

# **RELAZIONE LABORATORIO VIRTUALE**

Milani Francesco  
5 AI  
A.S. 2019/2020

# RELAZIONE DI SISTEMI E RETI

## INDICE

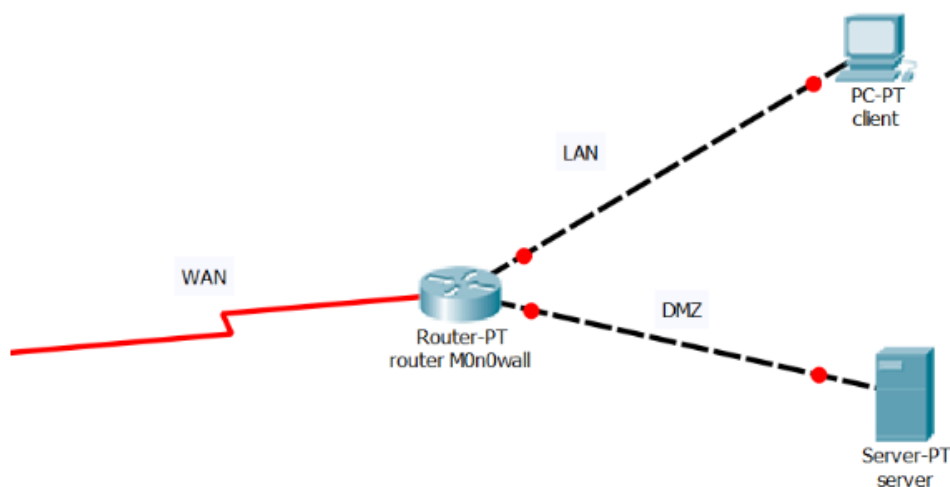
1. Scopo dell'esperienza	4
2. Creazione del client	5
2.1 Note sulla creazione del client	6
3. Installazione Debian	7
3.1 Impostazione utenti e password	
3.2 Configurazione orologio	9
3.3 Partizionamento dei dischi	9
3.4 Installazione sistema base	9
3.5 Driver da includere	9
3.6 Configuratore gestore pacchetti	9
3.7 Selezione e installazione del software	9
3.8 Installazione bootloader GRUB su disco fisso	9
3.9 Note sull'installazione di Debian	10
4. Configurazione del client	11
4.1 Installazione software aggiuntivo	11
4.2 Aggiunta uds al gruppo sudo	11
4.3 Comandi utili	11
4.4 Installazione GUI del client	12
4.5 Note sulla configurazione del client	13
5. Creazione e configurazione del server	14
5.1 Note sulla creazione e configurazione del server	14
6. Creazione e configurazione del router	15
6.1 Impostazioni schede di rete del router	15
6.3 Note sulla creazione e configurazione del router	16
7. Installazione e configurazione di m0n0wall	17
7.1 Ridenominazione schede di rete del router	17
8. Impostazioni di rete del client	18
9. Configurazione m0n0wall lato client	19
10. Configurazione m0n0wall dall'host	20
10.1 Configurazione interfacce	20
10.2 Configurazione regole DMZ	21
11. Configurazione degli Aliases	22

12. Migrazione indirizzi IP	23
12.1 Migrazione indirizzo IP client	23
12.2 Migrazione indirizzo IP server	24
13. Configurazione regole di firewall	25
13.1 Configurazione firewall: LAN	25
13.2 Configurazione firewall: WAN	26
13.3 Configurazione firewall: DMZ	27
14. Configurazione regole del NAT	28
15. Installazione di apache	29
16. Installazione e configurazione di una VPN	30
16.1 Configurazione OpenVPN	31
17. Installazione e configurazione SNMP	33
17.1 Configurazione SNMP su m0n0wall	33
17.2 Configurazione MRTG	34
17.3 Configurazione MRTG per grafici server	35
18. Installazione e configurazione Cacti	36
18.1 Installazione sul server	36

## 1. Scopo dell'esperienza

Lo scopo di questa esperienza è quello di riuscire a creare un laboratorio virtuale mediante l'uso di VirtualBox, gestendo 3 diverse reti: LAN, WAN e DMZ, e configurando correttamente tutti i loro componenti.

Per simulare client e server abbiamo dovuto utilizzare le macchine virtuali non per carenza fisica di dispositivi, ma bensì per ragioni di comodità e semplicità.



# VirtualBox

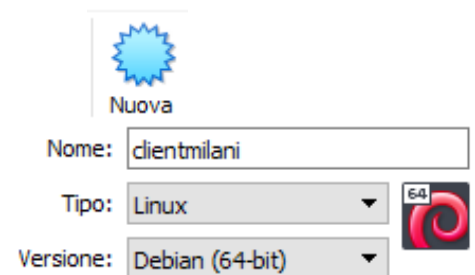


# debian

## 2. Creazione del client

- Aprire VirtualBox, e selezionare l'icona "Nuova"

- Nome: *clientmilani*
- Tipo: *Linux*
- Versione: *Debian 64 Bit*
- Dimensione memoria RAM: *1024 MB*

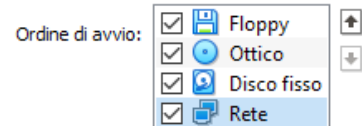


- Creare un nuovo disco fisso
  - Tipo di file: *VDI (VirtualBox Disk Image)*
  - Archiviazione: *Allocato dinamicamente*
  - Nome: *clientmilani.vdi*
  - Dimensione: *4GB*

☒ Crea subito un nuovo disco fisso virtuale

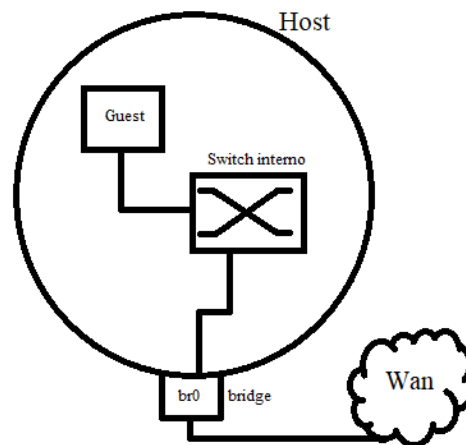
- Impostazioni del client

- Sistema → Ordine di avvio → spuntare *Rete*
- Rete:
  - Connessa a : *Scheda con Bridge*



