP perché il suo metodo non orientato alla connessione da migliore prestazioni. Comunque TCP viene usato per gli aggiornamenti dei database a causa della sua affidabilità.

Quando un'applicazione di utente ha bisogno di risolvere un nome simbolico in un indirizzo di rete, viene inviata una query dall'applicazione al processo resolver, che poi passa la query al name server. Il name server controlla le sue tabelle e restituisce l'indirizzo di rete corrispondente al nome simbolico. Se il name server non ha l'informazione richiesta , può inviare una richiesta ad un altro name server. Il processo è mostrato nella figura seguente




Quando un name server riceve una query da un resolver, ci sono diversi tipi di operazioni che il name server può effettuare. Le operazioni ricadono in due categorie

: non ricorsive[34] e ricorsive[35]. Un'operazione ricorsiva è una in cui il name server accede ad un altro name server per ottenere informazioni.

Operazioni non ricorsive effettuate dal name server includono una riposta completa alla richiesta del resolver , un riferimento ad un altro name server (a cui il resolver deve mandare una query), o un messaggio di errore. Quando è necessaria un'operazione ricorsiva il name server contatta un altro name server con la richiesta del resolver. Il name server remoto replica alla richiesta o con un indirizzo di rete o con un messaggio negativo , indicante fallimento. Le regole DNS proibiscono ad un name server remoto di inviare un riferimento ancora ad un altro server.

### Resource records[36]

L'infomazione richiesta per risolvere i nomi simbolici è gestita dal name server in un set di resource records, che sono voci[37] in un database. Resource Records (spesso abbreviato in RR) contiene informazioni in formato ASCII. Il formato dei record di risorse è mostrato nella figura seguente

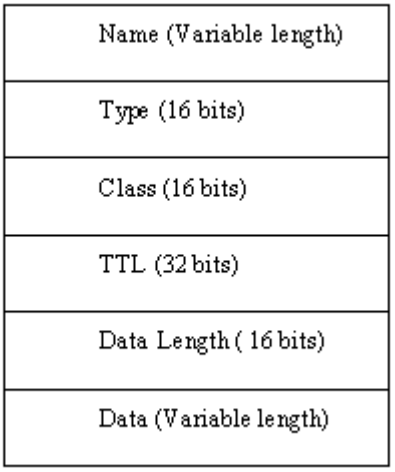
---

[34] nonrecursive

[35] recursive

[36] resource records resource records

[37] entries entry




IL campo Name è il nome del dominio della macchina cui si riferisce il record . Se non è specificato alcun nome viene sostituito il nome precedentemente usato.

Il campo Type identifica il tipo di record id risorse. I Resource records sono utilizzati per diversi scopi , come la mappatura di mi ad indirizzi o la definizione delle zone. Il record Type of resources è identificato da un codice mnemonico o da un numero. Questi codici e il loro significato sono mostrati nella tabella seguente.

| Number | Code | Description |
|---|---|---|
| **1** | A | Network address |
| **2** | NS | Authoritative name server |
| **3** | MD | Mail destination; now replaced by MX |
| **4** | MF | Mail forwarder; now replaced by MX |
| **5** | CNAME | Canonical alias name |
| **6** | SOA | Start of zone authority |
| **7** | MB | Mailbox domain name |
| **8** | MG | Mailbox member |
| **9** | MR | Mail rename domain |
| **10** | NULL | Null resource record |

| | | |
|---|---|---|
| ***11*** | WKS | Well-known service |
| ***12*** | PTR | Pointer to a domain name |
| ***13*** | HINFO | Host information |
| ***14*** | MINFO | Mailbox[38] information |
| ***15*** | MX | Mail exchange [39] |
| ***16*** | TXT | Text strings |
| ***17*** | RP | Responsible person |
| ***18*** | AFSDB | AFS-type services |
| ***19*** | X.25 | X.25 address [40] |
| ***20*** | ISDN | ISDN address [41] |
| ***21*** | RT | Route through |

Il campo Classe nel layout del record contiene un valore per la classe del record. Se non vi è specificato alcun valore viene sostituito con il valore dell'ultima classe utilizzata. I server di nome Internet hanno di solito il valore IN.

Il campo Time To Live (TTL) specifica l'ammontare di tempo in secondi durante il quale il record di risorse è valido nella cache. Se è usato il valore 0, il record non dovrebbe essere aggiunto alla cache[42]. Se il campo TTL è omesso viene usato un

[38] Mailbox

[39] Mail exchange

[40] X.25 address

[41] ISDN address ISDN

[42] cache

valore di default. Usualmente questo campo dice al name server per quanto tempo la voce è valida prima che esso chieda di aggiornarla.

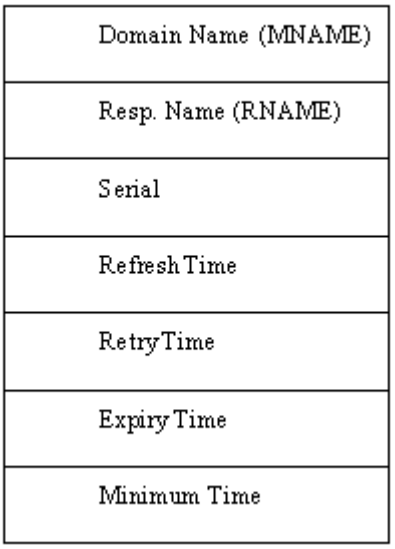
La sezione dati contiene due parti, consistenti nella lunghezza del record e nei dati stessi. Il campo Data Lenght contiene la lunghezza della sezione dati. I dati sono in un campo a lunghezza variabile che descrive la registrazione in qualche modo. L'uso di questo campo differisce con i differenti tipi di record delle risorse.

