Háskólinn í Reykjavík

T-409-TSAM

Assignment 5: The Botnet Saves the World

Katrín Ósk Kristinsdóttir
katrinok20@ru.is

Sævar Örn Valsson
saevarv20@ru.is

supervised by
Dr. Jacky Mallett & Margrét Halla

30. október 2023

The Assignment

This document is intended to follow server.cpp and client.cpp, that contain a server and a client that run together to communicate with other servers on skel.ru.is, along with a README.md file that contains instructions to run the code and use the client, and a Makefile that runs the code. The log files that are also submitted alongside these files shows the performance of the server and client.

Question 1

Implement client and server as described above. All local commands to the server must be implemented by a separate client over the network.

The server and client are in the following code

How the Server Works

The server is initialized with the command "./tsampgroup20 <port>"where the port is the port number the server is listening to. The server is based on breath first search concept were we start by sending a command to the server to connect to another server that this server is connected to. When our server receives what servers the other server is connected to we add it to a queue. The server then checks if there is a server queued to be connected to and if so it tries to connect to it and add it to a connection list. If the server cannot connect to the queued server within the time limit (5 seconds), it gets one more chance to be connected to, as it is put to the back of the queue. The next time the connection fails, the server throws away the queued server.

At the same time this is happening there is also a listening socket listening for incoming connections and trying to connect to it. The **max capacity** of the server as it stands is 10 servers but can be increased, having no capacity for clients. If a server tries to connect to us and the maximum capacity for servers is reached then the server will kick out a random connection to make space for the new connection. This ensures that the server maintains crawling mechanism, connecting to new servers that reach out and throwing out one of the old servers in return. If the server tries to connect to another server from the queue that it is already connected to, it removes the server from the queue, and only prints out a message to the log.

The only clients that are able to connect to our server send us the secret string that is hard coded into our client. These clients are the only clients that are listed as client in our connections list, so the server does not remove them to make a new connection. This was thought to be the easiest solution, so that our peers would not get to connect to us as clients.

How the Client Works

The Client is initialized by the executable $./chat_{c}lient < serverip >< serverport >$. Then it starts by connecting to the server straight away. For the server to recognize that its our client we send a secret token straight away upon connection so the server recognizes us and does not process us like some other server. The client can then send a variety of commands to the server which are all detailed in the README.md file.

Question 2

Provide a wireshark trace of communication between your client and server for all commands implemented.

The Wireshark trace is attached in the submitted tar file.

Question 3

Have been successfully connected to by an Instructor's server.

We have Successfully connected to all instructor servers, $Instr_1$, $Instr_2$ and $Instr_3$, along side Oracle and Number. The results can be seen in the log files that are attached.

Question 4

Successfully receive messages from at least 2 other groups (Provide timestamped log).

We have successfully received messages from at least two groups. The following results can be seen in the log files along with the timestamps.

Question 5

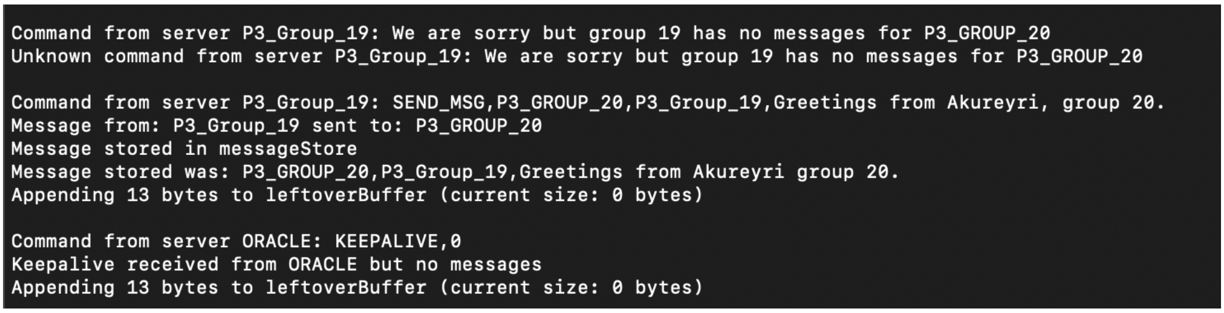
Successfully send messages to at least 2 other groups (Provide timestamped log).

We have successfully sent at least 2 messages to at least 2 other groups. The following results can be seen in the log files

Bonus

Connections to Akureyri

On our way through the botnet we have come across one encounter with a server from Akureyri, group 19, and the communication can be seen in figure 1 and 2. The timestamped log can also be found in clientAkureyri.txt in the log directory. The client side could not be put to a .txt file as it disappeared.

