

ANALIZA WYBRANYCH ELEMENTÓW SYSTEMU KOMUNIKACJI GŁOSOWEJ WYKORZYSTUJĄCYCH PROTOKÓŁ SIP

Streszczenie

Niniejszy artykuł jest poświęcony analizie systemu komunikacji głosowej opartemu o protokół SIP. Obecnie jest to najpopularniejsza metoda komunikacji głosowej. W artykule zaprezentowano podstawowe komponenty systemu wykorzystującego protokół SIP. Wskazano na wybrane, publicznie dostępne rozwiązania komponentów systemu SIP – terminala SIP Linphone i serwera SIP Kamailio. Rozwiązania te zostały kompleksowo przetestowane pod względem funkcjonalnym.

Przedstawione rozwiązania mogą być wykorzystywane w systemach realizujących usługi komercyjne. Ze względu na pełną dostępność kodu źródłowego Linphone i Kamailio przedstawione komponenty SIP mogą być wykorzystane do dydaktyki i do prowadzenia badań naukowych.

WSTĘP

Komunikacja głosowa jest jedną z pierwszych usług telekomunikacyjnych wprowadzonych na dużą skalę. Z upływem lat usługa ta ewoluowała, wykorzystując różne rozwiązania technologiczne. Przez wiele dziesięcioleci w telefonii wykorzystywane były wyłącznie techniki komutacji kanałów. Wraz z rozwojem techniki cyfrowej i zwiększeniem przepustowości sieci komputerowych, usługa ta zaczęła być powszechnie realizowana w sieci Internet, tworząc podwaliny dla telefonii IP. Obecnie usługa transmisji głosowej jest realizowana w dużej mierze z wykorzystaniem transmisji w sieci IP. I to nie tylko do transmisji realizowanej za pośrednictwem różnego typu „komunikatorów internetowych”, ale również do transmisji głosu w sieciach operatorów telekomunikacyjnych (zarówno sieci stacjonarnych, jak i nowych sieci komórkowych w standardzie UMTS).

Komunikacja głosowa w sieci IP była pierwotnie realizowana zgodnie ze standardem H.323 [3] zdefiniowanym przez ITU-T. Standard H.323 wykorzystywał te same metody kodowania i kompresji oraz te same protokoły, co wcześniejsze systemy konferencyjne definiowane przez ITU-T. Pozwalało to na łatwą integrację H.323 z istniejącymi rozwiązaniami.

Konkurencję dla rozwiązań ITU stanowiły prace prowadzone równolegle przez IETF w ramach grupy MMUSIC (*Multiparty Multimedia Session Control*). Jednym z najważniejszych rozwiązań grupy MMUSIC stał się protokół SIP (*Session Initiation Protocol*) [1][5]. Protokół ten charakteryzował się prostotą i łatwością implementacji. Dlatego już w 2000 roku protokół SIP stał się podstawowym elementem architektury projektu (ówczesnego) sieci 3GPP (*3rd Generation Partnership Project* – projekt rozwijający telefonię komórkową trzeciej generacji) tworząc podsystem IMS (*IP Multimedia Subsystem*) służący do strumieniowania multimediów w sieci IP.

W artykule zaprezentowano podstawowe komponenty systemu SIP wykorzystywane podczas komunikacji głosowej. Komponenty te zostały kompleksowo przetestowane. Niniejszy artykuł składa się z trzech rozdziałów. Rozdział pierwszy poświęcony jest budowie systemów komunikacji wykorzystujących do sygnalizacji protokół SIP. W rozdziale drugim zaprezentowano przykładowe rozwiązania programowe służące do budowy podstawowych komponentów systemu SIP. W rozdziale trzecim przedstawiono analizę funkcjonalną zaprezentowanych komponentów.

1. BUDOWA SYSTEMU WYKORZYSTUJĄCEGO PROTOKÓŁ SIP

Protokół SIP został zdefiniowany w RFC 3261 [5]. Podstawowe operacje w protokole są realizowane w oparciu o model klient-serwer. Komunikaty przesyłane są w formacie tekstowym o składni zbliżonej do stosowanej w protokole HTTP i zapisywane są zgodnie z formatem „*Internet Message Format*” [4].

W protokole SIP wykorzystywane są dwa typy adresów: adres użytkownika AOR (*Address of Record*) i adres identyfikujący urządzenie i użytkownika FQDN (*Fully Qualified Domain Name*).