Alcuni tipi di record delle risorse hanno un singolo pezzo di informazione nell'area dati . come un indirizzo, o al massimo tre pezzi di informazione. La sola eccezione è il record Star of Authority. I contenuti delle aree dati dei record (eccettuato il record Start of Authority) sono dati nella tabella seguente

| *RR Type* | *Fields in Data Area* |
|---|---|
| ***A*** | Address: A network address |
| ***NS*** | NSDNAME: The domain name of host |
| ***MG*** | MGNAME: The domain name of mailbox |
| ***CNAME*** | CNAME: An alias for the machine |
| ***HINFO*** | CPU: A string identifying CPU type |
| | OS: A string identifying operating system |
| ***MINFO*** | RMAILBX: A mailbox responsible for mailing lists |
| | EMAILBOX: A mailbox for error messages |
| ***MB*** | MADNAME: Now obsolete |
| ***MR*** | NEWNAME: Renames the address of a specific mailbox |
| ***MX*** | PREFERENCE: Specifies the precedence for delivery |

| | |
|---|---|
| | EXCHANGE: The domain name of the host that acts as mail exchange |
| ***NULL*** | Anything can be placed in the data field |
| ***PTR*** | PTRDNAME: A domain name that acts as a pointer to a location |
| ***TXT*** | TXTDATA: Any kind of descriptive text |
| ***WKS*** | Address: A network address |
| | Protocol: The protocol used |
| | Bitmap: Used to identify ports and protocols |

Il record Start of Authority [43]è usato per identificare le macchine all'interno di una zona. Vi è un solo record SOA in ogni zona. Il formato del campo dati è mostrato nella figura seguente. I campi del SOA sono usati principalmente per amministrazione e manutenzione del name server.




[43] Start of Authority

Il campo MNAME è il nome di dominio della sorgente di dati per la zona. Il campo RNAME (nome della persona responsabile) è il nome di dominio della mailbox dell'amministratore della zona. Il campo Serial contiene un numero di versione per la zona. Viene cambiato quando viene modificata la zona.

Il refresh time[44] è il numero di secondi tra i refresh dei dati per la zona. Il Retry Time [45]è il numero di secondi di attesa tra richieste di refresh senza successo. Il Expiry Time [46]è il numero di secondi dopo il quale l'informazione di zona non è più valida. Infine, il Minimum Time [47]è il numero di secondi da usare nel campo TTL all'interno della zona.

Alcuni esempi mostrano il semplice formato usato. I record di risorsa Address consistono del nome della macchina , il tipo di indicatore del resource record (A per RR di tipo Address) e l'indirizzo di rete. Un esempio di record di risorsa Address assomiglierebbe a qualcosa del genere:

TPCI_SCO_4 IN A 143.23.25.7

Il tag IN identifica il record come di classe Internet. Questo formato rende facile localizzare un nome derivare il suo indirizzo (l'inverso, derivare il nome

---

[44] Refresh Time

[45] Retry Time Retry

[46] Expiry Time

[47] Minimum Time

dall'indirizzo non è altrettanto facile e richiede un formato speciale chiamato IN-ADDR-ARPA[48] che vedremo in seguito) .

Per i record di servizio di tipo Well-Known Service [49]( o WKS) , il campo dati del record contiene tre campi usati per descrivere i servizi supportati all'indirizzo cui si riferisce il record. Un esempio di WKS è

```
TPCI_SCO.TPCI.COM    IN    WKS    143.23.1.34.
                    FTP  TCP  SMTP  TELNET
```

Sono indicati l'intero nome di dominio e indirizzo Internet, come anche IN per indicare che è un record di classe Internet. Il tipo di record è indicato con WKS. I protocolli supportati dalla macchina a quell'indirizzo sono elencati dopo l'indirizzo. In realtà, questi sono bitmap che corrispondono a porte. Quando il bit della porta è ad 1, il servizio è supportato.

### IN-ADDR-ARPA

I campi address , come nel record di tipo Address, usano un formato speciale chiamato IN-ADDR-ARPA. Questo abilita la mappatura all'inverso dall'indirizzo al host name. Per comprendere è utile cominciare con un record di risorse di formato standard. Uno dei tipi più semplici di resource record è per l'indirizzo (tipo A). un estratto dal file indirizzo è mostrato qui

```
TPCI_HPWS1    IN    A    143.12.2.50
```

---

[48] IN-ADDR-ARPA

[49] Well Known Service WKS

```
TPCI_HPWS2    IN     A     143.12.2.51

TPCI_HPWS3    IN     A     143.12.2.52

TPCI_GATEWAY  IN     A     143.12.2.100

              IN     A     144.23.56.2

MERLIN        IN     A     145.23.24.1

SMALLWOOD     IN     A     134.2.12.75
```

Ogni linea del file rappresenta un record di risorse. In questo caso, essi sono tutte voci semplici che hanno il nome simbolico della macchina (alias) , la classe della macchina (IN per Internet), A per mostrare che è un record di tipo Address, e l'indirizzo Internet. La voce per la macchina TPCI_GATEWAY ha due indirizzi corrispondenti poiché esso è un gateway tra due network. Il gateway ha differenti indirizzi su ogn'una delle reti, così ha due record nello stesso file.

Questo tipo di file rende facile la mappatura da nome a indirizzo. Il name server semplicemente cerca una riga con il nome simbolico richiesto dall'applicazione e restituisce l'indirizzo Internet alla fine della riga. I database sono indicizzati sul nome, così queste ricerche procedono molto velocemente.

La ricerca dall'indirizzo al nome non è così semplice . se i file dei record di risorse sono piccoli, i ritardi pe runa ricerca manuale non sono apprezzabili, ma con reti ampie vi possono essere migliaia o decine di migliaia di voci. L'indice è sul nome per cui la ricerca per indirizzo può essere molto lenta. Per risolvere questo processo è stato ideato IN-ADDR_ARPA . esso usa l'indirizzo del host come indice .