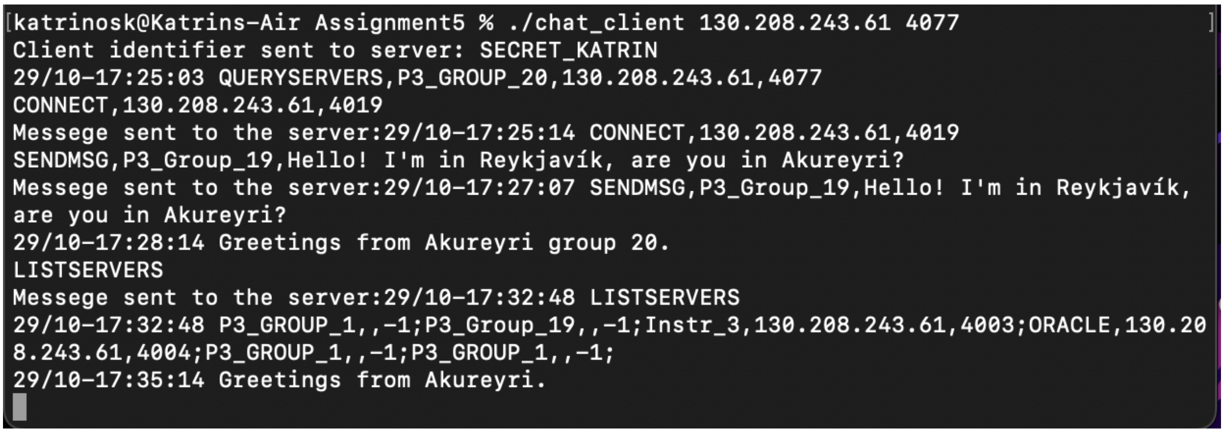


```
Command from server P3_Group_19: We are sorry but group 19 has no messages for P3_GROUP_20
Unknown command from server P3_Group_19: We are sorry but group 19 has no messages for P3_GROUP_20

Command from server P3_Group_19: SEND_MSG,P3_GROUP_20,P3_Group_19,Greetings from Akureyri, group 20.
Message from: P3_Group_19 sent to: P3_GROUP_20
Message stored in messageStore
Message stored was: P3_GROUP_20,P3_Group_19,Greetings from Akureyri group 20.
Appending 13 bytes to leftoverBuffer (current size: 0 bytes)

Command from server ORACLE: KEEPALIVE,0
Keepalive received from ORACLE but no messages
Appending 13 bytes to leftoverBuffer (current size: 0 bytes)
```

Mynd 1: Shows server side for when we got messages from Akureyri.

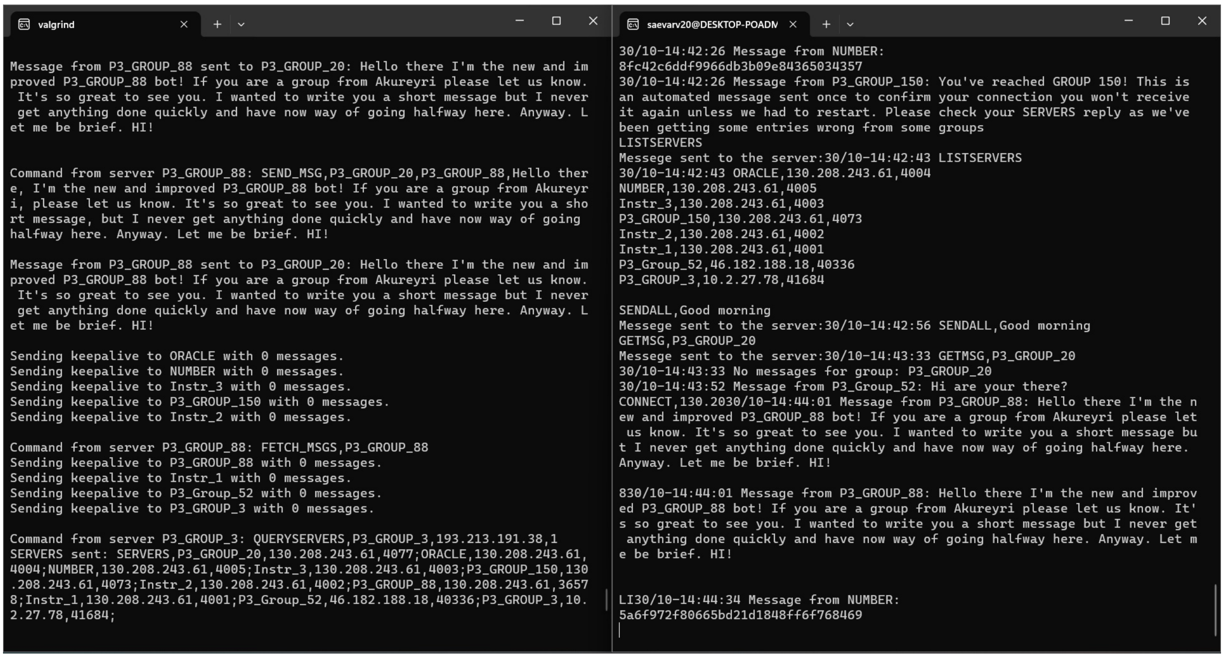


```
[katrinosk@Katrins-Air Assignment5 % ./chat_client 130.208.243.61 4077                              ]
Client identifier sent to server: SECRET_KATRIN
29/10-17:25:03 QUERYSERVERS,P3_GROUP_20,130.208.243.61,4077
CONNECT,130.208.243.61,4019
Messege sent to the server:29/10-17:25:14 CONNECT,130.208.243.61,4019
SENDMSG,P3_Group_19,Hello! I'm in Reykjavík, are you in Akureyri?
Messege sent to the server:29/10-17:27:07 SENDMSG,P3_Group_19,Hello! I'm in Reykjavík,
are you in Akureyri?
29/10-17:28:14 Greetings from Akureyri group 20.
LISTSERVERS
Messege sent to the server:29/10-17:32:48 LISTSERVERS
29/10-17:32:48 P3_GROUP_1,,-1;P3_Group_19,,-1;Instr_3,130.208.243.61,4003;ORACLE,130.20
8.243.61,4004;P3_GROUP_1,,-1;P3_GROUP_1,,-1;
29/10-17:35:14 Greetings from Akureyri.
```

