

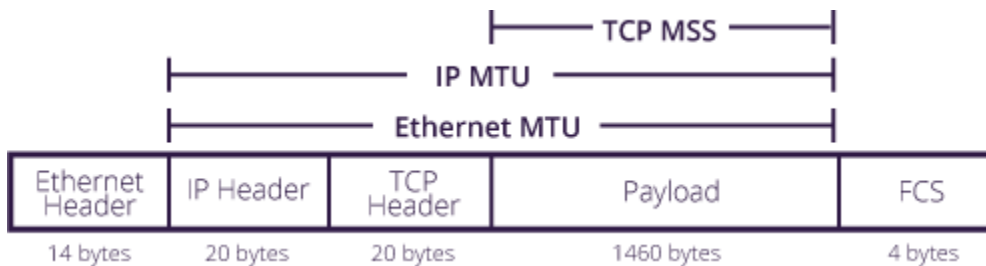
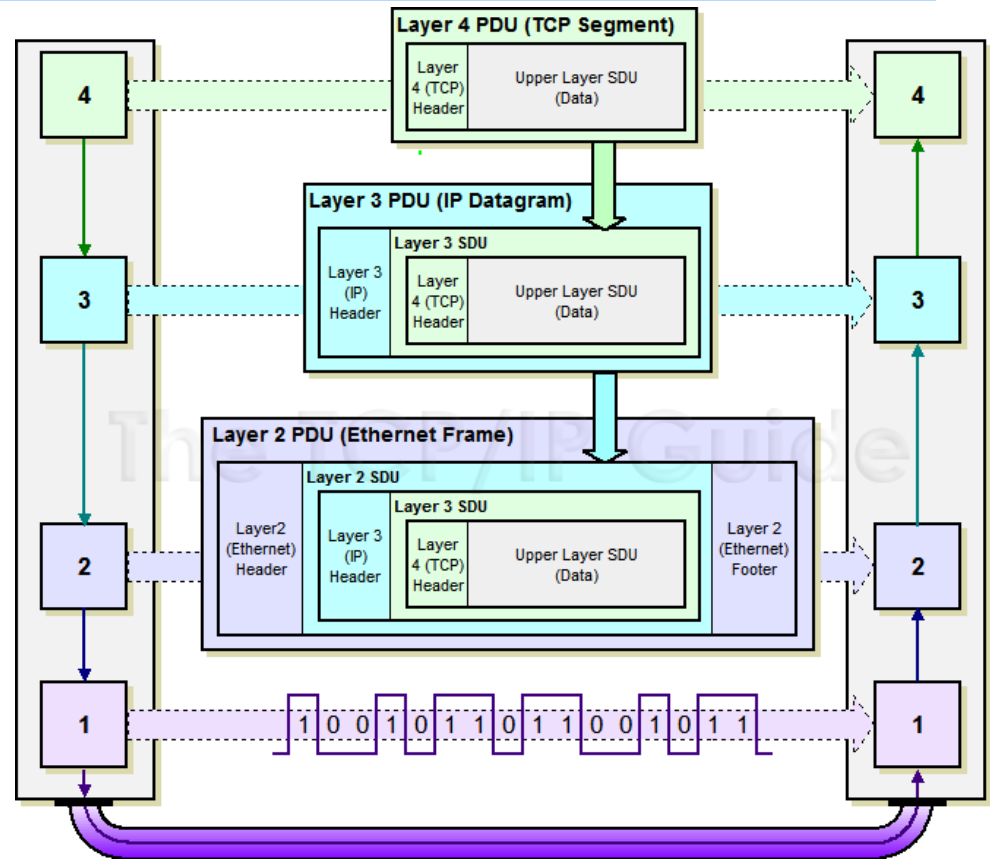
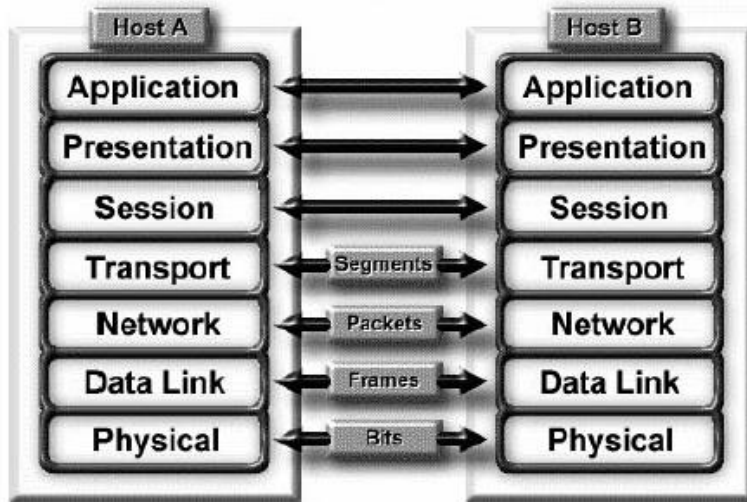
WORKSHOP PEMROGRAMAN JARINGAN