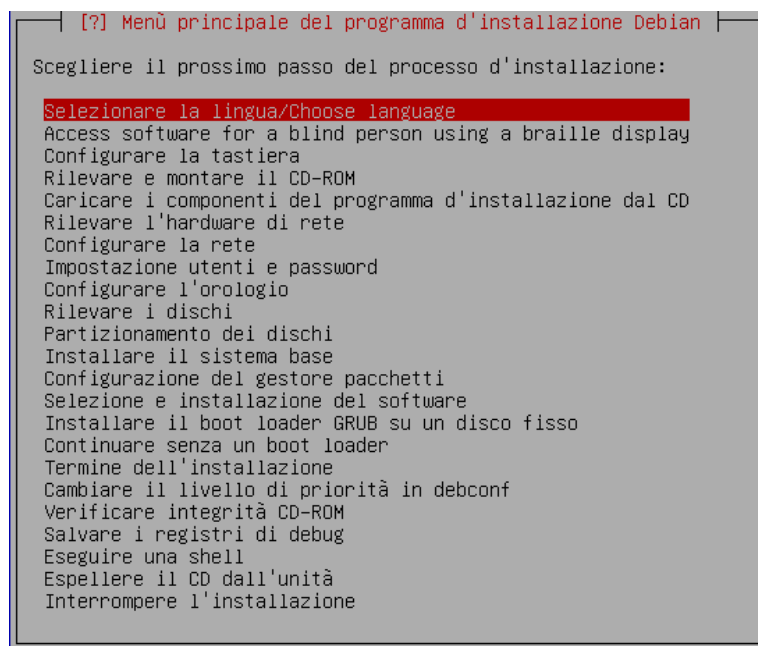
## 2.1 Note sulla creazione del client

- Il disco fisso viene allocato dinamicamente in quanto l'allocazione dinamica è meno prestante rispetto all'allocazione statica, ma è più utile per il nostro utilizzo.
- Connettiamo il computer ad una scheda con bridge, ovvero al posto del router verrà creato uno switch virtuale interno all'host. La scheda di rete quindi accetterà le proprie trame, insieme a quelle di broadcast e multicast, e si potrà istruire per ricevere MAC address specifici.



- Come indirizzo MAC verrà utilizzato un indirizzo generico generato automaticamente, che non corrisponde a nessun produttore, in modo da poterlo usare liberamente senza problemi.

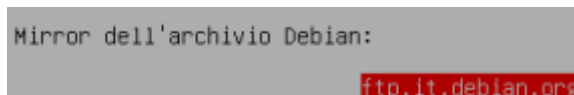
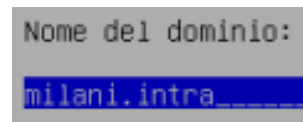
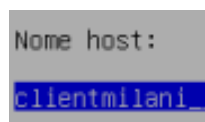
### 3. Installazione Debian



- Lingua: *Italiano*
  - Configurare i locale: *it\_IT*, *it\_IT@euro*
  - Locale prefinito: *UTF-8*



- Tastiera: *Italiano*
- Rilevare l'hardware di rete
  - Configurazione automatica
    - Nome host: *clientmilani*
    - Nome dominio: *milani.intra*
    - Mirror: *http*
    - Nazione del mirror dell'archivio Debian: *Italia*
    - Mirror dell'archivio Debian: *ftp.it.debian.org*



- Informazioni del proxy http: *http://apt-cacher.fermi.intra:3142*
- Versione Debian: *Stable – buster*

### 3.1 Impostazione utenti e password

- Shadow password: *Sì*
- Permettere il root: *Sì*
- Password: *lasolita*
- Account normale:
  - Nome: *utente di servizio*
  - Nome utente: *uds*
  - Password: *lasolita*

Abilitare le «shadow password»?

☒ **<Sì>**    ☐ <No>

Permettere l'accesso a root?

☒ **<Sì>**    ☐ <No>

### 3.2 Configurazione orologio

- NTP: *Sì*
- Server: *Sì*
- Fuso orario: *Europe/Rome*

### 3.3 Partizionamento dei dischi

- Manuale
- VBOX Hard Disk
- Tabelle: *msdos*
- Spazio libero 1:
  - Nuova partizione: *4.0GB*
  - Primaria
  - Inizio
  - Usare come: *ext4 con journaling*
  - Punto di mount: *“/”*
  - Opzioni:
    - *discard*
    - *noatime*
  - Etichetta: *linuxroot*
  - Flag avviabile: *disattivato*

SCSI3 (0,0,0) (sda) - 4.3 GB ATA VBOX HARDDISK

pri/log	4.3 GB	SPAZIO LIBERO
---------	--------	---------------

Usare come:	File system ext4 con journaling
Punto di mount:	/
Opzioni di mount:	discard,noatime
Etichetta:	linuxroot
Blocchi riservati:	5%
Utilizzo tipico:	standard
Flag avviabile:	disattivato



```
SCSI3 (0,0,0) (sda) - 4.3 GB ATA VBOX HARDDISK
n° 1 primaria 4.0 GB f ext4
pri/log 294.6 MB SPAZIO LIBERO
```

- Spazio libero 2:
  - Nuova partizione: 294.6 MB (*Tutto lo spazio rimanente*)
  - Primaria
  - Usare come: *area di swap*

```
Usare come:      area di swap
Flag avviabile:  disattivato
```

### 3.4 Installazione sistema base

- *linux-image-amd64*

```
linux-image-4.19.0-6-amd64
linux-image-amd64
```

### 3.5 Driver da includere

- *Generico*

```
generico: include tutti i driver disponibili
mirato: solo i driver necessari a questo sistema
```

