

Lorenzo Cioni

Mailing List e File Server

(note tecniche)

Lo scopo di queste note è quello di fornire un minimo di documentazione su due servizi attualmente definiti in via sperimentale sul Microvax del Laboratorio (Sistema Operativo VMS 5.2) e in attesa di essere migrati sulla Alphastation 200 4/166 con Sistema Operativo Open VMS 7.0. Tali servizi sono un servizio di mailing list e un servizio di file server.

Entrambi i servizi si appoggiano sul pacchetto **mx**. **mx** oltre ad essere un mail server tradizionale che consente agli utenti definiti sul Microvax di ricevere e spedire posta, interagendo con il tradizionale mailer VMS, contiene un programma detto Mailing List/File Server (**mlf**) che fornisce i servizi necessari per la distribuzione di messaggi a liste dette mailing list e la gestione di tali liste attraverso comandi inviati per posta elettronica da utenti abilitati (owner e system user) ed inoltre permette la distribuzione di file attraverso il servizio di posta elettronica, ossia fornisce un servizio di file server.

Mailing list

Il termine mailing list viene di solito usato per descrivere un indirizzo di e-mail che distribuisce messaggi ad uno o più sottoscrittori (o subscriber).

Attualmente non esiste uno standard riconosciuto per l'implementazione delle mailing list e pertanto le modalità di utilizzo variano in funzione del sistema su cui la mailing list è installata e configurata. Si possono tuttavia individuare mailing list di due tipi: internet e bitnet.

Nel caso di mailing list stile internet si hanno di solito due indirizzi: uno per la lista vera e propria ed uno per la gestione di richieste di tipo amministrativo, quali le richieste di sottoscrizione, cancellazione e simili.

L'indirizzo di tipo amministrativo coincide di solito con il nome della lista cui si aggiunge il suffisso "-request" per cui se la lista si chiama QLL-Lablin, il suo indirizzo è QLL-Lab@mvxpi4.sns.it mentre l'indirizzo amministrativo è QLL-Lab-request@mvxpi4.sns.it.

Nel caso di mailing list stile bitnet le liste sono di solito implementate usando il package **listserv** progettato ad hoc per la gestione automatica delle mailing list.

Un elemento listserv su un dato host è caratterizzato dall'indirizzo listserv@nome_host gestisce le mailing list presenti sul sistema e mette a disposizione una gestione automatica delle richieste di tipo amministrativo.

mlf fornisce un supporto sia per l'interfaccia di tipo internet sia per quella di tipo bitnet per la gestione automatica dei comandi, sebbene l'uso dell'elemento listserv richieda la necessità di definire un alias tramite **mcp** (Mail Control Program). La descrizione delle mailing list stile bitnet esula dagli scopi delle presenti note.

File server

Anche per i file server non si hanno standard universalmente accettati e varie sono le implementazioni dei file server attualmente in uso.

Alcuni dei file server accettano comandi via messaggi bitnet alcuni solo via e-mail, alcuni accettano comandi sulla linea subject mentre altri li accettano nel body dei messaggi.

Oltre a ciò variano da server a server anche le modalità con cui i file sono distribuiti a coloro che ne fanno richiesta.

Nel caso dell'**mlf**, l'interfaccia dei comandi per il file server accetta comandi via e-mail e distribuisce i file via e-mail. **mx** che è di nuovo il programma cui il servizio si appoggia consente l'uso di un nome qualunque per il file server sebbene di solito si usi FileServ.

Definizione e utilizzo di una mailing list

Per la definizione delle mailing list e dei file server si hanno essenzialmente due modalità. La prima prevede l'utilizzo di una procedura di comandi fornita da **mlf** mentre l'altra prevede la creazione di mailing list e file server mediante l'uso di comandi espliciti di **mcp**.

La procedura di comandi (**mlf_config.com**) è di tipo interattivo e produce un file di comandi **mcp** che, se eseguito all'interno di **mcp**, permette la definizione di mailing list e file server.

I passi da seguire sono i seguenti:

- esecuzione di **mlf_config.com** con produzione di **mlf_config.mcp**;
- esame del file **mlf_config.mcp**;
- esecuzione del file **mlf_config.mcp** nell'ambiente **mcp** come **@mlf_config.mcp**;
- esecuzione del comando **save** per rendere permanenti le modifiche e
- esecuzione del reset dei processi di routing e di **mlf** in modo che questi riconoscano i nuovi elementi ossia si esegue il comando **reset/cluster router, mlf**.

