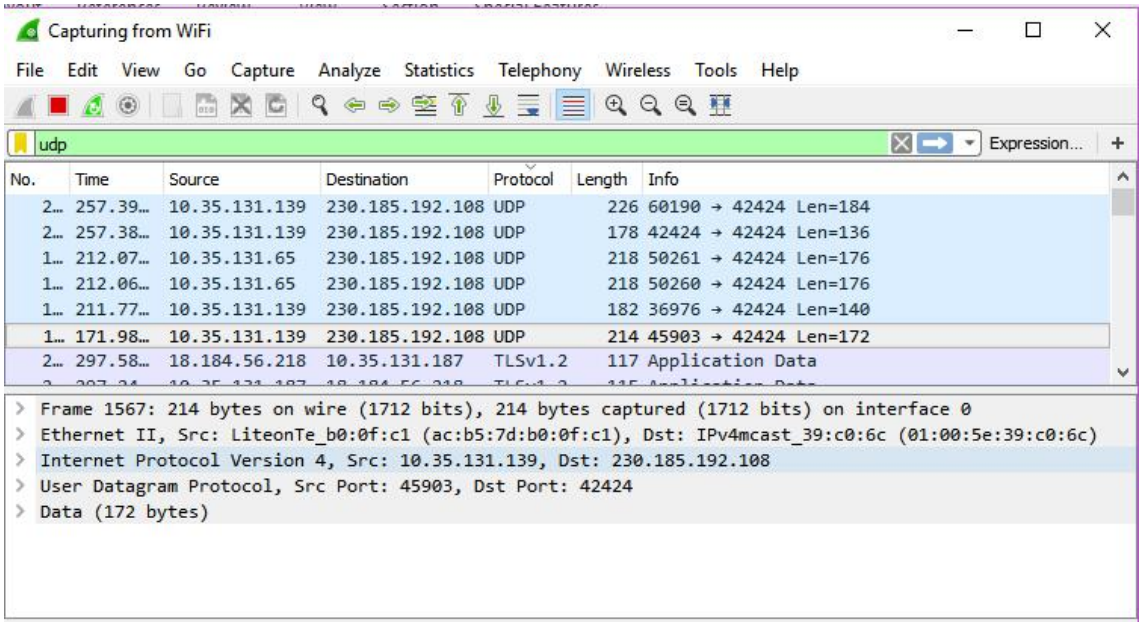
PR Lab 6

Protocol Description by reverse enginnering using WireShark

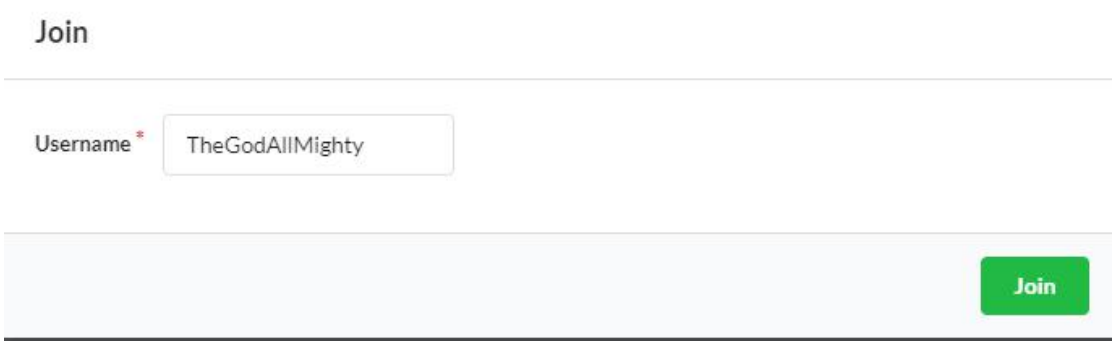
BizdigaStas FAF-151

Identifying packets:

By the teacher's tip, the protocol to look out for is udp, and as it's written down in the filters, we can see the following:

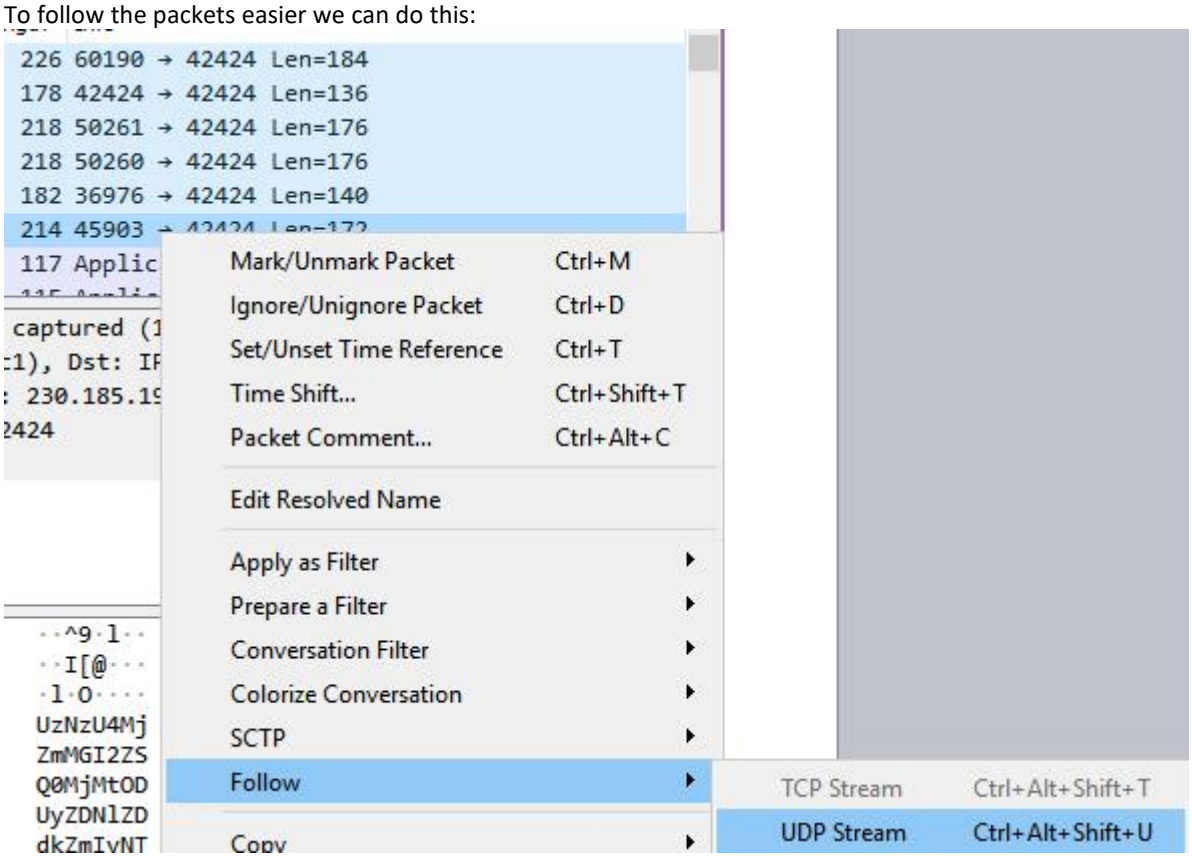


These are likely, messages from the other users on the app already. Let's try log in and see what happens...

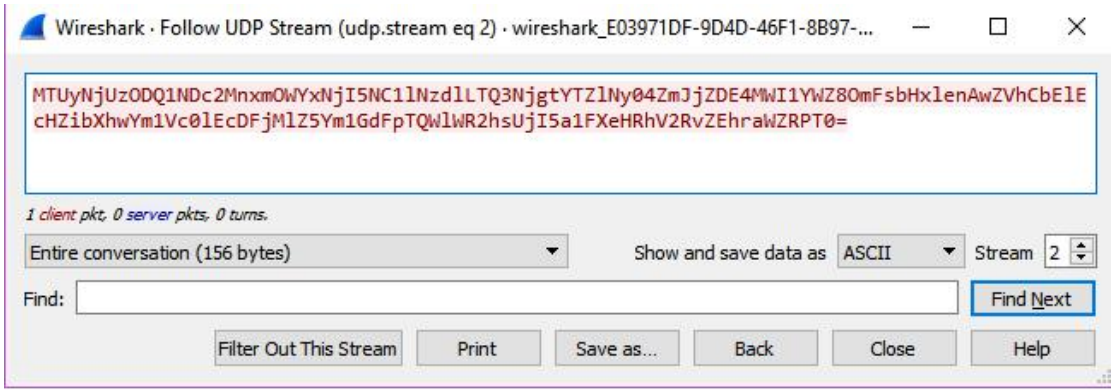


But before that, It would be nice to filter the other unnecessary packets by adding:
'udp.port == 42424'

