

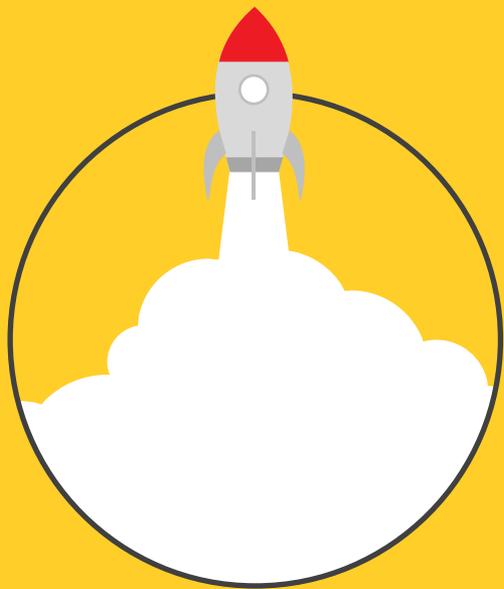
LRODRIGO



105568 - TECNOLOGIA DE REDES

AULA 0 - APRESENTAÇÃO DO CURSO

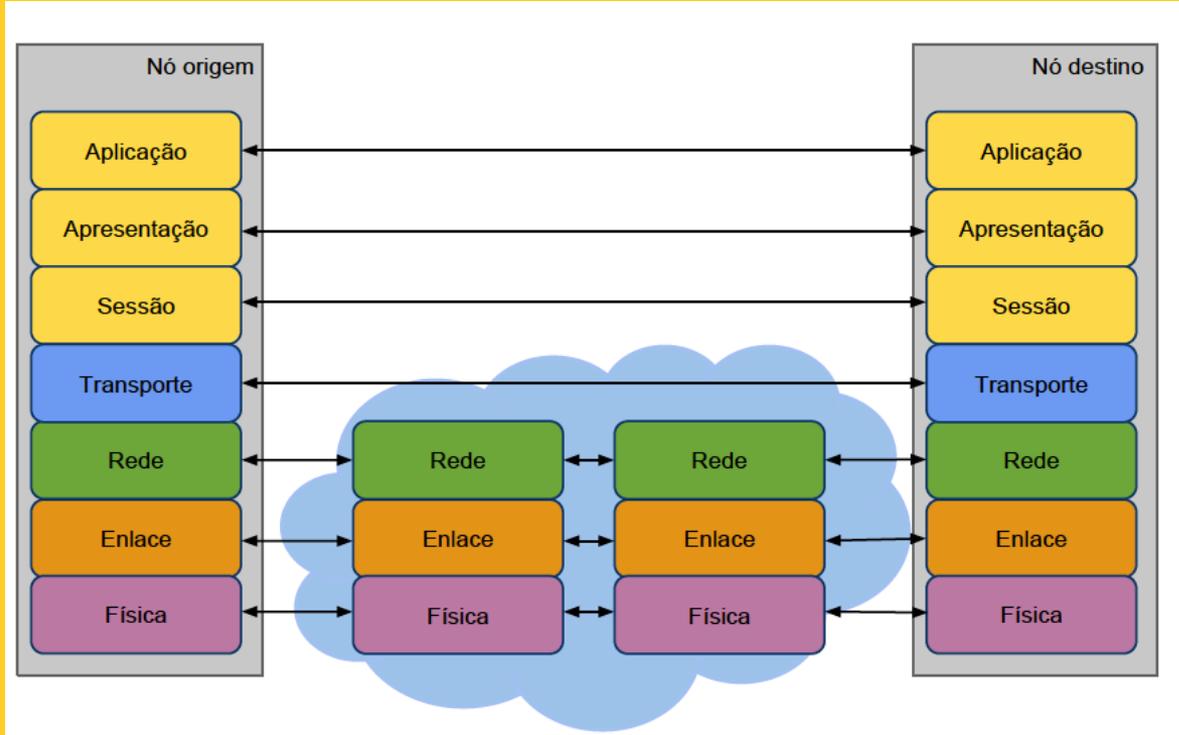
Luis Rodrigo - luis.goncalves@ucp.br - <http://lrodrigo.sgs.Incc.br>



TECNOLOGIA DE REDES

REVISÃO – RM/OSI E TCP/IP

RM/OSI E TCP/IP



RM/OSI

TCP/IP



Modelo de Referência OSI



Modelo de Referência TCP/IP



Pilha de Protocolos da Internet

TCP/IP: IPV4



An IPv4 address (dotted-decimal notation)

