

Design and Developement of a Cyber Range

Alexandru Mocanu

Università Degli Studi di Torino
Dipartimento di Informatica

February 17, 2020

Outline

- 1 Cyber Range
- 2 CTF - Capture the Flag
 - Tipologie CTF
- 3 Scenario immaginato
- 4 Tools & Software
- 5 Regole Firewall
 - Fase 1
 - Obiettivi
 - Regole specifiche per il Virtual Router
 - Regole specifiche per il Management
 - Regole specifiche per i Team
 - Fase 2
 - Obiettivi
 - Regole specifiche per il Virtual Router
 - Regole specifiche per il Management
 - Regole specifiche per i Team
- 6 Python Script

Cyber Range

È un ambiente virtuale usato da professionisti, e non, per testare:

- Affidabilità
- Sicurezza
- Prestazioni

di infrastrutture e sistemi IT.

CTF - Capture the Flag

È un gioco di **hacking** dove team (o singoli) cercano **vulnerabilità** in sistemi e software messi a disposizione dagli organizzatori della competizione al fine di sfruttarle e di collezionare le varie **flag** nascoste sul sistema bersaglio.

Tipologie CTF

Ci sono vari tipi di Capture the Flag, i più famosi sono:

- Jeopardy
- Attack/Defense
- Boot2Root

CTF - Jeopardy

I CTF di tipo Jeopardy sono probabilmente i più popolari, dato che si adattano bene a competizioni online.

In questo formato:

- diverse *challenge*, ognuna con un singolo *flag*

CTF - Jeopardy

I CTF di tipo Jeopardy sono probabilmente i più popolari, dato che si adattano bene a competizioni online.

In questo formato:

- diverse *challenge*, ognuna con un singolo *flag*
- **punteggio**: a seconda della difficoltà della singola *challenge*

CTF - Jeopardy

I CTF di tipo Jeopardy sono probabilmente i più popolari, dato che si adattano bene a competizioni online.

In questo formato:

- diverse *challenge*, ognuna con un singolo *flag*
- **punteggio**: a seconda della difficoltà della singola *challenge*
- **vincitore**: chi totalizza il punteggio maggiore

CTF - Jeopardy

I CTF di tipo Jeopardy sono probabilmente i più popolari, dato che si adattano bene a competizioni online.

In questo formato:

- diverse *challenge*, ognuna con un singolo *flag*
- **punteggio**: a seconda della difficoltà della singola *challenge*
- **vincitore**: chi totalizza il punteggio maggiore
- **durata**: a tempo

CTF - Jeopardy

I CTF di tipo Jeopardy sono probabilmente i più popolari, dato che si adattano bene a competizioni online.

In questo formato:

- diverse *challenge*, ognuna con un singolo *flag*
- **punteggio**: a seconda della difficoltà della singola *challenge*
- **vincitore**: chi totalizza il punteggio maggiore
- **durata**: a tempo

Generalmente con questo tipo di formato i partecipanti formano squadre, le quali assegnano una challenge a uno o più membri della squadra, per velocizzare la risoluzione delle challenge e la conquista delle flag.

CTF - Attack/Defense

Il formato Attack/Defense prevede che gli organizzatori forniscano ai giocatori (principalmente squadre) una o più macchine virtuali.

CTF - Attack/Defense

Il formato Attack/Defense prevede che gli organizzatori forniscano ai giocatori (principalmente squadre) una o più macchine virtuali.

I giocatori quindi avranno un doppio ruolo:

CTF - Attack/Defense

Il formato Attack/Defense prevede che gli organizzatori forniscano ai giocatori (principalmente squadre) una o più macchine virtuali.

I giocatori quindi avranno un doppio ruolo:

- **difendere** i servizi esposti sulla propria macchina virtuale

CTF - Attack/Defense

Il formato Attack/Defense prevede che gli organizzatori forniscano ai giocatori (principalmente squadre) una o più macchine virtuali.

I giocatori quindi avranno un doppio ruolo:

- **difendere** i servizi esposti sulla propria macchina virtuale
- **attaccare** i servizi dei team concorrenti

CTF - Attack/Defense

Il formato Attack/Defense prevede che gli organizzatori forniscano ai giocatori (principalmente squadre) una o più macchine virtuali.

I giocatori quindi avranno un doppio ruolo:

- **difendere** i servizi esposti sulla propria macchina virtuale
- **attaccare** i servizi dei team concorrenti

Si guadagna un **punteggio** per ogni flag conquistato.

