

Introduzione a INN -- InterNet News



Installazione e quadro generale di INN	710
Le variabili di ambiente	712
Caratteri jolly	712
Configurazione minima per l'uso locale puro e semplice	713
File «/etc/news/inn.conf»	715
File «/etc/news/distrib.paths»	717
File «/etc/news/newsfeeds»	718
File «/etc/news/expire.ctl»	721
File «/etc/news/nntp.access»	723
File «/var/lib/news/active»	725
File «/var/lib/news/history»	727
Demoni e altri programmi per l'uso minimo di INN	728
Avvio e conclusione del servizio NNTP	728
Utilizzo di «ctlinnd»	730
Operazioni di routine	733
Feed in ingresso utilizzando il protocollo NNTP	735
File «/etc/news/hosts.nntp»	735
File «/etc/news/incoming.conf»	736
Feed continuo in uscita utilizzando il protocollo NNTP	737
File «/etc/news/innfeed.conf»	737
File «/etc/news/newsfeeds»	738

Feed periodico in uscita utilizzando il protocollo NNTP	740
File «/etc/news/nntpsendctl»	740
File «/etc/news/newsfeeds»	741
Utilizzo di «nntpsend»	742
Ritrasmissione di articoli attraverso la posta elettronica	742
Prelievo di articoli utilizzando il protocollo NNTP	743
Utilizzo di «nntpget»	744
Replicazione dei gruppi di un altro sito	745
Riferimenti	746

INN, o InterNet News, è probabilmente il sistema più completo per la gestione delle news di Usenet. Purtroppo la sua documentazione non è soddisfacente, nel senso che presume una buona conoscenza di Usenet e il funzionamento del software precedente a INN (Bnews e C News). In questo capitolo viene dato un minimo di informazioni necessario per poter allestire un servizio di news chiuso, con qualche accenno alle possibilità di diffusione degli articoli assieme ad altri nodi, utilizzando il protocollo NNTP.

Installazione e quadro generale di INN

«

Per installare INN, data la sua complessità, è meglio se si dispone di un pacchetto già compilato e preparato per la propria distribuzione GNU. L'installazione dovrebbe utilizzare, in particolare, le collocazioni seguenti:

<code>‘/var/spool/news/’</code>	l’inizio della gerarchia contenente gli articoli dei gruppi gestiti;
<code>‘/var/lib/news/’</code>	l’inizio della gerarchia contenente i file amministrativi;
<code>‘/usr/lib/news/’</code>	l’inizio della gerarchia contenente i programmi, gli script e le librerie;
<code>‘/etc/news/’</code>	i file di configurazione che può essere opportuno modificare.

Naturalmente dovrebbero essere disponibili anche dei file per le pagine di manuale, collocati nelle posizioni consuete; inoltre dovrebbe essere disponibile un minimo di documentazione assieme ad alcuni esempi di configurazione.

I programmi, cioè i file eseguibili e gli script, potrebbero trovarsi solo nella directory `‘/usr/lib/news/bin/’`, aggiungendo qualche collegamento nelle directory normali (`‘/usr/bin/’` e `‘/usr/sbin/’`) solo per ciò che è stato ritenuto più importante.

Il funzionamento di INN dipende anche dall’esecuzione periodica di alcune operazioni amministrative, attraverso il controllo del sistema Cron. Per questo, se la propria distribuzione è stata organizzata anche in questo senso, è probabile che siano stati collocati alcuni script all’interno delle directory `‘/etc/cron*/’`, che poi vengono avviati in modo automatico in base alla configurazione predefinita di `‘/etc/crontab’`.

Infine è da ricordare che tutti i file e le directory di INN devono appartenere all’utente (e probabilmente anche al gruppo) **‘news’**. Anche i processi devono funzionare con i privilegi di questo utente amministrativo, con un’unica eccezione data dal programma **‘innd’**

che si occupa di fornire il servizio NNTP, che dovendo accedere alla porta `'nntp'`, corrispondente al numero 119, deve avere inizialmente i privilegi dell'utente `'root'`.

Le variabili di ambiente

«

INN dipende da un reticolo di script e programmi che sono controllati da una serie di variabili di ambiente. Queste sono definite all'interno di pezzi di script che vengono incorporati dagli altri e che generalmente risiedono nella directory `'/usr/lib/news/lib/'`. Per la precisione, dal momento che gli script possono essere di vario tipo e si vuole lasciare la possibilità di estendere il sistema a piacere, esiste un file di dichiarazione di queste variabili per ogni interprete: `'innshellvars'` per la shell Bourne, `'innshellvars.csh'` per la shell C, `'innshellvars.pl'` per l'interprete Perl e `'innshellvars.tcl'` per l'interprete Tcl. Bisogna sapere che se si intende modificare qualcosa in uno di questi file (ma sarebbe meglio evitare di farlo), occorre ripetere le modifiche anche sugli altri.

Generalmente si vede l'utilizzo di `'innshellvars'`, attraverso un'istruzione di incorporazione come quella seguente:

```
#!/bin/sh
...
. /usr/lib/news/lib/innshellvars
...
```

Caratteri jolly

«

In molte situazioni, i file di configurazione di INN ammettono l'uso di caratteri jolly (o metacaratteri), secondo le convenzioni stabilite nella pagina di manuale *wildmat(3)*. In linea di massima si può dire che si utilizzano le convenzioni normali riferite alle shell per l'u-

so dell'asterisco e del punto interrogativo. In particolare è ammesso anche la descrizione di intervalli di caratteri attraverso la notazione '[...]' e '[^...]'; inoltre è possibile togliere il significato speciale di un simbolo prefissandolo con una barra obliqua inversa.

Tabella u55.3. Modelli secondo *wildmat(3)*.

Modello	Descrizione
<code>\x</code>	Corrisponde al valore letterale di <i>x</i> .
<code>?</code>	Un carattere singolo.
<code>*</code>	Qualunque sequenza di caratteri, anche nulla.
<code>[xy...]</code>	Un carattere singolo tra quelli indicati tra parentesi.
<code>[^xy...]</code>	Un carattere singolo esclusi quelli indicati tra parentesi.
<code>[x-y]</code>	Un carattere singolo tra l'intervallo di <i>x</i> e <i>y</i> .
<code>[^x-y]</code>	Un carattere singolo escluso l'intervallo di <i>x</i> e <i>y</i> .