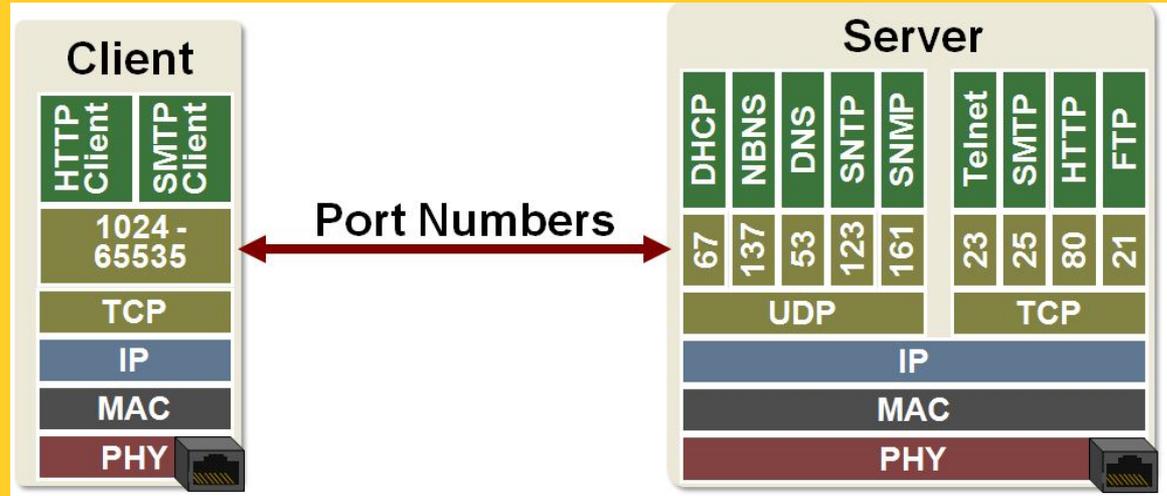
172 . 16 . 254 . 1

↓ ↓ ↓ ↓
10101100 . 00010000 . 11111110 . 00000001

1 byte = 8 bits

32 bits (4 x 8), or 4 bytes

TCP/IP:PORTAS



PROCESSO DE ENCAPSULAMENTO



Emissor

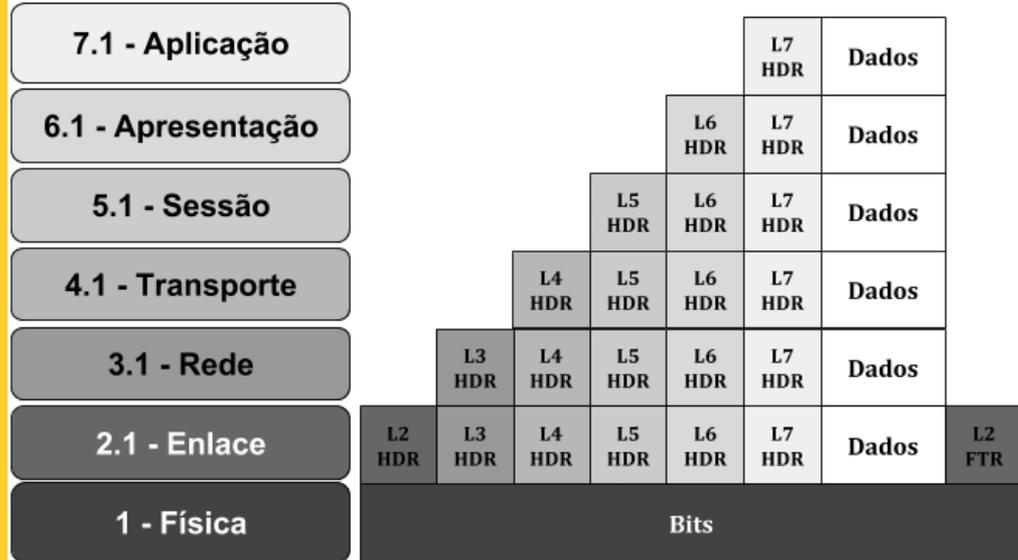


Dados

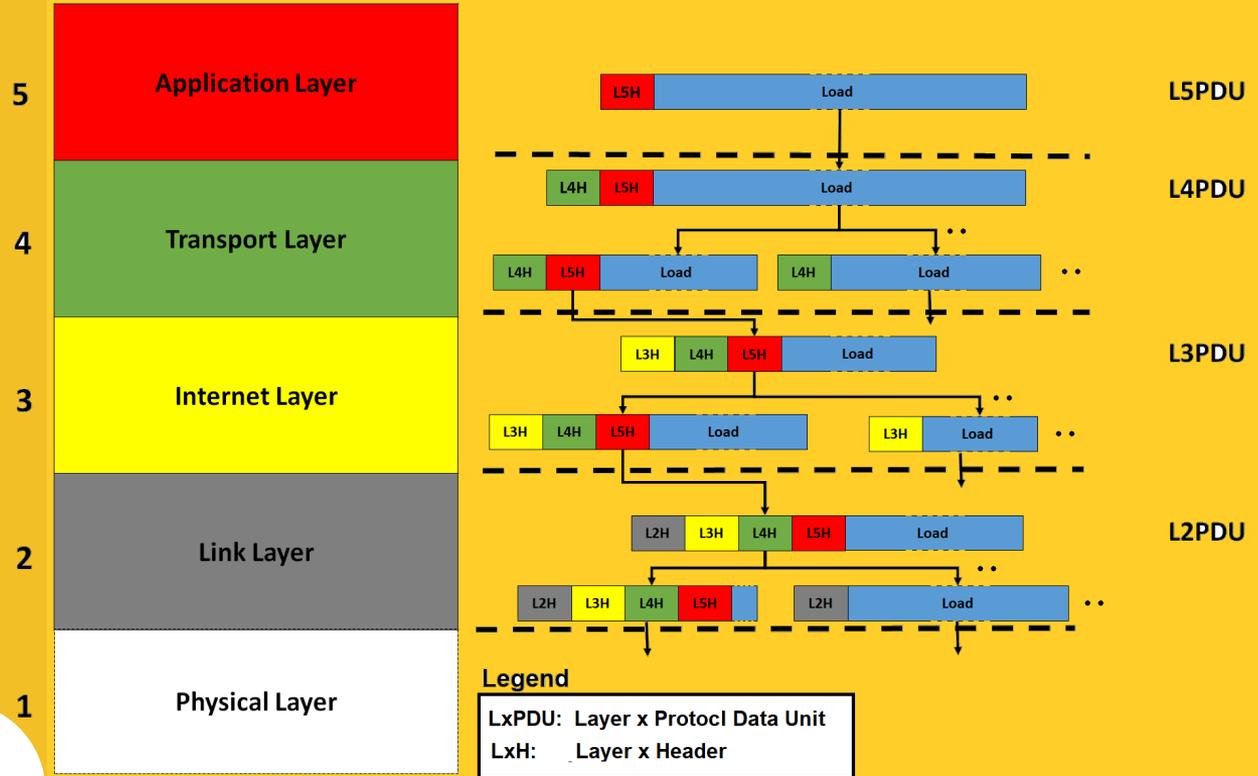
HDR = *Header* ou Cabeçalho