To follow the packets easier we can do this:



Which leads to:



This looks like a base64 encoded string, so as i decode it, this is the answer:

Decode from Base64 format

Simply use the form below

MTUyNjUzODQ1NDc2MnxmOWYxNjI5NC1INzdILTQ3NjgtYTZINy04ZmJjZDE4MWI1YWZ8OmFsbHxlenAwZVhCbEIEcHZibXhwYm1Vc0IEcDFjMIZ5Ym1GdFpTQWIWR2hsUjI5a1FXeHRhV2RvZEhraWZRPT0=

DECODE | UTF-8 | You may also select input charset.

Live mode ON | Decodes while you type or paste (strict format).

Note that decoding of binary data (like images, documents, etc.) does not work in live mode.

UPLOAD FILE | Decodes an entire file (max. 10MB).

1526538454762|f9f16294-e77e-4768-a6e7-8fbcd181b5af|:all|ezp0eXBlIDpvbmxpbmUsIDp1c2VybmFtZSAiVGhlR29kQWxtaWdodHkifQ==

Then I take the last part (| are delimiting the parts) and decode it again.

Decode from Base64 format

Simply use the form below

ezp0eXBlIDpvbmxpbmUsIDp1c2VybmFtZSAiVGhlR29kQWxtaWdodHkifQ==

DECODE | UTF-8 | You may also select input charset.

Live mode ON | Decodes while you type or paste (strict format).

Note that decoding of binary data (like images, documents, etc.) does not work in live mode.

UPLOAD FILE | Decodes an entire file (max. 10MB).

{:type :online, :username "TheGodAlmighty"}

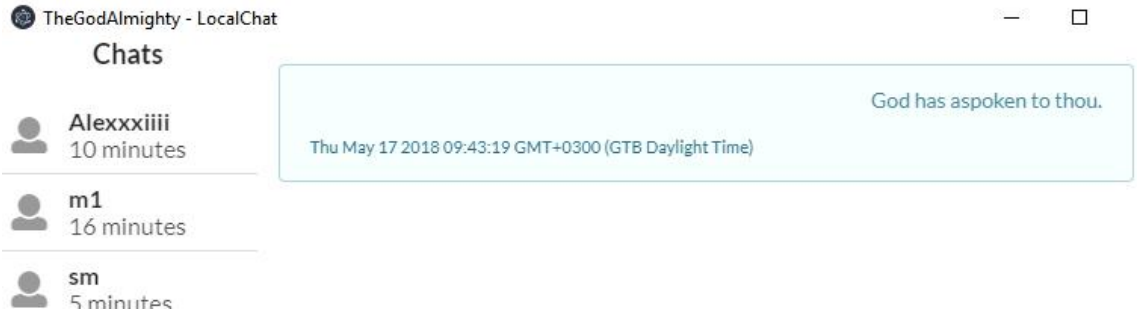
And I get the message from the system, telling that my user is now online.

After a bit of analysis I came to the conclusion that the first number,
aka:
1526538454762
is the time in miliseconds, this current value is Thu May 17 2018 9:27:34GMT+0300.

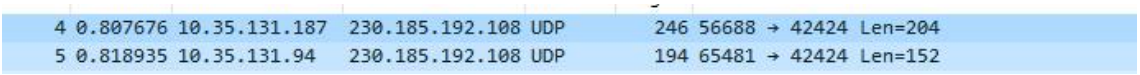
Next, is the uuid of the sender and receiver.
f9f16294-e77e-4768-a6e7-8fbcd181b5af - sender
:all - receiver

i.e. all receive this message, that my user is online.

Another example of message is when it is sent. Following the same procedure, we arrive at the same kind of result:



(sending message)



4	0.807676	10.35.131.187	230.185.192.108	UDP	246	56688 → 42424 Len=204
5	0.818935	10.35.131.94	230.185.192.108	UDP	194	65481 → 42424 Len=152

we get 2 packets

one contains the message and the other one the acknowledgement - a kind of notification that it was delivered.

