

# Sieci Komputerowe II

## Sprawozdanie projektu nr 16:

System nadzoru (detekcji awarii) serwerów sieciowych z wykorzystaniem protokołów ICMP oraz TCP

link do repozytorium: [https://gitlab.cs.put.poznan.pl/mikolaj.stankowiak/sieci\\_2017](https://gitlab.cs.put.poznan.pl/mikolaj.stankowiak/sieci_2017)

wykonali:

Mikołaj Stankowiak, 127234

Michał Wachowiak, 127300

grupa: wtorek 9:45

### 1. Założenia początkowe, opis działania

Serwer to aplikacja konsolowa napisana na platformę Linux w języku C. Klient to okienkowy program w środowisku Windows napisany w C#. Ideą projektu jest nawiązanie komunikacji klienta z serwerem, serwer w odpowiedzi ma wysyłać opóźnienie pakietu TCP do zapisanych we własnych strukturach danych serwerów sieciowych, podanych wcześniej przez klienta.

Komunikacja pomiędzy klientem a serwerem polega na spakowaniu potrzebnych danych do jednego ciągu bajtów. Zawartość spakowanego ciągu opisana jest w pliku readme.txt, w repozytorium projektu. Każdy ciąg bajtów oznaczony jest jednobajtową instrukcją sterującą, ciągiem argumentów o zmiennej długości oraz niepowtarzalną sekwencją bajtów 0xFFFFFFFF0. Funkcje odbierające i odczytujące dane realizują tę funkcję czytając bufor po jednym bajcie analizując zawartość bajtu w poszukiwaniu sekwencji końcowej.

Serwer przechowuje dane każdego użytkownika w osobnym pliku tekstowym, którego nazwa to nazwa użytkownika. Każdy serwer sieciowy przechowywany jest w plikach, jako jedna linia tekstowa zawierająca nazwę domenową i pięciocyfrowy nr portu, do którego łączy się serwer. Każdy klient może przesłać serwerowi maksymalny ping, po którym serwer sieciowy jest oznaczony, jako niedostępny. W przypadku braku pliku użytkownika, serwer tworzy pusty plik tekstowy.

### 2. Pliki nagłówkowe serwera:

- a. **all\_includes.h** - zawiera biblioteki wykorzystywane w projekcie, każdy inny plik nagłówkowy korzysta z tej biblioteki;

- b. **data\_out.h** - operacje na danych przesyłanych przez TCP jako jeden ciąg bajtów, pakowanie i rozpakowanie danych;
  - c. **io.h** - obsługa reprezentacji danych klientów jako pliki tekstowe, odpytywanie serwerów sieciowych;
  - d. **server\_table.h** - operacje na tablicy współdzielonej pamięci gromadzącej dane użytkowników;
  - e. **tests.h** - testy podstawowych struktur przeprowadzane w czasie braku klienta;
  - f. **thread\_functions.h** - funkcje współbieżnych wątków składających się na serwer;
  - g. **main.c** - funkcja główna, zawierająca początkowy kod serwera w tym oczekiwanie na nowe dane.
3. Pliki nagłówkowe klienta:
- a. **Logowanie.cs** – okno pierwszego połączenia z serwerem, test połączenia z kontrolą informacji zwrotnej;
  - b. **OknoGlowne.cs** – główne okno aplikacji, właściwe połączenie z serwerem, wyświetlanie danych;
  - c. **Dodawanie.cs** – pobieranie od użytkownika nowej nazwy domenowej z portem;
  - d. **data.cs** - operacje na danych przesyłanych przez TCP jako jeden ciąg bajtów, pakowanie i rozpakowanie danych.
4. Opcje kompilacji:
5. Obsługa: