Discrete simulation of routing

Discrete simulation of routing in C# with Windows Forms App GUI. For current progress, visit Task management. For more details, visit project Wiki.

Goal

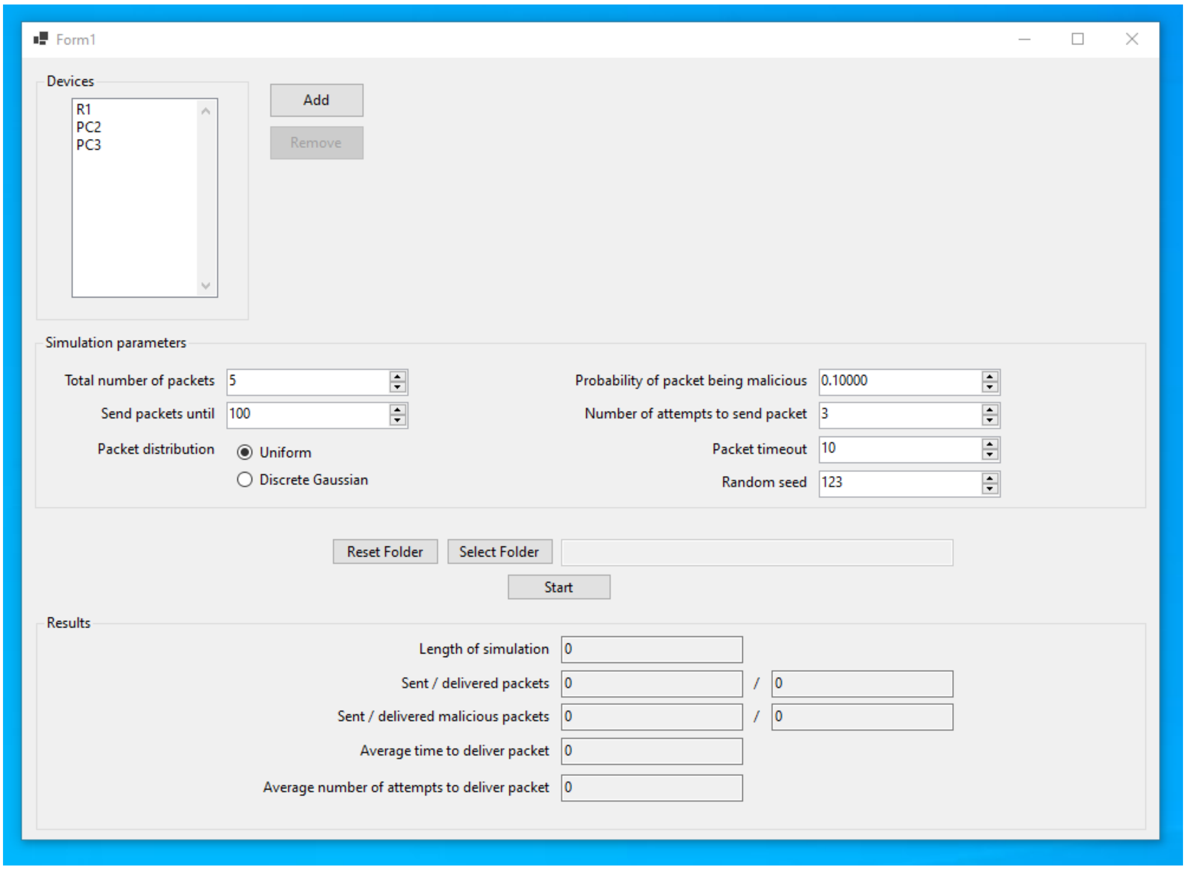
The goal of this project is to simulate routing in a static computer network.

Installation and running

Clone this repository and open `semestralka-routing-simulation` as project in Visual Studio. Build the project as Release and head over to `bin/Release/netcoreapp3.1` and run `semestralka-routing-simulation.exe`.

Example of use

After starting the program, GUI should show up.