Esso usa il tipo di record di risorse PTR per puntare dall'indirizzo al nome. Un esempio di file è il seguente

23.1.45.143.IN-ADDR-ARPA. PTR TPCI_HPWS_4.TPCI.COM

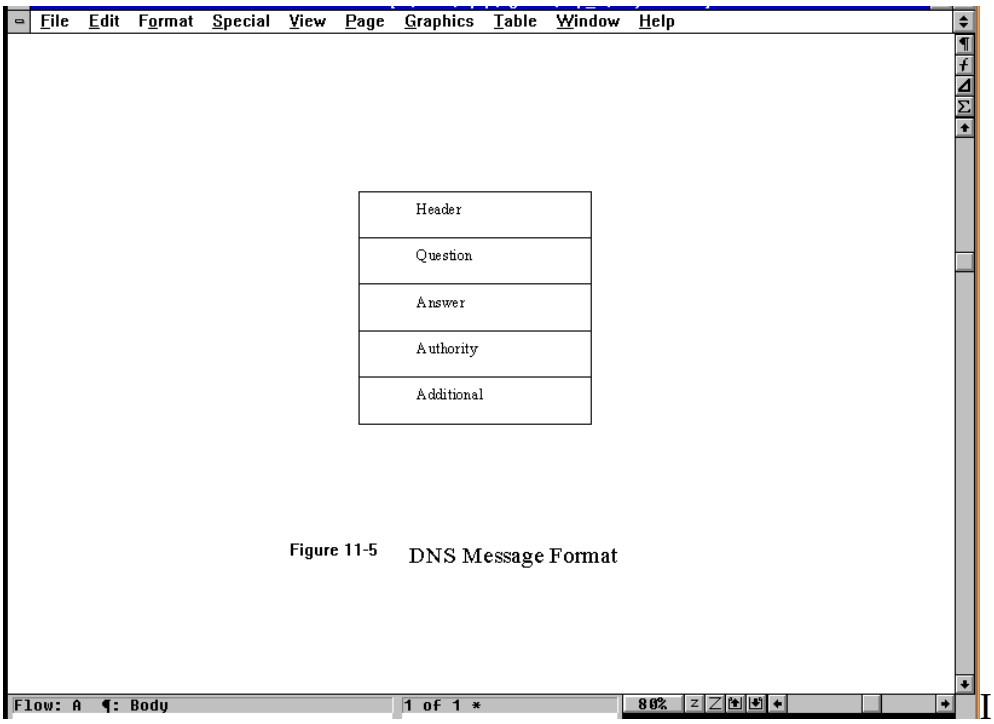
1.23.64.147.IN-ADDR-ARPA. PTR TPCI_SERVER.MERLIN.COM

3.12.6.123.IN-ADDR-ARPA. PTR BEAST.BEAST.COM

23.143.IN-ADDR-ARPA PTR MERLINGATEWAY.MERLIN.COM

**Messaggi**

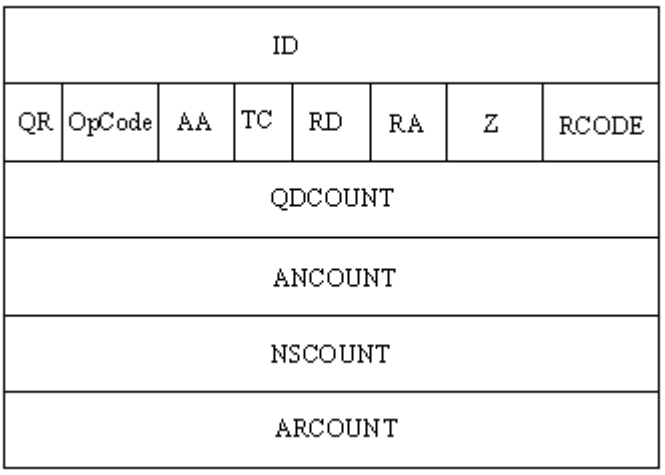
I messaggi DNS sono trasferiti fra name server per aggiornare i lo ro record di risorse. I campi di questi messaggi sono praticamente identici a quelli dei record stessi. Il formato è mostrato nella figura seguente




Il header ha diversi sottocampi che contengono informazioni circa il tipo di risposte o domande inviate. Il resto del messaggio consiste di quattro campi a lunghezza variabile:

- Question: l'informazione richiesta
- Answer: la risposta alla query
- Authority: il nome di altri server di nome che potrebbero avere l'informazione richiesta, se non è rapidamente disponibile sul name server target.
- Informazione addizionale: informazione che può essere fornita per rispondere alla query, o gli indirizzi dei name server se fu usato il campo Authority.

L'header di un messaggio DNS ha vari campi differenti esso stesso, come mostrato nella figura seguente




L'header è presente in tutti i messaggi DNS. Il campo ID è lungo 16 bit ed è usato per far corrispondere tra loro query e risposte. Il campo ad un solo bit QR è settato ad un valore 0 per indicare una query o un valore ad 1 per mostrare una risposta. Il campo OpCode è lungo 4 bit e può avere uno dei valori mostrati nella figura seguente

| *OpCode Value* | *Description* |
|---|---|
| ***0*** | Standard query |
| ***1*** | Inverse query |
| ***2*** | Server status request |

| | |
|---|---|
| ***3–15*** | Not used |

Il campo AA è il bit di authoritative answer. Un valore 1 in un messaggio di risposta indica che il name server è l'autorità riconosciuta per il nome di dominio cui si riferisce l'interrogazione. Il bit TC (truncation) è settato ad un valore 1 quando il messaggio è troncato a causa di una lunghezza eccessiva. Altrimenti il bit è settato a 0. il bit RD (recursion desired[50]) è settato ad 1 quando il name server riceve la richiesta di effettuare una query ricorsiva[51]. Il bit RA (recursion available) è settato ad 1 in una risposta quando il name server può effettuare ritorsioni.

