Linux IPv6 HOWTO (it)

Peter Bieringer

pb at bieringer.de

Linux IPv6 HOWTO (it)

Peter Bieringer

L'obiettivo di questo HOWTO è quello di rispondere alle domande, sia di base che avanzate, sull'implementazione di IPv6 sul sistema operativo Linux. Questo HOWTO fornirà al lettore informazioni sufficienti per installare, configurare ed usare applicazioni che utilizzano IPv6 su macchine Linux. Traduzione a cura di Michele Ferritto (m.ferritto--at--virgilio.it) e revisione a cura di Daniele Masini (d.masini--at--tiscali.it).

Diario delle revisioni

Revisione Versione 0.47.it.1 04-09-2004 Corretto da: MF
Si veda lo storico delle revisioni per maggiori dettagli
Revisione Versione 0.46.it.1 24-03-2004 Corretto da: MF
Si veda lo storico delle revisioni per maggiori dettagli
Revisione Versione 0.45.it.2 23-01-2004 Corretto da: MF
Si veda lo storico delle revisioni per maggiori dettagli
Revisione Versione 0.45.it.1 12-01-2004 Corretto da: MF
Si veda lo storico delle revisioni per maggiori dettagli
Revisione Versione 0.44.2.it.1 16-10-2003 Corretto da: MF
Si veda lo storico delle revisioni per maggiori dettagli

Sommario

Capitolo 1. Generale

* *CVS-ID: $Id: Linux+IPv6-HOWTO.lyx,v 1.96 2004/07/19 20:08:47 pbldp Exp $*

Informazioni sulle traduzioni disponibili possono essere trovate nella sezione Traduzioni.

1.1. Copyright, licenza d'uso e altro

1.1.1. Copyright

1.1.2. Licenza d'uso

1.1.3. A proposito dell'autore

1.1.3.1. Storia Internet/IPv6 dell'autore

- 1993: Sono entrato in contatto con Internet utilizzando client e-mail e news basati su console (per esempio si cerchi "e91abier" su groups.google.com (http://groups.google.com/), sono io).
- 1996: Ho ricevuto una richiesta per progettare un corso su IPv6, comprendente un seminario con il sistema operativo Linux.
- 1997: Ho iniziato a scrivere una guida su come installare, configurare e usare IPv6 sui sistemi Linux, chiamata IPv6 &Linux - HowTo (http://www.bieringer.de/linux/IPv6/) (si veda IPv6 &Linux - HowTo/History (http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/IPv6-HOWTO-0.html#history) per maggiori informazioni).
- 2001: Ho iniziato a scrivere questo nuovo Linux IPv6 HOWTO.

1.1.3.2. Contatti

L'autore può essere contattato via e-mail presso <pb at bieringer dot de> e anche tramite la sua homepage (http://www.bieringer.de/pb/).

Attualmente egli vive a Monaco [zona settentrionale di Schwabing] / Bavaria / Germania (sud) / Europa (centrale) / Terra (superfice/continente).

1.2. Categoria

Questo HOWTO dovrebbe essere elencato nella categoria "*Networking*/*Protocols*".

1.3. Versione, Storia e Da Fare

1.3.1. Versione

La versione attuale è indicata all'inizio del documento.

Per altre versioni/traduzioni disponibili, si veda anche http://www.bieringer.de/linux/IPv6/.

1.3.2. Storia

1.3.2.1. Storia principale

30-11-2001: Inizio del progetto del nuovo HOWTO.

02-02-2002: Molto del contenuto completato, primo rilascio pubblico del capitolo 1 (versione 0.10).

14-01-2002: Più completo, qualche revisione, rilascio pubblico dell'intero documento (versione 0.14).

16-08-2002: Procede la traduzione in Polacco

31-10-2002: Disponibile la traduzione in Cinese (si veda Traduzioni per maggiori dettagli)

10-11-2002: Procede la traduzione in Tedesco

10-02-2003: Disponibile la traduzione in Tedesco

09-04-2003: Procede la traduzione in Francese

09-05-2003: Disponibile la traduzione in Francese

15-08-2003: Procede la traduzione in Spagnolo

16-10-2003: Procede la traduzione in Italiano

12-03-2004: Disponibile la traduzione in Italiano

18-06-2004: Procede la traduzione in Greco

29-08-2004: La traduzione in Spagnolo è sospesa

1.3.2.2. Storia completa

Si veda lo Storico delle Revisioni alla fine di questo documento.

1.3.3. Da Fare

- Riempimento dei contenuti mancanti
- Terminare controllo grammaticale

1.4. Traduzioni

Le traduzioni devono sempre contenere l'URL, il numero di versione e il copyright del documento originale (ma anche quelli della traduzione). Si prega di non tradurre il changelog originale, non è molto utile. Sembra che la frequenza delle modifiche al documento sia inferiore ad una volta al mese. A partire dalla versione 0.27 inoltre sembra che la maggior parte dei contenuti ai quali ho contribuito, sia stata scritta. Le traduzioni devono sempre essere effettuate a partire dalla versione inglese.

1.4.1. Traduzioni disponibili

1.4.1.1. Italiano

La traduzione in Italiano, a cura di Michele Ferritto, è stata aggiornata il 4 settembre 2004 ed è relativa alla versione 0.47 del documento originale. Detta traduzione è disponibile presso il sito web dell'Italian Linux Documentation Project (http://ildp.pluto.it). Il traduttore può essere contattato all'indirizzo m.ferritto@toglimi.virgilio.it.

1.4.1.1.1. Altre traduzioni

Per informazioni sulle altre traduzioni disponibili si veda il documento originale presso il TLDP (http://www.tldp.org/HOWTO/Linux+IPv6-HOWTO/general-translations.html).

1.5. Informazioni tecniche

1.5.1. Sorgente originale di questo HOWTO

Questo HOWTO e' stato scritto con LyX versione 1.2.0 su di un sistema Red Hat Linux 7.3 con template SGML (libro DocBook). È disponibile presso TLDP-CVS / users / Peter-Bieringer (http://cvsview.tldp.org/index.cgi/LDP/users/Peter-Bieringer/) per contributi.

1.5.1.1. Wrapping delle linee di codice

Il wrapping delle linee di codice è ottenuto utilizzando l'applicazione "lyxcodelinewrapper.pl", creata dall'autore, che è possibile scaricare dal CVS per i propri scopi: TLDP-CVS / users / Peter-Bieringer (http://cvsview.tldp.org/index.cgi/LDP/users/Peter-Bieringer/).

1.5.1.2. Generazione dello SGML

L'output SGML è stato generato utilizzando le funzioni di esportazione di LyX.

Si sono resi necessari anche alcuni aggiustamenti per ottenere un codice SGML corretto (per i programmi Perl si veda anche TLDP-CVS / users / Peter-Bieringer (http://cvsview.tldp.org/index.cgi/LDP/users/Peter-Bieringer/)):

- L'esportazione delle tabelle di LyX non crea correttamente i tag "colspan" - strumento per risolvere: "sgmllyxtabletagfix.pl" (risolto a partire da LyX 1.2.0).
- LyX a volte usa speciali entità per le virgolette (a destra ed a sinistra), invece di quelle normali, che ancora esistono nello HTML generato. Qualche browser non riesce a comprenderle correttamente (ad esempio Opera 6 TP 2 o Konqueror) - strumento per risolvere: "sgmllyxquotefix.pl".

1.5.2. Riferimenti on-line alla versione HTML di questo HOWTO (collegamenti/anchors)

1.5.2.1. Pagina indice Master

Di solito, un riferimento alla pagina indice master è raccomandato.

1.5.2.2. Pagine dedicate

Poiché le pagine HTML vengono generate da un file SGML, i nomi dei file risultanti sono praticamente casuali. Tuttavia, alcune pagine sono etichettate in LyX, dando come risultato nomi statici. Queste etichette sono utili come riferimento e non dovrebbero essere modificate in futuro.

Se pensate che abbia dimenticato qualche tag, per favore fatemelo sapere così da poterlo aggiungere.

1.6. Prefazione

Alcune cose prima di tutto:

1.6.1. Quante versioni di HOWTO su Linux & IPv6 ci sono in giro?

Inclusa questa, ci sono tre (3) documenti HOWTO disponibili. Scusate se sono troppi ;-)

1.6.1.1. Linux IPv6 FAQ/HOWTO (obsoleto)

Il primo documento relativo a IPv6 fu scritto da *Eric Osborne* e chiamato Linux IPv6 FAQ/HOWTO (http://www.linuxhq.com/IPv6/) (per favore lo si usi solo per riferimenti storici). L'ultima versione è stata la 3.2.1 rilasciata il 14 Luglio 1997.

Se qualcuno fosse a conoscenza della data relativa alla prima stesura di tale HOWTO, mi mandi una e-mail (è un'informazione necessaria per la "storia").

1.6.1.2. IPv6 & Linux - HowTo (mantenuto)

* *Questo HOWTO è veramente chiamato "HowTo"*

Esiste una seconda versione chiamata IPv6 &Linux - HowTo (http://www.bieringer.de/linux/IPv6/) scritta da me (*Peter Bieringer*) in HTML puro. Ha visto la luce nell'Aprile del 1997 e la prima versione in inglese è stata pubblicata nel Giugno del 1997. Continuerò a mantenerla anche se si dissolverà lentamente (ma non del tutto) in favore del Linux IPv6 HOWTO che state leggendo.

1.6.1.3. Linux IPv6 HOWTO (questo documento)

Poiché IPv6 &Linux - HowTo (http://www.bieringer.de/linux/IPv6/) è scritto in puro HTML, non è del tutto compatibile con The Linux Documentation Project (TLDP) (http://www.tldp.org/), il sottoscritto (*Peter Bieringer*) verso la fine del Novembre 2001, ha ricevuto una richiesta per riscrivere lo IPv6 &Linux - HowTo (http://www.bieringer.de/linux/IPv6/) in SGML. Comunque, a causa della dismissione dell'HOWTO Future of IPv6 &Linux - HowTo (http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/IPv6-HOWTO-0.html#history) e del fatto che IPv6 sta divenendo uno standard sempre più utilizzato, ho deciso di scrivere un nuovo documento sulle questioni di base e avanzate che rimaranno importanti per i prossimi anni. Maggiori contenuti dinamici e qualche dettaglio avanzato si potranno trovare nel secondo HOWTO (IPv6 &Linux - HowTo (http://www.bieringer.de/linux/IPv6/)).

1.7. Termini utilizzati, glossario e abbreviazioni

1.7.1. Relativi alla rete

Base 10

Conosciuto sistema numerico decimale, nel quale ogni valore viene rappresentato con le cifre da 0 a 9.

Base 16

Utilizzato solitamente nei linguaggi di programmazione a basso ed alto livello, conosciuto anche come sistema numerico esadecimale, nel quale ogni valore viene rappresentato con le cifre da 0 a 9 ed i caratteri da A a F (senza distinzione tra maiuscole e minuscole).

Base 85

Sistema di rappresentazione di ogni valore con 85 differenti numeri e caratteri; esso porta a sequenze di cifre più corte - mai visto prima.

Bit

La più piccola unità di immagazzinamento dati, acceso/vero (1) o spento/falso (0).

Byte

Principalmente un insieme di 8 (ma non necessariamente - si vedano i sistemi più vecchi) bit.

Dispositivo

In questo caso, hardware per la connessione di rete, si veda anche NIC.

Dual homed host

Un dual homed host è un nodo (sistema) con due interfacce di rete (fisiche o virtuali) su due differenti collegamenti che non inoltra nessun pacchetto tra le interfacce.

Host

Generalmente un single homed host su di un collegamento. Normalmente ha solo una interfaccia di rete attiva, per esempio Ethernet o PPP.

Interfaccia

Essenzialmente lo stesso di "dispositivo", si veda anche NIC.

Intestazione IP (IP Header)

Intestazione di un pacchetto IP (ogni pacchetto di rete ha un'intestazione, il cui tipo dipende dal livello di rete).

Link (collegamento)

Il link è il meccanismo di comunicazione dei pacchetti di rete di livello 2 OSI; esempi sono Ethernet, Token Ring, PPP, SLIP, ATM, ISDN, Frame Relay, ...

Nodo

Un nodo è un host o un router.

Ottetto

Un insieme di 8 bit, attualmente è analogo al "byte".

Porta

Informazione necessaria al dispatcher TCP/UDP (livello 4 OSI) per il trasporto delle informazioni agli strati superiori.

Protocollo

Ogni livello di rete contiene di solito un campo protocollo per facilitare le operazioni di recapito delle informazioni trasportate al livello superiore come per esempio il livello 2 (MAC) e 3 (IP).

Router

Un router è un nodo con 2 o più interfacce di rete (fisiche o virtuali) capace di inoltrare pacchetti tra le interfacce.

Socket

Un socket IP è definito dal legame tra l'indirizzo IP del mittente (sorgente), l'indirizzo IP di destinazione e le relative Porte (binding).

Stack

Un insieme di livelli di rete.

Maschera di sottorete (Subnet mask)

Le reti IP usano maschere di bit per separare le reti locali da quelle remote.

Tunnel

Un tunnel è tipicamente una connessione punto-punto nella quale i pacchetti scambiati contengono al loro interno pacchetti di un altro protocollo, come per esempio in un tunnel IPv6-in-IPv4.

1.7.1.1. Abbreviazioni

ACL

Access Control List.

API

Application Programming Interface.

ASIC

Application Specified Integrated Circuit.

BSD

Berkeley Software Distribution.

CAN-Bus

Controller Area Network Bus (physical bus system).

ISP

Internet Service Provider

KAME

Progetto - uno sforzo congiunto di sei società giapponesi per fornire al pubblico uno stack IPv6 e IPsec (sia per IPv4 che per IPv6) libero, per varianti BSD www.kame.net (http://www.kame.net/).

LIR

Local Internet Registry

NIC

Network Interface Card.

RFC

Request For Comments - Insieme di note tecniche e organizzative che riguardano Internet.

USAGI

UniverSAl playGround for Ipv6 Project - lavori per mettere in produzione uno stack IPv6 di qualità per il sistema Linux.

1.7.2. Informazioni relative al testo

1.7.2.1. Carattere di collegamento tra più righe di codice

Il carattere speciale "¬" viene usato per segnalare che la linea di codice corrente fa parte di quella precedente, ma è rappresentata fisicamente su una riga diversa per una migliore visualizzazione nei file PDF e PS.

1.7.2.2. Segnaposto

Negli esempi generici a volte si può trovare la seguente dicitura:

```
<myipaddress>
```

Per l'uso concreto nella propria riga di comando o in uno script, tale dicitura deve essere sostituita con il contenuto opportuno (rimuovendo < e > naturalmente); in questo caso il risultato potrebbe essere, per esempio

```
1.2.3.4
```

1.7.2.3. Comandi nella shell

Comandi eseguiti da utenti non-root iniziano con il carattere $, ad esempio

```
$ whoami
```

Comandi eseguiti come utente root iniziano con il carattere #, ad esempio

```
# whoami
```

1.8. Requisiti per usare questo HOWTO

1.8.1. Prerequisti personali

1.8.1.1. Esperienza con gli strumenti Unix

Si dovrebbe avere familiarità con i principali strumenti Unix, ad esempio *grep*, *awk*, *find*, ... e conoscere le relative opzioni di comando più comuni.

1.8.1.2. Conoscenza della teoria delle reti

Si dovrebbe essere a conoscenza di livelli, protocolli, indirizzi, cavi, plug, ecc. Se si è nuovi in questo campo, qui c'è un buon punto di partenza: linuxports/howto/intro_to_networking (http://www.linuxports.com/howto/intro_to_networking/).

1.8.1.3. Esperienza con la configurazione di IPv4

Si deve avere necessariamente avuto qualche esperienza nella configurazione di IPv4, altrimenti sarà dura capire cosa sta veramente succedendo.

1.8.1.4. Esperienza con il Domain Name System (DNS)

Bisogna anche comprendere cosa è il Domain Name System (DNS), cosa fornisce e come va usato.

1.8.1.5. Esperienza con le strategie di network debugging

È necessario almeno capire come usare *tcpdump* e cosa può mostrare. Diversamente, il network debugging risulterà molto ostico.

1.8.2. Hardware compatibile con il sistema operativo Linux

Sicuramente si vorranno fare esperimenti con del vero hardware e non semplicemente leggere questo HOWTO addormentandosi qua e là. ;-7)

Capitolo 2. Fondamenti

2.1. Cosa è IPv6?

IPv6 è un nuovo protocollo di livello 3 OSI (si veda linuxports/howto/intro_to_networking/ISO - OSI Model (http://www.linuxports.com/howto/intro_to_networking/c4412.htm#PAGE103HTML)) il quale soppianterà IPv4 (conosciuto anche come IP). IPv4 fu progettato molto tempo fa (RFC 760 / Internet Protocol (http://www.faqs.org/rfcs/rfc760.html) del Gennaio 1980) e fin dal principio, ci sono state molte richieste per più indirizzi e migliori risorse. L'ultima RFC è RFC 2460 / Internet Protocol Version 6 Specification (http://www.faqs.org/rfcs/rfc2460.html). I cambiamenti più importanti di IPv6 sono la riprogettazione dell'intestazione (header), incluso l'aumento della dimensione degli indirizzi da 32 a 128 bit. Poichè il livello 3 OSI è responsabile del trasporto dal mittente al destinatario (end-to-end) dei pacchetti tramite l'instradamento (routing) basato sugli indirizzi, esso deve includere i nuovi indirizzi IPv6 (sorgente e destinazione) come IPv4.

Per maggiori informazioni sulla storia di IPv6 si dia uno sguardo alle vecchie RFC relative elencate ad esempio presso Guida IPv6 SWITCH / Riferimenti (http://www.switch.ch/lan/ipv6/references.html).

2.2. Storia di IPv6 su Linux

Gli anni 1992, 1993 e 1994 della storia di IPv6 sono trattati (in generale) dal seguente documento: IPv6 o IPng (IP next generation) (http://www.laynetworks.com/users/webs/IPv6.htm#CH3).

Da fare: miglior time-line, maggiori contenuti...

2.2.1. Inizio

Il primo codice relativo a IPv6 fu aggiunto al kernel Linux 2.1.8 nel Novembre 1996 da Pedro Roque. Era basato sull'API BSD:

```
diff -u --recursive --new-file v2.1.7/linux/include/linux/in6.h
¬ linux/include/linux/in6.h
--- v2.1.7/linux/include/linux/in6.h Thu Jan 1 02:00:00 1970
+++ linux/include/linux/in6.h Sun Nov 3 11:04:42 1996
@@ -0,0 +1,99 @@
+/*
+ * Types and definitions for AF_INET6
+ * Linux INET6 implementation
+ * + * Authors:
+ * Pedro Roque <******>
+ *
+ * Source:
+ * IPv6 Program Interfaces for BSD Systems
+ * <draft-ietf-ipngwg-bsd-api-05.txt>
```

Le linee mostrate sono state copiate dalla patch-2.1.8 (l'indirizzo e-mail è stato nascosto nel copia&incolla).

2.2.2. Nel mezzo

A causa della mancanza di manodopera, l'implementazione di IPv6 nel kernel non è riuscita a seguire le stesure discusse o i nuovi RFC rilasciati. Nell'Ottobre 2000, in Giappone è stato avviato un progetto chiamato USAGI (http://www.linux-ipv6.org/), il cui intento è quello di implementare tutto il supporto IPv6 mancante o sorpassato in Linux. Questo tiene traccia delle implementazioni IPv6 su FreeBSD realizzate dal progetto KAME (http://www.kame.net/). Di volta in volta essi creano delle istantanee rispetto agli attuali sorgenti "vanilla" del kernel di Linux.

2.2.3. Attuale

Sfortunatamente, la patch USAGI (http://www.linux-ipv6.org/) è così grande che gli attuali manutentori del networking di Linux non riescono ad includerla nei sorgenti di produzione del kernel della serie 2.4.x. Pertanto questa serie di kernel manca di alcune (molte) estensioni e inoltre non è conforme a tutte le attuali discussioni e RFC (si veda Statuto del gruppo di lavoro IP Versione 6 (ipv6) (http://www.ietf.org/html.charters/ipv6-charter.html)). Questo causa qualche problema di interoperabilità con gli altri sistemi operativi.

2.2.4. Futuro

USAGI (http://www.linux-ipv6.org/) sta ora facendo uso della nuova serie di kernel di sviluppo 2.5.x per inserirvi tutte le attuali estensioni. Con buone speranze i kernel della serie 2.6.x conterranno una reale implementazione di IPv6 aggiornata.

2.3. Che aspetto hanno gli indirizzi IPv6?

Come precedentemente accennato, gli indirizzi IPv6 sono lunghi 128 bit. Questa quantità di bit genera numeri decimali molto grandi che possono avere fino a 39 cifre:

```
2^128-1: 340282366920938463463374607431768211455
```

Tali numeri non sono indirizzi che possono effettivamente essere agevolmente memorizzati. Lo schema dell'indirizzo IPv6 è un insieme di bit (esattamente come IPv4, ma spesso di ciò non si è consapevoli). Pertanto una migliore notazione per così grandi numeri è l'esadecimale. In esadecimale, 4 bit (anche conosciuto come "nibble") sono rappresentati da un numero o un carattere da 0 a 9 e da A a F (10-15). Questo formato riduce la lunghezza dell'indirizzo IPv6 a 32 caratteri.

```
2^128-1: 0xffffffffffffffffffffffffffffffff
```

Questa rappresentazione non è ancora molto conveniente (possibile confusione o perdita di singole cifre esadecimali), così i progettisti di IPv6 hanno scelto un formato esadecimale con i due punti come separatori dopo ogni blocco di 16 bit. Inoltre, l'iniziale "0x" (l'indicatore usato nei linguaggi di programmazione per valori esadecimali) viene rimosso:

```
2^128-1: ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
```

Un esempio di indirizzo utilizzabile (si vedano i tipi di indirizzo più avanti) è:

```
3ffe:ffff:0100:f101:0210:a4ff:fee3:9566
```

Per semplificare, lo zero iniziale di ogni blocco di 16 bit può essere omesso:

```
3ffe:ffff:0100:f101:0210:a4ff:fee3:9566  ->
¬ 3ffe:ffff:100:f101:210:a4ff:fee3:9566
```

Una sequenza di 16 blocchi di bit contenenti solo zeri può essere rimpiazzata con "::". Ma non più di una alla volta, altrimenti non è più una rappresentazione unica.

```
3ffe:ffff:100:f101:0:0:0:1  ->  3ffe:ffff:100:f101::1
```

La riduzione più grande si può apprezzare con l'indirizzo IPv6 relativo a localhost:

```
0000:0000:0000:0000:0000:0000:0000:0001  ->  ::1
```

Esiste anche una rappresentazione (codificata in base 85) cosidetta *compatta* definita RFC 1924 / A Compact Representation of IPv6 Addresses (http://www.faqs.org/rfcs/rfc1924.html) (pubblicata il 1 Aprile 1996), mai vista da nessuna parte, probabilmente un pesce d'Aprile, ma di seguito è riportato un esempio:

```
# ipv6calc --addr_to_base85 3ffe:ffff:0100:f101:0210:a4ff:fee3:9566
Itu&-ZQ82s>J%s99FJXT
```

Info: *ipv6calc* è un programma per calcolare e convertire indirizzi IPv6 nei vari formati e può essere trovato qui: ipv6calc homepage (http://www.deepspace6.net/projects/ipv6calc.html) (Mirror (http://mirrors.bieringer.de/www.deepspace6.net/projects/ipv6calc.html)).

2.4. FAQ (Fondamentali)

2.4.1. Perchè il nome del successore di IPv4 è IPv6 e non IPv5?

In ogni intestazione IP, i primi 4 bit sono riservati per la versione del protocollo. Così teoricamente sono possibili numeri di protocollo da 0 a 15:

- 4: già utilizzato per IPv4
- 5: riservato per lo Stream Protocol (STP, RFC 1819 / Internet Stream Protocol Version 2 (http://www.faqs.org/rfcs/rfc1819.html)) (il quale non è mai stato veramente reso pubblico)

Il successivo numero libero è 6. Per cui è nato IPv6!

2.4.2. Indirizzi IPv6: perché un così grande numero di bit?

Durante la progettazione di IPv4, si pensò che 32 bit fossero sufficienti per il mondo. Guardando al passato, 32 bit sono stati sufficienti fino ad ora e può darsi che basteranno per qualche altro anno. Comunque, 32 bit non bastano per fornire in futuro, ad ogni dispositivo di rete un indirizzo globale. Si pensi ai telefoni mobili, alle auto (inclusi i dispositivi sulle loro CAN-bus), ai tostapane, ai frigoriferi, agli interruttori della luce, e cosi via ...

Per questo i progettisti hanno scelto 128 bit, 4 volte più lungo e 2^96 volte superiore rispetto all'IPv4 di oggi.

La dimensione effettivamente utilizzabile è comunque inferiore rispetto a quello che sembra. Questo perché nello schema degli indirizzi attualmente definito, 64 bit sono usati per gli identificatori di interfaccia e gli altri 64 bit sono usati per l'instradamento. Assumendo gli attuali stretti livelli di aggregazione (/48, /32, ...) è persino possibile esaurire lo spazio, ma speriamo non nel prossimo futuro.

Per maggiori informazioni si vedano anche RFC 1715 / The H Ratio for Address Assignment Efficiency (http://www.faqs.org/rfcs/rfc1715.html) e RFC 3194 / The Host-Density Ratio for Address Assignment Efficiency (http://www.faqs.org/rfcs/rfc3194.html).

2.4.3. Indirizzi IPv6: Perché un così piccolo numero di bit per un nuovo progetto?

Al momento ci sono (forse) alcune persone (conosco solo Jim Fleming...) su Internet che stanno pensando a IPv8 e IPv16, il loro progetto è ancora lontano dall'essere accettato e implementato. Nel frattempo 128 bit sono stati la migliore scelta per l'overhead dell'intestazione e il trasporto dati. Consideriamo la minima Maximum Transfer Unit (MTU) in IPv4 (576 ottetti) e in IPv6 (1280 ottetti), la lunghezza dell'intestazione di IPv4 è 20 ottetti (minima, può arrivare a 60 ottetti con le opzioni) quella di IPv6 è 48 ottetti (fissa). Questa è il 3.4 % della MTU di IPv4 e il 3.8 % della MTU di IPv6. Ciò significa che l'overhead dell'intestazione è quasi uguale. Più bit per gli indirizzi richiedono intestazioni più grandi e quindi maggiore overhead. Si consideri anche che la massima MTU su normali link (come Ethernet attualmente) è di 1500 ottetti (in casi particolari 9k ottetti utilizzando frame "Jumbo"). In definitiva, non sarebbe strutturalmente appropriato se il 10 % o 20 % dei dati trasportati in un pacchetto di livello 3 fosse usato per gli indirizzi e non per il payload.

Capitolo 3. Tipi di indirizzi

Come in IPv4, gli indirizzi IPv6 possono essere divisi in parte rete e parte host utilizzando le maschere di sottorete (subnet mask).

IPv4 ha dimostrato che in certi casi può essere utile assegnare più di un indirizzo ad una singola interfaccia, ognuno per un differente scopo (alias, multi-cast). Per rimanere estensibile in futuro, IPv6 va oltre e permette l'assegnazione di più di un indirizzo IPv6 ad una stessa interfaccia. Non ci sono attualmente limiti definiti da una RFC se non nell'implementazione dello stack IPv6 (per prevenire attacchi DoS).

Utilizzando un grande numero di bit per gli indirizzi, IPv6 definisce dei tipi basandosi sul valore di alcuni bit iniziali, che speriamo non verranno cambiati in futuro (diversamente dall'attuale IPv4 e la storia delle classi A, B, e C).

Inoltre il numero di bit è separato in una parte rete (i 64 bit più significativi) e in una parte host (i 64 bit meno significativi) per facilitare l'auto configurazione. A tal proposito, una valida URL per mostrare in dettaglio un dato indirizzo IPv6 è il Laboratorio Avanzato Gestione di Rete / Oracolo indirizzo IPv6 (http://steinbeck.ucs.indiana.edu:47401/).

3.1. Indirizzi senza un prefisso speciale

3.1.1. Indirizzo localhost

È uno speciale indirizzo per l'interfaccia di loopback, simile al "127.0.0.1"di IPv4. Con IPv6 l'indirizzo localhost è:

```
0000:0000:0000:0000:0000:0000:0000:0001
```

o compresso:

```
::1
```

I pacchetti con questo indirizzo come sorgente o destinazione non devono mai uscire dall'host che li invia.

3.1.2. Indirizzo non specificato

Questo è un indirizzo speciale come "any" o "0.0.0.0" in IPv4 . Per IPv6 è:

```
0000:0000:0000:0000:0000:0000:0000:0000
```

o:

```
::
```

Questi indirizzi sono principalmente usati/visti nel socket binding (per ogni indirizzo IPv6) o nelle tabelle di instradamento.

Nota: Questo tipo di indirizzo non può essere usato come destinazione.

3.1.3. Indirizzo IPv6 con indirizzo IPv4 incorporato

Ci sono due tipi di indirizzi IPv6 che possono contenere un indirizzo IPv4.

3.1.3.1. Indirizzo IPv6 mappato IPv4

Gli indirizzi compatibili IPv6 solo IPv4 sono a volte utilizzati/visualizzati per socket creati da un demone abilitato IPv6, ma relativo ad un indirizzo IPv4.

questi indirizzi sono definiti con un prefisso speciale di lunghezza 96 (a.b.c.d è l'indirizzo IPv4):

```
0:0:0:0:0:ffff:a.b.c.d/96
```

o nel formato compresso

```
::ffff:a.b.c.d/96
```

Per esempio, l'indirizzo IPv4 1.2.3.4 è rappresentato così:

```
::ffff:1.2.3.4
```

3.1.3.2. Indirizzo IPv6 compatibile IPv4

Utilizzato per il tunneling automatico (RFC 2893 / Transition Mechanisms for IPv6 Hosts and Routers (http://www.faqs.org/rfcs/rfc2893.html)), che sta per essere rimpiazzato da tunneling 6to4.

```
0:0:0:0:0:0:a.b.c.d/96
```

o nel formato compresso

```
::a.b.c.d/96
```

3.2. Parte rete, detta anche prefisso

I progettisti hanno definito alcuni tipi di indirizzo e ne hanno lasciati parecchi per future definizioni, attualmente sconosciute. RFC 2373 [Luglio 1998] / IP Version 6 Addressing Architecture (http://www.faqs.org/rfcs/rfc2373.html) definisce l'attuale schema di indirizzamento anche se c'è già una nuova bozza disponibile: draft-ietf-ipngwg-addr-arch-*.txt (ftp://ftp.ietf.org/internet-drafts/).

Si dia ora uno sguardo ai differenti tipi di prefisso (e quindi tipi di indirizzo):

3.2.1. Indirizzo link locale

Sono particolari indirizzi, validi soltanto nel link di un interfaccia. Usando questo indirizzo come destinazione, il pacchetto non attraverserà mai un router. È usato per comunicazioni link quali:

- c'è qualcun'altro su questo link?
- c'è qualcuno con un particolare indirizzo (per esempio per la ricerca di un router)?

Essi iniziano con (dove *"x"* è qualsiasi carattere esadecimale, normalmente *"0"*)

```
fe8x:  <- attualmente l'unico utilizzato.
fe9x:
feax:
febx:
```

Un indirizzo con questo prefisso si trova su ogni interfaccia abilitata IPv6 dopo una configurazione automatica stateless (che è il caso classico).

3.2.2. Indirizzo di tipo sito locale

Questi sono indirizzi simili a quelli specificati dalla RFC 1918 / Address Allocation for Private Internets (http://www.faqs.org/rfcs/rfc1918.html), oggi utilizzati da IPv4, con il vantaggio che chiunque usi questo tipo di indirizzo ha la possibilità di utilizzare 16 bit per gestire fino a 65536 sottoreti. È analogo alla 10.0.0.0/8 di IPv4.

Un altro vantaggio: poiché è possibile assegnare più di un indirizzo ad un'interfaccia IPv6, si può anche assegnare tale indirizzo locale in aggiunta ad uno globale.

Questo tipo di indirizzo inizia con:

```
fecx:  <- il pi&ugrave; comunemente usato.
fedx:
feex:
fefx:
```

(dove *"x"* è qualsiasi carattere esadecimale, normalmente *"0"*)

Si noti che ci sono dibattiti sul fatto di dismettere questo tipo di indirizzi poiché esistono diversi problemi. Per maggiori dettagli si veda draft-ietf-ipv6-deprecate-site-local-XY.txt (http://www.ietf.org/internet-drafts/).

A mio modesto parere, questo tipo di indirizzi sono ancora una buona scelta per prove di laboratorio.