Devices

R1
PC2
PC3

Simulation parameters

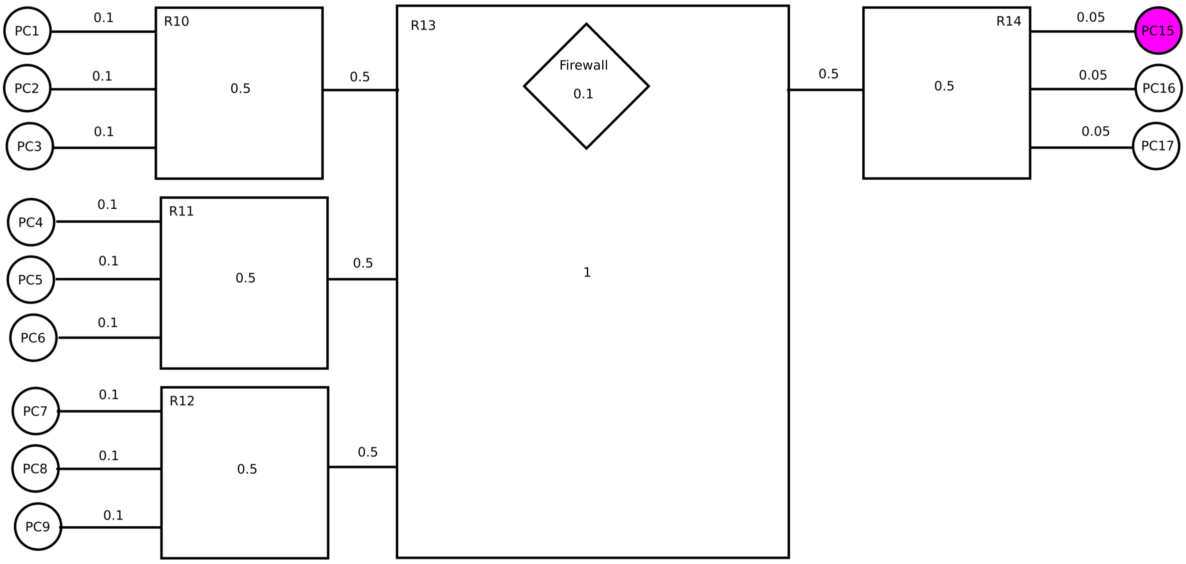
Total number of packets: 5
Send packets until: 100
Packet distribution: ● Uniform ○ Discrete Gaussian
Probability of packet being malicious: 0.10000
Number of attempts to send packet: 3
Packet timeout: 10
Random seed: 123

Results

Length of simulation: 0
Sent / delivered packets: 0 / 0
Sent / delivered malicious packets: 0 / 0
Average time to deliver packet: 0
