

### Passos para demonstração do sistema:

- 1) Criar 3 máquina virtuais. Escolher uma máquina para ser o **Servidor Rabbit (Máquina2)**. Nesta, deve-se configurar uma “conta”. A segunda máquina ficará o **Gerador de eventos (Máquina1)** (Producer.py), que vai ler o arquivo de texto com os dados de leitura dos sensores. A terceira máquina ficará com o **Receptor1** (Consumer.py), esta tratará os dados enviados pelo Gerador e encaminhará para um outro **Receptor2** (Receiver.py) que enviará tudo para o banco de dados que irá ficar na **Máquina Física**. Ambos receptores estão na **Máquina3**. Ao final, um **Sistema Feito em Html/Javascript** (que também ficará na **Máquina Física**) mostrará o mapa de calor da localização dos sensores no campo (sensores das chuteiras e luvas dos goleiros).
- 2) As 3 máquina virtuais (mininet) têm sistema operacional Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-24-generic i686) e foram criadas pelo VirtualBox. Para melhor manuseio, elas podem ser acessadas usando o Putty, para Windows, e o Terminal para sistemas Unix.
- 3) Todas as 3 máquinas devem ter a instalação dos seguintes softwares:
  - a) python 2.7
  - b) rabbitmq
  - c) spark-1.6.1-bin-hadoop2.6
  - d) spark-streaming-mqtt-assembly\_2.11-1.6.1
  - e) java-7-openjdk

### Passos para executar o sistema:

- 1) Abrir a 3 máquinas virtuais no VirtualBox; logar; verificar os IPs de cada uma pelo comando “ifconfig” no terminal; acessar todas elas usando o Putty (basta digitar o IP e entrar), no Windows, e o Terminal em sistemas Unix (usando o comando nome\_do\_usuario@ip\_da\_máquina).
- 2) Quando a maquina2 estiver ligada, pode-se acessar o “painel do Rabbit” usando um navegador e digitando: **ip\_do\_servidor\_rabbit:15672**
- 3) Executar na máquina3 os dois receptores:
  - a) Para o Receptor1, deve-se executar o programa consumer.py na pasta mostrada abaixo:  
Acessar a pasta:  
**\$ cd spark-1.6.1-bin-hadoop2.6/**  
  
Executar o arquivo python:  
**~/spark-1.6.1-bin-hadoop2.6\$ ./bin/spark-submit --master local[2] --jars ../rsi-psd-codes/psd/rabbitmq-spark-integration/spark\_jar/spark-streaming-mqtt-assembly\_2.11-1.6.1.jar ../rsi-psd-codes/psd/rabbitmq-spark-integration/maquina3/consumer.py arg1 arg2 arg3**  
  
onde:  
**arg1** = IP do servidor Rabbit  
**arg2** = número de linhas do grid (8,...,64)  
**arg3** = número de colunas do grid (13,...,100)

- b) Duplicar a sessão da máquina3 e executar o Receptor2, onde deve-se executar o programa receiver.py na pasta mostrada abaixo:

```
$ cd rsi-psd-codes/psd/rabbitmq-spark-integration/
```

Executar o arquivo python:

```
~/rsi-psd-codes/psd/rabbitmq-spark-integration$ maquina3/python receiver.py arg1
```

**onde:**

**arg1** = IP do servidor Rabbit

- Acessar o banco pelo terminal:

```
mysql -u root -p
```

```
show databases;
```

```
use nome_do_banco
```

```
show tables;
```

```
describe nome_da_tabela;
```

```
select * from nome_da_tabela;
```

- 4) Na máquina1 (Gerador) deve-se acessar a pasta e executar o programa producer.py feito em python 2.7

```
cd rsi-psd-codes/psd/rabbitmq-spark-integration/maquina1/producer.py arg1 arg2 arg3
```

**onde:**

**arg1** = IP do servidor Rabbit

**arg2** = arquivo de texto com os dados dos sensores

**arg3** = número do salto de linhas na leitura do arquivo