### 3.6 Configuratore gestore pacchetti

- Software non libero: *No*
- Contrib: *Sì*
- Repository API: *No*

### 3.7 Selezione e installazione del software

- *Nessun aggiornamento*
- *No raccolta statistiche*

### 3.8 Installazione bootloader GRUB su disco fisso

- Installa
- Bootloader nel master: *Sì* → */dev/sda*
- Installazione GRUB: *No*

```
Inserire il device manualmente
/dev/sda (ata-VBOX_HARDDISK_VB13db82bc-98ffef10)
```

### 3.9 Note sull'installazione di Debian

- Il nome Debian nasce dall'unione del nome del suo fondatore Ian Murdock con quello della sua fidanzata Debra
- Viene utilizzato come dominio un sito .intra in quanto questo tipo di dominio non è ancora vendibile quindi non è possibile sia utilizzato da altri.
- Nella rete della scuola è presente un cacher, ovvero uno spazio di memoria dove vengono memorizzati i pacchetti che sono già stati scaricati, in modo da poterli distribuire nella rete in caso di installazioni multiple, senza appesantire il traffico di download.
- UTC sta per Coordinated Universal Time, ed è il fuso orario di riferimento a partire dal quale sono calcolati tutti gli altri fusi orari del mondo. L' UTC+1 è per noi l'ora invernale, mentre L'UTC+2 è l'ora estiva. Per l'estate nel Regno Unito si utilizza il BST, ovvero il British Summer Time.

## 4. Configurazione del client

- Avviare la macchina
- Accedere come utente di servizio (uds)
- Accedere come root
  - *su -*

```
uds@clientmilani:~$ su -  
Password:  
root@clientmilani:~#
```

### 4.1 Installazione software aggiuntivo:

- *apt install less joe tcpdump mtr-tiny cowsay*
- *apt install sudo*
- *apt clean* → Cancella la cache di installazione

### 4.2 Aggiunta uds al gruppo sudo

- *adduser uds sudo*

- Riavviare la macchina

### 4.3 Comandi utili:

- *id* → Per visualizzare in che utente sono
- *id uds* → Per visualizzare chi è uds
- *pwd* → Print Working Directory
- *df -h* → Visualizzare il File System
- *apt upgrade* → Aggiornamento che scarica il software aggiuntivo
- *apt update* → Scansiona e ricarica l'elenco dei software aggiuntivi
- *apt dist-update* → Aggiorna i pacchetti evitando o alleggerendo le intradipendenze che potrebbero portare al blocco dell'aggiornamento
- *Shutdown -h now* → Spegne il computer

## 4.4 Installazione GUI del client

- Accedere come uds
- Accedere come root

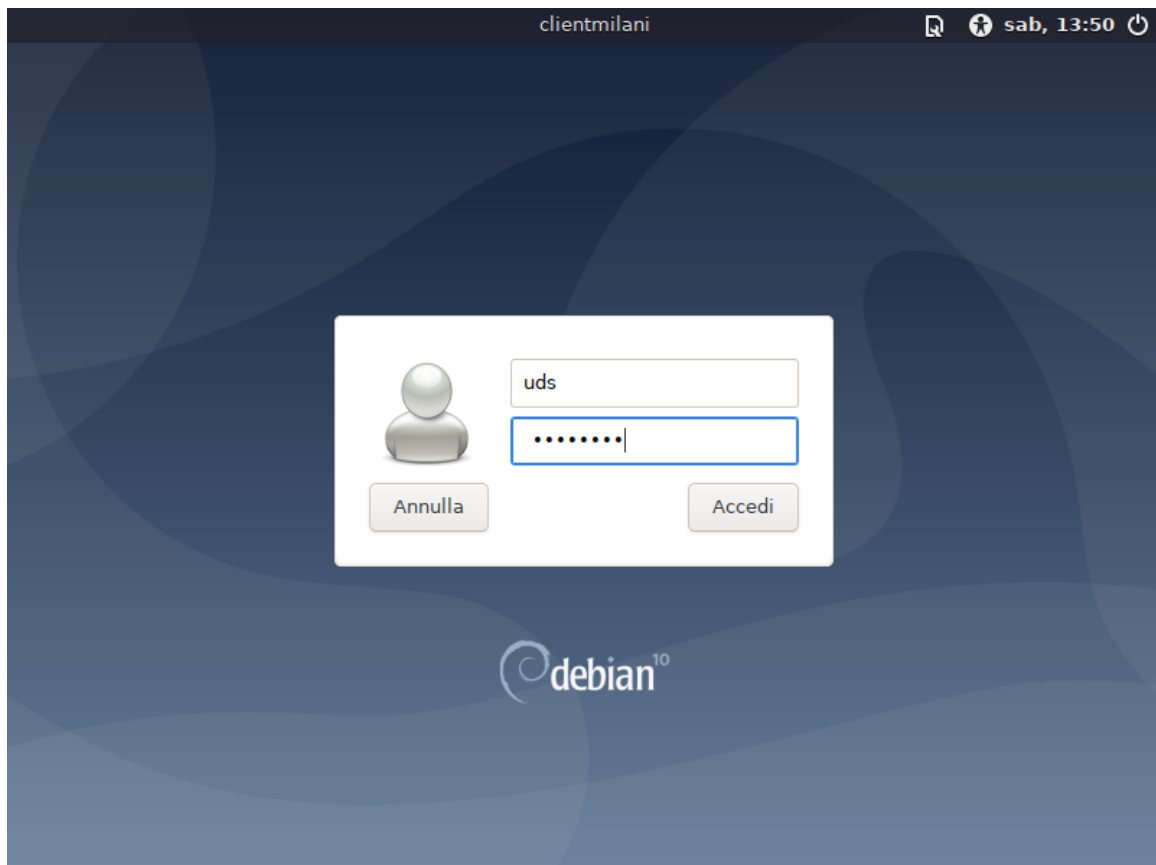
Installazione gestore login, windows manager e firefox

- *apt install light dm mate firefox*
- *apt clean*

Lightdm è il gestore del login grafico, mentre mate è il windows manager

- Riavviare i servizi
  - *cd /etc/init.d/*
  - *./lightdm status*
  - *./lightdm restart*

In questa maniera si riavvierà il gestore grafico, facendo quindi apparire il login grafico.

