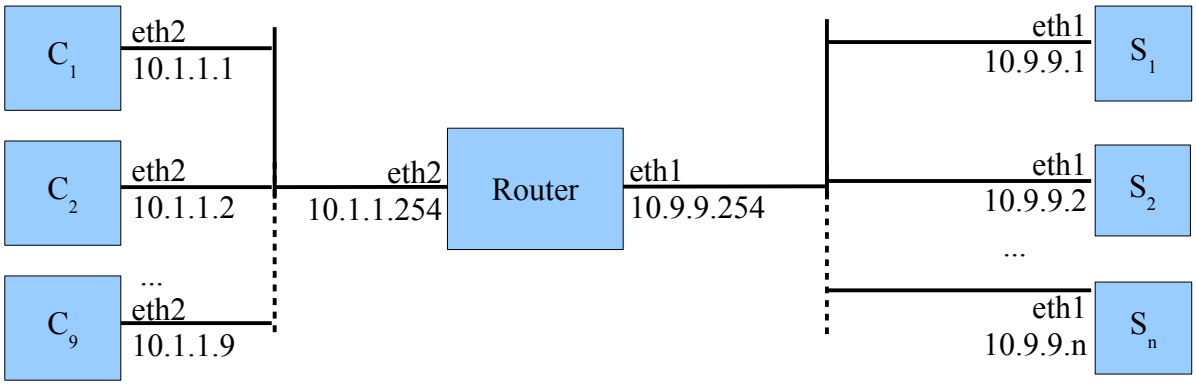
Laboratorio di Amministrazione di Sistemi T

Prova pratica - 19 dicembre 2013

Descrizione generale del problema

Si consideri la rete illustrata in figura, in cui i blocchi C_i (per i compreso tra 1 e 9) rappresentano workstation disponibili all'utenza, da usare come client, collocate su di una rete connessa attraverso un router ad una seconda rete, su cui sono attestati un pool di server.



Tutti i client possono essere utilizzati indifferentemente dagli utenti del sistema per far girare i propri processi, anche lasciandone attivi in background al logout (quindi possono esserci simultaneamente processi di più utenti su ognuno dei client).

Il router controlla periodicamente via SNMP lo stato dei processi attivi sui client, e imposta il proprio packet filter in modo da consentire loro di accedere ai server specificati dalle politiche di configurazione memorizzate su di una directory LDAP installata sul router stesso.

File da consegnare

routerinit.sh - Questo script esige un parametro che può valere start o stop. Al boot del sistema deve essere invocato col parametro start prima della configurazione della rete, per configurare il packet filter in modo da bloccare tutto il traffico non indispensabile; all'arresto del sistema deve essere invocato col parametro stop dopo la deconfigurazione della rete, per resettare il packet filter. Indicare nei commenti come automatizzare questo comportamento sfruttando le procedure standard del sistema.

discovery.sh - Questo script interroga via SNMP la macchina specificata come parametro sulla riga di comando, e restituisce i nomi degli utenti con uid ≥1000 che hanno processi attivi sulla macchina.

snmpd.conf – File di configurazione da installare sui client per consentire allo script *discovery.sh* di prelevare le informazioni richieste.

stats.schema - Definisce la classe `vincoli` che contiene gli attributi testuali `utente` e `server`; un'entry conserva nei valori dell'attributo `server` una potenziale lista di indirizzi che rappresentano i server che l'`utente` può utilizzare.

route.sh - Questo script gira sul router. A ciclo continuo, ripete l'interrogazione di tutti i client con *discovery.sh* e in base al risultato, al termine di ogni interrogazione, programma il packet filter secondo questi criteri:

- ad ogni utente deve essere consentito l'accesso ai server specificati nella corrispondente entry LDAP;
- i client non devono essere consapevoli della scelta del server: contattano sempre e solo l'indirizzo del router, che provvede alle necessarie operazioni per inoltrare le richieste ai server;

Indicare nei commenti come garantire che lo script sia permanentemente in esecuzione.

syslog.txt – Includere in questo file le direttive di configurazione dei demoni *rsyslogd* dei server e del router (solo le aggiunte rispetto al file di default, indicando chiaramente quali si applicano ai server e quali al router) che permettono di registrare sul file /var/log/serverstats.log del router tutti e soli i messaggi etichettati local6.notice prodotti sui server.

stats.sh – Questo script, lanciato sui server, esamina il traffico in transito per scoprire quali porte TCP vengono contattate dai client (senza tener conto di quanto traffico viene generato per ogni porta). Ogni volta che riceve il segnale USR1, invia le statistiche a syslog con etichetta local6.notice, sotto forma di singola riga che elenca tutte le porte viste separate da spazi, e le azzera.
Indicare nei commenti come garantire che lo script riceva automaticamente il segnale USR1 al minuto 10 di ogni ora.