Gli ultimi due comandi vanno usati anche se viene eseguita una creazione con comandi espliciti di **mcp**.

L'altra modalità come detto prevede l'uso di comandi espliciti. Si esaminano brevemente tali comandi insieme a quelli utilizzabili dagli utenti per accedere alla mailing list ed inoltre nel seguito si dà un semplice esempio di creazione di una mailing list associata ad un file server per la distribuzione dei messaggi archiviati su base mensile.

Il comando di **mcp** per la definizione di una mailing list è **define list** e prevede che vengano specificati almeno il nome della mailing list, il proprietario e i codici di protezione per classi (vedi oltre).

Il sistema di gestione della mailing list supporta l'archiviazione automatica dei messaggi, l'elaborazione del subscribing automatico e consente un limitato controllo remoto della mailing list. Tale sistema consente il settaggio di controlli di protezione della mailing list in modo da controllare le possibilità di subscribing e di invio di messaggi alla lista. Ogni mailing list ha quattro indirizzi locali:

- uno per la lista stessa (nome_lista@host);
- uno di tipo amministrativo (nome_lista-request@host);
- uno per il proprietario (owner) della lista e
- uno per il digest, se la lista supporta tale feature.

Il sistema di gestione della mailing list accetta richieste di iscrizione ed altri messaggi di controllo sull'indirizzo di tipo amministrativo o request address mentre la lista dei subscriber è mantenuta dall'agente **mlf** in un file ad hoc che è accessibile, ad esempio, mediante il comando **review**.

Archivi.

Una mailing list viene archiviata automaticamente dal sistema di gestione se, al momento della sua creazione, è stato usato il qualificatore **/archive** insieme a un nome di device e ad una directory esistente destinata a contenere l'archivio.

Il nome per il file archivio coincide, di default, con quello della mailing list e il tipo di default è yyyy-mmm ossia anno-mese. Se il default è mantenuto ogni mese viene automaticamente creato un nuovo file archivio.

Codici di protezione

Data una mailing list ad essa sono associate sia classi di protezione sia codici di protezione. Le classi sono le seguenti:

- system qualunque indirizzo che coincide con uno degli indirizzi definiti con il comando di **mcp define system_users indirizzo [, ...]**. Ogni utente il cui indirizzo è nella lista è membro della classe system;
- owner è l'indirizzo dell'owner della lista;
- group qualunque indirizzo contenuto nella subscriber list e
- world ogni altro indirizzo.

I codici di protezione sono:

- R, read consente ai membri della classe corrispondente l'uso del comando **review**;
- W, write consente ai membri della classe corrispondente l'invio di messaggi;
- E, enroll consente ai membri della classe corrispondente l'uso del comando **subscribe** e
- D, delete consente ai membri della classe corrispondente l'uso del comando

signoff.

Il codice **enroll** è significativo solo per la classe world mentre **delete** per la classe group. Le classi system e owner dovrebbero avere accessi di tipo RWED ed hanno anche implicitamente la possibilità di aggiungere e rimuovere, con comandi espliciti di **add** e **remove**, altri utenti dalla mailing list. Tali comandi devono essere inviati, come del resto gli altri, nel corpo di messaggi di e-mail spediti al request address.

Ad esempio si possono avere liste con i seguenti assegnamenti di codici di protezione alle classi:

G:RWED

W:RWE

chiunque può iscriversi e cancellarsi, vedere la lista e inviare messaggi, la lista è pubblica

G:RWED

W:E

la lista è semipubblica dato che solo chi è iscritto può esaminarla e inviare messaggi.