3.2.3. Indirizzo di tipo globale "(Aggregatable) global unicast"

Attualmente, c'è un unico indirizzo di tipo globale definito (il primo progetto, chiamato "provider based" fu cestinato alcuni anni fà RFC 1884 / IP Version 6 Addressing Architecture [obsoleto] (http://www.faqs.org/rfcs/rfc1884.html), se ne possono trovare alcune tracce nei più vecchi sorgenti del kernel Linux).

Inizia con (le *x* sono caratteri esadecimali)

```
2xxx:
3xxx:
```

Nota: il prefisso "aggregabile" (aggregatable) è stato depennato dalle bozze attuali. Ci sono alcuni ulteriori sottotipi definiti, come di seguito riportato:

3.2.3.1. Indirizzi 6bone di test

Questi sono i primi indirizzi globali che sono stati definiti ed utilizzati. Iniziano con

```
3ffe:
```

Esempio:

```
3ffe:ffff:100:f102::1
```

Un particolare indirizzo 6bone di test che non sarà mai globalmente unico, inizia con

```
3ffe:ffff:
```

ed è generalmente mostrato negli esempi, perché se venissero mostrati indirizzi reali, sarebbe possibile per qualcuno fare un copia & incolla sui suoi file di configurazione. In questo modo si potrebbe causare inavvertitamente la duplicazione di un indirizzo globalmente unico. Ciò potrebbe causare seri problemi all'host originale (per esempio potrebbe ricevere pacchetti di risposta per richieste che non sono mai state inviate da esso). Tuttavia è possibile richiedere uno di questi prefissi, si veda Come far parte del 6bone (http://www.6bone.net/6bone_hookup.html). Anche alcuni tunnel brokers distribuiscono prefissi di indirizzi 6bone di test.

3.2.3.2. Indirizzi 6to4

Questi indirizzi, pensati per uno speciale meccanismo di tunneling [RFC 3056 / Connection of IPv6 Domains via IPv4 Clouds (http://www.faqs.org/rfcs/rfc3056.html) e RFC 2893 / Transition Mechanisms for IPv6 Hosts and Routers (http://www.faqs.org/rfcs/rfc2893.html)], codificano un dato indirizzo IPv4 ed una possibile sottorete. Essi iniziano con

```
2002:
```

Per esempio, per rappresentare 192.168.1.1/5:

```
2002:c0a8:0101:5::1
```

Un semplice comando di shell può aiutare a generarli a partire da indirizzi IPv4:

```
ipv4="1.2.3.4"; sla="5"; printf "2002:%02x%02x:%02x%02x:%04x::1" `echo $ipv4 | tr "." " "` $sla
```

Si veda anche tunneling con il 6to4 e informazioni sui relay router 6to4.

3.2.3.3. Assegnato dal provider per routing gerarchico

Questi indirizzi sono delegati agli Internet service provider (ISP) ed iniziano con

```
2001:
```

I prefissi ai maggiori ISP (proprietari di backbone, anche detti LIR) sono delegati dai local registries ed attualmente assegnano prefissi di lunghezza 35.

Ogni ISP minore, può ottenere un prefisso di lunghezza 48.

3.2.3.4. Indirizzi riservati per esempi e documentazione

Attualmente, due intervalli di indirizzi sono riservati per esempi e documentazione:

```
3ffe:ffff::/32
2001:0DB8::/32   EXAMPLENET-WF
```

Questi intervalli di indirizzi dovrebbero essere filtrati in base agli indirizzi di provenienza e possibilmente NON dovrebbero essere instradati sui router di confine verso Internet.

3.2.4. Indirizzi multicast

Gli indirizzi multicast sono usati per servizi ad essi relativi.

Essi iniziano sempre con (*xy* è il valore dell'ambito di applicabilità - scope)

```
ffxy:
```

Si suddividono in ambiti di applicabilità (scope) e tipi:

3.2.4.1. Multicast scope

L'ambito di applicabilità multicast (multicast scope) è un parametro che specifica la distanza massima che può percorrere un pacchetto a partire dall'entità che lo invia.

Attualmente, vengono definite le seguenti regioni (ambiti di applicabilità):

- ffx1: node-local, i pacchetti non lasciano mai il nodo.
- ffx2: link-local, i pacchetti non sono mai inoltrati dai router, in questo modo non lasciano mai il link specificato.
- ffx5: site-local, i pacchetti non lasciano mai il sito.
- ffx8: organization-local, i pacchetti non lasciano mai l'organizzazione (non così facile da implementare, la gestione deve essere effettuata dal protocollo di instradamento).
- ffxe: global scope.
- gli altri sono riservati.

3.2.4.2. Tipi multicast

Ci sono numerosi tipi già definiti/riservati (si veda RFC 2373 / IP Version 6 Addressing Architecture (http://www.faqs.org/rfcs/rfc2373.html) per i dettagli). Alcuni esempi sono:

- Indirizzo di tutti i nodi: ID = 1h, indirizza tutti gli host sul nodo locale (ff01:0:0:0:0:0:0:1) o sul link connesso (ff02:0:0:0:0:0:0:1).
- Indirizzo di tutti i router: ID = 2h, indirizza tutti i router sul nodo locale (ff01:0:0:0:0:0:0:2), sul link connesso (ff02:0:0:0:0:0:0:2), o sul sito locale (ff05:0:0:0:0:0:0:2).

3.2.4.3. Indirizzo multicast link-local nodo sollecitato

Speciale indirizzo multicast utilizzato come destinazione nella scoperta del vicinato, perché a differenza di IPv4, ARP non esiste più in IPv6.

Un esempio di questo indirizzo somiglia a

```
ff02::1:ff00:1234
```

Il prefisso utilizzato mostra che questo è un indirizzo multicast link-local. Il suffisso viene generato dall'indirizzo di destinazione. In questo esempio, un pacchetto dovrebbe essere inviato all'indirizzo "fe80::1234" ma lo stack di rete non conosce l'attuale indirizzo di livello 2 (MAC). I 104 bit più significativi vengono rimpiazzati con "ff02:0:0:0:0:1:ff00::/104" ed i 24 bit meno significativi vengono lasciati inalterati. Questo indirizzo viene quindi usato 'on-link' per trovare il corrispondente nodo che deve inviare una risposta contenente il suo indirizzo di livello 2 (MAC).

3.2.5. Indirizzi anycast

Gli indirizzi anycast sono indirizzi speciali e vengono utilizzati per trattare cose come il server DNS più vicino, il server DHCP più vicino o simili gruppi dinamici. Questi vengono presi dallo spazio degli indirizzi unicast (aggregatable global o site-local per il momento). Il meccanismo anycast (dal punto di vista del client) sarà gestito dai protocolli di routing dinamici.

Nota: gli indirizzi anycast non possono essere usati come indirizzo del mittente, ma sono validi soltanto come indirizzi di destinazione.

3.2.5.1. Indirizzo anycast subnet-router

Un semplice esempio di un indirizzo anycast è il tipo subnet-router. Assumendo che un nodo abbia il seguente indirizzo globale IPv6:

```
3ffe:ffff:100:f101:210:a4ff:fee3:9566/64  <- Indirizzo del nodo
```

L'indirizzo anycast subnet-router sarà creato svuotando completamente il suffisso (i 64 bit meno significativi):

```
3ffe:ffff:100:f101::/64  <- indirizzo anycast subnet-router
```

3.3. Tipi di indirizzo (parte host)

Per problemi di configurazione automatica e mobilità è stato deciso di usare, nella maggior parte degli attuali tipi, i 64 bit meno significativi come parte host dell'indirizzo. Quindi ogni singola sottorete può possedere un grande numero di indirizzi.

Questa parte host può essere considerata in maniera diversa:

3.3.1. Generata automaticamente (anche detta stateless)

Con la configurazione automatica, la parte host dell'indirizzo viene calcolata convertendo, con il metodo EUI-64, l'indirizzo MAC di un interfaccia (se disponibile) in un indirizzo IPv6 unico. Se nessun indirizzo MAC è disponibile per questo dispositivo (come accade ad esempio con le periferiche virtuali), viene usato qualcos'altro (come l'indirizzo IPv4 o l'indirizzo MAC di una interfaccia fisica).

Si consideri ancora il primo esempio

```
3ffe:ffff:100:f101:210:a4ff:fee3:9566
```

qui

```
210:a4ff:fee3:9566
```

è la parte host generata a partire dall'indirizzo MAC della scheda di rete

```
00:10:A4:E3:95:66
```

utilizzando il Tutorial IEEE EUI-64 (http://standards.ieee.org/regauth/oui/tutorials/EUI64.html) sviluppato per gli identificatori EUI-48.

3.3.1.1. Problemi di Privacy con gli indirizzi generati automaticamente ed una soluzione

Poiché la parte host viene "calcolata automaticamente" è globalmente unica (a meno che un produttore di schede di rete usi lo stesso indirizzo MAC per più di una NIC) è possibile effettuare il tracciamento del client quando non viene utilizzato alcun proxy.

Ciò è un problema noto ed è stata definita una soluzione: la privacy extension, descritta in RFC 3041 / Privacy Extensions for Stateless Address Autoconfiguration in IPv6 (http://www.faqs.org/rfcs/rfc3041.html) (c'è anche già una nuova bozza disponibile: draft-ietf-ipngwg-temp-addresses-*.txt (ftp://ftp.ietf.org/internet-drafts/)). Utilizzando un valore casuale ed uno statico viene generato di volta in volta un nuovo suffisso. Nota: ciò è ragionevole soltanto per comunicazioni uscenti da client, ma non è utile nel caso di server noti.

3.3.2. Impostata manualmente

Per i server probabilmente è più facile ricordare semplici indirizzi e anche questo può essere agevolato. È possibile assegnare un indirizzo IPv6 addizionale ad un'interfaccia, per esempio

```
3ffe:ffff:100:f101::1
```

Per i suffissi manuali come "::1" mostrato nell'esempio precedente, è richiesto che il settimo bit più significativo sia impostato a 0 (il bit universale/locale degli identificatori generati automaticamente). Anche altre combinazioni di bit (altrimenti non scelte) sono riservate per indirizzi anycast.

3.4. Lunghezze del prefisso per il routing

Nella prima fase della progettazione fu pianificato di usare un approccio al routing completamente gerarchico per ridurre al massimo la dimensione delle tabelle di routing. Il motivo di questo approccio fu il numero delle attuali voci di routing di IPv4 nei router principali (> 104.000 nel Maggio 2001), riducendo la necessità di memoria nei router hardware (basati su ASIC "Application Specified Integrated Circuit") per memorizzare la tabella di routing ed incrementare la velocità (un numero di voci inferiore consente ricerche più veloci).

Il punto di vista attuale è che il routing debba essere sviluppato in maniera gerarchica principalmente per reti con un solo service provider. Con più di un ISP ciò non è possibile e si è soggetti ad una questione chiamata multi-homing (informazioni sul multi-homing: Procider-Internal Aggregation based on Geography to Support Multihoming in IPv6 (http://www.ietf.org/internet-drafts/draft-van-beijnum-multi6-isp-int-aggr-00.txt); GAPI: A Geographically Aggregatable Provider Independent Address Space to Support Multihoming in IPv6 (http://www.ietf.org/internet-drafts/draft-by-multi6-gapi-00.txt); Extension Header for Site-Multi-homing support (http://www.ietf.org/internet-drafts/draft-bagnulo-multi6-mhexthdr-00.txt); IPv6 Multihoming Solutions (http://arneill-py.sacramento.ca.us/ipv6mh/)).

3.4.1. Lunghezze del prefisso ("maschera di rete" - netmask)

Come per IPv4, è il percorso della rete instradabile affinché il routing abbia luogo. Poiché la notazione standard per indicare la maschera di rete con 128 bit non è agevole, i progettisti hanno utilizzato lo schema IPv4 Classless Inter Domain Routing (CIDR, RFC 1519 / Classless Inter-Domain Routing (http://www.faqs.org/rfcs/rfc1519.html)) che indica il numero di bit dell'indirizzo IP da utilizzare nelil'instradamento . Tale notazione è anche detta notazione "slash".

Un esempio:

```
3ffe:ffff:100:1:2:3:4:5/48
```

Questa notazione verrà espansa in:

- Rete:

```
3ffe:ffff:0100:0000:0000:0000:0000:0000
```

- Maschera di rete:

```
ffff:ffff:ffff:0000:0000:0000:0000:0000
```

3.4.2. Corrispondenza con una route

In circostanze normali (senza QoS) una ricerca in una tabella di instradamento dà come risultato la route con il numero di bit dell'indirizzo più significativo, cioè viene scelta la prima route che ha la lunghezza del prefisso più grande corrispondente a quella dell'indirizzo di destinazione considerato.

Per esempio, se una tabella di instradamento mostra le seguenti voci (la lista non è completa):

```
3ffe:ffff:100::/48     ::            U  1 0 0 sit1
2000::/3               ::192.88.99.1 UG 1 0 0 tun6to4
```

I seguenti indirizzi di destinazione dei pacchetti IPv6 saranno instradati attraverso i dispositivi elencati

```
3ffe:ffff:100:1:2:3:4:5/48  ->  instradato attraverso il dispositivo sit1
3ffe:ffff:200:1:2:3:4:5/48  ->  instradato attraverso il dispositivo tun6to4
```

Capitolo 4. Verificare se il sistema è pronto per IPv6

Prima di poter cominciare a usare IPv6 su di un host Linux, bisogna verificare se il sistema è pronto per IPv6. Potrebbe essere necessario effettuare qualche operazione per abilitarlo.

4.1. Kernel pronto per IPv6

Le moderne ditribuzioni Linux già contengono kernel pronti per IPv6; la funzionalità IPv6 è generalmente compilata come modulo ma è possibile che questo non venga caricato automaticamente all'avvio del sistema.

Si veda la pagina IPv6+Linux-Status-Distribution (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-distributions.html) per le informazioni più aggiornate.

Nota: non si dovrebbero più usare i kernel della serie 2.2.x perché non sono più aggiornati per IPv6.

4.1.1. Controllare il supporto IPv6 nel kernel corrente

Per controllare se il kernel corrente supporta IPv6, si dia un occhiata al proprio filesystem /proc. Deve esistere la seguente voce:

```
/proc/net/if_inet6
```

Un semplice test automatico potrebbe essere:

```
# test -f /proc/net/if_inet6 && echo "Running kernel is IPv6 ready"
```

Se esso fallisce è ovvio che il modulo IPv6 non è stato caricato.

4.1.2. Provare a caricare il modulo IPv6

Si può provare a caricare il modulo IPv6 eseguendo

```
# modprobe ipv6
```

Se tale comando ha successo, questo modulo dovrebbe essere caricato; lo si può controllare con il seguente comando:

```
# lsmod |grep -w 'ipv6' && echo "IPv6 module successfully loaded"
```

Il comando precedente dovrebbe quindi funzionare con successo.

Nota: lo scaricamento del modulo non è attualmente supportato e può causare, in alcune circostanze, il crash del kernel.

4.1.2.1. Caricamento automatico del modulo IPv6

È possibile caricare automaticamente il modulo IPv6 su richiesta. Basta aggiungere la seguente linea nel file di configurazione del loader dei moduli del kernel (normalmente /etc/modules.conf o /etc/conf.modules):

```
alias net-pf-10 ipv6  # carica automaticamente su richiesta il modulo IPv6
```

È anche possibile disabilitare il caricamento automatico del modulo usando la seguente linea

```
alias net-pf-10 off   # disabilita il caricamento automatico del modulo IPv6
```

Nota addizionale: nei kernel più recenti (serie 2.5 e successive), il meccanismo del caricatore dei moduli è stato modificato. Il nuovo file di configurazione è /etc/modprobe.conf invece di /etc/modules.conf, ma è comunque disponibile uno script per la traduzione. Per ulteriori dettagli si veda module-init-tool (http://his.luky.org/ftp/mirrors/linux/kernel/people/rusty/modules/).

4.1.3. Compilare il kernel con le funzionalità IPv6

Se entrambe le operazioni precedenti avessero esito negativo e il kernel considerato non avesse il supporto IPv6, si hanno le seguenti possibilità:

- Aggiornare la propria distribuzione con una versione avente il supporto IPv6 incluso (raccomandato per i principianti), si veda IPv6+Linux-Status-Distribution (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-distributions.html)
- Compilare un nuovo kernel "vanilla" (facile, se si conoscono le opzioni necessarie)
- Ricompilare i sorgenti del kernel della propria distribuzione Linux (a volte non così facile)
- Compilare un kernel con le estensioni USAGI

Se si decide di compilare un kernel, si dovrebbero avere precedenti esperienze in questa operazione e si dovrebbe leggere il Linux Kernel HOWTO (http://www.tldp.org/HOWTO/Kernel-HOWTO.html).

Un confronto abbastana aggiornato tra il kernel puro (vanilla) e quello con le estensioni USAGI è disponibile su IPv6+Linux-Status-Kernel (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-kernel.html).

4.1.3.1. Compilazione di un kernel "vanilla"

Consigli più dettagliati sulla compilazione di un kernel con IPv6 possono essere reperiti, per esempio, su IPv6-HOWTO-2#kernel (http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/IPv6-HOWTO-2.html#kernel).

Nota: quando possibile si dovrebbero usare kernel della serie 2.4.x o superiori, poiché il supporto IPv6, nella serie 2.2.x non è così aggiornato e necessita di alcune patch per ICMPv6 e per il supporto 6to4 (che si possono trovare su kernel series 2.2.x IPv6 patches (ftp://ftp.bieringer.de/pub/linux/IPv6/kernel)).

4.1.3.2. Compilazione del kernel con le estensioni USAGI

Come per il kernel vanilla, questa operazione è raccomandata solo agli utenti esperti che abbiano già familiarità con IPv6 e la compilazione del kernel. Si veda anche il progetto USAGI / FAQ (http://www.linux-ipv6.org/faq.html) e

Ottenere il miglior supporto IPv6 con Linux (Articolo) (http://www.deepspace6.net/docs/best_ipv6_support.html) (Mirror (http://mirrors.bieringer.de/www.deepspace6.net/docs/best_ipv6_support.html)).

4.1.4. Dispositivi di rete pronti per IPv6

Non tutti i dispositivi di rete esistenti hanno già (o avranno mai) la capacità di trasportare pacchetti IPv6. Lo stato attuale è reperibile presso IPv6+Linux-status-kernel.html#transport (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-kernel.html#transport).

Una questione importante è che a causa della struttura dei livelli di rete nell'implementazione del kernel, un pacchetto IPv6 non viene effettivamente riconosciuto dal suo numero di intestazione IP (6 invece di 4) ma piuttosto dal numero di protocollo di trasporto di livello 2. Pertanto ogni protocollo di trasporto che non usa tale numero di protocollo, non può inviare pacchetti IPv6. Nota: il pacchetto viene comunque trasportato sul collegamento, ma dal lato ricevente il recapito non funziona (come si può vedere, per esempio, con tcpdump).

4.1.4.1. Collegamenti che non saranno mai "idonei IPv6"

- Serial Line IP (SLIP, RFC 1055 / SLIP (http://www.faqs.org/rfcs/rfc1055.html)), attualmente dovrebbe essere più correttamente detto SLIPv4, dispositivo chiamato: slX
- Parallel Line IP (PLIP), simile a SLIP, nome del dispositivo: plipX
- ISDN con incapsulazione *rawip*, nome del dispositivo: isdnX

4.1.4.2. Collegamenti attualmente conosciuti come "idonei IPv6 ma non supportati"

- ISDN con incapsulazione *syncppp*, nome del dispositivo: ipppX (edizione di progetto dell'ipppd, sarà assorbito nel più generale livello PPP nel kernel serie 2.5.x).

4.2. Strumenti di configurazione di rete pronti per IPv6

Anche se si sta usando un kernel con IPv6, non si andrà molto lontano se non si hanno strumenti per la configurazione. Esistono diversi pacchetti che possono farlo.

4.2.1. Pacchetto net-tools

Il pacchetto net-tools include alcuni strumenti come ifconfig e route, che permettono di configurare IPv6 su di un interfaccia. Si guardi all'output di ifconfig -? o route -?, se viene mostrato qualcosa come IPv6 o inet6 allora il programma è predisposto IPv6.

Il controllo per ifconfig può essere effettuato tramite:

```
# /sbin/ifconfig -? 2>& 1|grep -qw 'inet6' && echo "utility 'ifconfig' is IPv6-ready"
```

Analogamente per route:

```
# /sbin/route -? 2>& 1|grep -qw 'inet6' && echo "utility 'route' is IPv6-ready"
```

4.2.2. Pacchetto iproute

Alexey N. Kuznetsov (attualmente un manutentore del codice di networking di Linux) ha creato un set di strumenti che configurano la rete attraverso il dispositivo netlink. Usando questi strumenti si hanno maggiori funzionalità rispetto a quelle fornite da net-tools ma non sono documentati granché e non sono indicati per i deboli di cuore.

```
# /sbin/ip 2>&1 |grep -qw 'inet6' && echo "utility 'ip' is IPv6-ready"
```

Se il programma /sbin/ip non viene trovato, si raccomanda caldamente di installare iproute.

- Lo si può prendere dalla propria distribuzione Linux (se lo contiene)
- Si può scaricare il tar-ball e ricompilarlo: Sorgente originale FTP (ftp://ftp.inr.ac.ru/ip-routing/) e mirror (mancante)
- È possibile cercare il pacchetto RPM appropriato presso RPMfind/iproute (http://rpmfind.net/linux/rpm2html/search.php?query=iproute) (talvolta è raccomandata la ricompilazione del pacchetto SRPMS)

4.3. Programmi di test/debug pronti per IPv6

Dopo aver preparato il proprio sistema per IPv6, lo si può usare per le comunicazioni di rete. Prima di tutto si dovrebbe imparare come esaminare i pacchetti IPv6 con uno sniffer. Ciò è caldamente raccomandato poiché in caso di problemi può aiutare a fornire una diagnosi molto velocemente.

4.3.1. Ping IPv6

Questo programma è normalmente incluso nel pacchetto *iputils*. È progettato per semplici test di trasporto: esso invia pacchetti echo-request ICMPv6 ed attende pacchetti echo-replay ICMPv6.

Utilizzo

```
# ping6 <hostwithipv6address>
# ping6 <ipv6address>
# ping6 [-I <device>] <link-local-ipv6address>
```

Esempio

```
# ping6 -c 1 ::1
PING ::1(::1) from ::1 : 56 data bytes
64 bytes from ::1: icmp_seq=0 hops=64 time=292 usec
--- ::1 ping statistics ---
```

```
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max/mdev = 0.292/0.292/0.292/0.000 ms
```

Suggerimento: ping6 ha bisogno dell'accesso raw al socket e quindi dei permessi di root. In questo modo se gli utenti (non-root) non possono usare ping6 ci possono essere due problemi:

1. ping6 non è nel path degli utenti (probabilmente perché ping6 di solito si trova in /usr/sbin) -> si aggiunga al path (non molto raccomandato)
2. ping6 non viene eseguito correttamente, generalmente a causa della mancanza dei permessi di root -> chmod u+s /usr/sbin/ping6

4.3.1.1. Specificare l'interfaccia per il ping IPv6

Usando indirizzi di tipo link-local per un ping IPv6, il kernel non sa attraverso quale dispositivo (fisico o virtuale) deve inviare il pacchetto - ogni dispositivo ha un indirizzo di tipo link-local. Il tentativo darà come risultato il seguente messaggio di errore:

```
# ping6 fe80::212:34ff:fe12:3456
connect: Invalid argument
```

In questo caso bisogna specificare anche l'interfaccia come mostrato qui:

```
# ping6 -I eth0 -c 1 fe80::2e0:18ff:fe90:9205
PING fe80::212:23ff:fe12:3456(fe80::212:23ff:fe12:3456) from
¬ fe80::212:34ff:fe12:3478 eth0: 56 data bytes
64 bytes from fe80::212:23ff:fe12:3456: icmp_seq=0 hops=64 time=445 usec
--- fe80::2e0:18ff:fe90:9205 ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss round-trip
¬ min/avg/max/mdev = 0.445/0.445/0.445/0.000 ms
```

4.3.1.2. Ping6 verso indirizzi multicast

Un meccanismo interessante per rilevare host attivi IPv6 su di un link consiste nell'effettuare ping6 verso l'indirizzo multicast link-local di tutti i nodi (all-node):

```
# ping6 -I eth0 ff02::1
PING ff02::1(ff02::1) from fe80:::2ab:cdff:feef:0123 eth0: 56 data bytes
64 bytes from ::1: icmp_seq=1 ttl=64 time=0.104 ms
64 bytes from fe80::212:34ff:fe12:3450: icmp_seq=1 ttl=64 time=0.549 ms (DUP!)
```

Diversamente da IPv4, dove le risposte ad un ping verso l'indirizzo di broadcast possono essere disabilitate, in IPv6 questa caratteristica non può essere disabilitata se non per mezzo del firewalling locale IPv6.

4.3.2. Traceroute IPv6

Questo programma normalmente è incluso nel pacchetto *iputils*. È simile al traceroute IPv4. Di seguito è riportato un esempio:

```
# traceroute6 www.6bone.net
traceroute to 6bone.net (3ffe:b00:c18:1::10) from 3ffe:ffff:0000:f101::2, 30
¬ hops max, 16 byte packets
 1 localipv6gateway (3ffe:ffff:0000:f101::1) 1.354 ms 1.566 ms 0.407 ms
 2 swi6T1-T0.ipv6.switch.ch (3ffe:2000:0:400::1) 90.431 ms 91.956 ms 92.377 ms
 3 3ffe:2000:0:1::132 (3ffe:2000:0:1::132) 118.945 ms 107.982 ms 114.557 ms
 4 3ffe:c00:8023:2b::2 (3ffe:c00:8023:2b::2) 968.468 ms 993.392 ms 973.441 ms
 5 3ffe:2e00:e:c::3 (3ffe:2e00:e:c::3) 507.784 ms 505.549 ms 508.928 ms
 6 www.6bone.net (3ffe:b00:c18:1::10) 1265.85 ms * 1304.74 ms
```

Nota: diversamente da alcune moderne versioni del traceroute IPv4, le quali possono utilizzare pacchetti echo-request ICMPv4 come pure pacchetti UDP (il default), l'attuale traceroute IPv6 è in grado di inviare soltanto pacchetti UDP. Come probabilmente già si saprà, i pacchetti echo-request ICMP sono più accettati dai firewall o dalle ACL sui router rispetto ai pacchetti UDP.

4.3.3. Tracepath6

Questo programma normalmente è incluso nel pacchetto *iputils*. È simile a traceroute6 e serve a tracciare il percorso per una data destinazione scoprendo anche il relativo valore di MTU. Di seguito si può vedere un esenpio:

```
# tracepath6 www.6bone.net
 1?: [LOCALHOST] pmtu 1480
 1: 3ffe:401::2c0:33ff:fe02:14 150.705ms
 2: 3ffe:b00:c18::5 267.864ms
 3: 3ffe:b00:c18::5 asymm 2 266.145ms pmtu 1280
 3: 3ffe:3900:5::2 asymm 4 346.632ms
 4: 3ffe:28ff:ffff:4::3 asymm 5 365.965ms
 5: 3ffe:1cff:0:ee::2 asymm 4 534.704ms
 6: 3ffe:3800::1:1 asymm 4 578.126ms !N
Resume: pmtu 1280
```

4.3.4. Tcpdump IPv6

Su Linux, tcpdump è lo strumento più utilizzato per la cattura dei pacchetti. Di seguito sono riportati alcuni esempi. Il supporto IPv6 è incluso nelle attuali release della versione 3.6.

tcpdump permette l'utilizzo di espressioni per filtrare i pacchetti, minimizzando il numero di quelli indesiderati:

- icmp6: filtra il traffico nativo ICMPv6
- ip6: filtra il traffico nativo IPv6 (ICMPv6 incluso)
- proto ipv6: filtra il traffico tunnel IPv6-in-IPv4

- not port ssh: sopprime la visualizzazione dei pacchetti SSH durante l'esecuzione di tcpdump in una sessione SSH remota

Anche alcune opzioni della linea di comando sono molto utili per catturare e visualizzare una maggior quantià di informazioni relative ad un pacchetto, per lo più interessanti per scavare a fondo nei pacchetti ICMPv6:

- "-s 512": aumenta la lunghezza della cattura di un pacchetto a 512 byte
- "-vv": output veramente verboso
- "-n": non fa la risoluzione dei nomi (indirizzo->nome), utile se la risoluzione DNS inversa non funziona bene

4.3.4.1. Ping IPv6 verso l'indirizzo nativo 3ffe:ffff:100:f101::1 su di un link locale

```
# tcpdump -t -n -i eth0 -s 512 -vv ip6 or proto ipv6
tcpdump: listening on eth0
3ffe:ffff:100:f101:2e0:18ff:fe90:9205 > 3ffe:ffff:100:f101::1: icmp6: echo
¬ request (len 64, hlim 64)
3ffe:ffff:100:f101::1 > 3ffe:ffff:100:f101:2e0:18ff:fe90:9205: icmp6: echo
¬ reply (len 64, hlim 64)
```

4.3.4.2. Ping IPv6 verso 3ffe:ffff:100::1 instradato attraverso un tunnel IPv6-in-IPv4

1.2.3.4 e 5.6.7.8 sono le estremità del tunnel (tutti gli indirizzi sono a puro titolo di esempio)

```
# tcpdump -t -n -i ppp0 -s 512 -vv ip6 or proto ipv6
tcpdump: listening on ppp0
1.2.3.4 > 5.6.7.8: 2002:ffff:f5f8::1 > 3ffe:ffff:100::1: icmp6: echo request
¬ (len 64, hlim 64) (DF) (ttl 64, id 0, len 124)
5.6.7.8 > 1.2.3.4: 3ffe:ffff:100::1 > 2002:ffff:f5f8::1: icmp6: echo reply (len
¬ 64, hlim 61) (ttl 23, id 29887, len 124)
1.2.3.4 > 5.6.7.8: 2002:ffff:f5f8::1 > 3ffe:ffff:100::1: icmp6: echo request
¬ (len 64, hlim 64) (DF) (ttl 64, id 0, len 124)
5.6.7.8 > 1.2.3.4: 3ffe:ffff:100::1 > 2002:ffff:f5f8::1: icmp6: echo reply (len
¬ 64, hlim 61) (ttl 23, id 29919, len 124)
```

4.4. Programmi pronti per IPv6

Le attuali distribuzioni già contengono i più importanti client e server abilitati per IPv6. Si veda prima di tutto su IPv6+Linux-Status-Distribution (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-distributions.html). Si può anche vedere IPv6 &Linux - Stato attuale - Applicazioni (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-apps.html) per controllare se il programma di interesse è già stato portato a IPv6 ed è utilizzabile con Linux. Per i programmi di uso comune ci sono alcuni suggerimenti disponibili presso IPv6 &Linux - HowTo - Part 3 (http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/IPv6-HOWTO-3.html) e IPv6 &Linux - HowTo - Part 4 (http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/IPv6-HOWTO-4.html).

4.5. Programmi client pronti per IPv6 (selezione)

Per far girare i test mostrati di seguito, è necessario che il sistema sia abilitato per IPv6 ed alcuni esempi mostrano indirizzi che sono raggiungibili solo se è disponibile una connessione al 6bone.

4.5.1. Controllare il DNS per risolvere indirizzi IPv6

A causa di aggiornamenti di sicurezza negli ultimi anni, ogni server Domain Name System (DNS) deve far girare software più recente che è in grado di gestire l'indirizzamento (intermedio) IPv6 di tipo AAAA (il più recente, chiamato A6 non è ancora di uso comune al momento poiché è supportato solo a partire da BIND9 o superiori e il supporto del dominio radice IP6.ARPA non esiste). Un semplice test per vedere se il sistema utilizzato può risolvere indirizzi IPV6 è

```
# host -t AAAA www.join.uni-muenster.de
```

e dovrebbe mostrare qualcosa del tipo:

```
www.join.uni-muenster.de. is an alias for ns.join.uni-muenster.de.
ns.join.uni-muenster.de. has AAAA address 3ffe:400:10:100:201:2ff:feb5:3806
```

4.5.2. Client telnet pronti per IPv6

Client telnet per IPv6 sono disponibili. Un semplice test può essere fatto con

```
$ telnet 3ffe:400:100::1 80
Trying 3ffe:400:100::1...
Connected to 3ffe:400:100::1.
Escape character is '^]'.
HEAD / HTTP/1.0
HTTP/1.1 200 OK
Date: Sun, 16 Dec 2001 16:07:21
GMT Server: Apache/2.0.28 (Unix)
Last-Modified: Wed, 01 Aug 2001 21:34:42 GMT
ETag: "3f02-a4d-b1b3e080"
Accept-Ranges: bytes
Content-Length: 2637
Connection: close
Content-Type: text/html; charset=ISO-8859-1
Connection closed by foreign host.
```

Se il client non gestisce l'indirizzo IPv6 e visualizza qualcosa del tipo "cannot resolve hostname" allora non è abilitato per IPv6.