W adresie AOR podawany jest identyfikator użytkownika zarejestrowanego w danej domenie. Przykładowo, użytkownik „Janek” jest zarejestrowany w domenie „firma.example.com”. Jego adres będzie posiadał zatem format „janek@firma.example.com”.

W przypadku adresu FQDN podawany jest pełny adres urządzenia i nazwa użytkownika, np. ktos4@203.0.113.34 gdzie „203.0.113.34” jest adresem IP danego terminala a „ktos4” jest nazwą użytkownika na tym terminalu. Adres FQDN jest używany najczęściej gdy używamy najprostszej konfiguracji systemu SIP posiadającej tylko terminale.

W systemie wykorzystującym protokół SIP, adresacja jest realizowana w oparciu o URI (*Uniform Resource Indicators*). URI jest tworzone na bazie adresu AOR lub FQDN poprzez poprzedzenie go identyfikatorem protokołu „sip:” (dla protokołu SIP bez szyfrowania) lub „sips:” (dla protokołu SIP z szyfrowaniem – *Secure SIP*). Przykładowo dla adresu AOR „janek@firma.example.com” i protokołu SIP bez szyfrowania URI będzie posiadał postać: „sip:janek@firma.example.com”.

System SIP składa się z kilku komponentów (Rys. 1): agenta użytkownika (ang. *User Agents* – UA), serwera proxy (ang. *proxy server*), serwera przekierowań (ang. *redirect server*) i serwera rejestracyjnego (ang. *registrar server*) [1]. Dodatkowo, w systemie może występować również serwer lokalizacji (ang. *location server*). Minimalny zestaw obejmuje dwa terminale SIP zawierające moduł UA. Bardziej rozbudowane konfiguracje zawierają także inne elementy systemu SIP. Komponenty SIP mogą być zlokalizowane na jednym fizycznie systemie komputerowym (np. serwer S1 na rys. 1 składa się z serwerów: SIP proxy, rejestracji i lokalizacji).

Terminalem SIP może być komputer PC lub smartfon z odpowiednią aplikacją realizującą funkcje modułu UA. Są także dostępne

dedykowane urządzenia realizujące funkcje terminala SIP (określone także jako telefony IP) wyposażone najczęściej w interfejs standardu Ethernet, za pomocą którego realizowane jest podłączenie do sieci IP. Urządzenia te również realizują funkcje modułu UA. Moduł UA składa się z dwóch elementów: realizującego funkcje klienta (ang. *user agent client, UAC*) i realizującego funkcje serwera (ang. *user agent server, UAS*).

Uniwersalnymi urządzeniami są serwery SIP proxy, które pośredniczą w realizacji połączeń w bardziej złożonych systemach, gdzie system końcowy jest ulokowany poza daną podsiecią (domeną). Taka sytuacja została przedstawiona na rysunku 1, gdzie terminale ulokowane są w dwóch podsieciach (domena A i domena B). Serwer SIP proxy przesyła komunikaty od terminala A do sieci docelowej terminala B bezpośrednio lub korzystając z innych systemów pośredniczących.

Serwer lokalizacji jest systemem, który pozwala na skojarzenie adresu wywołanego użytkownika z jego aktualną lokalizacją. Jeżeli stosowane są terminale stacjonarne to wystarczy w tym wypadku zastosowanie serwisu DNS. Serwis DNS na podstawie podanego adresu symbolicznego użytkownika ustali jego adres sieciowy.

W przypadku gdy użytkownik zmienia lokalizację i pracuje poza domeną, w której jest zarejestrowany, wymagany jest serwer rejestracyjny. Pozwala on na ustalenie aktualnej lokalizacji danego użytkownika.

2. ANALIZOWANE KOMPONENTY SYSTEMU SIP

Rolę terminala SIP może pełnić system komputerowy (komputer klasy PC, smartfon) z odpowiednim wyposażeniem i oprogramowaniem lub specjalizowane urządzenie. W artykule przeanalizowano pracę terminala SIP zbudowanego w oparciu o komputer PC i oprogramowanie Linphone. Linphone jest oprogramowaniem dostępnym na licencji GNU GPL (*General Public License*)¹. Oprogramowanie jest dostępne dla systemów operacyjnych: iOS, Android, Linux, MAC OSX, Windows Phone 8 oraz systemu Windows w wersji dla komputerów PC (oprogramowanie dla systemu Windows dla komputerów PC można pobrać ze strony:

<http://www.linphone.org/downloads-for-desktop.html>).

