

Grupa ćwic. IIIb	Grupa lab. 7	Rok 3 IS	Data wykonania. 22.10.09	Data odbioru
Nr ćwic./wersja 1	Temat ćwiczenia. Voice Conference			
Imiona i nazwiska. Grzegorz Gliński				Ocena i uwagi

Część praktyczna

1. Opis wykonanego ćwiczenia

* Użyte programy:

- Program implementujący protokół H.323: **SJPhone**
- Program implementujący protokół SIP: **ExpressTalk**
- Monitorowanie systemu: **CoolMon**

2. Wstęp teoretyczny

* **H.323** jest w dzisiejszych czasach standardem w multimedialnej komunikacji (głos, obraz, wiadomości tekstowe). Protokół ten przesyła dane w czasie rzeczywistym, dzięki czemu możemy kontrolować opóźnienia. Standard stworzony do wideokonferencji ma także pewne wady. Dane przesyłane są pakietowo, korzystając zarówno z TCP zapewniającego dostarczenie każdego pakietu, jak i UDP, który tego nie gwarantuje. Głównym konkurentem tego protokołu jest standard SIP (Session Initiation Protocol), mający rozszerzoną funkcjonalność względem opisywanego protokołu. Opracowany i zaakceptowany przez ITU w 1996 r. protokół H.323 zrewolucjonizował sposób prowadzenia telekonferencji i przesyłania głosu przez sieci IP. H.323 nadzoruje proces przesyłania danych multimedialnych w sieciach pakietowych, wykonując to zadanie w czasie rzeczywistym. Standardy wchodzące w skład protokołu H.323 definiują dokładnie, jak poszczególne elementy systemu pracujące zgodnie z tym protokołem inicjują sesje multimedialne i jak stanowiska pracy wymieniają między sobą skompresowane dane audio i wideo. Już na początku lat 90. poszczególne firmy zainteresowały się przesyłaniem głosu przez sieci IP. Po jakimś czasie na rynku pojawiły się pierwsze aplikacje realizujące takie zadanie. I tu powstał problem. Wszystkie aplikacje pracowały najczęściej w oparciu o firmowe rozwiązania, tak że użytkownicy nie mogli stosować w jednym środowisku urządzeń produkowanych przez różne firmy, gdyż nie były one ze sobą zgodne. Od razu zrodził się pomysł, aby opracować jeden standard, który obowiązywałby wszystkie firmy. I tak właśnie powstał protokół H.323. Trzeba tu od razu powiedzieć, że pierwsza wersja standardu H.323 koncentrowała się na przesyłaniu głosu przez sieci LAN. Po jakimś czasie projektanci tego rodzaju aplikacji postanowili wprowadzić tę technologię do Internetu. Okazało się wtedy, że protokół H.323 V.1 nie wystarcza. Dlatego zaczęto od razu myśleć o kolejnej wersji tego protokołu, spełniającej oczekiwania użytkowników pracujących w Internecie. Prace nad protokołem H.323 V.2 zakończono pod koniec 1997 r. i w styczniu 1998 protokół H.323 V.2 został oficjalnie zaakceptowany przez ITU.

* **Składniki protokołu H.323:**

Protokół H.323 definiuje cztery podstawowe składniki systemu przesyłającego w czasie rzeczywistym dane multimedialne:

- terminale,
- bramy,
- gatekeepers,
- jednostki kontrolne MCU (Multipoint Control Units).

Terminale pozwalają przesyłać dane multimedialne w czasie rzeczywistym. Terminale muszą zawsze obsługiwać dane audio (głos), a opcjonalne moduły terminalu mogą też przesyłać dodatkowo obrazy i dane.

W sieciach IP do przesyłania pakietów zawierających głos jest używany protokół UDP (User Datagram Protocol). Protokół ten nie przewiduje stosowania znaczników czasu, dlatego nie może kontrolować opóźnień. Aby uniknąć kłopotów wynikających z faktu, że protokoły IP/UDP nie mogą kontrolować opóźnień, projektanci aplikacji VoIP zaproponowali standard H.323.