4.5.3. Client ssh per IPv6

4.5.3.1. openssh

Le attuali versioni di openssh sono abilitate per IPv6. A seconda della configurazione pre-compilazione ci sono due comportamenti.

- --without-ipv4-default: il client tenta prima automaticamente una connessione IPv6 e se non funziona prova con IPv4
- --with-ipv4-default: la connessione di default è IPv4, la connessione IPv6 deve essere forzata come mostra il seguente esempio

```
$ ssh -6 ::1
user@::1's password: ******
[user@ipv6host user]$
```

Se il client non accetta l'opzione "-6" allora non è abilitato per IPv6, come la maggior parte dei pacchetti ssh versione 1.

4.5.3.2. ssh.com

Anche il client e il server SSH di SSH.com sono predisposti per IPv6 e sono gratutiti per tutte le macchine Linux e FreeBSD indifferentemente per uso personale o commerciale.

4.5.4. Browser web per IPv6

L'attuale stato dei browser web abilitati per IPv6 è reperibile presso IPv6+Linux-status-apps.html#HTTP (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-apps.html#HTTP).

La maggior parte di questi hanno dei problemi che non sono stati ancora risolti

1. Se si usa un proxy impostato soltanto per IPv4, le richieste IPv6 verranno inviate ma il proxy non riuscirà a comprenderle e quindi si avrà un errore. Soluzione: aggiornare il software del proxy (si veda più avanti).
2. Le impostazioni automatiche per il proxy (*.pac) non possono essere estese per gestire le richieste IPv6 in modo diverso da IPv4 (per esempio non usare il proxy) a causa della loro natura (sono in Java-script e cablati nel sorgente, come si può vedere nel codice sorgente di Maxilla).

Inoltre le vecchie versioni dei browser non sono in grado di interpretare le URL codificate con indirizzi IPv6 come http://[3ffe:400:100::1]/ (questa URL funziona solo con un browser abilitato per IPv6!).

Un rapido test consiste nel provare l'URL sopra mostrata con il browser, senza utilizzare nessun proxy.

4.5.4.1. URL per provare

Un buon punto di partenza per navigare con IPv6 è http://www.kame.net/. Se la tartaruga presente sulla pagina è animata, la connessione avviene tramite IPv6, altrimenti la tartaruga è immobile.

4.6. Programmi server IPv6

In questa parte dell'HOWTO sono menzionati problemi più specifici dei client. Quindi suggerimenti per i server abilitati per IPv6 come sshd, httpd, telnetd, ecc.. sono mostrati più avanti in Suggerimenti per demoni IPv6.

4.7. FAQ (verificare se il sistema è pronto per IPv6)

4.7.1. Utilizzo degli strumenti

4.7.1.1. D: Non riesco a fare ping6 a indirizzi link-local

Messaggio di errore: "*connect: Invalid argument*"

Il kernel non sa quale link fisico o virtuale si vuole usare per inviare pacchetti ICMPv6. Qindi viene visualizzato questo messaggio di errore.

Soluzione: Specificare l'interfaccia: "ping6 -I eth0 fe80::2e0:18ff:fe90:9205", si veda anche utilizzo del programma ping6.

4.7.1.2. D: Non posso fare ping6 o traceroute6 come utente (non root)

Messaggio di errore: "*icmp socket: Operation not permitted*"

Questi strumenti creano particolari pacchetti ICMPv6 e li inviano. Ciò viene fatto utilizzando nel kernel i socket raw, però questi possono essere utilizzati solo dall'utente "root". Perciò gli utenti normali ottengono tali messaggi di errore.

Soluzione: se è effettivamente necessario che tutti gli utenti possano usare questi strumenti, si può aggiungere il bit "suid" per mezzo del comando "chmod u+s /path/to/program", si veda anche utilizzo del programma ping6. Se non tutti gli utenti devono utilizzarli, si può cambiare il gruppo proprietario dei programmi, impostandolo per esempio a "wheel". Quindi si aggiungano gli utenti privilegiati a tale gruppo e si rimuova il bit di esecuzione per tutti gli altri utenti con il comando "chmod o-rwx /path/to/program". Oppure si può configurare "sudo" per abilitare la propria policy di sicurezza.

Capitolo 5. Configurazione delle interfacce

5.1. I vari dispositivi di rete

Su un nodo ci possono esssere differenti dispositivi di rete, che si possono raggruppare nlle seguenti classi

- Fisici, come eth0, tr0
- Virtuali, come ppp0, tun0, tap0, sit0, isdn0, ippp0

5.1.1. Dispositivi fisici

Le interfacce fisiche come Ethernet o Token-Ring sono quelle classiche e non richiedono trattamenti particolari.

5.1.2. Dispositivi virtuali

Le interfacce virtuali hanno sempre bisogno di un particolare supporto.

5.1.2.1. Interfacce tunnel IPv6-in-IPv4

Queste interfacce sono normalmente chiamate sit*x*. Il nome *sit* è un'abbreviazione per Simple Internet Transition. Questo tipo di dispositivo ha la capacità di incapsulare pacchetti IPv6 all'interno di pacchetti IPv4 e di inviarli all'estremo opposto di un tunnel.

sit0 ha un significato speciale e non può essere usata per tunnel dedicati.

5.1.2.2. Interfacce PPP

Le interfacce PPP prendono la loro capacità IPv6 da un demone PPP abilitato per IPv6.

5.1.2.3. Interfacce ISDN HDLC

La capacità IPv6 per l'HDLC con incapsulamento IP è già inclusa nel kernel.

5.1.2.4. Interfacce ISDN PPP

Le interfacce ISDN PPP (ippp) presenti nel kernel, non sono abilitate per IPv6. Inoltre l'abilitazione non è stata pianificata poiché nel kernel 2.5.+ esse saranno rimpiazzate da un più generico livello di interfaccia ppp.

5.1.2.5. SLIP + PLIP

Come menzionato precedentemente, queste interfacce non supportano il trasporto IPv6 (l'invio è OK ma la consegna al destinatario non funziona).

5.1.2.6. Dispositivo ether-tap

I dispositivi ether-tap sono abilitati per IPv6 ed inoltre supportano la configurazione stateless. Per il loro utilizzo è necessario il caricamento preventivo del modulo "ethertap".

5.1.2.7. Dispositivi tun

Non ancora provati personalmente.

5.1.2.8. ATM

01/2002: Non sono attualmente supportati dai kernel vanilla, ma soltanto dalle estensioni USAGI.

5.1.2.9. Altre

Ho per caso dimenticato qualche interfaccia?...

5.2. Attivazione/disattivazione delle interfacce

Possono essere usati due metodi per attivare/disattivare le interfacce.

5.2.1. Uso di "ip"

Utilizzo:

```
# ip link set dev <interface> up
# ip link set dev <interface> down
```

Esempio:

```
# ip link set dev eth0 up
# ip link set dev eth0 down
```

5.2.2. Uso di "ifconfig"

Utilizzo:

```
# /sbin/ifconfig <interface> up
# /sbin/ifconfig <interface> down
```

Esempio:

```
# /sbin/ifconfig eth0 up
```

```
# /sbin/ifconfig eth0 down
```

Capitolo 6. Configurazione degli indirizzi IPv6

Esistono vari modi per configurare un indirizzo IPv6 su di un'interfaccia. Si può usare "ifconfig" o "ip".

6.1. Visualizzare gli indirizzi IPv6 esistenti

Prima di tutto si deve controllare se e quali indirizzi IPv6 sono già configurati (forse automaticamente durante una configurazione automatica stateless).

6.1.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 addr show dev <interface>
```

Esempio per un host con configurazione statica:

```
# /sbin/ip -6 addr show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP&gt; mtu 1500 qdisc pfifo_ fast qlen 100
inet6 fe80::210:a4ff:fee3:9566/10 scope link
inet6 3ffe:ffff:0:f101::1/64 scope global
inet6 fec0:0:0:f101::1/64 scope site
```

Esempio per un host autoconfigurato

di seguito sono riportati alcuni indirizzi IPv6 configurati automaticamente ed il loro tempo di vita.

```
# /sbin/ip -6 addr show dev eth0
3: eth0: <BROADCAST,MULTICAST,PROMISC,UP&gt; mtu 1500 qdisc pfifo_fast qlen
¬ 100
inet6 2002:d950:f5f8:f101:2e0:18ff:fe90:9205/64 scope global dynamic
valid_lft 16sec preferred_lft 6sec
inet6 3ffe:400:100:f101:2e0:18ff:fe90:9205/64 scope global dynamic
valid_lft 2591997sec preferred_lft 604797sec inet6 fe80::2e0:18ff:fe90:9205/10
¬ scope link
```

6.1.2. Uso di "ifconfig"

Utilizzo:

```
# /sbin/ifconfig <interface>
```

Esempio (l'output è stato filtrato con grep per mostrare solo gli indirizzi IPv6). Qui si possono vedere differenti indirizzi IPv6 con differenti scope.

```
# /sbin/ifconfig eth0 |grep "inet6 addr:"
```

```
inet6 addr: fe80::210:a4ff:fee3:9566/10 Scope:Link
inet6 addr: 3ffe:ffff:0:f101::1/64 Scope:Global
inet6 addr: fec0:0:0:f101::1/64 Scope:Site
```

6.2. Aggiungere un indirizzo IPv6

L'aggiunta di un indirizzo IPv6 è simile al meccanismo degli indirizzi "IP ALIAS" nelle interfacce Linux IPv4.

6.2.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 addr add <ipv6address>/<prefixlength> dev <interface>
```

Esempio:

```
# /sbin/ip -6 addr add 3ffe:ffff:0:f101::1/64 dev eth0
```

6.2.2. Uso di "ifconfig"

Utilizzo:

```
# /sbin/ifconfig <interface> inet6 add <ipv6address>/<prefixlength>
```

Esempio:

```
# /sbin/ifconfig eth0 inet6 add 3ffe:ffff:0:f101::1/64
```

6.3. Rimuovere un indirizzo IPv6

Spesso non è necessario. Si faccia attenzione con la rimozione di indirizzi IPv6 non esistenti: a volte, usando vecchi kernel, può risultarne un crash.

6.3.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 addr del <ipv6address>/<prefixlength> dev <interface>
```

Esempio:

```
# /sbin/ip -6 addr del 3ffe:ffff:0:f101::1/64 dev eth0
```

6.3.2. Uso di "ifconfig"

Utilizzo:

```
# /sbin/ifconfig <interface> inet6 del <ipv6address>/<prefixlength>
```

Esempio:

```
# /sbin/ifconfig eth0 inet6 del 3ffe:ffff:0:f101::1/64
```

Capitolo 7. Configurazione di instradamenti IPv6 standard

Se si vuole lasciare il proprio link ed inviare pacchetti nel vasto mondo di Internet IPv6, si ha bisogno dell'instradamento (routing). Se c'è già un router abilitato per IPv6 sul proprio link, è possibile aggiungere instradamenti IPv6.

7.1. Visualizzare gli instradamenti IPv6 esistenti

Per prima cosa bisogna controllare quali e quanti indirizzi IPv6 sono già configurati (forse automaticamente durante la configurazione automatica).

7.1.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 route show [dev <device>]
```

Esempio:

```
# /sbin/ip -6 route show dev eth0
3ffe:ffff:0:f101::/64 proto kernel metric 256 mtu 1500 advmss 1440
fe80::/10             proto kernel metric 256 mtu 1500 advmss 1440
ff00::/8              proto kernel metric 256 mtu 1500 advmss 1440
default               proto kernel metric 256 mtu 1500 advmss 1440
```

7.1.2. Uso di "route"

Utilizzo:

```
# /sbin/route -A inet6
```

Esempio (l'otput è filtrato per l'interfaccia eth0). Qui si possono vedere differenti instradamenti IPv6 per i vari indirizzi su di una singola interfaccia.

```
# /sbin/route -A inet6 |grep -w "eth0"
3ffe:ffff:0:f101 ::/64 :: UA  256 0 0 eth0 <- Instradamento di interfaccia global
¬ address
fe80::/10        ::       UA  256 0 0 eth0 <- Instradamento di interfaccia link-local
¬ address
ff00::/8         ::       UA  256 0 0 eth0 <- Instradamento di interfaccia all multicast
¬ addresses
::/0             ::       UDA 256 0 0 eth0 <- Instradamento automatico di default
```

7.2. Aggiungere un instradamento IPv6 attraverso un gateway

Per lo più necessario per raggiungere l'esterno utilizzando un router abilitato IPv6 sul proprio link.

7.2.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 route add <ipv6network>/<prefixlength> via <ipv6address>
¬ [dev <device>]
```

Esempio:

```
# /sbin/ip -6 route add 2000::/3 via 3ffe:ffff:0:f101::1
```

7.2.2. Uso di "route"

Utilizzo:

```
# /sbin/route -A inet6 add <ipv6network>/<prefixlength> gw
¬ <ipv6address> [dev <device>]
```

Se l'indirizzo IPv6 è di tipo link-local può essere necessario specificare anche il dispositivo.

L'esempio seguente mostra l'aggiunta di un instradamento per tutti gli indirizzi globali (2000::/3) attraverso il gateway 3ffe:ffff:0:f101::1

```
# /sbin/route -A inet6 add 2000::/3 gw 3ffe:ffff:0:f101::1
```

7.3. Rimuovere un instradamento IPv6 attraverso un gateway

Non capita spesso di farlo manualmente, la maggior parte delle volte viene fatto tramite script di configurazione in fase di arresto (del sistema o dell'interfaccia).

7.3.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 route del <ipv6network>/<prefixlength> via <ipv6address>
¬ [dev <device>]
```

Esempio:

```
# /sbin/ip -6 route del 2000::/3 via 3ffe:ffff:0:f101::1
```

7.3.2. Uso di "route"

Utilizzo:

```
# /sbin/route -A inet6 del <network>/<prefixlength> [dev <device>]
```

Esempio per rimuovere l'instradamento precedentemente aggiunto:

```
# /sbin/route -A inet6 del 2000::/3 gw 3ffe:ffff:0:f101::1
```

7.4. Aggiungere un instradamento IPv6 attraverso un'interfaccia

Molto spesso non è necessario. Talvolta può esserlo in caso di link punto-punto dedicati.

7.4.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 route add <ipv6network>/<prefixlength> dev <device>
¬ metric 1
```

Esempio:

```
# /sbin/ip -6 route add 2000::/3 dev eth0 metric 1
```

La metrica "1" è utilizzata per compatibilità con la metrica usata dal router, poiché la metrica di default, utilizzando "ip", è "1024".

7.4.2. Uso di "route"

Utilizzo:

```
# /sbin/route -A inet6 add <network>/<prefixlength> dev <device>
```

Esempio:

```
# /sbin/route -A inet6 add 2000::/3 dev eth0
```

7.5. Rimuovere un instradamento IPv6 attraverso un'interfaccia

Molto spesso non è necessario effettuare tale operazione manualmente. È utilizzata negli script di configurazione in fase di shutdown.

7.5.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 route del <ipv6network>/<prefixlength> dev <device>
```

Esempio:

```
# /sbin/ip -6 route del 2000::/3 dev eth0
```

7.5.2. Uso di "route"

Utilizzo:

```
# /sbin/route -A inet6 del <network>/<prefixlength> dev <device>
```

Esempio:

```
# /sbin/route -A inet6 del 2000::/3 dev eth0
```

7.6. FAQ per gli instradamenti IPv6

7.6.1. Supporto di un instradamento IPv6 di default

Un'idea di base di IPv6 è stato l'instradamento gerarchico, per cui c'è bisogno di meno voci di instradamento nei router.

Esistono alcuni problemi nei kernel Linux attuali:

7.6.1.1. Client (non viene instradato nessun pacchetto!)

Il client può configurare un instradamento di default come il prefisso "::/0", può anche acquisire tale instradamento tramite la configurazione automatica, per esempio utilizzando radvd sul link come mostra il seguente esempio:

```
# ip -6 route show | grep ^default
default via fe80::212:34ff:fe12:3450 dev eth0 proto kernel metric 1024 expires
¬ 29sec mtu 1500 advmss 1440
```

7.6.1.2. I router nel caso di inoltro di pacchetti

I convenzionali kernel Linux (almeno <= 2.4.17) non supportano gli instradamenti di default. Lo si può impostare, ma la ricerca dell'instradamento fallisce nel momento in cui il pacchetto dovrebbe essere inoltrato (normale funzionamento di un router).

Perciò, al momento, il "default routing" può essere configurato soltanto utilizzando l'attuale prefisso global address "2000::/3".

Il progetto USAGI già, per mezzo di un particolare accorgimento, è in grado di supportarlo nelle sue estensioni.

Nota: sui router di confine si abbia cura di configurare il default routing senza il filtraggio degli indirizzi, altrimenti il traffico multicast o site-local non desiderato potrebbe venire inoltrato.

Capitolo 8. Scoperta del vicinato (Neighbor Discovery)

La scoperta del vicinato è il successore IPv6 dell'ARP (Address Resolution Protocol) di IPv4. Si possono recuperare informazioni sull'attuale vicinato ed in più si possono creare e cancellare voci. Il kernel tiene traccia dei vicini scoperti con successo (come l'ARP in IPv4). Si può indagare all'interno della tabella relativa ai vicini scoperti utilizzando "ip".

8.1. Visualizzare i vicini usando "ip"

Con il seguente comando si possono visualizzare i vicini IPv6 memorizzati nell'apposita tabella (scoperti o impostati manualmente)

```
# ip -6 neigh show [dev <device>]
```

Il seguente esempio mostra un vicino, costituito da un router raggiungibile

```
# ip -6 neigh show
fe80::201:23ff:fe45:6789 dev eth0 lladdr 00:01:23:45:67:89 router nud reachable
```

8.2. Manipolare la tabella relativa ai vicini utilizzando "ip"

8.2.1. Aggiungere una voce manualmente

Con il seguente comando si può aggiungere manualmente una voce:

```
# ip -6 neigh add <IPv6 address> lladdr <link-layer address> dev <device>
```

Esempio:

```
# ip -6 neigh add fec0::1 lladdr 02:01:02:03:04:05 dev eth0
```

8.2.2. Cancellare una voce manualmente

Come si aggiunge, una voce si può anche cancellare:

```
# ip -6 neigh del <IPv6 address> lladdr <link-layer address> dev <device>
```

Esempio:

```
# ip -6 neigh del fec0::1 lladdr 02:01:02:03:04:05 dev eth0
```

8.2.3. Configurazioni più avanzate

Lo strumento "ip" è poco documentato, ma molto potente. Si veda l’"help" on line per maggiori dettagli:

```
# ip -6 neigh help
Usage: ip neigh { add | del | change | replace } { ADDR [ lladdr LLADDR ]
          [ nud { permanent | noarp | stale | reachable } ]
          | proxy ADDR } [ dev DEV ]
       ip neigh {show|flush} [ to PREFIX ] [ dev DEV ] [ nud STATE ]
```

Sembra che alcune opzioni siano solo per IPv4... se potete contribuire con informazioni sui flag e l’uso avanzato, siete pregati di inviarmele.

Capitolo 9. Configurare tunnel IPv6-in-IPv4

Se si vuole uscire dal proprio link e non si hanno reti IPv6 nei dintorni, si necessita del tunneling IPv6-in-IPv4 per raggiungere l'intero mondo Internet-IPv6.

Ci sono alcuni tipi di meccanismi di tunnel ed alcune possibilità per configurarli.

9.1. Tipi di tunnel

Esiste più di una possibilità per inviare pacchetti IPv6 su link IPv4.

9.1.1. Tunneling statico punto-punto: 6bone

Questo è un tipo di tunnel dedicato verso un estremo esterno, il quale è a conoscenza della vostra rete IPv6 (per l'instradamento di ritorno) e dell'indirizzo IPv4 del vostro estremo del tunnel come definito in RFC 2893 / Transition Mechanisms for IPv6 Hosts and Routers (http://www.faqs.org/rfcs/rfc2893.html). Requisiti:

- L'indirizzo IPv4 del proprio estremo del tunnel deve essere statico, globalmente unico (pubblico) e raggiungibile dall'altro estremo (esterno) del tunnel
- Si deve avere un prefisso globale IPv6 assegnato (si veda il 6bone registry)
- L'estremo esterno di un tunnel deve essere in grado di instradare il vostro prefisso IPv6 al vostro estremo del tunnel (il più delle volte è richiesta una configurazione manuale sull'estremo esterno)

9.1.2. Tunneling automatico

Il tunneling automatico avviene quando un nodo si connette direttamente ad un altro nodo dopo averne prima ottenuto il relativo indirizzo IPv4.

9.1.3. Tunneling 6to4

Il tunneling 6to4 (RFC 3056 / Connection of IPv6 Domains via IPv4 Clouds (http://www.faqs.org/rfcs/rfc3056.html)) utilizza un semplice meccanismo per creare un tunnel in maniera automatica. Ogni nodo con un indirizzo IPv4 globalmente unico, può essere un estremo di un tunnel 6to4 (se nessu firewall IPv4 ne vieta il traffico). Il tunneling 6to4 non è generalmente uno-a-uno. Questo caso di tunneling può essere suddiviso in upstream e downstream. Inoltre, uno speciale indirizzo IPv6 indica che il nodo in questione utilizza il tunneling 6to4 per connettersi alla rete IPv6 pubblica.

9.1.3.1. Generazione di un prefisso 6to4

L'indirizzo 6to4 viene definito dal seguente schema (preso da RFC 3056 / Connection of IPv6 Domains via IPv4 Clouds (http://www.faqs.org/rfcs/rfc3056.html)):

```
|   3+13   |     32     |    16   |            64 bits             |
+---+------+-----------+--------+--------------------------------+
|  FP+TLA  |  V4ADDR   | SLA ID |          Interface ID          |
|  0x2002  |           |        |                                |
```

```
+---+------+----------+--------+-------------------------------+
```

Il campo formato da FP e TLA (16 bit) ha il valore 0x2002. V4ADDR è l'indirizzo IPv4 globalmente unico del nodo (in notazione esadecimale). SLA ID è l'identificatore della sottorete (65536 sottoreti locali possibili) ed è utilizzabile per rappresentare la propria struttura di rete locale.

Per i gateway questo prefisso viene generato normalmente usando uno SLA ID "0000" ed il suffisso "::1" (non è un obbligo, può essere anche di tipo arbitrario con ambito di applicazione locale) quindi l'indirizzo viene assegnato all'interfaccia del tunnel 6to4. Da notare che Microsoft Windows utilizza anche V4ADDR per il suffisso.

9.1.3.2. Tunneling upstream 6to4

Il nodo deve sapere a quale estremo esterno del tunnel devono essere inviati i suoi pacchetti IPv6 incapsulati in IPv4. Nei "primi" giorni del tunneling 6to4, vennero definiti dei router dedicati in grado di accettare upstream, si veda informazioni 6to4 di NSayer (http://www.kfu.com/~nsayer/6to4/) per un elenco dei router.

Oggigiorno, i router upstream possono essere trovati automaticamente, utilizzando l'indirizzo anycast 192.88.99.1. Dietro le quinte, i protocolli di routing sono in grado di gestire tale meccanismo, si veda RFC 3068 / An Anycast Prefix for 6to4 Relay Routers (http://www.faqs.org/rfcs/rfc3068.html) per i dettagli.

9.1.3.3. Tunneling downstream 6to4

Il downstream (da 6bone -> il proprio nodo abilitato 6to4) non è esattamente fissato e può variare dipendentemente dall'host esterno al quale i pacchetti vengono inviati. Esistono due possibilità:

- L'host esterno usa il 6to4 e rispedisce indietro i pacchetti direttamente al vostro nodo (si veda in basso)
- L'host esterno rispedisce indietro i pacchetti verso la rete pubblica IPv6 dipendentemente dall'instradamento dinamico, un relay router crea un tunnel automatico verso il vostro nodo.

9.1.3.4. Possibile traffico 6to4

- da 6to4 a 6to4: è di solito in un tunnel diretto tra i due host abilitati al 6to4
- da 6to4 ad un non 6to4: è inviato tramite tunneling upstream
- da non 6to4 a 6to4: è inviato tramite tunneling downstream

9.2. Visualizzare i tunnel esistenti

9.2.1. Uso di "ip"

Utilizzo:

```
# /sbin/ip -6 tunnel show [<device>]
```

Esempio:

```
# /sbin/ip -6 tunnel show
sit0: ipv6/ip remote any local any ttl 64 nopmtudisc
sit1: ipv6/ip remote 195.226.187.50 local any ttl 64
```

9.2.2. Uso di "route"

Utilizzo:

```
# /sbin/route -A inet6
```

Esempio (l'output è stato filtrato per mostrare solo i tunnel attraverso l'interfaccia virtuale sit0):

```
# /sbin/route -A inet6 | grep "\Wsit0\W*$"
::/96      ::               U   256  2  0  sit0
2002::/16  ::               UA  256  0  0  sit0
2000::/3   ::193.113.58.75  UG    1  0  0  sit0
fe80::/10  ::               UA  256  0  0  sit0
ff00::/8   ::               UA  256  0  0  sit0
```

9.3. Configurazione di un tunnel punto-punto

Ci sono 3 possibilità per aggiungere o rimuovere un tunnel punto-punto.

Una buona fonte di informazioni aggiuntive sulla configurazione di tunnel con "ip" è Configurare i tunnel con iproute2 (articolo) (http://www.deepspace6.net/docs/iproute2tunnel-en.html) (Mirror (http://mirrors.bieringer.de/www.deepspace6.net/docs/iproute2tunnel-en.html)).

9.3.1. Aggiungere un tunnel punto-punto

9.3.1.1. Uso di "ip"

Un metodo comune, attualmente utilizzato per minime quantità di tunnel.

Utilizzo per creare un device tunnel (deve essere specificato anche un TTL poiché il valore di default è 0).

```
# /sbin/ip tunnel add <device> mode sit ttl <ttldefault> remote
¬ <ipv4addressofforeigntunnel> local <ipv4addresslocal>
```

Utilizzo (esempio generico per tre tunnel):

```
# /sbin/ip tunnel add sit1 mode sit ttl <ttldefault> remote
¬ <ipv4addressofforeigntunnel1> local <ipv4addresslocal>
# /sbin/ip link set dev sit1 up
```

```
# /sbin/ip -6 route add <prefixtoroute1> dev sit1 metric 1
# /sbin/ip tunnel add sit2 mode sit ttl <ttldefault>
¬ <ipv4addressofforeigntunnel2> local <ipv4addresslocal>
# /sbin/ip link set dev sit2 up
# /sbin/ip -6 route add <prefixtoroute2> dev sit2 metric 1
# /sbin/ip tunnel add sit3 mode sit ttl <ttldefault>
¬ <ipv4addressofforeigntunnel3> local <ipv4addresslocal>
# /sbin/ip link set dev sit3 up
# /sbin/ip -6 route add <prefixtoroute3> dev sit3 metric 1
```

9.3.1.2. Uso di "ifconfig" e "route" (disapprovato)

Questo non è un sistema molto raccomandato per aggiungere un tunnel, poiché è un po' strano. Nessun problema se se ne aggiunge soltanto uno, ma volendone configurare più di uno, è piuttosto difficile chiudere il primo lasciando funzionare gli altri.

Utilizzo (esempio generico per tre tunnel):

```
# /sbin/ifconfig sit0 up
# /sbin/ifconfig sit0 tunnel <ipv4addressofforeigntunnel1>
# /sbin/ifconfig sit1 up
# /sbin/route -A inet6 add <prefixtoroute1> dev sit1
# /sbin/ifconfig sit0 tunnel <ipv4addressofforeigntunnel2>
# /sbin/ifconfig sit2 up
# /sbin/route -A inet6 add <prefixtoroute2> dev sit2
# /sbin/ifconfig sit0 tunnel <ipv4addressofforeigntunnel3>
# /sbin/ifconfig sit3 up
# /sbin/route -A inet6 add <prefixtoroute3> dev sit3
```

Importante: NON USARE QUESTO SISTEMA, poiché questa configurazione abilita implicitamente il "tunneling automatico" da qualunque macchina collegata ad Internet, ciò rappresenta un rischio, e non dovrebbe essere permesso.

9.3.1.3. Uso del solo "route"

È anche possibile configurare tunnel con lo stile Non Broadcast Multiple Access (NBMA), un modo facile per aggiungere più tunnel contemporaneamente, ma nessuno dei tunnel può essere numerato (caratteristica non richiesta).

Utilizzo (esempio generico per tre tunnel):

```
# /sbin/ifconfig sit0 up
# /sbin/route -A inet6 add <prefixtoroute1> gw
¬ ::<ipv4addressofforeigntunnel1> dev sit0
# /sbin/route -A inet6 add <prefixtoroute2> gw
¬ ::<ipv4addressofforeigntunnel2> dev sit0
# /sbin/route -A inet6 add <prefixtoroute3> gw
¬ ::<ipv4addressofforeigntunnel3> dev sit0
```

Importante: NON USARE QUESTO SISTEMA, poiché questa configurazione abilita implicitamente il "tunneling automatico" da qualunque macchina connessa ad Internet, ciò rappresenta un rischio, e non dovrebbe essere permesso.

9.3.2. Rimozione di tunnel punto-punto

Generalmente non è così necessario farlo manualmente, ma viene utilizzato negli script per lo shutdown pulito o il restart della configurazione IPv6.

9.3.2.1. Uso di "ip"

Utilizzo per la rimozione di un device tunnel:

```
# /sbin/ip tunnel del <device>
```

Utilizzo (esempio generico per tre tunnel):

```
# /sbin/ip -6 route del <prefixtoroute1> dev sit1
# /sbin/ip link set sit1 down
# /sbin/ip tunnel del sit1
# /sbin/ip -6 route del <prefixtoroute2> dev sit2
# /sbin/ip link set sit2 down
# /sbin/ip tunnel del sit2
# /sbin/ip -6 route del <prefixtoroute3> dev sit3
# /sbin/ip link set sit3 down
# /sbin/ip tunnel del sit3
```

9.3.2.2. Uso di "ifconfig" e "route" (disapprovato poiché non molto simpatico)

Non solo è strana la creazione ma anche lo shutdown... bisogna rimuovere i tunnel in ordine inverso, cioè l'ultimo creato và rimosso per primo.

Utilizzo (esempio generico per 3 tunnel):

```
# /sbin/route -A inet6 del <prefixtoroute3> dev sit3
# /sbin/ifconfig sit3 down
# /sbin/route -A inet6 del <prefixtoroute2> dev sit2
# /sbin/ifconfig sit2 down
# /sbin/route -A inet6 del <prefixtoroute1> dev sit1
# /sbin/ifconfig sit1 down
# /sbin/ifconfig sit0 down
```

9.3.2.3. Uso di "route"

È come la normale rimozione di instradamenti IPv6.

Utilizzo (esempio generico per tre tunnel):

```
# /sbin/route -A inet6 del <prefixtoroute1> gw
¬ ::<ipv4addressofforeigntunnel1> dev sit0
# /sbin/route -A inet6 del <prefixtoroute2> gw
¬ ::<ipv4addressofforeigntunnel2> dev sit0
# /sbin/route -A inet6 del <prefixtoroute3> gw
¬ ::<ipv4addressofforeigntunnel3> dev sit0
# /sbin/ifconfig sit0 down
```

9.3.3. Tunnel punto-punto numerati

A volte si rende necessario configurare un tunnel punto-punto con indirizzi IPv6 come si fa attualmente in IPv4. Ciò è possibile soltanto con il primo metodo di configurazione del tunnel (ifconfig+route - disapprovato) e con il terzo (ip+route). In questi casi si può aggiungere l'indirizzo IPv6 all'interfaccia del tunnel, come mostrato nella configurazione dell'interfaccia.

9.4. Configurazione di tunnel 6to4

Si faccia attenzione al fatto che il supporto per i tunnel 6to4 non è presente sui kernel vanilla della serie 2.2.x (si veda Controllo del sistema/kernel per maggiori informazioni). Si noti anche che la lunghezza del prefisso per un indirizzo 6to4 è 16 poiché da un punto di vista della rete, tutti gli host abilitati 6to4 sono sullo stesso livello 2.

9.4.1. Aggiugere un tunnel 6to4

Prima di tutto si deve calcolare il proprio prefisso 6to4, usando il proprio indirizzo locale IPv4 instradabile (pubblico) (se l'host non ha questo indirizzo IPv4 pubblico, in casi speciali è possibile fare il NAT sui gateway di confine):

Assumendo che il proprio indirizzo IPv4 sia

```
1.2.3.4
```

il prefisso 6to4 generato sarà

```
2002:0102:0304::
```

I gateway 6to4 locali dovrebbero sempre avere il suffisso "::1", perciò il proprio indirizzo locale 6to4 sarà

```
2002:0102:0304::1
```

Si usi per esempio il seguente comando per la generazione automatica:

```
ipv4="1.2.3.4"; printf "2002:%02x%02x:%02x%02x::1" `echo $ipv4 | tr "." " "`
```

Attualmente ci sono due modi per configurare un tunneling 6to4.