Il campo Z è lungo 3 bit e non è usato. Il campo RCODE è lungo 4 bit e può essere settato con uno dei valori mostrati nella figura seguente

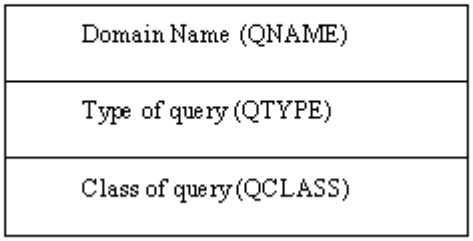
| ***RCODE Value*** | ***Description*** |
|---|---|
| ***0*** | No errors |
| ***1*** | Format error; name server unable to interpret the query |
| ***2*** | Name server problems have occurred |
| ***3*** | The name server could not find the domain reference in the query |
| ***4*** | Name server does not support this type of query |
| ***5*** | Name server cannot perform the requested operation for administrative reasons |
| ***6–15*** | Not used |

[50] recursion desired

[51] recursive query

Il campo QDCOUNT è un campo a 16 bit per i  numero di registrazioni nella sezione Question[52] . il campo ANCOUNT è un altro campo a 16 bit per il numero di repliche nella sezione Answer (il numero di record di risorse nella risposta). Il campo NSCOUNT è a 16 bit e specifica il numero di record di risorse di server nella sezione Authority del messaggio. Il campo ARCOUNT a 16 bit specifica il numero di record di risorse nella sezione del record Additional.

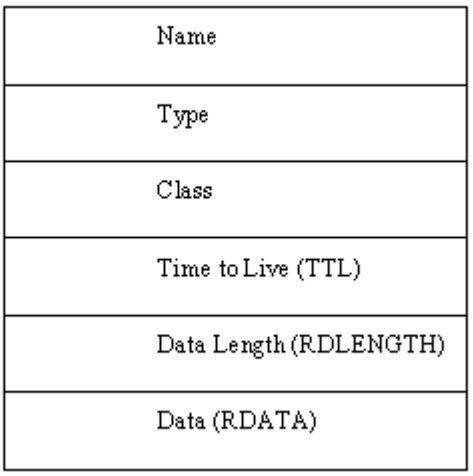
La sezione Question del messaggio ha tre campi come mostrato nella figura seguente




Il campo Question porta la query, che è identificata da questi campi. Il campo QNAME è il nome del dominio richiesto. Il QTYPE è il tipo di query . il campo QCLASS è la classe della query che può essere settato in accodo con le richieste della applicazione.

Gli ultimi tre campi nel messaggio DNS (Answer, Authority, Additional Infomation) hanno tutti lo stesso formato, come mostrato nella figura seguente

[52] question




Il campo Name contiene il nome del dominio del record di risorse. Il campo Type ha uno qualsiasi dei valori del tipo di record validi.

Il campo Class è la classe dei dati nel campo dati. Il campo TTL (Time to Live) è il numero di secondi durante i quali l'informazione è valida senza bisogno di aggiornamenti. Il campo RDLENGHT è la lunghezza dell'informazione nel campo dati. Infine il campo RDATA è l'informazione del record di risorse o altri dati. , in dipendenza della classe e del tipo di query o replica.

### Il Name Resolver

Dal punto di vista delle applicazioni, la risoluzione di un nome simbolico in un indirizzo di rete è facile. L'applicazione manda una query ad un processo chiamato name resolver . il name resolver potrebbe essere in grado di risolvere il nome direttamente , nel qual caso esso manda un messaggio di ritorno alla applicazione. Se il resolver non può determinare l'indirizzo di rete, esso comunica con il name server (che potrebbe mettersi in contatto con un altro name server).-

Il resolver è pensato per sostituire sistemi di risoluzione dei nomi esistenti su una macchina come i file /etc/hosts su macchine Unix. La sostituzione di questi meccanismi comuni è trasparente all'utente , sebbene l'amministratore debba sapere se il sistema di risoluzione dei nomi nativo viene usato su ogni macchina in modo da poter gestire le tabelle corrette.

Quando il resolver acquisisce informazioni da un name server , esso immagazzina le voci nella sua cache per ridurre la necessità di traffico se lo stesso nome simbolico viene usato di nuovo. L'ammontare di tempo per il quale il name resolver tiene i dati immagazzinati dipende al parametro Time to Live, o da un valore di default settato nel sistema.

Quando un name server non può risolvere un nome , esso può rimandare indietro un messaggio al resolver con l'indirizzo di un altro name server nel campo Authority del messaggio. Il resolver deve indirizzare poi un messaggio all'altro name server nella speranza che possa risolvere il nome. Il resolver può chiedere al primo name server di condurre la query esso stesso settando il bit RD (Recursive ) nel messaggio. Il name server può accettare o rifiutare la richiesta.

Il resolver usa sia UDP che tCP nella sua query , sebbene UDP sia più comune a causa della sua velocità. Comunque, query iterative o trasferimenti di grosse quantità di informazioni potrebbero spingere verso TCP per la sua maggiore affidabilità.

Sotto il sistema operativo Unix, sono in uso molte implementazioni diverse del name resolver. Il resolver fornito con le versioni BSD di Unix era particolarmente limitato,

non offrendo ne una cache né capacità di query iterativa. Per risolvere queste limitazioni, fu aggiunto il Berkeley Internet Name Domain (BIND) .

### Configurare un server DNS Unix.

Configurare un server DNS richiede la creazione o modifica di molti file e database. I file coinvolti nella maggior parte dei setup DNS e i loro scopi sono i seguenti:

named.hosts: definisce il dominio con la mappatura hostname-to-IP

named.rev: usato in IN-ADDR-ARPA per la mappatura IP-to_hostname

named.local: usato per risolver il driver loopback

named.ca: elenca i server di dominio radice

named.boot: usato per settare le locazioni di file e database

Il file principale è il named-boot che viene letto quando il sistema parte e definisce gli altri file nel set. Quindi, ogni cambiamento nei file viene riflesso nel file named.boot.

