VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

NÁVRH A REALIZACE SÍTĚ IMS S ROZŠÍŘENÝMI FUNKCEMI POMOCÍ OPEN IMS CORE.

DESIGN AND IMPLEMENTATION OF AN IMS NETWORK WITH EXTENDED FUNCTIONS USING OPEN IMS CORE.

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **Peter Šulgan**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Jiří Ježek**
SUPERVISOR

BRNO 2021

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní program **Telekomunikační a informační systémy**

Ústav telekomunikací

Student: Peter Šulgan ***ID:*** 211274
Ročník: 3 ***Akademický rok:*** 2020/21

NÁZEV TÉMATU:

Návrh a realizace sítě IMS s rozšířenými funkcemi pomocí Open IMS Core.

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte technologii IMS (IP Multimedia Subsystem). Na základě získaných znalostí navrhněte IMS síť implementující rozšířené funkce IMS (např. mediální brány, vybrané aplikační servery, mediální servery). Navrženou síť realizujte s pomocí open source IMS projektu Open IMS Core. Prakticky otestujte funkčnost navržené sítě.

DOPORUČENÁ LITERATURA:

[1] Gonzalo Camarillo and Miguel A. García-Martín. The 3G IP multimedia subsystem (IMS). Wiley, Chichester, 2.edice edition, 2006. JohnWiley & Sons Ltd. ISBN: 0-470-01818-6

[2] VINGARZAN, Dragos; WEIK, Peter; MAGEDANZ, Thomas. Design and implementation of an open IMS core. In: International Workshop on Mobile Agents for Telecommunication Applications. Springer, Berlin, Heidelberg, 2005. p. 284-293.

Termín zadání: 1.2.2021 ***Termín odevzdání:*** 31.5.2021

Vedoucí práce: Ing. Jiří Ježek

prof. Ing. Jiří Mišurec, CSc.
předseda rady studijního programu

ABSTRAKT

Bakalárska práca sa zameriava na architektúru IP Multimedia Subsystem, jej hlavné prvky a rozširujúce funkcie. Ďalej sa zaoberá implementáciou IMS siete v simulačnom prostredí Open IMS Core, ktoré zároveň stručne popisuje.
V prvej časti je charakteristika IMS architektúry, rozdelenie na vrstvy a definícia základných prvkov CSCF a HSS. Okrem toho stručne charakterizuje najčastejšie používané komunikačné protokoly a na koniec rozoberá štruktúru identity v IMS.
Druhá časť sa venuje open source prostrediu Open IMS Core, stručne definuje jednotlivé funkčné časti a obsahuje zoznam konfiguračných súborov.
Tretia časť bližšie zoznamuje s doplnkovými službami, ktoré ponúka architektúra IMS. Charakterizuje ich základné funkcie a spôsob práce.
Štvrtá časť sa venuje praktickej zložke bakalárskej práce. Opisuje postup konfigurácie Open IMS Core a testovanie funkčnosti. Je rozdelená na štyri časti podľa jednotlivých krokov konfigurácie.

KĽÚČOVÉ SLOVÁ

CSCF, DIAMETER, HSS, IMS, IMS presence, IMS zabezpečenie, IP Multimedia Subsystem, Núdzový hovor, Open IMS Core, SIP, URI

ABSTRACT

The bachelor thesis focuses on the architecture of the IP Multimedia Subsystem, its main elements and extension functions. It also deals with the implementation of the IMS network in the simulation environment Open IMS Core, which is also briefly described.
The first chapter describes the IMS architecture divided into layers and it also defines the basic elements such as CSCF and HSS. In addition, it briefly characterizes the most commonly used communication protocols and the structure of identity in IMS.
The second chapter deals with the open source environment Open IMS Core, it briefly defines the individual parts and contains a list of configuration files.
The third chapter introduces the additional services offered by the IMS architecture. It characterizes their basic functions and their way of working.
The fourth chapter focuses on the practical part of the bachelor thesis. It describes configuration procedure of the Open IMS Core and testing of its functionality. It is divided into four sections according to the individual configuration steps.

KEYWORDS

CSCF, DIAMETER, Emergency call, HSS, IMS, IMS presence, IMS security, IP Multimedia Subsystem, Open IMS Core, SIP, URI

ŠULGAN, Peter. *Návrh a realizace sítě IMS s rozšířenými funkcemi pomocí Open IMS Core*. Brno, 2021, 68 s. Bakalárska práca. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedúci práce: Ing. Jiří Ježek

VYHLÁSENIE

Vyhlasujem, že svoju bakalársku prácu na tému „Návrh a realizace sítě IMS s rozšířenými funkcemi pomocí Open IMS Core“ som vypracoval samostatne pod vedením vedúceho bakalárskej práce, s využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej bakalárskej práce ďalej vyhlasujem, že v súvislosti s vytvorením tejto bakalárskej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a/alebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona Českej republiky č. 121/2000 Sb., o práve autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávnych dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákonníka Českej republiky č. 40/2009 Sb.

Brno ....

podpis autora

POĎAKOVANIE

Rád by som poďakoval vedúcemu bakalárskej práce pánovi Ing. Jiřímu Ježkovi za trpezlivosť, konzultácie a odborné vedenie pri práci.

Obsah

Úvod **11**

1 IP Multimedia Subsystem (IMS) **12**
1.1 Charakteristika IMS . . . 12
1.2 Vrstvy IMS . . . 12
1.2.1 Vrstva pripojenia . . . 12
1.2.2 Riadiaca vrstva . . . 13
1.2.3 Aplikačná vrstva . . . 13
1.3 Hlavné prvky architektúry IMS . . . 14
1.3.1 Home Subscriber Server (HSS) . . . 14
1.3.2 Call Session Control Function (CSCF) . . . 15
1.3.3 Media Resource Function (MRF) . . . 17
1.3.4 Aplikačný server (AS) . . . 17
1.3.5 IMS brány . . . 18
1.3.6 Subscription Locator Function (SLF) . . . 20
1.3.7 Policy Decision Function (PDF) . . . 20
1.4 Protokoly IMS . . . 20
1.4.1 Session Initiation Protocol (SIP) . . . 20
1.4.2 DIAMETER . . . 21
1.4.3 Session Description Protocol (SDP) . . . 22
1.4.4 Real-time Transport Protocol (RTP) . . . 22
1.5 Identita IMS . . . 23
1.5.1 Public User Identities . . . 23
1.5.2 Private User Identities . . . 23
1.5.3 Vzťah medzi private UI a public UI . . . 24
1.5.4 Public Service Identities (PSI) . . . 25
1.6 Domáce a navštívené siete . . . 25

2 Základná architektúra Open IMS Core **27**
2.1 HSS . . . 27
2.2 MySQL databáza . . . 28
2.3 CSCF . . . 28
2.3.1 Proxy-CSCF . . . 28
2.3.2 Interrogating-CSCF . . . 28
2.3.3 Serving-CSCF . . . 28
2.3.4 CDiameterPeer . . . 28
2.3.5 IMS Service Control . . . 29

2.4 Možnosti konfigurácie HSS 29
2.5 Možnosti konfigurácie CSCF 29

3 Doplnkové služby **30**
3.1 Emergency-CSCF (E-CSCF) 30
3.2 IMS presence architektúra 32

4 Implementácia IMS architektúry pomocou Open IMS Core **36**
4.1 Konfigurácia localhostu 36
4.2 Konfigurácia na IP adrese 192.168.20.179 40
4.3 Konfigurácia E-CSCF 45
4.4 Konfigurácia zabezpečenia IPsec, TLS a THIG 49
4.4.1 IPsec zabezpečenie 49
4.4.2 TLS zabezpečenie 51
4.4.3 THIG (Topology Hiding Inter-network Gateway) 52

Záver **54**

Literatúra **57**

Zoznam symbolov, veličín a skratiek **59**

A Skrátené výpisy konfiguračných súborov **63**

B Konfiguračné súbory Open IMS Core **67**
B.1 Konfiguračné súbory DNS 67
B.2 Konfiguračné súbory CSCF 67
B.3 Konfiguračné súbory HSS 67

Zoznam obrázkov

1.1 Rozdelenie IMS na vrstvy 13
1.2 Architektúra IMS 14
1.3 Spojenie s PSTN cez IMS brány 19
1.4 Vzťah medzi private UI a public UI v 3GPP R5 [3] 24
1.5 Vzťah medzi private UI a public UI v 3GPP R6 [3] 25
2.1 Základná architektúra Open IMS Core [11] 27
3.1 Emergency architektúra Open IMS Core 31
3.2 Presence architektúra [13] 33
3.3 Proces zverejnenia informácií (publishing) [13] 34
3.4 Proces požiadania o informácie (subscribing) [13] 35
4.1 Nadviazanie a ukončenie hovoru medzi Alicou a Bobom 39
4.2 Nadviazanie a ukončenie hovoru medzi Bobom a Adamom 43
4.3 Flow diagram hovoru medzi Bobom a Adamom 44
B.1 Adresárová štruktúra elektronickej prílohy 68

Zoznam výpisov

4.1 Nastavenie DNS v súbore `named.conf` . . . 36
4.2 Nastavenie DNS resolveru v súbore `resolv.conf` . . . 36
4.3 Ping na adresu `open-ims.test` a `pcscf.open-ims.test` . . . 37
4.4 Spustenie SQL skriptov . . . 37
4.5 Definovanie premenných `JAVA_HOME` a `PATH` . . . 37
4.6 Zmena portu 8080 v súbore `hss.properties` . . . 38
4.7 Zmena portu 3868 v súbore `DiameterPeerHSS.xml` . . . 38
4.8 Zmena portu 3868 v súbore `scscf.xml` . . . 38
4.9 Zmena portu 3868 v súbore `icscf.xml` . . . 38
4.10 Výpis registrar obsahu na S-CSCF . . . 38
4.11 Nastavenie DNS resolveru v súbore `resolv.conf` . . . 40
4.12 Zmeny v súbore `DiameterPeerHSS.xml` . . . 40
4.13 Konfigurácia P-CSCF v súbore `pcscf.xml` . . . 41
4.14 Konfigurácia S-CSCF v súbore `scscf.xml` . . . 41
4.15 Konfigurácia I-CSCF v súbore `icscf.xml` . . . 41
4.16 Ping na adresu `oims` a `pcscf.oims` . . . 41
4.17 Výpis registrar obsahu s novým užívateľom Adam . . . 42
4.18 Definovanie premenných `JAVA_HOME` a `PATH` . . . 46
4.19 Zakomentované riadky v súbore `userdata.sql` . . . 46
4.20 Implicitná definícia premennej `JAVA_HOME` v súbore `startup.sh` . . . 46
4.21 Chyby pri načítaní súboru `lib_lost_client.so` . . . 47
4.22 Kontrola modulu curl a knižnice libcurl . . . 47
4.23 Kontrola prepojenia modulu curl a knižnice libcurl . . . 47
4.24 Konfigurácia E-CSCF v súbore `ecscf.cfg` . . . 48
4.25 Konfigurácia LRF v súbore `lrf.cfg` . . . 49
4.26 Implementácia IPsec zabezpečenia v P-CSCF . . . 50
4.27 Spôsob implementácie IPsec v P-CSCF . . . 50
4.28 Spôsob implementácie IPsec v S-CSCF . . . 51
4.29 Spôsob autentizácie v S-CSCF . . . 51
4.30 Základné nastavenie TLS v P-CSCF . . . 52
4.31 Načítanie TLS modulu a certifikátov v P-CSCF . . . 52
4.32 Konfigurácia THIG v I-CSCF . . . 53
4.33 Výpis komunikácie bez a s použitím THIG . . . 53
A.1 Nastavenie DNS v súbore `named.conf` . . . 63
A.2 Nastavenie DNS v súbore `named.conf.options` . . . 63
A.3 Nastavenie DNS v súbore `open-ims.dnszone` . . . 63
A.4 Zmeny v súbore `hss.properties` . . . 64

A.5 Zmeny v súbore `userdata.sql` 64
A.6 Konfigurácia P-CSCF v súbore `pcscf.cfg` 64
A.7 Konfigurácia S-CSCF v súbore `scscf.cfg` 65
A.8 Konfigurácia I-CSCF v súbore `icscf.cfg` 66
A.9 Konfigurácia I-CSCF v súbore `icscf.sql` 66
A.10 Zmena autentizačného algoritmu v `scscf.cfg` 66

Úvod

Dopyt verejnosti po multimediálnych službách neustále rastie, nezávisle od zariadenia, typu pripojenia alebo polohy, je žiadúce sprístupniť ich užívateľom. To vytvára nové požiadavky na sieťovú architektúru.

Telefónne siete narazili na viaceré prekážky, ale siete postavené na štandarde IP ich v mnohom prekonávajú. Potreba systému nezávislého od typu siete, prístupovej technológie, či aplikačnej služby mala za následok vytvorenie štandardizovanej architektúry IP Multimedia Subsystem.

IMS je špecifikovaný v 3GPP UMTS vydaní 5 a 6, systém využíva výhody internetu a aplikuje ich v mobilných sieťach. Nezávislosť na prístupovej technológii umožnuje ponúkať služby paketovým sieťam aj sieťam s prepájaním okruhov. Ponúka služby ako VoD (Video on Demand), IPTV, PoC (Push-to-talk over Cellular), instant messaging, video konferencie...

Cieľom práce je rozobrať a naštudovať problematiku IMS, následne vytvoriť návrh siete implementujúci rozšírené funkcie. Ďalším cieľom je návrh zrealizovať v prostredí Open IMS Core a prakticky otestovať jeho funkčnosť.

