

# PORTE e SOCKET

- Scaricate il file `testsocket.c`, compilatelo ed eseguitelo.  

```
# gcc -o testsocket testsocket.c  
# ./testsocket
```
- Molto probabilmente vi servirà il pacchetto `build-essential`:  

```
# sudo apt install build-essential
```
- Una volta capito il funzionamento potete usarlo per accedere ad altri servizi (tipo la posta) Come?... Spetta a voi capirlo.

```

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <unistd.h>

#include <sys/socket.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <arpa/inet.h>

void error(const char *m){
    perror(m);
    exit(0);
}

int main(void) {

    const char *s_addr = "157.138.7.88";
    const char i_port = 80;
    int s;
    struct sockaddr_in server_addr;
    char *message, server_reply[10000];

    printf("Create socket\n");

    s = socket(AF_INET, SOCK_STREAM, 0);
    if (s < 0) { error("socket()"); }

    memset(&server_addr, 0, sizeof(server_addr));
    server_addr.sin_family = AF_INET;
    server_addr.sin_addr.s_addr = inet_addr(s_addr);
    server_addr.sin_port = htons(i_port);
    printf("Trying connect\n");

```

```

    if (connect(s, (struct sockaddr
    *)&server_addr, sizeof(server_addr)) < 0) {
        error("connect()");
    }

    printf("Connect success!\n");

    /* Send some data */
    message = "GET /?st=1 HTTP/1.1\r\nHost:
    www.unive.it\r\n\r\n";
    if( send(s , message , strlen(message) , 0) <
    0)
    {
        error("Send failed");
    }
    printf("Data Send\n");

    /* Receive a reply from the server */
    if( recv(s, server_reply , 10000 , 0) < 0)
    {
        error("recv failed");
    }

    printf("Reply received\n");
    printf("%s",server_reply);

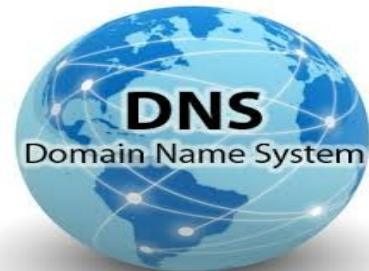
    close(s);
    return 0;
}

```

An orange circle with the text "DHCP server" in white, set within a light gray square frame.

**DHCP**  
server

# Servizi di Rete



# Operazioni periodiche (at)

- Provare il comando at:

```
# sudo apt install at
```

```
# at now
```

```
> touch /home/las/pippo.txt
```

```
> echo "coseacaso" > /home/las/pippo.txt
```

```
> PREMERE CTRL+D PER USCIRE
```

# Operazioni periodiche (crontab)

- Editate la crontab dell'utente:

```
# crontab -e
```

(Se vi chiede che editor usate e non sapete cosa fare scegliete nano).

- Inserite la seguente riga in fondo al file(sostituire las con il vostro user):

```
# m      h      dom      mon      dow      command
*/1      *      *      *      *      /bin/date >> /home/las/data.txt
```

```
* * * * * command(s)
^ ^ ^ ^ ^
| | | | |      allowed values
| | | | |      -----
| | | | ----- Day of week (0 - 7) (Sunday=0 or 7)
| | | ----- Month (1 - 12)
| | ----- Day of month (1 - 31)
| ----- Hour (0 - 23)
----- Minute (0 - 59)
```

- Per controllare che tutto funzioni:

```
# tail -f data.txt
ven 11 mar 2022, 10:00:01, CET
ven 11 mar 2022, 10:05:01, CET
```

# Servizi di base – installazione inetd

- **NB: per eseguire tutti questi comandi dovete essere root!!!**

```
# sudo -s
```

- Tra i servizi di base possiamo elencare ftp, tftp, telnet, rsh.
- In Debian/Ubuntu per installare il pacchetto basta lanciare il comando:

```
# apt install openbsd-inetd
```

- Il pacchetto è poi configurabile editando il file  
`/etc/inetd.conf`

```
# nano /etc/inetd.conf
```

- Provate ad attivare i servizi daytime, time ed echo:

```
time stream tcp nowait root internal
```

```
daytime stream tcp nowait root internal
```

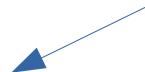
```
echo stream tcp nowait root internal
```

- Dopo di che li testate con:

```
# service inetd restart
```

```
# telnet localhost daytime
```

```
# telnet <ipvostravm> echo
```



Da echo si esce premendo  
ctrl+]

# Servizi di base – installazione inetd

- Installiamo ora anche il servizio di trasferimento file `ftp`:

```
# apt install ftpd
```

- ~~Editando il file `/etc/inetd.conf` dovrete trovare la riga di configurazione del servizio:~~

```
ftp stream tcp nowait root internal
```

- Dopo di che li testate con:

```
# service inetd restart
```

```
# ftp localhost ftp
```

```
# ftp <ipvostravm> ftp
```

# Servizi di base – installazione inetd

- Ora provate ad editare il file `hosts.deny` così:

```
# /etc/hosts.deny: list of hosts that are _not_  
# allowed to access the system. in.ftpd:ALL
```

```
ALL: ALL
```

- Piccolo test:

```
# tcpdchk -v
```

- Dopo di che riprovate a connettervi ai servizi echo, daytime, time ftp ecc...
- Se non dovesse più funzionare nulla... cosa fare?



