A Tale of Cats and a Mouse

Deliverables and Guidelines

The exercise must be programmed in Python. Implementation in any other language will lead to a reduced grade. You need to include a document that gives the instructions on how to run your program and describes the execution process. Put all the files in a directory which has the same name as your username then zip that directory into an archive and name it *username_BE2.zip*, e.g. *ysmith_BE2.zip* and return the zip through Moodle.

The deadline for submission is Oct 23rd 23:55.

Task Description
Once upon a time in a computing cluster called Ukko, four cats were trying to catch a mouse. The characters of the tale are: cheese-loving Mousie Mouse, driven Catty Cat, urban Jazzy Cat, attentive Listy Cat and overseeing Cordy Cat.

Each of the characters are made up by their python scripts and are run on the Ukko cluster. Use the following naming convention for scripts:

mouse.py	The mouse
chase_cat.py	For Catty and Jazzy
listy.py	For Listy
cordy.py	For Cordy

The script setup would be as follows:

1. The mouse starts by going to one of the Ukko nodes and listening on a port.
2. Cordy dispatches Catty and Jazzy such that they log into every possible Ukko nodes one node at a time and search if the mouse is running there. Both the cats must search for the mouse **concurrently!** The nodes searched by Catty and Jazzy must not overlap with each other.
3. Catty and Jazzy communicate with Cordy through Listy cat.
    - Listy keeps an open socket and writes the messages it receives from Catty and Jazzy to a file called **cmsg**.
    - Cordy then reads the messages from the cmsg file.
4. Cordy sends Catty and Jazzy to each Ukko node via ssh.
    - Cordy gives the commands to cats as following command line parameters:

S catname	Search the node (e.g. python3 chase_cat.py S Jazzy)
A catname	Attack the mouse on the node

Catty and Jazzy can send the following messages to Listy:

F ukkoXXX catname	Found the mouse on ukkoXXX (e.g. F ukko064 Catty),
G ukkoXXX catname	Got the mouse

A typical conversation flow in this tale would be as follows:

1. When a cat finds the mouse it will send the F message to Listy which will be written on cmsg.
2. Cordy will then make the other cat to search the same node.
3. When Cordy knows that both cats have found the mouse it can give the Attack command to either of the cats.
4. The cat attacks by connecting to the mouse using its port number and sending a MEOW message.
5. When the mouse receives the MEOW message, it will send an OUCH message to the attacking cat (indicating it has been caught).
6. The attacking cat will report back of a successful attack with the G message to Listy.

However, like every story, this tale must also have certain twists! Some of the constraints that you must apply are:

1. Select 10-20 nodes from Ukko to use and put their names on a file called **ukkonodes**. Start the mouse process on one of them. The program must work so that it does not matter which of the selected nodes is chosen for the mouse.
2. Jazzy and Catty need 12 seconds to search one node.
3. The attack can only happen after both Catty and Jazzy have found the mouse.
4. The attack on the mouse takes 6 seconds.
5. The attacking cat will wait 8 seconds for the OUCH message.
6. The mouse does not move.
7. The port number used with any socket communication must specified in a file called **port_number**, i.e., the port number is read from that file whenever needed by any process.
8. The node where Listy (listy.py) resides must be defined in a file called **listy_location**.

Please make sure to name all files as defined in this description.

You should be able to implement a working program. However, note that some parts are by nature fault-prone. Remember to comment your code.

And the cats lived happily ever after.

The End!