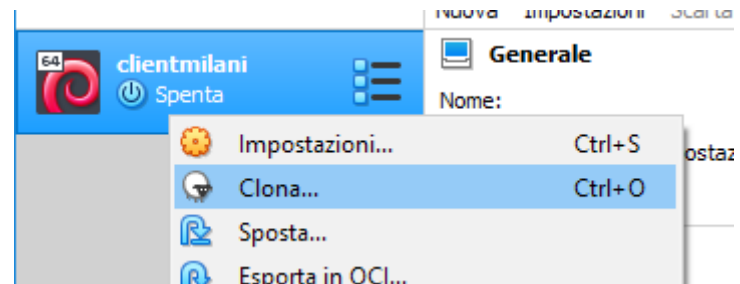


## 4.5 Note sulla configurazione del client

- E' opportuno lasciare inserito il CD di Debian anche dopo l'installazione, in quanto ha al suo interno altri eseguibili per l'installazione di programmi aggiuntivi. Per non avere problemi all'avvio è necessario spostare l'Hard Disk sopra al CD nella sequenza di avvio.
- Le directory sono determinate dal FHS, che sta per Filesystem Hierarchy Standard, ed è lo standard che definisce le directory principali ed il loro contenuto nel file system dei sistemi operativi Unix, tra cui i sistemi Linux.
- La storia tra Debian e Mozilla è controversa, infatti ci sono state vicende legali a causa del fatto che Debian utilizza solamente software libero. Il problema di Firefox stava nel fatto che l'applicazione in sé è libera, ma il logo è registrato, quindi andava contro le politiche di Debian.  
Per anni Debian ha quindi dovuto utilizzare IceWeasel, semplicemente Firefox con nome e logo diverso.  
La controversia si è risolta con la creazione di Firefox esr, che prevede l'assorbimento delle patch di sicurezza.

## 5. Creazione e configurazione del server

- Clonare il client
  - Nome: *servermilani*
  - Inizializzare nuovamente l'indirizzo MAC
  - Tipo: *Completa*



- Avviare il client
- Accedere come utente di servizio (uds)
- Accedere come root
  - *su -*
- Modificare il nome della macchina
  - *joe /etc/hosts*    `127.0.1.1    servermilani.milani.intra    servermilani`
    - *client.milani.intra* → *server.milani.intra*
    - *clientmilani* → *servermilani*
- Spegner la macchina

### 5.1 Note sulla configurazione del server

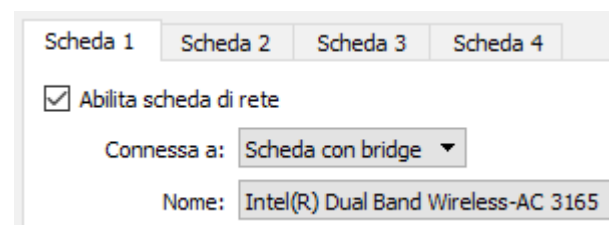
- Come icona della clonazione è raffigurata una pecora, in onore di Dolly, il primo mammifero ad essere stato clonato con successo da una cellula somatica.

## 6. Creazione e configurazione del router

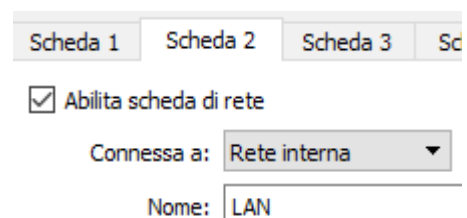
- Nuova macchina
  - Nome: *routermilani*
  - Tipo: *BSD*
  - Versione: *FreeBSD (32bit)*
  - Memoria RAM: *128 MB*
  - Disco Fisso
    - VDI
    - Statico
    - Memoria: *64 MB*
- Avviare la macchina
  - File ISO di m0n0wall → */home/itis/Internetfiles/monowall.iso*

### 6.1 Impostazioni schede di rete del router

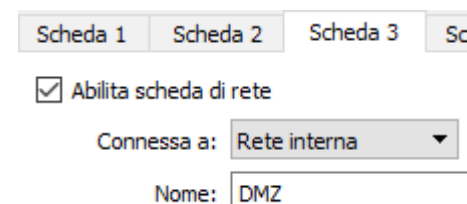
- Scheda 1:
  - Connessa a : *Scheda con bridge*



- Scheda 2:
  - Connessa a: *Rete Interna*
  - Nome: *LAN*

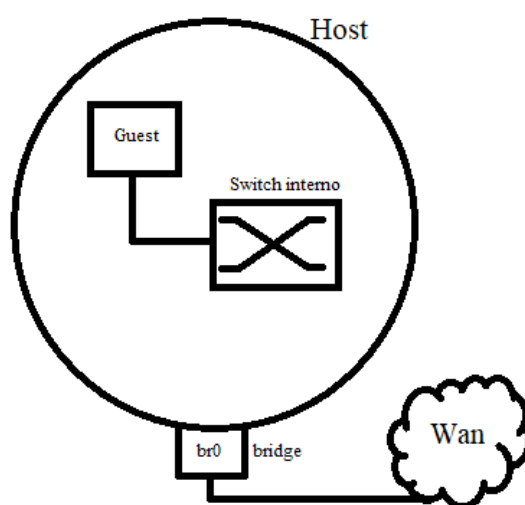


- Scheda 3:
  - Connessa a: *Rete Interna*
  - Nome: *DMZ*



## 6.3 Note sulla creazione e configurazione del router

- Nella rete interna viene creato uno switch virtuale, che permetterà la connessione tra i diversi guest, ma non la connessione verso l'esterno, in quanto non connesso.



- m0n0wall è un progetto ormai abbandonato, infatti l'ultima release risale al 15 gennaio 2014.

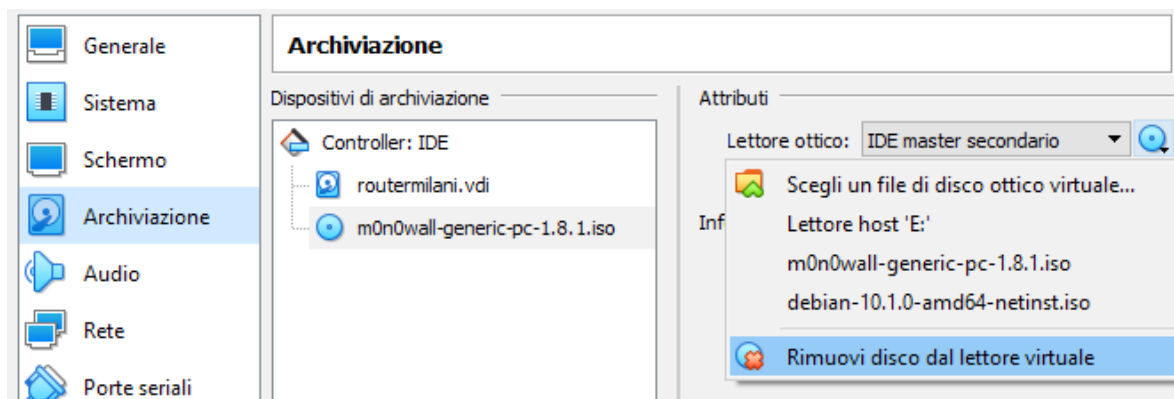


## 7. Installazione e configurazione di m0n0wall

```
m0n0wall console setup
*****
1) Interfaces: assign network ports
2) Set up LAN IP address
3) Reset webGUI password
4) Reset to factory defaults
5) Reboot system
6) Ping host
7) Install on Hard Drive
```

- Selezionare l'opzione 7 → *Install on Hard Drive*
- Spazio di memoria : *ad0*
- Attendere il riavvio

Dalle impostazioni rimuovere il disco di m0n0wall dal lettore virtuale



### 7.1 Ridenominazione schede di rete del router

- Avviare il router
- Selezionare l'opzione 1
  - Set up VLANs: *No*
  - Rinominare le schede nel seguente modo:

```
LAN -> em1
WAN -> em0
OPT1 -> em2
```

- Riavviare la macchina

## 8. Impostazioni di rete del client

- Scheda 1:
  - Connessa a: *Rete Interna*
  - Nome: *LAN*
- Accedere come uds
- Accedere come root
- Inserire il seguente comando
  - *ip addr* → Per controllare la propria connessione
- Lancio del DHCP manualmente
  - *dhclient enp0s3*

All'avvio di m0n0wall l'interfaccia WAN non presenta inizialmente un indirizzo IP, in quanto la WAN invia una richiesta DHCP a cui risponderà la mia infrastruttura. Basta aggiornare premendo il tasto invio e verrà così visualizzato l'indirizzo IP.

```
LAN IP address: 192.168.1.1  
WAN IP address: (unknown)
```

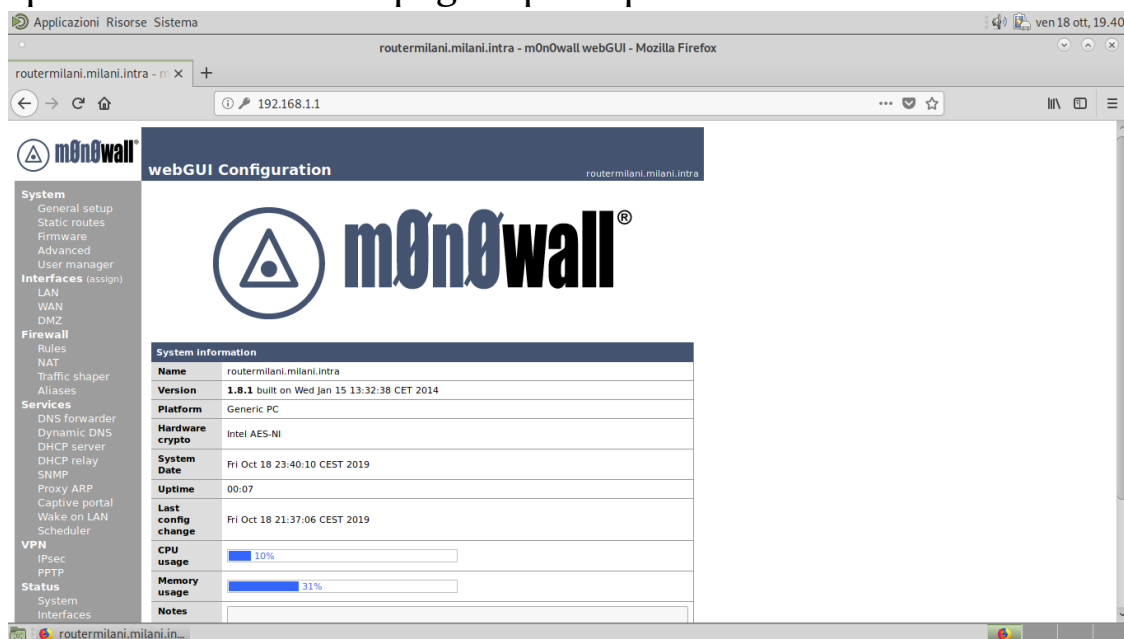


```
LAN IP address: 192.168.1.1  
WAN IP address: 192.168.1.219
```

## 9. Configurazione m0n0wall lato client

- Avviare Firefox
- Connettersi all'indirizzo 192.168.1.1
- Nome utente: *admin*
- Password: *mono*

Sarà quindi visualizzata la pagina principale di m0n0wall



- Firewall
  - Rules
  - Editare la riga presente su WAN
    - Rimuovere la spunta su *Block private networks*
    - Save
  - Creare una nuova regola su WAN
    - Source
      - Type: *Single Host*
      - Address: *172.30.4.1*
    - Destination port range
      - From: *HTTP*
      - To: *WAN*
- Apply changes