Mynd 2: The client side for when we got messages from Akureyri

Decode a hashed Message

We received a hashed message from the server Number. The following code snippet can be found in the log files The message was $3ca14c518d1bf901acc339e7c9cd6d7f$ which can be decoded with MD5 decoding to "hardware". We get multiple other so I'm going to set up in a table.



```
Message from P3_GROUP_88 sent to P3_GROUP_20: Hello there I'm the new and im
proved P3_GROUP_88 bot! If you are a group from Akureyri please let us know.
 It's so great to see you. I wanted to write you a short message but I never
 get anything done quickly and have now way of going halfway here. Anyway. L
et me be brief. HI!


Command from server P3_GROUP_88: SEND_MSG,P3_GROUP_20,P3_GROUP_88,Hello ther
e, I'm the new and improved P3_GROUP_88 bot! If you are a group from Akureyr
i, please let us know. It's so great to see you. I wanted to write you a sho
rt message, but I never get anything done quickly and have now way of going
halfway here. Anyway. Let me be brief. HI!

Message from P3_GROUP_88 sent to P3_GROUP_20: Hello there I'm the new and im
proved P3_GROUP_88 bot! If you are a group from Akureyri please let us know.
 It's so great to see you. I wanted to write you a short message but I never
 get anything done quickly and have now way of going halfway here. Anyway. L
et me be brief. HI!

Sending keepalive to ORACLE with 0 messages.
Sending keepalive to NUMBER with 0 messages.
Sending keepalive to Instr_3 with 0 messages.
Sending keepalive to P3_GROUP_150 with 0 messages.
Sending keepalive to Instr_2 with 0 messages.

Command from server P3_GROUP_88: FETCH_MSGS,P3_GROUP_88
Sending keepalive to P3_GROUP_88 with 0 messages.
Sending keepalive to Instr_1 with 0 messages.
Sending keepalive to P3_Group_52 with 0 messages.
Sending keepalive to P3_GROUP_3 with 0 messages.

Command from server P3_GROUP_3: QUERYSERVERS,P3_GROUP_3,193.213.191.38,1
SERVERS sent: SERVERS,P3_GROUP_20,130.208.243.61,4077;ORACLE,130.208.243.61,
4004;NUMBER,130.208.243.61,4005;Instr_3,130.208.243.61,4003;P3_GROUP_150,130
.208.243.61,4073;Instr_2,130.208.243.61,4002;P3_GROUP_88,130.208.243.61,3657
8;Instr_1,130.208.243.61,4001;P3_Group_52,46.182.188.18,40336;P3_GROUP_3,10.
2.27.78,41684;
```

```
30/10-14:42:26 Message from NUMBER:
8fc42c6ddf9966db3b09e84365034357
30/10-14:42:26 Message from P3_GROUP_150: You've reached GROUP 150! This is
an automated message sent once to confirm your connection you won't receive
it again unless we had to restart. Please check your SERVERS reply as we've
been getting some entries wrong from some groups
LISTSERVERS
Messege sent to the server:30/10-14:42:43 LISTSERVERS
30/10-14:42:43 ORACLE,130.208.243.61,4004
NUMBER,130.208.243.61,4005
Instr_3,130.208.243.61,4003
P3_GROUP_150,130.208.243.61,4073
Instr_2,130.208.243.61,4002
Instr_1,130.208.243.61,4001
P3_Group_52,46.182.188.18,40336
P3_GROUP_3,10.2.27.78,41684

SENDALL,Good morning
Messege sent to the server:30/10-14:42:56 SENDALL,Good morning
GETMSG,P3_GROUP_20
Messege sent to the server:30/10-14:43:33 GETMSG,P3_GROUP_20
30/10-14:43:33 No messages for group: P3_GROUP_20
30/10-14:43:52 Message from P3_Group_52: Hi are your there?
CONNECT,130.2030/10-14:44:01 Message from P3_GROUP_88: Hello there I'm the n
ew and improved P3_GROUP_88 bot! If you are a group from Akureyri please let
 us know. It's so great to see you. I wanted to write you a short message bu
t I never get anything done quickly and have now way of going halfway here.
Anyway. Let me be brief. HI!

830/10-14:44:01 Message from P3_GROUP_88: Hello there I'm the new and improv
ed P3_GROUP_88 bot! If you are a group from Akureyri please let us know. It'
s so great to see you. I wanted to write you a short message but I never get
 anything done quickly and have now way of going halfway here. Anyway. Let m
e be brief. HI!


LI30/10-14:44:34 Message from NUMBER:
5a6f972f80665bd21d1848ff6f768469
```