9.4.1.1. Uso di "ip" e di un dispositivo tunnel dedicato

Questo è attualmente il modo raccomandato (deve essere specificato un TTL poiché il valore di default è 0).

Creare un nuovo dispositivo tunnel

```
# /sbin/ip tunnel add tun6to4 mode sit ttl <ttldefault> remote any local <localipv4address>
```

Attivare l'interfaccia

```
# /sbin/ip link set dev tun6to4 up
```

Aggiungere un indirizzo locale 6to4 all'interfaccia (nota: è importante la lunghezza 16 del prefisso!)

```
# /sbin/ip -6 addr add <local6to4address>/16 dev tun6to4
```

Aggiungere un instradamento (di default) alla rete IPv6 globale utilizzando gli indirizzi anycast IPv4 all-6to4-routers

```
# /sbin/ip -6 route add 2000::/3 via ::192.88.99.1 dev tun6to4 metric 1
```

È stato segnalato che alcune versioni di "ip" (per es. SuSE Linux 9.0) non supportano come gateway indirizzi IPv6 compatibili IPv4, in questo caso deve essere usato il relativo indirizzo IPv6:

```
# /sbin/ip -6 route add 2000::/3 via 2002:c058:6301::1 dev tun6to4 metric 1
```

9.4.1.2. Uso di "ifconfig" e "route" e di un generico dispositivo tunnel "sit0" (disapprovato)

Questo è disapprovato poiché l'uso del dispositivo tunnel generico sit0 non permette di specificare il filtraggio per dispositivo.

Attivare l'interfaccia generica tunnel sit0

```
# /sbin/ifconfig sit0 up
```

Aggiungere l'indirizzo locale 6to4 all'interfaccia

```
# /sbin/ifconfig sit0 add <local6to4address>/16
```

Aggiungere l'instradamento (di default) alla rete IPv6 globale utilizzando l'indirizzo IPv4 anycast all-6to4-relays

```
# /sbin/route -A inet6 add 2000::/3 gw ::192.88.99.1 dev sit0
```

9.4.2. Rimuovere un tunnel 6to4

9.4.2.1. Uso di "ip" e di un dispositivo tunnel dedicato

Rimuovere tutti gli instradamenti attraverso questo dispositivo tunnel dedicato

```
# /sbin/ip -6 route flush dev tun6to4
```

Disabilitare l'interfaccia

```
# /sbin/ip link set dev tun6to4 down
```

Rimuovere il dispositivo tunnel creato

```
# /sbin/ip tunnel del tun6to4
```

9.4.2.2. Uso di "ifconfig" e "route" ed un dispositivo tunnel generico "sit0" (disapprovato)

Rimuovere l'instradamento (di default) attraverso l'interfaccia del tunnel 6to4

```
# /sbin/route -A inet6 del 2000::/3 gw ::192.88.99.1 dev sit0
```

Rimuovere l'indirizzo locale 6to4 dall'interfaccia

```
# /sbin/ifconfig sit0 del <local6to4address>/16
```

Disabilitare il dispositivo tunnel generico (si faccia attenzione a questo, potrebbe essere ancora in uso...)

```
# /sbin/ifconfig sit0 down
```

Capitolo 10. Configurare tunnel IPv4-in-IPv6

Sarà riempito in futuro. Per il momento, questi tunnel sono principalmente usati in ambienti di test ma sembra che attualmente manchi il supporto per Linux (03/2004).

Nel frattempo per maggiori informazioni: RFC 2473 / Generic Packet Tunneling in IPv6 Specification (http://www.faqs.org/rfcs/rfc2473.html).

Capitolo 11. Impostazioni del kernel nel filesystem /proc

Nota: l'origine di questa sezione è essenzialmente il file "ip-sysctl.txt" che è incluso negli attuali sorgenti del kernel nella directory "Documentation/networking". Ringraziamenti a Pekka Savola per il mantenimento della parte relativa a IPv6 in questo file. Inoltre è stato fatto un parziale copia &incolla del testo in questo documento.

11.1. Come accedere al filesystem /proc

11.1.1. Uso di "cat" e "echo"

L'utilizzo di "cat" e "echo" è il modo più semplice per accedere al filesystem /proc, sono richiesti però anche alcuni requisiti

- Il filesystem /proc deve essere abilitato nel kernel, significa cioè che durante la compilazione deve essere selezionata la seguente opzione

```
CONFIG_PROC_FS=y
```

- Il filesistem /proc deve essere prima montato e ciò può essere verificato utilizzando

```
# mount | grep "type proc"
none on /proc type proc (rw)
```

- Si necessita del permesso di lettura ed a volte anche di scrittura (di solito solo root) al filesystem /proc

Di solito, solo le voci in /proc/sys/* sono scrivibili, le altre sono in lettura e servono solo per il recupero di informazioni.

11.1.1.1. Recuperare un valore

Il valore di una voce può essere recuperato utilizzando "cat":

```
# cat /proc/sys/net/ipv6/conf/all/forwarding
0
```

11.1.1.2. Impostazione di un valore

Un nuovo valore può essere impostato (se la relativa voce è scrivibile) utilizzando "echo":

```
# echo "1" >/proc/sys/net/ipv6/conf/all/forwarding
```

11.1.2. Uso di "sysctl"

L'utilizzo del programma "sysctl" è un metodo moderno per accedere agli switch del kernel. Si può usare anche nel caso in cui il filesystem /proc non sia montato. Si ha però accesso solo a /proc/sys/*!

Il programma "sysctl" è incluso nel pacchetto "procps" (su sistemi Red Hat Linux).

- L'interfaccia sysctl deve essere abilitata nel kernel, in fase di compilazione deve cioè essere selezionata l'opzione seguente

```
CONFIG_SYSCTL=y
```

11.1.2.1. Recupero di un valore

Il valore di una voce può essere recuperato con:

```
# sysctl net.ipv6.conf.all.forwarding
net.ipv6.conf.all.forwarding = 0
```

11.1.2.2. Impostazione di un valore

Un nuovo valore può essere impostato (se la relativa voce è scrivibile) con:

```
# sysctl -w net.ipv6.conf.all.forwarding=1
net.ipv6.conf.all.forwarding = 1
```

Nota: Non usare spazi prima e dopo del segno "=" quando si impostano valori. Inoltre, nel caso di valori multipli sulla stessa riga è necessario racchiuderli tra virgolette, come per esempio

```
# sysctl -w net.ipv4.ip_local_port_range="32768 61000"
net.ipv4.ip_local_port_range = 32768 61000
```

11.1.2.3. Supplementi

Nota: in giro ci sono versioni di sysctl che visualizzano "/" invece di "."

Per maggiori dettagli si dia un'occhiata alla pagina man di sysctl.

Suggerimento: per cercare velocemente nelle impostazioni, utilizzare l'opzione "-a" (mostra tutte le voci) assieme a "grep".

11.1.3. Valori trovati nel filesystem /proc

Esistono vari formati (tipi di dati) dei valori nel filesystem /proc:

- BOOLEANO: semplicemente uno "0" (falso) o un "1" (vero)
- INTERO: un valore intero che può anche essere senza segno (unsigned)
- righe più sofisticate, con diversi valori: a volte viene anche visualizzata una riga di intestazione, in caso contrario si dia uno sguardo ai sorgenti del kernel per avere informazioni sul significato di ogni valore...

11.2. Voci in /proc/sys/net/ipv6/

11.2.1. conf/default/*

Cambia le impostazioni di default specifiche di un'interfaccia.

11.2.2. conf/all/*

Cambia tutte le impostazioni specifiche di un'interfaccia.

Un'eccezione: "conf/all/forwarding" ha un significato diverso.

11.2.2.1. conf/all/forwarding

- Tipo: BOOLEANO

Questo abilita l'inoltro globale IPv6 tra tutte le interfacce.

Con IPv6 non si può controllare l'inoltro per dispositivo, ma comunque tale controllo può essere effettuato utilizzando il set di regole netfilter-IPv6 (controllato con ip6tables) e specificando i dispositivi di input e di output (si veda Firewalling/Netfilter6 per maggiori dettagli). In IPv4 ciò è gestito in maniera diversa poiché si può controllare l'inoltro per dispositivo (la decisione viene presa sull'interfaccia dove il pacchetto arriva).

Inoltre configura tutte le impostazioni 'inoltro' Host/Router dell'interfaccia col valore specificato. Si veda successivamente per i dettagli. Ci si referisce a questo come inoltro globale (global forwarding).

Se questo valore è 0, l'inoltro IPv6 non è abilitato: i pacchetti non lasceranno mai un'interfaccia, né fisica né logica, come per esempio i tunnel.

11.2.3. conf/interface/*

Cambia le impostazioni particolari per interfaccia.

Il comportamento per certe impostazioni è diverso a seconda se l'inoltro locale è abilitato o meno.

11.2.3.1. accept_ra

- Tipo: BOOLEANO
- Funzionalità di default: abilitato se l'inoltro locale è disabilitato. Disabilitato in caso contrario.

Accetta gli avvisi router (Router Advertisements) e configura automaticamente l'interfaccia con i dati ricevuti.

11.2.3.2. accept_redirects

- Tipo: BOOLEANO
- Funzionalità di default: abilitato se l'inoltro locale è disabilitato. Disabilitato in caso contrario.

Accetta i redirect inviati da un router IPv6.

11.2.3.3. autoconf

- Tipo: BOOLEANO
- Default: VERO

Configura indirizzi link-local (si veda anche Tipi di indirizzo) usando indirizzi hardware di livello 2 (L2). Per esempio questo genera automaticamente un indirizzo come "fe80::201:23ff:fe45:6789" su di un'interfaccia con un indirizzo L2-MAC.

11.2.3.4. dad_transmits

- Tipo: INTERO
- Default: 1

Il numero totale di segnali di tipo Duplicate Address Detection da inviare.

11.2.3.5. forwarding

- Tipo: BOOLEANO
- Default: FALSO se l'inoltro globale è disabilitato (default), altrimenti VERO

Configura il comportamento da Host/Router specifico per interfaccia.

Nota: Si raccomanda di impostare la stessa configurazione su tutte le interfacce; gli scenari misti router/host sono piuttosto insoliti.

- Valore FALSO: Per default, viene assunto il comportamento Host. Questo significa:

  1. Il flag IsRouter non è impostato negli avvisi di vicinato (Neighbour Advertisements).
  2. Le sollecitazioni router (Router Solicitation) vengono inviate quando necessario.

3. Se accept_ra è VERO (default), vengono accettati gli avvisi router (e viene fatta la configurazione automatica).
4. Se accept_redirects è VERO (default), vengono accettati i redirect.

- Valore VERO: Se l'inoltro locale è abilitato, viene assunto il comportamento come router. Questo significa esattamente l'inverso rispetto a quanto detto prima:

1. Il flag IsRouter è impostato negli avvisi di vicinato.
2. Le sollecitazioni router non vengono inviate.
3. Gli avvisi router vengono ignorati.
4. I redirect vengono ignorati.

11.2.3.6. hop_limit

- Tipo: INTERO
- Default: 64

Il limite massimo di salti (hop) di default.

11.2.3.7. mtu

- Tipo: INTERO
- Default: 1280 (minimo richiesto per IPv6)

La Maximum Transfer Unit di default.

11.2.3.8. router_solicitation_delay

- Tipo: INTERO
- Default: 1

Secondi da attendere dopo l'attivazione dell'interfaccia prima di inviare sollecitazioni router.

11.2.3.9. router_solicitation_interval

- Tipo: INTERO
- Default: 4

Secondi di attesa tra una sollecitazione router e la successiva.

11.2.3.10. router_solicitations

- Tipo: INTERO
- Default: 3

Numero di sollecitazioni router da inviare prima di desumere l'assenza di router.

11.2.4. neigh/default/*

Cambia le impostazioni di default per la rilevazione di vicinato per alcuni speciali intervalli globali e valori di soglia:

11.2.4.1. gc_thresh1

- Tipo: INTERO
- Default: 128

Da completare con maggiori dettagli.

11.2.4.2. gc_thresh2

- Tipo: INTERO
- Default: 512

Da completare con maggiori dettagli.

11.2.4.3. gc_thresh3

- Tipo: INTERO
- Default: 1024

Parametro per impostare la dimensione della tabella relativa ai vicini.

Si aumenti questo valore nel caso in cui si abbiano molte interfacce e problemi con gli instradamenti che cominciano a funzionare ed a fallire misteriosamente, oppure nel caso che facendo funzionare lo Zebra (demone di routing) (http://www.zebra.org/) si abbia:

```
ZEBRA: netlink-listen error: No buffer space available, type=RTM_NEWROUTE(24), seq=426, pid=0
```

11.2.4.4. gc_interval

- Tipo: INTERO
- Default: 30

Da completare con maggiori dettagli.

11.2.5. neigh/interface/*

Cambia delle impostazioni particolari, per interfaccia, per la rilevazione del vicinato.

11.2.5.1. anycast_delay

- Tipo: INTERO
- Default: 100

Da completare con maggiori dettagli.

11.2.5.2. gc_stale_time

- Tipo: INTERO
- Default: 60

Da completare con maggiori dettagli.

11.2.5.3. proxy_qlen

- Tipo: INTERO
- Default: 64

Da completare con maggiori dettagli.

11.2.5.4. unres_qlen

- Tipo: INTERO
- Default: 3

Da completare con maggiori dettagli.

11.2.5.5. app_solicit

- Tipo: INTERO
- Default: 0

Da completare con maggiori dettagli.

11.2.5.6. locktime

- Tipo: INTERO
- Default: 0

Da completare con maggiori dettagli.

11.2.5.7. retrans_time

- Tipo: INTERO
- Default: 100

Da completare con maggiori dettagli.

11.2.5.8. base_reachable_time

- Tipo: INTERO
- Default: 30

Da completare con maggiori dettagli.

11.2.5.9. mcast_solicit

- Tipo: INTERO
- Default: 3

Da completare con maggiori dettagli.

11.2.5.10. ucast_solicit

- Tipo: INTERO
- Default: 3

Da completare con maggiori dettagli.

11.2.5.11. delay_first_probe_time

- Tipo: INTERO
- Default: 5

Da completare con maggiori dettagli.

11.2.5.12. proxy_delay

- Tipo: INTERO
- Default: 80

Da completare con maggiori dettagli.

11.2.6. route/*

Cambia le impostazioni globali per l'instradamento.

11.2.6.1. flush

Rimosso dalle recenti release del kernel - da completare con maggiori dettagli.

11.2.6.2. gc_interval

- Tipo: INTERO
- Default: 30

Da completare con maggiori dettagli.

11.2.6.3. gc_thresh

- Tipo: INTERO
- Default: 1024

Da completare con maggiori dettagli.

11.2.6.4. mtu_expires

- Tipo: INTERO
- Default: 600

Da completare con maggiori dettagli.

11.2.6.5. gc_elasticity

- Tipo: INTERO
- Default: 0

Da completare con maggiori dettagli.

11.2.6.6. gc_min_interval

- Tipo: INTERO
- Default: 5

Da completare con maggiori dettagli.

11.2.6.7. gc_timeout

- Tipo: INTERO
- Default: 60

Da completare con maggiori dettagli.

11.2.6.8. min_adv_mss

- Tipo: INTERO
- Default: 12

Da completare con maggiori dettagli.

11.2.6.9. max_size

- Tipo: INTERO
- Default: 4096

Da completare con maggiori dettagli.

11.3. Voci relative a IPv6 in /proc/sys/net/ipv4/

Al momento (e fino a quando IPv4 non sarà convertito in un modulo del kernel indipendente) alcune opzioni qui presenti vengono utilizzate anche per IPv6.

11.3.1. ip_*

11.3.1.1. ip_local_port_range

Questa impostazione è usata anche da IPv6.

11.3.2. tcp_*

Queste impostazioni sono usate anche da IPv6.

11.3.3. icmp_*

Queste impostazioni non vengono utilizzate da IPv6. Per abilitare il limite sul flusso (rate limiting) dei messaggi ICMPv6 (vivamente raccomandato a causa della possibilità di tempeste di ICMPv6) devono essere usate le regole netfilter-v6.

11.3.4. altre

Sconosciute, ma probabilmente non usate da IPv6.

11.4. Voci relative a IPv6 in /proc/net/

In /proc/net ci sono diversi valori disponibili in sola lettura. Tali informazioni non possono essere ricavate usando "sysctl", quindi si usi per esempio "cat".

11.4.1. if_inet6

- Tipo: Una riga per ogni indirizzo, che contiene più valori

Tutti gli indirizzi IPv6 configurati, vengono visualizzati in un formato speciale. L'esempio mostra solo quello dell'interfaccia di loopback. Il significato è indicato di seguito (si veda "net/ipv6/addrconf.c" per maggiori dettagli).

```
# cat /proc/net/if_inet6
00000000000000000000000000000001 01 80 10 80 lo
+------------------------------+ ++ ++ ++ ++ ++
|                                |  |  |  |  |
1                                2  3  4  5  6
```

1. Indirizzo IPv6 rappresentato con 32 caratteri esadecimali senza i due punti come separatore
2. Numero dispositivo netlink (indice di interfaccia) in esadecimale (si veda anche "ip addr")
3. Lunghezza del prefisso in esadecimale
4. Valore dell'ambito di applicazione (scope value) (si vedano i sorgenti del kernel "include/net/ipv6.h" e "net/ipv6/addrconf.c" per maggiori dettagli)
5. Flag dell'interfaccia (si veda "include/linux/rtnetlink.h" e "net/ipv6/addrconf.c" per maggiori dettagli)
6. Nome del dispositivo

11.4.2. ipv6_route

- Tipo: Una riga per ogni instradamento, che contiene più valori

Tutti gli instradamenti IPv6 configurati vengono visualizzati in un formato speciale. L'esempio riguarda solo l'interfaccia di loopback. Il significato viene indicato di seguito (si veda "net/ipv6/route.c" per maggiori dettagli).

```
# cat /proc/net/ipv6_route
00000000000000000000000000000000 00 00000000000000000000000000000000 00
+------------------------------+ ++ +------------------------------+ ++
|                                |  |                                |
1                                2  3                                4
¬ 00000000000000000000000000000000 ffffffff 00000001 00000001 00200200 lo
¬ +------------------------------+ +------+ +------+ +------+ +------+ ++
¬ |                                |        |        |        |        |
¬ 5                                6        7        8        9        10
```

1. Rete IPv6 di destinazione rappresentata con 32 caratteri esadecimali senza i due punti come separatore
2. Lunghezza del prefisso IPv6 di destinazione in esadecimale
3. Rete IPv6 mittente (source) rappresentata con 32 caratteri esadecimali senza i due punti come separatore
4. Lunghezza del prefisso IPv6 mittente (source) in esadecimale
5. L'hop IPv6 successivo rappresentato con 32 caratteri esadecimali senza i due punti come separatore
6. Metrica in esadecimale
7. Contatore di riferimento
8. Contatore di uso
9. Flag
10. Nome del dispositivo

11.4.3. sockstat6

- Tipo: Una riga per protocollo, con descrizione e valore

Statistiche sui socket IPv6 usati. Esempio:

```
# cat /proc/net/sockstat6
TCP6: inuse 7
UDP6: inuse 2
RAW6: inuse 1
FRAG6: inuse 0 memory 0
```

11.4.4. tcp6

Da completare.

11.4.5. udp6

Da completare.

11.4.6. igmp6

Da completare.

11.4.7. raw6

Da completare.

11.4.8. ip6_flowlabel

Da completare.

11.4.9. rt6_stats

Da completare.

11.4.10. snmp6

- Tipo: Una riga per la descrizione SNMP e il valore

Statistiche SNMP, possono essere recuperate tramite il server SNMP e la relativa tabella MIB con software di gestione di rete.

11.4.11. ip6_tables_names

Tabelle netfilter6 disponibili.

Capitolo 12. Interfaccia netlink al kernel

Da completare... Non ho esperienza con questo...

Capitolo 13. Debug di rete

13.1. Server socket binding

13.1.1. Uso di "netstat" per il controllo del server socket binding

È sempre interessante vedere quali socket server sono attivi in un dato momento su di un nodo. L'uso di "netstat" è una scorciatoia per ottenere tali informazioni:

Opzioni utilizzate: -nlptu

Esempio:

```
# netstat -nlptu
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
¬ PID/Program name
tcp        0      0 0.0.0.0:32768           0.0.0.0:*               LISTEN
¬ 1258/rpc.statd
tcp        0      0 0.0.0.0:32769           0.0.0.0:*               LISTEN
¬ 1502/rpc.mountd
tcp        0      0 0.0.0.0:515             0.0.0.0:*               LISTEN
¬ 22433/lpd Waiting
tcp        0      0 1.2.3.1:139             0.0.0.0:*               LISTEN
¬ 1746/smbd
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
¬ 1230/portmap
tcp        0      0 0.0.0.0:6000            0.0.0.0:*               LISTEN
¬ 3551/X
tcp        0      0 1.2.3.1:8081            0.0.0.0:*               LISTEN
¬ 18735/junkbuster
tcp        0      0 1.2.3.1:3128            0.0.0.0:*               LISTEN
¬ 18822/(squid)
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN
¬ 30734/named
tcp        0      0 ::ffff:1.2.3.1:993      :::*                    LISTEN
¬ 6742/xinetd-ipv6
tcp        0      0 :::13                   :::*                    LISTEN
¬ 6742/xinetd-ipv6
tcp        0      0 ::ffff:1.2.3.1:143      :::*                    LISTEN
¬ 6742/xinetd-ipv6
tcp        0      0 :::53                   :::*                    LISTEN
¬ 30734/named
tcp        0      0 :::22                   :::*                    LISTEN
¬ 1410/sshd
tcp        0      0 :::6010                 :::*                    LISTEN
¬ 13237/sshd
udp        0      0 0.0.0.0:32768           0.0.0.0:*
¬ 1258/rpc.statd
udp        0      0 0.0.0.0:2049            0.0.0.0:*
¬ -
```

```
udp        0      0 0.0.0.0:32770           0.0.0.0:*
¬ 1502/rpc.mountd
udp        0      0 0.0.0.0:32771           0.0.0.0:*
¬ -
udp        0      0 1.2.3.1:137             0.0.0.0:*
¬ 1751/nmbd
udp        0      0 0.0.0.0:137             0.0.0.0:*
¬ 1751/nmbd
udp        0      0 1.2.3.1:138             0.0.0.0:*
¬ 1751/nmbd
udp        0      0 0.0.0.0:138             0.0.0.0:*
¬ 1751/nmbd
udp        0      0 0.0.0.0:33044           0.0.0.0:*
¬ 30734/named
udp        0      0 1.2.3.1:53              0.0.0.0:*
¬ 30734/named
udp        0      0 127.0.0.1:53            0.0.0.0:*
¬ 30734/named
udp        0      0 0.0.0.0:67              0.0.0.0:*
¬ 1530/dhcpd
udp        0      0 0.0.0.0:67              0.0.0.0:*
¬ 1530/dhcpd
udp        0      0 0.0.0.0:32858           0.0.0.0:*
¬ 18822/(squid)
udp        0      0 0.0.0.0:4827            0.0.0.0:*
¬ 18822/(squid)
udp        0      0 0.0.0.0:111             0.0.0.0:*
¬ 1230/portmap
udp        0      0 :::53                   :::*
¬ 30734/named
```

13.2. Esempi di dump dei pacchetti di tcpdump

Qui vengono mostrati alcuni esempi di pacchetti catturati, probabilmente utili per il proprio debugging...

...maggiori contenuti prossimamente...

13.2.1. Scoperta del router

13.2.1.1. Avviso di router (Router advertisement)

```
15:43:49.484751 fe80::212:34ff:fe12:3450 > ff02::1: icmp6: router
¬ advertisement(chlim=64, router_ltime=30, reachable_time=0,
¬ retrans_time=0)(prefix info: AR valid_ltime=30, preffered_ltime=20,
¬ prefix=2002:0102:0304:1::/64)(prefix info: LAR valid_ltime=2592000,
¬ preffered_ltime=604800, prefix=3ffe:ffff:0:1::/64)(src lladdr:
¬ 0:12:34:12:34:50) (len 88, hlim 255)
```

Il router con indirizzo link-local "fe80::212:34ff:fe12:3450" invia un avviso all'indirizzo multicast all-node-on-link "ff02::1" contenente due prefissi "2002:0102:0304:1::/64" (durata 30 s) e "3ffe:ffff:0:1::/64" (durata 2592000 s) con inclusi i suoi indirizzi di livello 2 (MAC) "0:12:34:12:34:50".

13.2.1.2. Sollecitazione di router (Router solicitation)

```
15:44:21.152646 fe80::212:34ff:fe12:3456 > ff02::2: icmp6: router solicitation
¬ (src lladdr: 0:12:34:12:34:56) (len 16, hlim 255)
```

Il nodo con indirizzo link-local "fe80::212:34ff:fe12:3456" ed indirizzo di livello 2 (MAC) "0:12:34:12:34:56" sta cercando un router on-link, quindi invia questa sollecitazione all'indirizzo multicast all-router-on-link "ff02::2".

13.2.2. Scoperta del vicinato

13.2.2.1. Sollecitazione di scoperta del vicinato per rilevamento di indirizzi duplicati

I seguenti pacchetti sono inviati da un nodo con indirizzo di livello 2 (MAC) "0:12:34:12:34:56" durante la configurazione automatica, per controllare se un potenziale indirizzo è utilizzato o meno da un altro nodo sul link, inviandoli all'indirizzo multicast link-local solicited node.

- Il nodo vuole configurare il suo indirizzo link-local "fe80::212:34ff:fe12:3456" e controlla la presenza di duplicati

```
15:44:17.712338 :: > ff02::1:ff12:3456: icmp6: neighbor sol: who has
¬ fe80::212:34ff:fe12:3456(src lladdr: 0:12:34:12:34:56) (len 32, hlim 255)
```

- Il nodo vuole configurare il suo indirizzo globale "2002:0102:0304:1:212:34ff:fe12:3456" (dopo aver ricevuto l'avviso precedentemente mostrato) e controlla ora la presenza di duplicati

```
15:44:21.905596 :: > ff02::1:ff12:3456: icmp6: neighbor sol: who has
¬ 2002:0102:0304:1:212:34ff:fe12:3456(src lladdr: 0:12:34:12:34:56) (len 32,
¬ hlim 255)
```

- Il nodo vuole configurare il suo indirizzo globale "3ffe:ffff:0:1:212:34ff:fe12:3456" (dopo aver ricevuto l'avviso precedentemente mostrato) e controlla ora la presenza di duplicati

```
15:44:22.304028 :: > ff02::1:ff12:3456: icmp6: neighbor sol: who has
¬ 3ffe:ffff:0:1:212:34ff:fe12:3456(src lladdr: 0:12:34:12:34:56) (len 32, hlim
¬ 255)
```

13.2.2.2. Sollecitazione di scoperta del vicinato per la ricerca di host o gateway

- Il nodo vuole inviare pacchetti a "3ffe:ffff:0:1::10" ma non ha l'indirizzo di livello 2 (MAC) a cui inviarli, per cui invia ora la sollecitazione

```
13:07:47.664538 2002:0102:0304:1:2e0:18ff:fe90:9205 > ff02::1:ff00:10: icmp6:
¬ neighbor sol: who has 3ffe:ffff:0:1::10(src lladdr: 0:e0:18:90:92:5) (len 32,  hlim 255)
```

- Il nodo cerca ora "fe80::10"

```
13:11:20.870070 fe80::2e0:18ff:fe90:9205 > ff02::1:ff00:10: icmp6: neighbor
¬ sol: who has fe80::10(src lladdr: 0:e0:18:90:92:5) (len 32, hlim 255)
```

Capitolo 14. Supporto per la configurazione persistente di IPv6 nelle distribuzioni Linux

Alcune distribuzioni Linux già contengono il supporto per la configurazione persistente di IPv6 utilizzando quella esistente o nuova, file script e riferimenti a file di script IPv4.

14.1. Red Hat Linux ed i "cloni"

Da quando ho iniziato la stesura di IPv6 &Linux - HowTo (http://www.bieringer.de/linux/IPv6/) è stata mia intenzione abilitare una configurazione persistente di IPv6 la quale prendesse in considerazione la maggior parte dei casi desiderati come host, router, host dual-homed, router con un secondo troncone di rete, tunnel normali, tunnel 6to4 e così via. Oggigiorno esiste un set di configurazioni e file di script che fanno molto bene questo lavoro (non ho mai sentito di problemi reali ma non so quanti lo usano). Poiché questi file di configurazione e script vengono estesi di volta in volta, hanno una propria homepage: initscripts-ipv6 homepage (http://www.deepspace6.net/projects/initscripts-ipv6.html) (Mirror (http://mirrors.bieringer.de/www.deepspace6.net/projects/initscripts-ipv6.html)). Dato che ho iniziato la mia esperienza con IPv6 utilizzando un clone della Red Hat Linux 5.0, i miei sistemi di sviluppo IPv6 sono perlopiù basati su Red Hat Linux, è quindi piuttosto logico che gli script siano sviluppati per questo tipo di distribuzione (le cosiddette *ragioni storiche*). Inoltre è molto semplice estendere alcuni file di configurazione, crearne di nuovi e realizzare alcuni semplici collegamenti per configurare IPv6 durante la configurazione di IPv4.

Fortunatamente, da Red Hat Linux 7.1 in poi, è inclusa una parte dei miei script IPv6; ciò è dovuto, e lo sarà in avvenire, all'assistenza di Pekka Savola.

Anche Mandrake dalla versione 8.0 include un pacchetto di initscript IPv6 ma un piccolo bug ne rende inefficace l'utilizzo ("ifconfig" manca dell'"inet6" prima di "add").

14.1.1. Test per il supporto di IPv6 negli script di configurazione della rete

Si può verificare se la propria distribuzione Linux contenga o meno il supporto per la configurazione persistente di IPv6 con il mio set. Dovrebbe esistere la seguente libreria di script:

```
/etc/sysconfig/network-scripts/network-functions-ipv6
```

Test automatico:

```
# test -f /etc/sysconfig/network-scripts/network-functions-ipv6 && echo "Main
¬ IPv6 script library exists"
```

La versione della libreria IPv6 è importante se qualche caratteristica risultasse mancante. La si può ottenere lanciando il seguente comando (o più semplicemente guardando all'inizio del file):

```
# source /etc/sysconfig/network-scripts/network-functions-ipv6 &&
¬ getversion_ipv6_functions
20011124
```

Nell'esempio mostrato, la versione utilizzata è la 20011124. Si confronti questa con le ultime informazioni su initscripts-ipv6 homepage (http://www.deepspace6.net/projects/initscripts-ipv6.html) (Mirror (http://mirrors.bieringer.de/www.deepspace6.net/projects/initscripts-ipv6.html)) per vedere cosa è stato cambiato. È possibile trovare anche uno storico dei cambiamenti (change-log).

14.1.2. Brevi suggerimenti per abilitare IPv6 sulle attuali RHL 7.1, 7.2, 7.3, ...

- Verificare sul sistema se il modulo IPv6 è stato caricato o meno

```
# modprobe -c | grep net-pf-10
alias net-pf-10 off
```

- Se il risultato è "off" allora si abiliti l'IPv6 aggiungendo la seguente linea a /etc/sysconfig/network

```
NETWORKING_IPV6=yes
```

- Riavviare la macchina o riavviare la gestione della rete usando

```
# service network restart
```

- Adesso il modulo IPv6 dovrebbe essere stato caricato

```
# modprobe -c | grep ipv6
alias net-pf-10 ipv6
```

Se il sistema è su di un link che fornisce il router advertisement, verrà fatta una configurazione automatica. Per maggiori informazioni sulle impostazioni supportate, si veda /usr/share/doc/initscripts-$version/sysconfig.txt.

14.2. SuSE Linux

Nelle più recenti versioni 7.x è disponibile un supporto molto rudimentale, si veda /etc/rc.config per i dettagli.

A causa delle configurazioni e della struttura dei file di script molto diversi, è difficile (se non impossibile) utilizzare il set per le RedHat ed i cloni, con questa distribuzione. Nelle versioni 8.x le impostazioni di configurazione cambiano completamente.

14.2.1. SuSE Linux 7.3

- Come configurare il 6to4 IPv6 con SuSE 7.3 (http://www.feyrer.de/IPv6/SuSE73-IPv6+6to4-setup.html)

14.2.2. SuSE Linux 8.0

14.2.2.1. Configurazione indirizzo IPv6

Si inserisca il seguente valore nel file /etc/sysconfig/network/ifcfg-<Interface-Name>

```
IP6ADDR="<ipv6-address>/<prefix>"
```

14.2.2.2. Informazioni addizionali

Si veda il file /usr/share/doc/packages/sysconfig/README.

14.2.3. SuSE Linux 8.1

14.2.3.1. Configurazione indirizzo IPv6

Si inserisca il seguente valore nel file /etc/sysconfig/network/ifcfg-<Interface-Name>

```
IPADDR="<ipv6-address>/<prefix>"
```

14.2.3.2. Informazioni addizionali

Si veda il file /usr/share/doc/packages/sysconfig/Network.

