

**Nomes:** Paulo Victor Nogueira Rodrigues  
Gabriel Tomazini Marani

**RA:** 2265125  
**RA:** 2266083

## Relatório Wireshark

Inicialmente, para a análise dos pacotes trocados no Wireshark, foi necessário aplicar filtros na interface de rede local (localhost), aplicando filtragem para a visualização apenas aos pacotes transmitidos via protocolo TCP. Posteriormente, foram identificados e extraídos os dados referentes aos processos de *handshake* e de chamada do método “Buscar”, demonstrando parte do processo de troca de pacotes realizado. Os trechos a seguir apresentam os bytes capturados dessas duas etapas, respectivamente.

### Handshake Inicial : Cliente ao Servidor

O pacote analisado corresponde à mensagem inicial de handshake do protocolo *Pyro5*. A assinatura da mensagem inclui o identificador "PYRO", que indica que o protocolo utilizado é o *Pyro5 (Python Remote Objects)*. Em seguida, há o byte 0x01, representando a versão do protocolo, e o byte 0xf6, que indica que se trata de uma mensagem do tipo handshake.

|      |                         |                         |                    |
|------|-------------------------|-------------------------|--------------------|
| 0050 | 00 00 00 00 02 41 68 29 | 56 8b 4d fd 9f 3c d5 0e | .....Ah) V·M··<··  |
| 0060 | d2 e4 4d 28 00 00 4d c5 | 23 20 73 65 72 70 65 6e | ··M(··M· # serpen  |
| 0070 | 74 20 75 74 66 2d 38 20 | 70 79 74 68 6f 6e 33 2e | t utf-8 python3.   |
| 0080 | 32 0a 7b 27 68 61 6e 64 | 73 68 61 6b 65 27 3a 27 | 2·{'hand shake':'  |
| 0090 | 68 65 6c 6c 6f 27 2c 27 | 6d 65 74 61 27 3a 7b 27 | hello',' meta':{'  |
| 00a0 | 6d 65 74 68 6f 64 73 27 | 3a 7b 27 61 64 69 63 69 | methods' :{'adici  |
| 00b0 | 6f 6e 61 72 27 2c 27 72 | 65 6d 6f 76 65 72 27 2c | onar','r emover',  |
| 00c0 | 27 62 75 73 63 61 72 27 | 2c 27 61 74 75 61 6c 69 | 'buscar' , 'atuali |
| 00d0 | 7a 61 72 27 7d 2c 27 6f | 6e 65 77 61 79 27 3a 28 | zar'}, 'o neway':( |
| 00e0 | 29 2c 27 61 74 74 72 73 | 27 3a 28 29 7d 7d       | ), 'attrs ':(())}} |

Figura 1: Mensagem inicial

Além disso, o payload da mensagem está serializada no formato **serpent** usado para representar objetos, semelhante ao JSON. O trecho serializado é codificado em **UTF-8**. E mais importante, mostra os métodos remotos disponíveis com o trecho:

```
{ 'methods': {'adicionar','remover','buscar','atualizar' } }
```

Com os métodos revelados ao cliente, basta a seleção da opção para o servidor receber o retorno e realizar a execução.

## Utilização do Método de Busca

Para fins de exemplificação, tomemos o método “buscar” como base para a análise dos pacotes trocados entre o Cliente e o Servidor. Após a escolha e chamada do método, o retorno correspondente à solicitação de busca é exibido a seguir:

```
0040 50 59 52 4f 01 f6 04 01 00 00 00 01 00 00 00 53 PYRO....S
0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0060 00 00 00 00 00 00 00 4d c5 23 20 73 65 72 70 65 6e .....M # serpen
0070 74 20 75 74 66 2d 38 20 70 79 74 68 6f 6e 33 2e t utf-8 python3.
0080 32 0a 28 27 6f 62 6a 5f 65 31 63 39 37 36 32 62 2 ('obj_ e1c9762b
0090 36 33 61 31 34 30 31 37 38 38 65 64 63 61 62 62 63a14017 88edcabb
00a0 62 39 38 63 34 35 35 36 27 2c 27 62 75 73 63 61 b98c4556 ', 'busca
00b0 72 27 2c 28 32 2c 29 2c 7b 7d 29 r', (2,), {})
```

Figura 2: Método de Busca

Esse é o resultado referente ao iniciar o método com o parâmetro “2”, solicitando que o servidor retorne o objeto com esse ID no banco de dados. A estrutura da mensagem segue o padrão do Pyro5: identificador do objeto, nome do método, parâmetros posicionais e argumentos nomeados. Como demonstrado abaixo:

```
('obj_e1c9762b63a1401788edcabbcb98c4556', 'buscar', (2,), {})
```

Após a requisição do Cliente para o Servidor, logo vem o retorno com o seguinte trecho de bytes:

```

0040 50 59 52 4f 01 f6 05 01 00 40 00 01 00 00 00 81 PYRO.... @.....
0050 00 00 00 00 ab d1 9d 65 93 3c 4b 3b 81 fd 83 cf .....e <K;....
0060 1c 22 cb fc 00 00 4d c5 23 20 73 65 72 70 65 6e ."....M. # serpen
0070 74 20 75 74 66 2d 38 20 70 79 74 68 6f 6e 33 2e t utf-8 python3.
0080 32 0a 7b 27 5f 5f 63 6c 61 73 73 5f 5f 27 3a 27 2.{'__cl ass__':
0090 6f 62 6a 65 74 6f 73 2e 6f 62 6a 65 74 6f 43 52 objetos. objetoCR
00a0 55 44 27 2c 27 69 64 27 3a 32 2c 27 6e 6f 6d 65 UD', 'id' :2, 'nome
00b0 27 3a 27 63 72 6f 6e 61 6e 27 2c 27 63 6c 61 73 ':'crona n', 'clas
00c0 73 65 27 3a 27 6d 6f 6e 67 65 27 2c 27 65 73 70 se': 'mon ge', 'esp
00d0 65 63 69 65 27 3a 27 68 75 6d 61 6e 6f 27 2c 27 ecie': 'h umano',
00e0 6e 69 76 65 6c 27 3a 33 7d nivel':3 }

```

Figura 3: Retorno do Servidor

Como observado na Figura (3), pacote de dados retorna o personagem requerente ao ID solicitado no método “Buscar” realizado anteriormente pelo o Cliente.