Mochammad Zen Samsono Hadi, ST. MSc. Ph.D

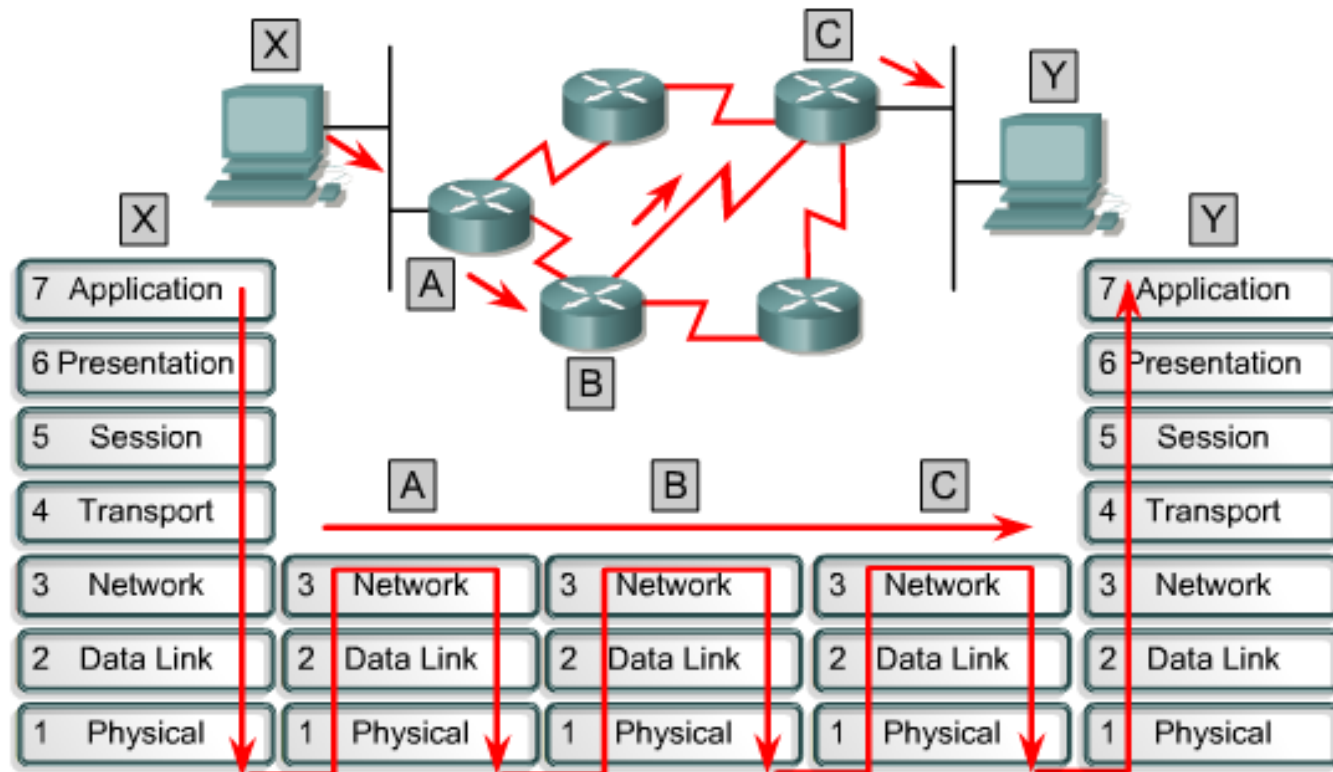
TOPIK PEMBAHASAN

- Mencetak nama komputer dan alamat IPv4
- Mendapatkan alamat IP secara remote
- Merubah alamat IPv4 ke format lainnya
