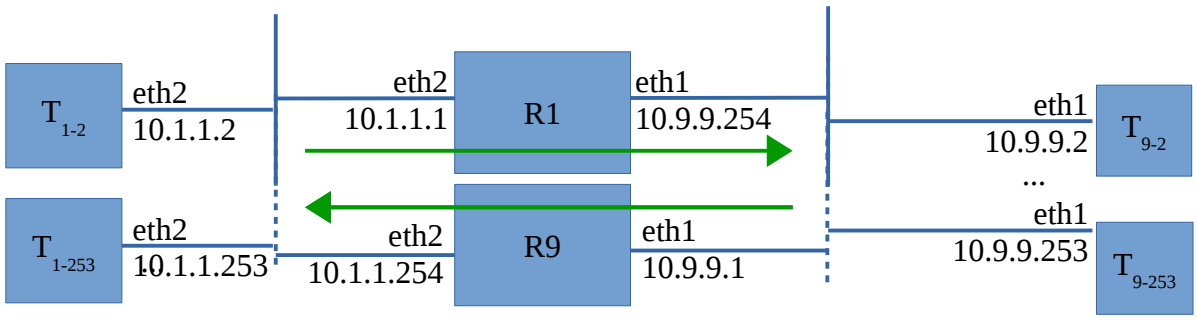
Laboratorio di Amministrazione di Sistemi T

Prova pratica del 13 febbraio 2017

Descrizione generale del problema



Un sistema di instradamento e controllo del traffico ad alta disponibilità è costituito da una coppia di macchine “controller” (R1 e R9) che in condizioni normali inoltrano unidirezionalmente il traffico da una rete “target” all’altra, bloccando il traffico anomalo (cioè quello che si discosta dai pattern leciti specificati da un file di configurazione).

Quando un certo flusso di traffico viene bloccato, l’azione viene memorizzata su di una directory LDAP ospitata dai router stessi, per poter essere ripristinata successivamente a un eventuale reset dei router.

In caso di guasto di uno dei due router, quello superstite può prenderne il posto.

File da consegnare
(tutti gli script e i file di configurazione sono collocati sui router,
a eccezione della generica descrizione da fornire nel primo file)

network.txt – Elencare i comandi di configurazione del networking (interfacce e routing) delle macchine nelle due reti locali, e i comandi di configurazione del packet filter di R1 e R9, che predispongono il sistema a funzionare come in figura, col router R1 che inoltra solamente i pacchetti dalla rete 1.1.1.0/24 alla rete 10.9.9.0/24 e il router R9 che inoltra solamente i pacchetti dalla rete 10.9.9.0/24 alla rete 1.1.1.0/24

stop.schema.ldif – Definire gli attributi LDAP *timestamp* (numerico), *source* e *destination* (testuali) e la classe *stop* che li preveda obbligatoriamente tutti. Le entry di questa classe conservano l'informazione che all'istante *timestamp* è stato deciso di bloccare il traffico tra le macchine *source* e *destination.*

insert.sh - Questo script accetta tre parametri (timestamp e due indirizzi ip) e inserisce una entry nella directory LDAP locale, costruita con tali dati.

watch.sh - Osserva senza interruzione tutto il traffico TCP tra le due reti, limitatamente ai primi 200 byte di ogni pacchetto. Quando un pacchetto contiene un pattern non compreso tra quelli elencati nel file `/etc/whitelist.regex`, invoca *insert.sh* passando come primo parametro l'ora corrente (con sufficiente precisione per evitare duplicati) e come secondo e terzo gli ip di sorgente e destinazione del pacchetto.
Le stesse tre informazioni devono essere inviate all'altro router per mezzo di syslog.
Indicare nei commenti: come configurare i demoni syslog dei router per consentire la scrittura sul file `/var/log/packets.log` di ognuno dei messaggi ricevuti dall'altro.

filter.sh - Quando questo script viene lanciato su di un router, aggiorna per mezzo di *insert.sh* la directory LDAP locale, prelevando dal file `/var/log/packets.log` le righe non ancora processate rispetto all'ultima esecuzione dello script stesso.
Successivamente, configura il packet filter locale perché blocchi il traffico relativo alle coppie di IP contenute nelle entry LDAP create nelle ultime due ore.
Indicare nei commenti: come far eseguire automaticamente lo script ogni 5 minuti.

failover.sh – Controlla via SNMP ogni 10 secondi se l'altro router ha il processo *watch.sh* in esecuzione. Se rileva che non è attivo, per prima cosa tenta di arrestare completamente l'altro router, poi ne assume il ruolo con le seguenti azioni

- configura le proprie interfacce per assumere anche gli indirizzi dell'altro router
- riconfigura il packet filter per consentire tutto il traffico tra le due reti, con l'eccezione delle coppie di macchine elencate nelle entry LDAP create nelle ultime due ore.

Indicare nei commenti:

- come configurare l'agent SNMP dei controller per consentire la verifica richiesta
- cosa bisogna predisporre per consentire ai router di spegnersi l'un l'altro