

Sockets em **Python**: Salvando a Vida

Road to Localhost



Quem sou eu

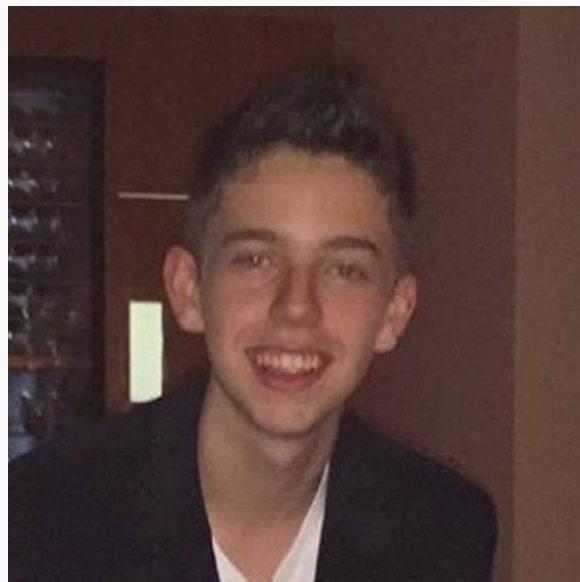
Vinícius Mesel

Twitter: @vmesel

github.com/vmesel/

me@vmesel.com

Pyhtonista e gamer



Para que serve Networking?



Tá mas para que serve realmente?

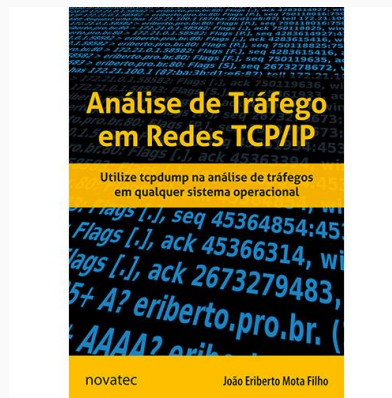
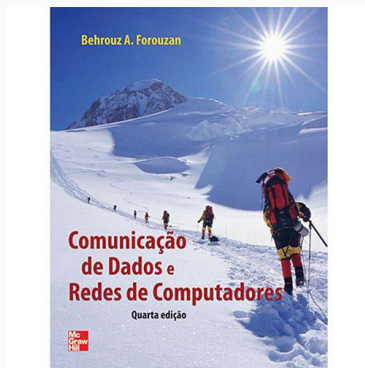
Ligar computadores na Internet;

Poder fazer softwares "**slaves**" se comunicarem com o "**master**";

Realizar a criação de softwares P2P;

Resumindo: **REALIZAR CONEXÕES ENTRE UMA MAQUINA E OUTRA, nesse caso com Phyton**

Como começar?



Para começar com Python Networking é necessário

- Paciência
- Ter um computador com Internet
- Gostar de abrir pacotes e fuçar
- Gostar de Python/Linguagem de programação com suporte a redes

O que você precisa saber

- O Python já tem uma biblioteca **SOCKET**
- Saber manipular bytes e strings
- Ter um mínimo de noção de como a Internet funciona

Métodos da biblioteca: Server

- **socket(*socket_family, socket_type*)**

O criador de um novo Socket

- **bind(*(hostname,port)*)**

Pega uma tupla de host e porta

- **listen()**

Começa a ouvir as conexões

- **accept()**

Aceita qualquer conexão entrante

Métodos para cliente

connect((hostname,port))

Inicia a conexão através de uma tupla

recv(buffer)

Tenta pegar dados da conexão TCP

send(bytes)

Tenta enviar bytes para o servidor

close()

Fecha a conexão

Manipulando Strings e Bytes

Um byte é igual a:

"String Aqui".encode() #Encoding para UTF-8 ou ASCII

Uma string a partir de byte é:

b"Bytes aqui".decode() # Decoding para uma string

Quando usar TCP e quando usar UDP

TCP checa todo o pacote para checar por falhas, e se houver algum dado que não foi enviado, ele reenvia(**SOCKET_STREAM**)

UDP manda o pacote e se houver falha nas comunicações, ele não reenvia (**SOCKET_DGRAM**)

Exemplo simples

Hora da DEMO **Pythonica!**

Obrigado pela atenção

Qualquer dúvida, me mande um email!

Vocês repararam algum easter egg?

me@vmesel.com