

# Comunicação entre Processos com Socket em Python

## Segurança da Informação

Charles Tim Batista Garrocho

Instituto Federal de São Paulo – IFSP  
Campus Campos do Jordão

`garrocho.ifspcj.o.edu.br/SEGA6`

`charles.garrocho@ifsp.edu.br`

Curso Superior de TADS



INSTITUTO FEDERAL

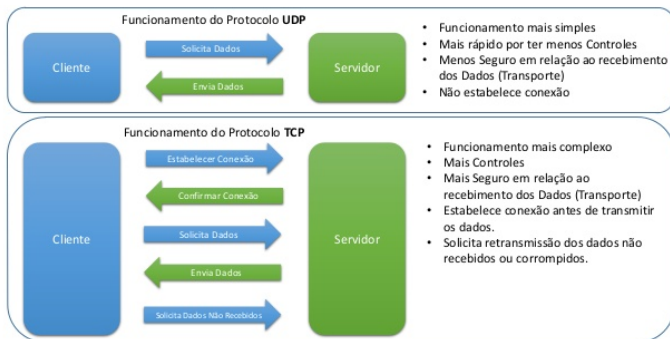
- Permite a troca de mensagens entre processos executando em hosts diferentes.
- Um socket deve ser associado (bind) a um **endereço IP** e a uma **porta** (0-65535).
- As portas 0 a 1023 são as portas reservadas por processos (FTP **21**, SMTP **25**, HTTP **80**).
- Um programa **servidor** cria um socket e o mantém escutando (*listening*) em determinada porta.
- O programa **cliente** cria um socket, conecta-o ao servidor (ip/porta) e envia/recebe dados.



# Socket: Serviços de Transporte

Há dois tipos de serviços de transportes via sockets:

- **Transmission Control Protocol (TCP):** confiável, orientado a conexão.
- **User Datagram Protocol (UDP):** não confiável, cada datagrama é um registro único e indivisível.



INSTITUTO FEDERAL

# Socket: Implementação em Python

**Acesse:** <https://wiki.python.org.br/SocketBasico>

*# Importando a biblioteca socket do Python*

**import socket**

*# Criando um socket TCP*

**s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)**

*# Criando um socket UDP*

**s = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)**



INSTITUTO FEDERAL

# Socket: Cliente TCP

Implementação do Cliente: Utiliza um soquete TCP e envia a mensagem *Ola Mundo!* para o servidor definido no **host** e **porta**.

```
import socket
host = '127.0.0.1'      # Endereco IP do Servidor
porta = 5000            # Porta que o Servidor esta
soquete = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
destino = (host, porta)
soquete.connect(destino)
soquete.send("Ola Mundo!")
soquete.close()
```

O endereço **127.0.0.1** é o endereço da própria máquina, reservado para a interface de loopback.



INSTITUTO FEDERAL

# Socket: Servidor TCP

Implementação do Servidor: Aguarda novas conexões realizadas pelos clientes através da função **accept**.

```
import socket
host = '127.0.0.1' # Endereco IP do Servidor
porta = 5000       # Porta que o Servidor esta
soquete = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
origem = (host, porta)
soquete.bind(origem)
soquete.listen(0)
while True:
    conexao, cliente = soquete.accept()
    print 'Concetado por', cliente
    mensagem = conexao.recv(1024)
    print "Cliente ", cliente[0], "Recebida: ", mensagem
    conexao.close()
```

O servidor vincula seu **serviço** ao endereço IP definido no **host** e **porta**.



INSTITUTO FEDERAL

## Atividade 1:

- Baixe a implementação em Python do Cliente e do Servidor TCP no site da disciplina.
- Após baixar, encontre um colega, e combine com seu colega para que ele seja um Servidor e você um Cliente, ou vice-versa.
- Vocês deveram utilizar a mesma porta no código, e a pessoa que for o cliente deverá saber o endereço IP da pessoa que for o Servidor.
- Vocês deveram se comunicarem de forma que o Cliente envie uma mensagem e o Servidor receba essa mensagem.

## Atividade 2:

- Criar um chat que permita varias pessoas de comunicarem.