Di default un utente può ottenere informazioni su una mailing list inviando un comando di **directory** al **mxserver** (mailing list stile bitnet) o all'indirizzo amministrativo (mailing list stile internet). Se si vuole evitare che ciò accada si può definire la mailing list usando il qualificatore **/private** al momento della sua creazione.

mlf risponde automaticamente alle richieste fatte sia all'indirizzo amministrativo sia all'interfaccia **mxserver**. Nel primo caso i comandi riconosciuti sono i seguenti (si hanno comandi analoghi nell'altro caso se non che devono contenere la specifica della mailing list cui si riferiscono):

add indirizzo	aggiunta esplicita di un utente alla mailing list;
help	causa l'invio di un file di help;
list	lista tutte le mailing list definite sull'host a patto che non siano private;
query	ritorna lo stato del richiedente nella mailing list;
quit	interrompe la sequenza di richieste;

il comando quit permette di evitare che l'agente che gestisce la mailing list elabori campi "non ortodossi" contenuti nei messaggi quali signature e simili;

remove indirizzo	cancella un subscriber dalla mailing list;
review	restituisce la lista dei subscriber;
set [no]mail	abilita/disabilita la ricezione dei messaggi dalla mailing list;
set [no]conceal	nasconde o no un subscriber agli altri;

tale comando abilita un utente ad essere elencato (noconceal) insieme agli altri nell'output di un comando **review**. Si noti che l'owner della mailing list vede tutti i subscriber indipendentemente da come essi settano tale valore mentre un generico subscriber ottiene l'elenco solo degli utenti che non hanno inviato una richiesta di set conceal.

set [no]repro	controlla se un subscriber riceve copia o no dei messaggi da lui inviati alla mailing list;
set [no]digest	controlla se un subscriber riceve tutta la posta o un sommario giornaliero di questa;
signoff	permette ad un subscriber di cancellarsi dalla mailing list;
subscribe	permette ad un utente di iscriversi alla mailing list.

Il comando **subscribe** viene gestito automaticamente solo se la classe world ha accesso di tipo E (enroll) alla mailing list altrimenti le richieste sono inviate all'owner della lista che provvede alla loro gestione manuale (con il comando **add**). Il comando **signoff** viene gestito automaticamente solo se la classe group ha accesso di tipo D (delete) alla mailing list altrimenti le richieste sono inviate all'owner della lista che provvede alla loro gestione manuale (con il comando **remove**). Il comando **review** viene gestito automaticamente solo se l'utente che lo invia ha accesso di tipo R (read) alla mailing list altrimenti la richiesta da

origine ad una risposta con un messaggio di errore. In caso di mailing list semipubbliche ciò significa che l'utente deve per prima cosa eseguire un subscribe e poi può richiedere un review della mailing list.

E' possibile controllare il testo del messaggio che viene inviato agli utenti quando inviano comandi quali **subscribe** o **signoff** oppure nel caso di un utente che invia una richiesta di iscrizione ad una lista che non ha il codice di protezione E (enroll) per la classe world. Il package a questo scopo fornisce file di testo ad hoc che possono essere customizzati in base alle esigenze delle singole mailing list. Tali file sono spediti all'utente aggiunto alla lista in risposta ai comandi **subscribe** e **add** (file `m1ist_add_message.txt`) oppure all'utente rimosso dalla lista in risposta ai comandi **signoff** e **remove** (file `m1ist_remove_message.txt`) oppure infine all'utente se la sua richiesta non può essere trattata automaticamente per cui deve essere girata all'owner della lista per una elaborazione esplicita (file `m1ist_forward_message.txt`). Tali file risiedono nella sottodirectory **mlf.mailing_lists** della root di **mx**.

Per eliminare la definizione di una mailing list dal database di configurazione di **mx** si ha un comando di **mcp** di **remove list** mentre il file che contiene la lista dei subscriber deve essere rimosso esplicitamente a mano.

Il proprietario (owner) di una mailing list ha la possibilità di aggiungere o rimuovere esplicitamente gli utenti dalla mailing list mediante i comandi **add** e **remove**.

Il primo, se inviato in un mail all'indirizzo amministrativo, ha la sintassi seguente:

```
add qualificatori "nome" <nome_utente@host.luogo.dominio>
```

in cui i qualificatori più significativi ammessi sono i seguenti:

/nonotify	l'utente aggiunto non riceve messaggi di notifica;
/nomail	l'utente aggiunto non riceve mail dalla mailing list, lo si usa per mailing list private o associate a file server (vedi);
/nocase	il sistema di gestione ignora la differenza minuscole/maiuscole nella parte nome_utente;
/conceal	l'utente è aggiunto ma non risulta compreso nell'output del comando review agli utenti della classe group;
/norepro	l'utente non riceve copia dei messaggi che invia alla mailing list;
/deny	permette di aggiungere un utente ad una lista chiusa (senza permesso di enroll alla classe world) ma senza che questi possa inviare messaggi alla mailing list o riceverli o cambiare il proprio stato o cancellarsi;
/access	permette di eseguire controlli sugli indirizzi di chi vuole accedere alla lista.