L = *Layer* ou Camada

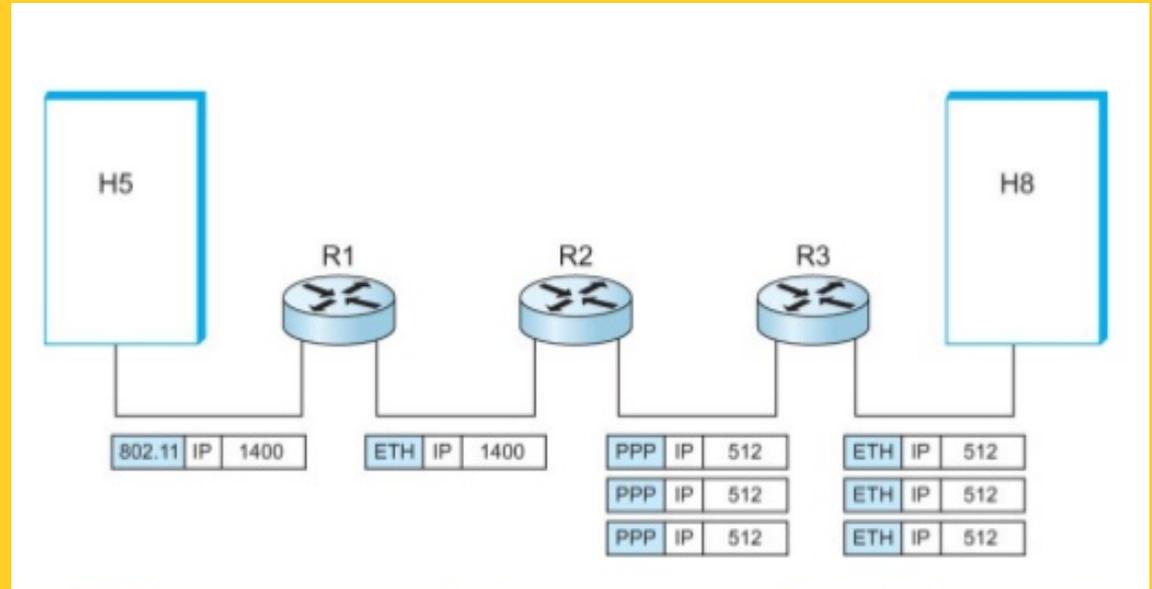
FTR = *Footer* ou Rodapé

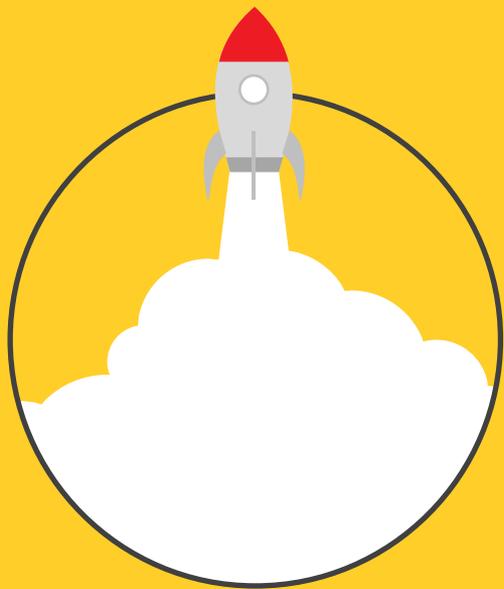


PROCESSO DE ENCAPSULAMENTO



PROCESSO DE FRAGMENTAÇÃO



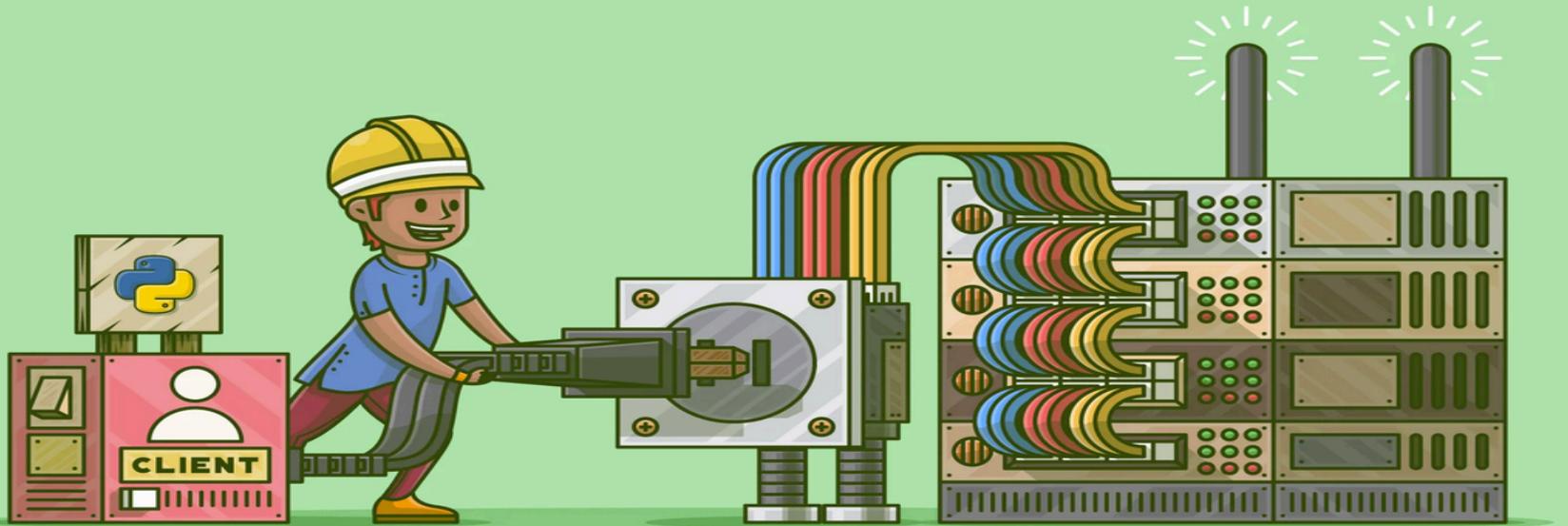


TECNOLOGIA DE REDES

REVISÃO – SOCKETS

SOCKETS

Sockets são os “endpoints”
de uma comunicação bilateral



SOCKETS



- São os “endpoints” de uma comunicação bilateral;
- Permitem a comunicação entre processos (IPC)
- Modelo Cliente x Servidor
- **Domínios:** AF_INET e AF_UNIX

SOCKETS



Domínio: **AF_INET**

- **IPv4** (todas as interfaces inclusive LoopBack)
- Ranges de **Portas**, definidos pela IANA

