OBIEKTY INTERNETU RZECZY

Laboratorium 1 zdalne
Temat: Komunikacja z węzłami wbudowanymi Internetu Rzeczy z wykorzystaniem protokołu UDP.
Imię i nazwisko: **Krzysztof Aleksander Pierczyk**

Warunki wstępne:
Warunki wstępne: a)zapoznać się z rozszerzeniem Copper dla przeglądarki Mozilla-Firefox, do interakcji z serwerem CoAP, b)zapoznać się z podstawami programowania sieciowego na platformie Arduino, (c) zapoznać się z obsługą programów do diagnostyki aplikacji sieciowych (Netcat, Wireshark z zainstalowanym pakietem Npcap – domyślna procedura).

Zadanie 1. Interakcja z serwerem CoAP za pomocą rozszerzenia Copper dla przeglądarki WWW.
Dla zapoznania się z protokołem CoAP przygotowano wzorcowe obrazy pamięci kodu dla platformy emulowanej za pomocą EBSimUnoEth, o nazwie ObirEth_coapserver_20200401.ino.hex i umieszczone w repozytorium GIT w: EBSimUnoEth\bin-dist\wsady-testowe\coap-server
Za pomocą specjalnie przygotowanej przeglądarki Mozilla Firefox[1] i zainstalowanej w niej wtyczce Copper, należy odczytać listę i atrybuty zasobów serwera CoAP oraz za pomocą metody GET odczytać treści przez nie serwowane. Po uruchomieniu emulatora EBSimUnoEth do interakcji z emulowanym urządzeniem należy używać powyższą przeglądarkę. Dla wszystkich wykonanych operacji na zasobach CoAP złap pakiety sieciowe za pomocą programu Wireshark – zgodnie z dodatkiem wireshark_i_lapanie_pakietow.pdf (umieszczonym na serwerze studia3[2]). Pakiety te zapisz na dysku i załącz w sposób jednoznaczny do sprawozdania, pamiętaj o umiejętnym filtrowaniu pakietów tak aby łapane były tylko te związane z konwersacją z użyciem protokołu CoAP.

Zadanie 2. Sterowanie za pomocą protokołu UDP
Używając Arduino IDE, ObirEthernetUDP oraz emulator EBSimUnoEth napisz prosty serwer UDP nasłuchujący na porcie ***2392*.** Jego zadaniem jest miałoby być odbieranie od klienta datagarmów UDP i odesłanie do tego klienta treści odebranych datagramu ale zapisanych „wspak”. Jako klienta tego serwera użyj program linii poleceń o nazwie Netcat[3], a testowanie wykonaj poprzez wysyłanie wymyślone przez siebie treści. Podobnie jak w zadaniu 1 „złap” zestaw umiejętnie wyfiltrowanych pakietów sieciowych za pomocą programu Wireshark.

Zadanie 3. Rozszerzenie serwera UDP o mechanizmy badania stanu urządzenia
Dokonaj modyfikacji zadania 2 tak aby serwer UDP przyjmował od klienta polecenia: ‘*NIECHN*’, ‘*NIECHO*’. Załóż że na końcu każdego z tych poleceń może pojawić się liczba w formacie ‘*HEX*’ o długości tej liczby nie większej niż 8 znaków. Dodatkowo serwer ma być gotowy do otrzymania datagramu z poleceniem określającym jaką operacje ma wykonać tutaj: ‘*’. Po wykonaniu tej operacji niech wynik będzie zachowany w pamięci serwera, by po otrzymaniu polecenia ‘*PODAJ*’ był on odesłany w formacie ‘*HEX*’ do nadawcy od którego otrzymano to polecenie – można założyć, że tylko jeden klient będzie współpracował z serwerem. Proszę pamiętać, iż serwer ma oczekiwać w osobnych pakietach UDP na każde z poleceń.

Dodatkowe informacje: Przygotowanie interakcji narzędzia netcat z serwerem UDP
Używając narzędzia `netcat` (w wywołaniu używamy skróconej postaci: `nc`), w zadaniach: 2 i 3 dla każdego wysłanego datagramu podaj w raporcie także ich payload – tak aby możliwe było porównanie użytego payload’u z treścią złapanych pakietów. W raporcie zamieść także zestawy argumentów wywołań narzędzia `netcat` których użyłeś podczas testowania swojego kodu. Jeżeli to możliwe stosuj wywołania narzędzia netcat typu wsadowego – czyli bez interakcji z użytkownikiem (np.: `data | nc –u 10.17.0.238 1234`). Narzędzie `netcat` (w wersji dla systemu Windows) jest do pobrania z lokalizacji[2].
Po zakończeniu prac nie zapomnij o: a)wysłaniu listu do prowadzącego zajęcia z tematem: „Obir - laboratorium 3 zdalne” i w treści listu informacji o zakończeniu przez siebie ćwiczenia, b)w treści proszę zamieścić koniecznie: swoje imię i nazwisko, natomiast swoje: sprawozdanie, kody źródłowe i załączniki proszę umieścić na repozytorium GIT przygotowanym przez prowadzących przedmiot, pliki z kodem nazwij odpowiednio: zadanie2.ino, zadanie3.ino.

Linki
1. http://cygnus.tele.pw.edu.pl/olek/rozne/mmm/FirefoxPortableESR_52_6_0_CoAP.zip
2. https://studia3.elka.pw.edu.pl/f-pl/20L/103A-TLSST-ISP-OBIR/priv//Laboratoria/wireshark%5Fi%5Flapanie%5Fpakietow.pdf
3. http://cygnus.tele.pw.edu.pl/olek/rozne/iii/nc.exe