Average number of attempts to deliver packet: 0

An example of network is pre-loaded, it is possible to run simulation right away and see the results at the bottom. The network is very simple, so it shouldn't be hard to skim through the devices and understand it. Here we will try to implement and simulate more complex network.

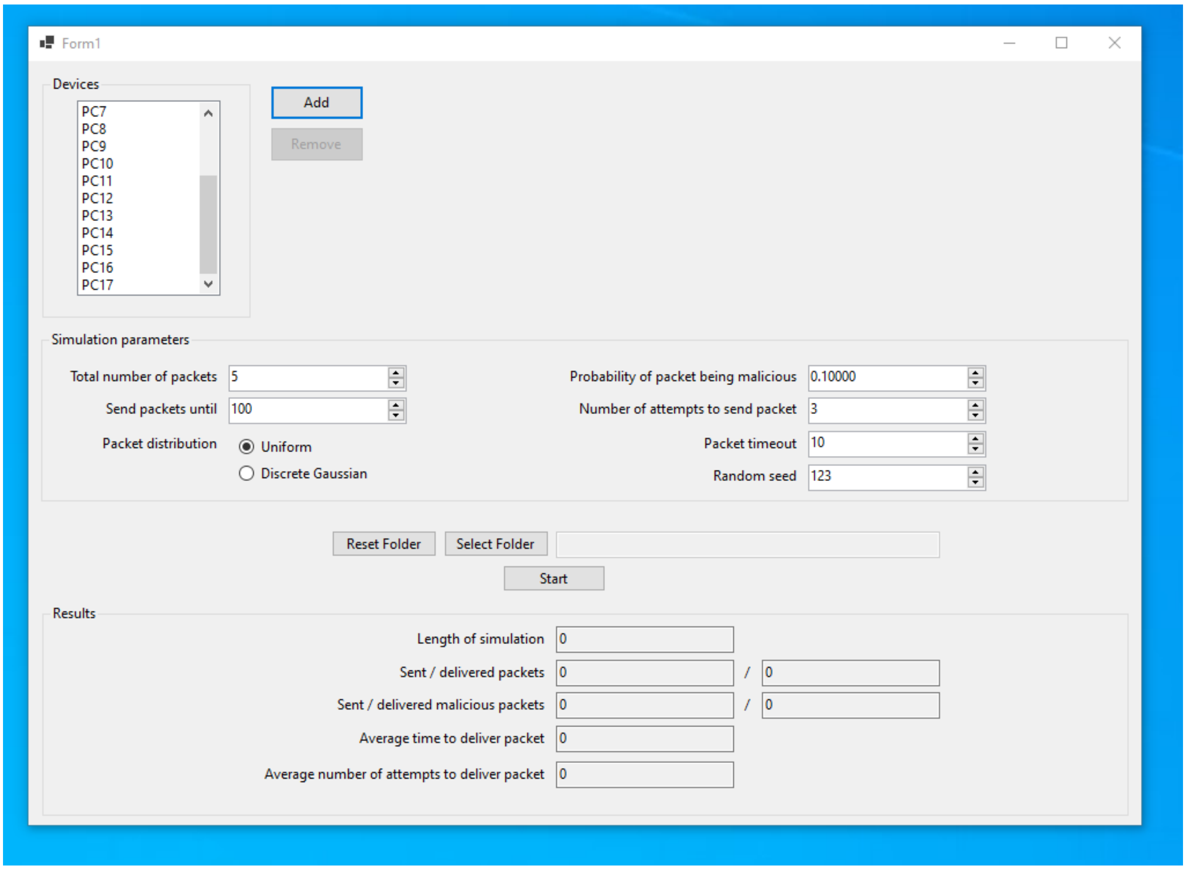
More specific description of individual controls can be found on wiki. Let us consider the network depicted below.



