SENSOR NETWORKS LAB (PR WS14/15)

Lab 4: Advanced UDP communication

Sanjeet Raj Pandey [313631] Sanjay Santhosh Kumar Muhammad Tauseef Khan [346301] Rajibul Alam Khan

> Supervisor: Vlado HANDZISKI Tim BORMANN



Telecommunication Networks Group Technical University Berlin Excercise 4.1: Echo and integers Repeat the instructions from the beginning of the chapter and again try to enter a word and numbers. You will see that the numbers will not be displayed properly, because netcat will interpret them as characters, not as numbers.

Entered digit is sent as ASCII code and in big endian order e.g 6 is sent as 360a while returning it via ECHO code as integer 06000000 shows in wireshark. While after converting digit from host to network order digit is sent as 00000006. One can see how endiness gets swapped while sending integers not to forget UDP network protocol is based on big-endian.

Excercise 4.3: *Memory Layout* Give an example for a 1-byte, 2-byte and 4-byte alignment. Use the following sequence: 16 bit, 16 bit, 8 bit, 32 bit and 16 bit.

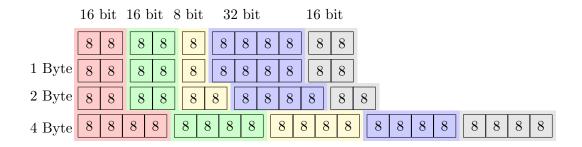


Figure 1: Alignment in General way for 16,16,8,32,16 bits arrangement

Memory alignment must always be taken care of during lower level programming. Depending on the architectures its can have different byte alignment system as in our example we are checkign for 1 byte, 2 byte and 4 byte alignment. As for 1 byte alignment every 8 bit is read one after another. But for 2 byte alignment third block with 8 bit makes reading next byte unaligned so padding of 1 byte is added depending on big endian after 8 bit cell or for little endian before 8 bit cell. In 4 byte alignment 1st two 16 bit cell and last 16 bit cell gets padded with 2 more byte as well as 8 bit cell on third gets padded with 3 more byte also respect to endianess of system.