1 IP Multimedia Subsystem (IMS)

IMS je sieťová architektúra pre prenos multimediálnych dát ako hlas, video, text. Zabezpečuje komunikáciu medzi širokou škálou zariadení naprieč rôznymi sieťovými štruktúrami. Je to celok infraštruktúry a riadiacich mechanizmov, ktoré rozhodujú nad smerovaním, spracovaním a riadením jednotlivých spojení medzi užívateľmi.

1.1 Charakteristika IMS

Systém je nezávislý od koncových zariadení a je ho možné použiť nezávisle od prístupovej technológie. Rozsah klasických multimediálnych služieb je rozšírení o nové funkcie, napríklad informácie o dostupnosti účastníka, QoS (Quality of Service), služby AAA (Authentication, Authorization and Accounting). Všetky služby sú prístupné pomocou aplikačných serverov, ktoré spravuje poskytovateľ. IMS dokáže spolupracovať so sieťami s prepínaním paketov ako aj sieťami s prepínaním okruhov, ako napríklad PSTN (Public Switched Telephone Network). Systém bol špecifikovaný v 3GPP UMTS vydaní 5 a 6, s cieľom rozvíjať mobilné siete. Používa viacero doporučení IETF, jedným z nich je napríklad signalizačný protokol SIP (Session Initiation Protocol). [1]

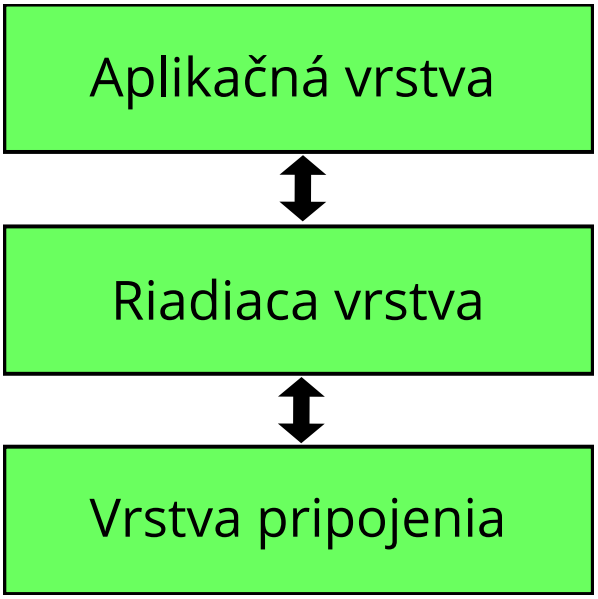
1.2 Vrstvy IMS

Architektúra systému IMS je rozdelená do vrstiev, čo zvyšuje prehľadnosť, škálovateľnosť a kompatibilitu medzi jednotlivými mechanizmami. Vrstvy si medzi sebou vymieňajú informácie a využívajú služby iných vrstiev alebo poskytujú svoje služby. IMS sa delí na 3 vrstvy:

- Vrstva pripojenia
- Riadiaca vrstva
- Aplikačná vrstva

1.2.1 Vrstva pripojenia

Zabezpečuje pripojenie koncových zariadení, prevažne cez IP štandard, a poskytuje prostriedky potrebné na vytvorenie spojenia medzi komunikujúcimi stranami. Pripojenie je nezávislé na prístupovej metóde, základnou požiadavkou je podpora protokolov IP a SIP. Zariadenia bez podpory IP a SIP, napríklad analógové telefóny, sa môžu pripojiť pomocou špecializovaných komunikačných brán. Zároveň zabezpečuje konverziu dát z iných sietí na formát používaný IMS systémom. Najčastejšie sa skladá z IP smerovačov, prípadne brán.



Obr. 1.1: Rozdelenie IMS na vrstvy

1.2.2 Riadiaca vrstva

Zaoberá sa vytváraním a spravovaním spojení medzi koncovými zariadeniami, respektíve ich modifikáciou. Základné prvky tejto vrstvy sú CSCF (Call Session Control Function) a HSS (Home Subscriber Server). CSCF riadi registráciu koncových zariadení a smerovanie správ protokolu SIP na aplikačný server, spravujúci konkrétnu službu. CSCF zároveň spolupracuje s vrstvou pripojenia a zabezpečuje AAA a QoS pre všetky služby, ktoré to požadujú. HSS obsahuje databázu s profilmi všetkých užívateľov, informuje o polohe užívateľa, o autorizácii, o vyžadovaných službách... [2]

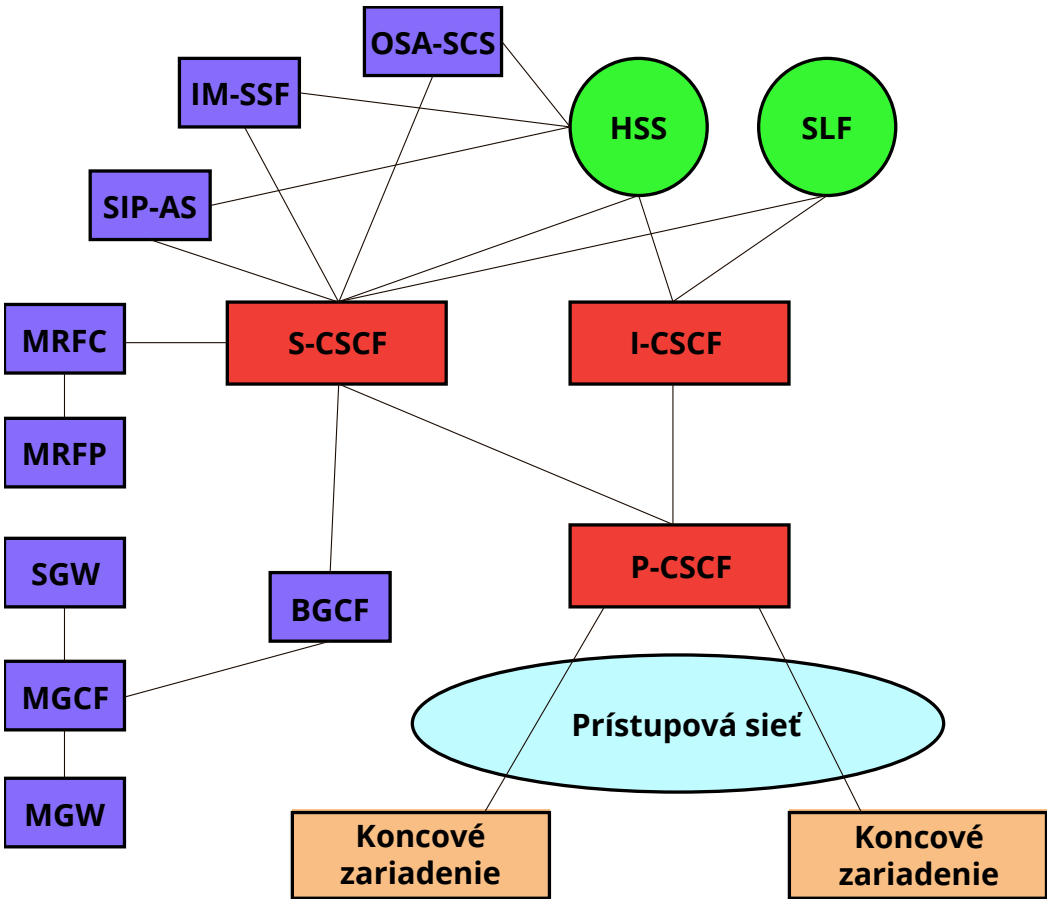
1.2.3 Aplikačná vrstva

Využíva servery na poskytovanie rôznych služieb a funkcií. Základné používané prvky sú AS (aplikačné servery), MRFC (Multimedia Resource Function Controller) a MRFP (Multimedia Resource Function Processor). AS riadi funkcie špecifické pre konkrétnu službu a interaguje s konkrétnym užívateľom. MRFP zabezpečuje doplnkové spracovanie multimediálnych dát, zároveň môže poskytovať video služby, textové služby, video konferencie, personalizované vyzváňacie tóny a iné. MRFC vystupuje ako SIP UA (User Agent) a komunikuje s CSCF, následne riadi a priraďuje prostriedky, ktorými disponuje MRFP. [2][3]

1.3 Hlavné prvky architektúry IMS

Medzi základné prvky patrí:

- Home Subscriber Server (HSS)
- Call Session Control Function (CSCF)
- Media Resource Function (MRF)
- Aplikačný server (AS)
- IMS brány
- Subscription Locator Function (SLF)
- Policy Decision Function (PDF)



Obr. 1.2: Architektúra IMS

1.3.1 Home Subscriber Server (HSS)

HSS je hlavné úložisko dát, týkajúcich sa užívateľov a služieb. Databáza obsahuje profily užívateľov, registračné informácie, prístupové informácie jednotlivých užívateľov a iné parametre, potrebné na vytváranie multimediálnych spojení. Ukladá

adresu S-CSCF (Serving-CSCF), ku ktorej je priradený užívateľ. HSS poskytuje nasledovné funkcie:

- identifikácia užívateľa, adresovanie
- informácie o autorizácii a autentizácii
- informácie o polohe užívateľa
- registračné informácie
- profil užívateľa (identita, vyžadované služby, konkrétne parametre služieb)

IMS architektúra môže obsahovať väčší počet HSS, ak je počet užívateľov príliš veľký. V tom prípade sa používa SLF (Subscription Locator Function) na vyhľadávanie konkrétneho HSS. Informácie o užívateľovi, sú vždy uložené iba na jednom HSS. Na komunikáciu s HSS sa využíva protokol DIAMETER. [3][4]

1.3.2 Call Session Control Function (CSCF)

CSCF je centrálny prvok celého IMS systému, nazývaný aj SIP server. Hlavnou úlohou je vytváranie, riadenie, monitorovanie a smerovanie signalizačných tokov protokolu SIP. CSCF poskytuje nasledovné funkcie:

- riadenie spojenia (registrácia, smerovanie, účtovanie, roaming)
- kombinovanie mediálnych tokov v rámci spojenia
- autentizácia užívateľa (USIM/ISIM)
- QoS pomocou PDF (Policy Decision Function)

ISC (IP Multimedia Service Control) rozhranie pomocou filtrov a SIP signalizácie zabezpečuje nezávislé fungovanie viacerých aplikačných služieb. ISC definuje filtre, priradené každému užívateľovi a uložené v HSS, pomocou ktorých CSCF rozhoduje, ktorá služba je vyžadovaná. Vzľadom na funkcionalitu rozlišujeme tri základné typy CSCF [3][5]:

- Proxy-CSCF (P-CSCF)
- Interrogating-CSCF (I-CSCF)
- Serving-CSCF (S-CSCF)

Proxy-CSCF (P-CSCF)

P-CSCF je prvým kontaktným bodom medzi koncovým zariadením a IMS sieťou. Všetky SIP správy od SIP UA (User Agent) prechádzajú cez P-CSCF, ktorý ich preposiela ostatným zariadeniam. Počas registrácie koncové zariadenie komunikuje iba s jedným P-CSCF, ktoré ho autentizuje a oznámi ostatným uzlom jeho identifikáciu. To zaručuje, že ostatné uzly sa nemusia zaoberať overovaním užívateľa,

pretože dôverujú P-CSCF. IMS sieť môže obsahovať väčší počet P-CSCF, čo zabezpečuje redundanciu a škálovateľnosť. Každé P-CSCF obsluhuje počet užívateľov úmerný svojmu výpočtovému výkonu. P-CSCF sa môže nachádzať v domácej sieti alebo v navštívenej sieti. P-CSCF zabezpečuje [2][3]:

- smerovanie SIP správ (tzv. sprostredkovateľ)
- modifikáciu SIP správ, aby splňovali pravidlá
- overenie správnosti SIP správ
- kompresiu/dekompresiu SIP správ
- integritu komunikácie s užívateľom pomocou IPsec/EPS
- autentizáciu a identifikáciu užívateľa
- detekciu polohy užívateľa (domáca/navštívená sieť)
- spravovanie QoS pomocou PDF (voliteľné)
- generovanie spoplatňovacích informácií
- detekciu tiesňových volaní
- určitý stupeň ochrany pred útokmi

Interrogating-CSCF (I-CSCF)

I-CSCF je bránou pre všetky spojenia smerované na daného operátora. Jeho adresa je uložená v DNS záznamoch. I-CSCF konktaktuje HSS pri pripojení užívateľa s požiadavkou na adresu S-CSCF, ktoré je priradené užívateľovi. V prípade, že ešte nemá priradené žiadne S-CSCF, I-CSCF mu ho priradí. Po určení S-CSCF už nemusia správy prechádzať cez I-CSCF. Komunikácia s HSS a SLF využíva protokol DIAMETER. I-CSCF zároveň poskytuje funkcionalitu THIG (Topology Hiding Inter-network Gateway), ktorá umožnuje skrytie informácií o sieti, počte serverov, DNS a iných. Pri použití THIG komunikácia vždy prechádza cez I-CSCF. Okrem spomenutých môže posielať aj spoplatňovacie informácie. IMS sieť často obsahuje viac I-CSCF z dôvodu redundancie, najčastejšie sú umiestnené v domácej sieti (výnimka je napríklad pri použití THIG). [2][3]

Serving-CSCF (S-CSCF)

S-CSCF je hlavným článkom riadenia a kontroly spojení. Všetky SIP správy, ktoré užívateľ prijme alebo pošle spracuje S-CSCF. Vykonáva nasledujúce hlavné úlohy:

- registrácia a pripojenie: S-CSCF vystupuje ako SIP Registrar, prijíma SIP registračné žiadosti, registruje užívateľa a sprístupňuje informácie o profile užívateľa pomocou HSS, identifikuje užívateľa pomocou autentizačných vektorov z HSS.
- kontrola spojenia: S-CSCF kontroluje prebiehajúce spojenia registrovaných užívateľov, preposiela SIP request a response správy medzi volaným a vola-

júcim, zabraňuje vykonávať nepovolené operácie alebo používať nepovolené typy médií a kodekov, sleduje parametre potrebné na účtovanie.
- SIP proxy: preposiela správy medzi účastníkmi a ostatnými CSCF alebo SIP servermi.
- prístup k AS: S-CSCF riadi prístup k AS a ich službám, kontroluje všetky SIP správy a smeruje ich na konkrétny AS, sprostredkuje užívateľom dostupné služby.

