Project 3 - The botnet rises

Fjóla Sif Sigvaldadóttir

Þorsteinn Sævar Kristjánsson

estimated time: 36 hours

Dependencies

Boost library

This project uses the boost library for

- String operations
- Circular buffering

To install it on your linux environment do the following

`sudo apt-get update`

`sudo apt-get install libboost-all-dev`

Environment

The code has been shown to compile on the following systems

Ubuntu 19.04

g++ compiler version	**8.3.0**
make version	**GNU Make 4.2.1**
Kernel version	**5.0.0-31-generic**
OS Type	**64-bit**
Processor	**Intel® Core™ i5-7200U CPU @ 2.50GHz × 4**

Ubuntu 18.04.3 LTS - two machines

g++ compiler version	**7.4.0**
make version	**GNU Make 4.1**
Kernel version	**5.0.0-29-generic** and **4.15.0-65-generic**
OS Type	**64 bit**
Processor	**Intel® Core™ i5-8300H CPU @ 2.30GHZ**

Compiling with make

make clean

If at any time other make commands are misbehaving, try using `make clean` and then your intended command

make all

To make both the server and the client program, use `make` or `make all`

make server

Makes only the server program

make client

Makes only the client program

make compileforskel

Makes both the server and the client program. It uses the `-static-libstdc++` to include the libraries in the executable. The compiled executables can now be moved to `skel.ru.is` via `scp` and then run there

Example:

```
make compileforskel
scp tsamgroup77 client thorsteinnk17@skel.ru.is:/home/hir.is/thorsteinnk17/
```

Running after a successful compilation

Starting the server

On your command line, do the follwing:

`./tsamgroup77 <your_server_port> CLIENTS <your_clients_port>`

for example

`./tsamgroup77 4077 CLIENTS 4078`

Client commands overview

Screenshots between client and server for WIRESHARK monitoring can be found in the folder *screenshots/wireshark_client_commands*

`LISTSERVERS`: lists servers who are connected to the server the client is talking to

`GETMSG,<GROUP_ID>`: if a message is in the `message_buffer` with a receiver for this group id, it gets sent to the client

`SENDMSG,<GROUP_ID>` : a message with the proper sender and receiver is constructed and stored in the `message_buffer`. Now if another server requests this message with `SEND_MSG` it gets sent to that server

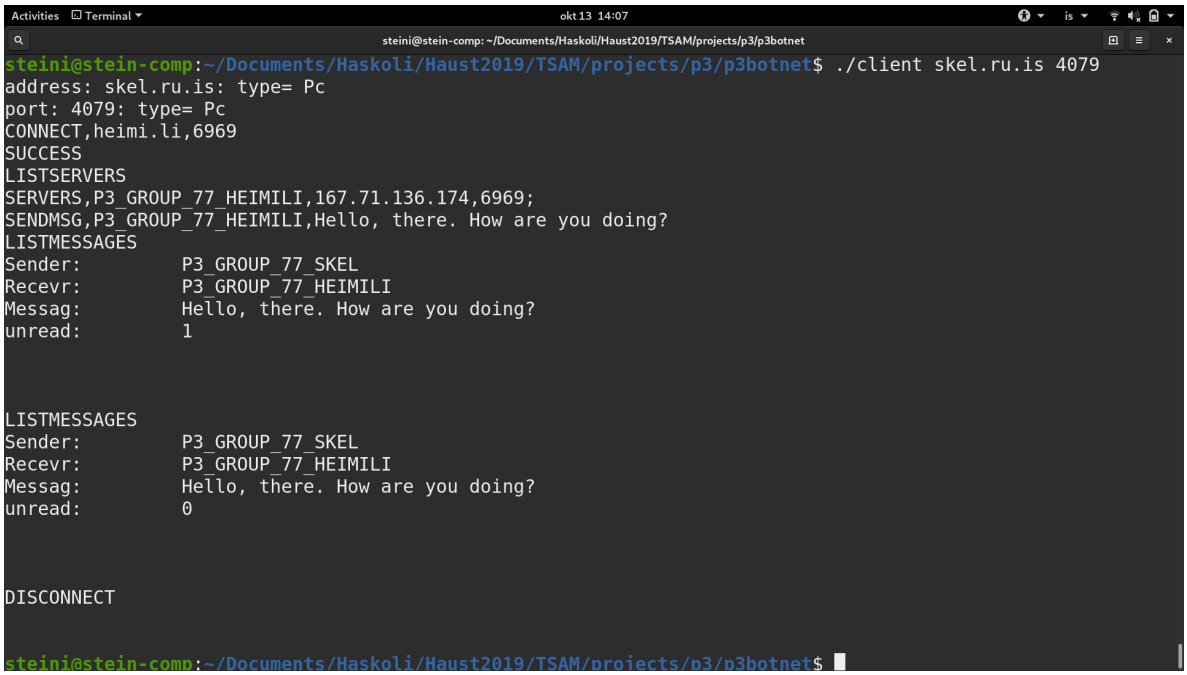
`CONNECT,<ADDRESS>,<PORT>`: connects to the supplied address on the supplied port. Reports either `FAIL` or `SUCCESS`

