

Laboratorio di Reti
A.A 2021/2022
Compito del 09/09/2022
Durata 2h30m

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

Esercizio 1

Si definiscano le seguenti regole iptables per la configurazione di un firewall di una macchina Linux. In particolare, si vogliono configurare alcune regole per la catena di INPUT e per quella di OUTPUT.

Catena di INPUT:

- impostare la politica di default per la chain di input per non accettare pacchetti;
- abilitare il traffico UDP in entrata sulla porta 5678;
- accettare il traffico in ingresso relativo a connessioni iniziate dalla macchina;

Catena di OUTPUT

- impostare la politica di default per la chain di output su accept;
- bloccare tutto il traffico indirizzato alla sottorete privata 192.168.5.0/24;
- scartare tutti i pacchetti relativi a connessioni TCP con porta di destinazione 32768;
- utilizzando la *tabella di NAT* si modifichi la porta di destinazione dei pacchetti TCP da 5000 a 5005;

Esercizio 2

Si supponga di dover decodificare una parte di messaggio contenuto in un protocollo applicativo per applicazioni IoT. La parte di header da decodificare ha dimensione 16 bit ed è codificata come big endian. I 16 bit comprendono i seguenti campi:

| Q/R (1 bit) | URG (1 bit) | CODE (2 bit) | SEQ_CODE (4 bit) | R_FORMAT (8 bit) |

Si scrivano le operazioni di maschera e bit shifting necessarie per reperire il valore delle diverse componenti. Inoltre, si supponga che dopo i 16 bit letti, se il bit Q/R è a 1, i successivi 16 bit presenti nel messaggio codificano un tipo di dato short (sempre codificato in big-endian). Si implementi la lettura del stringa dal buffer. (Si consulti l'helper del modulo short).

Esercizio 3

Si implementi una playbook Ansible, pensata per una distribuzione Linux di tipo Debian-based. La playbook deve installare e configurare il web server “nginx”, su un pool di macchine identificato dal nominativo “web-servers”. La playbook, deve installare il pacchetto di nginx, e apportare alcune modifiche alla configurazione di base. In particolare, la playbook Ansible deve sostituire il file di configurazione di default “/etc/nginx/nginx.conf” con il file “nginx.conf” presente sulla macchina controller e disabilitare il sito web di default, cancellando la directory “/etc/nginx/site-enabled/default”. Inoltre, la playbook deve creare una nuova directory alla path “/var/www/blog”, e associarla all’utente “nginx” e al gruppo “nginx”, con permission mask 0755.

Si realizzino i seguenti task:

- aggiornamento dell’indice dei repository;
- installazione di nginx;
- sostituire il file di configurazione “/etc/nginx/nginx.conf” con il file “nginx.conf” (presente sulla macchina controller);
- Disabilitare il sito web di default cancellando la directory /etc/nginx/site-enabled/default/;
- creazione della directory “/var/www/blog”;
- copia del contenuto della directory “blog” (presente sul controller Ansible) in “/var/www/blog”;
- enable e start di nginx;

Esercizio 4

Si implementi un’applicazione Client/Server in Python utilizzando le socket TCP che consenta a una macchina client di verificare se un particolare processo è in esecuzione su una macchina server. Il processo server deve mettersi in ascolto su una porta, che viene specificata in input dall’utente, e gestire le richieste dei client in modo sequenziale. Il client deve chiedere in input all’utente l’hostname e la porta del server a cui deve collegarsi.

Una volta connesso, il client invia al server una stringa contenente il nome del processo, ad esempio “firefox”. Al ricevimento del parametro di ricerca, il server esegue il comando corrispondente, ne legge l’output e verifica (utilizzando l’apposito operatore del linguaggio Python), se il processo è in esecuzione per inviare al client la risposta (SI / NO). Al termine della richiesta, il server deve chiudere la connessione con il client, mentre il client stampa a video l’output ricevuto dal server e termina.