Okrem spomenutého, S-CSCF poskytuje prekladové služby, založené na DNS E.164 Number Translation, ak užívateľ použije adresu v inom tvare ako SIP URI (Uniform Resource Identifier), napríklad telefónne číslo. Tiež uchováva prepojenie medzi lokáciou užívateľa (napríklad IP adresou) a SIP adresou (tzv. Public User Identity). Na komunikáciu s HSS používa protokol DIAMETER. IMS sieť často používa viac S-CSCF, pričom každý môže poskytovať odlišné funkcie. S-CSCF sa nachádza vždy v domácej sieti. [2][3]

1.3.3 Media Resource Function (MRF)

MRF je univerzálny mediálny server, ktorý slúži na manipuláciu s multimediálnymi dátami a dátovými tokmi. Vždy je lokalizovaný v domácej sieti. MRF sa delí na:
- Media Resource Function Controller (MRFC): vyhodnocuje inštrukcie a SIP správy od AS a S-CSCF, podľa ktorých rozhoduje aké operácie má vykonať MRFP. Nachádza sa v aplikačnej vrstve a s MRFP komunikuje cez protokol H.248/Megaco.
- Media Resource Function Processor (MRFP): poskytuje multimediálne funkcionality, napríklad zlučovanie mediálnych tokov, analýzu médií, preklad medzi rôznymi kodekmi, vytváranie štatistík. Nachádza sa na vrstve pripojenia.

Zároveň môže MRFC spoplatňovať poskytovanie prostriedkov MRFP, sledovať roaming užívateľa a riadiť konferenčné komunikačné spojenia. [3][5]

1.3.4 Aplikačný server (AS)

AS je systém poskytujúci konkrétne služby, ktorými disponuje. Vzhľadom na typ služby môže vystupovať ako SIP proxy, SIP UA alebo SIP B2BUA (Back-to-Back UA). Na komunikáciu s S-CSCF využíva protokol SIP. Existujú 3 typy AS:
- SIP AS: najbežnejší a pôvodný AS, ktorý poskytuje IMS služby založené na protokole SIP. Očakáva sa, že väčšina nových IMS služieb bude vyvinutých pre SIP AS.

- OSA-SCS (Open Service Access-Service Capability Server): poskytuje spojenie s OSA AS. Zahrňuje OSA funkcionality, napríklad zabezpečený prístup do IMS z externých sietí. Smerom k S-CSCF sa správa ako AS, smerom k OSA AS ako rozhranie na prístup do IMS.
- IM-SSF (IP Multimedia Service Switching Function): špecializovaný AS poskytujúci využitie služieb CAMEL (Customized Applications for Mobile network Enhanced Logic), ktoré boli vyvinuté pre GSM (Global System for Mobile Communications). Smerom k S-CSCF sa správa ako AS, opačne vystupuje ako SSF (Service Switching Function) a pripája gsmSCF (GSM Service Control Function) pomocou protokolu založenom na CAP (CAMEL Application Part). IM-SSF umožňuje gsmSCF riadiť IMS spojenie.

AS môže byť pripojený aj na HSS. SIP AS a OSA-SCS používajú protokol DIAMETER na prijímanie alebo posielanie dát smerom k HSS, IM-SSF rozhranie je založené na MAP (Mobile Application Part). AS môže byť lokalizovaný v domácej alebo externej sieti. V prípade lokalizácie v externej sieti AS nemá priame spojenie s HSS. [3]

1.3.5 IMS brány

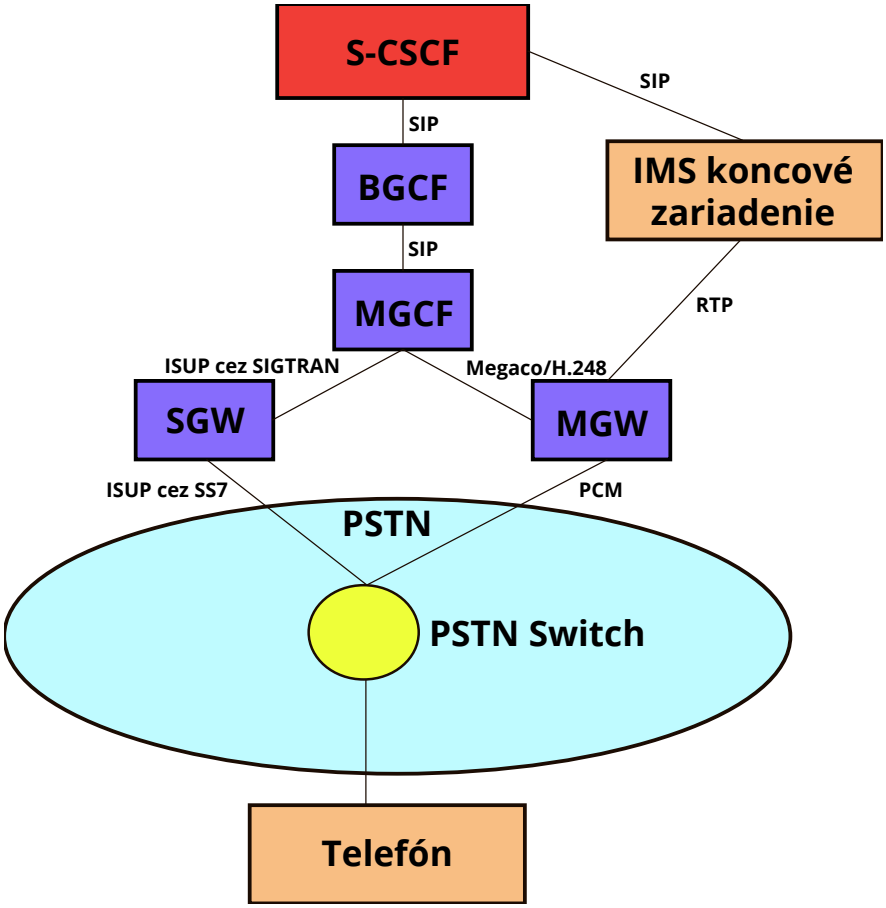
Na rozhraní vrstvy pripojenia a vonkajších sietí sa nachádza systém IMS brán. Jednotlivé časti sú:

Media Gateway (MGW)

MGW je pripojené na MGCF (Media Gateway Control Function), ktoré priamo riadi spôsob spracovania dát. Slúži ako rozhranie medzi IMS IP sieťou a sieťami s prepínaním okruhov alebo na prepojenie IMS siete s inými paketovými sieťami. Hlavnou úlohou je prekladanie toku dát z jedného formátu na druhý (najčastejšie preklad formátu dát zo siete s prepínaním okruhov na pakety alebo preklad medzi rôznymi IP kodekmi). Zároveň umožnuje komunikáciu IMS zariadenia so zariadením nepodporujúcim IMS štandard. Smerom k IMS zariadeniu má vytvorené RTP (Real-time Transport Protocol) spojenie, smerom k PSTN sieti využíva PCM (Pulse Code Modulation) time sloty. [3][6]

Signaling Gateway (SGW)

SGW poskytuje preklad signalizačných správ medzi IMS sieťou a sieťami s prepínaním okruhov, napríklad PSTN. Neprekladá správy aplikačnej vrstvy. SGW si s PSTN sieťou vymieňa správy pomocou protokolu SS7. Smerom k IMS používa na



Obr. 1.3: Spojenie s PSTN cez IMS brány

komunikáciu protokol SIGTRAN (Signaling Transport over IP). SGW je pripojené na MGCF. [2]

Media Gateway Control Function (MGCF)

MGCF poskytuje centrálne riadenie komunikačných spojení pre všetky IMS brány a slúži ako logický prvok pri komunikácii IMS siete so sieťou s prepínaním okruhov (prekladá a mapuje SIP správy a ISUP (ISDN User Part) správy). MGCF je pripojené na BGCF (Breakout Gateway Control Function), MGW a SGW. Riadi MGW (metóda master-slave) pomocou protokolu Megaco/H.248. Signalizáciu posiela cez SGW pomocou protokolu SIGTRAN. [6]

Breakout Gateway Control Function (BGCF)

BGCF je SIP server, ktorý zabezpečuje smerovanie založené na telefónnych číslach. BGCF určuje, v ktorej sieti bude vytvorené spojenie so sieťou s prepínaním okruhov. Ak určí sieť, v ktorej sa samo nachádza, priamo vyberie MGCF, ktoré bude spojenie

spravovať. Pri určení vonkajšej siete preposiela informácie na BGCF danej siete. BGCF vstupuje do komunikácie iba v prípade, že spojenie do PSTN siete inicializuje IMS koncové zariadenie, nie naopak. [3][6]

1.3.6 Subscription Locator Function (SLF)

SLF je vyhľadávací mechanizmus (mapovacia databáza), vyžadovaný ak IMS sieť obsahuje viac ako jeden HSS. I-CSCF, S-CSCF alebo AS využívajú SLF pri hľadaní adresy konkrétneho HSS, ktorý má v databáze uložený profil užívateľa pokúšajúceho sa o pripojenie alebo využívanie služieb. Na komunikáciu využíva protokol DIAMETER.

1.3.7 Policy Decision Function (PDF)

Dohliada na spojenie medzi užívateľom a P-CSCF a určuje parametre jednotlivých spojení, ktoré je užívateľ autorizovaný vytvárať. Každá nepovolená komunikácia je odfiltrovaná a zahodená. PDF môže byť implementované v P-CSCF alebo môže predstavovať samostatný uzol. [3]

1.4 Protokoly IMS

IMS architektúra, namiesto vytvorenia nových protokolov, využíva štandardizované protokoly skupín IETF a ITU-T. Systém je založený na IP paketovo orientovanej komunikácii, preto základným protokolom sieťovej vrstvy je protokol IP. Využívané transportné protokoly sú TCP, UDP a SCTP (Stream Control Transmission Protocol). Vyššie vrstvy používajú protokoly ako SIP, DIAMETER, SDP, RTP, RTCP, Megaco/H.248.

1.4.1 Session Initiation Protocol (SIP)

SIP bol špecifikovaný skupinou IETF (Internet Engineering Task Force), je definovaný v RFC 3261. Jedná sa o signalizačný protokol aplikačnej vrstvy, ktorý vytvára, upravuje a ukončuje multimediálne spojenia, pripája účastníkov do existujúcich spojení, riadi presmerovanie spojenia a rôzne ďalšie úlohy. Medzi základné informácie, ktoré SIP zisťuje a poskytuje patria:

- lokalizácia užívateľa: určenie koncových zariadení, ktoré budú komunikovať.
- dostupnosť užívateľa: zistenie stavu dostupnosti (obsadené, dostupný, presmerovanie) a či sa chce užívateľ pripojiť.
- možnosti užívateľa: určenie typu spojenia a parametrov spojenia, s ktorými dokáže užívateľ pracovať (typ dát, typ kodeku, prenosové rýchlosti...).

- vytvorenie spojenia: informovanie o začatí spojenia, zvolenie parametrov spojenia.
- riadenie spojenia: riadenie prenosu, modifikovanie parametrov spojenia, volanie služieb, ukončenie spojenia.

SIP spolupracuje s protokolmi ako RTP, RTCP, SDP a Megaco/H.248. Využíva model klient-server a štýl komunikácie žiadosť-odpoveď. Keďže ide o textovo orientovaný protokol vychádzajúci z HTTP, je možné využívať frameworky určené pre HTTP protokol. Je kompatibilný s IPv4 aj IPv6 a dokáže pracovať s transportnými protokolmi TCP, UDP a SCTP. Využíva adresovanie pomocou SIP URI.

Medzi základné typy SIP správ patria: REGISTER, INVITE, ACK, CANCEL, OPTIONS, BYE. IMS architektúra pridala alebo upravila niektoré SIP žiadosti a odpovede, aby lepšie vyhovovali účelom IMS siete, resp. zastrešovali nové funkcie. Medzi konkrétne správy patria [4][7]:

- INFO – pôvodne slúžila na doručenie doplňujúcich informácií o spojení; IMS pomocou INFO správy prenáša PSTN signalizáciu.
- REFER – informuje koncové zariadenie, aby kontaktovalo prvok s URI alebo URL adresou prenášanou v tele správy.
- UPDATE – modifikuje parametre spojenia predtým, ako bola prijatá konečná odpoveď na INVITE správu.
- SUBSCRIBE – prihlasuje užívateľa k definovanej službe a žiada informácie, ktoré ju popisujú.
- NOTIFY – oboznamuje užívateľa o udalosti, ktorá nastala.
- PRACK – potvrdzuje prijatie predbežnej odpovede.
- PUBLISH – nahráva informácie na server.
- MESSAGE – obsahuje textovú správu Instant Message.