CTF - Attack/Defense

Il formato Attack/Defense prevede che gli organizzatori forniscano ai giocatori (principalmente squadre) una o più macchine virtuali.

I giocatori quindi avranno un doppio ruolo:

- **difendere** i servizi esposti sulla propria macchina virtuale
- **attaccare** i servizi dei team concorrenti

Si guadagna un **punteggio** per ogni flag conquistato.

Generalmente sono eventi *offline* e con un limite di tempo.

CTF - Boot2Root

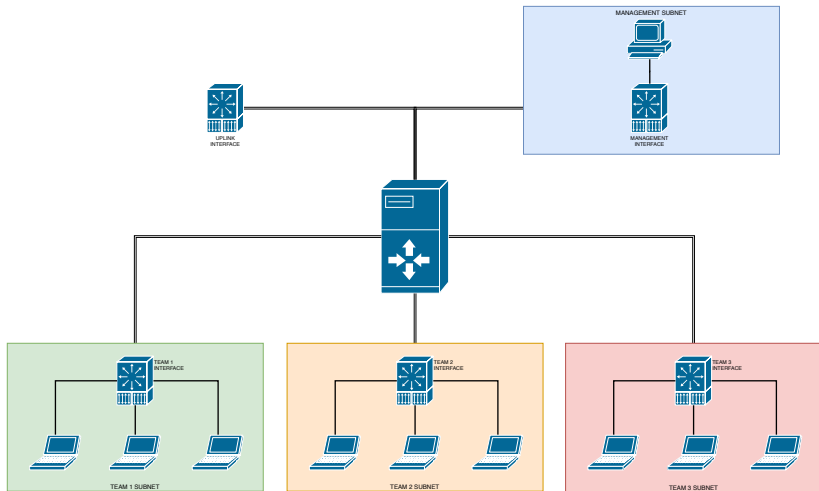
Consiste l'installazione di una *macchina virtuale* e lo scopo è di trovare e sfruttarne le vulnerabilità che permettano di avere l'accesso root alla stessa.

CTF - Boot2Root

Consiste l'installazione di una *macchina virtuale* e lo scopo è di trovare e sfruttarne le vulnerabilità che permettano di avere l'accesso root alla stessa.

Questo formato di CTF è principalmente preferito da singoli che vogliono esercitarsi a sfruttare vulnerabilità in uno scenario semi-reale, ma non preclude la possibilità di creare un team.

Scenario



Software

- iptables

Software

- iptables
- netplan

Software

- iptables
- netplan
- script python

Software

- iptables
- netplan
- script python
- script bash

Definizione

Iptables è un potente firewall integrato nel kernel Linux. Permette di definire, tramite regole, il filtraggio, la manipolazione dei pacchetti ed il NAT.

È strutturato in **tabelle**, ognuna contiene le proprie **catene** che possono, anche, essere definite dall'utente.

Ogni catena è una *lista di regole* che specificano l'**azione** da intraprendere con i pacchetti che corrispondono alla regola.

Quest'*azione* corrisponde al **target**.

TARGETS

Una regola di firewall specifica il criterio di selezione dei pacchetti ed il **target**.

TARGETS

Una regola di firewall specifica il criterio di selezione dei pacchetti ed il **target**.

Se il pacchetto **non corrisponde** → verrà esaminata la **regola successiva**.

TARGETS

Una regola di firewall specifica il criterio di selezione dei pacchetti ed il **target**.

Se il pacchetto **non corrisponde** → verrà esaminata la **regola successiva**.

Se il pacchetto **corrisponde** → verrà intrapresa l'azione definita dal *target*

TARGETS

Una regola di firewall specifica il criterio di selezione dei pacchetti ed il **target**.

Se il pacchetto **non corrisponde** → verrà esaminata la **regola successiva**.

Se il pacchetto **corrisponde** → verrà intrapresa l'azione definita dal *target*, che può essere:

ACCEPT lascia transitare il pacchetto

TARGETS

Una regola di firewall specifica il criterio di selezione dei pacchetti ed il **target**.

Se il pacchetto **non corrisponde** → verrà esaminata la **regola successiva**.