### Entrare nei Resource Record

Per la configurazione del server di esempio assumiamo di avere un sistema Unix che usa nomi e layout di rete standard. DNS consente di realizzare cose veramente complesse, ma è più semplice vedere cosa fanno i file ei record di risorse con un layout semplice.

Un record di risorse SOA viene inserito nel file named.hosts. Punti e virgola sono usati nel record per i commenti. Questo record di risorse è stato formattato come un solo record per riga per rendere chiare le sue voci, sebbene questo non sia necessario. Questo record di risorse definisce un confine superiore del dominio tcpi.com con

server.tcpi.com come server di nome primario per il dominio , root. Merlin.tcpi.com come indirizzo di posta elettronica della persona responsabile per il dominio:

```
tpci.com.   IN   SOA     server.tpci.com
   root.merlin.tpci.com (
   2  ; Serial number
   7200 ; Refresh (2 hrs)
   3600 ; Retry (1 hr)
   151200 ; Expire (1 week)
    86400 ); min TTL
```

si noti come l'informazione dal numero seriale al campo empire è racchiusa fra parentesi.questo è parte della sintassi del comando.

In aggiunta al record di risorse SOA, il file named.hosts contiene record Address. Questi record sono usati  per la mappatura reale dei nomi degli host sugli indirizzi IP. Alcuni record di risorse Address mostrano il formato di queste registrazioni :

```
artemis     IN     A     143.23.25.7
merlin      IN     A     143.23.25.9
pepper      IN     A     143.23.25.72
```

i nomi degli host non vengono dati come nomi di domini completi  poiché il server è in grado di dedurre l'intero nome. Se si vuole dare il nome completo, si deve fare seguire al nome un punto. L'esempio precedente diventa

```
artemis.tpci.com.     IN     A     143.23.25.7
```

```
merlin.tpci.com.      IN     A    143.23.25.9

pepper.tpci.com.      IN     A    143.23.25.72
```

Il record di risorse Pointer (PTR) è usato per mappare un indirizzo Ip ad un nome usando IN-ADDR-ARPA. Ad esempio il record

```
7.0.120.147.in-addr.arpa IN PTR merlin
```

indica che la macchina chiamata merlin ha l'indirizzo IP 147.120.0.7

I record di risorse Name Server puntano al server di nome che ha autorità per un particolare zona. I record Name Server (NS) sono usati quando una grande rete ha diverse sottoreti, ogn'uno con il suo server di nome. Una registrazione assomiglia a qualcosa del genere

```
tpci.com   IN   NS  merlin.tpci.com
```

Questo record indica che il server DNS per il dominio tpci.com è chiamato merlin.tpci.com. Se vi fossero diverse sottoreti usate in tpci.com , vi dovrebbe esser un record NS per ogni sottorete.

**Completare i file DNS**

DNS usa diversi file per contenere record di risorse che descrivono le zone usate da DNS. Il primo file di interesse è named.hosts che contiene i record SOA, NS, e A. tutte le voci nel file devono iniziare nella posizione del primo carattere di ogni riga. Ecco un esempio

```
; named.hosts files
```

```
; Start Of Authority RR
tpci.com.    IN    SOA     merlin.tpci.com
             root.merlin.tpci.com (
             2  ; Serial number
             7200 ; Refresh (2 hrs)
             3600 ; Retry (1 hr)
             151200 ; Expire (1 week)
             86400 ); min TTL
;
; Name Service RRs
tpci.com   IN   NS  merlin.tpci.com
subnet1.tpci.com IN NS goofy.subnet1.tpci.com
;
; Address RRs
artemis     IN     A    143.23.25.7
merlin      IN     A    143.23.25.9
windsor     IN     A    143.23.25.12
reverie     IN     A    143.23.25.23
bigcat      IN     A    143.23.25.43
pepper      IN     A    143.23.25.72
```

La prima sezione setta il record SOA, che definisce i parametri per TTL, expiry, refresh, e così via. Esso setta il server di nome per il dominio tpci.com a merlin.tpci.com. la seconda sezione usa il record di risorse NS per definire il server di nome per il dominio tpci.com come merlin.tpci.com (lo stesso di SOA) e una sottorete di tpci chiamata subnet1, per la quale il name server è goofy.subnet1.tpci.com . la terza sezione ha una lista della mappatura nome-indirizzo. C'è una registrazione in questa sezione per ogni macchina sulla rete.

Il file named.rev fornisce la mappatura inversa dell'indirizzo IP al nome della macchina ed è composto di record Pointer. Viene seguito lo stesso formato del file named.hosts , eccetto per lo scambio di nomi ed indirizzi IP e la conversione dell'indirizzo IP nello stile IN-ADDR-ARPA. Il file named.rev equivalente del file named.hosts visto prima è il seguente

```
; named.rev files
; Start Of Authority RR
23.143.in-addr.arpa    IN    SOA     merlin.tpci.com
        root.merlin.tpci.com (
        2  ; Serial number
        7200 ; Refresh (2 hrs)
        3600 ; Retry (1 hr)
        151200 ; Expire (1 week)
```

```
        86400 ); min TTL
;
; Name Service RRs
23.143.in-addr.arpa   IN   NS  merlin.tpci.com
100.23.143.in-addr.arpa   IN NS goofy.subnet1.tpci.com
;
; Address RRs
9.25.23.143.in-addr.arpa   IN   PTR merlin
12.25.23.143.in-addr.arpa  IN   PTR windsor
23.25.23.143.in-addr.arpa  IN   PTR reverie
43.25.23.143.in-addr.arpa  IN   PTR bigcat
72.25.23.143.in-addr.arpa  IN   PTR pepper
```

Ci deve essere un file named.rev per ogni zona o sottodominio sulla rete. Questi file possono avere nomi differenti o essere piazzati in directory diverse.