Oprogramowanie Linphone umożliwia budowę zarówno głosowego terminala SIP, jak i wideo terminala SIP. Obsługuje ono najważniejsze standardy kodowania i kompresji dźwięku (m.in. G722, GSM 6.10, G729, G711). W przypadku transmisji wideo obsługiwane są standardy wykorzystywane przez systemy konferencyjne: VP8, H264, MPEG4, H263. Oprogramowanie Linphone wspiera QoS oraz umożliwia kasowanie echa dla połączeń głosowych. Ważne jest też pełne wsparcie dla protokołów IPv4 i IPv6.

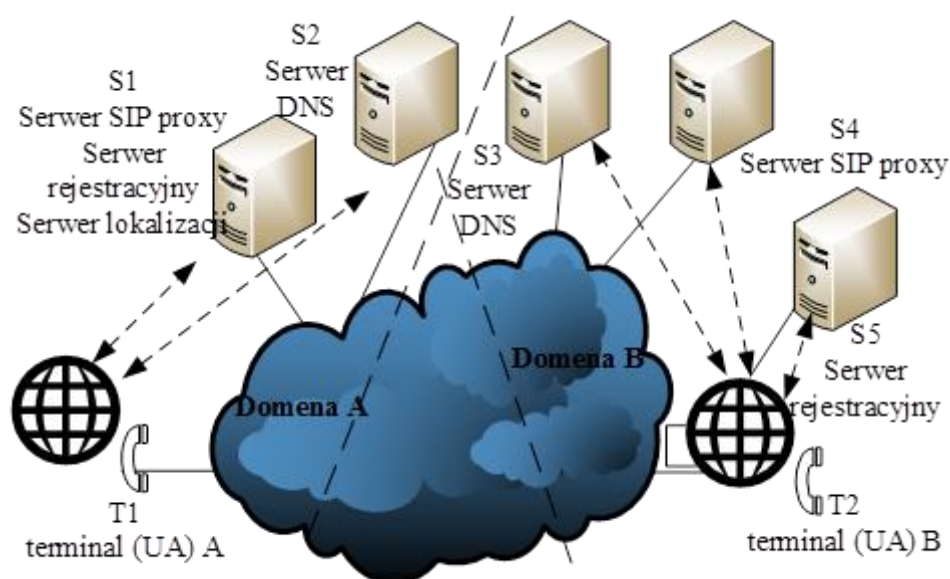
Praca z programem możliwa jest za pomocą interfejsu graficznego jak i za pomocą wiersza poleceń. Użycie wiersza poleceń możliwe jest poprzez polecenie `linphonec` (wersja terminalowa oprogramowania). Oprócz dwóch podstawowych trybów pracy z oprogramowaniem Linphone możliwa jest też praca oprogramowania w trybie procesu działającego w tle (demon), który jest sterowany poprzez narzędzie `linphonecsh`. Wersja ta jest przeznaczona do uruchamiania połączeń za pomocą złożonych skryptów sterujących lub przy współpracy z przeglądarką (polecenia są przekazywane poprzez odpowiednie wywołania javascript).

Oprogramowanie Linphone po instalacji tworzy domyślnego użytkownika *toto* nadając mu adres FQDN w oparciu o aktualną konfigurację sieciową interfejsów komputera np. `sip:toto@192.168.2.5` (gdzie „192.168.2.5” jest adresem IP komputera, na którym zainstalowany został program Linphone).

Terminal SIP, pracujący samodzielnie, bez innych elementów architektury SIP, umożliwia nawiązywanie połączenia z bezpośrednio wskazanymi innymi terminalami (dokładniej ze wskazanym użytkownikiem na danym terminalu). Używane muszą być w tym celu adresy FQDN.

W Linphone zaimplementowano protokół ICE (*Interactive Connectivity Establishment*) [6], który umożliwia połączenie dwóch terminali SIP, znajdujących się w różnych podsieciach za modułem NAT, bez elementów pośredniczących architektury SIP.

