DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master Practical Course
Computer Network Simulation

26.01.2023

Assignment 4

Part 2 - Results

Group 2

Zafer Tan Cankiri
Erdem Ege Marasli

Summary and Interpretation of The Collected Data:

We ran our hypothetical messaging application for the inter-bunker communication on a bunker network in OMNeT++ for 300 seconds with different configurations.

During the simulation, the following values are recorded as vectors:

- The number of successful lookup requests (that the server answers with a record) sent from the clients to the server.
- The number of unsuccessful lookup requests (that the server answers with a not found error) sent from the clients to the server.
- Link Layer Throughput for each host, router, and the server.
- Application Layer Throughput for each host, router, and the server.
- Request-Response/Communication Latency for each host, router, and the server.
- Link Utilization for each host, router, and the server.

In this report, we will investigate some examples of the given records. Much more results are generated with our pipeline which can be investigated for detailed analysis.

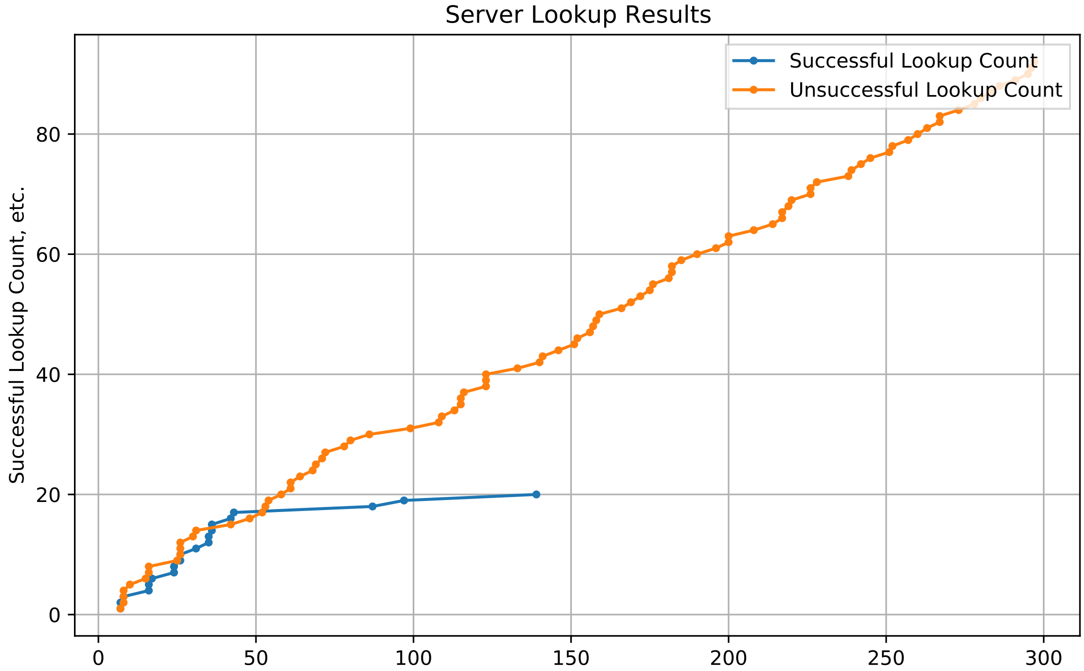

Figure 1: Server Lookup Results

Each host in the bunkers keeps a list of the people they are looking for. In addition, each host has an address book and they save the host they learn in these address books. To learn a host, they send a lookup message to the server. If a host knows all of the people that it is looking for, then it stops to send lookup messages and the whole system converges at a point in terms of sending lookup messages. However, to make it more interesting, we gave some names that are not exist in any of these bunkers to some of the hosts. Therefore, they keep looking for them until the end of the simulation. However, as we can also see in Figure 1, the number of successful lookup operations converges to a point and ends at a time since all hosts learns the people they can learn. There will be no more need to send lookup messages after this points.

Figure 2: S. Lookup: General

Figure 3: S. Lookup: ClearHeartBeatDropObs.