Configurazione minima per l'uso locale puro e semplice

Il componente più importante di INN è il demone '**innd**'. Questo viene avviato tramite uno script che potrebbe essere necessario ritoccare a seconda del modo in cui si vuole organizzare il sistema. Potrebbe trattarsi di '`/etc/rc.d/rc.news`', o qualcosa di simile, ma per controllarlo dovrebbe essere stato predisposto un altro script, per esempio '`/etc/rc.d/init.d/innd`', in grado di accettare i soliti comandi ('**start**', '**stop**', ecc.) e che soprattutto lo avvii con i pri-

vilegi dell'utente **'news'**. Si osservi a questo proposito il commento introduttivo di **'rc.news'**:

```
#!/bin/sh
## $Revision: 1.19 $
## News boot script.  Runs as "news" user.  Requires inndstart be
## setuid root.  Run from rc.whatever as:
##     su news -c /path/to/rc.news >/dev/console
```

In una parte avanzata di questo script ci dovrebbe essere qualcosa che assomiglia al pezzo seguente, dove si intende che tutto dipende dal contenuto delle variabili di ambiente che si possono vedere:

```
## Start the show.
echo 'Starting innd.'
eval ${WHAT} ${RFLAG} ${INNFLAGS}
```

I nomi e il numero delle variabili di ambiente indicate cambia da una distribuzione GNU all'altra, ma è importante sapere come viene avviato **'innd'** in modo preciso, perché a seconda di alcuni particolari della configurazione si devono utilizzare delle opzioni determinate. Analizzando il file che è stato usato per mostrare questo esempio, si osserva che:

```
## Pick ${INND} or ${INNDSTART}
WHAT=${INNDSTART}
```

il comando per avviare **'innd'** proviene dal contenuto della variabile di ambiente **'INNDSTART'**, che è stata definita all'interno di **'/usr/lib/news/lib/inshellvars'** (e dopo qualche ricerca si scopre che si tratta di **'/usr/lib/news/bin/inndstart'**);

```
## RFLAG is set below; set INNFLAGS in inn.conf(5)
RFLAG=""
```

Quindi si trova che la variabile **'RFLAG'** non contiene alcunché, ma potrebbe essere usata per inserire delle opzioni particolari; inoltre, da quanto si legge nel commento, la variabile **'INNFLAGS'** viene definita da qualche parte in base a una direttiva del file **'/etc/inn.'**

`conf`' e comunque dovrebbe essere inesistente.

In pratica, `inn` o `innstart` dovrebbe essere avviato senza opzioni. Se ne vengono trovate, è bene toglierle, almeno fino a che non è stato superato il primo stadio di utilizzo di INN.

File «`/etc/news/inn.conf`»

Dal nome, `/etc/inn.conf`, si intende che si tratti del file di configurazione più importante di INN. Il sistema di gestione dei pacchetti della propria distribuzione GNU/Linux dovrebbe provvedere a predisporlo in modo da permettere a INN di funzionare nel proprio sistema, ma è importante dargli un'occhiata ed eventualmente modificarlo. In generale conviene lasciare stare tutto com'è, tranne ciò che interessa.

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all'interno della pagina di manuale *inn.conf(5)*. In breve, le righe vuote, quelle bianche e quelle che iniziano con il simbolo '#' vengono ignorate. Le direttive sono composte da una sorta di assegnamento descritto secondo la sintassi seguente:

nome : *valore*

In particolare, tra i due punti che seguono il nome e il valore assegnato, ci deve essere almeno uno spazio orizzontale di qualunque tipo; inoltre, se il valore assegnato è rappresentato da una stringa contenente degli spazi, questa non deve essere racchiusa tra virgolette.

Le direttive su cui è molto importante intervenire sono poche; riguardano la definizione dell'«organizzazione» predefinita e i nomi

con cui si deve identificare il nodo che offre il servizio. L'organizzazione è un nome che viene abbinato al campo '**Organization:**' nell'intestazione degli articoli; di solito dovrebbe essere definito dal programma cliente dell'utente che spedisce l'articolo.

```
organization:          Azienda Brot
```

L'esempio mostra un'idea di ciò che potrebbe essere indicato come organizzazione. Si osservi il fatto che non sono state usate le virgolette per delimitare il nome. Questa informazione potrebbe essere definita anche attraverso la variabile di ambiente '**ORGANIZATION**', la quale, se esiste, prevale su quanto definito nel file '/etc/inn.conf'.

Un problema un po' più delicato riguarda invece la definizione delle direttive '**server:**', '**fromhost:**' e '**pathhost:**'. Per prima cosa conviene decidere il nome a dominio del servizio NNTP. Di solito si crea un alias opportuno, qualcosa che inizi per '**news.***', come nell'esempio seguente:

```
server:                news.brot.dg
```

Anche in questo caso c'è la possibilità di utilizzare una variabile di ambiente che, se esiste, prevale sulla direttiva: si tratta di '**NNTPSERVER**'.

Ma il nome canonico dell'elaboratore potrebbe essere *dinkel.brot.dg*. Nell'intestazione degli articoli deve apparire il campo '**From:**' e il campo '**Path:**'; per queste indicazioni si presentano due possibilità: il nome canonico dell'elaboratore oppure il nome a dominio utilizzato nella posta elettronica. In pratica, seguendo l'esempio si dovrebbero indicare le direttive seguenti:


```
pathhost:          dinkel.brot.dg
fromhost:          dinkel.brot.dg
```

Se però la rete locale identificabile con il nome *brot.dg* riceve la posta elettronica direttamente con il dominio *brot.dg* (*tizio@brot.dg*), potrebbe essere il caso di cambiare la definizione nel modo seguente:

```
pathhost:          brot.dg
fromhost:          brot.dg
```

File «*/etc/news/distrib.paths*»

Gli articoli possono distinguersi, oltre che per gruppi, anche per «distribuzioni». Si tratta del campo '**Distribution:**' che potrebbe apparire nell'intestazione di un messaggio. Se chi spedisce il messaggio non provvede a indicare il campo '**Distribution:**', è opportuno che sia stabilito qualche valore predefinito in base al gruppo a cui è diretto. Questo è lo scopo del file '*/etc/news/distrib.paths*'.

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all'interno della pagina di manuale *distrib.paths(5)*. In breve, le righe vuote, quelle bianche e quelle che iniziano con il simbolo '#' vengono ignorate. Le direttive sono composte da record suddivisi in tre elementi separati da due punti verticali (':'), secondo la sintassi seguente:

```
peso : modello : distribuzione
```

Il primo elemento è un numero maggiore di zero che serve a stabilire un ordinare di importanza delle direttive quando un certo gruppo può corrispondere a più modelli di record differenti: in quel caso si prende quello con il peso maggiore. Si osservi che l'esistenza del peso in questi record, rende indifferente l'ordine in cui questi appaiono.

Il secondo elemento è il nome di un gruppo o un modello che utilizza caratteri jolly secondo la convenzione di *wildmat(3)*: quando un articolo che non ha il campo **‘Distribution:’** nell’intestazione corrisponde a un modello (e non ce ne sono altri di peso maggiore che possono corrispondere) gli viene attribuita la distribuzione il cui nome si colloca nell’ultimo elemento di questo record.

```
1:*:world
10:test:local
10:test.*:local
10:local.*:local
```

L’esempio mostra una classificazione molto semplice: tutti gli articoli sono classificati come appartenenti alla distribuzione **‘world’**, tranne quelli del gruppo **‘test’** e dei gruppi che iniziano per **‘test.’** e **‘local.’**, che invece sono classificati come facenti parte della distribuzione **‘local’**.

