Cheatsheet 2: `tcpdump` e `netcat`

Laboratorio di Reti e Sistemi Distribuiti @ UniME

February 24, 2025

1 Introduzione

`tcpdump` e `netcat` (spesso abbreviato come `nc`) sono strumenti di rete ampiamente utilizzati per l'analisi del traffico di rete e la comunicazione tra sistemi.

2 Guida a `tcpdump`

2.1 Catturare pacchetti su un'interfaccia

Per catturare pacchetti su un'interfaccia specifica, si utilizza:

```
sudo tcpdump -i <interfaccia>
```

Esempio: Catturare pacchetti sull'interfaccia `eth0`:

```
sudo tcpdump -i eth0
```

2.2 Filtrare pacchetti per protocollo

È possibile filtrare i pacchetti in base al protocollo (ad esempio, TCP, UDP, ICMP):

```
sudo tcpdump -i <interfaccia> <protocollo>
```

Esempio: Catturare solo pacchetti TCP sull'interfaccia `eth0`:

```
sudo tcpdump -i eth0 tcp
```

2.3 Filtrare pacchetti per indirizzo IP

Per filtrare i pacchetti in base all'indirizzo IP di origine o destinazione, si utilizza:

```
sudo tcpdump -i <interfaccia> host <indirizzo-IP>
```

Esempio: Catturare pacchetti destinati o provenienti dall'indirizzo IP `192.168.1.1`:

```
sudo tcpdump -i eth0 host 192.168.1.1
```

2.4 Salvare i pacchetti in un file

Per salvare i pacchetti catturati in un file, si utilizza l'opzione `-w`:

```
sudo tcpdump -i <interfaccia> -w <file.pcap>
```

Esempio: Salvare i pacchetti catturati su `eth0` in un file chiamato `cattura.pcap`:

```
sudo tcpdump -i eth0 -w cattura.pcap
```

2.5 Leggere i pacchetti da un file

Per leggere i pacchetti da un file precedentemente salvato, si utilizza l'opzione `-r`:

```
sudo tcpdump -r <file.pcap>
```

Esempio: Leggere i pacchetti dal file `cattura.pcap`:

```
sudo tcpdump -r cattura.pcap
```

3 Guida a `netcat (nc)`

3.1 Inviare e ricevere dati

`netcat` può essere utilizzato per inviare e ricevere dati attraverso una connessione TCP o UDP.

- **Aprire una connessione in ascolto:**

  ```
  nc -l -p <porta>
  ```

- **Connettersi a una porta remota:**

  ```
  nc <indirizzo-IP> <porta>
  ```

Esempio: Aprire una connessione in ascolto sulla porta `1234`:

```
nc -l -p 1234
```

Esempio: Connettersi a un server sulla porta `1234`:

```
nc 192.168.1.100 1234
```

3.2 Trasferire file

`netcat` può essere utilizzato per trasferire file tra due sistemi.

- **Inviare un file:**

  ```
  nc -l -p <porta> > <file-ricevuto>
  ```

- **Ricevere un file:**

```
nc <indirizzo-IP> <porta> < <file-da-inviare>
```

Esempio: Ricevere un file sulla porta `1234`:

```
nc -l -p 1234 > file-ricevuto.txt
```

Esempio: Inviare un file alla porta `1234`:

```
nc 192.168.1.100 1234 < file-da-inviare.txt
```

3.3 Eseguire una shell remota

`netcat` può essere utilizzato per eseguire una shell remota.

- **Aprire una shell in ascolto:**

```
nc -l -p <porta> -e /bin/bash
```

- **Connettersi a una shell remota:**

```
nc <indirizzo-IP> <porta>
```

Esempio: Aprire una shell in ascolto sulla porta `1234`:

```
nc -l -p 1234 -e /bin/bash
```

Esempio: Connettersi a una shell remota sulla porta `1234`:

```
nc 192.168.1.100 1234
```

4 Note importanti

- `tcpdump` richiede i permessi di root per essere eseguito. Utilizzare `sudo` per eseguire i comandi