- Mendapatkan nama service, port dan protokol yang digunakan pada suatu komputer

Layering Pada OSI



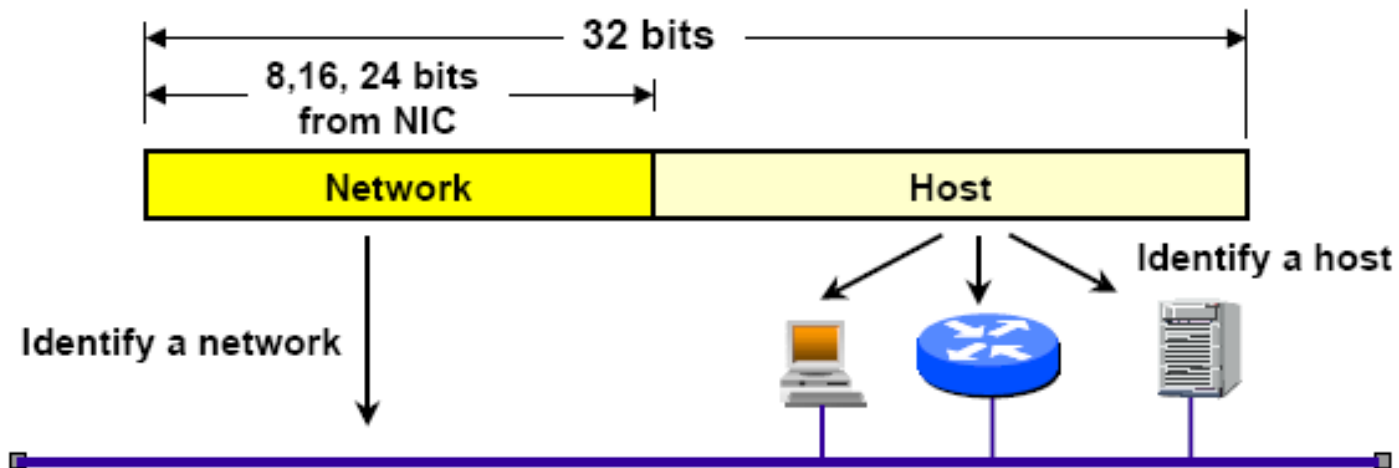
Packet Propagation



Each router provides its services to support upper-layer functions.