Użycie dodatkowych komponentów SIP pozwala na bardziej złożone tryby pracy protokołu SIP, w których można używać adresów AOR. W artykule przedstawiono analizę serwera SIP zbudowanego przy wykorzystaniu oprogramowania Kamailio (<http://www.kamailio.org/>) udostępnianego na licencji GNU GPL.



Rys. 1. Komponenty systemu SIP

¹ <http://www.linphone.org/>

Oprogramowanie Kamailio realizuje funkcje serwera SIP proxy, serwera rejestracyjnego, serwera lokalizacji i serwera przekierowań. Tym samym w połączeniu z terminalami SIP i ze zmodyfikowaną konfiguracją systemu DNS pozwala na utworzenie kompletnego systemu SIP. Realizuje ono także pełne wsparcie dla protokołów IPv4 i IPv6.

Kamailio jest systemem modułowym, do którego można dodawać własne, specjalizowane moduły. Posiada interfejsy API m.in. do języków: Perl, Java, Lua, Python. Współpracuje z typowymi systemami baz danych jak MySQL, PostgreSQL czy bazami zapisanymi w postaci pliku tekstowego.

Kamailio posiada pełną współpracę z systemem DNS (także DNSsec). Tym samym umożliwia realizację pełnej usługi lokalizacyjnej. Posiada również wbudowaną pamięć podręczną zapytań DNS aby uniknąć blokady systemu DNS w przypadku bardzo dużej ilości zapytań do danej domeny.

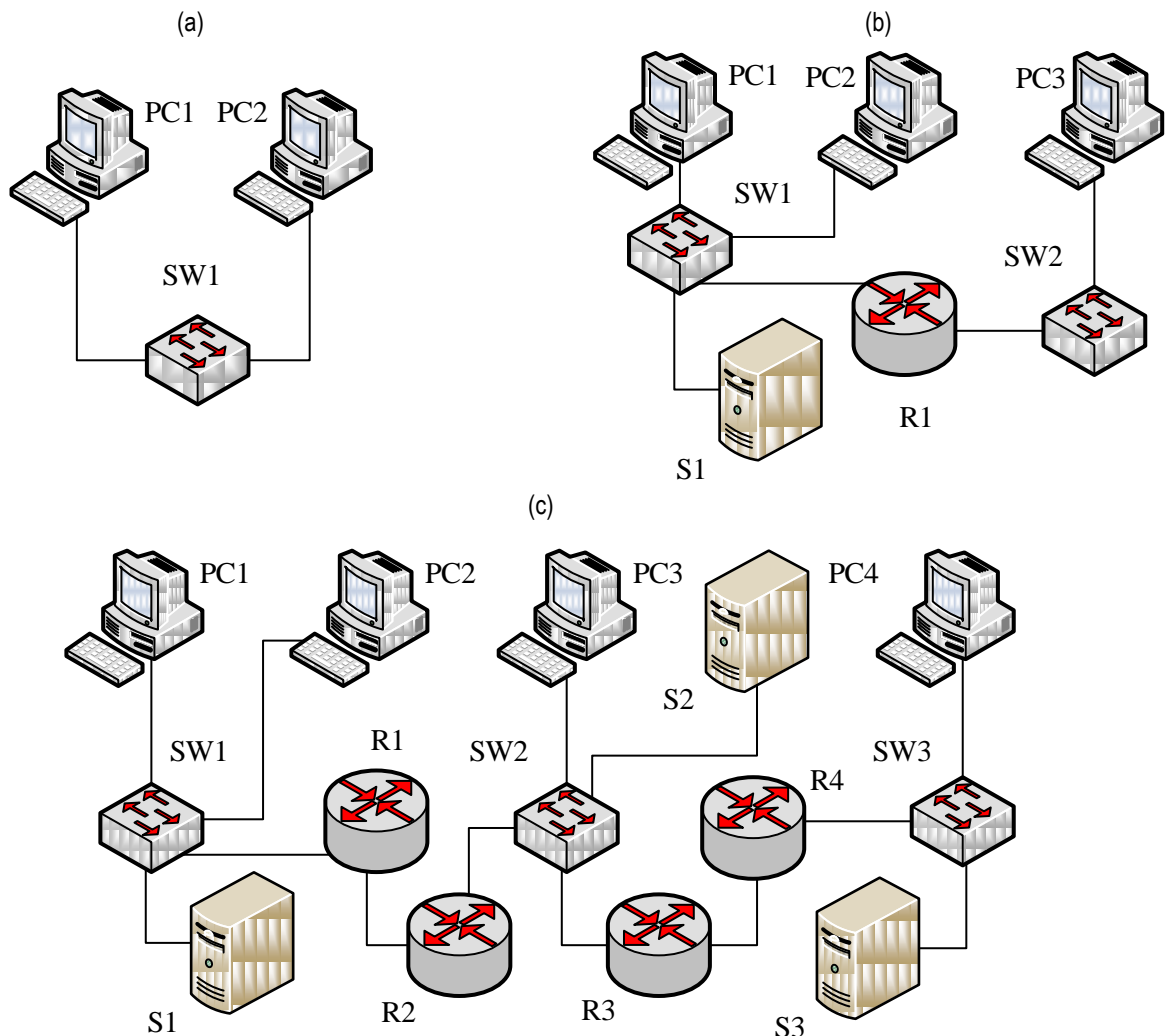
Konfiguracja Kamailio może być zmieniana podczas pracy systemu – nie jest konieczny restart oprogramowania po zmianie konfiguracji. Wspiera również równoważenie ruchu rozkładając ruch z wykorzystaniem złożonych algorytmów.