1.4.2 DIAMETER

DIAMETER je AAA protokol aplikačnej vrstvy, definovaný v RFC 6733, a vychádza z protokolu RADIUS (Remote Authentification Dial In User Service). Základnú funkcionalitu rozširujú DIAMETER aplikácie. Charakteristické funkcie sú:

- posielanie správ obsahujucich AVP (Attribute-value pair)
- poskytovanie AAA služieb
- účtovanie služieb
- upozornenia chybovými hláškami
- podpora nových príkazov, aplikácií a AVP

DIAMETER využíva model peer-to-peer a komunikáciu žiadosť-odpoveď. Na rozdiel

od protokolu RADIUS nevyužíva UDP, ale protokoly TCP a SCTP. Môze byť zabezpečený pomocou IPsec (Internet Protocol Security) alebo TLS (Transport Layer Security). [3][10]

1.4.3 Session Description Protocol (SDP)

SDP je definovaný v RFC 4566. Poskytuje jednoduchú reprezentáciu parametrov relácie jednotlivým účastníkom, ale nezaoberá sa spôsobom prenosu informácií alebo definovaním parametrov spojenia. Jedná sa o čisto informatívny protokol, neobsahuje transportnú časť, preto môže byť zahrnutý v celej škále sieťových architektúr. Využíva protokoly SIP, RTSP (Real Time Streaming Protocol) a HTTP. SDP môže obsahovať tieto informácie:

- názov a účel relácie
- čas, určujúci ako dlho je relácia aktívna
- použitý kodek
- typ kompresie dát
- adresy a porty koncových zariadení
- použiteľná šírka pásma
- transportný protokol
- typ média (audio, video, text)
- multicastové informácie

Pri použití SIP protokolu sa SDP prenáša v tele SIP správy. SDP parametre majú jednoduchý formát <označenie>= <hodnota>. Konkrétne príklady SDP polí [3][8]:

- v= verzia protokolu
- o= identifikácia relácie a vlastníka relácie
- s= názov relácie
- t= čas, určujúci ako dlho je relácia aktívna
- m= typ média
- c= informácie o pripojení
- k= kryptovací kľúč
- a= ďalšie atribúty

1.4.4 Real-time Transport Protocol (RTP)

RTP bol vytvorený skupinou IETF a je definovaný v RFC 3550. RTP zabezpečuje end-to-end prenos dát v reálnom čase, najčastejšie prenáša audio a video. Zároveň umožňuje identifikáciu typu prenášaných dát, číslovanie sekvencií, používanie časových značiek a sledovanie doručenia dát. Najčastejšie používa UDP protokol, ale

je kompatibilný aj s ostatnými transportnými protokolmi. RTP podporuje multicastové vysielanie dát. Zabezpečená verzia je SRTP (Secure Real-time Transport Protocol), definovaná v RFC 3771.

RTP sa používa spoločne s riadiacim protokolom RTCP (Real-time Transport Control Protocol), ktorý zabezpečuje prenos kontrolných dát a riadiacich informácií, vytváranie štatistík spojenia a riadenie QoS. [9]

1.5 Identita IMS

Operátor musí byť schopný presne identifikovať každého užívateľa, aby mohli medzi sebou nadviazať spojenie. Ako sa v telefónnych sieťach používa telefónne číslo, IMS na identifikáciu využíva public user identities a private user identities.

1.5.1 Public User Identities

Každý užívateľ v sieti IMS má priradenú jednu alebo viac public user identities, ktoré mu priradí operátor. Používajú sa na smerovanie SIP správ a môžu mať tvar SIP URI alebo TEL (Telephone) URI.

Tvar SIP URI má formát: `sip:first.last@operator.com`. Operátor môže meniť formát podľa svojich potrieb, napríklad použitie telefónneho čísla v SIP URI má formát: `sip:+1-222-333-0293@operator.com;user=phone`. Táto možnosť je často využívaná, pretože protokol SIP pri registrácii požaduje SIP URI, nie je možné použiť TEL URI.

TEL URI je druhý tvar public UI (User Identity): `tel:+1-222-333-0293`. TEL URI je potrebná pri spojení zo siete IMS do siete PSTN. Platí to aj opačne, pretože PSTN užívateľ môže vytočiť iba čísla.

Je bežné, že operátor priradí užívatelovi aspoň jednu SIP URI a jednu TEL URI. Dôvody pre používanie viac než jednej public UI sú napríklad využívanie rôznych typov služieb alebo oddelenie pracovnej sféry od súkromnej. IMS umožňuje v SIP správe REGISTER registrovať väčšie množstvo public UI, čím sa šetrí čas a prostriedky linky. [3]

1.5.2 Private User Identities

Každý užívateľ má priradenú private UI. Narozdiel od public UI nemá formát SIP alebo TEL URI, ale NAI (Network Access Identifier): `username@operator.com`.

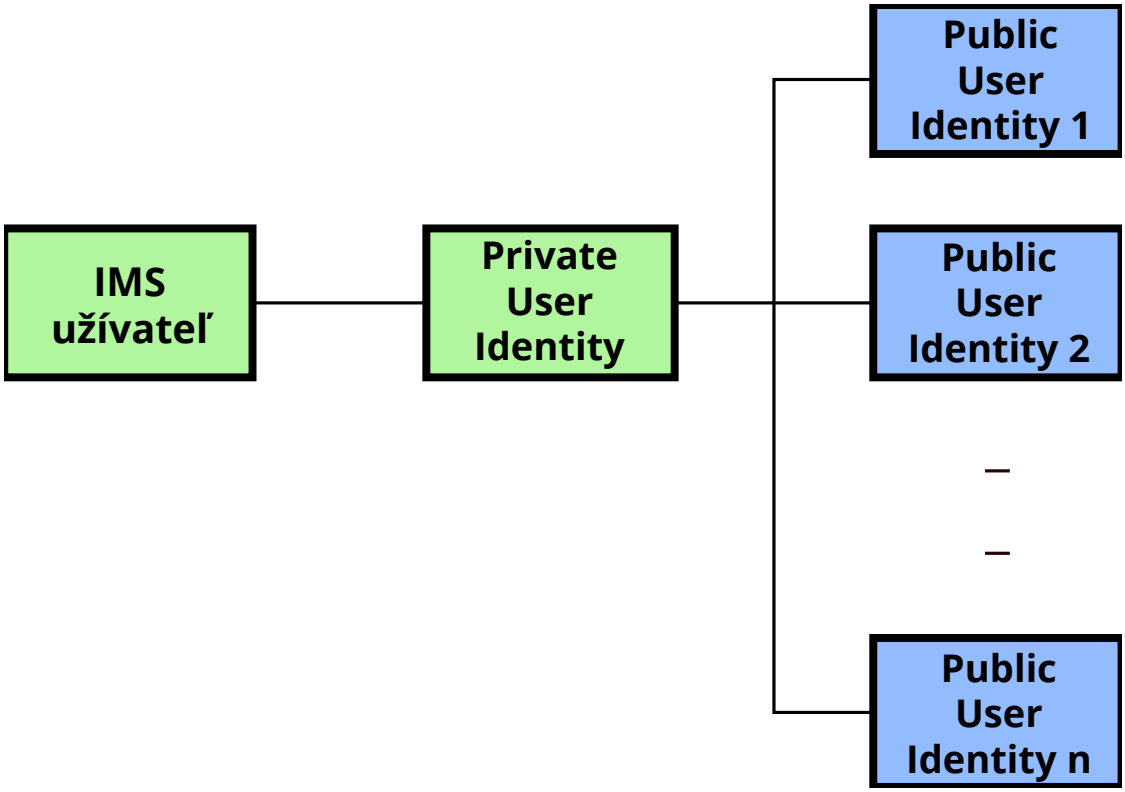
Private UI sa nepoužíva na preposielanie SIP správ, ale výhradne na identifikáciu a autentizáciu užívateľov. Private UI je uložená v zariadení alebo čipovej karte, preto užívateľ nemusí vedieť jej presný tvar. [3]

1.5.3 Vzťah medzi private UI a public UI

HSS ako hlavná databáza ukladá private UI a všetky public UI priradené užívateľovi. Spolu s S-CSCF mapuje private a public UI medzi sebou.

3GPP vydanie 5

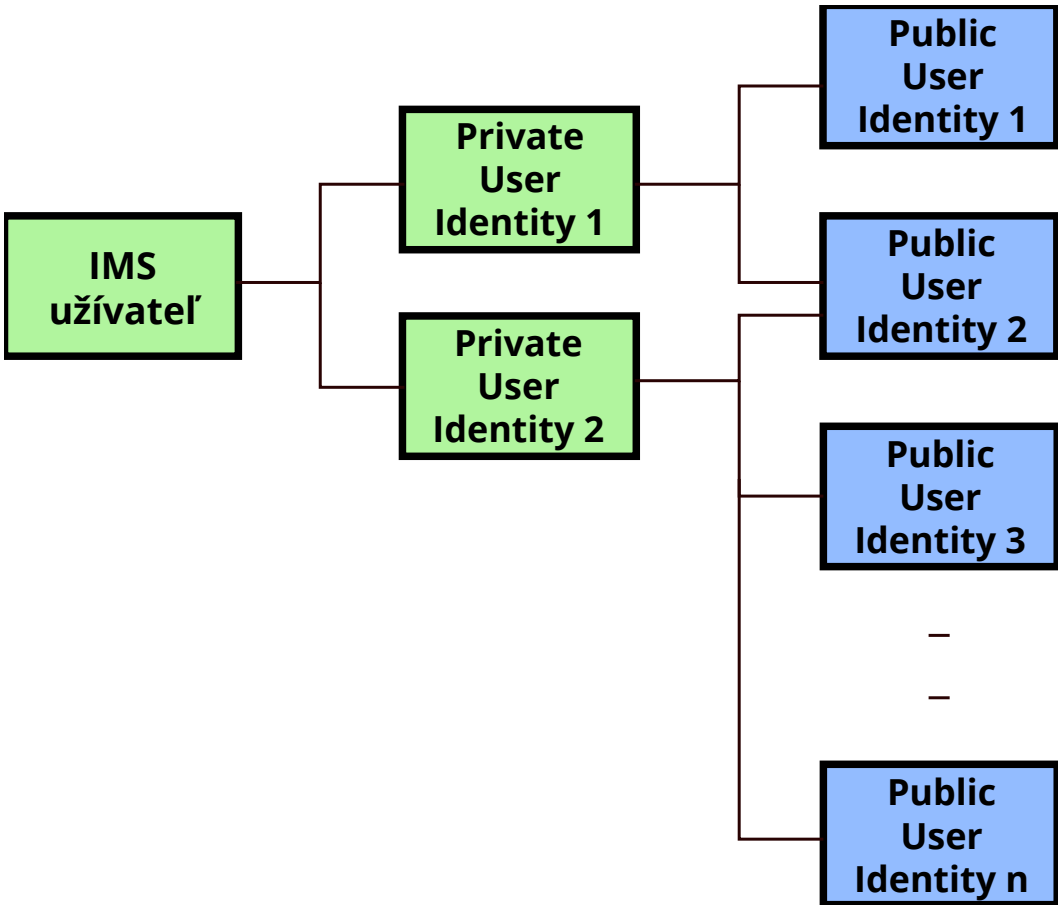
IMS užívateľovi je priradená jedna private UI a niekoľko public UI, to bolo štandardizované v 3GPP vydaní 5. [3]



Obr. 1.4: Vzťah medzi private UI a public UI v 3GPP R5 [3]

3GPP vydanie 6

3GPP vydanie 6 bolo rozšírené o možnosť prideliť užívateľovi viac než jednu private UI. Zároveň je možné linkovať jednu public UI s viacerými private UI, ale jedno koncové zariadenie stále pracuje iba s jednou private UI. To umožňuje v rovnakom čase používať jednu public UI z dvoch zariadení s rôznymi private UI. [3]



Obr. 1.5: Vzťah medzi private UI a public UI v 3GPP R6 [3]

1.5.4 Public Service Identities (PSI)

Koncept PSI bol uvedený v 3GPP vydaní 6, ide o identifikátory priradené konkrétnym službám na AS. Nie je možné ich prideliť užívateľovi a rovnako ako public UI majú tvar SIP URI alebo TEL URI. [3]

1.6 Domáce a navštívené siete

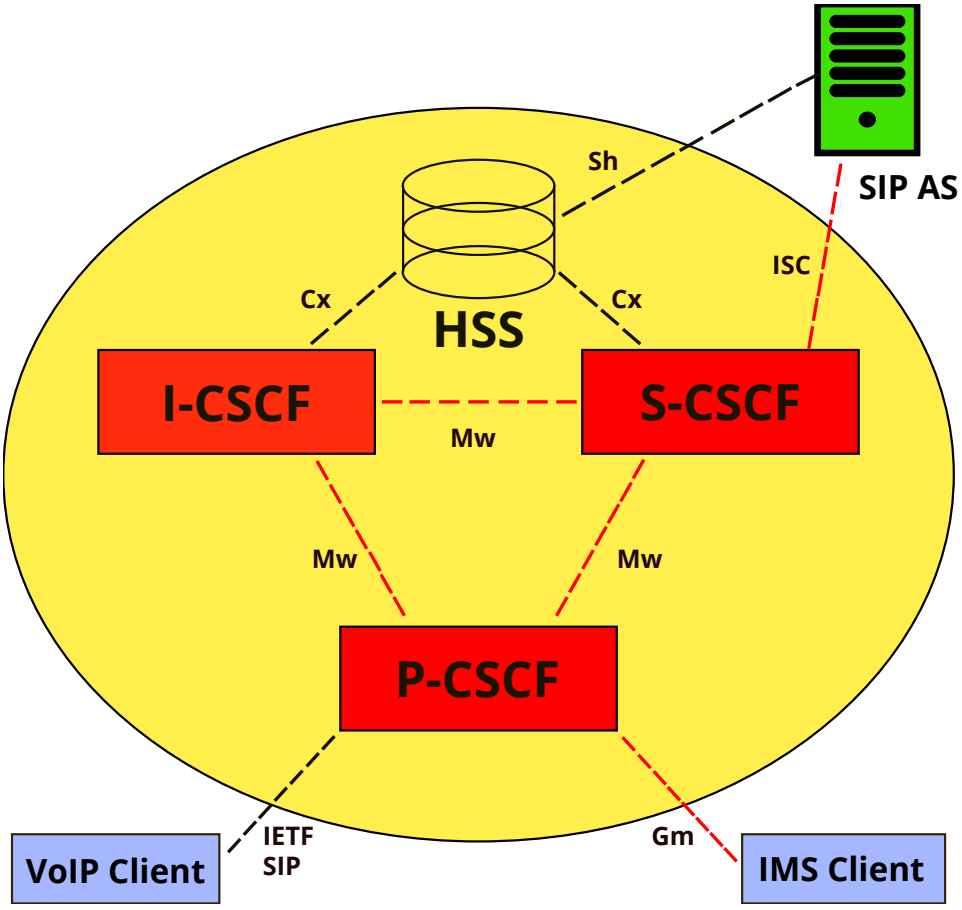
IMS prevzalo koncept domácich a navštívených sietí od GSM a GPRS (General Packet Radio Service). Ak zariadenie využíva infraštruktúru siete poskytovanú operátorom, ktorý danú sieť zároveň spravuje, nachádza sa v domácej sieti. Ak užívateľ prejde za hranicu pokrytia domácej siete, využíva infraštruktúru iného operátora a nachádza sa v navštívenej sieti.