Se il pacchetto **corrisponde** → verrà intrapresa l'azione definita dal *target*, che può essere:

ACCEPT lascia transitare il pacchetto

DROP scarta il pacchetto

TARGETS

Una regola di firewall specifica il criterio di selezione dei pacchetti ed il **target**.

Se il pacchetto **non corrisponde** → verrà esaminata la **regola successiva**.

Se il pacchetto **corrisponde** → verrà intrapresa l'azione definita dal *target*, che può essere:

ACCEPT lascia transitare il pacchetto

DROP scarta il pacchetto

RETURN ferma la "traversata" dell'attuale catena e ritorna alla catena chiamante (precedente)

TARGETS

Una regola di firewall specifica il criterio di selezione dei pacchetti ed il **target**.

Se il pacchetto **non corrisponde** → verrà esaminata la **regola successiva**.

Se il pacchetto **corrisponde** → verrà intrapresa l'azione definita dal *target*, che può essere:

ACCEPT lascia transitare il pacchetto

DROP scarta il pacchetto

RETURN ferma la "traversata" dell'attuale catena e ritorna alla catena chiamante (precedente)

catena definita dall'utente

CHAINS

Le catene predefinite sono:

INPUT per i pacchetti destinati ad un processo del server locale

CHAINS

Le catene predefinite sono:

INPUT per i pacchetti destinati ad un processo del server locale

OUTPUT per i pacchetti creati dal server locale

CHAINS

Le catene predefinite sono:

INPUT per i pacchetti destinati ad un processo del server locale

OUTPUT per i pacchetti creati dal server locale

FORWARD per i pacchetti inoltrati tramite il server locale

CHAINS

Le catene predefinite sono:

INPUT per i pacchetti destinati ad un processo del server locale

OUTPUT per i pacchetti creati dal server locale

FORWARD per i pacchetti inoltrati tramite il server locale

POSTROUTING per l'alterazione dei pacchetti appena prima dell'uscita

CHAINS

Le catene predefinite sono:

INPUT per i pacchetti destinati ad un processo del server locale

OUTPUT per i pacchetti creati dal server locale

FORWARD per i pacchetti inoltrati tramite il server locale

POSTROUTING per l'alterazione dei pacchetti appena prima dell'uscita

PREROUTING per l'alterazione dei pacchetti appena entrati

TABLES

Ogni tabella ha le proprie catene:

filter INPUT, FORWARD, OUTPUT.

TABLES

Ogni tabella ha le proprie catene:

`filter` INPUT, FORWARD, OUTPUT.

`nat` PREROUTING, INPUT, OUTPUT,
POSTROUTING.

TABLES

Ogni tabella ha le proprie catene:

filter INPUT, FORWARD, OUTPUT.

nat PREROUTING, INPUT, OUTPUT,
POSTROUTING.

mangle PREROUTING, INPUT, OUTPUT,
POSTROUTING, FORWARD.

TABLES

Ogni tabella ha le proprie catene:

filter INPUT, FORWARD, OUTPUT.

nat PREROUTING, INPUT, OUTPUT,
POSTROUTING.

mangle PREROUTING, INPUT, OUTPUT,
POSTROUTING, FORWARD.

raw PREROUTING, OUTPUT

TABLES

Ogni tabella ha le proprie catene:

filter INPUT, FORWARD, OUTPUT.

nat PREROUTING, INPUT, OUTPUT,
POSTROUTING.

mangle PREROUTING, INPUT, OUTPUT,
POSTROUTING, FORWARD.

raw PREROUTING, OUTPUT

security INPUT, OUTPUT, FORWARD.

Netplan

Netplan è uno strumento per la configurazione del networking sui sistemi linux.

Netplan

Netplan è uno strumento per la configurazione del networking sui sistemi linux.

Sfrutta un file di configurazione in formato YAML (Yet Another Markup Language)

Netplan

Netplan è uno strumento per la configurazione del networking sui sistemi linux.

Sfrutta un file di configurazione in formato YAML (Yet Another Markup Language) con:

- le interfacce da utilizzare

Netplan

Netplan è uno strumento per la configurazione del networking sui sistemi linux.

Sfrutta un file di configurazione in formato YAML (Yet Another Markup Language) con:

- le interfacce da utilizzare
- i parametri di configurazione per ogni interfaccia

Netplan

Netplan è uno strumento per la configurazione del networking sui sistemi linux.

Sfrutta un file di configurazione in formato YAML (Yet Another Markup Language) con:

- le interfacce da utilizzare
- i parametri di configurazione per ogni interfaccia
- il **renderer** da utilizzare.