3. ANALIZA PRACY SYSTEMU SIP

Analiza systemu komunikacji głosowej SIP została przeprowadzona w sieci testowej utworzonej w laboratorium. Do celów testowych utworzono wiele topologii (rys. 2): od prostej (składającej się tylko z dwóch terminali SIP znajdujących się w jednej sieci rys. 2a), poprzez szereg topologii wzbogaconych o dodatkowe elementy (rys. 2b), do złożonej, obejmującej kilka podsieci i wiele elementów systemu SIP (rys. 2c).

Do budowy sieci testowej wykorzystano wydajne systemy komputerowe (PC1, PC2, PC3 i PC4) wyposażone w procesory wielordzeniowe, wspierające wirtualizację VT-x². Zastosowano przełączniki sieciowe (SW1, SW2 i SW3) Cisco serii Catalyst 2960 z portami w standardzie Gigabit Ethernet. Jako routery (R1, R2 i R3) zastosowano routery dostępowe Cisco serii 2800 z portami Gigabit Ethernet i łączami szeregowymi obsługiwanymi przez karty WIC-2T.

Komputery PC (PC1, PC2, PC3 i PC4) dołączono do przełączników sieciowych (SW1, SW2 i SW3) łączami w standardzie Gigabit Ethernet. Również routery (R1, R2 i R3) dołączono do przełączników sieciowych łączami Gigabit Ethernet. Pomiędzy routerami R1 i R2 oraz pomiędzy routerami R3 i R4 stosowane były łącza Gigabit Ethernet lub łącza szeregowy. Zmian typu łącza pomiędzy wskaza-



Rys. 2. Testowane topologie sieciowe: a) prosta topologia, b) topologia z jednym serwerem SIP proxy (serwer S1), c) topologia rozbudowana, zawierająca serwery SIP (serwery S1 i S3) i serwer DNS wspierający usługę SIP (serwer S2)

² VT-x – Intel® Virtualization Technology – rozszerzenia instrukcji procesorów IA-32 wspierające sprzętowo wirtualizację [7]

nymi ruterami miała na celu wprowadzenie do systemu tzw. wąskiego gardła, dzięki któremu można było analizować występujące w systemach przeciążenia.

W topologii najprostszej (rys. 2a) przydzielono następujące adresy komputerom: PC1 adres 192.168.1.2, PC2 192.168.1.3. Na komputerze PC1 zdefiniowano użytkownika *jakis2*, którego nazwa wyświetlana to „Lech Chmielewski”. Jego adres SIP zgodnie z adresacją FQDN to `sip:jakis2@192.168.1.2`. Na komputerze PC2 zdefiniowano użytkownika *cos3*, którego nazwa wyświetlana to „Jan Wedrowny”. Jego adres SIP zgodnie z adresacją FQDN to `sip:cos3@192.168.1.3`.

W prostej topologii przanalizowana została łączność bezpośrednią pomiędzy telefonami programowymi Linphone. Analizowana

była transmisja informacji sygnalizacyjnej SIP. Analiza przeprowadzona została za pomocą aplikacji Wireshark (rys. 3).