File «/etc/news/newsfeeds»

«

Il file **‘/etc/news/newsfeeds’** è molto importante perché definisce in che modo è organizzato il feed, ovvero la propagazione dei gruppi. Inizialmente conviene occuparsi solo di ciò che si vuole ottenere attraverso la voce **‘ME’**, secondo la tradizione del software precedente a INN.

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all’interno della pagina di manuale *newsfeeds(5)* e qui vengono descritti solo alcuni aspetti elementari. In breve, le righe vuote, quelle bianche e quelle che iniziano con il simbolo **‘#’** vengono ignorate. Le direttive sono composte da record separati in elementi attraverso due punti verticali (**‘:’**) come descritto dalla sin-

tassi seguente, potendo essere continuate su più righe quando alla fine di una riga appare il simbolo ‘\’.

```
nome_sito [ / esclusione ] : modelli [ / distribuzioni ] : opzioni : parametri
```

Il primo elemento serve a definire il nome del sito verso cui si vuole siano inviati gli articoli. Si tratta di un nome relativo alla configurazione, in quanto il suo scopo potrebbe anche essere solo quello di creare un archivio degli articoli su un file. Spesso, per ciò che riguarda nomi riferiti a voci locali, si utilizza un punto esclamativo finale per evitare confusione con i nomi di siti reali. L'elemento può essere completato da un elenco di esclusione che si distingue in quanto separato da una barra obliqua ('/'). Ciò permette di indicare una serie di nomi di siti che, se presenti nel percorso dell'articolo (il campo '**Path:**'), fanno sì che questo venga escluso. Volendo essere più precisi, la sintassi per il primo elemento potrebbe essere espressa nel modo seguente:

```
nome_sito [ / nome_escluso [ , nome_escluso ] ... ]
```

Il secondo elemento serve a definire un elenco di modelli riferiti ai gruppi che si vogliono selezionare, seguito eventualmente da un elenco di distribuzioni. Se vengono indicate anche le distribuzioni, allora sono accettati solo gli articoli che appartengono a una di quelle. Volendo rendere con maggiore dettaglio la sintassi per il secondo elemento del record, si può definire lo schema seguente:

```
modello [ , modello ] ... [ / distribuzione [ , distribuzione ] ... ]
```

I modelli dei gruppi possono usare i soliti caratteri jolly e possono essere indicati anche in forma di negazione, attraverso il prefisso di un punto esclamativo. I nomi delle distribuzioni non possono contenere caratteri jolly, ma ammettono l'uso del punto esclamativo per negare un nome.

Gli ultimi due elementi del record sono un po' particolari e vengono descritti solo quando necessario.

Volendo realizzare un servizio locale e chiuso di news, all'interno di questo file è sufficiente collocare la direttiva seguente, eliminando o commentando tutto il resto.

```
ME:***
```

Il nome '**ME**' è una parola chiave che rappresenta il sito locale; di conseguenza si tratta del record che definisce quali gruppi e quali articoli vengono gestiti o accettati. L'asterisco nel secondo elemento indica che sono accettati tutti i gruppi e non si fanno discriminazioni per quanto riguarda la distribuzione eventuale. Gli ultimi due elementi non servono per questo tipo di situazione e sono vuoti.

Volendo essere un po' più dettagliati, magari in previsione di un'apertura all'esterno, si potrebbero definire i modelli relativi ai gruppi che si pensa di gestire, comprendendo anche quelli che devono restare relegati all'ambito locale.

```
ME:*,!junk,!control*,!local*::
```

In questo caso si intendono ricevere tutti gli articoli di qualunque gruppo, escludendo il gruppo '**junk**', i gruppi che iniziano per '**control**' e quelli che iniziano per '**local**'. Questi divieti riguardano solo la possibilità di ricevere articoli da un sito Usenet corrispondente e non limitano invece l'invio locale.

```
ME:*,@alt.binaries.*,!junk,!control*,!local*::
```

Questa è un'estensione dell'esempio precedente, in cui si utilizza una notazione che non è ancora stata descritta: il simbolo '@' posto davanti al modello '**alt.binaries.***' stabilisce che non vengono accettati articoli che siano stati inviati anche ai gruppi di quel modello. In pratica, se un articolo è indirizzato a un gruppo della gerarchia '**alt.binaries**' e anche a gruppi che si intendono gestire, questo articolo viene escluso completamente.

In generale, se non si sanno gestire le distribuzioni, sarebbe meglio evitare di porre delle limitazioni a tale riguardo.

Inizialmente sarebbe bene limitarsi a gestire solo il record '**ME**', commentando tutto il resto se dovesse esserci qualcosa, verificando che il demone '**innd**' sia avviato senza argomenti particolari.

File «/etc/news/expire.ctl»

Periodicamente dovrebbe essere avviata una procedura per l'eliminazione degli articoli troppo vecchi. Questo viene attuato attraverso il programma '**expire**', che di solito viene avviato tramite uno script. Il file di configurazione di '**expire**' è '/etc/news/expire.ctl'.

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all'interno della pagina di manuale *expire.ctl(5)*. Le righe vuote, quelle bianche e quelle che iniziano con il simbolo '#' vengono ignorate. Le direttive sono composte da record di due tipi, secondo le sintassi seguenti:

```
/remember/ :n_giorni
```

```
modello :M | U | A :n_giorni_min :n_giorni_predefinito :n_giorni_max
```

La prima delle due sintassi si riferisce al tempo in giorni durante il quale si deve conservare memoria delle stringhe di identificazione degli articoli che sono passati per il sito; in pratica si tratta del contenuto del campo **Message-ID:** di ogni articolo. Questa indicazione è molto importante e la durata in questione non può essere troppo breve se non si vuole rischiare di ricevere nuovamente un articolo che in precedenza è già stato visto.

La seconda forma si riferisce ai record successivi. Con questi è possibile distinguere i tempi di scadenza degli articoli in base al gruppo a cui sono stati destinati ed eventualmente anche al fatto che questi siano moderati o meno. Per questo, nel secondo elemento si indica una lettera e precisamente: **M** identifica i gruppi moderati, **U** quelli non moderati e **A** tutti i gruppi senza distinguere su questo particolare.

Gli ultimi tre elementi delimitano la durata minima e massima di validità degli articoli; in particolare il valore intermedio è quello predefinito nel caso in cui l'articolo non disponga di questa informazione.