Ogni indirizzo che fa matching con una delle entry della lista che segue il qualificatore /access ha attribuiti automaticamente i privilegi della classe group per quella mailing list. Se ad una mailing list possono inviare messaggi solo i membri della lista allora un indirizzo per il controllo degli accessi può essere specificato per permettere a ogni utente di un certo sito di inviare messaggi senza che tutti i possibili utenti si debbano prima iscrivere per poi poter inviare messaggi. E' possibile inoltre associare un file server ad una mailing list per cui chiunque voglia accedere al file server deve prima iscriversi alla mailing list associata altrimenti l'accesso gli sarà negato. L'uso del qualificatore /access consente un accesso senza restrizioni al file server da parte di certi indirizzi senza che ogni possibile indirizzo sia inserito nella lista dei subscriber della mailing list. Ad esempio se si vuole che gli utenti del dominio sns.it accedano liberamente ad un file server senza doversi prima iscrivere alla mailing list associata è possibile usare il comando **mcp**

```
add/access/nocase <*@*.sns.it>
```

Tutti gli indirizzi che soddisfano il pattern specificato hanno accesso ma sono settati come /nomail e /norepro in modo che non possano accedere alla mailing list ma solo al file server cui la mailing list è associata.

Il comando duale è quello di **remove** la cui sintassi è ad esempio la seguente:

```
remove/nonotify/nocase indirizzo
```

i cui qualificatori hanno un significato a questo punto noto.

Definizione e utilizzo di un file server

Il comando di **mcp define file_server** è utilizzato per la definizione di un file server. Ogni file server, dopo essere stato definito, può servire automaticamente richieste per singoli file o per gruppi di file.

E' possibile, inoltre, stabilire al momento della definizione di un file server, quando e in che quantità i file richiesti vadano spediti in modo da impedire che il loro trasferimento intasi il sistema di posta elettronica.

Un file server può, inoltre, essere associato ad una mailing list in modo che solo i subscriber della mailing list possano accedere ai servizi del file server.

Package

Un file server è progettato per gestire gruppi di file detti package. Al momento della creazione di un package per prima cosa è necessario creare una directory il cui nome coincide con quello del package, poi vanno copiati in tale directory i file che compongono il package e che devono essere spediti quando il package viene richiesto e il cui nome deve coincidere con quello del package stesso. In aggiunta deve essere inserito nella directory che contiene il package o in quella di livello immediatamente superiore un file che descrive il package che verrà spedito a chiunque faccia richiesta dell'elenco dei package disponibili su un certo file server. Tale file si deve chiamare nome_package.description.

La struttura suddetta funziona al meglio se viene fatto uso di un programma come **vms_share** per creare ogni singolo package. Tale programma è usato per raccogliere i file, formattarli in modo da massimizzare la probabilità di un loro trasferimento corretto e permette di suddividerli in chunks facilmente spedibili via e-mail. Quando tutti i chunks vengono riassemblati sul lato ricevente formano una procedura DCL che consente di ricreare i file originari. L'unico neo è che tale procedura va bene solo fra host con Sistema Operativo VMS.

Vediamo un esempio.

1) creo una directory che conterrà il package:

```
$ create/dir disk:[fileserv.nome]
```

2) copio il package nella directory:

```
$ copy file*.* disk:[fileserv.nome]
```

3) creo il file nome.description nella directory disk:[fileserv];

4) uso il comando **mcp** per definire il file server:

```
mcp> define file_server/root=disk:[fileserv.]
```

File di help

Esiste un file **fileserv_help.txt** che contiene una descrizione dei comandi messi a disposizione dal servizio di file server. Tale file deve essere customizzato per contenere l'indirizzo scelto per il file server (di default è FileServ@host.sito.dominio) e ogni altra informazione relativa al file server. La copia modificata deve essere contenuta nella root del file server in modo da essere spedita agli utenti che la richiedono mediante un mail al file server che contiene il comando **help**.

