

## Laboratorio di Amministrazione di Sistemi L-A

# Prova pratica - 22 febbraio 2007

Nell'ambito di una rete locale, un pool di 5 server offre agli utenti (che possono collegarsi da un generico client) un servizio di backup. Le credenziali degli utenti sono distribuite centralmente, in modo che ogni utente sia riconosciuto in modo uniforme da qualsiasi client e da qualsiasi server sulla rete, inoltre si supponga che le chiavi pubbliche degli utenti siano già depositate sui server, per consentire loro l'accesso senza password via ssh una volta che abbiano effettuato il login su di un client (ovviamente ai fini di eventuali test durante l'esame si dovrà provvedere manualmente alle operazioni necessarie).

La prima volta che un utente intende usufruire del servizio, si deve determinare quale dei server ha più spazio libero a disposizione e memorizzare l'associazione tra utente e server in una directory, da dove verrà recuperata ogni volta che l'utente chiederà di nuovo il servizio.

Il nucleo della procedura di backup vera e propria è costituito da una coppia di script; uno, eseguito sul client, genera un archivio con i file da copiare e lo trasmette via ssh all'altro, eseguito sul server, che lo estrae in uno spazio predisposto.

A completamento delle funzionalità, si deve predisporre sul client uno strumento che consenta all'utente di ricercare file depositati sul server.

Si ipotizzi che i client siano tutti collocati sulla rete 172.16.0.0/16, i 5 server abbiano indirizzi da 172.17.0.11 a 172.17.0.15 sulla rete 172.17.0.0/16, ed il traffico sia mediato da un router/firewall che utilizza rispettivamente sulle due reti l'indirizzo 172.16.0.1 e l'indirizzo 172.17.0.1. Il router/firewall svolge anche il ruolo di server LDAP.

## Dettagli e file da consegnare:

[net.sh](#) - script di configurazione della rete per la macchina che svolge i ruoli di router, firewall e server ldap. Deve contenere i comandi per la configurazione delle interfacce, delle tabelle di routing, e del packet filter. Quest'ultimo componente deve essere programmato per consentire solamente il traffico strettamente necessario.

[user.schema](#) - schema ldap che definisca la classe strutturale *bserver*, contenente gli attributi adatti a memorizzare *Username* dell'utente e *ServerIP* address del server scelto per i suoi backup

[select.sh](#) - script da utilizzare sul client, che non necessita di parametri e restituisce su standard output una coppia di valori separati da uno spazio.

Il primo valore è sempre lo username dell'utente che ha invocato lo script, il secondo valore è l'IP address del server da utilizzare per i suoi backup, che viene individuato dallo script medesimo secondo la procedura qui descritta:

- se è già presente nella directory LDAP un'entry per l'utente, si restituisce l'IP in essa contenuto
- in caso contrario si determina, tra i server, quello che al momento dispone della maggior quantità di spazio disponibile nella propria partizione */backups*, si inserisce nella directory l'entry che lo associa all'utente, e si restituisce l'IP

[bclient.sh](#) - script da utilizzare sul client, che accetta come parametro/i uno o più nomi di file e directory di cui fare una copia di backup, e procede come segue:

- utilizza **select.sh** per determinare lo username dell'utente che l'ha lanciato ed il server da contattare

- genera un archivio *tar* contenente tutti i file e le directory specificati come parametri, e lo invia al server per mezzo di una connessione *ssh* in modo che giunga allo standard input dello script **bserver.sh**. *Nota valida per gli script bclient e bserver: si legga il comportamento dell'opzione "-f" di tar dalla man page. Saranno particolarmente apprezzate soluzioni che non facciano uso di file temporanei per l'archivio.*
- Se il comando di archiviazione e copia sul server fallisce, bclient deve rischedulare la propria esecuzione dopo un'ora, ritentando all'infinito.

**bserver.sh** - lo script eseguito sul server non necessita di parametri:

- determina lo username dell'utente che lo ha invocato, e crea (se non esiste già) una directory */backups/<username>*
- all'interno di tale directory crea una ulteriore directory con nome nel formato *YYYYMMDD-hhmm*, rappresentativo della data ed ora di creazione della copia di backup (Es. 20060703-0930 per le 9:30 del 3 luglio 2006)
- all'interno di ques'ultima directory estrae l'archivio *tar* che riceve dallo standard input

**search.sh** - questo script deve essere invocato sul client in uno dei due modi possibili specificati di seguito (terminando immediatamente se rileva parametri non corretti). Può utilizzare altri script tra quelli descritti in precedenza, se utile.

1. Se il primo parametro è la stringa "file", deve esistere un secondo parametro che verrà interpretato come un nome di file. In tal caso si dovranno cercare, sul server, tutti file con tale nome presenti all'interno della directory dei backup dell'utente che ha lanciato la ricerca, stampando su standard output il loro nome (completo di percorso relativo a tale directory).
2. Se il primo parametro non è la stringa "file", allora si deve verificare che esso sia nel formato *YYYYMMDD-hhmm* (controllando solo la validità sintattica senza scendere nel dettaglio della validità dei valori numerici), ed elencare su standard output tutti i file contenuti nella directory */backups/<username>/YYYYMMDD-hhmm* (se esiste), completi di percorso relativo a tale directory. Se la directory non esiste si lasci semplicemente che il messaggio d'errore di sistema venga visualizzato all'utente.