Challenge 6 - Man in the middle

Now you work for the NSA. We have an important service being monitored and we want to decipher and modify communications going to and coming from this server. Fortunately our agents got access to both client and server code so you can study the communication protocol.

As you can see, the client uses a keyphrase to obtain a secret message from the server. We have obtained a keyphrase from an informer and we want to know the associated secret message.

We can't connect directly to the server but we found a backdoor on the client's router and we were able to install a man-in-the-middle service to spy and

modify communications. The address to the service is:

54.83.207.90:6969

Your mission is to replace the client's keyphrase with ours and decipher the associated secret message.

Once you connect to the man-in-the-middle service, you will receive strings like:

```
CLIENT->SERVER:hello?
```

This means that the client is sending the message "hello?" to the server. Now you can modify the message (if you want) and send it back. The man-in-the-middle service will submit your message to the server.

Submit & test your code

To test and submit code we provide a set of tools to help you. Download contest tools if you haven't already done that. You will then be able to test and submit your solution to this challenge with the challenge token.

```
Challenge token: kHpxcSnHnS53Fq-LRbwS
```

To test your program

```
./test_challenge kHpxcSnHnS53Fq-LRbwS path/program
```

A nice output will tell you if your program got the right solution or not. You can try as many times as you need.

To submit your program to the challenge

```
./submit_challenge kHpxcSnHnS53Fq-LRbwS
path/source_pkg.tgz path/program
```

Note that you first need to solve the test phase before submitting the code. During the submit phase, in some problems, we might give your program harder questions, so try to make your program failsafe.

Important: In this phase, you must provide the source code used to solve the challenge and, if necessary, a brief explanation of how you solved it.

Remember **you can only submit once!** Once your solution is submitted you won't be able to amend it to fix issues or make it faster, so please be sure your solution is finished before submitting it.

If you have any doubts, please check the info section.

Go ahead

I'm done! :)

Once you have submitted your code, hit refresh and continue to next challenge.

I'm stuck! :(

Be sure you follow the Tuenti Engineering twitter for updates and possible hints during the contest.

If this challenge is too hard and you are blocked, you will be able to skip it after two hours. Note that **you won't be able to complete it later**, and you have a limited number of challenges to skip.

Finally, if you run out of skips but are still really stuck with one problem, you will be able to skip it after 24 hours.

Challenge status:

Test case	Not done
Solution submitted	Not done
Skip	You still have to wait 0h, 30m and 0s to be able to skip this challenge

Refresh status

Tweet about this! #TuentiChallenge4

Share Follow @Tuentieng