Využívanie navštívenej siete je podmienené uzavretou dohodou medzi operátormi domácej a navštívenej siete. Dohoda ustanovuje parametre poskytovaných služieb ako napríklad ceny služieb, QoS, preposielanie účtovacích informácií.

Väčšina IMS prvkov sa nachádza v domácej sieti, najčastejšou výnimkou je P-CSCF. Keďže nie všetky siete začnú využívať IMS v rovnakom čase, respektíve nebudú implementovať 3GPP vydanie 5 alebo 6, očakáva sa, že P-CSCF bude vo väčšine prípadov lokalizované v domácej sieti. Navštívená sieť bude iba poskytovať telekomunikačné spojenie s IMS systémom, bez podpory konkrétnych funkcií. [3]

2 Základná architektúra Open IMS Core

Open IMS Core je open source projekt využívaný na vývoj a testovanie hlavných prvkov IMS architektúry, konkrétne CSCF a HSS. Projekt bol spustený v roku 2006.



Obr. 2.1: Základná architektúra Open IMS Core [11]

Open IMS Core obsahuje základné prvky IMS siete, tj. HSS, P-CSCF, I-CSCF a S-CSCF.

2.1 HSS

Jadrom HSS je HSSDiameterStac, ktorý pomocou DiameterPeer odosiela požiadavky. Odpovede prijíma pomocou CommandListener. HSS dáta sú uložené v databáze. FHoSS implementuje rozhrania:

- Sh – tvorí spoj medzi AS a HSS.
- Zh – komunikuje s BSF (Bootstrapping Server Function).
- Cx – využívané na komunikáciu s I-CSCF a S-CSCF.

FHoSS sa spravuje pomocou webového rozhrania, ktoré využíva Java Servlet a Apache Struts.

2.2 MySQL databáza

FHoSS obsahuje dva SQL skripty, potrebné pre správnu funkčnosť:

- `hss_db.sql` – vytvára tabuľky a databázu.
- `userdata.sql` – vytvára užívateľov a ich servisné profily.

2.3 CSCF

CSCF obsahuje jednotlivé moduly:

- Proxy-CSCF
- Interrogating-CSCF
- Serving-CSCF
- CDiameterPeer
- IMS Service Control

2.3.1 Proxy-CSCF

Modul implementuje funkcie P-CSCF servera, konfiguráciu obsahuje súbor `pcscf.conf`.

2.3.2 Interrogating-CSCF

Poskytuje služby I-CSCF servera, komunikuje s HSS cez rozhranie Cx pomocou protokolu DIAMETER. Pre správnu funkčnosť musí byť načítaný modul CDiameterPeer a databáza musí obsahovať tabuľky definované v súbore `icscf.conf`. Implementuje funkciu THIG, I-CSCF je konfigurovateľné cez súbor `icscf.conf`.

2.3.3 Serving-CSCF

Obsahuje funkcie S-CSCF servera, konfigurácia je uložená v súbore `scscf.conf`. Vyžaduje načítaný modul CDiameterPeer, ktorý komunikuje cez rozhranie Cx s HSS pomocou protokolu DIAMETER .

2.3.4 CDiameterPeer

Zabezpečuje komunikáciu zo strany CSCF prvkov (I-CSCF, S-CSCF) cez protokol DIAMETER. Každý server má vlastný konfiguračný súbor, ktorý je vo formáte xml.

2.3.5 IMS Service Control

Modul poskytuje podporu rozhrania ISC, cez ktoré komunikuje S-CSCF s AS. Vyžaduje načítaný modul Serving-CSCF.

2.4 Možnosti konfigurácie HSS

Nastavenie je možné meniť cez súbory:

- `DiameterPeerHSS.xml` – upravuje nastavenie DIAMETER HSS peerov.
- `hibernate.properties` – určuje parametre frameworku Hibernate.
- `hss.properties` – obsahuje nastavenie HSS.
- `log4j.properties` – nastavuje vlastnosti loggeru.

2.5 Možnosti konfigurácie CSCF

Na konfiguráciu využívame súbory:

- `icscf.xml`, `scscf.xml` – upravujú nastavenie DIAMETER CSCF peerov.
- `icscf.cfg`, `scscf.cfg`, `pcscf.cfg` – obsahujú konfiguráciu I-CSCF, S-CSCF a P-CSCF.

Konfigurácia je defaultne spustená na adrese localhost (127.0.0.1). [11]

3 Doplnkové služby

3.1 Emergency-CSCF (E-CSCF)

Jednou z požiadaviek na celulárne siete po celom svete je možnosť zavolať na núdzové čísla. Podmienky sa líšia podľa krajiny, niektoré vyžadujú schopnosť volať na núdzové čísla bez SIM karty, resp. bez užívateľského statusu v danej sieti, iné zase vyžadujú presnú lokalizáciu volajúceho.

IMS systém v prvej fázi narazil na viaceré problémy, napríklad neschopnosť niektorých prístupových technológií povoliť prístup od neregistrovaného zariadenia alebo problém pri roamingu užívateľa do krajiny s odlišným núdzovým číslom, P-CSCF v domácej sieti nedokázal číslo správne vyhodnotiť.

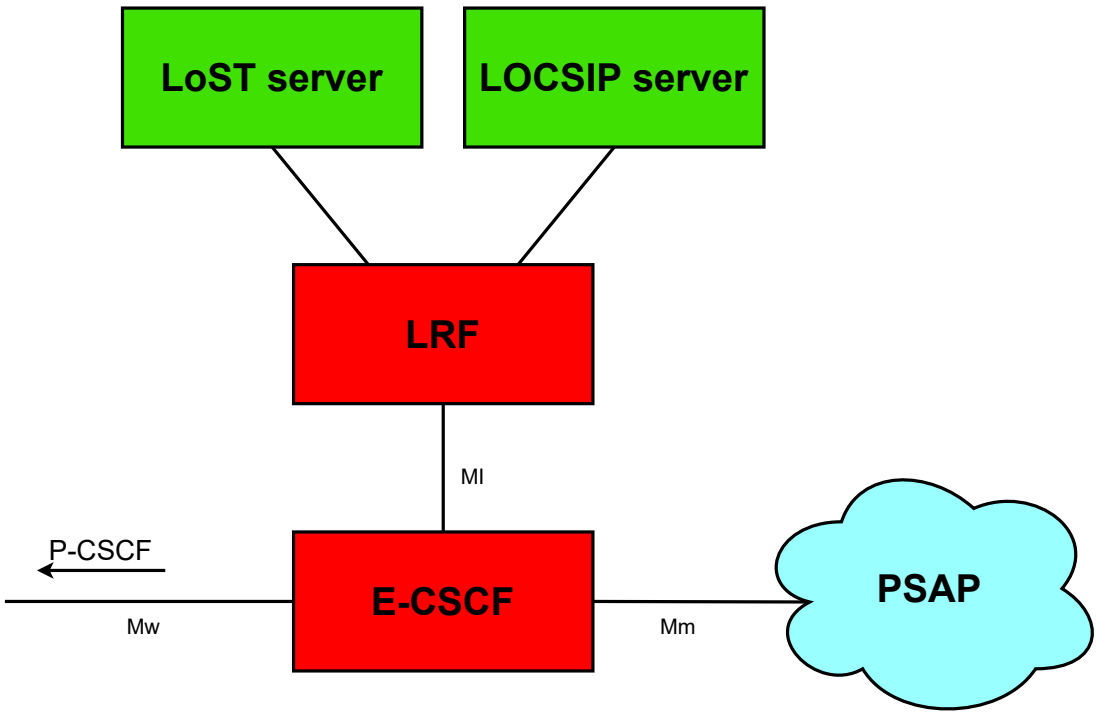
Prvotným riešením bolo využitie domény s prepínaním okruhov, ktorá už núdzové volania podporovala. IMS koncové zariadenie dokázalo rozlíšiť núdzové číslo a hovor smerovalo na doménu s prepínaním okruhov namiesto do IMS siete. Pri roamingu, resp. nevyhodnotení čísla ako núdzového, terminál smeruje hovor do IMS siete, ale P-CSCF s nastaveným zoznamom núdzových čísel rôznych krajín rozpozná núdzové číslo a posiela odpoveď s inštrukciami na presmerovanie do siete s prepínaním okruhov.

Pokročilejšie riešenie vzniklo pri predstavení nového prvku Emergency-CSCF. E-CSCF je hlavným bodom celého procesu pri vytáčaní núdzového čísla. Medzi hlavné úlohy, ktoré zabezpečuje patrí:

- prijímanie núdzového spojenia od P-CSCF a S-CSCF.
- získavanie informácií o polohe zariadenia od LRF (Location Retrieval Function) v prípade, že ich neobsahuje žiadosť o núdzové spojenie.
- overenie polohy koncového zariadenia pomocou LRF.
- získavanie informácií potrebných na smerovanie od LRF.
- smerovanie žiadosti o núdzové spojenie na BGCF alebo PSAP (Public Safety Answering Point).
- vygenerovanie čísla pre koncové zariadenie, ktoré nemá žiadnu identitu, napríklad neobsahuje SIM kartu.
- ukladanie informácií o koncovom zariadení, ktoré žiada o spojenie (iba ak je to vyžadované zákonom danej krajiny).

Ďalšími kľúčovými entitami núdzovej architektúry sú LFR, LOCSIP (Location in SIP/IP) server a LoST (Location to Service Translation Protocol) server. Architektúra je zobrazená na obrázku 3.1. LRF poskytuje adresu PSAP a polohu užívateľa, ak nie je uvedená v INVITE správe. LOCSIP slúži na zistenie polohy užívateľa na základe názvu domény, polohu posiela prvku LRF. LoST mapuje adresy

PSAP ku geografickým lokalitám, pri procese núdzového volania určí najbližšiu adresu PSAP vzhľadom k polohe užívateľa a odošle ju LRF. V nasledujúcej časti je priblížení priebeh komunikácie.



Obr. 3.1: Emergency architektúra Open IMS Core

Pri vytváraní núdzového hovoru môžu nastať štyri situácie:

- užívateľ je v domácej sieti a vykonal nenúdzovú registráciu; môže priamo vytvoriť núdzový hovor.
- užívateľ nevykonal registráciu; bez ohľadu v akej sieti sa nachádza (domácej alebo navštívenej) musí vykonať núdzovú registráciu a následne môže vytvoriť núdzový hovor.
- užívateľ je v navštívenej sieti a vykonal nenúdzovú registráciu; musí vykonať núdzovú registráciu a následne môže vytvoriť núdzový hovor.
- užívateľ nemá dostatočné údaje, aby sa mohol autentizovať a registrovať; môže vykonať anonymný núdzový hovor.

Pri núdzovej registrácii sa v správe REGISTER v poli Contact pridáva parameter sos, pole vyzerá nasledovne: `<sip:alice@example.com;sos>`. Podľa parametru sos P-CSCF vie rozoznať, že sa jedná o núdzovú komunikáciu. P-CSCF zároveň

rozoznáva núdzové čísla, ktoré má uložené v pamäti, a mapuje ich na SIP URI, napríklad: 112 -> `sip:service:sos.ambulance`. Pred posielaním núdzových správ ďalej do siete najprv zamení núdzové číslo za SIP URI. Núdzové správy sú z P-CSCF smerované priamo na E-CSCF. E-CSCF následne požiada LRF o adresu PSAP, prípadne ak počiatočná správa neobsahuje polohu, tak aj o zistenie polohy. LRF prijme správu s požiadavkami a môžu nastať dva prípady:

- správa od E-CSCF obsahuje polohu; LRF pošle správu s polohou a volanou núdzovou službou na LoST server, ktorý vyberie najvhodnejší PSAP; LoST server následne pošle naspäť adresu PSAP vo formáte SIP URI.
- správa od E-CSCF neobsahuje polohu; LRF posiela správu LOCSIP serveru, ktorý podľa doménového názvu odhadne polohu užívateľa a pošle ju naspäť LRF; LRF následne posiela správu s polohou a volanou službou LoST serveru ako v predchádzajúcom bode.