## 10. Configurazione m0n0wall dall'host

- Accesso tramite indirizzo WAN
- System
  - General setup
    - Hostname: *routermilani*
    - Domain: *milani.intra*
    - Username: *admin*
    - Password: *lasolita*
    - Time: *Europe/Rome*
- Save

### 10.1 Configurazione interfacce

- Interfaces: *WAN*
  - Hostname: *routermilani*
  - Description: accesso web al m0n0wall dal pc ospitante
- Interfaces: *OPT1*
  - Enable Optional 1 Interface
  - Description: *DMZ*
  - Bridge with: *none*
  - IP address: *192.168.101.1*

## 10.2 Configurazione regole DMZ










- Firewall
  - Rules
    - DMZ
      - Aggiungi nuova regola
      - Action: *Block*
      - Protocol: *Any*
      - Source: *DMZ Subnet*
      - Destination: *LAN Subnet*
      - Description: *Block: DMZ to LAN*
      - Save
      - Aggiungi nuova regola basata su quella appena creata
      - Action: *Pass*
      - Source: *DMZ*
      - Destination: *Any*
      - Protocol: *Any*
      - Description: *Allow: DMZ to ANY*
      - Save

## 11. Configurazione degli Aliases

Gli aliases sono una maniera comoda di ridenominazione degli indirizzi ip, in pratica è possibile sostituire gli ip con nomi a propria scelta. In questo modo, anche in caso di modifica degli indirizzi IP, sarà sufficiente cambiare una sola volta l'indirizzo, e tutti i campi collegati a quell'alias saranno aggiornati automaticamente

- Firewall
  - Aliases

### Firewall: Aliases

	Name	Address	Description	
<input type="checkbox"/>	host-client	192.168.31.100	l'ip del mio host in LAN	
<input type="checkbox"/>	host-pcospitante	172.30.4.1	Il computer da cui opero (windows)	
<input type="checkbox"/>	host-router-dmz	192.168.131.1	router monowall DMZ	
<input type="checkbox"/>	host-router-lan	192.168.31.1	router monowall LAN	
<input type="checkbox"/>	host-server	192.168.131.250	server in DMZ	
<input type="checkbox"/>	lan-labsistemi	172.30.4.0	la rete in cui appoggia la mia WAN	
<input type="checkbox"/>	server-aptcacher	172.30.1.199	Indirizzo del server apt-cacher-fermi.intra	
				 

## 12. Migrazione indirizzi IP

E' possibile che possa emergere la necessità di dover cambiare una serie di IP nella nostra rete contemporaneamente. Il rischio maggiore è quello di perdere l'accesso al router modificando gli IP in maniera errata.

Nel nostro caso, si deve migrare l'IP della rete LAN, da 192.168.1.0 a 192.168.31.0 .

Il numero 31 indica il numero di postazione nel laboratorio.

### 12.1 Migrazione indirizzo IP client

- Agire dalla modalità [root@client](#)
- Inserire il seguente comando:
  - *ifconfig enp0s3 **tempIP** netmask 255.255.255.0*
  - *route add default gw **newGW***
- Entrare nella configurazione web di m0n0wall
- Interfaces: LAN
  - IP: 192.168.31.1
- Services: DHCP server → LAN
  - Range: 192.168.31.100 to 192.168.31.199
- Riavviare il router
- Agire dalla modalità [uds@client](#)
- *sudo dhclient enp0s3* → Richiesta DHCP

In questo momento l'interfaccia di rete avrà due IP assegnati, per risolvere questo problema è necessario:

- Agire dalla modalità [root@client](#)
- *ifdown enp0s3* → disabilita interfaccia
- *ifup enp0s3* → abilita interfaccia

## 12.2 Migrazione indirizzo IP server

- Connettersi in ssh al server dal client
  - Inserire il comando: `ssh uds@192.168.101.1`
- Inserire il seguente comando:
  - `ifconfig enp0s3 templP netmask 255.255.255.0`
  - `route add default gw newGW`
- Entrare nella configurazione web di m0n0wall
- Interfaces: DMZ
  - IP: `192.168.131.1`
- Services: DHCP server → DMZ
  - Range: `192.168.131.100` to `192.168.131.199`
- In seguito si assegnerà al server l'indirizzo statico `192.168.131.250`
  - Inserire il comando:
    - `ifconfig enp0s3 192.168.131.250 netmask 255.255.255.0`
    - `route add default gw 192.168.131.1`
- Riavviare il router
- Agire dalla modalità [uds@server](#)
- `sudo dhclient enp0s3` → Richiesta DHCP

In questo momento l'interfaccia di rete avrà due IP assegnati, per risolvere questo problema è necessario:

- Agire dalla modalità [root@server](#)
- `ifdown enp0s3` → disabilita interfaccia
- `ifup enp0s3` → abilita interfaccia



## 13. Configurazione regole di firewall

Si procede ora alla configurazione delle regole di firewall per regolare gli accessi alle varie zone della rete.

### 13.1 Configurazione del firewall: LAN

#### Firewall: Rules

LAN

WAN

DMZ

	Proto	Source	Port	Destination	Port	Description
✗	TCP/UDP	LAN net	*	! host-router-lan	53 (DNS)	Block DNS request not to router
↑	*	LAN net	*	*	*	Default LAN to any

1. Blocco le richieste DNS dalla LAN che non sono indirizzate al router
2. La LAN può andare ovunque, come di Default

## 13.2 Configurazione del firewall: WAN

### Firewall: Rules

	LAN	WAN	DMZ
↑			
↑			
↑			
↑			

	Proto	Source	Port	Destination	Port	Description
↑	TCP	host-pcospitante	*	WAN address	80 (HTTP)	allow: accesso web al m0n0wall dal pc ospitante
↑	TCP	host-pcospitante	*	host-server	22 (SSH)	NAT Server in SSH
↑	ICMP	*	*	WAN address	*	Allow ping to WAN
↑	TCP	*	*	host-server	80 (HTTP)	Accesso al WebServer dalla porta 80