Rozmowę inicjuje użytkownik *jakis2* (rys. 3) z terminala na komputerze PC1 (192.168.1.2) (komunikat *INVITE SDP*). Terminal na komputerze PC2 (192.168.1.3) odpowiada komunikatami potwierdzającymi otrzymanie żądania nawiązania połączenia (komunikat *Trying*) i sygnalizacją, że użytkownik na wywoływanym terminalu jest powiadamiany o żądaniu połączenia (komunikat *Ringin*). Gdy użytkownik odpowiada na dzwonięcie odbierając połączenie, terminal SIP wysyła komunikat *Ok. SDP*. Komunikat ten jest potwierdzany przez terminal wywołujący komunikatem *ACK*. Zakończenie połączenia inicjuje użytkownik terminala na komputerze PC2 wysyłając komunikat *BYE*. Komunikat ten jest potwierdzany przez terminal na komputerze PC1.

Time	192.168.1.2	192.168.1.3	Comment
55.157854000		INVITE SDP (opu...)	SIP From: "Lech Chmielewski" <sip:jakis2@192.168.1.2> To:sip:cos3@192.168.1.3
55.203051000	100 Trying		SIP Status
55.317145000	180 Ringing		SIP Status
59.023116000	200 Ok SDP (opu...)		SIP Status
59.052805000	ACK		SIP Request
63.986876000	BYE		SIP Request
63.996358000	200 Ok		SIP Status

Rys. 3. Komunikaty sygnalizacyjne SIP podczas łączności bezpośredniej pomiędzy terminalami SIP

- (a)
- [-] Session Initiation Protocol (INVITE)
 - [+] Request-Line: INVITE sip:jakis3@192.168.1.3 SIP/2.0
 - [+] Message Header
 - [+] Message Body
 - [-] Session Description Protocol
 - Session Description Protocol version (v): 0
 - [+] Owner/Creator, Session Id (o): jakis2 2165 931 IN IP4 10.0.3.15
 - Session Name (s): Talk
 - [+] Connection Information (c): IN IP4 10.0.3.15
 - [+] Time Description, active time (t): 0 0
 - [+] Media Description, name and address (m): audio 7078 RTP/AVP 124 111 110 0 8 101
 - [+] Media Attribute (a): rtpmap:124 opus/48000
 - [+] Media Attribute (a): fmp:124 useinbandfec=1; usedtx=1
 - [+] Media Attribute (a): rtpmap:111 speex/16000
 - [+] Media Attribute (a): fmp:111 vbr=on
 - [+] Media Attribute (a): rtpmap:110 speex/8000
 - [+] Media Attribute (a): fmp:110 vbr=on
 - [+] Media Attribute (a): rtpmap:101 telephone-event/8000
 - [+] Media Attribute (a): fmp:101 0-15
 - (b)
 - [-] Media Description, name and address (m): audio 7078 RTP/AVP 124 111 110 0 8 101
 - Media Type: audio
 - Media Port: 7078
 - Media Protocol: RTP/AVP
 - Media Format: DynamicRTP-Type-124
 - Media Format: DynamicRTP-Type-111
 - Media Format: DynamicRTP-Type-110
 - Media Format: ITU-T G.711 PCMU
 - Media Format: ITU-T G.711 PCMA
 - Media Format: DynamicRTP-Type-101
 - [-] Media Attribute (a): rtpmap:124 opus/48000
 - Media Attribute Fieldname: rtpmap
 - Media Format: 124
 - MIME Type: opus
 - Sample Rate: 48000

Rys. 4. Przykładowy opis sesji SIP przekazany przez terminal PC1 do terminala PC2 w komunikacie „Request: Invite SDP”: a) lista parametrów, b) przykładowa zawartość opisu mediów i pierwszego z podanych przez SDP atrybutów