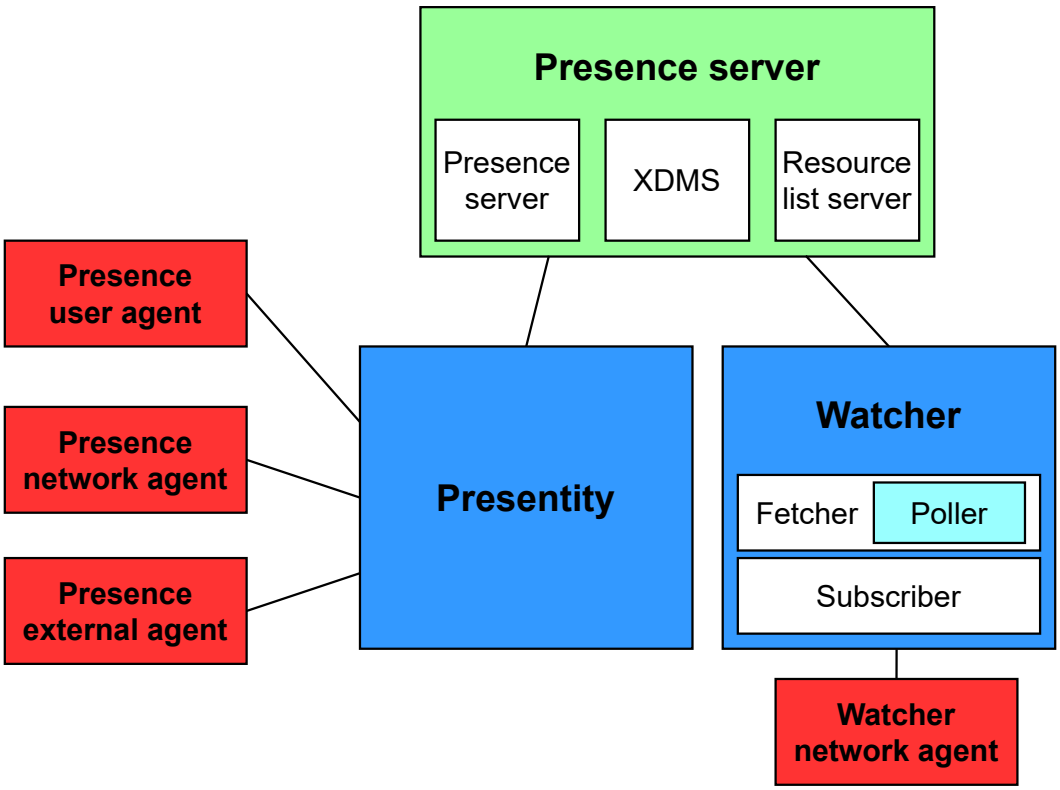
Po získaní potrebných informácií posiela LRF správu s PSAP adresou, prípadne polohou, smerom na E-CSCF. E-CSCF následne posiela správu INVITE na adresu PSAP, správa môže obsahovať doplňujúce informácie ako lokalizáciu na mape alebo čas od prijatia prvej núdzovej správy. V prípade, že niektorá z predošlých operácií zlyhá alebo prebehne bez výsledku, E-CSCF môže mať definovanú takzvanú „poslednú možnosť“ (Last Routing Option), ktorá obsahuje adresu defaultného PSAP, na ktorý E-CSCF bude posielať správy. [3][12]

3.2 IMS presence architektúra

Pri online komunikácii je veľmi dôležité vedieť stav ostatných účastníkov, či sú online, práve s niekým volajú alebo sú neaktívni. Podľa toho sa vieme ďalej rozhodnúť komu a kedy zavoláme. Preto vznikol presence systém, ktorý využívajú viaceré technológie, resp. komunikačné aplikácie ako Skype, Zoom, Microsoft Teams a iné.

IMS systém implementuje presence architektúru, zobrazenú na obrázku 3.2, ktorá je rozdelená do troch vrstiev: presence agenty, presence entity a samotný presence server. Presence agenti získavajú potrebné dáta a sú rozdelení podľa zdrojov informácií:

- Presence user agent – získava informácie z užívateľských zariadení, súčasť UE.
- Presence network agent – komunikuje s prvkami v IMS sieti, napríklad HSS, S-CSCF, AAA server a iné, je súčasťou sieťových prvkov (S-CSCF, AS, HSS...).
- Presence external agent – získava informácie zo zariadení, ktoré sa nachádzajú v iných sieťach.
- Watcher user agent – poskytuje informácie pre watcher a užívateľské zariadenia, súčasť UE.



Obr. 3.2: Presence architektúra [13]

Entity sú špecifické tým, že dokážu pracovať so SIP správami. Delia sa na dva typy, presentity (presence entity) poskytuje informácie získané agentmi a watcher sleduje stav užívateľských zariadení. Watcher je rozdelený na tri časti:

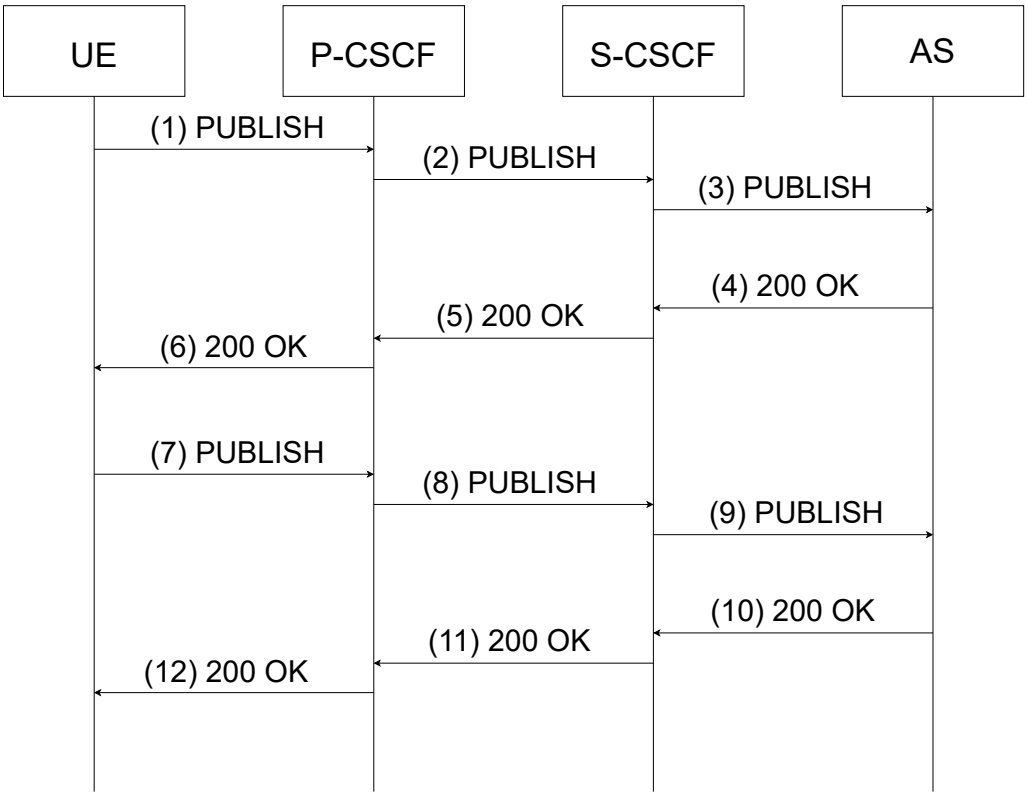
- Fetcher – pozoruje stav zariadení len v prítomnom čase.
- Poller – pozoruje stav zariadení v predom definovaných intervaloch, je to špeciálny druh fetcheru.
- Subscriber – sleduje zmeny stavu zariadení.

Presence server zbiera, spracováva a poskytuje užívateľské informácie, napríklad stav, schopnosti zariadenia, profilová fotka alebo avatar, popis užívateľa, geolokáciu, časovú známku a iné. Informácie sú uložené v XML súboroch.

Resource list server vytvára zoznamy užívateľov, ktoré má watcher sledovať. Tým eliminuje problém, kedy by bola kvôli každému užívateľovi vysielaná samostatná požiadavka.

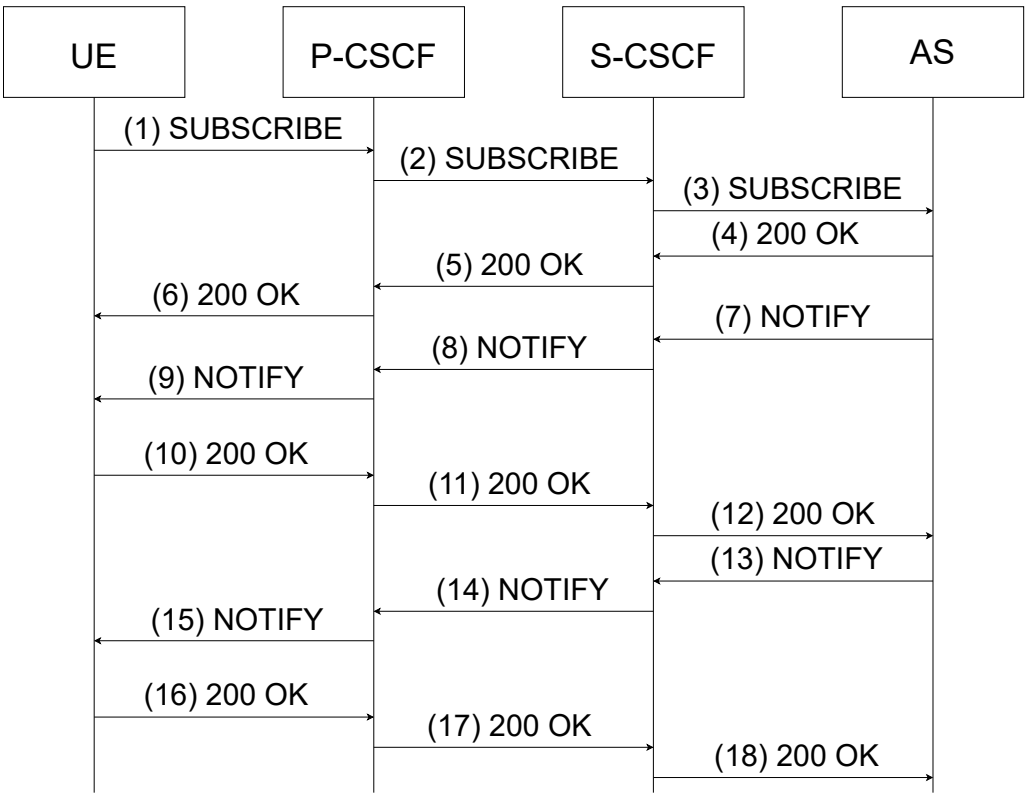
XDMS (XML Data Management Server) uchováva XML zoznamy. Užívatelia tak môžu meniť koho chcú sledovať alebo meniť oprávnenie kto môže o nich získavať informácie. Zároveň watcher môže obdržať link priamo do XDMS databáze, kde sa nachádzajú informácie, ktoré požaduje.

Komunikácia pozostáva z dvoch procesov, zverejnenia informácií (publishing) a požiadania o informácie (subscribing). Proces zverejnenia je zobrazený na obrázku 3.3. Predpokladáme, že zariadenie je zaregistrované v IMS sieti. UE posiela informácie v SIP správe PUBLISH cez P-CSCF a S-CSCF na presence AS, server odpovedá čo najskôr správou 200 OK. Ak UE mení svoj status, posiela novú správu PUBLISH, ktorá obsahuje len informácie, ktoré sa zmenili.



Obr. 3.3: Proces zverejnenia informácií (publishing) [13]

Proces požiadania je zobrazený na obrázku 3.4. UE, ktorý vystupuje ako watcher, vytvorí SIP požiadavku SUBSCRIBE, ktorá obsahuje zoznam vyžadovaných informácií. Správa je cez P-CSCF a S-CSCF poslaná na presence AS. Server overí identitu a oprávnenia UE a v prípade zhody odošle odpoveď 200 OK. Následne server posiela správu NOTIFY, ktorá obsahuje vyžadované dáta. Pri zmene statusu zariadenia, ktoré sleduje watcher UE, server posiela ďalšiu správu NOTIFY bez nutnosti prijatia novej požiadavky. [13]



Obr. 3.4: Proces požiadania o informácie (subscribing) [13]

4 Implementácia IMS architektúry pomocou Open IMS Core

Jedným z cieľov bakalárskej práce je implementácia základných prvkov IMS systému pomocou prostredia Open IMS Core. Ako základ bola použitá architektúra zobrazená na obrázku 2.1, ktorá obsahuje všetky potrebné entity CSCF a HSS. Inštaláciu Open IMS Core je možné vykonať priamo na systéme Linux (doporučuje sa verzia Ubuntu) z SVN úložiska alebo je možné stiahnuť, z oficiálnych stránok Open IMS Core, virtuálny obraz Ubuntu, ktorý obsahuje všetky potrebné súbory pre funkčnosť Open IMS Core.

Keďže používame OS Windows, konkrétne verziu 10, rozhodli sme sa využiť možnosť virtuálneho obrazu Ubuntu, s ktorým pracujeme pomocou softvéru VMware Workstation 12 Player.