Netplan

Netplan è uno strumento per la configurazione del networking sui sistemi linux.

Sfrutta un file di configurazione in formato YAML (Yet Another Markup Language) con:

- le interfacce da utilizzare
- i parametri di configurazione per ogni interfaccia
- il **renderer** da utilizzare.

Da questo file, Netplan, genera la configurazione finale per il *renderer* scelto.

Configurazione netplan

Supponiamo di avere tre TEAM e le seguenti interfacce:

ens33 interfaccia di UPLINK

Configurazione netplan

Supponiamo di avere tre TEAM e le seguenti interfacce:

ens33 interfaccia di UPLINK

ens37 interfaccia per il subnet di MANAGEMENT

Configurazione netplan

Supponiamo di avere tre TEAM e le seguenti interfacce:

ens33 interfaccia di UPLINK

ens37 interfaccia per il subnet di MANAGEMENT

ens38 interfaccia per il subnet del TEAM 1

Configurazione netplan

Supponiamo di avere tre TEAM e le seguenti interfacce:

ens33 interfaccia di UPLINK

ens37 interfaccia per il subnet di MANAGEMENT

ens38 interfaccia per il subnet del TEAM 1

ens39 interfaccia per il subnet del TEAM 2

Configurazione netplan

Supponiamo di avere tre TEAM e le seguenti interfacce:

ens33 interfaccia di UPLINK

ens37 interfaccia per il subnet di MANAGEMENT

ens38 interfaccia per il subnet del TEAM 1

ens39 interfaccia per il subnet del TEAM 2

ens40 interfaccia per il subnet del TEAM 3

Configurazione netplan

Con le interfacce definite prima avremo:

```
network:
  ethernets:
    ens33:
      dhcp4: true
      dhcp6: false
    ens37:
      addresses:
        - 172.168.2.128/24
      dhcp4: false
      dhcp6: false
    ens38:
      addresses:
        - 172.168.3.100/24
      dhcp4: false
      dhcp6: false
```

Configurazione netplan

```
ens39:
  addresses:
  - 172.168.4.100/24
  dhcp4: false
  dhcp6: false
ens40:
  addresses:
  - 172.168.5.100/24
  dhcp4: false
  dhcp6: false
renderer: networkd
version: 2
```

Script realizzato in python

Lo script esegue tutto il lavoro di configurazione, sia per netplan sia per le regole di firewall (quindi iptables).

Per essere eseguito, lo script, ha bisogno di alcuni parametri in input.

Esso genera un file di configurazione, in formato JSON, editabile anche manualmente, che viene utilizzato per configurare la competizione.

Formato file di configurazione

```
"ManagementInterface": "ens37",  
"ManagementInterfaceAddress": "172.168.2.100",  
"NumberOfTeams": 3,  
"Team1Interface": "ens38",  
"Team1InterfaceAddress": "172.168.3.100",  
"Team2Interface": "ens39",  
"Team2InterfaceAddress": "172.168.4.100",  
"Team3Interface": "ens40",  
"Team3InterfaceAddress": "172.168.5.100",  
"UplinkAddress": "172.168.1.128",  
"UplinkInterface": "ens33",  
"Log": "3/sec"
```


Parametri dello script

-l configura la gara in modo interattivo

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink
- ua indirizzo dell'interfaccia di uplink

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink
- ua indirizzo dell'interfaccia di uplink
- mi nome dell'interfaccia di Management

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink
- ua indirizzo dell'interfaccia di uplink
- mi nome dell'interfaccia di Management
- ma indirizzo dell'interfaccia di Management

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink
- ua indirizzo dell'interfaccia di uplink
- mi nome dell'interfaccia di Management
- ma indirizzo dell'interfaccia di Management
- masq come effettuare il masquerading (false, IP singolo, IP per ogni squadra, true)

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink
- ua indirizzo dell'interfaccia di uplink
- mi nome dell'interfaccia di Management
- ma indirizzo dell'interfaccia di Management
- masq come effettuare il masquerading (false, IP singolo, IP per ogni squadra, true)
- t le interfacce delle squadre

Parametri dello script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink
- ua indirizzo dell'interfaccia di uplink
- mi nome dell'interfaccia di Management
- ma indirizzo dell'interfaccia di Management
- masq come effettuare il masquerading (false, IP singolo, IP per ogni squadra, true)
- t le interfacce delle squadre
- l il limite di logging

