

SYSTEMY RÓWNOLEGŁE I ROZPROSZONE

2015-2016

semestr letni

Ćwiczenia laboratoryjne

Gniazda w modelu p2p

1. Wprowadzenie.

Model peer-to-peer jest - obok klasycznego modelu klient-serwer – jedną z najpopularniejszych architektur rozproszonych. W modelu p2p każdy węzeł sieci pełni jednocześnie rolę serwera i klienta. Pozwala to na zdecentralizowanie sieci i bardzo dobrą skalowalność.

2. Zadania.

W ramach zadania należy (w zwartej formie) napisać samodzielnie program w architekturze peer-to-peer, który samodzielnie zlokalizuje inne programy w lokalnej podsieci i pobierze z nich wymagane dane, jednocześnie udostępniając innym programom możliwość pobrania lokalnie przechowywanej porcji danych. W czasie planowania programu, można (a nawet należy) wesprzeć się odpowiednimi źródłami. Należy ustalić jakie dane powinien precyzować odpowiedni „protokół” służący efektywnej komunikacji między różnymi programami. Jednoczesność szukania i udostępniania danych może wymagać wielu wątków i wielu gniazd.

3. Proces.

Najpierw grupa jako całość musi ustalić protokół, którym będą wymieniać dane. Powinien on zawierać wszystkie informacje pozwalające na efektywną komunikację!

Realizacja zadań będzie się odbywała indywidualnie, w celu testowania programu można oczywiście współpracować ze sobą.

Efektom pracy powinno być program o poniższych cechach:

- działa pod kontrolą systemu z rodziny linux
- w języku C lub C++
- z wykorzystaniem gniazd w standardzie POSIX
- z wykorzystaniem wątków
- z wykorzystaniem więcej niż 1 gniazda

Każda osoba otrzyma arbitralnie przydzielony **numer**, oznaczający który **werset** ze wskazanego wiersza K.K. Baczyńskiego ma udostępniać jej program.

Zasadniczym zadaniem programu jest:

- odkrycie w lokalnej podsieci wszystkich programów udostępniających dane w zdefiniowanym „protokole”
- pobranie od nich danych
- ułożenie danych (wersów) we właściwej kolejności (wg numerów)
- *jednoczesne* udostępniania swoich danych lokalnych innym

Uwaga: w celu przejrzystego działania programu, musi on jawnie informować o:

- odkrytych węzłach w podsieci
- wysłanych informacjach do innych
- otrzymanych informacjach
- aktualnego stanu zgromadzonych wersetów i ich kolejności

4. Ewaluacja.

Na ocenę waszej pracy, będą miały wpływ następujące czynniki:

- a. poprawność implementacji
- b. dobór źródeł

Punktacja	0 pkt	1 pkt	2 pkt
(a) poprawność implementacji	Implementacja niekompletna lub niedziałająca lub w złej technologii.	Implementacja działa poprawnie we właściwej technologii.	Implementacja z pełną obsługą błędów.
(b) dobór źródeł	Cytowania z niemożliwą do sprawdzenia wiarygodnością (wikipedia, fora internetowe etc.). Cytowania niezwiązane z treścią.	Cytowania wiarygodnych źródeł ze sprawdzonym autorem (nazwisko/firma).	Cytowania posiadają datę dostępu. Cytowania źródeł oryginalnych. Cytowania źródeł obcojęzycznych

5. Źródła.

- a. [http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994(E).zip) (dostęp 03.05.2016)
- b. <http://man7.org/linux/man-pages/man7/socket.7.html> (dostęp 03.05.2016)
- c. <http://tldp.org/LDP/LG/issue74/tougher.html> (dostęp 03.05.2016)
- d. <https://msdn.microsoft.com/en-us/library/windows/desktop/ms738545.aspx> (dostęp 03.05.2016)
- e. <https://msdn.microsoft.com/en-us/library/windows/desktop/ms741416.aspx> (dostęp 03.05.2016)

6. Konkluzja.

Powyższe zadanie pozwoli Wam stworzyć program, który funkcjonalnie będzie realizował koncepcję p2p, przy jednoczesnym dość prostym zadaniu właściwym.