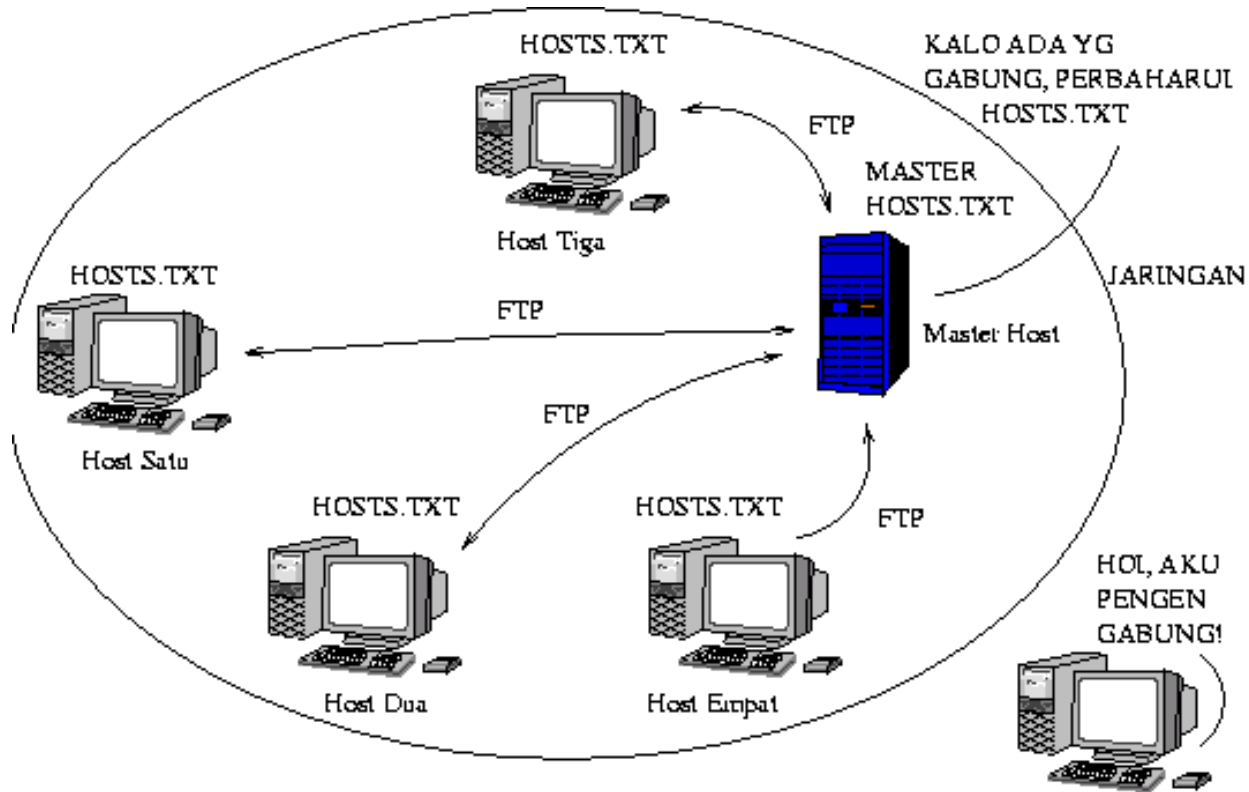
Alamat IPv4

IP Addressing



Penamaan Device

- Sebelum adanya DNS, tahun 1970-an ARPAnet menggunakan pemetaan dengan bentuk tabel host pada berkas HOSTS.TXT.



Mencetak nama komputer dan alamat IPv4

- Listing 1.1

Library python utk pemrograman jaringan

```
import socket
```

Modul dari library socket

```
def print_machine_info():
```

```
    host_name = socket.gethostname()
```

```
    ip_address = socket.gethostbyname(host_name)
```

```
    print ("Host name: %s" %host_name)
```

```
    print ("IP address: %s" %ip_address)
```

```
if __name__ == '__main__':
```

```
    print_machine_info()
```

```
$ python 1_1_local_machine_info.py
```

```
Host name: llovizna
```

```
IP address: 127.0.1.1
```

Socket library

- A socket is an abstract representation of a communication endpoint.
- <http://docs.python.org/3/library/socket.html>

```
gethostname(...)
```

```
gethostname() -> string
```

```
Return the current host name.
```

```
gethostbyname(...)
```

```
gethostbyname(host) -> address
```

```
Return the IP address (a string of the form  
'255.255.255.255') for a host.
```

Network API



Mendapatkan alamat IP remote

```
import socket
def get_remote_machine_info():
    remote_host = 'www.python.org'
    try:
        print ("IP address of %s: %s" %(remote_host,
            socket.gethostbyname(remote_host)))
    except socket.error as err_msg:
        print ("%s: %s" %(remote_host, err_msg))
if __name__ == '__main__':
    get_remote_machine_info()
```

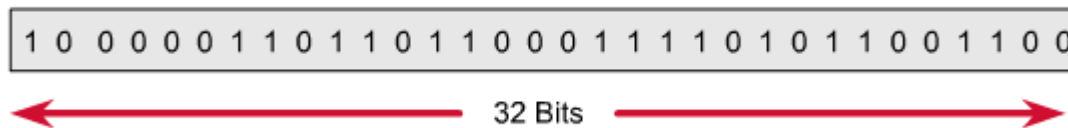
```
$ python 1_2_remote_machine_info.py
IP address of www.python.org: 151.101.36.223
```

```
$ python 1_2_remote_machine_info.py
www.pytgo.org: [Errno -2] Name or service not known
```

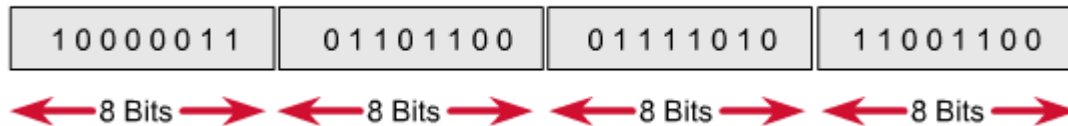
Merubah alamat IPv4 ke format lain

IP Address terdiri 32 bits.

Terbagi menjadi dua bagian Bagian networkID dan Bagian HostID, hal ini tergantung dari subnetmask (akan dibicarakan lebih lanjut).



32 bit dibagi menjadi 4 bagian setiap bagian terdiri dari 8 bit.



Untuk kemudahan dikonversi menjadi desimal.



199

Inet function

- `inet_aton(ip_address)`
 - The `ip_address`, which needs to be converted from the dotted-quad, string format to 32 bit packed binary format.
- `inet_ntoa(ip_address)`
 - `ip_address` – The address in 32-bit packed format which is to be converted
- `Hexlify(data)`
 - Return the hexadecimal representation of the binary *data*. Every byte of *data* is converted into the corresponding 2-digit hex representation.

Merubah alamat IPv4 ke format lain

```
import socket
from binascii import hexlify
def convert_ip4_address():
    for ip_addr in ['127.0.0.1', '192.168.0.1']:
        packed_ip_addr = socket.
            inet_aton(ip_addr)
        unpacked_ip_addr = socket.inet_ntoa
            (packed_ip_addr)
        print ("IP Address: %s => Packed: %s,
            Unpacked: %s" %(ip_addr,
                hexlify(packed_ip_addr),
                unpacked_ip_addr))
if __name__ == '__main__':
    convert_ip4_address()
```

```
$ python 1_3_ip4_address_conversion.py
```

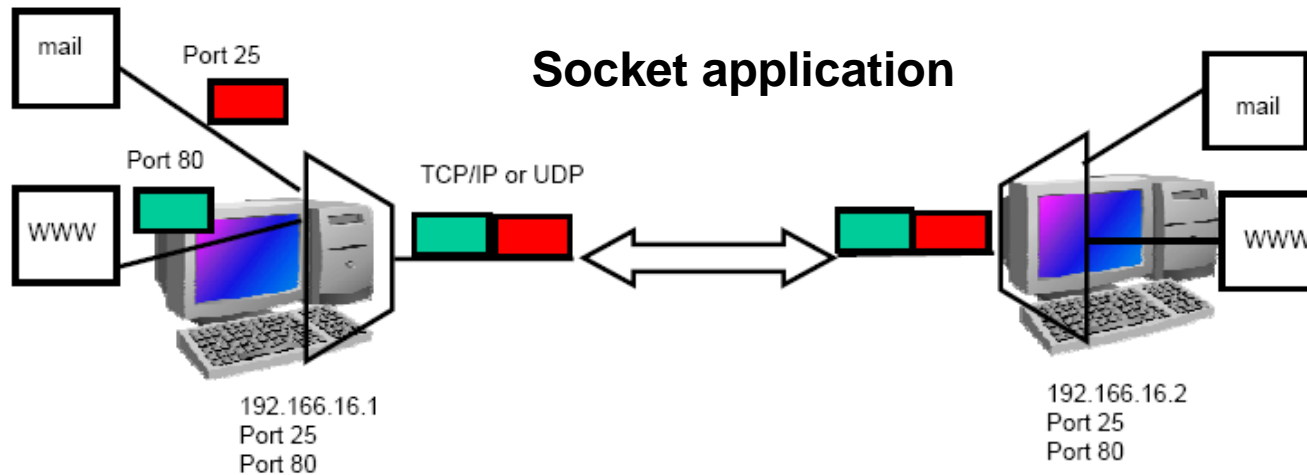
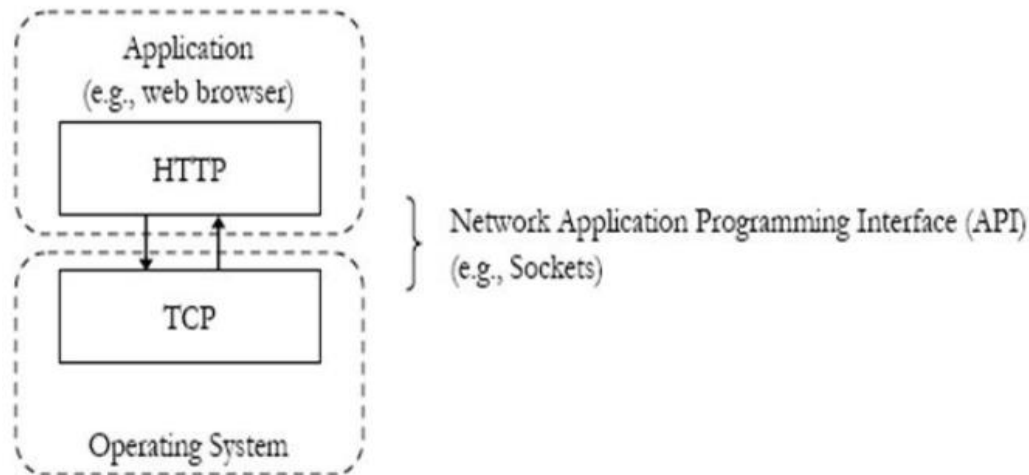
```
IP Address: 127.0.0.1 => Packed: 7f000001, Unpacked:
```

```
127.0.0.1
```

```
IP Address: 192.168.0.1 => Packed: c0a80001, Unpacked: 192.168.0.1
```

Mendapatkan nama service, port dan protokol

Network API



- TCP / IP and UDP are multiplexing different applications on one IP link

Mendapatkan nama service, port dan protokol

```
import socket

def find_service_name():
    protocolname = 'tcp'
    for port in [80, 25]:
        print ("Port: %s => service name: %s" %(port,
socket.getservbyport(port, protocolname)))
    print ("Port: %s => service name: %s" %(53, socket.getservbyport(53,
'udp'))))
if __name__ == '__main__':
    find_service_name()
```

```
$ python 1_4_finding_service_name.py
Port: 80 => service name: http
Port: 25 => service name: smtp
Port: 53 => service name: domain
```

TUGAS

- Buatlah program di python dengan menu pilihan sebagai berikut (buat dalam bentuk fungsi):

MENU PILIHAN:

1. MENGETAHUI SERVICE DAN PROTOKOL PADA JARINGAN

2. MENGETAHUI ALAMAT IP DARI SEBUAH WEBSITE

Jika pilihan 1:

INPUT:

Masukkan port protokol: **23**

OUTPUT:

Port: 23 => service name: **telnet**

Jika pilihan 2:

INPUT:

Masukkan alamat web: **www.google.com**

OUTPUT:

Anda mengakses **www.google.com** dengan alamat IP **74.125.24.105**
dari komputer **AISHA-PC** dengan alamat IP **10.252.23.123**

Setelah itu, muncul konfirmasi:

ANDA INGIN MENGULANG (Y/T)?