Si osservi l'esempio seguente:

```
/remember/:21  
*:A:7:10:14  
*:M:14:17:21  
test*:A:1:1:1
```

Bisogna considerare che i gruppi rientrano sotto il controllo dell'ultimo record che coincide. In questo caso: le stringhe di identificazione degli articoli vengono conservate per 21 giorni; tutti i gruppi vengono conservati per un minimo di sette giorni, fino a un massimo di 14 (con un valore predefinito di 10); però i gruppi moderati sono conservati più a lungo (da 14 a 21 giorni); ma i gruppi che iniziano per **'test'** sono conservati solo un giorno. Sarebbe stato molto diverso se l'ordine fosse il seguente:

```
/remember/:21
test*:A:1:1:1
*:M:14:17:21
*:A:7:10:14
```

In questo caso, tutti i gruppi verrebbero conservati per un minimo di sette giorni, fino a un massimo di 14, compresi quelli che iniziano per **'test'**.

File «/etc/news/nnrp.access»

Il demone **'innd'** si avvale a sua volta di **'nnrpd'** per le connessioni con i programmi clienti (attraverso il protocollo NNTP) che si limitano a consultare gli articoli e a spedirne di nuovi. Il file di configurazione di **'nnrpd'** è `'/etc/news/nnrp.access'` con il quale si regolano questi accessi. <<

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all'interno della pagina di manuale *nnrp.access(5)*. Le righe vuote, quelle bianche e quelle che iniziano con il simbolo **'#'** vengono ignorate. Le direttive sono composte da record secondo la sintassi seguente:

```
modello_nodi : permessi : [utente] : [parola_d'ordine] : modello_gruppi
```

Il primo elemento permette di rappresentare un gruppo di nodi che possono accedere attraverso i caratteri jolly di INN. Il secondo serve a indicare i permessi di accesso, che sono costituiti dalla possibilità di leggere gli articoli (in questo caso si usa la lettera ‘R’, *read*) e dalla possibilità di spedire degli articoli (lo si rappresenta con la lettera ‘P’, *post*). Il terzo e il quarto elemento, se utilizzati, permettono di indicare un nominativo-utente e una parola d’ordine in chiaro. Il quinto elemento permette di individuare i gruppi a cui ci si riferisce, attraverso l’uso dei soliti caratteri jolly.

Come al solito, viene preso in considerazione l’ultimo record corrispondente all’accesso che viene tentato, per cui conviene mettere prima i record generici e alla fine quelli più dettagliati. In generale, bisogna evitare di concedere l’accesso a tutti, tanto che il file di configurazione predefinito viene fornito come si vede nell’esempio seguente:

```
# Default to no access
*:: -no- : -no- :!*
# Allow access from localhost
localhost:RP:::*
```

In pratica, si vieta espressamente l’accesso indiscriminato attraverso il record

```
*:: -no- : -no- :!*
```

dove quel ‘-no-’ ‘-no-’ è solo un modo appariscente per far capire che si tratta di una politica assolutamente sconsigliabile, per cui si concede l’accesso (sia per la lettura che per la spedizione di articoli) solo al nodo locale.


```
localhost:RP:::*
```

In sostituzione di questo record predefinito si potrebbe concedere l'accesso a tutta la propria rete locale, in un modo simile a quello seguente:

```
*.brot.dg:RP:::*
```

L'esempio seguente mostra in particolare un record con cui si concede l'accesso a qualunque nodo per la lettura dei gruppi `'comp.os.linux.*'`.

```
*:R:::comp.os.linux.*
```

L'accesso può essere limitato in base all'indicazione di un nominativo-utente e di una parola d'ordine, come nell'esempio seguente:

```
*.brot.dg:RP:::*  
*:RP:ignoto:segreto:*
```

In questo caso, l'idea è quella di permettere l'accesso indiscriminato dai nodi appartenenti al dominio *brot.dg* e di concederlo anche all'esterno, a patto che si fornisca il nominativo `'ignoto'` e la parola d'ordine `'segreto'`.

Evidentemente, se il file `'/etc/news/nnrp.access'` contiene l'indicazione di accessi controllati da una parola d'ordine, è necessario che non sia concessa la lettura di questo file agli utenti comuni.

File `«/var/lib/news/active»`

Inizialmente, sono disponibili alcuni gruppi amministrativi (`'control'`, `'junk'`, `'to'`) e uno di prova, `'test'`. Per fare qualche prova, ciò è più che sufficiente. Volendo aggiungere qualche gruppo si potrebbe modificare il file `'/var/lib/news/active'`, anche

se per questo sarebbe meglio utilizzare il programma ‘**ctlinnd**’ che viene descritto in seguito. È opportuno comunque conoscere il contenuto di questo file, che può contenere solo righe composte da quattro elementi secondo la sintassi seguente:

```
nome_del_gruppo n_iniziale n_finale opzione
```

La cosa migliore per cominciare è dare un’occhiata alla situazione iniziale.

```
control 0000000000 0000000001 y
junk 0000000000 0000000001 y
test 0000000000 0000000001 y
to 0000000000 0000000001 y
```

Il primo elemento rappresenta il nome del gruppo, il secondo rappresenta il numero attuale degli articoli presenti e il terzo indica il numero successivo. Per esempio, se si leggesse

```
test 0000000010 0000000011 y
```

significherebbe che nella directory ‘/var/spool/news/test/’ c’è, o c’è stato, il file ‘10’, mentre il prossimo articolo in questo gruppo verrebbe inserito nel file ‘11’.

L’ultimo elemento serve a stabilire il funzionamento del gruppo. La lettera ‘**y**’ rappresenta un gruppo per il quale sono ammesse le spedizioni di articoli da parte dei clienti; in pratica rappresenta la situazione più comune. Per conoscere le altre opzioni disponibili e il loro significato si può consultare la pagina di manuale *active(5)*, comunque vengono riepilogate nella tabella u55.26.

Tabella u55.26. Elenco delle opzioni riferite ai gruppi all'interno del file 'active'.

Opzione	Descrizione
y	È ammessa la lettura e la spedizione di articoli.
n	È ammessa solo la lettura degli articoli.
x	Il gruppo è disabilitato localmente.
j	Il gruppo non viene gestito.
m	Il gruppo è moderato e le spedizioni devono essere approvate.
= <i>gruppo</i>	Gli articoli vengono spostati nel gruppo indicato.

Come già accennato, inizialmente è meglio modificare questo file solo attraverso il programma 'ctlinnd'.

File «/var/lib/news/history»

L'archivio storico degli articoli che sono stati visti viene conservato nel file '/var/lib/news/history' e in altri file con la stessa radice e con un'estensione particolare ('history.*'). Generalmente, questo deve essere creato la prima volta che si installa INN. Si procede semplicemente nel modo seguente:

```
# su news [Invio]
```

Si ottengono i privilegi dell'utente 'news', dal momento che devono essere creati file che appartengono a questo nome.

```
news$ touch /var/lib/news/history [Invio]
```

Viene creato il file `‘/var/lib/news/history’` vuoto.

```
news$ /usr/lib/news/bin/makehistory -ro [Invio]
```

Crea gli altri file abbinati per la gestione dell'archivio storico e l'operazione è conclusa.