Il file named.local contiene una voce per il driver loopback (che ha sempre l'indirizzo IP 127.0.0.1). questo file deve contenere informazioni circa la mappatura IN-ADDR-ARPA del driver loopback. Un file named.local ha il seguente aspetto:

```
; named.local files
; Start Of Authority RR
0.0.127.in-addr.arpa    IN    SOA     merlin.tpci.com
        root.merlin.tpci.com (
```

```
            2  ; Serial number
            7200 ; Refresh (2 hrs)
            3600 ; Retry (1 hr)
            151200 ; Expire (1 week)
            86400 ); min TTL
;
; Name Service RR
0.0.127.in-addr.arpa   IN   NS  merlin.tpci.com
;
; Address RR
1.0.0.127.in-addr.arpa   IN   PTR localhost
```

il file named.ca è usato per specificare server di nome cui il sistema può ricorrere. Le macchine specificate nel file dovrebbero essere stabili e non soggette a rapidi cambiamenti. Un esempio di file named.ca è il seguente:

```
; named.ca
; servers for the root domain
;
.  99999999  IN   NS  ns.nic.ddn.mil.
   99999999  IN   NS  ns.nasa.gov.
```

```
   99999999  IN   NS  ns.internic.net

; servers by address

;

ns.nic.ddn.mil   99999999   IN  A  192.112.36.4

ns.nasa.gov      99999999   IN  A  192.52.195.10

ns.internic.net  99999999   IN  A  198.41.0.4
```

in questo file soltanto tre server DNS sono stati specificati. Un normale file named.ca può avere una dozzina di name server , in dipendenza della loro prossimità al vostro sistema. I server specificati nel file named.ca sono ogn'uno identificato da due voci. Una da il dominio radice seguita dal nome del server di nome , l'altra ha l'indirizzo IP del name server. Il campo Time To Live è settato ad un valore molto alto perché ci si aspetta che questi server siano sempre disponibili.

Il file named.boot è usato per lanciare il caricamento dei daemon DNS e specificare i server di nome primario e secondario sulla rete. Un esempio di file è il seguente

```
; named.boot

directory    /usr/lib/named

primary    tpci.com    named.hosts

primary    25.143.in-addr.arpa    named.rev

primary    0.0.127.in-addr.arpa    named.local

cache    .    named.ca
```

La prima riga del file named.boot ha la directory chiave seguita dalla directory dei file di configurazioni DNS. Ogni riga seguente dice a DNS i file che esso dovrebbe usare per trovare le informazioni di configurazione. La prima riga, per esempio, setta named.hosts come il file per localizzare il server primario di tpci.com. le informazioni IN-ADDR-ARPA è tenuta nel file named.rev per la sottorete 143.25. l'informazione del localhost è nel file named.local, e infine le informazioni sul server e sulla cache sono in named.ca.

Un server di nome secondario è configurato in maniera soltanto leggermente differente rispetto ad un server primario. La differenza è che il file named.boot punta al server primario.

**Far partire i daemon DNS**

Il passo finale nella configurazione DNS è di assicurare che il daemon DNS chiamato named viene caricato quando il sistema Unix parte. Questo è realizzato usualmente attraverso lo script di startup rc. La maggior parte delle versioni Unix ha le routine per lo startup DNS già inserite nello script di startup, usualmente nella forma di un controllo per il file named.boot. se esiste il file named.boot parte il daemon DNS. Il codice usualmente assomiglia a questo:

```
# Run DNS server if named.boot exists

if [ -f /etc/inet/named.boot -a -x /usr/sbin/in.named ]

then

   /usr/sbin/in.named
```

fi

### Configurare un client

Configurare una macchina Unix per usare un server DNS primario per la risoluzione è un processo rapido. Prima il file /etc/resolv.conf viene modificato per includere l'indirizzo del server primario. Per esempio, un file resolv.conf potrebbe essere del seguente tipo:

```
domain tpci.com

nameserver   143.25.0.1

nameserver   143.25.0.2
```

la prima riga stabilisce il nome del dominio, che è seguita dagli indirizzi IP dei server di nome disponibili.

### *Protocollo BOOTP*

TCP/IP ha bisogno di conoscere un indirizzo Internet prima di poter comunicare con altre macchine. Questo può causare problemi quando una macchina è inizialmente caricata o non ha propri dischi[53] dedicati. Precedentemente abbiamo visto come, per ottenere un indirizzo IP potesse essere utilizzato il protocollo Reverse Address Resolution Protocol, ma è in uso comune un'alternativa il protocollo bootstrap[54] o

[53] disk disk drive

[54] bootstrap protocol bootstrap

BOOTP[55] . BOOTP usa UDP per abilitare una macchina senza disco[56] a determinare il suo indirizzo IP senza usare RARP.

Macchine senza disco di solito contengono informazioni di startup nella loro PROM[57]. Poiché essa deve essere tenuta piccola e consistente tra molti modelli di workstation senza disco per ridurre codici, è impossibile integrare un protocollo completo come TCP/IP in un singolo chip[58]. È anche impossibile inserire un indirizzo IP nel chip, poiché il chip può essere usato in molte differenti macchine sulla stessa rete. Questo forza una macchina senza disco che ha appena effettuato il bootstrap a determinare il suo indirizzo IP da un'altra macchina sulla rete. (In pratica la macchina senza disco deve determinare anche l'indirizzo IP del server di rete che esso userà così come l'indirizzo del più vicino gateway IP).

BOOTP supera alcuni dei problemi di RARP. RARP richiede accesso diretto al hardware della rete, che può causare problemi quando si ha a che fare con server. Inoltre, RARP fornisce soltanto un indirizzo IP. Quando devono essere inviati grandi pacchetti , questo spreca un sacco di spazio che potrebbe essere usato per informazioni utili. BOOTP fu sviluppato per usare UDP e può essere implementato all'interno di un programma applicativo. BOOTP richiede inoltre soltanto un singolo

---

[55] BOOTP

[56] diskless machine diskless machine

[57] PROM

[58] chip

pacchetto di informazioni per fornire tutte le informazioni che una nuova workstation senza disco richiede per iniziare le operazioni.