Obiettivi Fase 1

- Isolamento dei Team

Obiettivi Fase 1

- Isolamento dei Team
- Solo Management e Virtual Router possono iniziare una connessione verso gli altri

Obiettivi Fase 1

- Isolamento dei Team
- Solo Management e Virtual Router possono iniziare una connessione verso gli altri
- Solo Management e Virtual Router possono connettersi all'esterno, attraverso l'UPLINK

Obiettivi Fase 1

- Isolamento dei Team
- Solo Management e Virtual Router possono iniziare una connessione verso gli altri
- Solo Management e Virtual Router possono connettersi all'esterno, attraverso l'UPLINK
- LOG dei pacchetti entranti e dei pacchetti inoltrati

Regole specifiche per il Virtual Router

Connessione all'esterno attraverso UPLINK

```
$ iptables -P OUTPUT ACCEPT
```

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -m conntrack --ctstate ESTABLISHED -j ACCEPT
```


Regole specifiche per il Virtual Router

Ricevere connessioni da MANAGEMENT (qualsiasi tipo)

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -i $MANAGEMENT_INTERFACE -j ACCEPT
```

```
$ iptables -P OUTPUT ACCEPT
```

Regole specifiche per il Virtual Router

Connessioni ai TEAM e ricezione risposta

```
$ iptables -P OUTPUT ACCEPT
```

```
$ iptables -A INPUT -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

Regole specifiche per il Virtual Router

Connessione loopback

```
$ iptables -P OUTPUT ACCEPT
```

```
$ iptables -A INPUT -i lo -j ACCEPT
```

Regole specifiche per il Virtual Router

Blocco connessioni dall'esterno e dai TEAM (se non iniziate da VR)

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

Regole specifiche per il Management

Connessione all'esterno attraverso UPLINK

```
$ iptables -P FORWARD DROP
```

```
$ iptables -t nat -A POSTROUTING -o $UPLINK -j MASQUERADE
```

```
$ iptables -A FORWARD -i $MANAGEMENT_INTERFACE -j ACCEPT
```

```
$ iptables -A FORWARD -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

Regole specifiche per il Management

Connessione al VIRTUAL ROUTER

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -i $MANAGEMENT_INTERFACE -j ACCEPT
```

```
$ iptables -P OUTPUT ACCEPT
```

Regole specifiche per il Management

Connessione ai TEAM

```
$ iptables -P FORWARD DROP
```

```
$ iptables -A FORWARD -i $MANAGEMENT_INTERFACE -j ACCEPT
```

```
$ iptables -A FORWARD -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

Regole specifiche per il Management

Blocco connessioni dall'esterno e dai TEAM (se non iniziate da MANAGEMENT)

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -m conntrack --ctstate ESTABLISHED -j ACCEPT
```


Regole specifiche per i Team

Rispondere a connessioni iniziate da MANAGEMENT

```
$ iptables -P FORWARD DROP
```

```
$ iptables -A FORWARD -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

Regole specifiche per i Team

Rispondere a connessioni iniziate da VR

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

Regole specifiche per i Team

Blocco inizio connessioni verso:

- MANAGEMENT

```
$ iptables -P FORWARD DROP
```

- VIRTUAL ROUTER

```
$ iptables -P INPUT DROP
```

- altri TEAM

```
$ iptables -P FORWARD DROP
```

- esterno tramite UPLINK

```
$ iptables -P FORWARD DROP
```

Regole per il LOG

LOG dei pacchetti entranti

```
$ iptables -N LOGGING
```

```
$ iptables -A INPUT -j LOGGING
```

```
$ iptables -A LOGGING -m limit --limit $LOGLIMIT -j LOG --log-prefix  
"COMPETITION-LOG: " --log-level 4
```

Regole per il LOG

LOG dei pacchetti inoltrati

```
$ iptables -N LOGGING
```

```
$ iptables -A LOGGING -m limit --limit $LOGLIMIT -j LOG --log-prefix  
"COMPETITION-LOG: " --log-level 4
```

```
$ iptables -A FORWARD -j LOGGING
```

Obiettivi Fase 2

- Solo Management e Virtual Router possono iniziare una connessione verso chiunque