14.3. Debian Linux

Le informazioni seguenti sono dovute al contributo di Stephane Bortzmeyer <bortzmeyer at nic dot fr>

1. Ci si assicuri che IPv6 sia attivo, che sia compilato nel kernel o che il modulo sia caricato. Per quest'ultimo ci sono tre possibilità, aggiungerlo in /etc/modules, utilizzare il trucco pre-up mostrato più avanti o utilizzando kmod (qui non descritto).
2. Si configurino le interfacce. Assumiamo di avere eth0 ed un indirizzo (3ffe:ffff:1234:5::1:1). Si modifichi /etc/network/interfaces come segue:

```
iface eth0 inet6 static
        pre-up modprobe ipv6
```

```
address 3ffe:ffff:1234:5::1:1
# To suppress completely autoconfiguration:
# up echo 0 > /proc/sys/net/ipv6/conf/all/autoconf
netmask 64
# The router is autoconfigured and has no fixed address.
# It is magically
# found. (/proc/sys/net/ipv6/conf/all/accept_ra). Otherwise:
#gateway 3ffe:ffff:1234:5::1
```

Ora si riavvii o si digiti semplicemente

```
# ifup --force eth0
```

si ha così il proprio indirizzo statico.

14.3.1. Altre informazioni

- IPv6 su Debian Linux (http://people.debian.org/~csmall/ipv6/) di Craig Small
- HOWTO per Freenet6 &Utenti Debian (http://www.jipo.org/jim/Jims_LAN_IPv6_global_connectivity_howto.html) di Jean-Marc V. Liotier (annunciato il 24.12.2002 sulla mailing list users@ipv6.org)

Capitolo 15. Configurazione automatica e mobilità

15.1. configurazione automatica stateless

È supportata e vista sull'indirizzo assegnato link-local dopo che un'interfaccia abilitata per IPv6 viene attivata.

15.2. configurazione automatica stateful con l'utilizzo del Router Advertisement Daemon (radvd)

Da completare. Si veda autoconfigurazione con demone radvd più avanti.

15.3. Dynamic Host Configuration Protocol v6 (DHCPv6)

Dopo una lunga discussione, finalmente la RFC 3315 / Dynamic Host Configuration Protocol for IPv6 (DHCPv6) (http://www.faqs.org/rfcs/rfc3315.html) è stata terminata. Al momento dell'aggiornamento di questa sezione (03/2004) sono disponibili due implementazioni:

- http://klub.com.pl/dhcpv6/ di Tomasz Mrugalski <thomson at klub dot com dot pl>
- DHCPv6 su Sourceforge (http://dhcpv6.sourceforge.net/)

15.4. Mobilità

Da completare.

Per il momento, si veda homepage IPv6 Mobile per Linux(MIPL) (http://www.mipl.mediapoli.com/) per maggiori dettagli. Ulteriori informazioni si possono trovare qui:

- draft-oneill-mipv6-cao-??.txt/MIPv6 Care of Address Option (http://www.ietf.org/internet-drafts/)
- draft-mccann-mobileip-80211fh-01.txt/Mobile IPv6 Fast Handovers for 802.11 Networks (http://www.ietf.org/internet-drafts/)
- draft-haberman-ipv6-anycast-rr-??.txt/IPv6 Anycast Binding using Return Routability (http://www.ietf.org/internet-drafts/)
- draft-mun-aaa-localkm-mobileipv6-??.txt/Localized Key Management for AAA in MobileIPv6 (http://www.ietf.org/internet-drafts/)
- draft-thubert-nemo-ro-taxonomy-??.txt/Taxonomy of Route Optimization Models in the NEMO Context (http://www.ietf.org/internet-drafts/)
- draft-le-aaa-diameter-mobileipv6-??.txt/Diameter Mobile IPv6 Application (http://www.ietf.org/internet-drafts/)

- draft-wakikawa-manet-globalv6-02.txt/Global Connectivity for IPv6 Mobile Ad Hoc Networks (http://www.ietf.org/internet-drafts/)
- draft-ietf-mobileip-fast-mipv6-??.txt/Fast Handovers for Mobile IPv6 (http://www.ietf.org/internet-drafts/)
- draft-ietf-mobileip-ipv6-??.txt/Mobility Support in IPv6 (http://www.ietf.org/internet-drafts/)
- draft-ohnishi-mobileip-v6vpngateway-??.txt/Mobile IPv6 VPN using Gateway Home Agent (http://www.ietf.org/internet-drafts/)
- draft-ietf-mobileip-hmipv6-??.txt/Hierarchical MIPv6 mobility management (HMIPv6) (http://www.ietf.org/internet-drafts/)
- draft-mkhalil-ipv6-fastra-??.txt/IPv6 Fast Router Advertisement (http://www.ietf.org/internet-drafts/)
- draft-okazaki-mobileip-abk-??.txt/Securing MIPv6 Binding Updates Using Address Based Keys (ABKs) (http://www.ietf.org/internet-drafts/)
- draft-vriz-mobileip-hbhlmap-??.txt/Hop-by-Hop Local Mobility Agents Probing for Mobile IPv6 (http://www.ietf.org/internet-drafts/)
- draft-thubert-nemo-reverse-routing-header-??.txt/IPv6 Reverse Routing Header and its application to Mobile Networks (http://www.ietf.org/internet-drafts/)
- draft-ietf-mobileip-mipv6-ha-ipsec-??.txt/Using IPsec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents (http://www.ietf.org/internet-drafts/)
- draft-suh-rmm-??.txt/Regional Mobile IPv6 mobility management (http://www.ietf.org/internet-drafts/)
- draft-mccann-mobileip-ipv6mipv4-??.txt/IPv6 over Mobile IPv4 (http://www.ietf.org/internet-drafts/)
- draft-kempf-mobileip-fmipv6-sem-??.txt/Improving the Architectural Alignment for FMIPv6 (http://www.ietf.org/internet-drafts/)
- draft-le-aaa-mipv6-requirements-??.txt/Mobile IPv6 Authentication, Authorization, and Accounting Requirements (http://www.ietf.org/internet-drafts/)
- draft-hwang-rohc-mipv6-??.txt/RObust Header Compression (ROHC): A Compression Profile for Mobile IPv6 (http://www.ietf.org/internet-drafts/)
- Cellular IP (http://comet.ctr.columbia.edu/cellularip/) on Columbia University, USA
- Mobile Networking Through Mobile IP (http://www.computer.org/internet/v2n1/perkins.htm) by Charles E. Perkins, Sun Microsystems
- LANCASTER MOBILE IPv6 PACKAGE (http://www.cs-ipv6.lancs.ac.uk/ipv6/MobileIP/)
- Testbed for MIND project on IPv6 (http://gsyc.escet.urjc.es/Mobiquo/Mind/documentacion/MontajeMaquetaIPv6_en/MontajeMaquetaIPv6_en.html)
- HS247/Mobile IP Web Links (http://hs247.com/modules.php?name=Web_Links&l_op=viewlink&cid=16)
- Mobile IPv6 Issue List (http://www.piuha.net/~jarkko/publications/mipv6/MIPv6-Issues.html)
- Hop-by-Hop Local Mobility Agents Probing for Mobile IPv6 (http://www.ietf.org/internet-drafts/draft-vriz-mobileip-hbhlmap-01.txt)
- IPv6 Reverse Routing Header and its application to Mobile Networks (http://www.ietf.org/internet-drafts/draft-thubert-nemo-reverse-routing-header-01.txt)
- Using IPsec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents (http://www.ietf.org/internet-drafts/draft-ietf-mobileip-mipv6-ha-ipsec-01.txt)

- Regional Mobile IPv6 mobility management (http://www.ietf.org/internet-drafts/draft-suh-rmm-00.txt)
- IPv6 over Mobile IPv4 (http://www.ietf.org/internet-drafts/draft-mccann-mobileip-ipv6mipv4-03.txt)
- Improving the Architectural Alignment for FMIPv6 (http://www.ietf.org/internet-drafts/draft-kempf-mobileip-fmipv6-sem-00.txt)
- Mobile IPv6 Authentication, Authorization, and Accounting Requirements (http://www.ietf.org/internet-drafts/draft-le-aaa-mipv6-requirements-01.txt)
- RObust Header Compression (ROHC): A Compression Profile for Mobile IPv6 (http://www.ietf.org/internet-drafts/draft-hwang-rohc-mipv6-00.txt)
- Cellular IP (http://comet.ctr.columbia.edu/cellularip/) all Columbia University, USA
- Mobile Networking Through Mobile IP (http://www.computer.org/internet/v2n1/perkins.htm) di Charles E. Perkins, Sun Microsystems
- LANCASTER MOBILE IPv6 PACKAGE (http://www.cs-ipv6.lancs.ac.uk/ipv6/MobileIP/)
- Testbed for MIND project on IPv6 (http://gsyc.escet.urjc.es/Mobiquo/Mind/documentacion/MontajeMaquetaIPv6_en/MontajeMaquetaIPv6_en.html)
- HS247/Mobile IP Web Links (http://hs247.com/modules.php?name=Web_Links&l_op=viewlink&cid=16)

Capitolo 16. Firewalling

Il firewalling IPv6 è importante, specialmente se si usa IPv6 su reti interne con indirizzi globali. Poiché, diversamente dalle reti IPv4 dove i comuni host interni sono protetti automaticamente utilizzando indirizzi IPv4 privati come RFC 1918 / Address Allocation for Private Internets (http://www.faqs.org/rfcs/rfc1918.html) o indirizzamento IP privato automatico (APIPA) ricerca con Google: Microsoft + APIPA (http://www.google.com/search?q=apipa+microsoft), in IPv6 gli indirizzi globali sono normalmente utilizzati e qualcuno con una connessione IPv6 potrebbe raggiungere tutti i nodi interni abilitati IPv6.

16.1. Firewalling utilizzando netfilter6

Il firewalling IPv6 nativo è supportato solo nei kernel versione 2.4+. Nei vecchi 2.2- si può solo filtrare IPv6-in-IPv4 con il protocollo 41.

Attenzione: non c'è nessuna garanzia che le regole qui descritte o gli esempi riportati possano veramente proteggere il proprio sistema!

Si faccia un controllo accurato del proprio insieme di regole dopo l'installazione, si veda la Sezione 17.3 per maggiori dettagli.

Si noti inoltre che il progetto USAGI attualmente sta lavorando per finire il tracciamento della connessione per IPv6! Questo renderà, in futuro, l'insieme delle regole più semplice e sicuro!

16.1.1. Maggiori informazioni

- Progetto netfilter (http://www.netfilter.org/)
- archivio della mailing list utenti netfilter (http://lists.samba.org/pipermail/netfilter/)
- archivio della mailing list sviluppatori netfilter (http://lists.samba.org/pipermail/netfilter-devel/)
- Stato non ufficiale delle informazioni (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-kernel.html#netfilter6)

16.2. Preparazione

16.2.1. Procurarsi i sorgenti

Procurarsi gli ultimi sorgenti del kernel: http://www.kernel.org/

Procurarsi l'ultimo pacchetto iptables:

- Tarball del sorgente (per le patch del kernel): http://www.netfilter.org/
- Sorgente RPM per il rebuild dei binari (per sistemi RedHat): ftp://ftp.redhat.com/redhat/linux/rawhide/SRPMS/SRPMS/ o probabilmente anche presso http://www.netcore.fi/pekkas/linux/ipv6/ (http://www.netcore.fi/pekkas/linux/ipv6/)

16.2.2. Estrazione dei file

Impostare la directory di lavoro con quella che contiene i file scaricati:

```
# cd /path/to/src
```

Spacchettare e rinominare i sorgenti del kernel

```
# tar z|jxf kernel-version.tar.gz|bz2
# mv linux linux-version-iptables-version+IPv6
```

Spacchettare i sorgenti di iptables

```
# tar z|jxf iptables-version.tar.gz|bz2
```

16.2.3. Applicare le ultime patch relative a iptables/IPv6 ai sorgenti del kernel

Cambiare la directory di lavoro con quella relativa ad iptables

```
# cd iptables-version
```

Applicare le patch pendenti

```
# make pending-patches KERNEL_DIR=/path/to/src/linux-version-iptables-version/
```

Applicare le patch addizionali relative a IPv6 (non ancora incluse nel kernel vanilla)

```
# make patch-o-matic KERNEL_DIR=/path/to/src/linux-version-iptables-version/
```

Rispondere sì alle seguenti opzioni (iptables-1.2.2)

- ah-esp.patch
- masq-dynaddr.patch (necessaria solo per sistemi con IP dinamico assegnato a connessioni WAN come PPP o PPPoE)
- ipv6-agr.patch.ipv6
- ipv6-ports.patch.ipv6
- LOG.patch.ipv6
- REJECT.patch.ipv6

Verifica estensioni IPv6

```
# make print-extensions
Extensions found: IPv6:owner IPv6:limit IPv6:mac IPv6:multiport
```

16.2.4. Configurazione, compilazione e installazione del nuovo kernel

Cambiare la directory di lavoro con quella contenente i sorgenti del kernel

```
# cd /path/to/src/linux-version-iptables-version/
```

Modifica del Makefile

```
- EXTRAVERSION =
+ EXTRAVERSION = -iptables-version+IPv6-try
```

Lanciare configure, abilitare le voci relative a IPv6

```
Code maturity level options
      Prompt for development and/or incomplete code/drivers: yes
Networking options
      Network packet filtering: yes
      The IPv6 protocol: module
           IPv6: Netfilter Configuration
                 IP6 tables support: module
                 All new options like following:
                       limit match support: module
                       MAC address match support: module
                       Multiple port match support: module
                       Owner match support: module
                       netfilter MARK match support: module
                       Aggregated address check: module
                       Packet filtering: module
                            REJECT target support: module
                            LOG target support: module
                       Packet mangling: module
                       MARK target support: module
```

Configurare anche le altre voci relative al proprio sistema.

Compilazione e installazione: si veda la sezione kernel qui e gli altri HOWTO.

16.2.5. Rebuild ed installazione dei binari di iptables

Ci si assicuri che l'albero dei sorgenti del kernel sia disponibile anche tramite /usr/src/linux/

Rinominare la vecchia directory

```
# mv /usr/src/linux /usr/src/linux.old
```

Creare un nuovo link simbolico

```
# ln -s /path/to/src/linux-version-iptables-version /usr/src/linux
```

Rebuild del pacchetto RPM

```
# rpm --rebuild /path/to/SRPMS/iptables-version-release.src.rpm
```

Installare il nuovo pacchetto iptables (iptables + iptables-ipv6)

- Sui sistemi RH 7.1 di solito c'è già una vecchia versione installata, perciò si può usare l'opzione "freshen"

```
# rpm -Fhv /path/to/RPMS/cpu/iptables*-version-release.cpu.rpm
```

- Se non fosse già installato, si usi l'opzione "install"

```
# rpm -ihv /path/to/RPMS/cpu/iptables*-version-release.cpu.rpm
```

- Normalmente sui sistemi RH 6.2 non è installato nessun kernl 2.4.x, perciò i requisiti non vanno bene. Si usi l'opzione "--nodeps" per installarlo

```
# rpm -ihv --nodeps /path/to/RPMS/cpu/iptables*-version-release.cpu.rpm
```

Probabilmente può essere necessario creare un link simbolico per le librerie iptables dove il programma possa cercarle

```
# ln -s /lib/iptables/ /usr/lib/iptables
```

16.3. Utilizzo

16.3.1. Verifica del supporto

Caricare il modulo, se così compilato

```
# modprobe ip6_tables
```

Verifica della capacità

```
# [ ! -f /proc/net/ip6_tables_names ] && echo "Current kernel doesn't support
¬ 'ip6tables' firewalling (IPv6)!"
```

16.3.2. Imparare ad usare ip6tables

16.3.2.1. Elencare di tutte le voci netfilter IPv6

- Elenco ridotto

```
# ip6tables -L
```

- Elenco esteso

```
# ip6tables -n -v --line-numbers -L
```

16.3.2.2. Elencare un filtro specifico

```
# ip6tables -n -v --line-numbers -L INPUT
```

16.3.2.3. Inserire una regola di log con opzioni sul filtro di ingresso (input)

```
# ip6tables --table filter --append INPUT  -j LOG --log-prefix "INPUT:"
¬ --log-level 7
```

16.3.2.4. Inserire una regola drop sul filtro di ingresso (input)

```
# ip6tables --table filter --append INPUT  -j DROP
```

16.3.2.5. Cancellare una regola tramite il suo numero

```
# ip6tables --table filter --delete INPUT 1
```

16.3.2.6. Permettere il traffico ICMPv6

Utilizzando vecchi kernel (kernel 2.4.5 non patchati e iptables-1.2.2) nessun tipo può essere specificato

- Accettare i pacchetti ICMPv6 entranti attraverso i tunnel

```
# ip6tables -A INPUT -i sit+ -p icmpv6 -j ACCEPT
```

- Permettere l'uscita dei pacchetti ICMPv6 attraverso i tunnel

```
# ip6tables -A OUTPUT -o sit+ -p icmpv6 -j ACCEPT
```

I kernel più recenti permettono di specificare i tipi dei pacchetti ICMPv6:

```
# ip6tables -A INPUT -p icmpv6 --icmpv6-type echo-request -j ACCEPT
```

16.3.2.7. Limitare la frequenza di ricezione dei pacchetti

Poiché può accadere (come l'autore ha già avuto modo di vedere) che possano verificarsi tempeste di pacchetti ICMPv6, si dovrebbe sfruttare la possibilità di limitare la frequenza di ricezione (rate) almeno per i pacchetti ICMPv6. In aggiunta anche la frequenza del logging dovrebbe essere limitata per prevenire attacchi DoS verso il syslog e la partizione di memorizzazione del file di log. Un esempio di limitazione del rate ICMPv6 è:

```
# ip6tables -A INPUT --protocol icmpv6 --icmpv6-type echo-request -j ACCEPT --match limit --limit 30/
```

16.3.2.8. Permettere SSH in ingresso

Qui viene mostrato un esempio per un insieme di regole che permette connessioni SSH in ingresso, da uno specifico indirizzo IPv6

- Permettere SSH entrante da 3ffe:ffff:100::1/128

```
# ip6tables -A INPUT -i sit+ -p tcp -s 3ffe:ffff:100::1/128 --sport 512:65535
¬ --dport 22 -j ACCEPT
```

- Permettere pacchetti di risposta (al momento il tracking della connessione in IPv6 non fa parte dell'implementazione principale di netfilter6)

```
# ip6tables -A OUTPUT -o sit+ -p tcp -d 3ffe:ffff:100::1/128 --dport 512:65535
¬ --sport 22 ! --syn j ACCEPT
```

16.3.2.9. Abilitare il traffico tunnel IPv6-in-IPv4

Per accettare pacchetti tunnel IPv6-in-IPv4, si devono inserire delle regole relative a questi pacchetti nel proprio firewall IPv4, per esempio

- Accettare il traffico entrante IPv6-in-IPv4 sull'interfaccia ppp0

```
# iptables -A INPUT -i ppp0 -p ipv6 -j ACCEPT
```

- Permettere l'uscita del traffico IPv6-in-IPv4 sull'interfaccia ppp0

```
# iptables -A OUTPUT -o ppp0 -p ipv6 -j ACCEPT
```

Se si ha solo un tunnel statico, si può specificare anche l'indirizzo IPv4

- Accettare il traffico entrante IPv6-in-IPv4 sull'interfaccia ppp0 che arriva dall'estremo del tunnel 1.2.3.4

```
# iptables -A INPUT -i ppp0 -p ipv6 -s 1.2.3.4 -j ACCEPT
```

- Permettere l'uscita del traffico IPv6-in-IPv4 sull'interfaccia ppp0 verso l'estremo del tunnel 1.2.3.4

```
# iptables -A OUTPUT -o ppp0 -p ipv6 -d 1.2.3.4 -j ACCEPT
```

16.3.2.10. Protezione dalle richieste di connessione TCP entranti

MOLTO RACCOMANDATO! Per ragioni di sicurezza si dovrebbe assolutamente inserire una regola che blocchi le richieste di connessione TCP entranti. Si adatti l'opzione "-i" se sono in uso altri nomi di interfaccia!

- Bloccare le richieste di connessione TCP entranti a questo host

```
# ip6tables -I INPUT -i sit+ -p tcp --syn -j DROP
```

- Bloccare le richieste di connessione TCP entranti agli host dietro a questo router

```
# ip6tables -I FORWARD -i sit+ -p tcp --syn -j DROP
```

Può darsi che queste regole vadano posizionate successivamente ad altre, ma è comunque una cosa alla quale bisogna pensare. Il miglior sistema è creare uno script ed eseguire le regole secondo un ordine stabilito.

16.3.2.11. Protezione dalle richieste UDP entranti

ANCHE QUESTO RACCOMANDATO! Come menzionato nelle informazioni sul firewall, è possibile controllare le porte sulle sessioni UDP/TCP uscenti. In questo modo, se tutti i propri sistemi locali IPv6 usano porte locali, cioè da 32768 a 60999, c'è anche la possibilità di filtrare le connessioni UDP (finché il tracking non sarà funzionante) con:

- Bloccare i pacchetti UDP entranti che non sono risposte a richieste inviate da questo host

```
# ip6tables -I INPUT -i sit+ -p udp ! --dport 32768:60999 -j DROP
```

- Bloccare i pacchetti UDP entranti che non sono risposte a richieste inoltrate dagli host dietro a questo router

```
# ip6tables -I FORWARD -i sit+ -p udp ! --dport 32768:60999 -j DROP
```

16.3.3. Esempio dimostrativo

Le seguenti righe mostrano una configurazione più sofisticata come esempio. Buona creazione di regole netfilter6....

```
# ip6tables -n -v -L
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 extIN      all      sit+   *       ::/0                 ::/0
    4   384 intIN      all      eth0   *       ::/0                 ::/0
    0     0 ACCEPT     all      *      *       ::1/128              ::1/128
    0     0 ACCEPT     all      lo     *       ::/0                 ::/0
    0     0 LOG        all      *      *       ::/0                 ::/0
¬        LOG flags 0 level 7 prefix ‘INPUT-default:’
    0     0 DROP       all      *      *       ::/0                 ::/0

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 int2ext    all      eth0   sit+    ::/0                 ::/0
    0     0 ext2int    all      sit+   eth0    ::/0                 ::/0
    0     0 LOG        all      *      *       ::/0                 ::/0
¬        LOG flags 0 level 7 prefix ‘FORWARD-default:’
    0     0 DROP       all      *      *       ::/0                 ::/0

Chain OUTPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 extOUT     all      *      sit+    ::/0                 ::/0
    4   384 intOUT     all      *      eth0    ::/0                 ::/0
    0     0 ACCEPT     all      *      *       ::1/128              ::1/128
    0     0 ACCEPT     all      *      lo      ::/0                 ::/0
    0     0 LOG        all      *      *       ::/0                 ::/0
¬        LOG flags 0 level 7 prefix ‘OUTPUT-default:’
    0     0 DROP       all      *      *       ::/0                 ::/0

Chain ext2int (1 references)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 ACCEPT     icmpv6    *       *       ::/0                 ::/0
    0     0 ACCEPT     tcp      *      *       ::/0                 ::/0
¬        tcp spts:1:65535 dpts:1024:65535 flags:!0x16/0x02
    0     0 LOG        all      *      *       ::/0                 ::/0
¬        LOG flags 0 level 7 prefix ‘ext2int-default:’
    0     0 DROP       tcp      *      *       ::/0                 ::/0
    0     0 DROP       udp      *      *       ::/0                 ::/0
    0     0 DROP       all      *      *       ::/0                 ::/0
```

```
Chain extIN (1 references)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 ACCEPT     tcp      *      *       3ffe:400:100::1/128  ::/0
¬         tcp spts:512:65535 dpt:22
    0     0 ACCEPT     tcp      *      *       3ffe:400:100::2/128  ::/0
¬         tcp spts:512:65535 dpt:22
    0     0 ACCEPT     icmpv6    *      *       ::/0                 ::/0
    0     0 ACCEPT     tcp      *      *       ::/0                 ::/0
¬         tcp spts:1:65535 dpts:1024:65535 flags:!0x16/0x02
    0     0 ACCEPT     udp      *      *       ::/0                 ::/0
¬         udp spts:1:65535 dpts:1024:65535
    0     0 LOG        all      *      *       ::/0                 ::/0
¬         limit: avg 5/min burst 5 LOG flags 0 level 7 prefix ‘extIN-default:’
    0     0 DROP       all      *      *       ::/0                 ::/0

Chain extOUT (1 references)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 ACCEPT     tcp      *      *       ::/0
¬ 3ffe:ffff:100::1/128tcp spt:22 dpts:512:65535 flags:!0x16/0x02
    0     0 ACCEPT     tcp      *      *       ::/0
¬ 3ffe:ffff:100::2/128tcp spt:22 dpts:512:65535 flags:!0x16/0x02
    0     0 ACCEPT     icmpv6    *      *       ::/0                 ::/0
    0     0 ACCEPT     tcp      *      *       ::/0                 ::/0
¬         tcp spts:1024:65535 dpts:1:65535
    0     0 ACCEPT     udp      *      *       ::/0                 ::/0
¬         udp spts:1024:65535 dpts:1:65535
    0     0 LOG        all      *      *       ::/0                 ::/0
¬         LOG flags 0 level 7 prefix ‘extOUT-default:’
    0     0 DROP       all      *      *       ::/0                 ::/0

Chain int2ext (1 references)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 ACCEPT     icmpv6    *      *       ::/0                 ::/0
    0     0 ACCEPT     tcp      *      *       ::/0                 ::/0
¬         tcp spts:1024:65535 dpts:1:65535
    0     0 LOG        all      *      *       ::/0                 ::/0
¬         LOG flags 0 level 7 prefix ‘int2ext:’
    0     0 DROP       all      *      *       ::/0                 ::/0
    0     0 LOG        all      *      *       ::/0                 ::/0
¬         LOG flags 0 level 7 prefix ‘int2ext-default:’
    0     0 DROP       tcp      *      *       ::/0                 ::/0
    0     0 DROP       udp      *      *       ::/0                 ::/0
    0     0 DROP       all      *      *       ::/0                 ::/0

Chain intIN (1 references)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 ACCEPT     all      *      *       ::/0
¬ fe80::/ffc0::
```

```
    4   384 ACCEPT     all      *      *       ::/0                 ff02::/16

Chain intOUT (1 references)
 pkts bytes target     prot opt in     out     source               destination
¬
    0     0 ACCEPT     all      *      *       ::/0
¬ fe80::/ffc0::
    4   384 ACCEPT     all      *      *       ::/0                 ff02::/16
    0     0 LOG        all      *      *       ::/0                 ::/0
¬         LOG flags 0 level 7 prefix ‘intOUT-default:’
    0     0 DROP       all      *      *       ::/0                 ::/0
```

Capitolo 17. Sicurezza

17.1. Sicurezza del nodo

È caldamente raccomandata l'applicazione di tutte le patch disponibili e la disabilitazione dei servizi non necessari. Inoltre si colleghino i servizi solo agli indirizzi IPv4/IPv6 necessari e si installi un firewall locale.

Da completare con maggiori dettagli...

17.2. Limitazioni di accesso

Molti servizi usano la libreria tcp_wrapper per il controllo di accesso. Qui viene descritto l'uso di tcp_wrapper.

Da completare con maggiori dettagli...

17.3. Verifica sicurezza di IPv6

Attualmente non ci sono in giro strumenti comodi capaci di controllare un sistema sulla rete per problemi di sicurezza inerenti a IPv6. Né Nessus (http://www.nessus.org/) né nessun altro security scanner commerciale, per quanto ne so, è in grado di effettuare la scansione di indirizzi IPv6.

17.3.1. Implicazioni legali

ATTENZIONE: si abbia cura di effettuare la scansione solo sui propri sistemi o solo dopo aver ricevuto un ordine scritto, diversamente si potrebbe incorrere in problemi legali. SI CONTROLLINO DUE VOLTE gli indirizzi IPv6 di destinazione prima di avviare una scansione.

17.3.2. Verifica di sicurezza con netcat abilitato IPv6

Con il netcat abilitato IPv6 (si veda IPv6+Linux-status-apps/security-auditing (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-apps.html#security-auditing) per maggiori dettagli) si può eseguire un portscan tramite un apposito script che scorre un intervallo di porte, cattura banner e così via. Esempio di utilizzo:

```
# nc6 ::1 daytime
13 JUL 2002 11:22:22 CEST
```

17.3.3. Verifica di sicurezza con nmap abilitato IPv6

NMap (http://www.insecure.org/nmap/), uno dei migliori portscanner nel mondo, supporta IPv6 sin dalla versione 3.10ALPHA1. Esempio di utilizzo:

```
# nmap -6 -sT ::1
Starting nmap V. 3.10ALPHA3 ( www.insecure.org/nmap/ )
```

```
Interesting ports on localhost6 (::1):
(The 1600 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
53/tcp     open        domain
515/tcp    open        printer
2401/tcp   open        cvspserver
Nmap run completed -- 1 IP address (1 host up) scanned in 0.525 seconds
```

17.3.4. Verifica di sicurezza con strobe abilitato IPv6

Strobe è un portscanner a basso budget (rispetto a NMap), ma esiste tuttavia una patch disponibile per abilitare IPv6 (si veda IPv6+Linux-status-apps/security-auditing (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-apps.html#security-auditing) per maggiori dettagli). Esempio di utilizzo:

```
# ./strobe ::1 strobe 1.05 (c) 1995-1999 Julian Assange <proff@iq.org>.
::1 2401 unassigned unknown
::1 22 ssh Secure Shell - RSA encrypted rsh
::1 515 printer spooler (lpd)
::1 6010 unassigned unknown
::1 53 domain Domain Name Server
```

Nota: lo sviluppo di strobe non è andato molto avanti, il numero di versione mostrato non è quello giusto.

17.3.5. Risultati della verifica

Se il risultato di una verifica non coincide con la propria politica di sicurezza, si utilizzi il firewalling IPv6 per tappare i buchi, per esempio usando netfilter6 (si veda Firewalling/Netfilter6 per maggiori dettagli).

Info: Informazioni più dettagliate concernenti la sicurezza IPv6 si possono trovare qui:

- Firewalling Considerations for IPv6 / draft-savola-v6ops-firewalling-??.txt (http://www.ietf.org/internet-drafts/)
- IPv6 Neighbour Discovery trust models and threats / draft-ietf-send-psreq-??.txt (http://www.ietf.org/internet-drafts/)
- Security Considerations for 6to4 / draft-savola-v6ops-6to4-security-??.txt (http://www.ietf.org/internet-drafts/)
- Access Control Prefix Router Advertisement Option for IPv6 / draft-bellovin-ipv6-accessprefix-??.txt (http://www.ietf.org/internet-drafts/)
- Requirements for Plug and Play IPsec for IPv6 applications /draft-kobayakawa-ipsec-ipv6-pnpipsec-reqts-??.txt (http://www.ietf.org/internet-drafts/)
- Security of IPv6 Routing Header and Home Address Options / draft-savola-ipv6-rh-ha-security-??.txt (http://www.ietf.org/internet-drafts/)

Capitolo 18. Cifratura ed autenticazione

Diversamente da IPv4 la cifratura e l'autenticazione sono funzioni vincolanti di IPv6. Queste vengono normalmente implementate utilizzando IPsec (il quale può essere usato anche da IPv4).

A causa però dell'indipendenza della cifratura e dell'autenticazione dal protocollo di scambio dele chiavi, ci sono attualmente alcuni problemi di interoperabilità a questo proposito.

18.1. Supporto nel kernel

18.1.1. Supporto nel kernel Linux vanilla 2.4.x

Attualmente mancante nel vanilla 2.4. Esiste un problema nel tenere il sorgente del kernel Linux libero dalle leggi relative alla importazione/esportazione dei sistemi crittografici. Questo è uno dei casi per cui il Progetto FreeS/WAN (http://www.freeswan.org/) (IPsec solo per IPv4) non è ancora contenuto nel sorgente vanilla.

18.1.2. Supporto nel kernel USAGI

Il progetto USAGI ha prelevato, nel Luglio 2001, il codice FreeS/WAN IPv6 dall'IABG / IPv6 Project (http://www.ipv6.iabg.de/downloadframe/) e lo ha incluso nelle proprie estensioni del kernel. Attualmente stanno lavorando sull'implementazione di IPsec nei kernel serie 2.5.x.

18.1.3. Supporto nel kernel Linux vanilla 2.5.x

Attualmente, le estensioni IPsec del kernel USAGI stanno per essere migrate nei kernel di sviluppo 2.5.x.