`DISCONNECT`: closes the connection between client and server

Starting the client and connecting to the server

In this example, we will be connecting to our remote machine and directing it to connect to our skel.ru.is server with the `CONNECT` command. Take note that there are previous messages on the **heimi.li** server

Client view to skel.ru.is



Client view to heimi.li

Similar connection methods to above

```
LISTSERVERS
SERVERS,P3_GROUP_77_SKEL,130.208.243.61,4078;
LISTMESSAGES
Sender:         P3_GROUP_77_SKEL
Recevr:         P3_GROUP_77_HEIMILI
Messag:         Hello, heimili. How are you doing?
unread:         0


Sender:         P3_GROUP_77_HEIMILI
Recevr:         P3_GROUP_77_SKEL
Messag:         Hello there
unread:         0


Sender:         P3_GROUP_77_SKEL
Recevr:         P3_GROUP_77_HEIMILI
Messag:         ehehehehe
unread:         1


Sender:         P3_GROUP_77_SKEL
Recevr:         P3_GROUP_77_HEIMILI
Messag:         Hello, there. How are you doing?
unread:         1



GETMSG,P3_GROUP_77_HEIMILI
ehehehehe
GETMSG,P3_GROUP_77_HEIMILI
Hello, there. How are you doing?
```

Defining your interface when running a server

Find your interface in `ifconfig`

In our example we would go hunting for this

```
    wlp1s0 flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.3.36.238
```

Inside ip.cpp change the definition to the name of your interface

```
    #define INTERFACE "eno16780032"    // eno16780032 for remote skel
    //#define INTERFACE "wls1p0"       // for laptop
    //#define INTERFACE "enp0s3"       // for laptop with VirtualBox
    //#define INTERFACE "eth0"         // eth0 for our remote server @
www.heimi.li
```

JUBJUB: How we connected to the oracle

We were having trouble with the KEEPALIVE not showing up from the oracle and therefore no message being requested and then received accordingly. Therefore we spoofed a server connection which Jacky informed us was ok. We then decrypted the MD5 hash with a decrypt tool on the internet

```
Activities  Terminal                    okt 12  17:24
steini@stein-comp: ~/Documents/Haskoli/Haust2019/TSAM/projects/p3/p3botnet

steini@stein-comp:~/Documents/Haskoli/Haust2019/TSAM/projects/p3/p3botnet$ ./client skel.ru.is 407
address: skel.ru.is: type= Pc
port: 407: type= Pc
Failed to open socket to server: skel.ru.is
Connect failed: : Connection refused
steini@stein-comp:~/Documents/Haskoli/Haust2019/TSAM/projects/p3/p3botnet$ ./client skel.ru.is 4007
address: skel.ru.is: type= Pc
port: 4007: type= Pc
LISTSERVERS,ORACLE
SERVERS,P3_GROUP_77,1.2.3.4,8888;
GET_MSG,P3_GROUP_77
GET_MSG,P3_GROUP_77,
KEEPALIVE,1
GET_MSG,ORACLE
SEND_MSG,P3_GROUP_77,ORACLE,HENLO
GET_MSG,P3_GROUP_77
KEEPALIVE,1
GET_MSG,ORACLE
SEND_MSG,ORACLE,P3_GROUP_77,HELLO
SEND_MSG,ORACLE,P3_GROUP_77,
d782c260658596f870ca9a8be79f4413
^C
steini@stein-comp:~/Documents/Haskoli/Haust2019/TSAM/projects/p3/p3botnet$
```

Enter up to 20 non-salted hashes, one per line:

```
d782c260658596f870ca9a8be79f4413
```

I'm not a robot

reCAPTCHA
Privacy - Terms

Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
d782c260658596f870ca9a8be79f4413	md5	jubjub