Mynd 3: Hashed message to decode from Number

The sentence could be **Mikhail, not Boris, is receiving the hardware stuff that Anna Nikolai comes with.**

Message	Decoded
$21582c6c30be1217322cdb9aebaf4a59$	that
$3ca14c518d1bf901acc339e7c9cd6d7f$	hardwere
$37c09709af3da468f7a1bc723e943ec7$	mikhail
$d529e941509eb9e9b9cfaeae1fe7ca23$	not
$2091c76f726f21a61b6d2f8b885cc39d$	Boris
$e374307dc474b38fb89368677fbabfda$	Nikolai
$c13d88cb4cb02003daedb8a84e5d272a$	stuff
$0639f5c0e2228ccdf3385f88f1579491$	receiving
$97a9d330e236c8d067f01da1894a5438$	Anna
forgot to copy hash	the
forgot to copy hash	comes

Tafla 1: Hashing decode table

5 different groups you can show messages received from

The files in included contain 4 log files for a client and 1 for which when we connected to Akureyri. We have separate log files containing messages to and from groups in the Botnet. The groups are and can be found in: They are in total 21. We connected to one group in Akureyri. The logServer files include the log that was on the server side for the client with the same number in the filename. So the log for the server for the logClient4.txt is in logServer4.txt.

Log File	**Group**
LogClient1	38, 99, 150, 88, 69, 12, 8, NUMBER, ORACLE
LogClient2	17, 15, 89
LogCLient3	22, 100, 16, 39
LogClient4	44, 35, 36, 52
clientAkureyri	19

Tafla 2: Messages from groups

The security issues that may come up on the botnet

This botnet consists of servers made by students learning computer networking, meaning that this botnet most definitely has some security issues. We do not know how our peers might have coded their servers and which validations they may have included, so we are going to address some of the security issues that our server has.

First of all, the client connects by a certain security string, which is written in pure letters. The purpose of this was to only let our client connect to our server as a client, so that no other servers or clients can use client commands to control our server. The string used has just plain text and remains the same always. Hence, it is very susceptible to eavesdropping and someone else might make a server or a client that has this secret string to connect to our server and control it (not mentioning the fact that our code is on a github repository that is not private, which leaks our secret string). Some kind of an encryption could be implemented in our server and client to ensure secure communication. Messages between server is also not encrypted, so anyone can read the messages.

We do not have any validations for the group ids, so a server can be named anything as we store it in our connections list, like our own group id. If a server uses our group id, we might send our own mesages to that server, which may result in our client not getting our messages. Additionally, we do not have any validation for servers, so any attacker can pretend to be a server and send malicious data.

Our server accepts any connection and reads the data that is sent, not validating that data, putting a lot of trust in the server/client that connects. This makes our server very vulnerable to a denial of service attack, as anyone can send flow our server with connection requests or data. Moveover, the commands that are sent to our server are not sanitized well enough, risking a potential command injection attack that can inject malicious code into our server, that can crash it or control it.