# NTP – installazione e configurazione

- In debian/ubuntu esiste il pacchetto ntp:

```
# apt install ntp
```

```
# apt install ntpdate
```

```
# apt install ntpstat
```

- Per la configurazione è sufficiente editare il file

```
/etc/ntp.conf;
```

inserendo nella direttiva *server* il server **ntp** di riferimento:

```
server ntp.dsi.unive.it
```

- A casa vostra, se volete:

```
server ntp1.inrim.it
```

# Altri comandi per la gestione del clock

- **rdate**: permette la sincronizzazione con un server che abbia a disposizione il servizio time:

```
# apt install rdate  
# rdate -s ntp1.inrim.it
```

- Avendo attivato il servizio time sulla vm possiamo fare:

```
# rdate -s 192.168.100.xx
```

- **hwclock**: gestisce il clock hardware del pc.

```
# hwclock -w //scrive l'ora del so sul bios
```

**NB**: Sono comandi di amministrazione... quindi funzionano solo come se siete **super user (root)** o tramite i comandi **sudo** o **su!!!**

# DHCP - installazione

- Affinchè la nuova scheda venga vista è necessario riavviare la vm.
- Per trovare la scheda al riavvio:

```
# dmesg | grep ens
```

- Dopo l'avvio configurate la scheda con netplan o al volo usando il comando ifconfig:

```
# ifconfig ens38 192.168.200.1 netmask 255.255.255.0 up
```

- In debian/ubuntu si installa con i comandi:

```
# apt install isc-dhcp-server
```

```
# apt install isc-dhcp-client
```

- Editate il file `/etc/dhcp/dhcpd.conf` come segue:

# DHCP- Esempio: file /etc/dhcp/dhcpd.conf

```
option domain-name "<vostrogruppo>.localdomain";
option domain-name-servers 8.8.8.8, 8.8.4.4;
default-lease-time 600;
max-lease-time 7200;

subnet 192.168.200.0 netmask 255.255.255.0 {
range 192.168.200.100 192.168.200.200;
option domain-name-servers      8.8.8.8, 8.8.4.4, 192.168.200.1;
option ntp-servers 192.168.200.1;
option routers 192.168.200.1;
option subnet-mask 255.255.255.0;
}
```

# DHCP - installazione

- Riavviate il servizio:

```
# service isc-dhcp-server restart
```

```
# systemctl restart isc-dhcp-server
```

- Controllate l'esito tramite il comando :

```
# journalctl -xe
```

- Come faccio a sapere se funziona?
- Dovete creare una seconda vm e agganciarla alla stessa rete del vostro server dhcp (VMnet2).
- Provate a farlo da soli.

# Installazione DNS

- **NB: nelle seguenti slide sostituite 192.168.64.129 con l'ip della vostra macchina virtuale o del portatile hp (192.168.100.xx) e la rete 192.168.64.0/24 con la rete di cui fa parte la vostra vm o la rete 192.168.100.0/24 .**
- Colleghiamoci via ssh alla VM:  

```
# ssh <utente>@192.168.64.129
```
- Diventiamo root:  

```
# sudo -s
```
- Installiamo i pacchetti necessari per bind:  

```
# apt install bind9 bind9utils bind9-doc
```
- Prima di tutto, editiamo il file che indica a bind la configurazione di partenza, forzando il solo uso a ipv4 (se non esiste createlo):  

```
# nano /etc/default/bind9
```

- Modificare(o aggiungere) la riga `OPTIONS="-u bind"` in `OPTIONS="-u bind -4"`
- Riavviamo bind:  

```
# service bind9 restart (systemctl restart bind9)
```

# Installazione DNS

- Cominciamo con la configurazione di bind. Prima di tutto dobbiamo specificare che opzioni di bind vogliamo attive. Editiamo quindi il file `/etc/bind/named.conf.options`:

```
# nano /etc/bind/named.conf.options
```

- Sostituiamo il contenuto con:

```
options {  
    directory "/var/cache/bind";  
    recursion yes; # abilita la ricorsione  
    allow-recursion {127.0.0.0/8; 192.168.64.0/24; 192.168.200.0/24;};  
    listen-on { 127.0.0.1; 192.168.64.129; 192.168.200.1;}; # indirizzi ip su cui  
restare in ascolto  
    allow-transfer { none; };      # disabilita il trasferimento dai fra le zone  
    allow-query-cache { 127.0.0.0/8; 192.168.64.0/24; 192.168.200.0/24; };  
    allow-query { 127.0.0.0/8; 192.168.64.0/24; 192.168.200.0/24; };  
    forwarders {  
        8.8.8.8;  
        8.8.4.4;  
    };  
};
```

# Installazione DNS

- Editiamo ora il file `/etc/bind/named.conf.local`:

```
# nano /etc/bind/named.conf.local
```

- Sostituiamo il contenuto con:

```
Zone "<vostrogruppo>.localdomain" {  
    type master;  
    file "/etc/bind/zones/db.<vostrogruppo>.localdomain"; # file della zona  
    //allow-transfer { 192.168.64.130; }; # eventuale dns secondario  
};  
  
Zone "64.168.192.in-addr.arpa" {  
    type master;  
    file "/etc/bind/zones/db.64.168.192"; # 192.168.64.0/24  
    //allow-transfer { 192.168.64.130; }; # eventuale dns secondario  
};
```



# Installazione DNS

- Procediamo con la creazione dei file che descrivono la definizione della zona.
- Se non esiste creiamo la directory delle zone:

```
# mkdir -p /etc/bind/zones/
```

- Definiamo la zona diretta creando il file  
/etc/bind/zones/db.<vostrogruppo>.localdomain:

```
# nano  
/etc/bind/zones/db.<vostrogruppo>.localdomain
```

# Installazione DNS

- Inseriamo il contenuto del file `dnsdiretto.txt` che trovate su moodle:

```
$TTL      604800
$ORIGIN .
@         IN      SOA      <vostropc>.<nomegruppo>.localdomain.
admin.<nomegruppo>.localdomain. (
                                2024032601 ; seriale, indica la versione, di solito si mette la
data di modifica in formato aaaa/mm/gg
                                ; più un incrementale
                                604800      ; Refresh
                                86400       ; Retry
                                2419200    ; Expire
                                604800 )    ; Negative Cache TTL

; name servers - NS records
                IN      NS      <vostropc>.<nomegruppo>.localdomain.

; name servers - A records
<vostropc>      A        192.168.64.129
```

# Installazione DNS

- Procediamo con la creazione dei file che descrivono la definizione della zona.
- Se non esiste creiamo la directory delle zone:  

```
# mkdir -p /etc/bind/zones/
```
- Definiamo la zona inversa creando il file  
`/etc/bind/zones/db.64.168.192:`
- ```
# nano /etc/bind/zones/db.64.168.192
```

# Installazione DNS

- Inseriamo il contenuto del file `dnsinverso.txt` che trovate su moodle:

```
$ORIGIN .
$TTL      604800
@         IN      SOA      <vostropc>.<nomegruppo>.localdomain. Admin.<nomegruppo>.localdomain. (
                                2024032601      ; seriale, indica la versione, di solito si
                                ; mette
                                ; la data di modifica in formato aaaa/mm/gg
                                ; più un incrementale
                                604800      ; Refresh
                                86400      ; Retry
                                2419200    ; Expire
                                604800 )    ; Negative Cache TTL

; name servers - NS records
                IN      NS      <vostropc>.<nomegruppo>.localdomain.

; PTR Records
xx            PTR      <vostropc>.<nomegruppo>.localdomain.
```

# Installazione DNS – solo se avete fatto dhcp

- Inseriamo il contenuto del file `dnsinversodhcp.txt` che trovate su moodle nel file `zones/db.192.168.200:`

```
$TTL      604800
@         IN      SOA      <vostropc>.<nomegruppo>.localdomain. admin..<nomegruppo>.localdomain. (
                                2024032601      ; seriale, indica la versione, di solito si
                                ; mette
                                ; la data di modifica in formato aaas/mm/gg
                                ; più un incrementale
                                604800      ; Refresh
                                86400      ; Retry
                                2419200    ; Expire
                                604800 )    ; Negative Cache TTL

; name servers - NS records
        NS      <vostropc>.<nomegruppo>.localdomain.

; PTR Records
1       PTR      <vostropc>.<nomegruppo>.localdomain.
129     PTR      test01.localdomain.
130     PTR      test02.localdomain.
```

# Installazione DNS

- Testiamo ora le varie configurazioni fatte.
- Anzitutto vediamo se bind è configurato correttamente:

```
# named-checkconf
```

- Se il comando non ritorna nulla, la configurazione è ok

- Poi controlliamo se la definizione delle zone è corretta. Iniziamo dalla zona diretta:

```
# named-checkzone localdomain.it /etc/bind/zones/db.localdomain
```

# Installazione DNS

- Controlliamo ora la zona inversa:

```
# named-checkzone <vostrarete> /etc/bind/zones/db.192.168.64
```

```
# named-checkzone <vostrarete> /etc/bind/zones/db.192.168.200
```

- Se i comandi precedenti non hanno dato errori, possiamo procedere con l'attivare il nostro dns:

```
# systemctl restart bind9 (o service bind9  
restart)
```

```
# ufw allow bind9
```

# Installazione DNS

- Infine un po di prove (ipvostravm=ipportatilehp):

```
# host <vostropc>.<nomegruppo>.localdomain. 127.0.0.1
```

```
# host <vostropc>..<nomegruppo>.localdomain.  
<ipvostropc>
```

```
# host www.unive.it <ipvostropc>
```

```
# host www.google.it <ipvostropc>
```