1. Permette l'accesso al m0n0wall dal pc ospitante in porta 80
2. Permette la connessione dalla WAN al Server in DMZ in SSH
3. Permette di eseguire il ping su tutti i dispositivi della WAN
4. Indirizza la porta 80 al WebServer ( Servirà in seguito con l'installazione di apache )

## 13.3 Configurazione del firewall: DMZ

### Firewall: Rules

LAN

WAN

DMZ

	Proto	Source	Port	Destination	Port	Description
↑	TCP	DMZ net	*	host-client	22 (SSH)	Allow: Server to LAN SSH via port 5022
✗	*	DMZ net	*	LAN net	*	Block: DMZ to LAN
↑	ICMP	DMZ net	*	*	*	Allow: Ping to any
↑	UDP	DMZ net	*	*	53 (DNS)	Allow: DNS to any
↑	UDP	DMZ net	*	*	123	Allow: NTP to any
↑	TCP	DMZ net	*	apt-cacher	3142	Allow: Only update from server apt-cacher
↑	*	DMZ net	*	*	*	Allow: DMZ to any (Normally disabled)

1. Permette l'SSH dal Server al Client in porta 5022
2. Blocca gli accessi dalla DMZ alla LAN
3. Permette il ping verso tutte le reti
4. Permette il DNS verso tutte le reti
5. Permette l'NTP verso tutte le reti
6. Permette l'update dei programmi attraverso l'apt-cacher della scuola
7. Permette gli accessi dalla DMZ a tutte le reti (Disabilitata)

## 14. Configurazione regole del NAT

### Firewall: NAT: Inbound

Inbound

Server NAT

1:1

Outbound

If	Proto	Ext. port range	NAT IP	Int. port range	Description
WAN	TCP	22 (SSH)	host-server	22 (SSH)	Server in SSH
DMZ	TCP	22 (SSH)	host-client	8888	Block: Server SSH in port 22
WAN	TCP	80 (HTTP)	host-server	80 (HTTP)	Reindirizzamento al server "Apache"
DMZ	TCP	5022	host-client	22 (SSH)	DMZ to LAN in SSH via port 22 or 5022
WAN	TCP	8080	host-router	80 (HTTP)	Reindirizzamento al router m0n0wall dalla porta 8080

1. Permette l'SSH verso il Server
2. Blocca l'SSH dal Server in porta 22 e lo indirizza verso una porta vuota per mandarlo in timeout
3. Regola che reindirizza la porta 80 al WebServer di Apache
4. Permette l'SSH dal Server alla LAN attraverso la porta 5022
5. Regola che reindirizza la richiesta alla porta 8080 alla configurazione del router m0n0wall

## 15. Installazione di Apache

Procederemo ora all'installazione di Apache sul server, che ci servirà per la creazione del WebServer.

### 1. Agire dalla modalità [uds@servermilani](mailto:uds@servermilani)

-Inserire i comandi: - *sudo apt update*  
(Per aggiornare i pacchetti contenuti nei repository)

- *sudo apt install apache2*

- *sudo service apache2 start*  
(Per accendere il WebServer di apache)

### 2. Modifica dell'index del WebServer:

Prima di procedere alla creazione di un nuovo index come pagina principale rinomino il file index.html già presente

- Inserire il comando *sudo mv index.html index.html1*

Ora creare un nuovo file html con la prima pagina del nostro WebServer

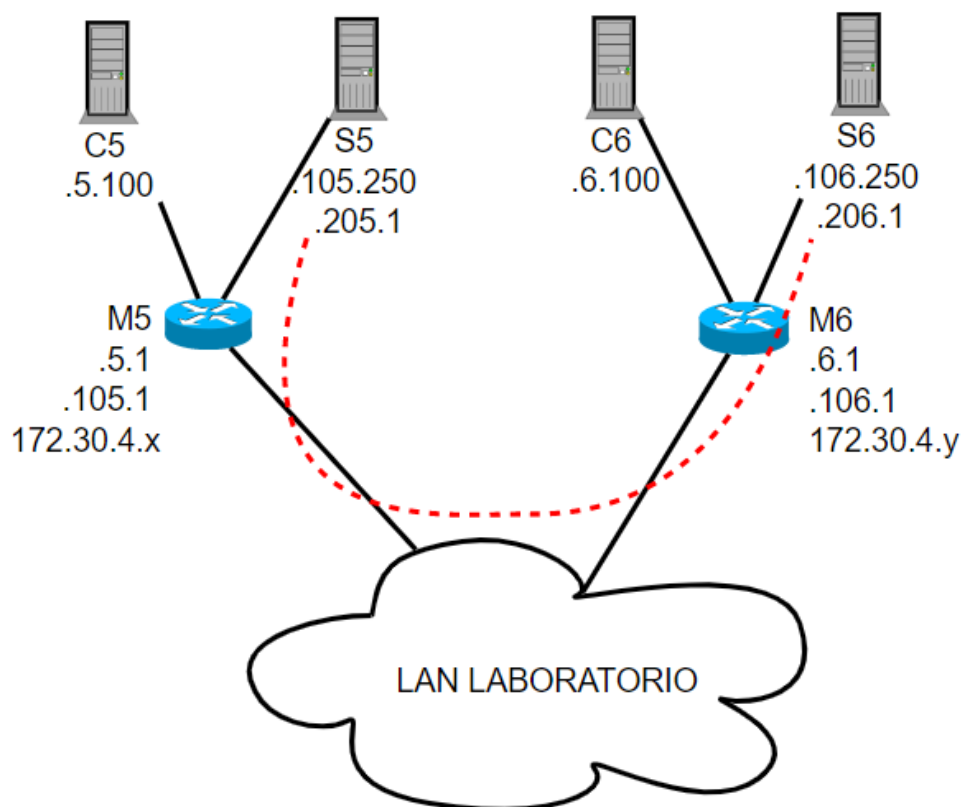
- Inserire il comando *sudo touch index.html*

## 16. Installazione e configurazione di una VPN

La configurazione di una VPN su m0n0wall creerà una interfaccia di rete in più. L'obiettivo è quello di riuscire a pingare dal server1 al server 2 e viceversa. Il server 1 sarà il nostro, mentre il server 2 sarà quello del nostro vicino di banco, nel mio caso Zen.