Log delle transazioni

Per ogni messaggio ricevuto dal file server viene creato un file di log delle transazioni che contiene il risultato dei comandi contenuti in un messaggio. Quando tutti i comandi sono stati eseguiti il file viene inviato all'utente che ha inviati il messaggio cui il file di log si riferisce in modo che l'utente conosca lo stato dei file richiesti in relazione a quando gli saranno spediti e altre informazioni correlate.

E' possibile che il gestore del file server crei un file (**fileserv_transaction.txt**) che contiene informazioni visibili a tutti gli utenti che accedono al file server. Tale file deve essere contenuto nella root del file server e il suo contenuto viene inserito automaticamente all'inizio di ogni file di log che viene spedito. Tale file funziona come un utile header per informare gli utenti di periodi di indisponibilità previsti per il servizio e di modifiche nella disponibilità dei package.

Comandi di file server

Il file server è in grado di accettare nel corpo dei messaggi a lui indirizzati i seguenti comandi: **sendme**, **list** o **directory**, **help**, **quit** e **address**.

Il comando **sendme** accetta come argomento sia il nome di un package sia il nome di uno o più file in modo che il file server spedisca l'elemento corrispondente all'utente che ha fatto la richiesta. La spedizione può essere ritardata a ore "morte" se il servizio è stato definito con tale caratteristica.

Il comando **list** (o **directory**) elenca tutti i package disponibili oppure quelli che matciano con un pattern passato come argomento al comando.

Il comando **help** causa l'invio del file opportuno. Il comando **quit** fa sì che le linee restanti del mail siano ignorate e può essere utile qualora il messaggio contenga una signature e infine il comando **address** fornisce l'indirizzo a cui il file server deve spedire i file se l'utente richiedente non vuole che venga usato l'indirizzo che compare nel campo "Reply-to" oppure "From" se tali campi possono essere corrotti dal passaggio attraverso più gateway.

*Il comando **define file_server***

Il comando **define file_server** consente la definizione di un file server basato sul mailer **mx** in modo che sia possibile distribuire gruppi di file via e-mail. Il file server risponde a comandi posti uno per linea nel corpo dei messaggi inviati dagli utenti all'utente associato al file server.

La sintassi del comando è la seguente:

```
mcp> define qualificatori file_server nome
```

dove i qualificatori possibili sono i seguenti:

<code>/begin_send_period=hh:mm</code>	dice a che ora il file server può iniziare a spedire file se la loro dimensione supera quella specificata dal qualificatore <code>/[no]delay_threshold</code> , il default sono le 17:00;
<code>/[no]delay_threshold=size</code>	senza la parte opzionale dice quale è la massima dimensione dei file che è possibile spedire sempre, indipendentemente dal valore precedente e dal valore di <code>/end_sed_period</code> ; conviene settare un valore alto altrimenti si rischia che sia differito anche l'invio del file di help;
<code>/[no]description=testo</code>	individua un testo che dà una breve descrizione del file server, il default è <code>nodescription</code> ;
<code>/end_sed_period=hh:mm</code>	dice da che ora in poi il file server non può più spedire file la cui dimensione supera quella specificata dal qualificatore <code>/[no]delay_threshold</code> , di default sono le 9:00;
<code>/[no]host_limit=hostlim</code>	specifica il max numero di byte che possono essere spediti a un singolo host;
<code>/[no]mailing_list=listname</code>	individua la mailing list associata al file server;
<code>/manager=indirizzo</code> sono	indica quale è l'utente che gestisce il file server e al quale sono

"girati" tutti i messaggi di errore relativi ad accessi al file server;

/root=directory è la directory radice del file server;

/[no]server_limit=serverlim setta il massimo numero di bit che possono essere spediti al giorno dal server;

/[no]user_limit=serverlim setta il massimo numero di bit che possono essere spediti al giorno ad un singolo utente;

Un esempio di definizione di una mailing list con file server associato

Vediamo ora un esempio di creazione di una mailing list con associato un file server che contiene la posta inviata alla mailing list e archiviata su base mensile.

