# Gateway

## Acknowledgements & ARQ

Your task is to design a gateway that forwards data from a sensor to a server. The aim of the assignment is to get to know sockets, datagram packets and threads and to design a protocol i.e. packet layout and packet handling, for the communication between two nodes.

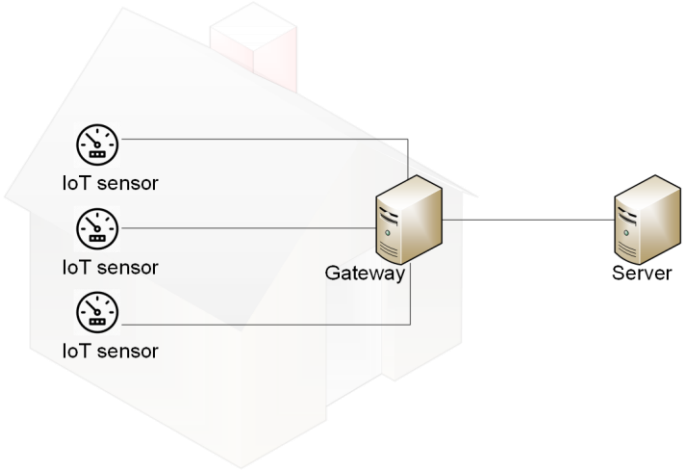


**Figure 1:** Example topology that demonstrates the connection of a number of sensors in the Internet of Things (IoT) through a gateway to a server. A possible real-world example for this would be a number of water meters that use 802.15.4 to connect through a 6LoWPAN gateway to an MQTT broker at datacentre provider.

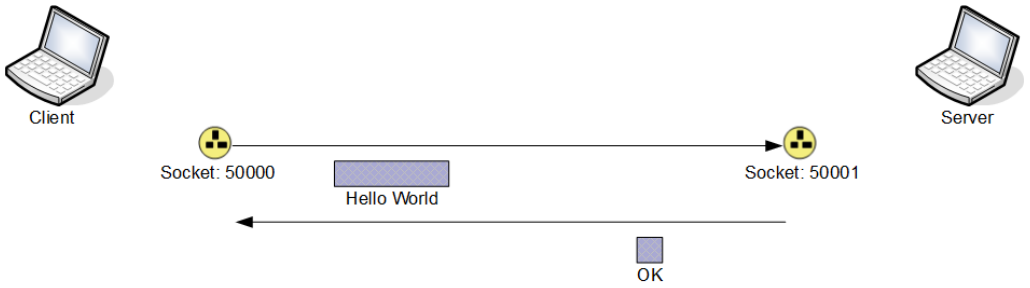


**Figure 2:** Current simplistic implementation

The current implementation provides functionality that transmits a packet from a client to a server, containing a String entered by the user. The server responds to incoming packets with a packet containing the String “OK”. The Client opens a port on port number 50000, creates a packet from the input provided by the user and sends this packet to another port on the local machine with the port number 50001.

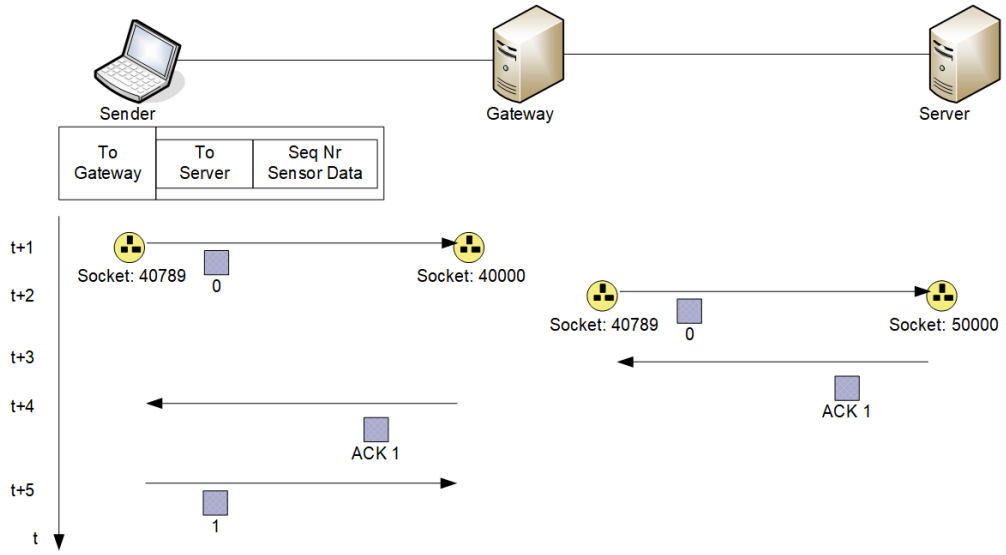


**Figure 3:** Scenario that your implementation should address

Your implementation should allow a sender node to send a packet to a gateway with a packet that should be forwarded to a server. The server would reply to the receipt of a packet with an acknowledgement. The packet from the sender should carry a sequence number and the acknowledgement should carry the sequence number that the server expects next. The sender should resend the packet if it has not received an acknowledgement after a given time.

The setSoTimeout method of the DatagramSocket class causes an exception to be thrown if a receive method has not returned within a given time. This timeout mechanism may form the basis for a timer mechanism.

As an extension to the simplistic version of one sender communicating with one server, your extended implementation should allow for the gateway and the server to handle multiple clients at any given time and the server to send a negative acknowledgement to the client if it receives a packet with a segment number that it did not expect.

The implementation should be accompanied by a report that explains the design and implementation of the protocol, the choices that you have made and the advantages and disadvantages that these decisions introduced. The description of the design should be accompanied by snapshots of some of the packets that were transmitted by your implementation. The explanations of the packets should highlight the management information in these packets and illustrate how this information is used by your implementation. The report should conclude with a reflection on the assignment as a whole, what went well for you and what you could have done better, and an estimation of the time that you spent on the assignment.

## Submission Details

The files that contain the implementation and the report should be submitted through Blackboard. Every file should contain the name of the author and the student number. The source files of the implementation should be submitted as an archived file e.g. ".zip" or ".tar.gz". The report should be submitted as either word- or pdf-document.

The name of the archive file and the report should be include the name and the student number of the author; for example, "123456-John-Doe-FlowC-implement.zip" and "123456-John-Doe-FlowC-Report.pdf" where the name of the student is John Doe and the student # is 123456. The deadline for the submission is given in Blackboard.

## Marking Scheme

The marks for the assignment will be split into 60% for the simplistic implementation of a client communicating through a gateway with a server and 40% for the implementation of a gateway and a server that handle multiple clients and a server that handles unexpected sequence numbers. The marks for the implementations will be split 50% for the implementation and 50% for the documentation through the report.