Le due interfacce dovranno avere un proprio indirizzo IP, che sarà dato dal  $192.168.200 + \text{numerohost}.1$ .

Nel nostro caso il mio indirizzo sarà  $192.168.231.1$ , mentre quello di Zen sarà  $192.168.202.1$ .



## 16.1 Configurazione OpenVPN

### 1. Agire dal server in modalità [uds@server](#)

- `sudo apt update`
- `sudo apt upgrade`
- `sudo apt install openvpn`

### 2. Generare una chiave che verrà condivisa con l'altro server

- `openvpn --genkey --secret nomechiave.key`  
(La chiave si troverà in `/etc/openvpn`)

### 3. Creazione del file di configurazione

- `nano -T 4 /etc/openvpn/nome.conf`

Inserire le seguenti righe:

- `dev tun10`
- `port 1194`
- `proto udp`
- `ifconfig 192.168.231.1 192.168.202.1`
- `remote 172.30.4.93`
- `secret /etc/openvpn/nomechiave.key`
- `log-append /var/log/openvpn-nome.log`
- `comp-lzo`
- `float`

4. La chiave si copierà nell'altro server attraverso il protocollo sftp. Per comodità useremo FileZilla

5. Nell'altro server andremo a creare un altro file di configurazione identico a quello del nostro server, in modo tale da poter permettere il ping ad entrambi.

6. Avviare OpenVPN

- `openvpn --config etc/openvpn/nome.conf --verb6`
- `/etc/init.d/openvpn start`
- `systemctl enable openvpn openvpn@serverconfig`

7. Test della VPN

Pingare dal server 1 al server 2 e viceversa.



## 17. Installazione e configurazione SNMP

Ora andremo ad installare e configurare il protocollo SNMP, utile per la gestione ed il monitoraggio dei dispositivi di rete.

### 17.1 Configurazione SNMP su m0n0wall

#### 1. Services → SNMP

Inserire i seguenti parametri

- System location: Bassano Del Grappa
- System Contact: ITIS E. Fermi

#### 2. Aggiungere una regola nel firewall WAN per l'interrogazione di SNMP

- Protocol= UDP
- Source= DMZ net
- Port= \*
- Destination= host-server
- Port= 161
- Description= Allow: SNMP connection

## 17.2 Configurazione MRTG

MRTG (Multi Router Traffic Grapher) è utile per avere una visualizzazione, basilare, dei grafici di rete.

### 1. Agire dalla modalità [uds@server](#)

- `sudo apt-get install snmp snmpd -y`
- `sudo apt-get install mrtg -y`

### 2. Avviare il servizio snmp

- `sudo systemctl enable snmpd`
- `sudo systemctl start snmpd`

### 3. Creare la cartella per mrtg

- `sudo mkdir /var/www/mrtg`

La cartella di default è posizionata in `/var/lib/mrtg`

Per permettere l'accesso alla cartella da parte dell'apache si creerà un link simbolico, che permetterà di lasciare la cartella nella posizione originale, quindi senza creare problemi ad eventuali file di configurazione, ma potrà essere visionata anche da un'altra posizione. Nel nostro caso il link simbolico verrà creato con la cartella `/var/www/mrtg`

- `cd /var/www/mrtg`
- `ln -s /var/lib/mrtg mrtg`

### 4. Generare il file di configurazione mrtg

- `cfgmaker public@localhost -output=/etc/mrtg.cfg`

### 5. Generare il file HTML per visualizzare i grafici

- `indexmaker -clicktext -title="Milani MRTG" --output=index.html /etc/mrtg.cfg`

## 17.3 Configurazione MRTG per grafici server

Agire dalla modalità [uds@server](#)

1. Inserire il comando *snmpconf* ed inserire le seguenti opzioni

- *snmp, snmpd* → *all*
- *snmp2.conf* → *2*
- *various* → *1*
- *disk usage* → *2*
- *mount point* → */*
- *minimum amount* → *100 000*
- *finished*
- *finished*
- *quit*

2. Inserire il comando *sudo nano /etc/snmp/snmpd.conf*

- Commentare la riga *-V systemonly*
- Decomentare la riga *rocommunity local*

3. Inserire i seguenti comandi

- *sudo systemctl restart snmpd*
- *sudo cfmaker [public@localhost](#) > /etc/mrtg\_server.cfg*
- *sudo indexmaker /etc/mrtg.cfg > /var/www/mrtg/index.html*

## 18. Installazione e configurazione di Cacti

Cacti è uno strumento di monitoraggio e rappresentazione grafica dei dispositivi di rete.

Useremo Cacti al posto di MRTG in quanto dispone di molte più opzioni e di un'interfaccia grafica migliore rispetto ai grafici MRTG

Con cacti andremo a tracciare:

- m0n0wall
- Server → Dischi, dispositivi di rete
- Apparecchiatura a scelta dell'istituto

### 18.1 Installazione sul server

Agire dalla modalità `uds@server`

1. Inserire i comandi:

- `sudo apt update`
- `sudo apt -y upgrade`
- `sudo apt install mariadb-server mariadb-client`

Agire dalla modalità [`root@server`](#)

2. Inserire il comando:

- `mysql_secure_installation`
- `password lasolita`

3. Inserire il comando:

- `sudo nano etc/mysql/mariadb.conf.d/50-server.cnf`

Modificare le righe:

- `character-set-server = utf8mb4`
- `collation-server = utf8mb4_unicode_ci`

#### 4. Inserire i comandi:

- `sudo systemctl restart mariadb`
- `sudo apt-get install php php-mysql php-snmp`
- `sudo apt-get install cacti snmp`

#### - Configure database for cacti with dbconfig-common?

- Yes
- password lasolita

#### - Credenziali:

- admin
- lasolita