18.2. Utilizzo

A differenza di FreeS/WAN, nell'IPsec del kernel 2.5.x il demone IKE "racoon" (preso da KAME) viene usato al posto di "pluto" di FreeS/WAN. Ha una sintassi di configurazione diversa da quella di "pluto", si noti inoltre che la configurazione di IPsec è divisa in 2 parti (IKE e la configurazione di topologia). Fino a quando non sarà inclusa documentazione qui, si dia uno sguardo a Linux Advanced Routing &Traffic Control HOWTO / IPSEC (http://lartc.org/howto/lartc.ipsec.html).

Capitolo 19. Qualità del servizio (QoS)

IPv6 supporta il QoS tramite l'uso delle etichette di flusso (Flow Label) e delle classi di traffico. Questo meccanismo può essere controllato con "tc" (contenuto nel pacchetto "iproute").

Informazioni addizionali:

- IPv6 Flow Label Specification (http://www.ietf.org/internet-drafts/draft-ietf-ipv6-flow-label-03.txt)

Da completare con maggiori contenuti...

Capitolo 20. Suggerimenti per demoni abilitati IPv6

Vengono mostrati alcuni suggerimenti per demoni abilitati IPv6.

20.1. Berkeley Internet Name Daemon BIND (named)

IPv6 è supportato dalla versione 9. Si utilizzi sempre la versione più recente disponibile. Deve essere utilizzata almeno la 9.1.3, le versioni più vecchie possono contenere buchi di sicurezza utilizzabili da remoto.

20.1.1. Ascoltare su indirizzi IPv6

Nota: diversamente da IPv4 le attuali versioni non permettono di legare il socket del server a indirizzi IPv6 dedicati, perciò sono validi solo (indirizzi) *qualsiasi* o *nessuno*. Poiché questo può rappresentare un problema di sicurezza, si controlli anche la sezione Access Control List (ACL) più avanti!

20.1.1.1. Abilitare BIND named per l'ascolto su un indirizzo IPv6

Per abilitare l'ascolto IPv6, bisogna cambiare le seguenti opzioni

```
options {
        # qui ci saranno sicuramente anche altre opzioni
        listen-on-v6 { any; };
};
```

Dopo il riavvio dovrebbe risultare qualcosa tipo

```
# netstat -lnptu |grep "named\W*$"
tcp 0 0 :::53         :::*      LISTEN 1234/named
# incoming TCP requests
udp 0 0 1.2.3.4:53    0.0.0.0:*        1234/named
¬ # incoming UDP requests to IPv4 1.2.3.4
udp 0 0 127.0.0.1:53  0.0.0.0:*        1234/named
¬ # incoming UDP requests to IPv4 localhost
udp 0 0 0.0.0.0:32868 0.0.0.0:*        1234/named
¬ # dynamic chosen port for outgoing queries
udp 0 0 :::53         :::*             1234/named
¬ # incoming UDP request to any IPv6
```

Un semplice test può essere

```
# dig localhost @::1
```

e dovrebbe visualizzare un risultato.

20.1.1.2. Disabilitare il BIND named per l'ascolto su un indirizzo IPv6

Per disabilitare l'ascolto IPv6, occorre cambiare le seguenti opzioni

```
options {
        # qui ci saranno sicuramente anche altre opzioni
        listen-on-v6 { none; };
};
```

20.1.2. Access Control List (ACL) di indirizzi IPv6 abilitati

È possibile definire le ACL relative a indirizzi IPv6 abilitati e queste dovrebbero essere usate praticamente sempre. Un esempio può essere:

```
acl internal-net {
        127.0.0.1;
        1.2.3.0/24;
        3ffe:ffff:100::/56;
        ::1/128;
        ::ffff:1.2.3.4/128;
};
acl ns-internal-net {
        1.2.3.4;
        1.2.3.5;
        3ffe:ffff:100::4/128;
        3ffe:ffff:100::5/128;
};
```

Queste ACL possono essere usate, per esempio, per le richieste (query) dei client e per il trasferimento delle zone a name server secondari. Ciò impedisce l'utilizzo del proprio name server di caching dall'esterno con IPv6.

```
options {
        # qui ci saranno sicuramente anche altre opzioni
        listen-on-v6 { none; };
        allow-query { internal-net; };
        allow-transfer { ns-internal-net; };
};
```

È anche possibile configurare le opzioni *allow-query* e *allow-transfer* per la maggior parte delle singole definizioni di zona.

20.1.3. Inviare query con indirizzo IPv6 dedicato

Quest'opzione non è richiesta, ma a volte necessaria:

```
query-source-v6 address <ipv6address|*> port <port|*>;
```

20.1.4. Indirizzi IPv6 dedicati per zona

È anche possibile definire alcuni indirizzi IPv6 per zona.

20.1.4.1. Indirizzo sorgente di trasferimento

L'indirizzo sorgente di trasferimento viene usato per i trasferimenti di zona uscenti:

```
transfer-source-v6 <ipv6addr|*> [port port];
```

20.1.4.2. Indirizzo sorgente di notifica

L'indirizzo sorgente di notifica viene usato per i messaggi di notifica uscenti:

```
notify-source-v6 <ipv6addr|*> [port port];
```

20.1.5. Esempi di file di zona IPv6

Alcune informazioni possono essere reperite anche da IPv6 DNS Setup Information (articolo) (http://www.isi.edu/~bmanning/v6DNS.html). Forse può essere d'aiuto anche il IPv6 Reverse DNS zone builder for BIND 8/9 (strumento web) (http://tools.fpsn.net/ipv6-inaddr/).

20.1.6. Servire dati DNS relativi a IPv6

Per IPv6 vengono definiti nuovi tipi e zone root per i reverse lookup:

- AAAA e il reverse IP6.INT: specificati in RFC 1886 / DNS Extensions to support IP version 6 (http://www.faqs.org/rfcs/rfc1886.html), utilizzabile a partire dalla versione di BIND 4.9.6
- A6, DNAME (ORA DISAPPROVATO!) e il reverse IP6.ARPA: specificati in RFC 2874 / DNS Extensions to Support IPv6 Address Aggregation and Renumbering (http://www.faqs.org/rfcs/rfc2874.html), utilizzabili a partire dalla versione di BIND 9 ma si vedano anche le ultime informazioni sullo stato attuale in draft-ietf-dnsext-ipv6-addresses-00.txt (http://www.ietf.org/internet-drafts/)

Forse verrà riempito con più contenuti, intanto si dia uno sguardo alle RFC segnalate e anche a

- AAAA e il reverse IP6.INT: IPv6 DNS Setup Information (http://www.isi.edu/~bmanning/v6DNS.html)
- A6, DNAME (ORA DISAPPROVATO!) e il reverse IP6.ARPA: si guardi nei capitoli 4 e 6 del'Administrator Reference Manual (ARM) del BIND 9, distribuito con il pacchetto bind o lo si prenda qui: BIND version 9 ARM (PDF) (http://www.nominum.com/content/documents/bind9arm.pdf)

Poiché IP6.INT è disapprovato (ma ancora in uso), un server DNS che supporta informazioni IPv6 deve essere in grado di fornire entrambi i tipi di reverse zone.

20.1.6.1. La miglior consuetudine attuale

Poiché ci sono alcuni problemi nell'uso di questi nuovi formati, la miglior consuetudine è:

Supporto della risoluzione diretta:

- AAAA

Supporto della risoluzione inversa (reverse lookup):

- Formato nibble inverso per la zona ip6.int (PER COMPATIBILITÀ CON IL PASSATO)
- Formato nibble inverso per la zona ip6.arpa (RACCOMANDATO)

20.1.7. Controllare la connessione IPv6

Per verificare se il BIND è in ascolto e se fornisce o meno dati su di un socket IPv6, si vedano i seguenti esempi.

20.1.7.1. Connessione IPv6 con accesso negato dall'ACL

Specificando un server dedicato per la query, una connessione può essere forzata con:

```
$ host -t aaaa www.6bone.net 3ffe:ffff:200:f101::1
Using domain server:
Name: 3ffe:ffff:200:f101::1
Address: 3ffe:ffff:200:f101::1#53
Aliases:
Host www.6bone.net. not found: 5(REFUSED)
```

La voce di log relativa è del tipo:

```
Jan 3 12:43:32 gate named[12347]: client
¬ 3ffe:ffff:200:f101:212:34ff:fe12:3456#32770:
 query denied
```

Se si vedono righe di questo tipo nel log, si controlli se le richieste di questi client debbano essere permesse o meno e si riveda quindi la propria configurazione dell'ACL.

20.1.7.2. Connessione IPv6 con successo

Una connessione IPv6 avviene con successo se l'output è analogo a:

```
$ host -t aaaa www.6bone.net 3ffe:ffff:200:f101::1
Using domain server:
Name: 3ffe:ffff:200:f101::1
Address: 3ffe:ffff:200:f101::1#53
Aliases:
www.6bone.net. is an alias for 6bone.net.
6bone.net. has AAAA address 3ffe:b00:c18:1::10
```

20.2. Internet super daemon (xinetd)

IPv6 è supportato da xinetd (http://www.xinetd.org/) a partitre dalla versione 1.8.9. Si utilizzi sempre la versione più recente disponibile. Deve essere utilizzata almeno la versione 2.3.3, versioni più vecchie possono avere buchi di sicurezza utilizzabili da remoto.

Alcune distribuzioni Linux contengono un pacchetto extra per xinetd IPv6, altre lanciano xinetd con le estensioni IPv6 soltanto se viene impostata la seguente variabile d'ambiente: NETWORKING_IPV6="yes", nella maggior parte dei casi questo viene fatto tramite /etc/sysconfig/network (nel caso di distribuzioni tipo Red Hat). Nelle distribuzioni più recenti, un unico file binario supporta sia IPv4 che IPv6.

Se si abilita un servizio incorporato, come per esempio daytime, modificando il file di configurazione /etc/xinetd.d/daytime come

```
# diff -u /etc/xinetd.d/daytime.orig /etc/xinetd.d/daytime
--- /etc/xinetd.d/daytime.orig Sun Dec 16 19:00:14 2001
+++ /etc/xinetd.d/daytime Sun Dec 16 19:00:22 2001
@@ -10,5 +10,5 @@
        protocol = tcp
        user = root
        wait = no
-       disable = yes
+       disable = no
 }
```

Dopo il riavvio di xinetd si dovrebbe ottenere un risultato positivo del tipo:

```
# netstat -lnptu -A inet6 |grep "xinetd*"
tcp 0 0 ::ffff:192.168.1.1:993  :::*  LISTEN  12345/xinetd-ipv6
tcp 0 0 :::13                   :::*  LISTEN  12345/xinetd-ipv6 <- service
¬ daytime/tcp
tcp 0 0 ::ffff:192.168.1.1:143  :::*  LISTEN  12345/xinetd-ipv6
```

L'esempio riportato mostra uno xinetd che ascolta su IMAP e IMAP-SSL IPv4.

Nota: le versioni precedenti hanno il problema che xinet abilitato solo per IPv4 non si avvia su di un nodo abilitato per IPv6 ed anche un xinetd abilitato per IPv6 non si avvia su di un nodo abilitato solo per IPv4. Tale problema è stato risolto nelle ultime versioni, perlomeno dalla versione 2.3.11.

20.3. Server Web Apache2 (httpd2)

Il server web Apache supporta IPv6 in maniera nativa, grazie ai manutentori, dalla versione 2.0.14. Le patch disponibili per i più vecchi 1.3.x non sono aggiornate e non dovrebbero essere usate in ambiente pubblico, ma sono comunque disponibili presso KAME / Misc (ftp://ftp.kame.net/pub/kame/misc/).

20.3.1. Ascolto su indirizzi IPv6

Nota: gli host virtuali su indirizzi IPv6 non funzionano nelle versioni fino alla 2.0.28 (è disponibile una patch per la 2.0.28). Si provi sempre prima l'ultima versione disponibile poiché quelle precedenti hanno qualche problema di sicurezza.

20.3.1.1. Host virtuale in ascolto solo su indirizzo IPv6

```
Listen [3ffe:ffff:100::1]:80
<VirtualHost [3ffe:ffff:100::1]:80>
        ServerName ipv6only.yourdomain.yourtopleveldomain
        # ... sicuramente piu' linee di configurazione
</VirtualHost>
```

20.3.1.2. Host virtuale in ascolto su indirizzo IPv6 e IPv4

```
Listen [3ffe:ffff:100::2]:80
Listen 1.2.3.4:80
<VirtualHost [3ffe:ffff:100::2]:80 1.2.3.4:80>
        ServerName ipv6andipv4.yourdomain.yourtopleveldomain
        # ... sicuramente piu' linee di configurazione
</VirtualHost>
```

Questo, dopo il riavvio, dovrebbe dare un risultato analogo a

```
# netstat -lnptu |grep "httpd2\W*$"
tcp 0 0 1.2.3.4:80          0.0.0.0:* LISTEN 12345/httpd2
tcp 0 0 3ffe:ffff:100::1:80 :::*      LISTEN 12345/httpd2
tcp 0 0 3ffe:ffff:100::2:80 :::*      LISTEN 12345/httpd2
```

Per un semplice test si può usare l'esempio di telnet già mostrato.

20.3.1.3. Note addizionali

- Apache2 supporta un metodo chiamato "sendfile" per accelerare la fornitura di dati. Alcuni driver di NIC supportano inoltre il checksum offline. In alcuni casi, ciò può portare a problemi di connessione e checksum TCP non validi. In questi frangenti, si disabiliti "sendfile" ricompilando con l'opzione di configurazione "--without-sendfile" o utilizzando la direttiva "EnableSendfile off" nel file di configurazione.

20.4. Router Advertisement Daemon (radvd)

Il router advertisement daemon è molto utile in una LAN, se i client devono essere configurati automaticamente. Il demone stesso dovrebbe girare sul gateway di default IPv6 (non è obbligatorio che questo sia anche il gateway di default IPv4, perciò si faccia attenzione a chi invia, sulla propria LAN, gli avvisi di instradamento).

Si possono specificare alcune informazioni e flag che devono essere contenute negli avvisi. Le più comunemente usate sono

- Prefisso (necessario)

- Tempo di vita del prefisso
- Frequenza dell'invio degli avvisi (opzionale)

Dopo un'appropriata configurazione, il demone invia gli avvisi attraverso le interfacce specificate, i client - si spera - le ricevono e configurano automaticamente gli indirizzi con i prefissi e l'instradamento di default ricevuti.

20.4.1. Configurazione di radvd

20.4.1.1. Configurazione semplice

Di solito, il file di configurazione di radvd è /etc/radvd.conf. Un semplice esempio è riportato di seguito:

```
interface eth0 {
        AdvSendAdvert on;
        MinRtrAdvInterval 3;
        MaxRtrAdvInterval 10;
        prefix 3ffe:ffff:0100:f101::/64 {
                AdvOnLink on;
                AdvAutonomous on;
                AdvRouterAddr on;
        };
};
```

Dal lato client ciò risulta come

```
# ip -6 addr show eth0
3: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 100
    inet6 3ffe:ffff:100:f101:2e0:12ff:fe34:1234/64 scope global dynamic
       valid_lft 2591992sec preferred_lft 604792sec
    inet6 fe80::2e0:12ff:fe34:1234/10 scope link
```

Poiché il tempo di vita non è stato definito, viene indicato un valore molto elevato.

20.4.1.2. Configurazione speciale 6to4

Dalla versione 0.6.2pl3 è supportata la (ri)-generazione automatica del prefisso dipendentemente dall'indirizzo IPv4 dell'interfaccia specificata. Questo meccanismo può essere utilizzato per distribuire avvisi in una LAN dopo che il tunneling 6to4 subisce variazioni. Il più delle volte viene utilizzato dietro ad un router Linux dial-on-demand dinamico. Poiché il tempo di vita di questo prefisso è sicuramente molto breve (dopo ogni dial-up c'è un diverso prefisso valido), esso viene configurato al valore minimo:

```
interface eth0 {
        AdvSendAdvert on;
        MinRtrAdvInterval 3;
        MaxRtrAdvInterval 10;
        prefix 0:0:0:f101::/64 {
                AdvOnLink off;
                AdvAutonomous on;
                AdvRouterAddr on;
```

```
                Base6to4Interface ppp0;
                AdvPreferredLifetime 20;
                AdvValidLifetime 30;
        };
};
```

Lato client questo si traduce in (assumendo che ppp0 abbia attualmente l'indirizzo locale IPv4 1.2.3.4):

```
# /sbin/ip -6 addr show eth0
3: eth0: <BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast qlen 100
   inet6 2002:0102:0304:f101:2e0:12ff:fe34:1234/64 scope global dynamic
      valid_lft 22sec preferred_lft 12sec
   inet6 fe80::2e0:12ff:fe34:1234/10 scope link
```

Poiché è stato definito un tempo di vita molto piccolo, tale prefisso verrà velocemente scartato se non saranno ricevuti avvisi relativi ad esso.

Nota aggiuntiva: se non si è usato lo speciale supporto 6to4 negli initscript, è necessario creare un instradamento speciale sull'interfaccia interna del router, altrimenti si avranno alcuni problemi di instradamento di ritorno. Per l'esempio mostrato qui:

```
# /sbin/ip -6 route add 2002:0102:0304:f101::/64 dev eth0 metric 1
```

questo instradamento deve essere rimpiazzato ogni volta che il prefisso cambia, come nel caso in cui all'interfaccia dial-up viene assegnato un nuovo indirizzo IPv4.

20.4.2. Debugging

Un programma chiamato "radvdump" può essere d'aiuto per guardare negli avvisi ricevuti e inviati. Semplice da usare:

```
# radvdump
Router advertisement from fe80::280:c8ff:feb9:cef9 (hoplimit 255)
        AdvCurHopLimit: 64
        AdvManagedFlag: off
        AdvOtherConfigFlag: off
        AdvHomeAgentFlag: off
        AdvReachableTime: 0
        AdvRetransTimer: 0
        Prefix 2002:0102:0304:f101::/64
                AdvValidLifetime: 30
                AdvPreferredLifetime: 20
                AdvOnLink: off
                AdvAutonomous: on
                AdvRouterAddr: on
        Prefix 3ffe:ffff:100:f101::/64
                AdvValidLifetime: 2592000
                AdvPreferredLifetime: 604800
```

```
          AdvOnLink: on
          AdvAutonomous: on
          AdvRouterAddr: on
      AdvSourceLLAddress: 00 80 12 34 56 78
```

L'output mostra ogni pacchetto di avviso ricevuto in formato leggibile. Si dovrebbero vedere i propri valori configurati, altrimenti forse non è il proprio radvd che invia gli avvisi... si cerchi un altro router sul link (e si consideri per il tracciamento il valore LLAddress che corrisponde all'indirizzo MAC).

20.5. tcp_wrapper

il tcp_wrapper è una libreria che può essere d'aiuto per proteggere i servizi dagli abusi.

20.5.1. Capacità di filtraggio

Si può utilizzare tcp_wrapper per

- Filtrare in base all'indirizzo sorgente (IPv4 o IPv6)
- Filtrare in base all'utente (richiede un demone ident in esecuzione sul client)

20.5.2. Quali programmi usano tcp_wrapper

Sono conosciuti i seguenti:

- Ogni servizio chiamato da xinetd (se xinetd viene compilato per utilizzare la libreria tcp_wrapper)
- sshd (se compilato per utilizzare tcp_wrapper)

20.5.3. Utilizzo

tcp_wrapper è controllato da due file chiamati /etc/hosts.allow e /etc/hosts.deny. Per maggiori informazioni si veda

```
$ man hosts.allow
```

20.5.3.1. Esempio del file /etc/hosts.allow

Ogni servizio che debba essere filtrato in maniera positiva (cioè le cui connnessioni sono accettate) ha bisogno di una riga all'interno di questo file.

```
sshd:           1.2.3. [3ffe:ffff:100:200::]/64
daytime-stream: 1.2.3. [3ffe:ffff:100:200::]/64
```

Nota: ci sono in giro implementazioni difettose, che utilizzano la seguente errata descrizione di rete IPv6: [3ffe:ffff:100:200::/64]. Si spera che queste versioni vengano corrette al più presto.

20.5.3.2. Esempio del file /etc/hosts.deny

Questo file contiene tutti le voci di filtraggio negativo e di solito dovrebbe negare tutto utilizzando

```
ALL: ALL
```

Se il nodo è particolarmente sensibile, si può rimpiazzare la linea standard indicata sopra, con quella indicata qui di seguito, ciò però può causare un attacco DoS (a carico del mailer e della directory di spool) nel caso in cui vengano fatte molte connessioni in poco tempo. Un logwatch forse potrebbe essere una soluzione migliore per questo problema.

```
ALL: ALL: spawn (echo "Attempt from %h %a to %d at `date`"
 | tee -a /var/log/tcp.deny.log | mail root@localhost)
```

20.5.4. Logging

A seconda della voce presente nel file di configurazione del demone syslog, /etc/syslog.conf, il tcp_wrapper solitamente scrive i log nel file /var/log/secure.

20.5.4.1. Connessione rifiutata

Il rifiuto di una connessione IPv4 verso un servizio daytime, gestito tramite xinetd, produce una riga simile a quella dell'esempio seguente

```
Jan 2 20:40:44 gate xinetd-ipv6[12346]: FAIL: daytime-stream libwrap
¬ from=::ffff:1.2.3.4
Jan 2 20:32:06 gate xinetd-ipv6[12346]: FAIL: daytime-stream libwrap
 from=3ffe:ffff:100:200::212:34ff:fe12:3456
```

Il rifiuto di una connessione IPv4 verso un servizio sshd in doppio ascolto, produce una riga simile a quella dell'esempio seguente

```
Jan 2 20:24:17 gate sshd[12345]: refused connect from ::ffff:1.2.3.4
¬ (::ffff:1.2.3.4)
Jan 2 20:39:33 gate sshd[12345]: refused connect
 from 3ffe:ffff:100:200::212:34ff:fe12:3456
¬ (3ffe:ffff:100:200::212:34ff:fe12:3456)
```

20.5.4.2. Connessione permessa

L'accettazione di una connessione IPv4 verso un servizio daytime, gestito tramite xinetd, produce una riga simile a quella del seguente esempio

```
Jan 2 20:37:50 gate xinetd-ipv6[12346]: START: daytime-stream pid=0
¬ from=::ffff:1.2.3.4
Jan 2 20:37:56 gate xinetd-ipv6[12346]: START: daytime-stream pid=0
```

```
 from=3ffe:ffff:100:200::212:34ff:fe12:3456
```

L'accettazione di una connessione IPv4, verso un servizio sshd in doppio ascolto, produce una riga simile a quella del seguente esempio

```
Jan 2 20:43:10 gate sshd[21975]: Accepted password for user from ::ffff:1.2.3.4
¬ port 33381 ssh2
Jan 2 20:42:19 gate sshd[12345]: Accepted password for user
 from 3ffe:ffff:100:200::212:34ff:fe12:3456 port 33380 ssh2
```

20.6. vsftpd

20.6.1. Stare in ascolto su indirizzi IPv6

Si modifichi il file di configurazione, di solito /etc/vsftpd/vsftpd.conf, aggiustando l'opzione di ascolto così

```
listen_ipv6=yes
```

E' tutto.

20.7. proftpd

20.7.1. Stare in ascolto su indirizzi IPv6

Si modifichi il file di configurazione, di solito /etc/proftpd.conf, ma si faccia attenzione, non ha senso al 100% nel caso di installazione con host virtuale

```
<VirtualHost 192.0.2.1>
        ...
        Bind 2001:0DB8::1
        ...
</VirtualHost>
```

E' tutto.

20.8. Altri Demoni

Al giorno d'oggi è piuttosto semplice, basta cercare un opzione della linea di comando o un'opportuna direttiva di configurazione per abilitare l'ascolto su IPv6. Si veda la pagina di manuale (man) del demone o si controllino le relative FAQ. Può capitare che si possa associare un demone esclusivamente all’indirizzo IPv6-"any" (::) piuttosto che ad un indirizzo IPv6 dedicato, a causa della scarsità del supporto (dipende da cosa è stato implementato dal programmatore...).

Capitolo 21. Programmazione (uso delle API)

Non ho esperienza nella programmazione IPv6, forse questo capitolo sarà completato da altri o spostato in un altro HOWTO.

Maggiori informazioni possono essere trovate qui:

- RFC 2553 / Basic Socket Interface Extensions for IPv6 (http://www.faqs.org/rfcs/rfc2553.html)
- Draft / Advanced Sockets API for IPv6 / draft-ietf-ipngwg-rfc2292bis-XY.txt (http://www.ietf.org/internet-drafts/)
- Porting applications to IPv6 HowTo (http://jungla.dit.upm.es/~ecastro/IPv6-web/ipv6.html) di Eva M. Castro

Capitolo 22. Interoperabilità

In tutto il mondo esistono alcuni progetti che controllano l'interoperabilità tra i diversi sistemi operativi a proposito dell'implementazione delle funzioni di IPv6. Alcune URL:

- TAHI Project (http://www.tahi.org/)

A seguire maggiori contenuti...

Capitolo 23. Ulteriori informazioni e URL

23.1. Libri stampati, articoli, riviste online (misto)

23.1.1. Libri stampati (inglese)

23.1.1.1. Cisco

- Cisco Self-Study: Implementing IPv6 Networks (IPV6) (http://www.amazon.com/exec/obidos/tg/detail/-/1587050862/copernicshopper/103-1420916-1341420), di Regis Desmeules. Cisco Press; ISBN 1587050862; 500 pagine; prima edizione (11 Aprile 2003). Nota: sarà pubblicato il 11 Aprile 2003.
- Configuring IPv6 with Cisco IOS (http://www.amazon.com/exec/obidos/tg/detail/-/1928994849/copernicshopper/103-1420916-1341420), di Sam Brown, Sam Browne, Neal Chen, Robbie Harrell, Edgar, Jr. Parenti (Curatore), Eric Knipp (Curatore), Paul Fong (Curatore) 362 pagine; Syngress Media Inc; ISBN 1928994849; (12 Luglio 2002).

23.1.1.2. Generale

- IPv6 Essentials (http://www.sunny.ch/publications/f_ipv6.htm) di Silvia Hagen, Luglio 2002, O'Reilly Numero d'Ordine: 1258 (http://www.oreilly.com/catalog/ipv6ess/), ISBN 0-5960-0125-8, 352 pagine. TdC, Indice, Capitoli di esempio ecc. (http://www.oreilly.com/catalog/ipv6ess/); O'Reilly Pressrelease (http://press.oreilly.com/ipv6ess.html)
- IPv6: The New Internet Protocol, di Christian Huitema; pubblicato da Prentice-Hall; ISBN 0138505055. Descrizione: questo libro, scritto da Christian Huitema - un membro dell'InternetArchitecture Board, dà un'eccellente descrizione di IPv6, la differerenza con IPv4, quando e perché è stato sviluppato. Fonte: http://www.cs.uu.nl/wais/html/na-dir/internet/tcp-ip/resource-list.html
- IPv6 Networks (http://www.epinions.com/book_mu-3402412/display_~full_specs) di Niles Kitty; (ISBN 0070248079); 550 pagine; Pubblicato in data 05/01/1998.
- Implementing IPV6. Supporting the Next Generation Internet Protocols (http://www.amazon.com/exec/obidos/tg/detail/-/0764545892/copernicshopper/103-1420916-1341420) di P. E. Miller, Mark A. Miller; Editore: John Wiley & Sons; ISBN 0764545892; seconda edizione (15 Marzo 2000); 402 pagine.
- Big Book of IPv6 Addressing Rfcs (http://www.amazon.com/exec/obidos/tg/detail/-/0126167702/copernicshopper/103-1420916-1341420) di Peter H. Salus (Compilatore), Morgan Kaufmann Publishers, Aprile 2000, 450 pagine ISBN 0126167702.
- Understanding IPV6 (http://www.epinions.com/book_mu-3922588/display_~full_specs) di Davies, Joseph; ISBN 0735612455; Data di pubblicazione 05/01/2001; Numero di Pagine: 350. Understanding IPV6 (http://www.microsoft.com/MSPress/books/4883.asp) di Davies, Joseph; ISBN 0735612455; Data di Pubblicazione 13/11/2002; Numero di Pagine 544.

- Migrating to IPv6 - IPv6 in Practice (http://www.amazon.com/exec/obidos/tg/detail/-/0471498920/copernicshopper/103-1420916-1341420). Di Marc Blanchet Editore: John Wiley & Sons; ISBN 0471498920; Prima edizione (Novembre 2002); 368 pagine.
- Programming IPv6 (http://www.amazon.de/exec/obidos/ASIN/0672323443/qid%3D1038151777/028-1904439-6654140) di Sean Walton; 560 pagine; Addison-Wesley Professional; ISBN 0672323443 Nota: Sarà pubblicato il 1 Agosto 2003.
- Wireless boosting IPv6 (http://www.nwfusion.com/news/2000/1023ipv6.html) di Carolyn Duffy Marsan, 23/10/2000.
- Una ricerca sulla rete O'Reilly per la parola chiave IPv6 (http://www.oreillynet.com/search/index.ncsp?sp-q=IPv6) ha restituito 29 risultati (28 Gennaio 2002).