Bramy H.323 świadczą swe usługi klientom H.323, tak aby mogły się komunikować z rozwiązaniami opartymi na innym protokole niż H.323. Brama to element sprzęgający sieć IP z siecią telefoniczną. Typowa brama H.323 zapewnia na przykład komunikację między terminalami H.323 i telefonami dołączonymi do standardowej sieci telefonicznej (opartej na komutacji obwodów). Brama musi więc dysponować mechanizmami konwertującymi różne formaty i obsługiwać sieci oparte na różnych technologiach.

Gatekeeper to element świadczący swe usługi punktom końcowym H.323 (takie jak translacja adresów i zarządzanie przepustowością łącza), kontrolujący proces inicjowania połączeń. W sieciach H.323 są to elementy opcjonalne. Jeśli w sieci istnieją takie opcjonalne elementy, punkty końcowe muszą używać swoich własnych usług, tak aby mogły się z nimi komunikować. Standardy H.323 definiują dokładnie, jakiego rodzaju usługi musi świadczyć gatekeeper, i określają dodatkowe funkcje, które może realizować taki element. W standardzie określony jest również protokół komunikacji GK - EP (end point) i jest nim H.225 (RAS)

Jednostki MCU obsługują konferencje, w których udział biorą co najmniej trzy (lub więcej) punkty końcowe. Jednostka MCU zarządza zasobami konferencji, prowadzi negocjacje między punktami końcowymi (uzgadniając na przykład metodę kodowania danych audio i wideo) i może (ale nie musi) sterować strumieniami pakietów zawierających dane multimedialne. MCU zawiera dwa rodzaje modułów: Wymagany Multipoint Controller - który np. obsługuje sygnalizację H.255.0 oraz odpowiada za negocjację parametrów komunikacji między punktami końcowymi (end points). MCU (Mostek konferencyjny) może również zawierać moduł Multipoint Processor - odpowiedzialny za np. Miksowanie różnych danych multimedialnych, translacje formatów, ewentualną redystrybucję strumieni do użytkowników.

* **SIP (ang. Session Initiation Protocol)**, protokół inicjowania sesji, to zaproponowany przez IETF standard dla zestawiania sesji pomiędzy jednym lub wieloma klientami. Jest obecnie dominującym protokołem sygnalizacyjnym dla Voice over IP i stopniowo zastępuje H.323.

SIP ma w zamierzeniu dostarczać zestaw funkcji obsługi połączenia i innych cech obecnych w publicznej sieci telefonicznej (PSTN). Jako taki, zawiera funkcje które umożliwiają znane ze stacjonarnej telefonii operacje: wybieranie numeru, dzwonek w telefonie, sygnał zajętości itp. Jednakże ich implementacja i używana terminologia jest odmienna.

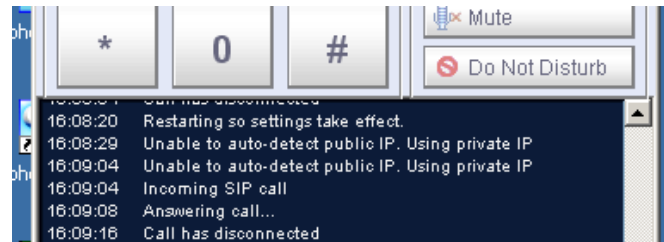
SIP implementuje również wiele bardziej zaawansowanych możliwości obsługi połączenia dostępnych w protokole sygnalizacyjnym SS7, mimo to jednak te dwa protokoły bardzo różnią się od siebie. SS7 to wysoce scentralizowany protokół charakteryzujący się złożoną architekturą sieciową i 'nieinteligentnymi' węzłami końcowymi (tradycyjne terminale telefoniczne). SIP to protokół typu peer-to-peer. Wymaga jedynie prostej (a przez to wysoce skalowalnej) sieci szkieletowej. Inteligencja rozproszona jest na brzegach sieci - zaszyta w węzłach końcowych (końcowe urządzenia zaimplementowane sprzętowo lub jako oprogramowanie). Stoi to więc w przeciwieństwie do funkcji oferowanych przez SS7, których dostarcza sama sieć. Istnieje wiele innych protokołów sygnalizacyjnych dla VoIP, jednakże SIP zdefiniowano pośród społeczności internetowej a nie telekomunikacyjnej. SIP jest standardem zarządzanym przez IETF. Starsze i bardziej złożone protokoły VoIP były zazwyczaj propozycjami zgłaszanymi przez ITU-T.