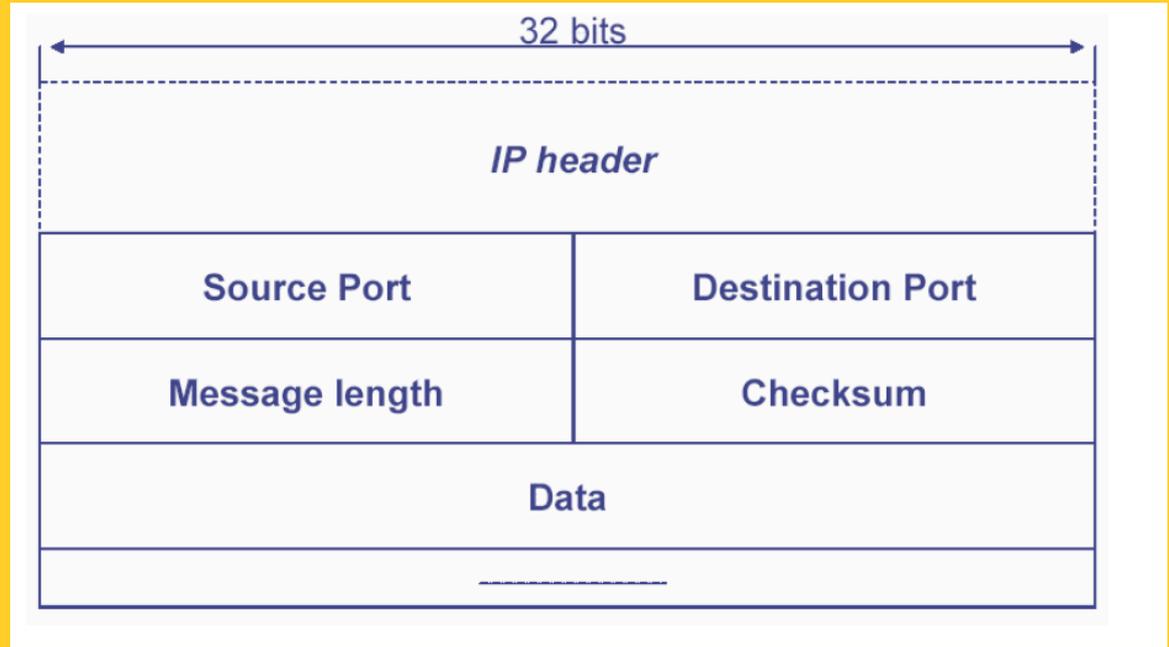
SOCKETS



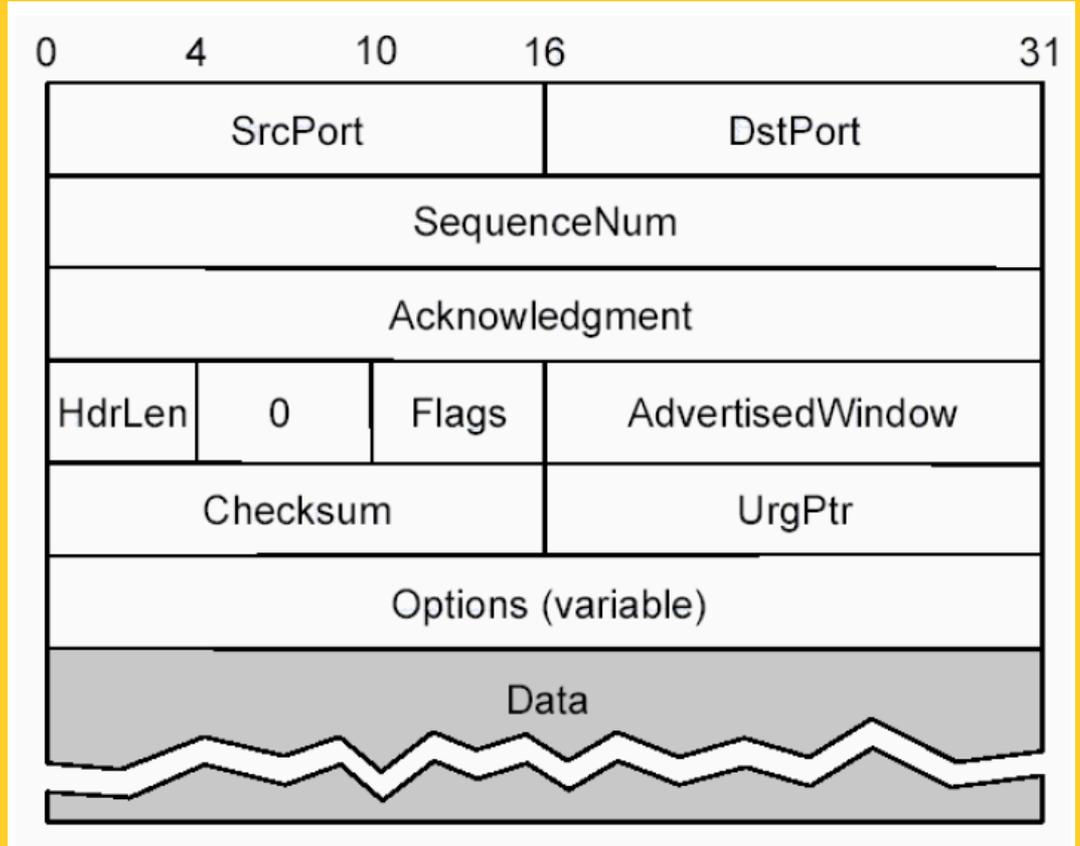
Domínio: `AF_INET`

- `raw_sockets`
- `datagram_sockets`
- `stream_sockets`

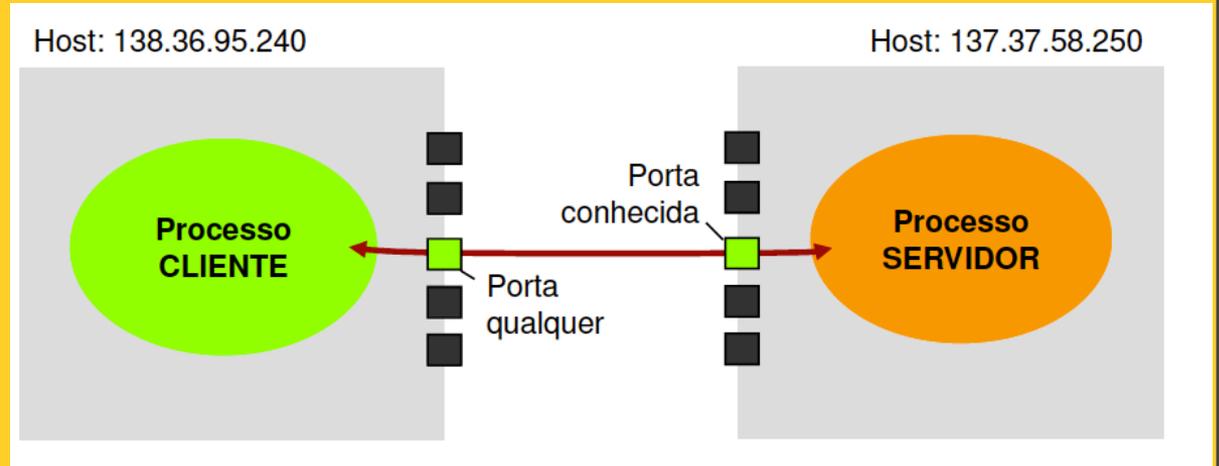
SOCKETS:UDP



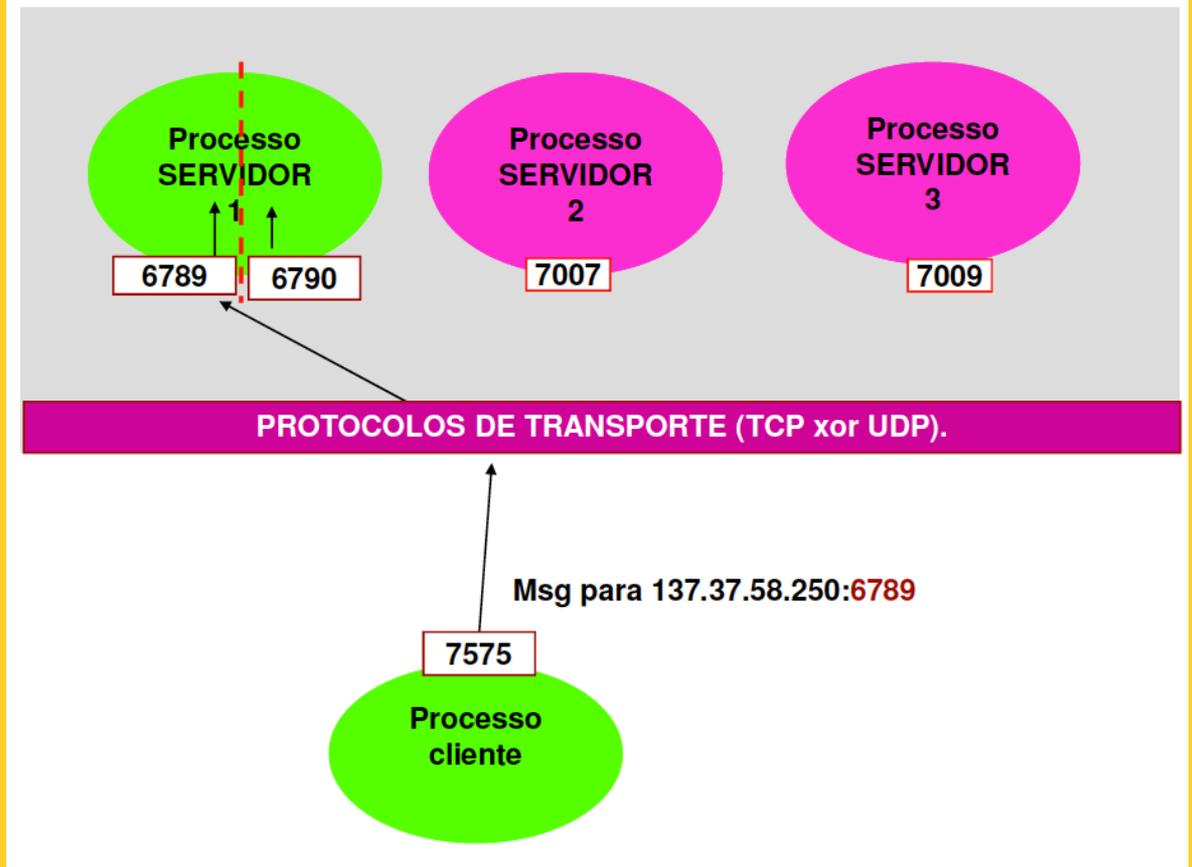
SOCKETS:TCP



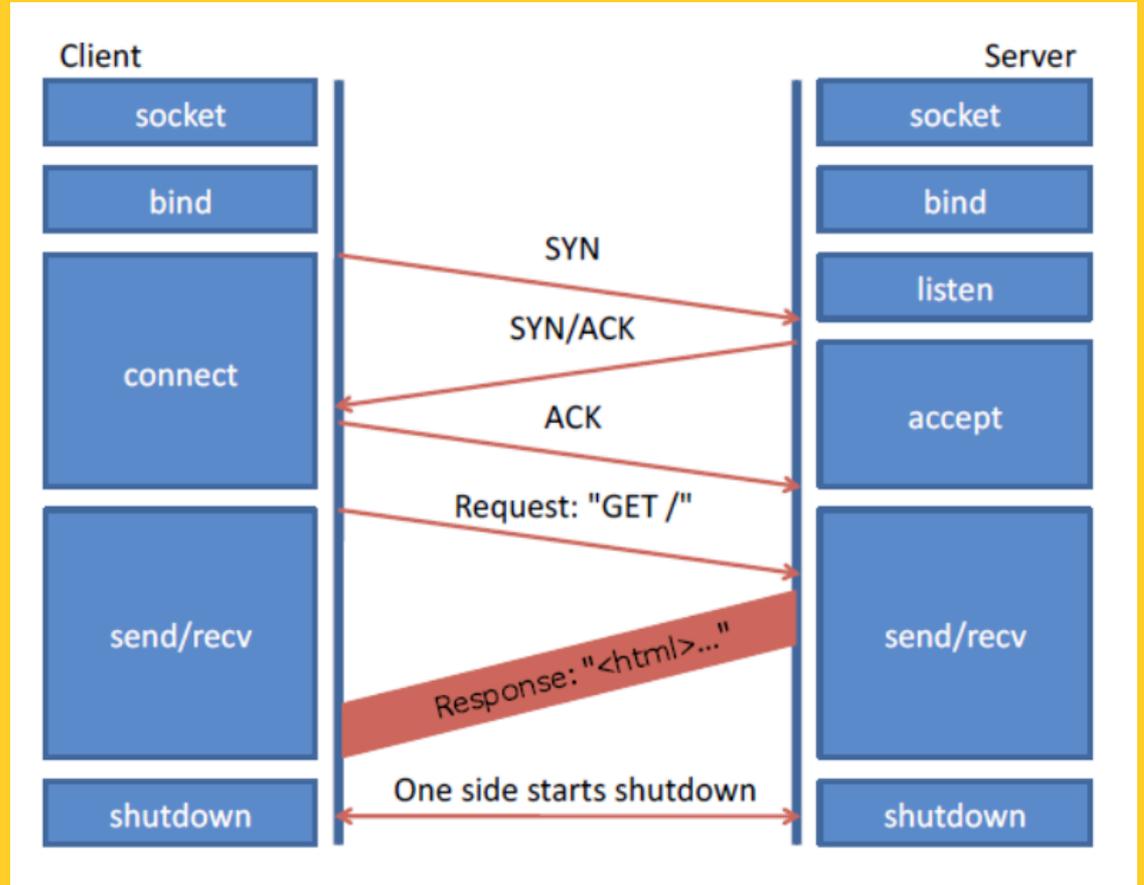
SOCKETS: MODELO CLIENTE-SERVIDOR



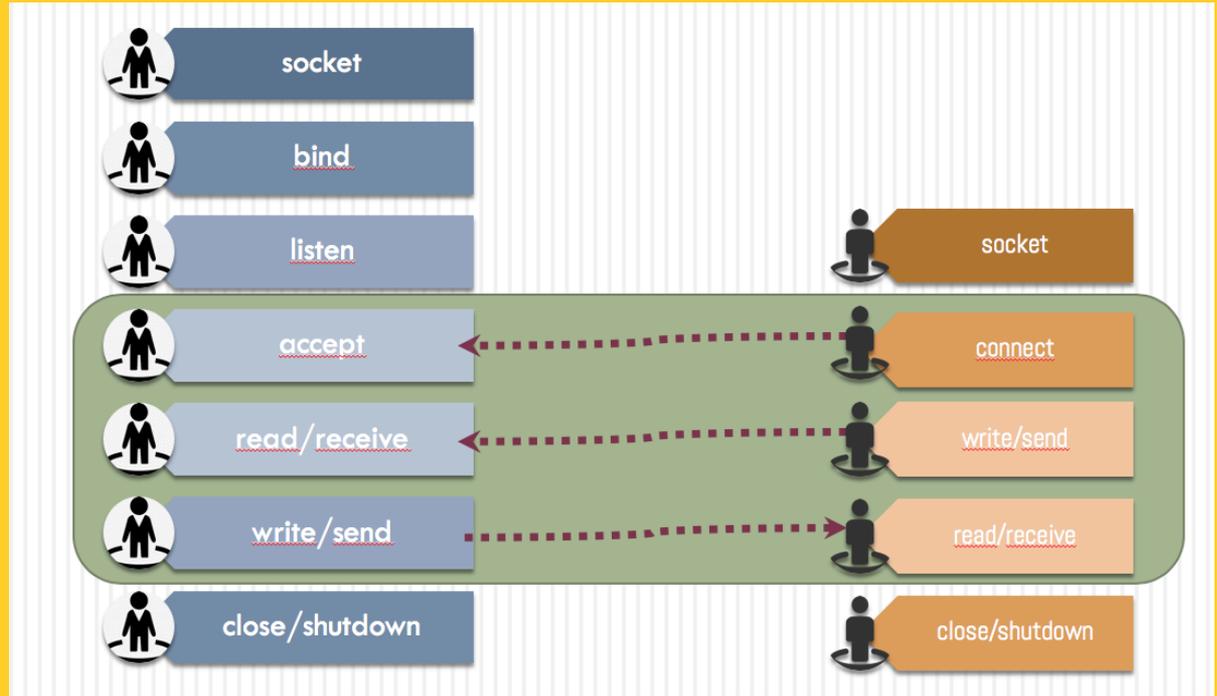
SOCKETS: MODELO CLIENTE-SERVIDOR

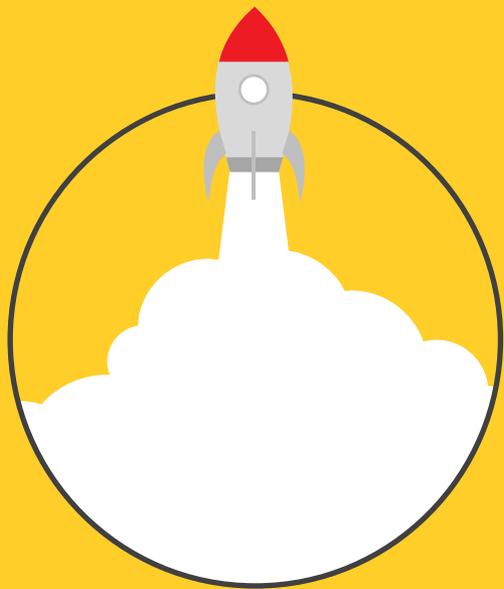


SOCKETS: MODELO CLIENTE-SERVIDOR



SOCKETS: CICLO DE VIDA





TECNOLOGIA DE REDES

REVISÃO – PYTHON

PYTHON

Python

```
>>> class Learning:
...     def __init__(self, name, age, gender):
...         self.title = learn