4.1 Konfigurácia localhostu

Po stiahnutí a spustení virtuálneho obrazu je nutné dokončiť konfiguráciu niektorých prvkov. Defaultná doména je nastavená na adrese `open-ims.test`. DNS server bol nastavený, aby používal localhost adresu 127.0.0.1 a názov defaultnej domény, ako je vidieť z výpisu 4.1 a 4.2. Súbor `named.conf` sa nachádza v adresári `/etc/bind`, `resolv.conf` je v adresári `/etc`.

```
zone "open-ims.test" IN{
    type master;
    file "/etc/bind/open-ims.dnszone";
};
```

Výpis 4.1: Nastavenie DNS v súbore `named.conf`

```
domain open-ims.test
search open-ims.test
nameserver 127.0.0.1
```

Výpis 4.2: Nastavenie DNS resolveru v súbore `resolv.conf`

Správnosť nastavenia DNS a pridanie nového záznamu bolo overené pomocou pingu na adresu `open-ims.test` a `pcscf.open-ims.test`.

```
root@ubuntu-vm:~# ping open-ims.test
64 bytes from localhost (127.0.0.1): ttl=64 time=0.014 ms
64 bytes from localhost (127.0.0.1): ttl=64 time=0.028 ms
                          -
root@ubuntu-vm:~# ping pcscf.open-ims.test
64 bytes from localhost (127.0.0.1): ttl=64 time=0.010 ms
64 bytes from localhost (127.0.0.1): ttl=64 time=0.031 ms
                          -
```

Výpis 4.3: Ping na adresu `open-ims.test` a `pcscf.open-ims.test`

Rovnako je nutné inicializovať MySQL databázu, ktorá beží na localhost adrese. Prístup je umožnený cez užívateľa root, ktorý má administrátorské oprávnenia, a bez hesla. Inicializácia je zobrazená na výpise 4.4, využité sú súbory `icscf.sql`, `hss_db.sql` a `userdata.sql`.

```
mysql -u root -p -h localhost < ser_ims/cfg/icscf.sql
mysql -u root -p -h localhost < FHoSS/scripts/hss_db.sql
mysql -u root -p -h localhost < FHoSS/scripts/userdata.sql
```

Výpis 4.4: Spustenie SQL skriptov

Zároveň je potrebné definovať cestu k adresáru, v ktorom sa nachádza balíček Java. Ten je nevyhnutný pre spustenie súčastí systému ako napríklad Tomcat server. Virtuálny obraz má nainštalované viaceré verzie, ale využívaná je verzia Java SE 1.6.0.14.

```
export JAVA_HOME=/usr/lib/jvm/java-6-sun-1.6.0.14
export PATH=$PATH:$JAVA_HOME/lib
```

Výpis 4.5: Definovanie premenných `JAVA_HOME` a `PATH`

Spustenie CSCF pvrkov pomocou skriptov prebehlo v poriadku a bolo vidieť pravidelné logovacie správy, pri spustení HSS bola vypísaná chybová hláška, ktorá informovala o obsadenom porte 8080, na ktorom sa mal defaultne spustiť Tomcat server. Kvôli rovnkaej chybe bolo potrebné zmeniť aj port 3868, na ktorom komunikuje HSS pomocou protokolu DIAMETER. Voľné použité čísla portov sú 8008 a 3871. Zmeny je vidieť na nasledujúcich výpisoch:

```
host=127.0.0.1
port=8008
```

Výpis 4.6: Zmena portu 8080 v súbore `hss.properties`

```
<Acceptor port="3871" bind="127.0.0.1"/>
```

Výpis 4.7: Zmena portu 3868 v súbore `DiameterPeerHSS.xml`

```
<Peer FQDN="hss.open-ims.test" Realm="open-ims.test"
                          port="3871"/>
```

Výpis 4.8: Zmena portu 3868 v súbore `scscf.xml`

```
<Peer FQDN="hss.open-ims.test" Realm="open-ims.test"
                          port="3871"/>
```

Výpis 4.9: Zmena portu 3868 v súbore `icscf.xml`

Pomocou software Monster verzia 0.9.8 bolo uskutočnené pripojenie defaultne nakonfigurovaných užívateľov Alice a Boba do systému Open IMS Core. Následne bol úspešne vykonaný hovor medzi Alicou a Bobom, ktorý sme zachytili pomocou Wiresharku, ako vidieť na obrázku 4.1.

```
------------------ Registrar Contents begin ------------------
[3] P: <sip:alice@open-ims.test> R[1] Early-IMS: <> Barred: []
     CCF1: <pri_ccf_address> CCF2: <>
     C: <sip:alice@127.0.0.1:5062> Exp: [3514] SOS: []
        Path: <sip:term@pcscf.open-ims.test:4060;lr>
        UA: <Fokus MONSTER Version: 0.9.8-SNAPSHOT>
[52] P: <sip:bob@open-ims.test> R[1] Early-IMS: <> Barred: []
     CCF1: <pri_ccf_address> CCF2: <>
     C: <sip:bob@127.0.0.1:5064> Exp: [3593] SOS: []
        Path: <sip:term@pcscf.open-ims.test:4060;lr>
        UA: <Fokus MONSTER Version: 0.9.8-SNAPSHOT>
     S: Event [0] Exp: [1828] <sip:pcscf.open-ims.test:4060>
------------------ Registrar Contents ends ------------------
```

Výpis 4.10: Výpis registrar obsahu na S-CSCF

Obr. 4.1: Nadviazanie a ukončenie hovoru medzi Alicou a Bobom

4.2 Konfigurácia na IP adrese 192.168.20.179

Ďalším krokom je konfigurácia pomocou novej DNS domény a IP adresy. Názov domény je `oims` a IP adresa je 192.168.20.179, adresa bola staticky priradená z DHCP (Dynamic Host Configuration Protocol) serveru na MAC adresu virtuálneho PC. Zmeny konfigurácie DNS serveru boli vykonané v súboroch:

- `resolv.conf` – výpis 4.11
- `named.conf` – príloha A.1
- `named.conf.options` – príloha A.2
- `open-ims.dnszone` – príloha A.3

```
domain oims
search oims
nameserver 192.168.20.179
```

Výpis 4.11: Nastavenie DNS resolveru v súbore `resolv.conf`

Zároveň musela byť upravená konfigurácia HSS a všetkých CSCF prvkov na novú doménu a adresu. Zmeny HSS boli vykonané v konfiguračných súboroch:

- `DiameterPeerHSS.xml` – výpis 4.12
- `hss.properties` – príloha A.4
- `userdata.sql` – príloha A.5

```
FQDN="hss.oims"
Realm="oims"
                          -
<Peer FQDN="icscf.oims" Realm="oims" port="3869"/>
<Peer FQDN="scscf.oims" Realm="oims" port="3870"/>
<Acceptor port ="3871" bind="192.168.20.179"/>
```

Výpis 4.12: Zmeny v súbore `DiameterPeerHSS.xml`

Na konfiguráciu CSCF celku boli použité konfiguračné súbory jednotlivých prvkov:

- `pcscf.cfg` – príloha A.6
- `pcscf.xml` – výpis 4.13
- `scscf.cfg` – príloha A.7
- `scscf.xml` – výpis 4.14
- `icscf.cfg` – príloha A.8
- `icscf.xml` – výpis 4.15
- `icscf.sql` – príloha A.9

```
FQDN="pcscf.oims"
Realm="oims"
                          -
<Peer FQDN="clf.oims" Realm="oims" port="3868"/>
<Acceptor port ="3867" bind="192.168.20.179"/>
<DefaultRoute FQDN="clf.oims" metric="10"/>
```

Výpis 4.13: Konfigurácia P-CSCF v súbore `pcscf.xml`

```
FQDN="scscf.oims"
Realm="oims"
                          -
<Peer FQDN="hss.oims" Realm="oims" port="3871"/>
<Acceptor port ="3870" bind="192.168.20.179"/>
                          -
<DefaultRoute FQDN="hss.oims" metric="10"/>
```

Výpis 4.14: Konfigurácia S-CSCF v súbore `scscf.xml`

```
FQDN="icscf.oims"
Realm="oims"
                          -
<Peer FQDN="hss.oims" Realm="oims" port="3871"/>
<Acceptor port ="3869" bind="192.168.20.179"/>
<DefaultRoute FQDN="hss.oims" metric="10"/>
```

Výpis 4.15: Konfigurácia I-CSCF v súbore `icscf.xml`

Nastavenie DNS domény bolo overené pomocou pingu na adresy `oims` a `pcscf.oims`.

```
root@ubuntu-vm:~# ping oims
64 bytes from oims (192.168.20.179): ttl=64 time=0.011 ms
64 bytes from oims (192.168.20.179): ttl=64 time=0.025 ms
                            -
root@ubuntu-vm:~# ping pcscf.oims
64 bytes from oims (192.168.20.179): ttl=64 time=0.008 ms
64 bytes from oims (192.168.20.179): ttl=64 time=0.037 ms
                            -
C:\Users\testcenter> ping oims
Pinging oims [192.168.20.179] with 32 bytes of data:
Reply from 192.168.20.179: bytes=32 time<1ms TTL=64
```

Výpis 4.16: Ping na adresu `oims` a `pcscf.oims`

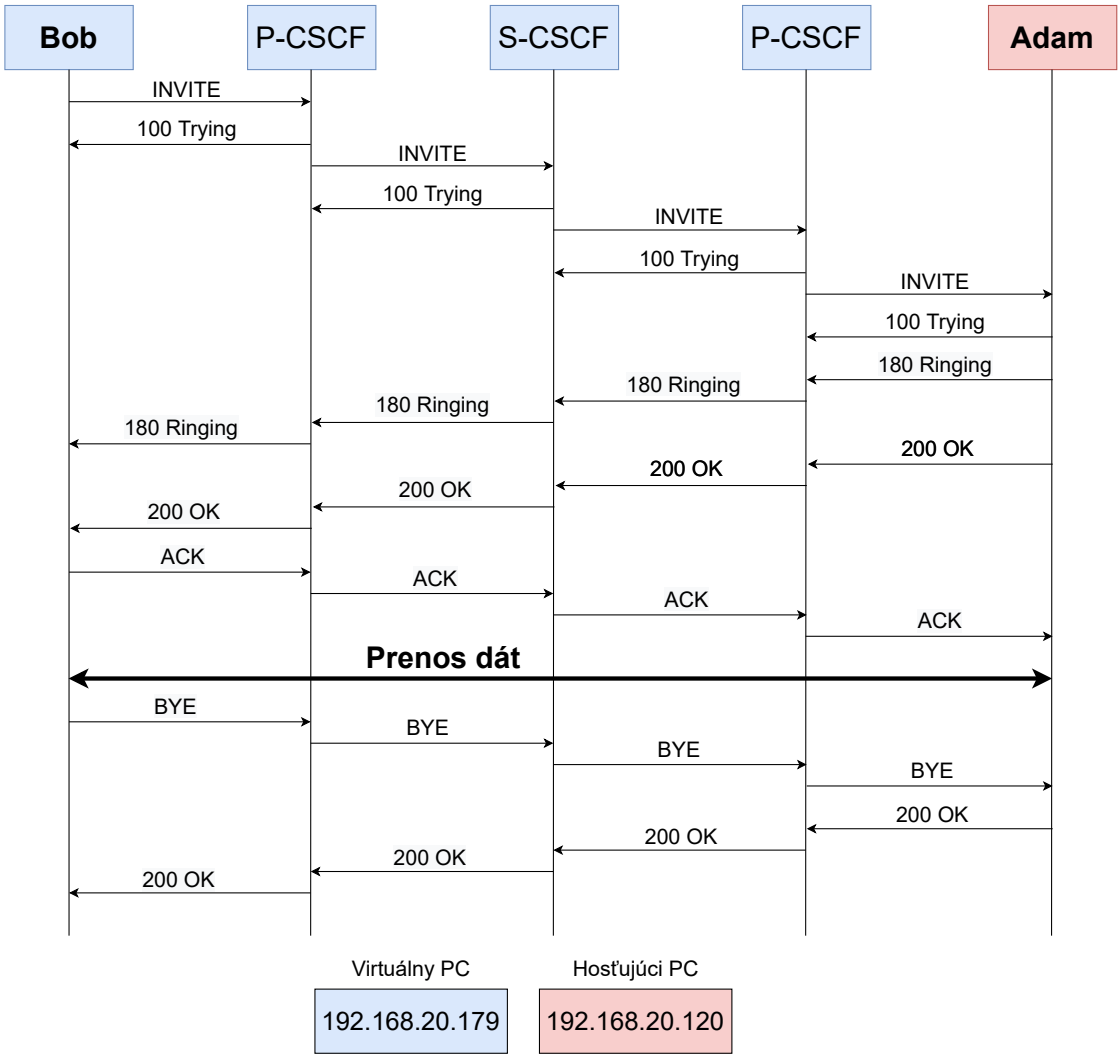
Zároveň bol z hosťujúceho PC (IP 192.168.20.120) zaregistrovaný nový užívateľ s menom Adam. Najprv boli v HSS databáze vytvorené a prelinkované public a private UI záznamy, následne bolo v súbore `scscf.cfg` nutné zmeniť autentizačný algoritmus z módu AKAv1-MD5 na mód MD5, pretože komunikačný klient Zoiper, ktorý je využívaný na hosťujúcom PC, nepodporuje starší mód AKAv1-MD5. Na virtuálnom PC bol do IMS siete registrovaný užívateľ Bob, registrovaní užívatelia sú zobrazení na výpise 4.17.

```
------------------ Registrar Contents begin ------------------
[23] P: <sip:adam@oims> R[1] Early-IMS: <> Barred: []
     CCF1: <pri_ccf_address> CCF2: <>
     C: <sip:adam@192.168.20.120:5060;rinstance=0fae01ac90
                    56953f;transport=UDP> Exp: [1229] SOS: []
        Path: <sip:term@pcscf.oims:4060;lr>
        UA: <Z 3.9.32144 r32121>
     S: Event [0] Exp: [1262] <sip:pcscf.oims:4060>
[54] P: <sip:bob@oims> R[1] Early-IMS: <> Barred: []
     CCF1: <pri_ccf_address> CCF2: <>
     C: <sip:bob@192.168.20.179:5064> Exp: [2762] SOS: []
        Path: <sip:term@pcscf.oims:4060;lr>
        UA: <Fokus MONSTER Version: 0.9.8-SNAPSHOT>
     S: Event [0] Exp: [2793] <sip:pcscf.oims:4060>
------------------ Registrar Contents ends ------------------
```

Výpis 4.17: Výpis registrar obsahu s novým užívateľom Adam

Testovanie dostupnosti IMS siete mimo virtuálny PC bolo otestované vytvorením hovoru medzi Bobom a Adamom, resp. virtuálnym PC (IP 192.168.20.179) a hosťujúcim PC (IP 192.168.20.120). Hovor bol zachytený pomocou Wiresharku a je zobrazený na obrázku 4.2. Prehľadnejší flow diagram, vytvorený na základe zachytenej komunikácie, je na obrázku 4.3. Dva prvky P-CSCF na obrázku 4.3 sú jedna reálna entita P-CSCF, ale kvôli prehľadnosti diagramu bolo použité dané zobrazenie.