Normally heartbeat messages are needed to be renewed at least every 15 seconds. If one host does send a new heartbeat message after 15 seconds, the server accepts that the given host is disconnected from the system, so the lookup request fails. With the ClearHeartBeatDropObservation configuration, we decreased this time to 3 seconds. Figure 2 shows the result for the general configuration in which the threshold is 15 seconds. Figure 3 shows the result when the threshold is 3 seconds. When we investigate both figures we can clearly see that unsuccessful lookups are much higher for the 3-second threshold configuration as expected since much more hosts will be accepted as disconnected. Also, the 3-second threshold configuration converges later.

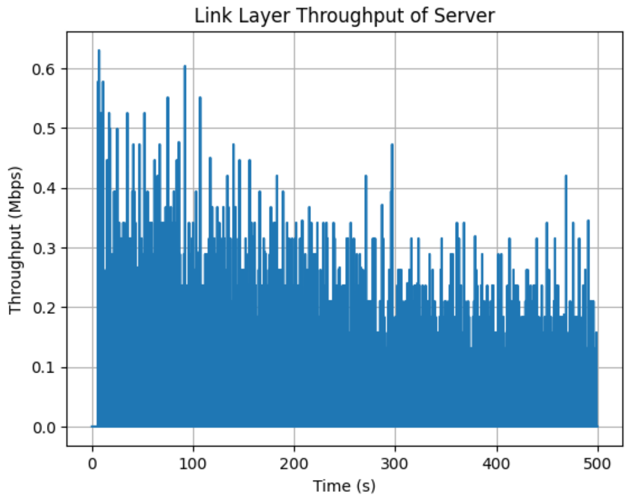

Figure 4: LinkLayer Throughput with 120 clients

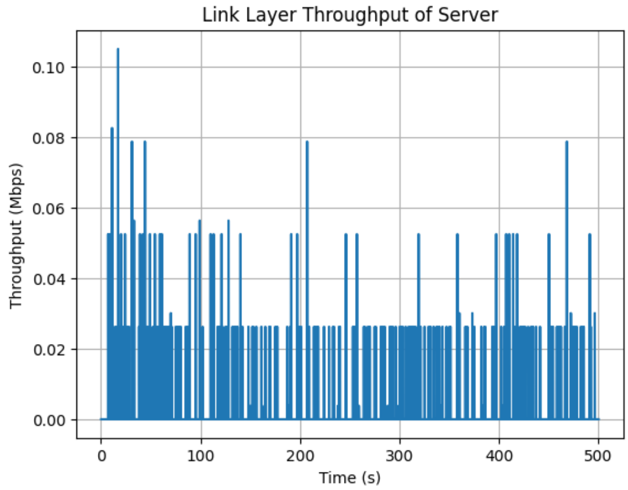

Figure 5: LinkLayer Throughput with 10 clients

As can be seen in Figure 4 and Figure 5, there is a high correlation between the number of clients and the server throughput. Since we have only one server in our system, this server should serve all of the clients in all of the bunkers. Therefore, when the number of clients increases, the total load on this single server also increases which shows the risk of a single point of failure and the necessity of load distribution.

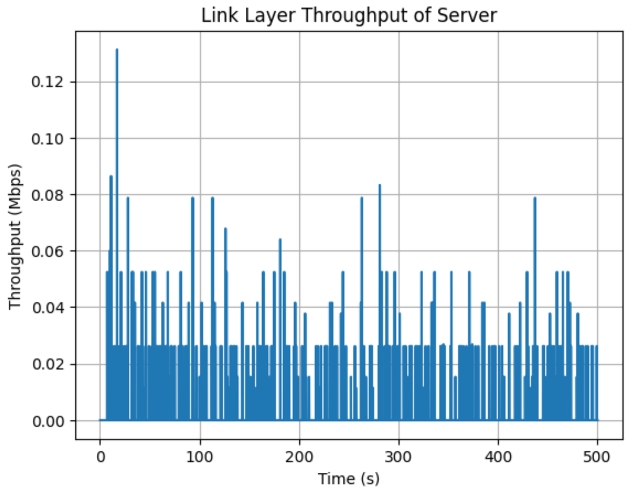

Figure 6: LinkLayer Throughput (Centralized)