Numbers represent speed of links, routers and firewalls (how fast do they transfer / process packets), in order to be closer to reality we can assume these values are in Gigabits per second. A purple computer is infected computer, ie it is a source of malicious packets - let us assume that about 1 in 20 packets it sends is malicious.

Because our program doesn't use speed as a metric, we are going to inverse the values - that way, we acquire how many seconds it takes to transfer a Gigabit of data (or equivalently how much many milliseconds does it take to transfer a Megabit of data), which is a certain metric of *slowness*.

Let's start with adding all the needed devices. After starting the program, there are 3 predefined devices, so all we need to do is add another 14 simply by clicking Add repeatedly.



Devices

PC7
PC8
PC9
PC10
PC11
PC12
PC13
PC14
PC15
PC16
PC17

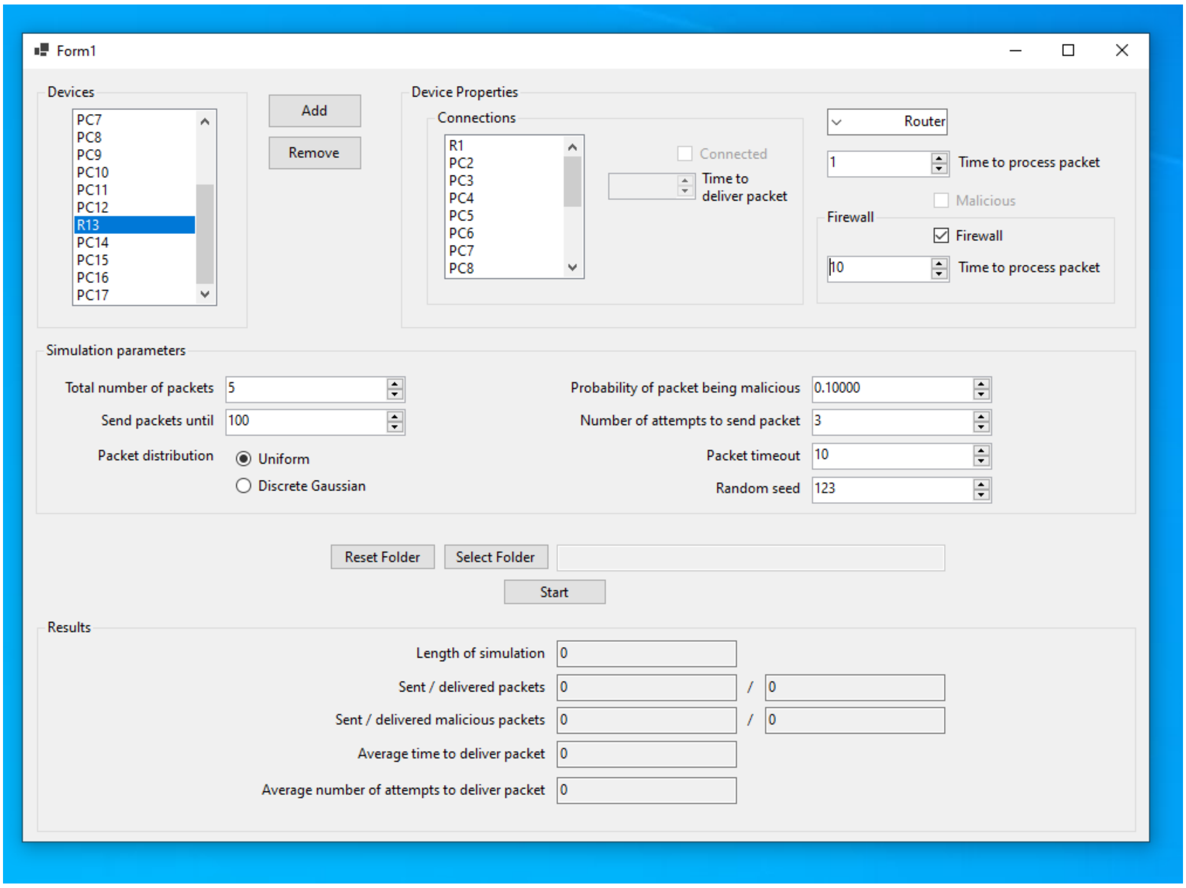
Simulation parameters

Total number of packets	5
Send packets until	100
Packet distribution	● Uniform ○ Discrete Gaussian
Probability of packet being malicious	0.10000
Number of attempts to send packet	3
Packet timeout	10
Random seed	123

Results

Length of simulation	0	
Sent / delivered packets	0	0
Sent / delivered malicious packets	0	0
Average time to deliver packet	0	
Average number of attempts to deliver packet	0	

We won't be showing how to setup all the devices, but without loss of generality, let's do R13. First we click on PC13, in Device properties we change it from Computer to Router, change its' time to process packet to 1/1 = 1, check Firewall and add time to process packet by firewall to 1/0.1 = 10.



□ Connected
Time to deliver packet
Router
Time to process packet: 1
□ Malicious
Firewall
☑ Firewall
Time to process packet: 10