...         self.subtitle = python
...         self.paragraph = everyday
...
>>> Programmer = Learning("learn", python, "everyday")
>>> print Sue
<_main_.Programmer instance at 0x32111320>
>>> print Programmer.subtitle
python
```

PYTHON 3



```
#!/usr/bin/env python3
x = int(input("Informe um numero : "))
msg = 'Maior que zero' if x > 0 else 'Igual ou menor que zero'
print(msg)
```

PYTHON: IF



```
# Lembre de usar raw_input() no Python 2
x = int(input("Informe um numero :"))
if x > 0:
    msg = 'Maior que zero'
elif x == 0:
    msg = 'Igual a zero'
else:
    msg = 'Menor que zero'
print(msg)
```

PYTHON: MAIN FUNCTION



```
#!/usr/bin/env python3
def main():
    print("Olá Mundo !!")

if __name__ == "__main__":
    main()
```

PYTHON: STRINGS



```
#!/usr/bin/env python3
name = 'Batata'
idade=16
print('Olá, ' + name + '!') # Ruim
print('Olá, %s!' % name)     # Antigo
print('Olá, {}'.format(name)) # Novo
print('Olá, {} - idade {} anos!'.format(name,idade))
```

PYTHON: STRINGS

Formatação Nomeada

```
>>> TMPL = 'Você obteve um erro no arquivo {file} na linha {line}'
```

```
>>> print(TMPL.format(file='a.py', line=5))
```

```
Você obteve um erro no arquivo a.py na linha 5
```



PYTHON: STRINGS

Formatação Posicionada

```
>>> print('{0}, {0} e {1}'.format('repete', 'não repete'))
```

```
repete, repete e não repete
```



PYTHON: STRINGS

Quantidade de Caracteres

```
>>> TEST_RESULTS_TMPL = '{nome:40} {status:10}'  
>>> print(TEST_RESULTS_TMPL.format(nome='NDU', status='Failed'))
```

```
NDU                               Failed
```

```
>>> print(TEST_RESULTS_TMPL.format(nome='Cluster expansion', status='Succeed'))  
Cluster expansion                 Succeed
```

PYTHON: OPERAÇÕES



Operação	Significado
<	Menos que
<=	Menos ou igual que
>	Maior que
>=	Maior ou igual que
==	Igual
!=	diferente
is	objeto é idêntico
is not	o objeto não é idêntico



Perguntas?

105568 - TECNOLOGIA DE REDES

AULA 1 - REVISÃO DE PYTHON E SOCKETS



Prof. Luis Rodrigo – luis.goncalves@ucp.br – <http://lrodrigo.sgs.Incc.br>