Per determinare l'indirizzo IP di una workstation senza disco , BOOTP usa le capacità broadcast di IP. (occorre ricordare che IP abilita alcuni indirizzi di rete speciali che sono inviati in broadcast a tutte le macchine della rete). Questo permette alla workstation di inviare un messaggio anche quando non consoce l'indirizzo della macchina di destinazione o perfino il suo.

BOOTP pone tutti i compiti di comunicazione nella stazione senza disco. Esso specifica che tutti i messaggi UDP inviati sulla rete usano checksum e che i bit Do Not Fragment sia settato. Questo tende a ridurre il numero di datagrammi persi, interpretati in maniera non corretta o duplicati.

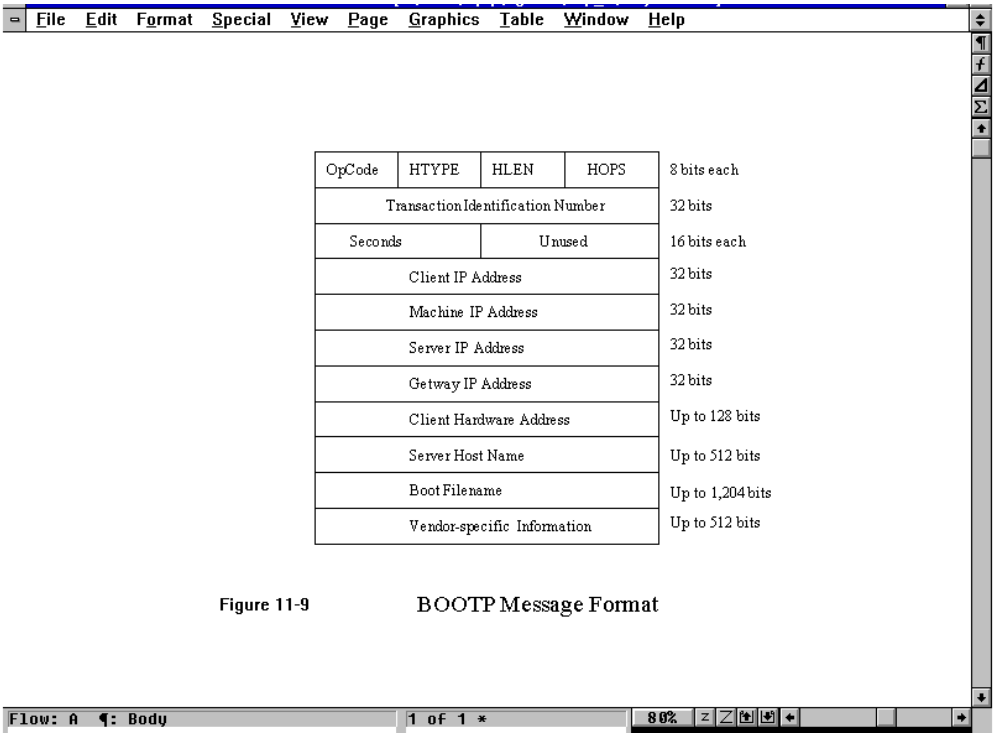
Per gestire la perdita di un messaggio , BOOTP usa un semplice set di timer. Quando un messaggio è stato inviato parte un timer. Se non si è ricevuta replica quando si esaurisce il timer, il messaggio viene rinviato. Il protocollo prevede che il timer sia settato ad un valore casuale, che si incrementa ogni volta che il timer si ferma fino a raggiungere un valore massimo , dopo il quale esso viene di nuovo settato ad un valore casuale. Questo evita traffici elevati dopo che diverse macchine falliscono nello stesso momento e tentano di inviare in broadcast messaggi BOOTP nello stesso momento.

BOOTP usa i termini server e client per fare riferimento alle macchine. Il client è la macchina che inizia una query , e il server è la macchina che risponde alla query. Da queste definizioni si vede che i termini client e server non hanno relazione fisica con

alcuna workstation , poiché il ruolo di ogni macchina può cambiare con il traffico. Poiché la maggior parte dei sistemi possono gestire multipli flussi di traffico allo stesso momento, è possibile che una macchina sia contemporaneamente client e server.

### Messaggi BOOTP

I messaggi BOOTP hanno formati fissi per semplicità e per permettere al software BOOTP di potere essere integrato nel piccolo spazio di una PROM. Il formato di un messaggio BOOTP è mostrato nella figura seguente




Il campo OpCode è usato per segnalare o una richiesta (settato al valore 1) o una replica (settato al valore 2). Il campo HTYPE il tipo di hardware di rete. Il campo HLEN indica la lunghezza di un indirizzo hardware.

Il campo HOPS tiene traccia del numero di volte in cui il messaggio è instradato. Quando il client manda il messaggio di richiesta , un valore 0 è posto nel campo HOPS. Se il server decide di instradare il messaggio verso un'altra macchina , esso incrementa il campo HOPS .

Il campo Transaction Number è un intero assegnato dal client al messaggio e rimane invariato dalla richiesta alla replica. Questo abilita il confronto fra repliche e richieste corrette. Il campo Seconds è il numero di secondi da cui il cliente è stato avviato, assegnato dal cliente quando il messaggio viene inviato.

Il campo Client IP Address è riempito per quanto possibile dal client. Questo potrebbe dare come risultato un indirizzo parziale o nessuna informazione , in dipendenza della conoscenza che il client ha della rete. Ogni informazione non nota è settata a 0 (così il campo potrebbe essere 0.0.0.0 se non si sa nulla circa l'indirizzo di rete). Se il client vuole informazioni da un server particolare , esso può porre l'indirizzo del server nel campo Server IP Address. In maniera similare, se il client conosce il nome del server lo pone nel campo Server Host Name. Se i campi sono settati a 0, ogni server può rispondere . se è dato un server specifico o un gateway , solo quella macchina risponde al messaggio.