W komunikatach „Request: Invite SDP” i „Status: 200 OK SDP” są przesyłane informacje w formacie SDP, które opisują możliwości transmisji mediów przez poszczególne telefony (rys. 4). Opis sesji zawiera informacje identyfikujące system i użytkownika, który daną sesję stworzył, nazwę sesji i informacje o wykorzystywanych mediach (rys. 4a). Warto zwrócić uwagę, że protokół SDP,

w opisie sesji, wskazuje jako tworzący sesję terminal o adresie 10.0.3.15. Jest to adres drugiej karty sieciowej posiadanej przez terminal PC1, która nie bierze udziału w transmisji. Ze względu na to, że jest to pierwsza karta w systemie (posiada niższy identyfikator w liście interfejsów) została ona użyta do identyfikacji systemu. Podobną zasadę stosuje się też w innych rozwiązaniach sieciowych (np. w protokołach routingu), gdzie w systemach posiadających wiele interfejsów, jeden z interfejsów używany jest do identyfikacji systemu. Przedstawiony w komunikacie (rys. 4b) opis możliwości terminala wskazuje na dostępne metody kodowania i kompresji (np. G. 711) używany podczas transmisji protokół transportowy (protokół RTP ze standardowym profilem do transmisji audio [1]) czy częstotliwość próbkowania (48 kHz).

Wprowadzenie do systemu SIP uniwersalnego serwera realizującego funkcje proxy, serwera rejestracji i lokalizacji (serwer S1 na Rys. 2b) pozwala na użycie adresacji AOR w obrębie jednej domeny. Łączenie się z użytkownikami z innych domen wymaga innych komponentów systemu SIP (m.in. dedykowanej usługi DNS). W analizowanym systemie jako serwery SIP wykorzystano oprogramowanie Kamailio uruchomione na dedykowanych systemach komputerowych pełniących role serwerów.

```
/* dodawanie aliasów domeny */
alias=my.pl

/* wskazanie adresu, na którym
pracuje serwer SIP */
listen=192.168.1.12

/* wskazanie portu, na którym
nasłuchuje serwer SIP */
port=5060
```

Rys. 5. Przykładowa konfiguracja serwera SIP wykorzystującego oprogramowanie Kamailio zawarta w pliku *kamailio.cfg*

Konfiguracja serwera SIP wykorzystującego oprogramowanie Kamailio jest przechowywana w pliku *kamailio.cfg* (Rys. 5). Aby nasz serwer SIP obsługiwał domenę *my.pl* należy w pliku *kamailio.cfg* ustawić naszą domenę (słowo kluczowe *alias*) i podać adres IP naszego serwera SIP (słowo kluczowe *listen*). Oprócz tego w konfiguracji wskazywany jest port, na którym serwer SIP nasłuchuje (słowo kluczowe *port*).

W systemie z serwerem SIP, użytkownik po uruchomieniu terminala SIP, poprzez jego funkcje, loguje się na serwerze SIP. Przykładowo gdy użytkownik *teodor* z domeny *my.pl* rejestruje się na serwerze SIP wysyła komunikat rejestracyjny (Rys. 6). Jak widać na rysunku 6 przesyłane są podstawowe dane użytkownika. Wskazywany jest również typ terminala wykorzystywany przez użytkownika (tu: *Linphone*).

Aby system oparty o protokół SIP był w pełni funkcjonalny ko-

```
Session Initiation Protocol (REGISTER)
Request-Line: REGISTER sip:my.pl SIP/2.0
Method: REGISTER
Request-URI: sip:my.pl
[Resent Packet: False]
Message Header
Via: SIP/2.0/UDP 192.168.1.3:5060;branch=z9hg4bk.whke-M8dJ;rport
From: <sip:teodor@my.pl>;tag=p2S8stoQo
To: sip:teodor@my.pl
CSeq: 192 REGISTER
Call-ID: Tc9e4YZB~K
Max-Forwards: 70
Supported: replaces, outbound
Contact: <sip:teodor@192.168.1.3>;+sip.instance="<urn:uuid:c8e6c976-f19e-475e-99b9-f0d222664bc0>"
Expires: 3600
User-Agent: Linphone/3.7.0 (belle-sip/1.3.0)
```

Rys. 6. Przykładowa rejestracja użytkownika na serwerze SIP

```
$TTL 86400
$ORIGIN my.pl.
@ IN SOA dns1.my.pl. root.my.pl. (
    2015091801 ; serial
    2H ; refresh
    1H ; retry
    7D ; expire
    1D ; TTL
)
@ IN NS dns1.my.pl.
@ IN A 192.168.2.15
dns1 IN A 192.168.2.15

sipproxy1.my.pl. 43200 IN A 192.168.1.12
_sip_udp.my.pl. 43200 IN SRV 0 0 5060 sipproxy1.my.pl.
```