Obiettivi Fase 2

- Solo Management e Virtual Router possono iniziare una connessione verso chiunque
- Solo Management e Virtual Router possono connettersi all'esterno, attraverso l'UPLINK

Obiettivi Fase 2

- Solo Management e Virtual Router possono iniziare una connessione verso chiunque
- Solo Management e Virtual Router possono connettersi all'esterno, attraverso l'UPLINK
- Team possono comunicare tra loro

Obiettivi Fase 2

- Solo Management e Virtual Router possono iniziare una connessione verso chiunque
- Solo Management e Virtual Router possono connettersi all'esterno, attraverso l'UPLINK
- Team possono comunicare tra loro
- Team non possono iniziare connessioni verso MANAGEMENT e VIRTUAL ROUTER
- LOG dei pacchetti entranti e dei pacchetti inoltrati

Regole specifiche per il Virtual Router

Le regole rimangono le stesse della FASE 1

Regole specifiche per il Management

Connessione all'esterno attraverso UPLINK

```
$ iptables -P FORWARD ACCEPT
```

Regole specifiche per il Management

Connessione al VIRTUAL ROUTER

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -i $MANAGEMENT_INTERFACE -j ACCEPT
```

```
$ iptables -P OUTPUT ACCEPT
```

Regole specifiche per il Management

Connessione ai TEAM

```
$ iptables -P FORWARD ACCEPT
```

```
$ iptables -A FORWARD -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

Regole specifiche per il Management

Blocco connessioni dall'esterno e dai TEAM (se non iniziate da MANAGEMENT)

```
$ iptables -P FORWARD ACCEPT
```

```
$ iptables -A FORWARD -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

```
$ iptables -A FORWARD -o $MANAGEMENT_INTERFACE -j DROP
```

Regole specifiche per i Team

Rispondere a connessioni iniziate da MANAGEMENT

```
$ iptables -P FORWARD ACCEPT
```

```
$ iptables -A FORWARD -m conntrack --ctstate ESTABLISHED -j ACCEPT
```

```
$ iptables -A FORWARD -o $MANAGEMENT_INTERFACE -j DROP
```

Regole specifiche per i Team

Rispondere a connessioni iniziate da VR

```
$ iptables -P INPUT DROP
```

```
$ iptables -A INPUT -m conntrack --ctstate ESTABLISHED -j ACCEPT
```


Regole specifiche per i Team

Connessioni verso altri TEAM

```
$ iptables -P FORWARD ACCEPT
```

Regole specifiche per i Team

Blocco inizio connessioni verso:

■ MANAGEMENT

```
$ iptables -P FORWARD ACCEPT
```

```
$ iptables -A FORWARD -o $MANAGEMENT_INTERFACE -j DROP
```

■ VIRTUAL ROUTER

```
$ iptables -P INPUT DROP
```

■ esterno tramite UPLINK

```
$ iptables -P FORWARD ACCEPT
```

```
$ iptables -A FORWARD -o $UPLINK -j DROP
```

Regole per il LOG

Le regole sono le stesse della FASE 1

NAT

Il NAT, ovvero *Network Address Translation*, conosciuto anche come **network masquerading**, è una tecnica che consiste nel modificare gli **indirizzi IP** contenuti negli **header** dei pacchetti in transito su un sistema che agisce da **router** all'interno di una comunicazione tra due o più *host*.

NAT

Il NAT, ovvero *Network Address Translation*, conosciuto anche come **network masquerading**, è una tecnica che consiste nel modificare gli **indirizzi IP** contenuti negli **header** dei pacchetti in transito su un sistema che agisce da **router** all'interno di una comunicazione tra due o più *host*.

Nel nostro caso verrà mascherato l'indirizzo sorgente del pacchetto.

MASQUERADING

Lo script è stato ideato per permettere quattro situazioni di *masquerading*:

- Un indirizzo unico per tutti i TEAM

MASQUERADING

Lo script è stato ideato per permettere quattro situazioni di *masquerading*:

- Un indirizzo unico per tutti i TEAM, che può essere:
 - definito dall'amministratore di gara

MASQUERADING

Lo script è stato ideato per permettere quattro situazioni di *masquerading*:

- Un indirizzo unico per tutti i TEAM, che può essere:
 - definito dall'amministratore di gara
 - generato in modo casuale

MASQUERADING

Lo script è stato ideato per permettere quattro situazioni di *masquerading*:

- Un indirizzo unico per tutti i TEAM, che può essere:
 - definito dall'amministratore di gara
 - generato in modo casuale
- Indirizzi diversi per tutti i TEAM

MASQUERADING

Lo script è stato ideato per permettere quattro situazioni di *masquerading*:

- Un indirizzo unico per tutti i TEAM, che può essere:
 - definito dall'amministratore di gara
 - generato in modo casuale
- Indirizzi diversi per tutti i TEAM, che possono essere:
 - definiti dall'amministratore di gara

MASQUERADING

Lo script è stato ideato per permettere quattro situazioni di *masquerading*:

- Un indirizzo unico per tutti i TEAM, che può essere:
 - definito dall'amministratore di gara
 - generato in modo casuale
- Indirizzi diversi per tutti i TEAM, che possono essere:
 - definiti dall'amministratore di gara
 - generati in modo casuale

Regole per il Masquerading

La regola per il **masquerading** è strutturata in questo modo:

```
$ iptables -t nat -A POSTROUTING -o $interface -j SNAT  
    --to-source $MASQUERADING_ADDRESS
```

Ricordiamo i parametri accettati dallo script

- I configura la gara in modo interattivo
- G mostra la configurazione attuale (del file di config)
- L mostra tutte le interfacce della macchina
- p per specificare la fase (1 o 2)
- ui nome dell'interfaccia di uplink
- ua indirizzo dell'interfaccia di uplink
- mi nome dell'interfaccia di Management
- ma indirizzo dell'interfaccia di Management
- masq come effettuare il masquerading (false, IP singolo, IP per ogni squadra, true)
 - t le interfacce delle squadre
 - l il limite di logging

Esempi

```
-p 1 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq true -t ens38 ens39 ens40 -l 4/sec
```

Avrà i seguenti effetti sulla tabella FILTER:

```
Chain INPUT (policy DROP)
target      prot opt source                destination
LOGGING     all  —   anywhere              anywhere
ACCEPT      all  —   anywhere              anywhere
ACCEPT      all  —   172.168.1.1          anywhere
ACCEPT      all  —   anywhere              anywhere
ACCEPT      all  —   anywhere              anywhere                                ctstate ESTABLISHED

Chain FORWARD (policy DROP)
target      prot opt source                destination
LOGGING     all  —   anywhere              anywhere
ACCEPT      all  —   anywhere              anywhere
ACCEPT      all  —   anywhere              anywhere                                ctstate ESTABLISHED

Chain OUTPUT (policy ACCEPT)
target      prot opt source                destination

Chain LOGGING (2 references)
target      prot opt source                destination
LOG         all  —   anywhere              anywhere                                limit: avg 4/sec
burst 5 LOG level warning prefix "COMPETITION-LOG: "
```

Esempi

```
-p 1 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq true -t ens38 ens39 ens40 -l 4/sec
```

Avrà i seguenti effetti sulla tabella NAT:

Chain PREROUTING (policy ACCEPT)	
target	prot opt source destination
Chain INPUT (policy ACCEPT)	
target	prot opt source destination
Chain OUTPUT (policy ACCEPT)	
target	prot opt source destination
Chain POSTROUTING (policy ACCEPT)	
target	prot opt source destination
MASQUERADE	all — anywhere anywhere

Esempi

```
-p 1 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq true -t ens38 ens39 ens40 -l 4/sec
```

Avrà i seguenti effetti sulla configurazione delle interfacce:

```
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.1.128 netmask 255.255.255.0 broadcast 172.168.1.255

ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.2.100 netmask 255.255.255.0 broadcast 172.168.2.255

ens38: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.3.100 netmask 255.255.255.0 broadcast 172.168.3.255

ens39: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.4.100 netmask 255.255.255.0 broadcast 172.168.4.255

ens40: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.5.100 netmask 255.255.255.0 broadcast 172.168.5.255