MTUyNjUzOTM5OTI1MHxmOWYxNjI5NC1INzdILTQ3NjgtYTZINy04ZmJjZDE4MWI1YWZ8NzVjZjk0YzEtMTA1I
IWEJsSURwamFHRjBMQ0E2ZEhoMEIDSkhiMIFnYUdGekIHRnpjRzIyWIc0Z2RHOGdkR2h2ZFM0aWZRPT0=

decoded:
1526539399250 - time in milisec
f9f16294-e77e-4768-a6e7-8fbcd181b5af - sender uuid
75cf94c1-1054-4b4c-8be0-568c1eaba65f - receiver uuid
ezp0eXBlIDpjaGF0LCA6dHh0ICJHb2QgaGFzIGFzcG9rZW4gdG8gdGhvdS4ifQ== - message

message decoded:
{:type :chat, :txt "God has aspoken to thou."}

Next packet:
MTUyNjUzOTQwMTU0M3w3NWNmOTRjMS0xMDU0LTRiNGMtOGJlMC01NjhjMWVhYmE2NWZ8ZjlmMTYyOTQtZTc3ZS00NzY4LWE2ZTctOGZiY2QxODFiNWFmfGV6cDBlWEJsSURwa1pXeHBkbVZ5WldSOQ
==

1526539401543 - time
75cf94c1-1054-4b4c-8be0-568c1eaba65f -sender
f9f16294-e77e-4768-a6e7-8fbcd181b5af -receiver
ezp0eXBlIDpkZWxpdmVyZWR9 - msg

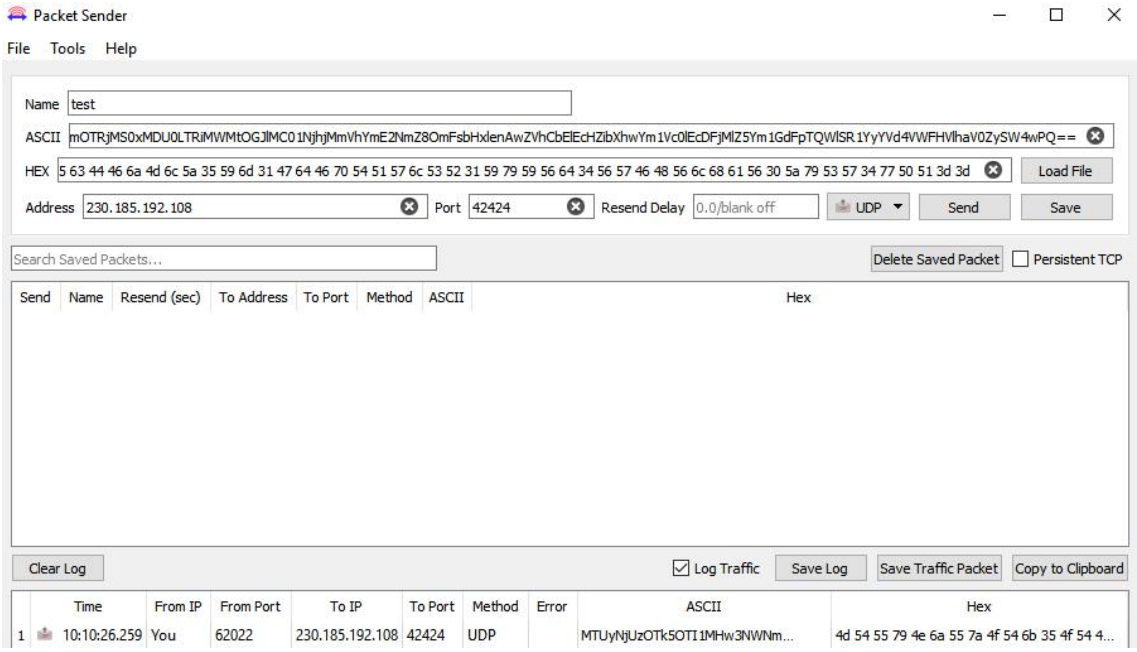
message decoded:
{:type :delivered}

Since the protocol is clear we can send a message using wireshark too.
It is done by creating a fake user first:

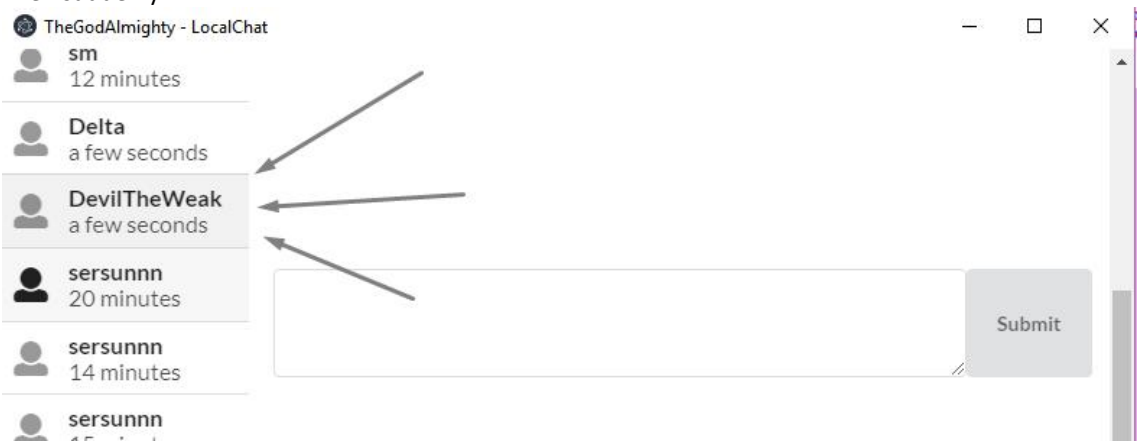
1. Create our name. The name for new user will be DevilTheWeak. So the message is: `{:type :online, :username "DevilTheWeak"}`
2. Now we need to encode it to Base64. The result will be `ezp0eXBlIDpvbmxpbmUsIDp1c2VybmFtZSAiRGV2aWxUaGVXZWFrIn0=`
3. We need to rebuild the previous packet structure now: (new time and new uuid) 1526539999250 | 75cf94c1-1054-4b1c-8be0-568c2eaba66f | `:all|ezp0eXBlIDpvbmxpbmUsIDp1c2VybmFtZSAiRGV2aWxUaGVXZWFrIn0=`
4. Now we need to encode the contents to Base64: `MTUyNjUzOTk5OTI1MHw3NWNmOTRjMS0xMDU0LTRiMWMtOGJlMC01NjhjMmVhYmE2NmZ8OmFsbHxlenAwZVhCbElEcHZibXhwYm1Vc0lEcDFjMlZ5Ym1GdFpTQWlSR1YyYVd4VWFHVlhaV0ZySW4wPQ==`
5. We can send the packet now. More of the data that weneed can be found in wireshark by looking at the packet, such as : destination ip, port.

Then sending the message:

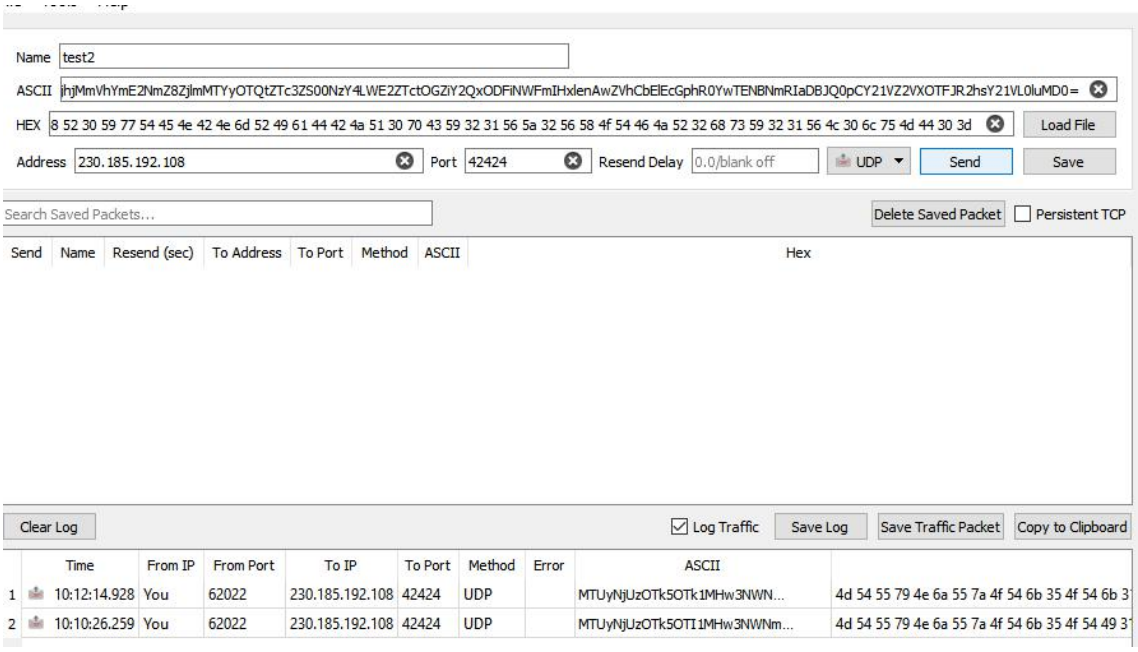
1. Encoding the text message : `{:type :chat, :txt "Are you here?"}` to Base64 : ezp0eXBlIDpjaGF0LCA6dHh0ICJBcmUgeW91IGhlcmU/In0=
2. Getting UUID of our fake user :75cf94c1-1054-4b1c-8be0-568c2eaba66f
3. Finding a UUID for to who send the message, we will use the UUID of the first user we created f9f16294-e77e-4768-a6e7-8fbcd181b5af (god)
4. Making mymessage : 1526539999950 | 75cf94c1-1054-4b1c-8be0-568c2eaba66f | f9f16294-e77e-4768-a6e7-8fbcd181b5af | ezp0eXBlIDpjaGF0LCA6dHh0ICJBcmUgeW91IGhlcmU/In0=
5. Econding it to Base64 : MTUyNjUzOTk5OTk1MHw3NWNmOTRjMS0xMDU0LTRiMWMtOGJlMC01NjhjMmVhYmE2NmZ8ZjlmMTYyOTQtZTc3ZS00NzY4LWE2ZTctOGZiY2QxODFiNWFmIHxlenAwZVhCbElEcGphR0YwTENBNmRIaDBJQ0pCY21VZ2VXOTFJR2hsY21VL0luMD0=
6. Sending it using Packet Sender.

