Laboratorio di Reti
Compito del 22/02/2024
Durata 2h

Slide del corso: http://public.aries.ing.man/LaboratorioReti/

Esercizio 1
(editor di testo)

Si configurino le seguenti regole iptables per un firewall su una macchina Linux dotata di due interfacce di rete (eth0 e eth1), per garantire sicurezza e controllo del traffico di rete in ingresso e in uscita.

Catena INPUT (eth0):
- Impostare la politica di default per non consentire traffico in ingresso;
- Accettare tutti i pacchetti provenienti dalla sottorete 172.16.0.0/16;
- Scartare tutti i pacchetti UDP che non sono destinati alla porta 53 (DNS);
- Accettare pacchetti relativi alle connessioni già stabilite.

Catena OUTPUT (eth1):
- Impostare la politica di default su ACCEPT;
- Bloccare tutto il traffico in uscita verso la sottorete 10.0.0.0/8 eccetto il traffico HTTP e HTTPS;
- Consentire il traffico in uscita per il servizio di DNS.

Catena FORWARD:
- Impostare la politica di default su DROP;
- Consentire il forwarding dei pacchetti tra le due interfacce solo se destinati alla porta TCP 80 o 443.

Esercizio 2
(editor di testo)

Si implementi una playbook Ansible per configurare un server di log su una distribuzione Linux di tipo Debian-based, identificato dal gruppo “log-servers”. La playbook deve includere l’installazione e la configurazione dei pacchetti rsyslog e logrotate.

In particolare, si vogliono realizzare i seguenti task:
- aggiornamento dell’indice dei repository;
- upgrade dei pacchetti;
- installazione dei pacchetti “rsyslog” e “logrotate”;

- copia dei file di configurazione personalizzati con Jinja2 “rsyslog.conf” e ”logrotate.conf” che utilizzano il file vars.yml nelle rispettive cartelle /etc/rsyslog e /etc/logrotate
- effettuare l’enable e lo start dei due servizi sopra installati utilizzando l’apposito modulo di Ansible;

Esercizio 3
(editor di testo + interprete python)

Si sviluppi un'applicazione Client/Server in Python utilizzando le socket TCP. L'applicazione deve consentire a un client di verificare la lista di tutti gli utenti (esclusi gli account associati a processi demoni) che hanno un account su una macchina server Linux. Il processo server deve mettersi in ascolto su una porta specificata dall'utente e gestire le richieste dei client in modo sequenziale.

Il client richiede all'utente di inserire l'hostname e la porta del server a cui collegarsi. Una volta stabilita la connessione, il client invia al server il comando "user_list" e si mette in attesa di ricevere la risposta. Il server riceve la richiesta e la elabora, consultando il file /etc/passwd per verificare gli utenti presenti sulla macchina. A tal fine, si supponga che gli utenti associati ai processi demoni abbiano in /etc/passwd la stringa "/sbin/nologin". Il server deve quindi selezionare solamente gli utenti che non contengono “/sbin/nologin” ed estrarre solamente l’informazione relativa al nome. Si consiglia di visionare il file “/etc/passwd” per capire come è organizzato.