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
```


Esempi

```
-p 2 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq true -t ens38 ens39 ens40 -l 4/sec
```

Avrà i seguenti effetti sulla tabella FILTER:

Chain INPUT (policy DROP)

target	prot	opt	source	destination	
LOGGING	all	—	anywhere	anywhere	
ACCEPT	all	—	anywhere	anywhere	
ACCEPT	all	—	172.168.1.1	anywhere	
ACCEPT	all	—	anywhere	anywhere	ctstate ESTABLISHED
ACCEPT	all	—	anywhere	anywhere	

Chain FORWARD (policy ACCEPT)

target	prot	opt	source	destination	
LOGGING	all	—	anywhere	anywhere	
ACCEPT	all	—	anywhere	anywhere	
ACCEPT	all	—	anywhere	anywhere	ctstate ESTABLISHED
DROP	all	—	anywhere	anywhere	
DROP	all	—	anywhere	anywhere	

Chain OUTPUT (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain LOGGING (2 references)

target	prot	opt	source	destination	
LOG	all	—	anywhere	anywhere	limit: avg 4/sec burst 5 LOG level warning prefix "COMPETITION-LOG: "

Esempi

```
—p 2 —ui ens33 —mi ens37 —ma 172.168.2.100 —masq true —t ens38 ens39 ens40 —l 4/sec
```

Avrà i seguenti effetti sulla tabella NAT:

Chain PREROUTING (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain INPUT (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain OUTPUT (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain POSTROUTING (policy ACCEPT)

target	prot	opt	source	destination
MASQUERADE	all	—	anywhere	anywhere
SNAT	all	—	anywhere	anywhere to:136.133.12.28
SNAT	all	—	anywhere	anywhere to:76.138.130.11
SNAT	all	—	anywhere	anywhere to:11.49.228.130

Esempi

```
-p 1 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq true -t ens38 ens39 ens40 -l 4/sec
```

Avrà i seguenti effetti sulla configurazione delle interfacce:

```
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.1.128 netmask 255.255.255.0 broadcast 172.168.1.255

ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.2.100 netmask 255.255.255.0 broadcast 172.168.2.255

ens38: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.3.100 netmask 255.255.255.0 broadcast 172.168.3.255

ens39: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.4.100 netmask 255.255.255.0 broadcast 172.168.4.255

ens40: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 172.168.5.100 netmask 255.255.255.0 broadcast 172.168.5.255

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
```

Esempi

```
-p 2 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq 123.123.123.123
-t ens38 ens39 -l 4/sec
```

Avrà i seguenti effetti sulla tabella FILTER:

Chain INPUT (policy DROP)

target	prot	opt	source	destination	
LOGGING	all	—	anywhere	anywhere	
ACCEPT	all	—	anywhere	anywhere	
ACCEPT	all	—	172.168.1.1	anywhere	
ACCEPT	all	—	anywhere	anywhere	ctstate ESTABLISHED
ACCEPT	all	—	anywhere	anywhere	

Chain FORWARD (policy ACCEPT)

target	prot	opt	source	destination	
LOGGING	all	—	anywhere	anywhere	
ACCEPT	all	—	anywhere	anywhere	
ACCEPT	all	—	anywhere	anywhere	ctstate ESTABLISHED
DROP	all	—	anywhere	anywhere	
DROP	all	—	anywhere	anywhere	

Chain OUTPUT (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain LOGGING (2 references)

target	prot	opt	source	destination	
LOG	all	—	anywhere	anywhere	limit: avg 4/sec

burst 5 LOG level warning prefix "COMPETITION-LOG: "

Esempi

```
-p 2 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq 123.123.123.123
-t ens38 ens39 -l 4/sec
```

Avrà i seguenti effetti sulla tabella NAT:

Chain PREROUTING (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain INPUT (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain OUTPUT (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

Chain POSTROUTING (policy ACCEPT)

target	prot	opt	source	destination
--------	------	-----	--------	-------------

MASQUERADE	all	—	anywhere	anywhere
------------	-----	---	----------	----------

SNAT	all	—	anywhere	anywhere
------	-----	---	----------	----------

SNAT	all	—	anywhere	anywhere
------	-----	---	----------	----------

to:123.123.123.123

to:123.123.123.123

Esempi

```
-p 2 -ui ens33 -mi ens37 -ma 172.168.2.100 -masq 123.123.123.123  
-t ens38 ens39 -l 4/sec
```

Avrà i seguenti effetti sulla configurazione delle interfacce:

```
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 172.168.1.128 netmask 255.255.255.0 broadcast 172.168.1.255  
  
ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 172.168.2.100 netmask 255.255.255.0 broadcast 172.168.2.255  
  
ens38: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 172.168.3.100 netmask 255.255.255.0 broadcast 172.168.3.255  
  
ens39: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 172.168.4.100 netmask 255.255.255.0 broadcast 172.168.4.255  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0
```