Demoni e altri programmi per l'uso minimo di INN

«

Dopo aver definito una configurazione minima, anche senza aver aggiunto alcun gruppo a quelli predefiniti, si può fare qualche esperimento con l'uso di un cliente come Netscape o qualcosa di simile. Prima però occorre avviare il servizio NNTP.

Avvio e conclusione del servizio NNTP

«

Il servizio NNTP è gestito principalmente dal demone `‘innnd’` che, per quanto riguarda gli accessi da parte di clienti per la lettura e la spedizione di articoli, si avvale a sua volta di `‘nnrpd’`. In pratica, a seconda della situazione, può capitare di vedere funzionare solo `‘innnd’`, oppure anche una o più copie di `‘nnrpd’` come sottoprocessi controllati sempre da `‘innnd’`. All'inizio del capitolo si è accennato al fatto che normalmente `‘innnd’` viene avviato attraverso uno script che potrebbe chiamarsi `‘rc.news’` e si trova probabilmente nella directory `‘/etc/rc.d/’`. È già stato spiegato anche che conviene dargli un'occhiata ed eventualmente può essere il caso di modificarlo. Oltre a `‘innnd’`, questo script dovrebbe avviare `‘innwatch’` per controllare che il sistema di news non superi lo spazio disponibile nel file system. In pratica, una volta avviato il servizio, si potrebbero osservare questi processi:

```
init-+-...
  |-...
  |-innd---2*[nnrpd]
  |-...
  |-rc.news---innwatch---sleep
  |-...
  \-...
```

Per avviare il servizio NNTP attraverso lo script `rc.news` occorre accedere con i privilegi dell'utente `news`.

```
news$ /etc/rc.d/rc.news [Invio]
```

Per disattivare il servizio, si utilizza un programma apposito per inviare un comando adatto a `innd`: si tratta di `ctlinnd`. Nell'esempio mostrato sotto, prima viene inviato il comando `throttle` per bloccare il servizio, quindi viene inviato il comando `shutdown` per fare in modo che `innd` concluda del tutto il suo lavoro.

```
news$ /usr/lib/news/bin/ctlinnd throttle 'blocco del servizio'
[Invio]
```

```
news$ /usr/lib/news/bin/ctlinnd shutdown 'chiusura del
servizio' [Invio]
```

Dal momento che lo script `rc.news` aveva avviato anche `innwatch`, occorre preoccuparsi di eliminare anche questo processo, per esempio nel modo seguente:

```
news$ killall innwatch [Invio]
```

Per semplificare tutto questo, la propria distribuzione GNU dovrebbe avere organizzato uno script aggiuntivo da collocarsi all'interno di `/etc/rc.d/init.d/` o in una posizione simile, allo scopo di poter avviare e concludere il servizio in modo più semplice:

```
/etc/rc.d/init.d/innd start | stop
```

Le prime volte è probabile che il servizio non si avvii, a causa di errori di configurazione. Evidentemente è necessario osservare i file delle registrazioni per vedere se appare la segnalazione della ragione per cui **'innd'** non parte. Spesso si tratta di file mancanti o di errori nei permessi dei file che non consentono l'accesso all'utente di sistema **'news'**.

Utilizzo di «ctlinnd»

«

Il programma **'ctlinnd'** è uno dei pochi che potrebbe risultare accessibile nell'ambito dei percorsi normali di ricerca degli eseguibili. In pratica, potrebbe esserci un collegamento simbolico nella directory **'/usr/sbin/'** che permette di avviarlo senza dover indicare il percorso (**'/usr/lib/news/bin/'**).

```
ctlinnd [opzioni] comando [argomenti_del_comando]
```

'ctlinnd' serve solo a inviare un comando a **'innd'**, il quale risponde e l'esito determina il modo in cui **'ctlinnd'** termina. Generalmente si ottiene un **'Ok'** se tutto va bene, salvo alcuni comandi per i quali non viene generata alcuna risposta. I tipi di comando che possono essere usati sono molti e qui ne vengono descritti solo alcuni. Per conoscere l'uso dettagliato di **'ctlinnd'** conviene consultare la pagina di manuale *ctlinnd(8)*.

Comando	Descrizione
<p>pause <i>motivazione</i></p>	<p>Il comando 'pause' serve a impedire le nuove connessioni, pur mantenendo quelle esistenti. Subito dopo viene chiuso l'archivio storico. L'argomento di questo comando è una stringa che serve a spiegare la ragione, in modo che possa essere annotata nel registro del sistema.</p>
<p>throttle <i>motivazione</i></p>	<p>Il comando 'throttle' serve a chiudere le connessioni esistenti e a rifiutarne delle nuove. Subito dopo viene chiuso l'archivio storico. L'argomento di questo comando è una stringa che serve a spiegare la ragione, in modo che possa essere annotata nel registro del sistema.</p>
<p>go [<i>motivazione</i>]</p>	<p>Questo comando viene usato dopo aver utilizzato 'pause' o 'throttle' per riaprire l'archivio storico e consentire nuovamente le connessioni. La stringa di motivazione dovrebbe coincidere con quella utilizzata per interrompere il servizio.</p> <p>Il comando 'go' può essere usato per ripristinare il servizio dopo altri tipi di comandi, come descritto all'interno di <i>ctlinnd(8)</i>.</p>

Comando	Descrizione
shutdown <i>motivazione</i>	Il comando 'shutdown' serve a chiudere il servizio NNTP. Di solito è preferibile utilizzarlo dopo un comando 'throttle' . L'argomento di questo comando è una stringa che serve a spiegare la ragione, in modo che possa essere annotata nel registro di sistema.
newgroup <i>nome_gruppo</i> ← ↪ [<i>opzione</i> [<i>creatore</i>]]	Il comando 'newgroup' permette di creare un nuovo gruppo localmente. L'opzione si riferisce a ciò che può essere messo nel quarto elemento dei record del file <code>"/var/lib/news/active"</code> ; se non viene specificato, si tratta della lettera 'y' che abilita l'uso normale. L'ultimo argomento è il nome del creatore del gruppo. La creazione del gruppo aggiorna il file <code>"/var/lib/news/active"</code> e genera le directory necessarie a partire da <code>"/var/spool/news/"</code> .
rmggroup <i>nome_gruppo</i>	Questo comando elimina un gruppo attraverso la modifica del file <code>"/var/lib/news/active"</code> . La directory del gruppo eliminato non viene toccata e si lascia fare alla procedura di eliminazione degli articoli scaduti.

Segue la descrizione di alcuni esempi.

- news\$ /usr/lib/news/bin/ctlinnd throttle 'blocco del servizio' [Invio]

Blocca il servizio NNTP senza chiudere il funzionamento di **'innd'**.