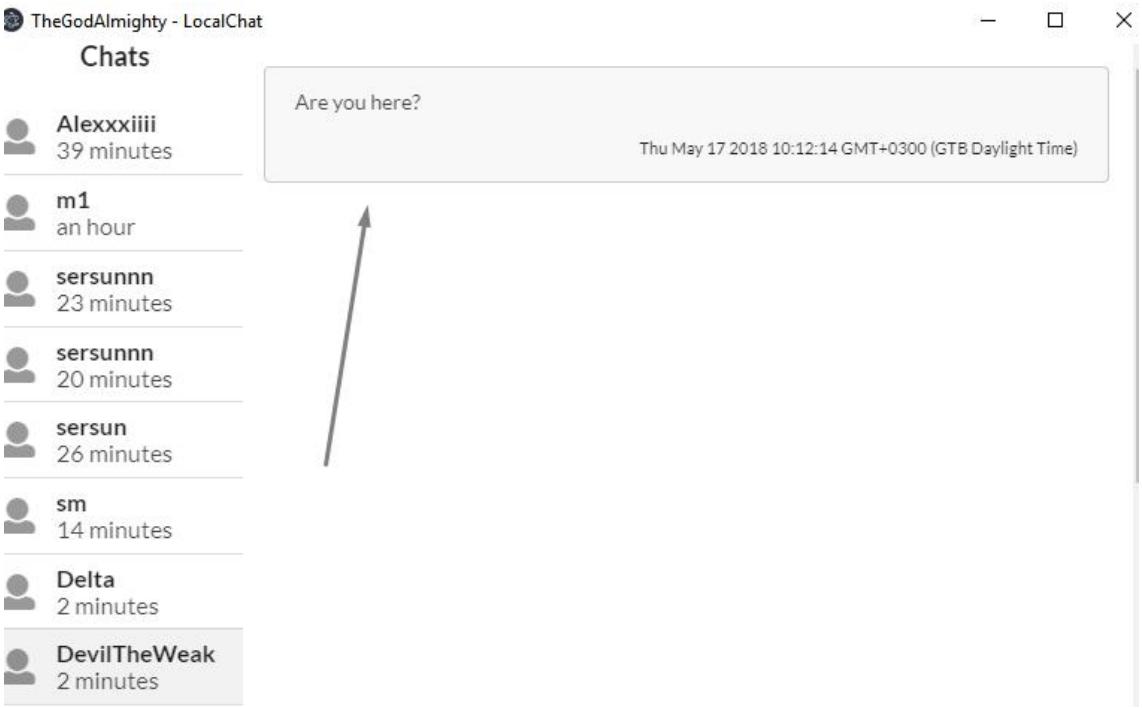


Then suddenly:



NEXT MESSAGE:





It works!!

Creepy.