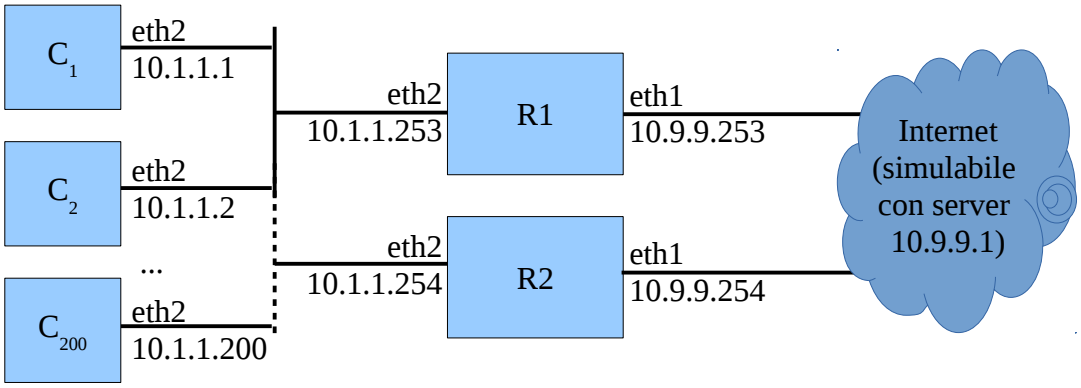
Laboratorio di Amministrazione di Sistemi T

Prova pratica – 6 giugno 2016

Descrizione generale del problema

Si consideri la rete illustrata in figura, in cui i blocchi C_i (per i compreso tra 1 e 200) rappresentano workstation disponibili all'utenza, da usare come client, collocate su di una rete privata connessa a Internet attraverso due router/firewall configurati per garantire alta disponibilità.



I router possono essere utilizzati indifferentemente; i client operano la scelta in base al numero di connessioni servite, basandosi su di un registro conservato in directory LDAP replicata sui due router, e mantengono sotto controllo il router scelto per commutare sull'altro in caso di guasto.

Si dia per ipotesi che i router non si guastino mai contemporaneamente.

File da consegnare

tutti gli script/file sono eseguiti/utilizzati su entrambi i router, salvo indicazione contraria

usage.schema.ldif – Definire i tipi di attributi LDAP *ipclient* e *iprouter* (testuali) e *timestamp* (intero), e la classe *gw* che li contenga obbligatoriamente tutti. Le entry di classe *gw* rappresentano l'informazione che l'host con *ipclient* ha scelto *iprouter* come default gateway all'istante *timestamp.*

gw.sh (in funzione sui client) – Questo script interroga una directory LDAP per determinare quale dei due router ha il minor numero di client che lo stanno utilizzando, e impostarlo poi come default gateway del client su cui viene lanciato. La query può essere fatta su R1 o su R2 indifferentemente; suggerimento: si noti che se il primo non risponde non ha senso impostarlo come gateway.
Successivamente lo script non termina, ma inizia a inviare un "ping" ogni secondo al gateway prescelto, e nel caso non riceva risposta per tre volte consecutive commuta il default gateway sull'altro router, proseguendo con lo stesso tipo di monitoraggio sul nuovo gateway.
Ogni volta che lo script effettua una scelta di default gateway, tenta di registrarla su entrambe le directory LDAP, assicurandosi che l'entry che riguarda il proprio client sia unica.

init.sh – Questo script, appena avviato su di un router deve

- configurare il packet filter locale per consentire solo il traffico necessario ai vari script di questo testo;
- sostituire l'intero contenuto della directory LDAP locale col contenuto della directory dell'altro router, solo dopo aver verificato via SNMP che lo stesso script (init.sh) non sia in esecuzione sull'altro router

Indicare nei commenti come configurare gli agenti SNMP dei router per consentire il controllo.

check.sh – Questo script rileva su quale router è in esecuzione, poi con una query alla directory LDAP locale determina quali client lo stanno utilizzando come default gateway, e controlla via SSH su ognuno di essi che effettivamente la configurazione del routing sia coerente. Nel predisporre il controllo, si tenga conto del numero elevato di client, e si garantisca che possano essere raccolte tutte le risposte nel giro di pochi secondi.
Per ogni client su cui è configurato un default gateway incoerente con quello memorizzato in LDAP, deve essere attivata (evitando duplicazioni) su entrambi i router una regola di iptables che permetta di loggare ogni pacchetto da e per tale client.
Indicare nei commenti:

- come eseguire automaticamente lo script ogni 5 minuti;
- come predisporre i sistemi perché i router possano eseguire comandi sui client;
- come configurare il sistema di logging perché ogni messaggio di iptables generato su ognuno dei router venga scritto sul file `/var/log/orphans.log` di entrambi i router.

reset.sh – Questo script esamina continuamente il file `/var/log/orphans.log`. Per ogni riga che legge, determina se l'IP client in essa contenuto è stato osservato più di 10 volte nei due minuti precedenti.
Se si verifica questa condizione

- si collega al client e termina tutti i processi che stanno utilizzando socket di rete
- rimuove su entrambi i router la relativa regola di logging inserita da `check.sh`