- `news$ /usr/lib/news/bin/ctlinnd shutdown 'chiusura del servizio' [Invio]`

Blocca il servizio NNTP e termina il funzionamento di **'innd'**.

- `news$ /usr/lib/news/bin/ctlinnd newgroup prova.discussioni.varie [Invio]`

Crea il gruppo **'prova.discussioni.varie'** e gli attribuisce l'opzione **'y'** in modo predefinito.

- `news$ /usr/lib/news/bin/ctlinnd rmggroup prova.discussioni.varie [Invio]`

Elimina il gruppo **'prova.discussioni.varie'** senza eliminare materialmente gli articoli ancora esistenti.

Operazioni di routine

L'eliminazione degli articoli troppo vecchi, secondo quanto configurato con il file `/etc/news/expire.ctl`, viene fatta dal programma **'expire'**, che però viene avviato solitamente tramite lo script **'news.daily'**. In pratica, attraverso il sistema Cron viene avviato giornalmente un comando come quello seguente:

```
su - news -c "/usr/lib/news/bin/news.daily"
```

Eventualmente, **'news.daily'** viene avviato con qualche opzione, come nel caso seguente:

```
su - news -c "/usr/lib/news/bin/news.daily delayrm"
```

Lo script **'news.daily'** serve anche per sistemare i file delle registrazioni (che dovrebbero trovarsi nella directory `/var/log/`

news/'), provvedendo alla loro rotazione, oltre che per avvisare l'amministratore del servizio, cioè l'utente **'news'**, per mezzo della posta elettronica. **'news.daily'** accetta delle opzioni nella riga di comando, composte da delle parole chiave:

```
news.daily [opzione] ...
```

Gli argomenti possibili sono molti e qui vengono descritte solo alcune delle opzioni. Eventualmente si può consultare la pagina di manuale *news.daily(8)*.

Opzione	Descrizione
delayrm	Ritarda la cancellazione degli articoli, accumulandoli in un file temporaneo.
nostat	Generalmente 'news.daily' genera una serie di informazioni dettagliate sullo stato del sistema delle news. Con questa opzione si evita tale elaborazione.
notdaily	Se si vuole avviare 'news.daily' al di fuori della sua cadenza giornaliera normale, conviene farlo con questa opzione, in modo che le operazioni di rotazione e archiviazione dei file delle registrazioni non siano svolte (assieme ad altre operazioni simili legate alla temporizzazione normale del suo utilizzo).
noexpire	Generalmente 'news.daily' utilizza il programma 'expire' per eliminare gli articoli troppo vecchi. Con questa opzione gli articoli non vengono rimossi.

Opzione	Descrizione
norotate	In condizioni normali 'news.daily' archivia ed esegue la rotazione dei file delle registrazioni. Con questa opzione non tocca i file delle registrazioni.

Feed in ingresso utilizzando il protocollo NNTP

Il feed degli articoli può avvenire in diversi modi, sia dal punto di vista del protocollo utilizzato, sia per il modo in cui viene temporizzato. In generale, attraverso Internet (o le intranet) si usa prevalentemente il protocollo NNTP. INN controlla il feed in ingresso attraverso il file `‘/etc/news/incoming.conf’`, oppure, se si tratta di una versione più vecchia, `‘/etc/news/hosts.nntp’`.

File «/etc/news/hosts.nntp»

Il file di configurazione `‘/etc/news/hosts.nntp’` riguarda le versioni di INN più vecchie. Serve a definire quali siano i nodi remoti che possono diffondere gli articoli verso il sistema di news locale.

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all'interno della pagina di manuale *hosts.nntp(5)*. La presenza di questa pagina di manuale lascia intendere che sia ancora necessario l'utilizzo di questo file. Le righe vuote, quelle bianche e quelle che iniziano con il simbolo `‘#’` vengono ignorate. Le direttive sono composte da record secondo la sintassi seguente:

```
nodo : [ [parola_d'ordine] : [elenco_modelli_di_gruppi] ]
```

Considerato che si tratta di un file superato, non vale la pena di descriverne i dettagli. Basti sapere che per consentire la connessione è sufficiente indicare il nome del nodo seguito da due punti.

```
weizen.mehl.dg:
```

L'esempio mostra il caso in cui ci si attenda di avere il feed esclusivamente dal nodo *weizen.mehl.dg*. Eventualmente, ammesso che possa servire a qualcosa, si può aggiungere anche il nome del nodo locale:

```
localhost:  
dinkel.brot.dg:  
weizen.mehl.dg:
```

File «/etc/news/incoming.conf»

«

Il file di configurazione `/etc/news/incoming.conf` riguarda le versioni di INN più recenti. Serve a definire i nodi remoti che possono diffondere gli articoli verso il sistema di news locale, oltre che stabilire il numero massimo di connessioni che possono instaurarsi simultaneamente.

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all'interno della pagina di manuale *incoming.conf(5)*; la presenza di questa pagina di manuale fa intendere che sia necessario l'utilizzo del file di configurazione relativo. Le righe vuote, quelle bianche e quelle che iniziano con il simbolo `#` vengono ignorate. Le direttive possono essere di vario tipo, ma soprattutto possono essere suddivise in sezioni `peer` e `group`. Piuttosto di analizzare in dettaglio la sintassi di questo file, viene mostrato un esempio che dovrebbe essere sufficiente per iniziare.

```
# Definisce in modo globale il numero massimo di connessioni: 10
max-connections: 10

# Definisce l'accesso da parte dell'elaboratore weizen.mehl.dg
peer weizen {
    hostname:    weizen.mehl.dg
}
}
```

Nell'esempio appena mostrato sono state definite solo due cose: il numero massimo di connessioni in generale, fissando il valore a 10, e il fatto che *weizen.mehl.dg* può inviarcì il suo feed di articoli.

Feed continuo in uscita utilizzando il protocollo NNTP

Il feed in uscita rappresenta il flusso di articoli che viene diffuso presso i nodi corrispondenti. Questo può avvenire fondamentalmente in modo continuo, attraverso **'innfeed'**, o in modo differito a cadenza regolare, attraverso **'nntpsend'**. **'innfeed'** viene avviato normalmente da **'innd'** in base alla configurazione del file `'/etc/newsfeeds'`.

Nelle prossime sezioni viene descritto cosa fare per utilizzare **'innfeed'** nelle connessioni continue, ovvero di tipo a flusso (*stream*).

File `«/etc/news/innfeed.conf»`

Dovendo utilizzare **'innfeed'** per la diffusione degli articoli, è necessario predisporre il file `'/etc/news/innfeed.conf'`. Questo dovrebbe essere già stato predisposto abbastanza bene da chi ha preparato il pacchetto INN da installare.