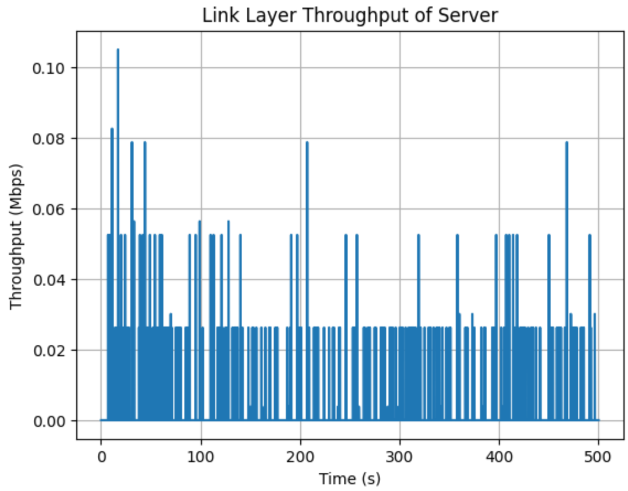

Figure 7: LinkLayer Throughput with Mec

Only the WarnerAdmin app which sends alert messages can run with MEC in our current implementation. Our other applications run on the server no matter if MEC support exists or not. If there is MEC support WarnerAdmin app runs on the MEC. When we investigated the graphs we can see that the throughput of the server with MEC is slightly lower than the centralized server architecture. This is the expected behavior since with the MEC, 3 different MEC hosts handle the Warning alert messages not a single server. Since the client number is kinda low and the interval for the WarnerAdmin app is higher than the other apps observed differences are not that much but still observable. If the interval for the WarnerAdmin app is configured really low, the difference would be severe.

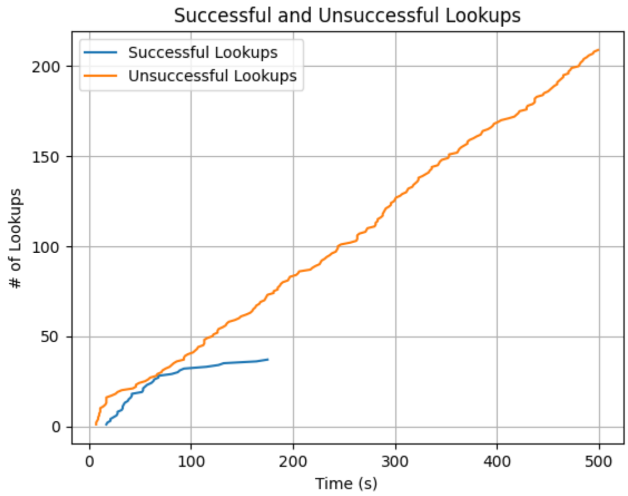

Figure 8: S. Lookup: Centralized Server

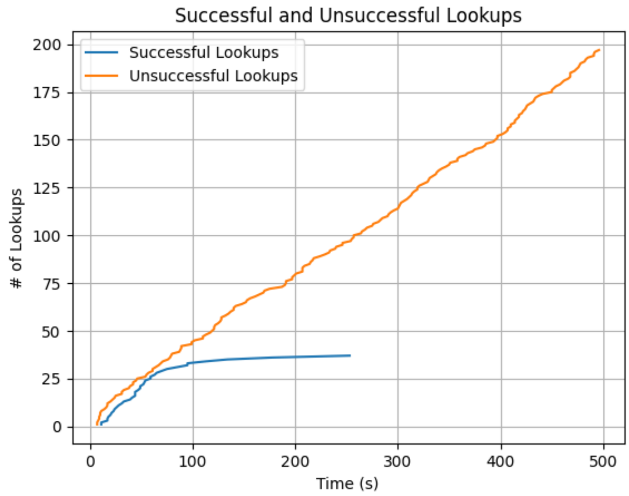

Figure 9: S. Lookup: MEC

Only the WarnerAdmin app which sends alert messages can run with MEC in our current implementation. Our other applications run on the server no matter if MEC support exists or not. So our heartbeat and lookup requests go to the server all the time. Because of this behavior, we expected the same successful and unsuccessful lookup graph on both MEC and Centralized server architecture. The basic logic seems really similar when we compared the graphs, however, there is still some small difference which we didn't understand the reason for. Since the LCM Proxy ("ualcmp") and the server share the same link over the mainrouter. The extra traffic required to communicate with the LCM proxy may cause some congestion on the link and affect the timing of the lookup results. However, we are not 100% sure about this and further experimentation may be required.