23.1.2. Libri stampati (tedesco)

- Technik der IP-Netze (TCP/IP incl. IPv6) bei Amazon.de (http://www.amazon.de/exec/obidos/ASIN/3446215018/) Anatol Badach, Erwin Hoffmann Carl Hanser Verlag München, Wien, 2001 ISBN 3-446-21501-8 Kap. 6: Protokoll IPv6 S.205-242 Kap. 7: Plug&Play-Unterstützung bei IPv6 S.243-276 Kap. 8: Migration zum IPv6-Einsatz S.277-294 Kap. 9.3.4: RIP für das Protokoll IPv6 (RIPng) S.349-351 Kap. 9.4.6: OSPF für IPv6 S.384-385 Kommentar: tw. nicht ganz up-to-date bzw. nicht ganz fehlerfreie Abbildungen Homepage des Buches und Tabelle mit Fixes (http://www.fehcom.de/tipn/tipn.html)
- Internet-Sicherheit (Browser, Firewalls und Verschlüsselung) bei Amazon.de (http://www.amazon.de/exec/obidos/ASIN/3446217258/) Kai Fuhrberg 2. akt. Auflage 2000 Carl Hanser Verlag München, Wien, ISBN 3-446-21333-3 Kap.2.3.1.4. IPv6 S.18-22 Kurz angerissen werden: RFC1825 - Security Association Konzept RFC1826 - IP authentication Header RFC1827 - IP Encapsulation Security Payload
- IPv6. Das neue Internet- Protokoll. Technik, Anwendung, Migration bei Amazon (http://www.amazon.de/exec/obidos/ASIN/389864149X) Hans Peter Dittler 2. akt. und erweiterte Auflage 2002 dpunkt.verlag, ISBN 3-89864-149-X
- Das neue Internetprotokoll IPv6 bei Amazon (http://www.amazon.de/exec/obidos/ASIN/3446216855) Herbert Wiese 2002 Carl Hanser Verlag, ISBN 3446216855

23.1.3. Articoli, ebooks, riviste online (misto)

- Getting Connected with 6to4 (http://www.onlamp.com/pub/a/onlamp/2001/06/01/ipv6_tutorial.html) di Huber Feyrer, 06/01/2001
- How Long the Aversion to IP Version 6 (http://researchcenter.zdnet.com/data/detail?id=1001711171_237&type=RES&x=120054357) - Rivista di META Group Inc., l'accesso completo necessita della registrazione (gratuita) presso META Group Inc.
- Transient Addressing for Related Processes: Improved Firewalling by Using IPv6 and Multiple Addresses per Host; scritto da Peter M. Gleiz, Steven M. Bellovin (PC-PDF-Version (http://www.securiteinfo.com/ebooks/pdf/tarp.pdf); Palm-PDF-Version (http://www.securiteinfo.com/ebooks/palm/tarp.pdf); PDB-Version (http://www.securiteinfo.com/ebooks/pdb/tarp.pdb))

- IPv6, théorie et pratique (http://www.oreilly.fr/catalogue/ipv6-3ed.html) (francese) 3e édition, mars 2002, O'Reilly, ISBN 2-84177-139-3
- IPSec (http://www.securiteinfo.com/crypto/IPSec.shtml) (lingua: francese)
- Internetworking IPv6 with Cisco Routers (http://www.ip6.com/index.html) di Silvano Gai, McGrawHill Italia, 1997. I 13 capitoli e l'appendice A-D sono scaricabili cone documenti PDF.
- Secure and Dynamic Tunnel Broker (http://www.vermicelli.pasta.cs.uit.no/ipv6/students/vegars/) di Vegar Skaerven Wang, Master of Engineering Thesis in Computer Science, 2 Giugno 2000, Facoltà di Scienze, Dip. di Scienza dei Computer, Università di Tromso, Norvegia.
- Aufbruch in die neue Welt - IPv6 in IPv4 Netzen (http://www.netobjectdays.org/pdf/99/stja/doering.pdf) von Dipl.Ing. Ralf Döring, TU Illmenau, 1999
- Migration and Co-existence of IPv4 and IPv6 in Residential Networks (http://www.csc.fi/~psavola/residential.html) di Pekka Savola, CSC/FUNET, 2002

23.1.4. Pubblicazioni di settore (riassunti, bibliografie, risorse online)

- Standardisation Report (http://www.ist-ipv6.org/standards/IPv6STDReptDec02.pdf), Dicembre 2002 pubblicato da IPv6 Cluster (http://www.ist-ipv6.org/), 1mb (ogni 4 mesi il rapporto standard viene aggiornato su: http://www.ist-ipv6.org/standards.html).
- GEANT IPv6 Workplan (http://www.ipv6.ac.uk/gtpv6/workplan.html)
- Mobility support in IPv6 (http://citeseer.nj.nec.com/perkins96mobility.html) di C.E. Perkins e D.B. Johnson. A Proseguimento della Seconda conferenza annuale internazionale sul Mobile (1996).
- A simulation study on the performance of Mobile IPv6 in a WLAN-based cellular network (http://www.ingenta.com/isis/searching/ExpandSearch/ingenta?year_to=2002&year_from=1997&date_type=range&title=IPv6&title_ di Perez Costa X.; Hartenstein H. -- Computer Networks, Settembre 2002, vol. 40, nr. 1, pagg. 191-204(14) -- Elsevier Science.
- IPv6 Trials on UK Academic Networks: Bermuda Project Aug.2002 (http://www.ipv6.ac.uk/bermuda2/): Partecipanti - Ottenere la connessione - Progetti consegnabili - Topologia di rete - Assegnamento indirizzo - Accesso Wireless IPv6 - Migrazione a IPv6 - Presentazione dei progetti - Internet 2 - Altri progetti IPv6 - IPv6 fora e standard Bermuda 2...
- http://www.ipv6.ac.uk/
- A scalable parallel internet router that enables the QoS through merging ATM with IPv6 (http://www.ingenta.com/isis/searching/ExpandSearch/ingenta?year_to=2002&year_from=1997&date_type=range&title=IPv6&title_ Di Song S. -- Computer Communications, 1 Maggio 2002, vol. 25, nr. 7, pagg. 647-651(5) -- Elsevier Science.
- Linux IPv6: Which One to Deploy? (http://liinwww.ira.uka.de/cgi-bin/bibshow?e=Pt0mjovy.kpvsobm/vojrvf) Linux Journal, Vol. 96, pag. 86, 88-90, Aprile 2002.
- An overview and analysis of mobile Internet protocols in cellular environments (http://www.ingenta.com/isis/searching/ExpandSearch/ingenta?year_to=2002&year_from=1997&date_type=range&title=IPv6&title_ Chao H-C. -- Internet Research: Electronic Networking Applications and Policy, 24 Ottobre 2001, vol. 11, no. 5, pagg 435-450(16) -- MCB University Press
- IPv6 for Future Wireless Networks (http://www.ingenta.com/isis/searching/ExpandSearch/ingenta?year_to=2002&year_from=1997&date_type=range&title=IPv6&title_

Nielsen T. -- Wireless Personal Communications, Giugno 2001, vol. 17, nr. 2/3, pagg. 237-247(11) -- Kluwer Academic Publishers, Dordrecht, Olanda

- IPv6 presso l'Università di Southampton (http://www.ipv6.ecs.soton.ac.uk/)
- Seamless Support for Mobile Internet Protocol Based Cellular Environments (http://www.ingenta.com/isis/searching/ExpandSearch/ingenta?year_to=2002&year_from=1997&date_type=range&title=IPv6&title_ Chao H-C.; Chu Y-M. -- International Journal of Wireless Information Networks, Luglio 2001, vol. 8, nr. 3, pagg 133-153(21) -- Kluwer Academic/Plenum Publishers, New York, U.S.A.
- IPv6: The Solution for Future Universal Networks (http://liinwww.ira.uka.de/cgi-bin/bibshow?e=Njtd0MODT0modt3111/vojrvf). Note della Conferenza in Computer Science, Vol. 1818, pagg 82-??, 2000.
- Modeling and performance analysis for IPv6 traffic with multiple QoS classes (http://www.ingenta.com/isis/searching/ExpandSearch/ingenta?year_to=2002&year_from=1997&date_type=range&title=IPv6&title_ Zhang L.; Zheng L. -- Computer Communications, 1 October 2001, vol. 24, no. 15, pp. 1626-1636(11) -- Elsevier Science.
- Threshold-Based Registration (TBR) in Mobile IPv6 (http://liinwww.ira.uka.de/cgi-bin/bibshow?e=Njtd0MODT0modt3111/vojrvf). Note della conferenza in Computer Science, Vol. 1818, pagg 150-??, 2000.
- IPv6 Performance Analysis on FreeBSD Workstation Using Simple Applications (http://liinwww.ira.uka.de/cgi-bin/bibshow?e=Njtd0MODT0modt3111/vojrvf). Note della conferenza in Computer Science, Vol. 1961, pagg 33-??, 2000.
- Microsoft Research IPv6 Implementation (MSRIPv6): MSRIPv6 Configuring 6to4 - Connectivity with MSR IPv6 - Our 6Bone Node... (http://www.research.microsoft.com/msripv6/)
- New frontiers in cybersegmentation: marketing success in cyberspace depends on IP address (http://www.ingenta.com/isis/searching/ExpandSearch/ingenta?year_to=2002&year_from=1997&date_type=range&title=IPv6&title_ Louvieris P.; Driver J. -- Qualitative Market Research: An International Journal, 27 Giugno 2001, vol. 4, nr. 3, pagg 169-181(13) -- MCB University Press.
- QoS-Conditionalized Handoff for Mobile IPv6 (http://liinwww.ira.uka.de/cgi-bin/bibshow?e=Njtd0MODT0modt3113c/vojrvf). Note della conferenza in Computer Science, Vol. 2345, pagg 721-??, 2002.

23.1.5. Altre

Per maggiori informazioni si veda l'URL: Guida IPv6 SWITCH / Riferimenti (http://www.switch.ch/lan/ipv6/references.html)

23.2. Conferenze, Incontri, Summit

23.2.1. 2002

- Renater - Conférence IPv6 2002 (http://www.renater.fr/IPv6-2002/)

- IPv6 Deployment Summit at INET 2002 (http://www.ipv6summit.com/)

Ho lasciato qualcosa? Eventuali suggerimenti sono ben accetti!

23.2.2. 2003

Eventuali suggerimenti sono ben accetti!

23.3. Informazione online

23.3.1. Entrare a far parte del backbone IPv6

Da completare più avanti con maggiori contenuti... i suggerimenti sono ben accetti!

23.3.1.1. Registri Globali

- Backbone di test IPv6: 6bone (http://www.6bone.net/), Come entrare a far parte del 6bone (http://www.6bone.net/6bone_hookup.html), Teilnahme am 6bone (http://www.join.uni-muenster.de/6bone/6bone-teilnahme.html) (lingua tedesca), partecipazione al 6bone (http://www.join.uni-muenster.de/6bone/6bone-participation.html) (lingua inglese)

23.3.1.2. I principali registri regionali

- America: ARIN (http://www.arin.net/), ARIN / pagine di registrazione (http://www.arin.net/registration/ipv6/index.html), ARIN / linee guida IPv6 (http://www.arin.net/library/index.htm)
- EMEA: Ripe NCC (http://www.ripe.net/), Ripe NCC / pagina di registrazione (http://www.ripe.net/ripencc/mem-services/registration/), Ripe NCC / registrazione IPv6 (http://www.ripe.net/ripencc/mem-services/registration/ipv6/ipv6.html)
- Asia/Pacifico: APNIC (http://www.apnic.net/), APNIC / IPv6 guida alle risorse (http://www.apnic.net/services/ipv6_guide.html)
- America Latina e Caraibi: LACNIC (http://lacnic.org/), Servizi di registrazione IPv6 (http://lacnic.net/en/bt-IPv6.html), Politica di allocazione IPv6 (http://lacnic.net/en/chapter-4-en.pdf)
- Africa: AfriNIC (http://www.afrinic.org/)

Inoltre un elenco delle maggiori allocazioni (lunghezza prefisso 32) per ogni registro locale è disponibile qui: Ripe NCC / IPv6 allocations (http://www.ripe.net/ripencc/mem-services/registration/ipv6/ipv6allocs.html).

23.3.1.3. Tunnel broker

Nota: Un elenco dei tunnel broker disponibili può essere trovato nella sezione Tunnel broker più avanti.

- Il codice sorgente (http://www.vermicelli.pasta.cs.uit.no/ipv6/students/vegars/TunnelBroker/) usato nella tesi per il Master di Vermicelli sul tunnel broker, Università di Tromso.
- Ex IPng. risorse, tunnel broker e IPv6, ora migrati al Sistema SixXs (http://www.sixxs.net/main/).
- Pagina IPv6-con-Linux (http://sites.inka.de/lina/linux/ipv6.html) di Eckes.
- RIPE6.net (http://www.ripe6.net/): Configurare un tunnel IPv6 verso un tunnel broker; Instradare IPv6 dal vostro router ad altri client nella vostra rete...
- tunnelc - uno script tunnel client basato su perl: freshmeat.net: Dettagli del progetto per tunnel client (http://freshmeat.net/projects/tunnelc) SourceForge: Informazioni Progetto - tunnelc (http://sourceforge.net/projects/tunnelc) (anche qui (http://tunnelc.sourceforge.net/))
- Mailing list Tunnelbroker (DE) (http://www.ipv6.uni-leipzig.de/~6bone/tb/maillist.html)
- Linux Advanced Routing & Traffic Control HOWTO, Capitolo 6: tunneling IPv6 con Cisco e/o 6bone (http://howtos.linuxbroker.com/howtoreader.shtml?file=Adv-Routing-HOWTO.html#LARTC.TUNNEL-IPV6.ADDRESSING).

Si veda anche qui per maggiori informazioni e URL: ipv6-net.org (http://www.ipv6-net.de/).

23.3.1.4. 6to4

- informazioni 6to4 di NSayer (http://www.kfu.com/~nsayer/6to4/)
- RFC 3068 / An Anycast Prefix for 6to4 Relay Routers (http://www.faqs.org/rfcs/rfc3068.html)

23.3.1.5. ISATAP

- ISATAP (Intra-Site Automatic Tunnel Access Protocol) Information (http://www.join.uni-muenster.de/Dokumente/Howtos/Howto_ISATAP.php?lang=en) by JOIN (http://www.join.uni-muenster.de/)

23.3.2. Ultime notizie

Da completare più avanti... i suggerimenti sono ben accetti!

- hs247 / notizie e informazioni IPv6 (http://hs247.com/)
- bofh.st / ultime notizie IPv6 (http://bofh.st/ipv6/) (attualmente [Gennaio 2002] obsoleto...), anche homepage per il canale #IPv6 su IRCnet
- ipv6-net.org (http://www.ipv6-net.de/), forum tedesco

23.3.3. Riferimenti al protocollo

23.3.3.1. Request For Comment (RFC) relativi a IPv6

La pubblicazione dell'elenco degli RFC relativi a IPv6 va oltre lo scopo di questo documento, le URL elencate vi potranno condurre a qualche lista:

- Elenco ordinato per Stato di standardizzazione di IPng (http://playground.sun.com/pub/ipng/html/specs/standards.html) o Attuali specifiche di IPng (http://playground.sun.com/pub/ipng/html/specs/specifications.html) di Robert Hinden
- Specifiche relative a IPv6 (http://www.ipv6.org/specs.html) su IPv6.org

23.3.3.2. Attuali bozze dei gruppi di lavoro

Attuali (anche) bozze relative a IPv6 si possono trovare qui:

- IP Versione 6 (ipv6) (http://www.ietf.org/ids.by.wg/ipv6.html)
- Next Generation Transition (ngtrans) (http://www.ietf.org/ids.by.wg/ngtrans.html)
- Configurazione Host Dinamica (dhc) (http://www.ietf.org/ids.by.wg/dhc.html)
- Estensioni del Domain Name System (dnsext) (http://www.ietf.org/ids.by.wg/dnsext.html)
- IP Mobile (mobileip) (http://www.ietf.org/ids.by.wg/mobileip.html)
- Ottenere qualsiasi informazione su IPv6, dalle rassegne generali, attraverso bozze &RFC, fino alle implementazioni (http://playground.sun.com/pub/ipng/html/ipng-main.html) (inclusa la disponibilità di stack su varie piattaforme & codice sorgente per stack IPv6)
- Specifiche IPv6 (http://www.sumitomo.com/htmls/randd/ipv6/doc.html) - Gli ultimi RFC e raccolte di Internet Drafts.

23.3.3.3. Altre

- Magie di Rete / IPv6, Internet Protocol versione 6 (http://www.networksorcery.com/enp/protocol/ipv6.htm), intestazioni protocollo IPv6
- Guida/Riferimenti IPv6 SWITCH (http://www.switch.ch/lan/ipv6/references.html), enorme elenco di riferimenti IPv6 mantenuto da Simon Leinen
- Laboratorio di Amministrazione di Rete Avanzato / Oracolo indirizzo IPv6 (http://steinbeck.ucs.indiana.edu:47401/) mostra in dettaglio gli indirizzi IPv6

23.3.4. Maggiori informazioni

Da completare con maggiori contenuti più avanti... i suggerimenti sono ben accetti!

DeepSpace6 / ancora link interessanti (http://www.deepspace6.net/sections/links.html)

23.3.4.1. Relativi a Linux

- DeepSpace6 / (Non solo) Portale Linux IPv6 (http://www.deepspace6.net/) - Italia (Mirror (http://mirrors.bieringer.de/www.deepspace6.net/))
- IPv6-HowTo for Linux di Peter Bieringer (http://www.bieringer.de/linux/IPv6/) - Germania e il suo Bieringer / IPv6 - archivio software (ftp://ftp.bieringer.de/pub/linux/IPv6/)
- Stato di Linux+IPv6 di Peter Bieringer (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status.html) - Germania (sta diventando obsoleto)
- DeepSpace6 / Pagina dello stato di IPv6 (ttp://www.deepspace6.net/docs/ipv6_status_page_apps.html) - Italia (Mirror (http://mirrors.bieringer.de/www.deepspace6.net/docs/ipv6_status_page_apps.html)) (prenderà il posto della precedente)
- progetto USAGI (http://www.linux-ipv6.org/) - Giappone, e il loro progetto USAGI - archivio software (ftp://ftp.linux-ipv6.org/pub/)
- Pagina Linux IPv6 di Gav (http://www.bugfactory.org/~gav/ipv6/)

23.3.4.2. Relative a Linux per distribuzione

PLD

Distribuzione Linux PLD (http://www.pld-linux.org/) ("leader del mercato" per il contenuto di pacchetti abilitati IPv6)

Red Hat

Red Hat Linux (http://www.redhat.com/), Pacchetti IPv6 di Pekka Savola (http://www.netcore.fi/pekkas/linux/ipv6/)

Debian

Debian Linux (http://www.debian.org/), Informazioni e stato di IPv6 di Craig Small (http://people.debian.org/~csmall/ipv6/), Jim’s insignificant LAN IPv6 global connectivity HOWTO (http://www.jipo.org/jim/Jims_LAN_IPv6_global_connectivity_howto.html)

SuSE

SuSE Linux (http://www.suse.com/)

Mandrake

Linux Mandrake (http://www.linux-mandrake.com/)

Per maggiori informazioni si veda Stato delle distribuzioni IPv6+Linux (http://www.bieringer.de/linux/IPv6/status/IPv6+Linux-status-distributions.html).

23.3.4.3. Generale

- IPv6.org (http://www.ipv6.org/)
- 6bone (http://www.6bone.net/)

- Centro risorse IPv6 UK (http://www.cs-ipv6.lancs.ac.uk/) - UK
- JOIN: informazioni IPv6 (http://www.join.uni-muenster.de/JOIN/ipv6/texte-englisch/informationsquellen.html) - Germania, della squadra del progetto JOIN che mantiene anche Collegamenti a pagine WWW esterne comprendenti IPv6/IPng (http://www.join.uni-muenster.de/JOIN/ipv6/texte-englisch/www.html).
- progetto TIPSTER6 (http://tipster6.ik.bme.hu/tipster6_en.html) - Ungheria, "Testare Tecnologia Sperimentale e Servizi IPv6 in Ungheria".
- progetto WIDE (http://www.v6.wide.ad.jp/) - Giappone
- Guida IPv6 SWITCH (http://www.switch.ch/lan/ipv6/) - Svizzera
- Angolo IPv6 di Hubert Feyrer (http://www.feyrer.de/IPv6/) - Germania
- Progetto Vermicelli (http://www.vermicelli.pasta.cs.uit.no/ipv6/) - Norvegia
- Forum IPv6 (http://www.ipv6forum.com/) - un consorzio mondiale di leader nella distribuzione di Internet, Reti di Ricerca &Didattica...
- Playground.sun.com / Pagina Informazioni IPv6 (http://playground.sun.com/pub/ipng/html/ipng-main.html) - mantenuta da Robert Hinden e Nokia. Contiene qualsiasi informazione su IPv6, dalle rassegne generali, attraverso bozze & RFC, fino alle implementazioni (inclusa la disponibilità di stack su varie piattaforme & codice sorgente per stack IPv6).
- Centro di ricerca Ames della NASA (http://www.nas.nasa.gov/Groups/LAN/IPv6/) (obsoleto)
- 6INIT (http://www.6init.com/) - Iniziativa Internet IPv6 - un quinto progetto di struttura di lavoro Europeo sotto il programma IST.
- Task Force IPv6 (Unione Europea) (http://www.ipv6-taskforce.org/)
- Documento Progetto IPv6 (http://www.v6.sfc.wide.ad.jp/v6doc/) (lingua giapponese)
- 6init (http://www.6init.org/) - IPv6 INternet IniTiative
- Visione D'insieme di IP Next Generation (http://www.isoc.org/HMP/PAPER/PT1/html/pt1.html.hinden)
- Home page del gruppo IPng (http://ganges.cs.tcd.ie/4ba2/ipng/).
- IPv6: La nuova versione del protocollo Internet (http://www.usenix.org/publications/library/proceedings/ana97/summaries/deering.html), di Steve Deering.
- IPv6: La Prossima Generazione del Protocollo Internet (http://www.garykessler.net/library/ipv6_exp.html), di Gary C. Kessler.
- IPv6: Protocollo di Prossima Generazione (http://www.3com.com/nsc/ipv6.html) - 3Com
- Letteratura della Ricerca IPv6 (IPng) (http://www.mediaport.org/~iamano/lr.zip), di Mike Crawfurd.
- Iniziativa Internet Prossima Generazione (http://www.ngi.gov/)
- sito internet II (http://www.internet2.org/) e Gruppo di Lavoro internet2 (http://ipv6.internet2.edu/) - Presentazione (HTML + PPT) (http://ipv6.internet2.edu/presentations/) dal laboratorio IPv6: (Autoconfigurazione Stateless, Indirizzamento IPv6, USAGI, Indirizzamento IPv6 indipendente dal provider e altri argomenti).
- NetworkWorldFusion: Ricerca / Trovare Documenti: ricercato per parola chiave IPv6 (http://search.nwfusion.com/query.html?qt=IPv6&qp=&ch=cn&) (102 documenti trovati il 22.12.2002)
- The Register (http://www.theregister.co.uk/) (La ricerca di IPv6 il 22.12.2002, ha dato come risultato 30 documenti)
- ricerca su ZDNet per IPv6 (http://zdnet.search.com/search?cat=279&q=IPv6)

- Ricerca su TechTarget per IPv6 (http://whatis.techtarget.com/wsearchResults/1,290214,sid9,00.html?query=IPv6)
- Elenco risorse per IPv6 &TCP (http://www.faqs.org/faqs/internet/tcp-ip/resource-list/index.html)
- Strumenti IPv6 Klingon (http://ipv6.klingon.nl/), Strumenti IPv6 Klingon (accesso solo per IPv6 nativo) (http://www.ipv6.klingon.nl/): esempi di firewall IPv6, test della larghezza di banda e portscanner

Ho dimenticato qualcosa? I suggerimenti sono ben accetti!

23.3.4.4. Ricerca di Mercato

- La storia di due direzioni di Tecnologia Wireless: Outsourcing dello sviluppo dei Processori e IPv6 (http://www.seminarinformation.com/wconnect/wc.dll?sis~details0~307~TSN)Gruppo Yankee - 4/1/2002 - 12 Pagine - ID: YANL768881
- Atlante Mondiale di Internet: Americhe (http://www.marketresearch.com/product/display.asp?SID=88602378-241489274-186851952&ProductID=803907); IDATE - 2/1/2002 - 242 Pagine - ID: IDT803907. Paesi coperti: America Centrale, Nord America, Sud America; Elenco: Prezzo: $ 3,500.00; stralcio: Panorama dei mercati di accesso a Internet nel mondo. Valutazione di mercato e previsione fino al 2006 per 34 paesi: struttura del mercato: principali ISP e quote di mercato; numero di abbonati, di ISP.
- Crescita del primo interesse per IPv6 (http://www.amazon.com/exec/obidos/tg/detail/-/B000065T8E/copernicshopper/103-1420916-1341420) di IDC (Autore); Prezzo di Listino: $1,500.00; Edizione: e-book (Acrobat Reader); Editore: IDC; ISBN B000065T8E; (1 Marzo 2002)

23.3.4.5. Brevetti

- Database Brevetti Canadesi: Home (http://patents1.ic.gc.ca/intro-e.html), Ricerca (http://patents1.ic.gc.ca/srch_sim-e.html) (Ricerca di Base, semplicemente si metta IPv6 nel campo di ricerca ;-); trovati 84 documenti il 22.12.2002)
- Espacenet (http://www.european-patent-office.org/espacenet/info/index.htm) - Informazioni sui brevetti Europei: Uffici Nazionali, Membri di Espacenet (http://www.european-patent-office.org/espacenet/info/access.htm)(IPv6: 84 documenti, 22.12.2002)
- Ricerca Delphion: Pagina di Ricerca Brevetti (http://www.delphion.com/research/). Registrazione di base (gratuita) richiesta. Esempi trovati il 21.12.2002 cercando IPv6: Metodo di comunicazione tra terminali IPv4 e terminali IPv6 e apparati di conversione IPv4-IPv6 (http://www.delphion.com/details?pn=US06118784__) Traduttore per reti IP, sistemi di rete che usano il traduttore e metodo di accoppiamento di reti IP (http://www.delphion.com/details?pn=US06038233__)

23.3.5. Per paesi

23.3.5.1. Europe

- www.ist-ipv6.org (http://www.ist-ipv6.org/): Cluster IPv6 IST, Progetti di Ricerca e Sviluppo IPv6 Europei

- Euro6IX (http://www.euro6ix.org/): Backbone Internet Exchange IPv6 Europeo

23.3.5.2. Austria

- IPv6@IKNnet e Gruppo di Ricerca MIPv6 (http://www.ikn.tuwien.ac.at/~ipv6/): TU Vienna, Austria (IPv6: progetto, pubblicazioni, diploma / tesi di dottorato, Andamento delle Conferenze ecc.)

23.3.5.3. Australia

- Pagine Australiane IPv6 di Carl (http://oversteer.bl.echidna.id.au/IPv6/) (contenuto vecchio)

23.3.5.4. Belgio

- BELNET (http://vivaldi.belnet.be/ipv6/): La Rete di Ricerca Belga
- Euronet (http://www.ipv6.euronet.be/): uno dei più grandi ISP del Belgio...

23.3.5.5. Brasile

- BR6bone (http://www.6bone.rnp.br/)
- Conferenza IPv6 in Brasile (http://www.ipv6summit.com.br/)
- IPv6 do Brasil (http://www.ipv6dobrasil.com.br/)

23.3.5.6. Cina

- Nokia Cina - Sito IPv6 (http://www.ipv6.com.cn/)
- Iniziativa Internet6 di Nokia Cina (http://www.internet6.com.cn/)

23.3.5.7. Cecoslovacchia

- IPv6 su Cecoslovacchia On Line (http://bsd-ipv6.vol.cz/)

23.3.5.8. Germania

- IPv6-net.org (http://www.ipv6-net.de/): forum IPv6 Tedesco

23.3.5.9. Francia

- Renater (http://www.renater.fr/Projets/IPv6/index.htm): Pagina Progetto IPv6 Renater
- IPv6 - RSVP - ATM presso INRIA (http://www.inria.fr/recherche/equipes/ipv6.fr.html)
- Documentazione IPv6 NetBSD (http://www.netbsd.org/fr/Documentation/network/ipv6/)

23.3.5.10. Ungheria

- Testare Tecnologia Sperimentale e Servizi IPv6 in Ungheria (http://tipster6.ik.bme.hu/tipster6_en.html)
- Pagina Ungherese di informazione su IPv6 (http://www.ipv6.fsz.bme.hu/)

23.3.5.11. India

- IPv6 presso il BITS (http://ipv6.bits-pilani.ac.in/)

23.3.5.12. Italia

- Project6 (http://project6.ferrara.linux.it/): Networking IPv6 con Linux
- Edisontel (http://www.6bone.it/): Portale IPv6 di Edisontel

23.3.5.13. Giappone

- Gruppo utenti Linux IPv6 JP (http://www.v6.linux.or.jp/)
- IPv6 Yamaha (http://www.rtpro.yamaha.co.jp/RT/ipv6/) (spiacente, tutto in giapponese...)

23.3.5.14. Corea

- ETRI (http://www.krv6.net/): Instituto di Ricerca di Elettronica e Telecomunicazioni
- Forum IPv6 Corea (http://www.ipv6.or.kr/english/index.new.htm): Progetto di Sviluppo IPv6 Coreano

23.3.5.15. Messico

- IPv6 Messico (http://www.ipv6.unam.mx/) (versione spagnola & inglese): Homepage del progetto IPv6 dell'Università Nazionale Autonoma del Messico (UNAM)

23.3.5.16. Olanda

- SURFnet (http://www.ipv6.surfnet.nl/): Backbone IPv6 SURFnet
- STACK (http://www.stack.nl/), STACK (IPv6) (http://www.stack.nl/ipv6/): Associazione degli studenti di computer dell'Università della Tecnologia di Eindhoven, Olanda
- IPng.nl (http://www.ipng.nl/): collaborazione tra WiseGuys e Intouch

23.3.5.17. Portogallo

- Pagine IPv6 di Miguel Rosa (http://ipng.ip6.fc.ul.pt/) (non funzionanti?)
- FCCN (Fondazione Nazionale per la Computazione Scientifica) (http://www.fccn.pt/projectos/ipv6/index_html)
- Università dell'Algarve, Portogallo (http://www.ipv6.ualg.pt/)
- IPv6 - MFA (http://www.ipv6.mfa.eti.br/)

23.3.5.18. Russia

- Forum IPv6 per la Russia (http://www.ipv6.ru/): Centro Internet dell'Università Statale Yaroslavl

23.3.5.19. Svizzera

- SWITCH (http://www.switch.ch/network/ipv6/references.html): La Rete Svizzera di Didattica &Ricerca

23.3.5.20. Regno Unito

- IPv6 nel R.U. (http://www.ipv6.org.uk/)
- Centro risorse IPv6 del R.U. (http://www.cs-ipv6.lancs.ac.uk/)
- Home IPv6 di British Telecom (http://www.bt.com/ipv6/): Esperimento IPv6 dell'ISP BT, primo Internet Exchange IPv6 del R.U. ecc.

23.3.6. Per sistemi operativi

23.3.6.1. *BSD

- progetto KAME (http://www.kame.net/) (*BSD)
- FAQ sul networking IPv6 di NetBSD (http://www.netbsd.org/Documentation/network/ipv6/)
- Progetto Documentazione FreeBSD (http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/ipv6.html)
- Port di FreeBSD: Ipv6 (http://www.freebsd.org/ports/ipv6.html)

- BUGAT - Gruppo utenti BSD Austria - www.bugat.at (http://www.bugat.at/): Tunnel IPv6 FreeBSD (http://www.bugat.at/inforum/contentview.php/mini-howto/freebsd-ipv6.ihtml) (lingua Tedesca)

23.3.6.2. Cisco IOS

- Pagina di ingresso Cisco IOS IPv6 (http://www.cisco.com/warp/public/732/Tech/ipv6/)
- IPv6 per il Software Cisco IOS (http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/ipv6/ftipv6c.htm), File 2 di 3: Agosto 2002 -- Tavola dei Contenuti: IPv6 per il Software Cisco IOS; Documentazione Specifiche di Configurazione; Abilitare l'instradamento IPv6 e Configurazione; Indirizzamento IPv6; Abilitare Globalmente il Trattamento di IPv6.
- Prontuario Cisco sull'Internet Networking, Capitolo IPv6 (http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ipv6.htm)

23.3.6.3. Compaq

- IPv6 presso Compaq (http://www.compaq.com/ipv6/) - Presentazioni, Pagine Bianche, Documentazione...

23.3.6.4. HPUX

- FAQ di comp.sys.hp.hpux (http://www.faqs.org/faqs/hp/hpux-faq/index.html)

23.3.6.5. IBM

- Ora che IBM ha annunciato la disponibilità dello z/OS V1.4, che c'è di nuovo in questa release? (http://search390.techtarget.com/ateQuestionNResponse/0,289625,sid10_cid486367_tax292523,00.html) Questa domanda fu posta il 15 Agosto 2002

23.3.6.6. Microsoft

- Microsoft Windows 2000 IPv6 (http://www.microsoft.com/windows2000/technologies/communications/ipv6/default.asp)
- MSRIPv6 (http://www.research.microsoft.com/msripv6) - Rete di Ricerca Microsoft - Homepage IPv6
- Iniziare con l'anteprima della Tecnologia IPv6 per Windows 2000 (http://msdn.microsoft.com/downloads/sdks/platform/tpipv6/start.asp)
- Internet Connection Firewall Non Blocca il Traffico del Protocollo Internet Versione 6 (http://support.microsoft.com/default.aspx?scid=kb;en-us;306203) (6.11.2001)
- Numeri di Protocollo Internet (http://support.microsoft.com/default.aspx?scid=kb;en-us;289892) (8.10.2002)
- Aggiornamento sull'anteprima della Tecnologia IPv6 (http://support.microsoft.com/default.aspx?scid=kb;en-us;273826) (16.10.2002)

- HOW TO: Installare e configurare IP Versione 6 su Windows .NET Enterprise Server (http://support.microsoft.com/default.aspx?scid=kb;en-us;325449) (26.10.2002)
- Il Servizio Router 6to4 di Windows .NET Server Termina quando si avvisa un indirizzo 2002 sull'Interfaccia Pubblica (http://support.microsoft.com/default.aspx?scid=kb;en-us;329984) (28.10.2002)
- msdn - Microsoft Windows CE .NET - comandi IPv6 (http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wcetcpip/htm/cmconIPv6exe.asp)
- msdn - ricerca per IPv6 (http://search.microsoft.com/default.asp?qu=IPv6&boolean=ALL&nq=NEW&so=RECCNT&p=1&ig=01&i=00&i=01&i=02&i=03 (100 risultati, 22.12.2002)

23.3.6.7. Solaris

- Pagina IPv6 Sun Microsystems per Solaris 8 (http://www.sun.com/solaris/ipv6/)
- Solaris 2 Frequently Asked Questions (FAQ) 1.73 (http://www.cs.uu.nl/wais/html/na-dir/Solaris2/FAQ.html)

23.3.6.8. Sumitoma

- Sumitomo Electric ha implementato IPv6 sui router della famiglia Suminet 3700 (http://playground.sun.com/pub/ipng/html/ipng-implementations.html#Sumitomo)

23.3.6.9. ZebOS

- ZebOS Server Routing Software (http://www.ipinfusion.com/products/server/products_server.html) di IpInfusion

23.3.7. Sicurezza IPv6

- Internet Security Systems: Centro per la Sicurezza, Ricerca Database X-Force (http://www.iss.net/security_center/search.php?type=3&type=3&pattern=IPv6) (21.12.2002 - 6 argomenti trovati relativi a IPv6)
- Progetto IPsec NIST (http://csrc.nist.gov/ipsec/) (Istituto Nazionale degli Standard e delle Tecnologie, NIST)
- Information Security (http://www.infosecuritymag.com/index.shtml), Ricerca per IPv6 (http://search.atomz.com/search/?sp-q=IPv6&sp-a=sp1000c0bf) (21.12.2002 - 9 articoli trovati)
- NewOrder.box.sk (ricerca per IPv6) (http://neworder.box.sk/search.php3?srch=IPv6) (Articoli, exploit, file database ecc.)

