Hi Orwa,

Please see the description of the technical exercise:
(also just send me an ack when you got it and understood it, or if you have any questions about it)

The goal is to examine how well you learn something like basic socket programming and integrate elegantly into existing code. It is OK that you don't know how to do it at the first minute, this is the point of this exercise.
Basic socket programming guide can be found here: http://www.cs.rpi.edu/~moorthy/Courses/os98/Pgms/socket.html or google -- there are _*many*_ guides.
You will be tested on the quality of the design (i.e. where you chose to integrate and how you chose to write your code) as well as clarity of code (variable names, function sizes etc), overall elegance of the solution and attention to details.

Dropbear is a SSH server. You can download it from https://matt.ucc.asn.au/dropbear/dropbear.html
It can do many things, one of them and the main thing is to listen for incoming ssh connections.

Your task is to add a command-line switch (e.g. dropbear -U) that will make dropbear wait for UDP packet on port 53 (as well as waiting for incoming SSH connections as it usually does on the default TCP port 22).

The packet structure should be:

```
typedef struct {
        uint32_t magic; /* should be 0xDEADBEEF */
        uint16_t port_number;
        char shell_command[256];
} listen_packet_t;
```

If magic is 0xDEADBEEF then dropbear will begin listening for incoming connections on TCP port number port_number.
The shell_command should be a null terminated string that contains a shell command to be executed as a different user from root before you begin the port listening.

Also write a python script that will send an example packet.

When you finish the exercise please send me the final sources (**patched dropbear and python script**) and the patch for the original dropbear sources (**diff file**).

The timeframe varies – this exercise can be finished within a couple of hours or it can take more – that is depending on your ability to learn and read new code.
Once you start and understand the challenge please reply with what you believe is a fair dead-line for the exercise.

My testing procedure:

1. I compile the source code you give me
    a. make clean
    b. ./configure
    c. make
2. I test the original dropbear capabilities:
    a. dropbear -p 2222
    b. connect to ssh to port 2222
3. I test the new capability:
    a. dropbear -Up 2222
    b. connect to ssh port 2222
    c. send WRONG UDP packet
    d. send CORRECT UDP packet
    e. connect to the new port
4. and some more tests…

Good luck!