3. Wykonanie ćwiczenia

Ćwiczenie było podzielone na dwa etapy: łączenie się pomiędzy dwoma komputerami, oraz konferencja czterech komputerów (jednakże z racji tego że podczas wykonywania ćwiczenia jedna osoba była nieobecna, próba została przeprowadzona na trzech a nie czterech komputerach). Podczas każdej próby monitorowane było obciążenie procesora, pamięci oraz ilość wysyłanych i odbieranych danych poprzez sieć. Każda próba składała się z 2 etapów: rozmowy (podczas gdy użytkownicy rozmawiali) oraz ciszy (gdy przy włączonym monitorze nie była prowadzona rozmowa). Do monitorowania pracy systemu został użyty program CoolMon, którego praca została zarejestrowana na zrzutach ekranu. Niestety wystąpił problem techniczny z programem ExpressTalk, zaraz po nawiązaniu połączenia było ono zrywane przez jedną ze stron (o czym druga strona nie była powiadamiana).

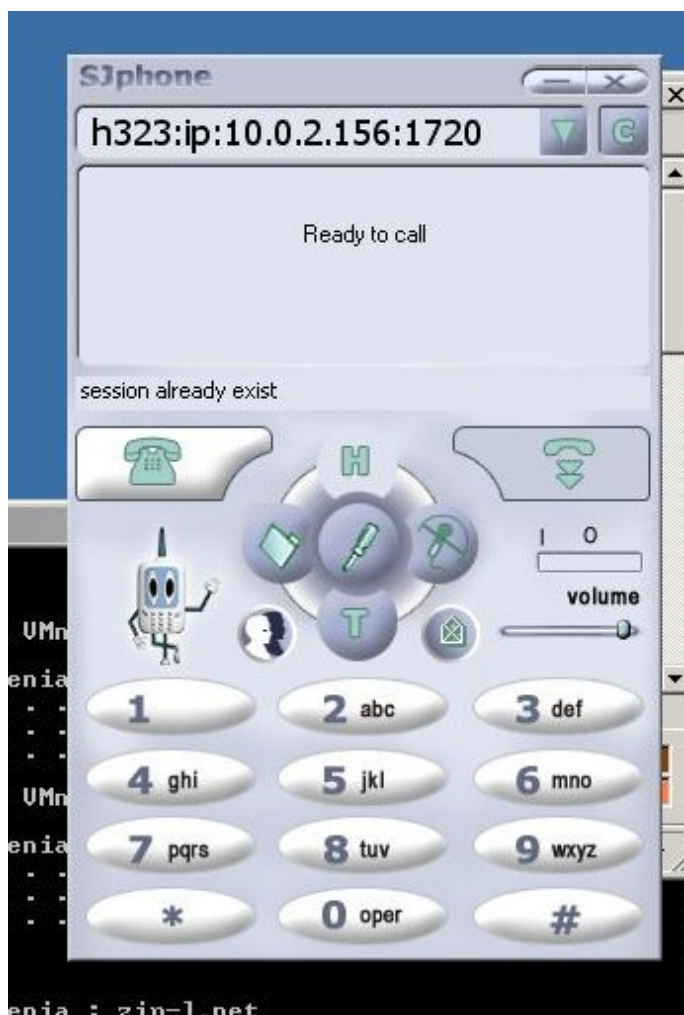


Express Talk, strona nawiązująca połączenie



ExpressTalk, strona odbierająca, jedna z wielu prób

Kolejnym etapem było przeprowadzenie prób na aplikacji wykorzystującej protokół H.323. Naszym wyborem był program SJPhone, który implementuje zarówno SIP jak i H.323.