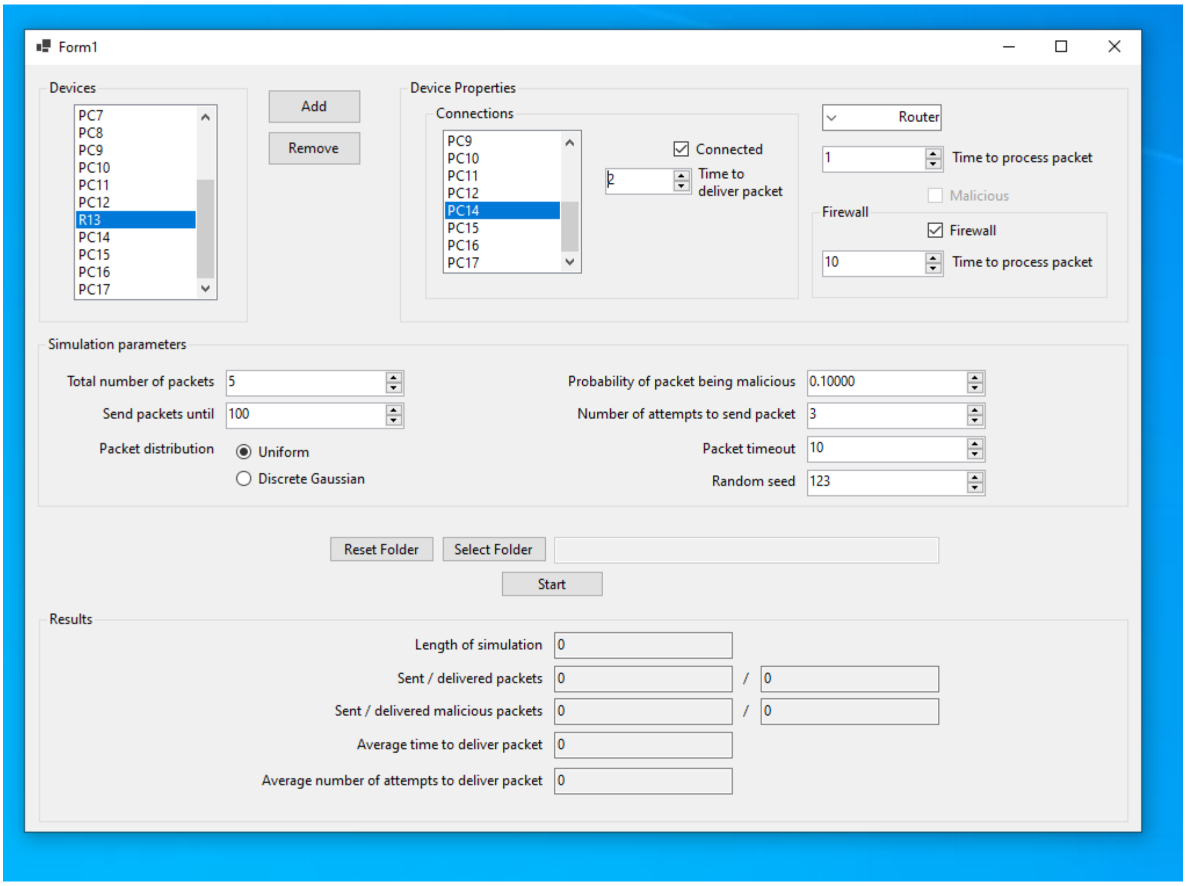
Simulation parameters

Total number of packets: 5
Send packets until: 100
Packet distribution: ● Uniform ○ Discrete Gaussian
Probability of packet being malicious: 0.10000
Number of attempts to send packet: 3
Packet timeout: 10
Random seed: 123

Results

Length of simulation: 0
Sent / delivered packets: 0 / 0
Sent / delivered malicious packets: 0 / 0
Average time to deliver packet: 0
Average number of attempts to deliver packet: 0

Looking at the picture of network, we can see it is connected to R10, R11, R12, R14. We did not create those yet, but we already know these are currently PC10, PC11, PC12, PC14, so we are going to connect R13 to those. To connect a device, select it in Connections, check Connected and set a time to deliver. Again, we are not going to show all the connections, but let's look for example at connection R13 to R14.



☑ Connected
Time to deliver packet: 2
Router
Time to process packet: 1
□ Malicious
Firewall
☑ Firewall
Time to process packet: 10

Simulation parameters

Total number of packets: 5
Send packets until: 100
Packet distribution: ● Uniform ○ Discrete Gaussian
Probability of packet being malicious: 0.10000
Number of attempts to send packet: 3
Packet timeout: 10
Random seed: 123

Results

Length of simulation: 0
Sent / delivered packets: 0 / 0
Sent / delivered malicious packets: 0 / 0
Average time to deliver packet: 0
Average number of attempts to deliver packet: 0

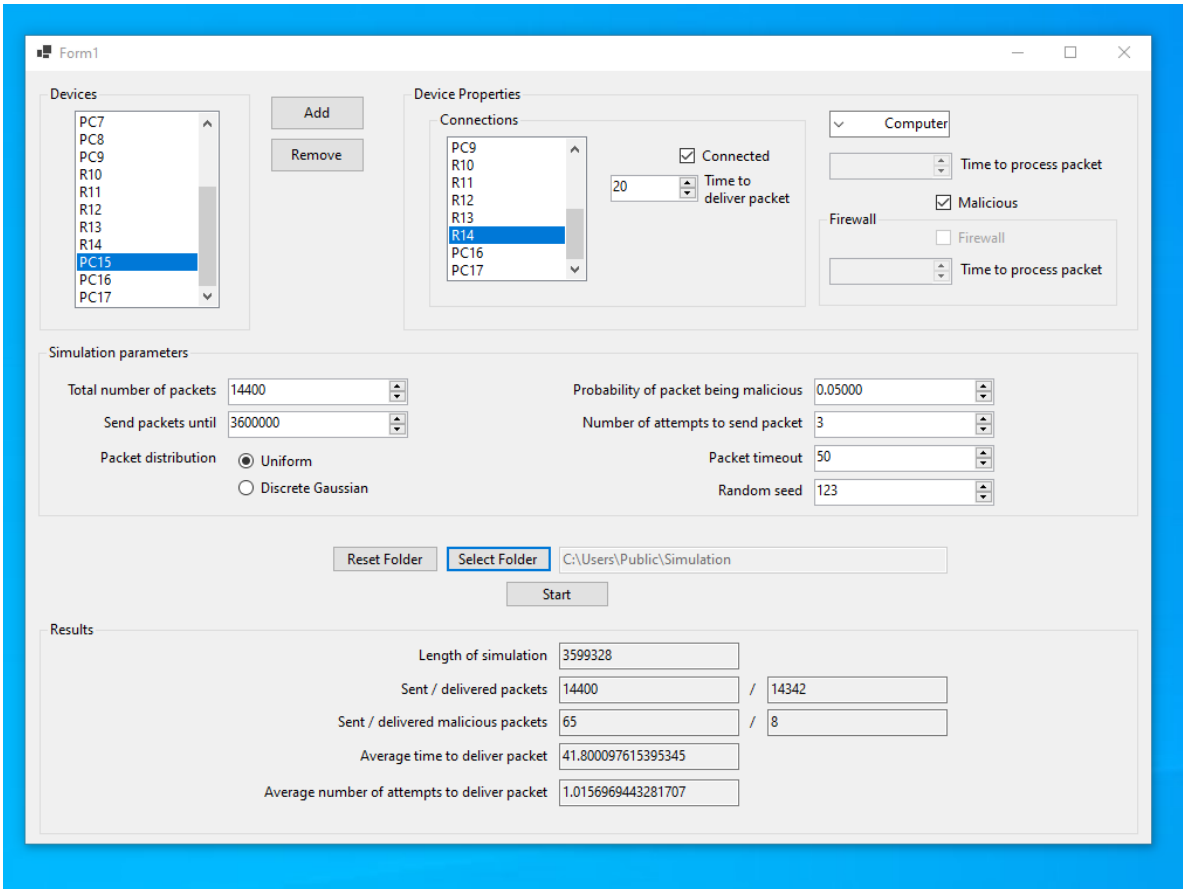
After doing this for all devices, we can move on to simulation parameters (don't forget to check if there are no extra connections / other settings left of from predefined example). Again, we are going to attempt to model some scenario at least remotely close to reality.

Say we want to simulate single hour of traffic inside this network. Consider the *slowness* we defined as how many milliseconds does it take to transfer single Megabit. That gives us 60 * 60 * 1000 = 3,600,000 (amount of milliseconds in single hour) for variable Send packets until. How many packets are sent in an hour from 12 computers? Well again, according to *slowness* as we defined it, one packet contains (uh) 1 Megabit of data. Considering 1 computer would send 20 Megabits of data per minute, this gives us 12 * 20 * 60 = 14400 total packets.

Because we are considering only single hour of traffic, it doesn't make much sense to choose different distribution than uniform (Gaussian distribution would make sense for example when modelling a whole day of traffic). As we've decided earlier, probability of packet being malicious will be around 1 in 20.

We will give every packet 3 attempts to be delivered. The longest it can possibly take a packet to go from one PC to another one without any delays on its' path is 49 milliseconds - making timeout to be 50 is a bit cruel, but the packet has 2 more tries to be delivered. Also the density of packets in time (about one packet for every 250 milliseconds) is very low, so it shouldn't cause any trouble.

Last but not least, you can select a folder where to store results. In our case, we will select `C:\Users\Public\Simulation`. Now we can click Start and let the simulation happen.



☑ Connected

Time to deliver packet: 20

Time to process packet:

☑ Malicious

□ Firewall

Time to process packet:

Simulation parameters

Total number of packets: 14400

Send packets until: 3600000

Packet distribution: ● Uniform ○ Discrete Gaussian

Probability of packet being malicious: 0.05000

Number of attempts to send packet: 3

Packet timeout: 50

Random seed: 123

C:\Users\Public\Simulation

Results

Length of simulation: 3599328

Sent / delivered packets: 14400 / 14342

Sent / delivered malicious packets: 65 / 8

Average time to deliver packet: 41.800097615395345

Average number of attempts to deliver packet: 1.0156969443281707

Now we can look at the results and draw some conclusions. Looking at the statistics of packets, we can see that only single (non-malicious) packet wasn't delivered at all. At the same time, there were a few malicious packets that reached another computer. Overall the network seems to be well suited for the traffic we presented it with. We can also see that it took on average about 42 milliseconds to deliver a packet and that most of the packets have been delivered on first attempt.