Il campo Vendor-specific è usato ,come suggerisce il nome, per informazioni di implementazione specifiche per ogni venditore. Questo campo è opzionale. I primi 32 bit definiscono il formato dell'informazione rimanente. Questi primi bit sono noti

come magic cookie[59] . Dopo il magic cookie vi sono set di informazioni in un formato a tre campi: un tipo, una lunghezza ed un valore. Vi sono diversi tipi identificati da RFC internet come mostrato nella tabella seguente . il campo Lenght non è usato peri i tipi 0 e 255, ma deve essere presente per i tipi 1 e 2. la lunghezza può variare in dipendenza del numero di voci negli altri tipi di messaggi.

| Type | Code | Length | Description |
|---|---|---|---|
| ***Padding*** [60] | 0 | -- | Used only for padding messages |
| ***Subnet Mask*** | 1 | 4 | Subnet mask for local network |
| ***Time of Day*** | 2 | 4 | Time of Day |
| ***Gateways*** | 3 | Number of entries | IP addresses of gateways |
| ***Time Servers*** | 4 | Number of entries | IP addresses of time servers |
| ***IEN116 Server*** | 5 | Number of entries | IP addresses of IEN116 servers |
| ***DomainName Server*** | 6 | Number of entries | IP addresses of Domain Name Servers |
| ***Log Server*** | 7 | Number of entries | IP addresses of log servers |
| ***Quote Server*** | 8 | Number of entries | IP addresses of quote servers |
| ***LPR Servers*** | 9 | Number of entries | IP addresses of lpr servers |
| ***Impress*** | 10 | Number of entries | IP addresses of impress servers |
| ***RLP Server*** | 11 | Number of entries | IP addresses of RLP servers |
| ***Hostname*** | 12 | Number of bytes | Client host name in host name |
| ***Boot size*** | 13 | 2 | Integer size of boot file |
| ***Unused*** | 128–254 | -- | Not used |
| ***End*** | 255 | -- | End of list |

[59] magic cookie cookie

[60] padding

Il campo BOOT Filename può specificare un filename da cui ottenere un'immagine di memoria che permette alla workstation senza disco di effettuare correttamente il boot. Questo permette che l'immagine di memoria sia ottenuta da una macchina mentre gli indirizzi reali siano ottenuti da un'altra. Se questo campo è settato a 0, il server seleziona l'immagine di memoria da inviare.

Il processo di boot segue due passi. Il primo è di usare BOOTP per ottenere informazioni circa l'indirizzo di memoria del client e almeno un'altra macchina (un gateway o server). Il secondo passo usa un protocollo differente per ottenere un'immagine di memoria dal client.

### *Network Time Protocol[61] (NTP)*

Le temporizzazione sono veramente importanti per le reti, non solo per assicurare che i timer interni sono gestiti in maniera appropriata , ma anche per la sincronizzazione dei clock per mandare messaggi e timestamp all'interno di quei messaggi. Alcuni sistemi si affidano sul tempo per l'instradamento. Assicurare che i clock siano consistenti e accurati è un compito spesso trascurato, ma esso rimane importante abbastanza da avere una procedura formale definita un RFC Internet. Il protocollo che gestisce gli standard di tempo è chiamato Network Time Protocol (NTP). Esso può essere usato sia con TCP che UDP, la porta 37 è dedicata ad esso.

Le operazioni di NTP si basano sull'ottenere un tempo accurato da una query ad un server di tempo primario[62], il quale ottiene le sue informazioni di timing da una

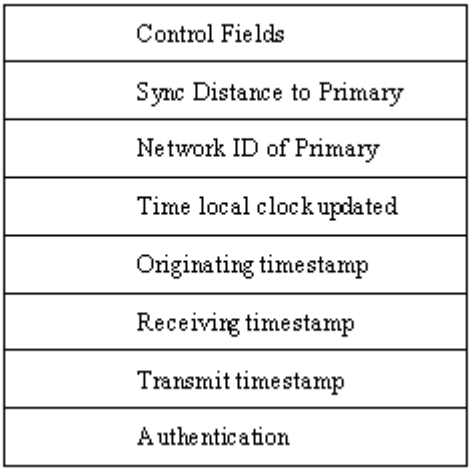
[61] Network Time Protocol NTP

sorgente di tempo standard (come il National Institute of Standards and Technology[63] negli Stati Uniti).-

Il time server interroga l'orologio standard (chiamato anche master clocking source) e setta il proprio tempo.

Una volta che il primari time server ha un tempo accurato esso manda messaggi NTP a server di tempo secondari sulla rete. Server secondari possono comunicare con altri servers secondari usando NTP, sebbene l'accuratezza si perda con ogni comunicazione a causa della latenza nelle reti. Eventualmente, questi messaggi sul tempo possono essere mandati a gateway e macchine individuali nella rete, se l'amministratore lo decide. Usualmente ogni rete ha almeno un primary time server e un server secondario.

Il formato di un messaggio NTP è mostrato nella figura seguente.




---

[62] primary time server

[63] National Institute of Standards and Technology

diversi campi di controllo sono usati per procedure di sincronizzazione e aggiornamento. Il campo Sync Distance to Primary è una stima del ritardo di round-trip occorso al clock primario.

Vi sono diversi timestamp[64] nel messaggio NTP. Il Time Local Clock Update è il tempo in cui l'orologio locale del generatore dei messaggi era stato aggiornato. Il timestamp Originate è il tempo in cui il messaggio fu inviato. Il timestamp Receive è il tempo in cui il messaggio fu ricevuto. Il timestamp Transmit è il tempo in cui il messaggio fu trasmesso dopo la ricezione.

Tutti i timestamp sono calcolati da un offset del numero di secondi dal 1° gennaio 1900. i campi timestamp sono a 32 bit, i primi 32 per un numero intero e gli ultimi 32 per una frazione. Il campo Authentication è opzionale e può esser usato per autenticare un messaggio.

[64] timestamp