Program Sjphone gotowy do pracy

Po nawiązaniu połączenia prowadzone były próby Na dwóch i 3 komputerach, podczas ciszy i podczas rozmowy.

```
KB Write: 10,45
KB Read Sum: 73 845,77
KB Write Sum: 11 069,06
Proc %: 00,00
Processes: 38  Threads: 466
0 days, 00 hrs, 58 mins, 07 secs
PageUsed: 266/PageTotal: 4066
RAMUsed: 325/RAMTotal: 511
Date 22-10-09 Time 16:01
10.0.2.174, 192.168.222.1, 192.168.255.1
STUDENT54
KB Read: 10,45
```

Program SJphone podczas ciszy (2 komputery)

```
KB Write: 10,56
KB Read Sum: 68 201,69
KB Write Sum: 4 886,43
Proc %: 00,00
Processes: 39  Threads: 468
0 days, 00 hrs, 37 mins, 28 secs
PageUsed: 267/PageTotal: 4066
RAMUsed: 327/RAMTotal: 511
Date 22-10-09 Time 15:40
10.0.2.174, 192.168.222.1, 192.168.255.1
STUDENT54
KB Read: 10,95
```

Program SJphone podczas rozmowy (2 komputery)

Jak możemy zaobserwować po wykonanych zrzutach podczas rozmowy, nieznacznie zmienia się wykorzystanie sieci a użycie procesora czy pamięci praktycznie się nie zmienia. Możemy stwierdzić że różnica jest niezauważalna dla zwykłego użytkownika.

Uzyskane wyniki zostały zebrane w tabeli:

	Wychodzące KiB	Przychodzące KiB	Użycie CPU	Użycie pamięci
Cisza (2 komputery)	10,45KiB/s	10,45KiB/s	0,00%	325MiB
Rozmowa (2 komputery)	10,56KiB/s	10,95KiB/s	0,00%	327MiB
Cisza (3 komputery)	13,34KiB	13,35KiB/s	0,01%	323MiB
Rozmowa (3 komputery)	14,01KiB/s	13,95KiB/s	0,00%	324MiB

Niestety posiadana przez nas wersja aplikacji nie pozwalała na zmianę ustawień kodeków dla protokołu H.323. Podczas prób zaznaczona była opcja: „Do not send silence”, więc teoretycznie powinniśmy uzyskać spadek użycia łącza podczas ciszy.

4. Wnioski

Jak możemy zaobserwować (patrząc na wyniki zebrane w tabeli), dla użycia procesora i pamięci praktycznie się nie zmieniają, jedyny skok użycia procesora był obserwowany podczas inicjowania połączenia (do 35%). Brak zmiany użycia połączenia sieciowego jest spowodowany najprawdopodobniej tym, że podczas prób w pracowni przebywało wiele osób i mikrofon zbierał szum z pracowni. W normalnych warunkach program powinien podczas ciszy wysyłać znacznie mniej danych. Trzeba dodać że pomiar cechuje się dość dużym błędem. Jest to spowodowane tym że użycie zasobów systemowych oraz sieci, mierzymy dla całego systemu a nie tylko dla programu SJPhone. Sam program jest dość przejrzysty, dość łatwo się go obsługuje, transmitowany dźwięk jest w dość dobrej jakości, jednak posiadana przez nas wersja (jak się później okazało nie była to najnowsza wersja aplikacji), miała dość małe możliwości konfiguracyjne jeśli chodzi o zmianę kodeków oraz innych parametrów transmisji danych. Nie byliśmy w stanie ustalić dlaczego program ExpressTalk zrywa połączenie. Podczas jednej próby udało się uzyskać połączenie (słyszeliśmy się nawzajem) ale trwało to tylko kilka sekund i nie zdążyliśmy wykonać potrzebnych pomiarów. Sądzymy że mogło to być spowodowane konfiguracją systemu Microsoft Windows (ustawienia zapory). Jak wiadomo system Windows XP jest dość nieprzewidywalny jeśli chodzi o zaporę (wyświetla informację że aplikacja została zablokowana a tak naprawdę połączenie jest nawiązane).