23.3.8. Elenchi di applicazioni

- IPv6.org / Applicazioni abilitate IPv6 (http://www.ipv6.org/v6-apps.html)

- Freshmeat / ricerca per IPv6 (http://freshmeat.net/search/?q=IPv6), attualmente (14 Dicembre 2002) 62 progetti
- Forum IPv6: Elenco Router IPv6 (http://www.ipv6forum.com/navbar/links/v6routerlist.htm)

23.3.8.1. Strumenti di analisi

- Ethereal (http://ethereal.planetmirror.com/) - Ethereal è un analizzatore di protocollo libero per Unix e Windows
- Radcom RC100-WL (http://www.ip6.com/us/analyzer.htm) - Download dell'analizzatore di protocollo Radcom RC100-WL versione 3.20

23.3.8.2. Prodotti IPv6

- 6wind (http://www.6wind.com/) - soluzioni per Router IPv4/IPv6, QoS, Multicast, Mobilità, Sicurezza/VPN/Firewall.
- patch di Fefe per IPv6 con djbdns (http://www.fefe.de/dns/) Agosto 2002 -- Cos'è djbdns e perché necessita di IPv6? djbdns è un server DNS completamente maturo che si avvicina alle performance di BIND.
- Home page IPv6 di Hitachi Ltd. (http://www.v6.hitachi.co.jp/) Prodotti IPv6 GR2000 - router IPv6; Toolnet6 - Protocol Exchange software per IPv6.
- Suite ZebOS Server Routing (http://www.ipinfusion.com/products/server/products_server.html)
- SPA Mail Server 2.21 (http://download.com.com/3000-2165-10153543.html?tag=lst-0-21)
- Inframail (Advantage Server Edition) 4.0 (http://download.com.com/3000-2165-8202652.html?tag=lst-0-2)
- HTTrack Website Copier 3.2 (http://download.com.com/3000-2377-10149393.html?tag=lst-0-1)
- CommView 3.4 (http://download.com.com/3000-2085-10132748.html?tag=lst-0-1)
- Posadis 0.50.6 (http://download.com.com/3000-2104-10149750.html?tag=lst-0-1)
- Xceed Winsock Library 1.2 (http://download.com.com/3000-2070-10132771.html?tag=lst-0-1)
- CommView 3.4 (http://download.com.com/300-2085-10132748.html?tag=lst-0-1)
- TCP Wrapper (pronto per IPv6) (ftp://ftp.porcupine.org/pub/ipv6/)

23.3.8.3. SNMP

- FAQ su SNMP di comp.protocpols.snmp Parte 1 di 2 (http://www.cs.uu.nl/wais/html/na-dir/snmp-faq/part1.html)

23.4. Infrastruttura IPv6

23.4.1. Statistiche

- Storico della tabella di routing IPv6 (http://www.space.net/~gert/RIPE/) creato da Gert Döring, Space.Net (http://www.space.net/)
- Statistiche dell'Utilizzo di www6.vermicelli.pasta.cs.uit.no (http://ftp2.no.netbsd.org/statistics/webalizer/www6.vermicelli.pasta.cs.uit.no/) (ultimi 12 mesi)
- Elenco Ufficiale delle statistiche del Webserver 6bone (http://6bone.informatik.uni-leipzig.de/ipv6/stats/stats.php3)
- Dati di Allocazione di IPv6 &Vista dei Risultati (http://www.ripe.net/ripe/meetings/archive/ripe-42/presentations/ripe42-ipv6-survey/sld001.html), IPv6 WG, Ripe 42, Ripe NCC

23.4.2. Internet Exchange

Un'altra lista degli Internet Exchange IPv6 si può trovare qui: Sito Web Exchange IPv6 (http://www.v6nap.net/) o Stato IPv6 degli IXP in Europa (http://www.euro-ix.net/isp/choosing/search/matrix.php)

23.4.2.1. Estonia

- TIX (http://tix.estpak.ee/) (tallinn interneti exchange con supporto ipv6)

23.4.2.2. Europa

- Euro6IX (http://www.euro6ix.net/), Backbone Internet Exchange IPv6 Europeo

23.4.2.3. Francia

- Internet Exchange IPv6 Nazionale Francese (http://www.fnix6.net/) (attivo dal 1.11.2002). FNIX6 fornisce un'interconnessione FastEthernet ad alta velocità affidabile e gratuita tra ISP che si trovano su TeleCity Paris.

23.4.2.4. Germania

- INXS (http://www.inxs.de/): (Cavo & Wireless) Monaco e Amburgo

23.4.2.5. Giappone

- NSPIXP-6 (http://www.wide.ad.jp/nspixp6/): Internet Exchange basato su IPv6 a Tokyo
- JPIX (http://www.jpix.co.jp/), Tokyo

23.4.2.6. Corea

- 6NGIX (http://www.ngix.ne.kr/)

23.4.2.7. Olanda

- AMS-IX (http://www.ams-ix.net/home.html): Amsterdam Internet Exchange

23.4.2.8. Regno Unito

- UK6X (http://www.uk6x.com/): Londra
- XchangePoint (http://www.xchangepoint.net/): Londra

23.4.2.9. USA

- 6TAP (http://www.6tap.net): Chicago. Supporta il peering in tutto il mondo.
- NY6IX (http://www.ny6ix.net/): Internet Exchange basato su IPv6 nella città di New York
- 6IIX (http://www.6iix.net/): New York, Los Angeles e Santa Clara
- PAIX (http://www.paix.net/): Palo Alto

23.4.3. Tunnel broker

Si veda anche: http://www.deepspace6.net/docs/tunnelbrokers.html

23.4.3.1. Belgio

- Wanadoo (http://tunnel.be.wanadoo.com/)

23.4.3.2. Canada

- Freenet6 (http://www.freenet6.net/) - Delega /48, Canada Ottenere IPv6 Usando Freenet6 su Debian (http://www.linuxjournal.com/article.php?sid=5963&mode=thread&order=0) creatore Freenet6 (http://www.viagenie.qc.ca/en/index.shtml)

23.4.3.3. Cina

- CERNET-Nokia (http://tb.6test.edu.cn/)

23.4.3.4. Estonia

- Estpak (http://tunnelbroker.ipv6.estpak.ee/?tunnel&PHPSESSID=aa2184190cc2cc6d3a6f6ddd01ae3635)

23.4.3.5. Europa

- XS26 Tunnel Broker Distribuito (http://www.xs26.net/), USA & Europa

23.4.3.6. Germania

- JOIN (http://www.join.uni-muenster.de/welcome-e.html), Università di Münster
- Tunnelbroker di Leipzig (http://joshua.informatik.uni-leipzig.de/), Germania - Utenti Dialup con indirizzi IP dinamici, possono ottenere un indirizzo IPv6 statico... JOIN (http://tunnel.ipv6-net.de/), Tunnelbroker di Leipzig, (attualmente offline ?) Das offizielle IPv6 Forum (http://tunnel.ipv6-net.de/), Leipziger Tunnelbroker Flusso dal vivo IPv6 sperimentale! (http://aopteryx.informatik.uni-leipzig.de:8000/live.mp3) 6bone Knoten Leipzig (http://6bone.informatik.uni-leipzig.de/) Info bez. Hackangriff (2001) (http://www.mail-archive.com/ipv6@uni-muenster.de/msg00056.html) Tunnelbroker per JOIN (http://bbackbone.informatik.uni-leipzig.de/). Questo tunnel broker è a disposizione solo per gli studenti ed i membri dello staff di collegi ed università
- Berkom (http://fix.ipv6.berkom.de/cgi-bin/tb.pl)

23.4.3.7. Italia

- Centro Studi e Laboratorio di Telecomunicazioni (https://carmen.cselt.it/ipv6tb/) (Pagina di Download: TunnelBroker Versione 2.1. (http://carmen.cselt.it/cgi-bin/download.pl?pkg=TunnelBroker)) IPv6 Tunnel Broker: Istruzioni di installazione (http://carmen.cselt.it/ipv6/tools/ipv6tb/Installing-ipv6tb.html)
- EdisonTel (http://www.6bone.it/)
- Comv6 (http://www.comv6.com/)
- Bersafe (http://www.bersafe.it) (lingua italiana)
- LAB Telecom Italia (http://carmen.ipv6.tilab.com/) (Pagina di Download del Software Tunnelbroker (http://carmen.ipv6.tilab.com/cgi-bin/download.pl?pkg=TunnelBroker))

23.4.3.8. Giappone

- Iniziativa Internet Giappone (http://www.iij.ad.jp/en/IPv6/) (Lingua Giapponese (http://www.iij.ad.jp/IPv6/)) - con servizio di linea nativa IPv6 e servizio di tunneling IPv6

23.4.3.9. Malesia

- Manis (http://tbroker.manis.net.my/)

23.4.3.10. Olanda

- XS26 - "Accesso a Six" (http://www.xs26.net/) - con POP nella Repubblica Slovacca, Repubblica Ceca, Olanda, Germania e Ungheria.
- IPng Olanda (http://www.ipng.nl/) - Intouch, SurfNet, AMS-IX, UUNet, Cistron, RIPE NCC e AT&T sono connessi presso lo AMS-IX. È possibile ottenere (previo possesso di alcuni requisiti) un tunnel statico.
- Utenti SURFnet (http://www.ipv6.surfnet.nl/)

23.4.3.11. Norvegia

- UNINETT (http://www.uninett.no/testnett/index.en.html), Guida al Servizio IPv6 (per gli Utenti): tunnelbroker & allocazione indirizzo Tunnel Broker di UNINETT (http://tunnelbroker.uninett.no) (link corretto, ma non disponibile - 23.12.2002) Uninett-Autoupdate-HOWTO (http://www.guruz.de/Uninett-Autoupdate-HOWTO)

23.4.3.12. Spagna

- Consulintel (http://tb.consulintel.euro6ix.org/)

23.4.3.13. Svizzera

- Tunnelbroker AS8758 (http://tunnelbroker.as8758.net/), Dolphins Network Systems (attivo dal 20.12.2002)

23.4.3.14. Regno Unito

- NTT Europa (http://www.uk.v6.ntt.net/), NTT (http://www.nttv6.net/), Regno Unito - Prova di IPv6. Tunnel IPv4 e connessioni leased line native IPv6. I POP si trovano a Londra UK, Dusseldorf Germania, New Jersey USA (Costa Est), Cupertino USA (Costa Ovest), Tokyo Giappone
- Servizio Tunnel Broker IPv6 BtexacT (https://tb.ipv6.btexact.com/)
- IPNG-UK (http://ipng.org.uk/)

23.4.3.15. USA

- ESnet (http://www.es.net/hypertext/welcome/pr/ipv6.html), USA - Rete delle Scienze dell Energia: Registro Tunnel & Delegazione Indirizzo per siti direttamente connessi a ESnet e collaboratori ESnet.
- 6REN (http://www.6ren.net/), USA - L'iniziativa 6ren viene coordinata dalla Rete delle Scienze dell'Energia (ESnet), la rete per il programma di ricerca sull'energia del dipartimento americano dell'energia, presso il Laboratorio Nazionale Lawrence Berkeley dell'Università della California.
- Tunnel Broker Distribuito XS26 (http://www.xs26.net/), USA & Europa
- Hurricane Electric (http://ipv6tb.he.net/), US backbone; Hurrican Electric Tunnelbroker (http://tunnelbroker.net/) (anche disponibile su http://tunnelbroker.com/) Comunicato Stampa: Hurricane Electric Upgrades IPv6 Tunnel

Broker (http://www.he.net/releases/release6.html) Tunnel Broker Endpoint Autoupdate (http://ipv6.he.net/tunnelbroker-update.php), Script Perl

- Mondo Services (http://www6.mondoservices.net/)
- DHIS Tunnel Dinamico (http://www.dhis.org/atncp/), Sistema di Informazione Host Dinamico, Università di Bradford

23.4.3.16. Singapore

- http://tunnel-broker.singnet.com.sg/, con NAT e opzione IPsec

23.4.3.17. Altri tunnel broker...

- Relay Router 6to4 Pubblici (http://www.kfu.com/~nsayer/6to4/) (boicottaggio di MS IIE!)

23.4.4. Servizi nativi IPv6

Nota: Questi servizi sono disponibili solo con una connessione IPv6 valida!

23.4.4.1. Server per giochi

- Quake2 (http://www.viagenie.qc.ca/en/ipv6/quake2/ipv6-quake2.shtml) su IPv6

23.4.4.2. Server IRC

- Cyconet (http://ipv6.cyconet.org/?id=server) (IRCnet Server Cyconet su IPv6)

23.4.4.3. Stazioni radio, flussi musicali

- Flusso sperimentale IPv6 dal vivo! (http://aopteryx.informatik.uni-leipzig.de:8000/live.mp3), Università di Leipzig, Germania

23.4.4.4. Server Web

- Casa del Linux IPv6 HOWTO di Peter Bieringer (http://www.ipv6.bieringer.de/)

Ho dimenticato qualcosa? I suggerimenti sono ben accetti!

23.5. Mailing list

Elenchi di mailing list sono disponibili presso:

- Progetto JOIN / Elenco di mailing list relative a IPv6 (http://www.join.uni-muenster.de/JOIN/ipv6/texte-englisch/ipv6.infoquellen.html)
- DeepSpace6 / Mailing List (http://www.deepspace6.net/sections/lists.html)

Le mailing list più importanti sono elencate nella seguente tabella:

.

Focus	Indirizzo e-mail per richiesta	Cosa sottoscrivere	Indirizzo e-mail lista	Lingua	Accesso tramite WWW
Networking del kernel Linux incluso IPv6	majordomo (at) oss.sgi.com	netdev	netdev (at) oss.sgi.com	Inglese	Archivio (http://oss.sgi.com/projects/netdev/ar
Linux e IPv6 in generale (1)	majordomo (at) list.f00f.org	linux-ipv6	linux-ipv6 (at) list.f00f.org (moderata)	Inglese	
Implementazione Linux del protocollo IPv6	Su Web, si veda l'URL		project6 (at) ferrara.linux.it	Inglese	Info (http://project6.ferrara.linux.it/sectio Sottoscrizione (http://mailman.ferrara.linux.it/listinf
IP(v6) mobile per Linux	majordomo (at) list.mipl.mediapoli.com	mipl	mipl (at) list.mipl.mediapoli.com	Inglese	Info (http://www.mipl.mediapoli.com/mai Archivio (http://www.mipl.mediapoli.com/mai archive/)
utenti Linux IPv6 con estensioni USAGI	usagi-users-ctl (at) linux-ipv6.org		usagi-users (at) linux-ipv6.org	Inglese	Info / Ricerca (http://www.linux-ipv6.org/ml/index.html#usagi-users), Archivio (http://www.linux-ipv6.org/ml/usagi-users/)
IPv6 su Debian Linux	su Web, si veda l'URL		debian-ipv6 (at) lists.debian.org	Inglese	Info/Sottoscrizione/Archivio (http://lists.debian.org/debian-ipv6/)

IPv6/6bone in Germania	majordomo (at) atlan.uni-muenster.de	ipv6	ipv6 (at) uni-muenster.de	Tedesco/Inglese	Info (http://www.join.uni-muenster.de/JOIN/ipv6/texte-englisch/mailingliste.html), Archivio (http://www.join.uni-muenster.de/local/majordomo/ipv6/)
6bone	majordomo (at) isi.edu	6bone	6bone (at) isi.edu	Inglese	Info (http://www.6bone.net/6bone_email.h Archivio (http://mailman.isi.edu/pipermail/6bo
Discussioni IPv6	majordomo (at) sun-roof.eng.sun.com	ipng	ipng (at) sun-roof.eng.sun.com	Inglese	Info (http://playground.sun.com/pub/ipng/ Archivio (ftp://playground.sun.com/pub/ipng/m archive/), Mirror dell’archivio (http://www.wcug.wwu.edu/lists/ipng
Utenti IPv6 in generale	majordomo (at) ipv6.org	users	users (at) ipv6.org	Inglese	Info (http://www.ipv6.org/mailing-lists.html), Archivio (http://news.gmane.org/thread.php?gr
Bugtracking delle applicazioni Internet (2)	bugtraq-subscribe (at) securityfocus.com		bugtraq (at) securityfocus.com (moderata)	Inglese	Info (http://online.securityfocus.com/popu Archivio (http://online.securityfocus.com/archi
IPv6 in generale	su Web, si veda l’URL		ipv6 (at) ipng.nl	Inglese	Info/Sottoscrizione (http://mailman.ipng.nl/mailman/listi Archivio (http://mailman.ipng.nl/pipermail/ipv
majordomo (at) mfa.eti.br	majordomo (at) mfa.eti.br	ipv6	ipv6 (at) mfa.eti.br	Portoghese	Info (http://www.marcelo.pro.br/mailman/

(1) raccomandata per argomenti comuni di Linux & IPv6.

(2) molto raccomandata se si forniscono applicazioni server.

Ho dimenticato qualcosa? I suggerimenti sono ben accetti!

Inoltre le seguenti mailing list & newsgroups sono disponibili via web:

- ipv6 (France) (http://www.tile.net/lists/showlists.php?list_id=35905) Descrizione: ipv6 Cette liste existe pour discuter en français de IP version 6. Elle s'addresse aux personnes desirant demarer des aujourd'hui des tests IPv6. Ce n'est en aucun cas un substitut des listes de l'IETF. Pour de plus amples informations: http://www.urec.fr/IPng
- Tunnelbroker Maillingliste (Germania) (http://www.ipv6.uni-leipzig.de/~6bone/tb/maillist.html)
- ipv6 (Ungheria) (http://www.tile.net/lists/showlists.php?list_id=36611) Descrizione: ipv6 Az IPv6 protokoll listaja Konfiguracios es adminisztracios kerdesek az IPv6-al kapcsolatban. (Archivum) (http://www.ipv6.fsz.bme.hu/mlists/ipv6/ipv6.html)
- student-ipv6 (India) (http://groups.yahoo.com/group/student-ipv6) Descrizione: Questo è il gruppo relativo allo Student Awareness dell'IPv6 in India
- IPV6-CNR@LISTSERV.CNR.IT (Italia) (http://www.lsoft.com/scripts/wl.exe?SL1=IPV6-CNR&H=LISTSERV.CNR.IT) Descrizione: Gruppo di interesse IPv6 del CNR
- ipv6-jp (Giappone) (http://www.tile.net/lists/showlists.php?list_id=14761)
- ipv6 (Giappone) (http://www.tile.net/lists/showlists.php?list_id=37305)
- IPV6@LISTS.UTWENTE.NL (Olanda) (http://www.lsoft.com/scripts/wl.exe?SL1=IPV6&H=LISTS.UTWENTE.NL) Descrizione: IPv6 overleg
- IPV6@NIC.SURFNET.NL (Olanda) (http://www.lsoft.com/scripts/wl.exe?SL1=IPV6&H=NIC.SURFNET.NL) Descrizione: AMS-IX list for IPv6 related matters
- sun-ipv6-users (http://groups.yahoo.com/group/sun-ipv6-users) Descrizione: Si prega di riportare prolemi/suggerimenti riguardanti l'implementazione SUN Microsystems di IPng
- IPv6-BITS (http://groups.yahoo.com/group/IPv6-BITS) Descrizione: Questo gruppo coordinerà il lavoro del progettto Vertebrae.
- openbsd-ipv6 (http://www.tile.net/lists/showlists.php?list_id=23220)
- IPv6 (http://www.tile.net/lists/showlists.php?list_id=35203) Descrizione: Questa mailing list tratta discussioni tecniche sulla possibiltà dell'IPv6/IPsec WRT di OpenBSD.
- linux-bangalore-ipv6 (http://groups.yahoo.com/group/linux-bangalore-ipv6) Descrizione: La lista del Bangalore Linux User Group
- gab (http://www.tile.net/lists/showlists.php?list_id=32034) Descrizione: L'intento è quello di discutere sui piani relativi all'indirizzamento di IPv6.
- ipv6-bsd-user (http://www.tile.net/lists/showlists.php?list_id=37430) Descrizione: Questa mailing list è relativa all'implementazione IPv6 di INRIA/IMAG. Cette liste de discussion est au sujet de l'implementation INRIA/IMAG IPv6. Elle est bilingue Francais/Anglais. La mailing list è bilingua, francese e inglese. Se si desidera contattare gli implementatori, si provi ipv6-bsd-core@imag.fr. Si vous voulez contacter les implementeurs, essayez ipv6-bsd-core@imag.fr.
- gated-ipv6 (http://www.tile.net/lists/showlists.php?list_id=41375)
- packet-switching (http://groups.yahoo.com/group/packet-switching) Descrizione: Questa mailing list fornisce un forum per la discussione della teoria dello scambio dei pacchetti (packet switching), la tecnologia, l'implementazione e l'applicazione in ogni aspetto di rilievo, incluso, senza limitazioni, LAPB, X.25, SDLC, P802.1d, LLC, IP, IPv6, IPX, DECNET, APPLETALK, FR, PPP, IP Telephony, sistemi LAN PBX, protocolli di

gestione come SNMP, e-mail, sistemi a finestra di rete trasparente (network transparent window), implementazione e verifica dei protocolli, test di conformità e strumenti per la manutenzione o lo sviluppo dei sistemi di scambio dei pacchetti.

- mumbaiinternetgroup (http://groups.yahoo.com/group/mumbaiinternetgroup) Descrizione: Questo forum discute gli argomenti e gli sviluppi relativi ad Internet nella zona Asia/Pacifico. Vi si possono trovare discussioni su IPv4, IPv6, DNS, Numeri di Autonomous System, Internet Governence e molto di più...
- de.comm.protocols.tcp-ip Descrizione: Umstellung auf IPv6 Sorgente: Chartas der Newsgruppen in de.* (http://www.faqs.org/faqs/de-newsgroups/chartas/index.html)
- Hurricane Electric IPv6 Forum (http://ipv6.he.net/forum/)
- Gruppo Google: comp.protocols.tcp-ip (http://groups.google.com/groups?hl=en&lr=&ie=UTF-8&oe=UTF8&safe=off&group=comp.protocols.tcp-ip)
- Gruppo Google: linux.debian.maint.ipv6 (http://groups.google.com/groups?hl=en&lr=&ie=UTF-8&oe=UTF8&safe=off&group=linux.debian.maint.ipv6)
- Gruppo Google: microsoft.public.platformsdk.networking.ipv6 (http://groups.google.com/groups?hl=en&lr=&ie=UTF-8&oe=UTF8&safe=off&group=microsoft.public.platformsdk.networking.ipv6)
- Gruppo Google: fa.openbsd.ipv6 (http://groups.google.com/groups?hl=en&lr=&ie=UTF-8&oe=UTF8&safe=off&group=fa.openbsd.ipv6)

23.6. Strumenti on line

23.6.1. Testing tools

- finger, nslookup, ping, traceroute, whois: Centro Risorse UK IPv6 / Pagina di test (http://www.cs-ipv6.lancs.ac.uk/ipv6/testing/)
- ping, traceroute, tracepath, registro 6bone, DNS: JOIN / Strumenti di test (http://www.join.uni-muenster.de/lab/testtools.html) (solo lingua tedesca, ma non dovrebbero esserci problemi per chi non la capisce)
- traceroute6, whois: IPng.nl (http://www.ipng.nl/)
- AAAA Lookup Checker http://www.cnri.dit.ie/cgi-bin/check_aaaa.pl

23.6.2. Recupero di informazioni

- Registro 6BONE (http://www.kessens.com/~david/6bone/)
- Elenco di tutti i blocchi IP IPv6-aggregated mondiali (http://www.dfn.de/service/ipv6/ipv6aggis.html) (aggiornamento giornaliero) Lo stesso servizio dal RIPE (http://www.ripe.net/ripencc/mem-services/registration/ipv6/ipv6allocs.html)

23.6.3. Looking Glass IPv6

- IMAG (http://lookingglass.imag.fr/), Francia
- Looking Glass IPv6 presso SURRIEL (http://linux.uninet.edu/lg/)
- APAN (http://www.jp.apan.net/cgi-bin/ipv6/mrlg), Giappone (APAN-JP)
- Looking Glass DRENv6 (http://www.v6.dren.net/lg/)

23.6.4. Applicazioni di aiuto

- Calcolatrice di prefissi IPv6 (http://www.tdoi.org/prefcalc.php) di TDOI (http://www.tdoi.org/)
- verifica record DNS (http://www.maths.tcd.ie/cgi-bin/check_dns.pl)

23.7. Addestramento, Seminari

- Addestramento e Laboratorio IPv6 (http://www.aerasec.de/workshops/ipv6.html), AERAsec, Germania (solo in lingua Tedesca per ora)
- Migrazione a IPv6 (http://www.seminarinformation.com/wconnect/wc.dll?sis~details0~194045), Learning Tree International
- CIW Internetworking Professional Training CBT CD (http://www.e-trainonline.com/html/ciw_internetworking_profession.html#IPv6)
- Pagine di Addestramento (http://www.trainingpages.net/x/category.html?kw=125), R.U. - Ricerca per IPv6 (13 Corsi, 22.12.2002)

Ho dimenticato qualcosa? I suggerimenti sono ben accetti!

23.8. 'La Scoperta On line' ...

IPv6: Addressing The Needs Of the Future (http://www.amazon.com/exec/obidos/tg/detail/-/B00006334Y/copernicshopper/103-1420916-1341420) [DOWNLOAD: PDF] del Yankee Group (Autore) Prezzo di Listino: $595.00 Edizione: e-book (Acrobat Reader) Pagine: 3 (tre) Editore: MarketResearch.com; ISBN B00006334Y; (1 Novembre 2001)

;-) Il numero delle copie sembra interessante...

Capitolo 24. Storico delle revisioni / Ringraziamenti / La Fine

Le versioni x.y sono pubblicate su Internet.

Le versioni x.y.z sono in lavorazione e vengono pubblicate come file LyX e SGML su CVS. Poichè Deep Space 6 esegue il mirror di questi file SGML e li genera indipendentemente dalle versioni pubbliche di TLDP, queste verranno mostrate qui ed anche sui loro mirror.

24.1. Storico delle revisioni

24.1.1. Versioni 0.x

24.1.1.1. Versione Inglese (di Peter Bieringer)

Lo storico delle revisioni del documento originale è reperibile qui: TLDP / Linux+IPv6-HOWTO / Revision History (http://www.tldp.org/HOWTO/Linux+IPv6-HOWTO/revision-history.html)

24.1.1.2. Versione Italiana (di Michele Ferritto)

0.47.it.1

4-09-2004/MF Aggiornamento traduzione.

0.46.it.1

24-03-2004/MF Aggiornamento traduzione.

0.45.it.2

23-01-2004/MF Integrazione correzioni e revisione definitiva.

0.45.it.1

12-01-2004/MF Aggiornamento traduzione e revisione di massima.

0.44.2.it.1

16-10-2003/MF Prima traduzione e revisione di massima.

24.2. Ringraziamenti

Il modo più rapido per essere aggiunti a questo simpatico elenco consiste nell'inviare correzioni e/o aggiornamenti a me ;-).

Se si vuole realizzare una revisione importante, si può utilizzare il file LyX originale (si veda original source) e inviare l'output risultante da diff, poiché il diff rispetto al file SGML non è di molto aiuto.

24.2.1. Ringraziamenti principali

- David Ranch <dranch at trinnet dot net>: Per avermi incoraggiato a scrivere questo HOWTO, per i suoi commenti editoriali sulle prime poche revisioni ed i suoi contributi a vari test effettuati sul mio sito web IPv6. Lo ringrazio anche per le sue revisioni più importanti ed i suoi suggerimenti.
- Pekka Savola <pekkas at netcore dot fi>: Per revisioni importanti, spunti e suggerimenti.
- Martin F. Krafft <madduck at madduck dot net>: Per il controllo grammaticale e la revisione generale del documento.
- John Ronan <j0n at tssg dot wit dot ie>: Per il controllo grammaticale.
- Georg Käfer <gkaefer at gmx dot at>: Per aver rilevato una non corretta generazione del PDF (messo a posto dal manutentore di LDP Greg Ferguson), gli spunti per libri tedeschi, il grande elenco di URL, il controllo di tutte le URL, i tanti suggerimenti, le correzioni ed i contributi e la traduzione in tedesco.
- Michel Boucey <mboucey at free dot fr>: Per aver trovato refusi ed alcune URL non funzionanti, contribuito con alcuni suggerimenti ed URL ed aver effettuato la traduzione in francese.
- Michele Ferritto <m dot ferritto at virgilio dot it>: Per aver trovato errori e per la traduzione in italiano.

24.2.2. Altri ringraziamenti

24.2.2.1. Relativi alla tecnica del documento

Scrivere un HOWTO LDP come principiante (in LyX ed esportandolo a DocBook per conformità a SGML) non è così facile come qualcuno dice. Ci sono alcune strane trappole... Ciononostante, ringrazio:

- Gli autori della LDP Author Guide (http://www.tldp.org/LDP/LDP-Author-Guide/)
- B. Guillon: Per il suo DocBook con LyX HOWTO (http://perso.libertysurf.fr/bgu/doc/db4lyx/)

24.2.2.2. Ringraziamenti relativi al contenuto

I ringraziamenti per le correzioni ed i suggerimenti sono elencati qui, sicuramente in futuro cresceranno

- S. P. Meenakshi <meena at cs dot iitm dot ernet dot in>: Per il suggerimento di usare un programma di "invio mail" da shell su tcp_wrapper/hosts.deny
- Frank Dinies <FrankDinies at web dot de>: Per la correzione di un errore nella spiegazione dell'indirizzo IPv6
- John Freed <jfreed at linux-mag dot com>: Per aver trovato un errore nella spiegazione dell'indirizzo IPv6 multicast
- Craig Rodrigues <crodrigu at bbn dot com>: Per suggerimenti sulla configurazione di IPv6 per RHL
- Fyodor <fyodor at insecure dot org>: Mi ha fatto notare delle informazioni obsolete riguardo a nmap
- Mauro Tortonesi <mauro at deepspace6 dot net>: Per alcuni suggerimenti

- Tom Goodale <goodale at aei-potsdam dot mpg dot de>: Per alcuni suggerimenti
- Martin Luemkemann <mluemkem at techfak dot uni-bielefeld dot de>: Per un suggerimento
- Jean-Marc V. Liotier <jim at jipo dot com>: Ha trovato un errore
- Yaniv Kaul <ykaul at checkpoint dot com>: Ha trovato un errore
- Arnout Engelen <arnouten at bzzt dot net>: Per aver inviato una nota su di una bozza adottata come RFC
- Stephane Bortzmeyer <bortzmeyer at nic dot fr>: Per aver dato un contributo sulla configurazione persistente su Debian
- lithis von saturnsys <lithis at saturnsys dot com>: Per aver riportato un URL errato
- Guy Hulbert <gwhulbert at rogers dot com>: Per aver inviato una nota per indicare che probabilmente la RFC1924 è un pesce d'Aprile
- Tero Pelander <tpeland at tkukoulu dot fi>: Per aver riportato un URL non funzionante
- Walter Jontofsohn <wjontof at gmx dot de>: Suggerimenti per SuSE Linux 8.0/8.1
- Benjamin Hofstetter <benjamin dot hofstetter at netlabs dot org>: Per aver riportato un URL sbagliato
- J.P. Larocque <piranha at ely dot ath dot cx>: Per aver riportato l'URL dell'archivio degli utenti della mailing list users at ipv6 dot org
- Jorrit Kronjee <jorrit at wafel dot org>: Per aver segnalato URL non funzionanti
- Colm MacCarthaigh <colm dot maccarthaigh at heanet dot ie>: Suggerimenti per il problema di sendfile con Apache2
- Tiago Camilo <tandre at ipg dot pt>: Ha contribuito con alcune URL sull'IPv6 mobile
- Harald Geiger: Per aver riportato un bug su come è descritto il conteggio del bit universale/globale
- Bjoern Jacke <bjoern at j3e dot de>: Per avermi sollecitato ad aggiornare alcune informazioni relative a xinetd
- Christoph Egger <cegger at chrrr dot com>: Per aver segnalato che "ip" ha dei problemi con gli indirizzi compatibili IPv4 su SuSE Linux 9.0 ed avermi stimolato per aggiungere un suggerimento sull'esempio del 6to4-radvd
- David Lee Haw Ling <hawling at singnet dot com dot sg>: Per aver inviato informazioni su di un tunnel broker
- Michael H. Warfield <mhw at iss dot net>: Per aver inviato una nota sul suffisso per i router 6to4
- Tomasz Mrugalski <thomson at klub dot com dot pl>: Invio di aggiornamenti per la sezione DHCPv6
- Jan Minar <jjminar at fastmail dot fm>: Segnalazione di piccoli errori
- Kalin KOZHUHAROV <kalin at tar dot bz>: Sistemazione di una spiegazione non troppo chiara
- Roel van Dijk <rdvdijk at planet dot nl>: Segnalazione di URL non funzionanti
- Catalin Muresan <catalin dot muresan at astral dot ro>: Segnalazione di piccoli errori
- Dennis van Dok <dvandok at quicknet dot nl>: Segnalazione di piccoli errori

24.3. La Fine

Grazie per la lettura. Spero che questo possa essere d'aiuto!

Se si ha qualche domanda, ci si iscriva alla mailing list appropriata e si descriva il proprio problema fornendo più informazioni possibili.