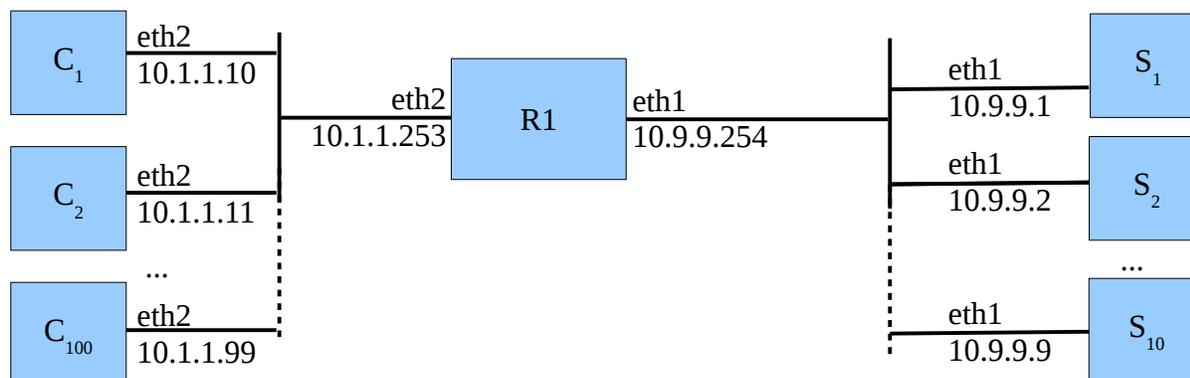


Laboratorio di Amministrazione di Sistemi T

Prova pratica - 27 giugno 2016

Descrizione generale del problema

Si consideri la rete illustrata in figura, in cui i blocchi C_i (per i compreso tra 10 e 99) rappresentano workstation disponibili all'utenza, da usare come client, e i blocchi S_j (per j compreso tra 1 e 9) rappresentano dispositivi che ospitano server LDAP



Di norma, i client sono raggruppati in 10 blocchi di 10 macchine con indirizzi consecutivi, e ogni blocco è servito da un server (esempio: i client 10.1.1.10 ... 10.1.1.19 sono serviti da 10.9.9.1). LDAP è utilizzato per memorizzare statistiche di utilizzo dei client, che caricano dati periodicamente.

I server LDAP in questo scenario hanno capacità di elaborazione limitata ed è fondamentale, per evitare malfunzionamenti, che i server sovraccarichi vengano esclusi temporaneamente dal pool, reindirizzando le richieste ad altri.

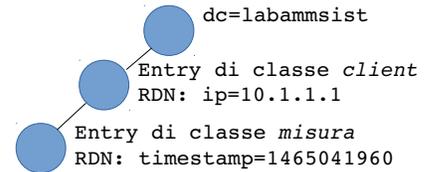
Il router si occupa dello smistamento delle richieste, in modo che i client non sappiano quale server stanno utilizzando, servendosi del packet filter: i client quindi contattano solamente l'indirizzo del router, che provvede a reindirizzare i pacchetti sul server assegnato.

Il router svolge anche il monitoraggio dei server via SNMP per individuare casi di sovraccarico e riconfigurare lo smistamento client-server in modo opportuno.

File da consegnare

(tra parentesi, a fianco del nome, la macchina su cui deve girare lo script)

stats.schema.ldif (router) - Definire gli attributi LDAP *ip* (testuale), *load*, *ram* e *timestamp* (interi), e la classe *client* contenga obbligatoriamente *ip* e la classe *misura* che contenga obbligatoriamente gli altri. Le entry collocate come nell'esempio rappresentano l'informazione che la macchina con indirizzo "*ip*" all'istante "*timestamp*" ha un carico medio nell'ultimo minuto pari a "*load*", e una quantità "*ram*" di memoria disponibile.



monitor.sh (client) - Questo script rileva via SNMP il carico medio nell'ultimo minuto e la RAM libera sulla macchina su cui viene lanciato, e memorizza i dati in una nuova entry sul server LDAP.

Indicare nei commenti:

- come eseguire automaticamente lo script ogni 5 minuti;
- come configurare gli agenti snmp dei client per consentire la rilevazione;

load.sh (server) - Questo script gira senza mai interrompersi e misura in locale ogni 5 secondi con `vmstat` l'uso di `cpu user-space ("us")` del server. Se rileva per 4 volte consecutive un valore superiore a 95, logga il messaggio "overload" + il proprio indirizzo, se rileva per 4 volte consecutive un valore inferiore a 50, logga il messaggio "ok" + il proprio indirizzo.

Indicare nei commenti come configurare il sistema di logging dei server e del router perché i messaggi vengano scritti sul file `/var/log/server_carichi` del router.

route.sh (router) - Questo script (utilizzabile da `init.sh` e `switch.sh`) accetta 3 parametri: il primo può essere o la stringa "off" o l'indirizzo di un server, gli altri due sono l'indice iniziale e finale di un range di client.

Se invocato con "off" (de)configura il packet filter per eliminare eventuali regole di indirizzamento delle richieste ricevute dai client verso un server; se invocato con un indirizzo di server, configura il packet filter per indirizzare tutte le richieste ricevute da un client appartenente al range al server specificato.

Esempi:

```
route.sh 10.9.9.7 21 30      fa sì che tutte le richieste LDAP provenienti da
                             10.1.1.21 ... 10.1.1.30 vengano inviate a 10.9.9.7
route.sh off 21 30          fa sì che tutte le richieste LDAP provenienti da
                             10.1.1.21 ... 10.1.1.30 non vengano inviate a nessun server
```

init.sh (router) - Questo script deve configurare il packet filter per

- consentire solo il traffico necessario ai vari script di questo testo;
- predisporre la situazione iniziale di inoltro del traffico LDAP tra i blocchi di client e i server (client da 10 a 19 → server 1, client da 20 a 29 → server 2, ecc.)

Fatto ciò, lo script lancia `switch.sh` e lo controlla, riavviandolo in caso di terminazione accidentale.

copy.sh (router) - Questo script (utilizzabile da `switch.sh`) accetta due parametri, entrambi indirizzi ip di server, e copia dal primo al secondo tutte le entry LDAP con timestamp maggiore di quello memorizzato in `/etc/last_timestamp`. Al termine aggiorna questo file inserendovi l'istante corrente.

switch.sh (router) - Questo script mantiene in memoria un elenco dei server utilizzabili (inizialmente tutti), e controlla continuamente il log `/var/log/server_carichi`.

Ogni volta che compare un nuovo messaggio "overload" relativo a un server (che indicheremo con SO)

- toglie SO dall'elenco dei server utilizzabili e sceglie un server a caso SN tra quelli rimasti
- aggiorna SN col contenuto di SO per mezzo di `copy.sh`;
- individua i client che stavano usando SO, e li ridirige verso SN usando `route.sh`