Rys. 7. Przykładowa konfiguracja serwera DNS (serwer S2 na rysunku 2c) dla obsługi lokalizacji serwera SIP w domenie *my.pl*

nieczne jest jeszcze odpowiednie skonfigurowanie systemu DNS. System DNS musi umożliwiać lokalizację serwerów SIP dla danej domeny. W tym celu serwery DNS dla obsługi tej funkcjonalności posiadają specjalny typ rekordu SRV zdefiniowany w RFC2782 [2]. Format rekordu jest następujący:

```
_Service._Proto.Name      TTL      Class      SRV
Priority Weight Port Target
```

gdzie: *Service* jest typem usługi (tu: sip), *Proto* oznacza wykorzystywany w usłudze protokół, *Name* oznacza nazwę domeny SIP, *TTL* czas życia wpisu, *Priority* priorytet, *Weight* waga, *Port* port i *Target* nazwę serwera, konwertowaną później na adres IP.

Dla naszej domeny *my.pl* został uruchomiony serwer DNS (serwer S2 na rysunku 2c). Wykorzystano tutaj oprogramowanie *bind9*³. Konfiguracja serwera DNS została przedstawiona na rysunku 7. Serwer ten posiada adres *192.168.2.15*. Serwer DNS dla domeny *my.pl* wskazuje serwer SIP proxy o nazwie *sipproxy1.my.pl* (rekord typu *SRV*). Z nazwą tą w rekordzie typu *A* skojarzony został adres IP serwera SIP proxy *192.168.1.12*.

PODSUMOWANIE

W pełni funkcjonalny system wykorzystujący protokół SIP można zbudować w oparciu o oprogramowanie dostępne na licencji GNU GPL. Tego typu zestaw oprogramowania może być wykorzystywany podczas ćwiczeń laboratoryjnych z przedmiotu „Systemy sygnalizacji i zarządzania” do analizy koncepcji sygnalizacji wykorzystującej protokół SIP. Uniwersalność modułów prezentowanego w artykule oprogramowania i dostępność kodów źródłowych pozwala na wykorzystanie tego zestawu programów do realizacji badań naukowych z zakresu sieci i usług telekomunikacyjnych.

BIBLIOGRAFIA

1. Chodorek R.R., Pach A.R., *Transmisja multikastowa w sieciach IP*. Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 2003.
2. Gulbrandsen A., Vixie P., Esibov L., *A DNS RR for specifying the location of services (DNS SRV)*. RFC 2782, February 2000.
3. ITU-T, *Packet-Based Multimedia Communications Systems*. ITU-T Recommendation H.323, 2000.
4. Resnick P. (red.), *Internet Message Format*. RFC 2822, April 2001.
5. Rosenberg J., Schulzrinne H., Camarillo G., Johnston A., Peterson J., Sparks R., Handley M., Schooler E., *SIP: Session Initiation Protocol*. RFC 3261, June 2002.
6. Rosenberg J., *Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols*. RFC 5245, April 2010.
7. Stasiak A., Skowroński Z., *Wirtualizacja - kierunek rozwoju platform n-procesorowych*. Przegląd telekomunikacyjny, vol. LXXXI, nr 6, 2008.

ANALYSIS OF SELECTED ELEMENTS OF THE VOICE COMMUNICATION SYSTEM WHICH USES THE SIP PROTOCOL

Abstract

This article is devoted to the analysis of voice communication system based on SIP protocol. It is now the most common method of voice communication. The article presents the basic components of the system using SIP protocol. Indicated on selected publicly available SIP solutions components - Linphone SIP terminal and the SIP server Kamailio. These solutions have been functionally tested in test network.

The solutions can be used in the commercial systems. Due to the full availability of the SIP components source code (Linphone and Kamailio) they can be used for teaching and research.

Autorzy:

dr inż. **Agnieszka Chodorek** – Politechnika Świętokrzyska, Wydział Elektrotechniki, Automatyki i Informatyki, Katedra Systemów Informatycznych; 25-314 Kielce; al. Tysiąclecia Państwa Polskiego 7. E-mail: a.chodorek@tu.kielce.pl

dr inż. **Robert Chodorek** – AGH Akademia Górniczo-Hutnicza, Wydział Informatyki, Elektroniki i Telekomunikacji, Katedra Telekomunikacji; 30-059 Kraków; Al. A. Mickiewicza 30. E-mail: chodorek@agh.edu.pl

³ <http://www.bind9.net/>