La sintassi e le direttive che possono essere utilizzate in questo file sono descritte all'interno della pagina di manuale *innfeed.conf(5)*;

come al solito, le righe vuote, quelle bianche e quelle che iniziano con il simbolo '#', vengono ignorate.

Se tutto va bene, si dovrebbe porre attenzione solo alla dichiarazione dei nodi a cui inviare gli articoli per la loro diffusione. Per esempio, la direttiva

```
peer schwarz {  
    hostname:  schwarz.mehl.dg  
}
```

identifica il nodo *schwarz.mehl.dg* e gli attribuisce il nomignolo '**schwarz**' che serve poi per definire la duplicazione degli articoli per questo scopo nel file `/etc/news/newsfeeds`'.

Il tipo di connessione che si intende mostrare qui è di tipo a flusso continuo (*stream*), di conseguenza, prima della dichiarazione dei nodi dovrebbe apparire la direttiva seguente:

```
streaming:  true
```

Eventualmente si può essere sicuri ripetendola nella dichiarazione del nodo:

```
peer schwarz {  
    hostname:  schwarz.mehl.dg  
    streaming: true  
}
```

File «`/etc/news/newsfeeds`»

«

Come già accennato, per fare in modo che '**innfeed**' venga avviato da '**inn**' nel modo corretto, occorre predisporre opportunamente il file `/etc/news/newsfeeds`'. In precedenza è stato mostrato solo come attivare la diffusione locale degli articoli, per mezzo della voce standard '**ME**'; adesso occorre indicare che è necessario diffondere gli articoli attraverso '**innfeed**':

```
# Innfeed funnel master; individual peers feed into the funnel.
# Note that innfeed with "-y" and no peer in innfeed.conf
# would cause a problem that innfeed drops the first article.
innfeed!\
    !*\
    :Tc,Wnm*/usr/lib/news/bin/startinnfeed
```

Di solito, la direttiva che si vede nell'esempio è già contenuta nel file standard che viene installato con INN; eventualmente si tratta solo di togliere i commenti che ne impediscono l'attivazione.

Senza entrare troppo nel dettaglio (che comunque può essere approfondito con la lettura di *newsfeeds(5)*), si può affermare che viene creato un feed attraverso un condotto. Questo, come si legge dal commento originale, viene definito «imbuto» (*funnel*). Osservando bene, si vede che nel secondo elemento è indicato il modello `!*`, cosa che impedisce la corrispondenza con qualunque articolo; infatti occorre indicare espressamente quali nodi alimentare in questo modo attraverso altre direttive successive.

```
# A real-time feed through innfeed.
schwarz\
    :!junk,!control,!test,!local*\
    :Tm:innfeed!
```

Come si vede dall'esempio, viene creato un feed verso il nodo indicato nel file `/etc/news/innfeed.conf` con il nomignolo di `schwarz`, per tutti i gruppi che non siano `junk`, `control`, `test` e nemmeno che inizino per `local`. Questo flusso viene incanalato verso `innfeed` attraverso la direttiva denominata `innfeed!` (quella di prima).

Evidentemente, dovendo fare il feed nello stesso modo verso altri nodi, basterebbe aggiungere altre direttive di questo tipo che si rivolgono sempre alla voce `innfeed!`.

Per riepilogare un po', viene mostrato un esempio complessivo che comprende anche una dichiarazione ipotetica della diffusione locale.

```
ME:*,!junk,!control*,!local*::  
  
innfeed!:*:Tc,Wnm*:/usr/lib/news/bin/startinnfeed  
  
schwarz:!junk,!control,!test,!local*\  
      :Tm:innfeed!
```

Feed periodico in uscita utilizzando il protocollo NNTP

«

Per fare il feed periodico in uscita attraverso il protocollo NNTP, si utilizza il programma **'innxmit'**, di solito attraverso lo script **'nntpsend'**. Per ottenere tale risultato è opportuno predisporre il file `"/etc/news/nntpsend.ctl"` con l'elenco dei nodi che si vogliono servire in questo modo, quindi è necessario predisporre nel file `"/etc/news/newsfeeds"` la dichiarazione di questa forma di feed per ognuno di questi nodi.

File «/etc/news/nntpsend.ctl»

«

Il file `"/etc/news/nntpsend.ctl"` contiene la configurazione per lo script **'nntpsend'** e serve a elencare i nodi ai quali si vuole inviare il feed a intervalli regolari, attraverso il programma **'innxmit'**.

Le direttive di questo file sono dei record e la sintassi relativa può essere approfondita leggendo la pagina di manuale *nntpsend.ctl(5)*.


```

## Control file for nntpsend.
## Format:
##   site:fqdn:max_size:[<args...>]
##   <site>           The name used in the newsfeeds file for this site;
##                   this determines the name of the batchfile, etc.
##   <fqdn>           The fully-qualified domain name of the site,
##                   passed as the parameter to innxmit.
##   <size>           Size to truncate batchfile if it gets too big;
##                   see shrinkfile(1).
##   <args>           Other args to pass to innxmit
## Everything after the pound sign is ignored.

heiden:heiden.mehl.dg::

```

L'esempio, che riporta anche i commenti originali del file, mostra un record con il quale si vuole definire il feed verso il nodo *heiden.mehl.dg*, identificato ai fini della configurazione con il nomignolo **'heiden'**.

File «/etc/news/newsfeeds»

È necessario intervenire anche nel file «/etc/news/newsfeeds» per convogliare una copia degli articoli verso ogni nodo per il quale si utilizza questa forma differita di diffusione. L'esempio seguente dichiara il feed verso un file che poi viene letto da **'innxmit'** per l'invio verso il nodo *heiden.mehl.dg*, identificato nel file di configurazione «/etc/news/nntpsend.ctl» con il nomignolo **'heiden'**.

```

# Feed all local non-internal postings to nearnet; sent off-line via
# nntpsend or send-nntp.
amico-mio\
    :!junk,!control,!test,!local*\
    :Tf,Wnm:heiden

```

Si osservi che nel primo elemento del record è stato usato un nome di fantasia per identificare la voce, mentre l'ultimo fa riferimento al nomignolo fissato nel file «/etc/news/nntpsend.ctl».

Per essere precisi, in questo caso viene creato un file nella directory `/var/spool/news/out.going/` con i riferimenti agli articoli da usare per il feed, che poi viene letto da `nntpsend` quando è il momento di fare il trasferimento.

Utilizzo di «nntpsend»

«

Lo script `nntpsend` è il mezzo più comodo per comandare il programma `innxmit` allo scopo di fare il feed per mezzo del protocollo NNTP. Se viene utilizzato senza argomenti, `nntpsend` legge il file di configurazione `/etc/news/nntpsend.ctl` e diffonde gli articoli verso i nodi che vi trova elencati, in base al contenuto dei file relativi accumulati in precedenza in base alla configurazione di `/etc/news/newsfeeds`.

Questo, come tutto ciò che riguarda INN, deve essere avviato con i privilegi dell'utente `news`:

```
news$ /usr/lib/news/bin/nntpsend [Invio]
```

Se si utilizza questa forma di diffusione degli articoli, conviene predisporre il sistema Cron al riguardo, eventualmente attraverso uno script simile a quello seguente:

```
#!/bin/sh
su - news -c /usr/lib/news/bin/nntpsend
```

Ritrasmissione di articoli attraverso la posta elettronica

«

Una forma alternativa di feed è la trasmissione di una copia degli articoli verso un indirizzo di posta elettronica. Si ottiene questo semplicemente inserendo le direttive necessarie nel file `/etc/`

news/newsfeeds'. Per la precisione, si deve definire un imbuto, per esempio la direttiva seguente:

```
# Imbuto per l'invio attraverso la posta elettronica.  
mailer!!*:Tp,W*/bin/mail -s "Articoli da Usenet" *
```

Come si vede, viene utilizzato '/bin/mail' a cui viene aggiunta l'indicazione dell'oggetto («Articoli di Usenet») e l'indirizzo è rappresentato dall'asterisco finale. Per ottenere effettivamente l'invio dei messaggi occorre indicare altre direttive, una per ogni indirizzo, che utilizzano l'imbuto appena creato.

```
# Spedisce i gruppi comp.os.linux e alt.comp.os.linux a  
# tizio@dinkel.brot.dg  
tizio@dinkel.brot.dg:!*,comp.os.linux,alt.comp.os.linux:Tm:mailer!  
  
# Spedisce il gruppo it.cultura.linguistica.italiano a  
# caio@dinkel.brot.dg  
caio@dinkel.brot.dg:!*,it.cultura.linguistica.italiano:Tm:mailer!
```

Si osservi in particolare che nel secondo elemento di questi record viene indicato inizialmente di escludere tutti i gruppi, con il modello '!*', quindi di includere ciò che si desidera. Se non si facesse così, si otterrebbe l'invio degli articoli di tutti i gruppi.

Prelievo di articoli utilizzando il protocollo NNTP

Il prelievo di articoli non dovrebbe essere una tecnica usuale per ottenere il feed da un sito remoto, però potrebbe essere utile quando l'accesso a Internet è fatto attraverso una linea commutata: nel momento in cui si apre questa linea, oltre che inviare gli articoli prodotti nella rete locale, si vogliono ricevere quelli nuovi provenienti dall'esterno. Questo prelievo si può ottenere attraverso il programma 'nntpget'.

Utilizzo di «nntpget»

«

Il programma ‘**nntpget**’ non dispone di un file di configurazione ed è fatto per essere gestito comodamente attraverso degli script esterni, che però probabilmente sono mancanti. Come si vede dalla sintassi, a parte le opzioni che in pratica sono necessarie, è indispensabile indicare il nodo dal quale prelevare gli articoli aggiornati.

```
nntpget [opzioni] nodo
```

Il programma ‘**nntpget**’ va visto probabilmente solo come compendio al sistema locale di gestione delle news; in tal senso è praticamente necessario che sia in funzione il demone ‘**innd**’, in modo che ‘**nntpget**’ possa sapere quali articoli caricare e quali no. Si osservi l’esempio seguente:

```
news$ /usr/lib/news/bin/nntpget -o -v -t '990324 000000' ←  
↪roggen.brot.dg [Invio]
```

L’opzione ‘**-o**’ richiede espressamente la comunicazione con il demone ‘**innd**’ per conoscere quali articoli vale la pena di caricare dal sito remoto; l’opzione ‘**-v**’ fa in modo di avere qualche informazione in più; l’opzione ‘**-t '990324 000000'**’ fa in modo che vengano cercati solo gli articoli più recenti rispetto all’ora zero del 24/03/1999; l’ultimo argomento indica di contattare il nodo *roggen.brot.dg*.

In questa situazione, l’indicazione di una data di riferimento attraverso l’opzione ‘**-t**’ è obbligatoria e il formato è stabilito dal server:

In pratica: anno, mese, giorno, spazio, ore, minuti, secondi.

Consultando la pagina di manuale di **'nntpget'** si può leggere in che modo sostituire l'opzione **'-t'** con **'-f'**, allo scopo di usare un file al posto della data, sfruttando la sua data di modifica come riferimento per il prelievo degli articoli.

Utilizzando **'nntpget'** in questo modo, è necessario che il server che viene contattato consenta l'uso del comando **'NEWNEWS'**, che forse deve essere abilitato espressamente. Con le versioni recenti di INN occorre la direttiva **'allownews true'** nel file **'/etc/news/inn.conf'**.

Replicazione dei gruppi di un altro sito

Fino a questo punto si è visto che per creare un gruppo si può utilizzare il comando **'ctlinnd newgroup nome'**. In alternativa si può intervenire direttamente nel file **'/var/lib/news/active'**, ma poi c'è il problema di creare fisicamente le directory che devono ospitare gli articoli. Per preparare rapidamente un sito Usenet, può essere conveniente il prelievo di una copia di questo file da uno dei siti corrispondenti attraverso il programma **'actsync'**.

'actsync' viene configurato attraverso il file **'/etc/news/actsync.cfg'** e si avvale generalmente di **'/etc/news/actsync.ign'** per stabilire quali sono i gruppi da ignorare e quali da tenere. Per conoscere i dettagli sul funzionamento di **'actsync'**

e sul modo di configurarlo attraverso i file citati, occorre leggere la pagina di manuale *actsync(8)*. A titolo informativo sulle possibilità di ‘**actsync**’ vengono mostrati un paio di esempi.

```
news$ /usr/lib/news/bin/actsync -o a news.brot.dg [Invio]
```

Il comando mostrato sopra, permette di accedere al servizio NNTP di *news.brot.dg*, emettendo attraverso lo standard output un risultato simile a quello seguente, che in pratica riproduce un file ‘active’, ottenuto togliendo i gruppi da escludere in base alla configurazione di ‘**actsync**’.

```
comp.os.linux 0000000002 0000000123 y
alt.comp.os.linux 0000000145 0000000345 y
...
```

Il comando seguente, invece di mostrare il contenuto del file ‘active’, serve ad aggiornare i gruppi locali in base all’esito ottenuto. In pratica ‘**actsync**’ si avvale di ‘**ctlinnd**’ per questo.

```
news$ /usr/lib/news/bin/actsync -p 0 -o x -z 0 news.brot.dg
[Invio]
```

Riferimenti

«

- Olaf Kirch, *NAG, The Linux Network Administrators’ Guide*
- Rich Salz, James Brister, *Installing InterNet News*
‘/usr/share/doc/inn/Install.ms’
- Ian Jackson, Miquel van Smoorenburg, *Configuring Debian GNU/Linux’s INN package*
‘/usr/share/doc/inn/...’