Per prima cosa si crea la directory per la mailing list e quindi con un comando di **mcp** si crea una mailing list semipubblica:

```
$ create/dir dua0:[archives.maillist]
$ mcp
mcp> define list "lablin" -
      /owner="cioni@mvxpi4.sns.it" -
      /protection=(S:RWED,O:RWED,G:RWED, W:E) -
      /archive=dua0:[archives.lablin] -
      /nocase
mcp> save
mcp> reset
```

Gli ultimi due comandi hanno, rispettivamente, la funzione di registrare nel data base di configurazione di **mx** la mailing list appena creata e di far ripartire gli agenti di gestione con la nuova configurazione come configurazione corrente.

Una volta che una mailing list è stata creata e registrata e i processi (o agenti) di gestione sono stati re-inizializzati le richieste di tipo amministrativo vanno spedite nel caso dell'esempio all'indirizzo lablin-request@mvxpi4.sns.it mentre i messaggi che si vuole siano distribuiti a tutti i subscriber vanno spediti all'indirizzo di lista e cioè lablin@mvxpi4.sns.it.

Dato il settaggio, la posta rispedita alla lista viene spedita all'owner della lista perché questi la gestisca a mano, prendendo i provvedimenti necessari in funzione del tipo di problema, fino ad arrivare alla rimozione degli utenti che causano il problema. L'archivio viene creato nella directory specificata sotto forma di file il cui nome coincide con il nome della lista e cioè in questo caso con lablin e il tipo ha il formato yyyy-mmm cioè anno-mese. A questo punto non resta che creare il file server associato alla mailing list per la distribuzione di tali file archivio dietro richieste di messaggi opportuni inviati al file server. Per far ciò uso il comando:

```
mcp> define file_server "lablinarch" -
      /manager="cioni@mvxpi4.sns.it" -
      /root=dua0:[archives.] -
      /mailing_list=lablin
```

Tale comando deve essere, come nel caso precedente, seguito dai comandi save e reset. Le richieste vanno inviate via e-mail all'indirizzo lablinarch@mvxpi4.sns.it. Il file server risponde a richieste di invio di parte o di tutta la posta archiviata mensilmente per la mailing list lablin. Il fatto che il file server sia associato ad una mailing list impedisce ad utenti non iscritti alla mailing list di ottenere i file archiviati. A questo punto perché l'installazione sia completa devono essere creati i file fileserv_help.txt e lablin.description che hanno il compito di descrivere sia il file server sia la mailing list.

Qualora si voglia accedere ad un file server associato ad una mailing list i messaggi vanno spediti all'indirizzo giusto, nel caso dell'esempio lablinarch@mvxpi4.sns.it e devono contenere i comandi opportuni.

Ad esempio **list** consente di ottenere l'elenco dei package disponibili mentre **list <nome>** limita il campo d'azione al package specificato: nel caso di "lablinarch" **list lablin** fornisce informazioni sulla **mailing list** lablin.

Il comando **sendme** fa sì che l'agente che gestisce il file server spedisca al richiedente la posta inviata dai subscriber alla lista e archiviata di default su base mensile.

Con **sendme lablin** si ottiene tutta la posta disponibile mentre se la richiesta è circoscritta a un mese ben preciso è possibile specificarlo nel comando tenendo presente la convenzione anno-mese per cui ad esempio **sendme lablin.1997-03** fa sì che la posta spedita sia quella relativa al mese di marzo dell'anno specificato. Il richiedente riceverà in risposta alla propria richiesta un messaggio che contiene indicazioni sull'esito dei comandi inviati al server e una indicazione su quando i file richiesti gli/le saranno spediti.

Cenni ad altri comandi di mcp

Oltre ai comandi visti, altri comandi di **mcp** utili per la gestione di file server e mailing list sono i seguenti:

reset	per informare i vari processi di modifiche della configurazione;
status	per conoscere lo stato dei processi di gestione;
show file	per listare i file server definiti;
show list	per listare le mailing list definite;
show system_users	per listare gli utenti della classe system;
save	per rendere permanenti le modifiche alla configurazione.

e altri per i quali si rimanda alla documentazione e all'help in linea dell'ambiente **mcp**.

Bibliografia

Cioni Lorenzo, *Note Tecniche*, Manuali del Laboratorio di Linguistica, Scuola Normale Superiore, Marzo 1997

Cioni Lorenzo, *MX, Mailing List & File Server* Manuali del Laboratorio di Linguistica, Scuola Normale Superiore, Aprile 1997

Matt Madison & Hunter Goatley, *Message Exchange, Installation and maintenance manuals*, MadGoat Software, 1995.