Obr. 4.2: Nadviazanie a ukončenie hovoru medzi Bobom a Adamom



Obr. 4.3: Flow diagram hovoru medzi Bobom a Adamom

4.3 Konfigurácia E-CSCF

Ako prvé bolo nutné stiahnuť chýbajúce prvky, ktoré pripravená VM neobsahuje. VM používa Ubuntu 9.04, ktoré je momentálne zastaralé, preto sťahovanie zlyhalo z dôvodu nesplnených bezpečnostných požiadaviek, server odmietol pripojenie. Pokus o aktualizáciu SVN modulu, cez ktorý bolo inicializované sťahovanie, na zabezpečenejšiu verziu takisto zlyhal, pretože pre Ubuntu 9.04 sa aktualizácie nevydávajú už približne desať rokov a oficiálne servery už neexistujú.

Jediná možnosť bola stiahnuť novšiu verziu Ubuntu, stiahnuť Open IMS Core a vykonať celú konfiguráciu odznova. Zvolená bola verzia Ubuntu 12.04, stiahnutá vo formáte VMDK (Virtual Machine Disk) a použitá rovnako pomocou virtualizačného programu VMware Workstation 12 Player.

Pred samotnou inštaláciou Open IMS Core musí operačný systém obsahovať všetky prerekvizity:

- GCC (GNU Compiler Collection) kompilátor
- modul make
- modul ant
- JDK (Java Development Kit), verzia minimálne 1.5
- modul bison
- modul flex
- DBMS (Database Management System), napríklad MySQL
- knižnicu libxml2, verzia development minimálne 2.6
- knižnicu libmysql, verzia development
- Linux kernel, verzia minimálne 2.6
- modul ipsec-tools
- modul curl
- knižnicu libcurl4-gnutls-dev
- modul openssl
- bind server alebo iný DNS server

Prerekvizity boli stiahnuté a aktualizované, následne bol vytvorený priečinok s názvom `OpenIMSCore` v priečinku `/opt`. Ďalej priečinky `ser_ims` a `FHoSS` v zložke `OpenIMSCore`. Štruktúru je nutné dodržať, pretože Open IMS Core ju má pevne definovanú. Z SVN úložiska boli stiahnuté súbory CSCF a emergency prvkov do priečinku `ser_ims` a z druhého úložiska súbory HSS a databáz do priečinku `FHoSS`. Ďalším bodom bola samotná inštalácia Open IMS Core, pri ktorej sa objavila chybová hláška:
„Major version 51 is newer than 50, the highest major version supported by this compiler. It is recommended that the compiler be upgraded.“
Po preskúmaní bolo zistené, že JDK v6 sa pokúša kompilovať kód vo formáte JDK

v7. Bol stiahnutý Java v7 balík a museli byť aktualizované premenné `JAVA_HOME` a `PATH`.

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export PATH=$PATH:$JAVA_HOME/lib
```

Výpis 4.18: Definovanie premenných `JAVA_HOME` a `PATH`

Pri opätovnom pokuse nainštalovať Open IMS Core sa zobrazila chybová hláška:
„fatal error: curl/types.h: No such file or directory, compilation terminated.“
Inštalačný priečinok modulu curl neobsahoval súbor types.h, príčina bola v zmene štruktúry modulu curl v priebehu inovovania. Od verzie 7.20 bol súbor types.h vynechaný a jeho funkcionalita presunutá do ostatných častí. Preto bol modul odinštalovaný a nahradený verziou 7.19, ktorá danú požiadavku spĺňa.

Na tretí pokus prebehla inštalácia úspešne a bola zahájená inicializácia databáz. Pri pokuse inicializovať `userdata.sql` konzola vypísala chybu:
„ERROR 1136: Column count doesn't match value count at row 1: at line 41.“
Z výpisu vyplýva, že program sa pokúša zapísať do tabuľky viac údajov ako je možné. Syntax príkazu na riadku 41 bola identická ako pri predpripravenej VM, aj databáza s tabuľkou boli rovnaké. Ako riešenie bol príkaz zakomentovaný, pretože iba vkladá do tabuľky hodnoty aplikačného servera, ktorý pri práci momentálne nepoužívame.

```
-- LOCK TABLES 'application_server' WRITE;
-- INSERT INTO 'application_server' VALUES (1,'default_as'...
-- UNLOCK TABLES;
```

Výpis 4.19: Zakomentované riadky v súbore `userdata.sql`

Po úspešnej inicializácii databáz boli spúšťané jednotlivé prvky. Pri pokuse o štart HSS spúšťací skript nenašiel inštalačný priečinok s Javou. Pomocou kontrolného výpisu vloženého do kódu bolo zistené, že premenná `JAVA_HOME` sa skriptu javí ako prázdna. Riešením bolo implicitné definovanie premennej v kóde skriptu `startup.sh`.

```
#Start-up
#-------------------------------------------------------------
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
$JAVA_HOME/bin/java -cp $CLASSPATH de.fhg.fokus.HSSContainer
```

Výpis 4.20: Implicitná definícia premennej `JAVA_HOME` v súbore `startup.sh`

Po oprave sa bez problémov sputili prvky HSS, S-CSCF a I-CSCF. Pri prvkoch P-CSCF, E-CSCF a LRF sa vyskytol problém pri práci s modulom curl, kedy nedokázali načítať súbor `lib_lost_client.so` kvôli nedefinovanej metóde, ktorú obsahoval. Chyby je vidieť na výpise 4.21.

```
ERROR: load_module: could not open module
</opt/OpenIMSCore/ser_ims/modules/pcscf/pcscf.so>:
/usr/local/lib/ser/lib_lost_client.so:
undefined symbol: curl_easy_perform

ERROR: load_module: could not open module
</opt/OpenIMSCore/ser_ims/modules/ecscf/ecscf.so>:
/usr/local/lib/ser/lib_lost_client.so:
undefined symbol: curl_easy_perform

ERROR: load_module: could not open module
</opt/OpenIMSCore/ser_ims/modules/lrf/lrf.so>:
/usr/local/lib/ser/lib_lost_client.so:
undefined symbol: curl_easy_perform
```

Výpis 4.21: Chyby pri načítaní súboru `lib_lost_client.so`

Boli skontrolované všetky prerekvizity a kontrolný výpis kompilácie kódu CSCF prvkov. Na nasledujúcich výpisoch 4.22 a 4.23 je kontrola modulu curl, knižnice libcurl a cesty smerujúcej na knižnicu libcurl.

```
@osboxes:~$ curl --version
curl 7.19.0 (x86_64-pc-linux-gnu)
Protocols: dict file ftp ftps http https imap imaps
ldap pop3 pop3s rtmp rtsp smtp smtps telnet tftp

@osboxes:~$ curl-config --version
libcurl 7.19.0
```

Výpis 4.22: Kontrola modulu curl a knižnice libcurl

```
@osboxes:~$ curl-config --libs
-L/usr/local/lib/  -L/usr/local/lib/ser -lcurl -Wl,
                -Bsymbolic-funcions -Wl,-z,relro
```

Výpis 4.23: Kontrola prepojenia modulu curl a knižnice libcurl

Samotný súbor `lib_lost_client.so` nie je možné preskúmať, pretože sa jedná o skompilovanú dynamickú knižnicu (shared object) a proces kompilácie sa nedá vrátiť. Rovnaký problém sa objavil aj v mailing listoch, ale vlákno neobsahovalo funkčné riešenie (link: mailinglist). Nájdené riešenia smerovali na nenainštalovaný modul curl alebo knižnicu libcurl a problém s prepojením, tieto možnosti boli vyvrátené na základe predchádzajúcich výpisov.

Posledný pokus bol stiahnuť súbory z SVN úložiska, presunúť ich na predpripravenú VM, skompilovať a otestovať funkčnosť, to by malo odstrániť potenciálne problémy s kompatibilitou novších modulov v Ubuntu 12.04. Predpripravená VM bola pripravená iba na základnú architektúru Open IMS Core, pretože v nej chýbali moduly curl a libcurl, potrebné pre komunikáciu s E-CSCF. Moduly boli stiahnuté v skomprimovanom formáte a následne nainštalované. Inštalácia Open IMS Core prebehla v poriadku, ale pri spustení prvkov P-CSCF, E-CSCF a LRF konzola vypísala presne rovnaké chyby ako pri inštalácii na Ubuntu 12.04. Problém je preto pravdepodobne v kompatibilite úzko zviazanej s konkrétnymi modulmi alebo priamo v stiahnutom kóde, obe možnosti sú používateľom neovplyvniteľné. Na výpisoch 4.24 a 4.25 je zobrazená potenciálna konfigurácia prvkov E-CSCF a LRF, je vidieť nastavenie adresy LOCSIP a LoST serveru alebo nastavenie PSAP SIP URI v rámci „poslednej možnosti“.

```
listen=192.168.20.179
port=7060
alias=ecscf.oims:7060
                              -
loadmodule "/opt/OpenIMSCore/ser_ims/modules/ecscf/ecscf.so"
                              -
modparam("ecscf","name","sip:ecscf.oims:7060")
modparam("ecscf","lrf_sip_uri","sip:lrf.oims:8060")
modparam("ecscf","use_default_psap",1)
modparam("ecscf","default_psap_uri","sip:sos@oims")
modparam("rr", "enable_full_lr", 1)
```

Výpis 4.24: Konfigurácia E-CSCF v súbore `ecscf.cfg`

```
listen=192.168.20.179
port=8060
alias=lrf.oims:8060
                              -
loadmodule "/opt/OpenIMSCore/ser_ims/modules/lrf/lrf.so"
                              -
modparam("lrf","name","sip:lrf.oims:8060")
modparam("lrf","using_lost_srv",1)
modparam("lrf","lost_server",
                 "http://lost@oims:8180/lost/LoSTServlet")
modparam("lrf","enable_locsip", 1)
modparam("lrf","locsip_srv_uri", "sip:locsip@oims:9180")
modparam("rr", "enable_full_lr", 1)
```

Výpis 4.25: Konfigurácia LRF v súbore `lrf.cfg`

4.4 Konfigurácia zabezpečenia IPsec, TLS a THIG

Open IMS Core pracuje defaultne bez zapezečenia, ale ponúka tri možnosti ako chrániť komunikáciu a užívateľské dáta:

- protokol IPsec – slúži na šifrovanie a autentizáciu na tretej vrstve; používa trasnportný alebo tunelový mód a zabezpečenie metódou AH (Authentication Header) alebo ESP (Encapsulating Security Payload).
- protokol TLS – poskytuje zabezpečenie primárne na aplikačnej vrstve; nástupca SSL; bežne používaný v HTTPS.
- THIG (Topology Hiding Inter-network Gateway) – v SIP správe šifruje napríklad polia Via, Route a Record Route; zabraňuje odhaleniu vnútornej topológie siete; najčastejšie sa používa pri posielaní správ mimo domácu sieť.

4.4.1 IPsec zabezpečenie

IPsec zabezpečenie je možné implementovať troma spôsobmi:

- voliteľné pri spojení s P-CSCF
- povinné pri spojení s P-CSCF
- povinné pri spojení s P-CSCF a S-CSCF

Na výpise 4.26 je zobrazená konfigurácia prvku P-CSCF nutná pre podporu IPsec zabezpečenia. Nastavenie bolo vykonané v súbore `pcscf.cfg`. Jednotlivé moduly

riadia vytváranie žiadostí a odpovedí v oboch smeroch, posledný modul riadi zahadzovanie správ. V súbore `pcscf.cfg` sa zároveň určuje, či bude použitý prvý alebo druhý spôsob implementácie IPsec, obe možnosti je vidieť na výpise 4.28.

```
modparam("pcscf","use_ipsec",1)
modparam("pcscf","ipsec_host","192.168.20.179")
modparam("pcscf","use_port_c",4060)
modparam("pcscf","use_port_s",4060)
modparam("pcscf","ipsec_P_Inc_Req","/opt/OpenIMSCore/ser_ims/
modules/pcscf/ipsec_P_Inc_Req.sh")
modparam("pcscf","ipsec_P_Out_Rpl","/opt/OpenIMSCore/ser_ims/
modules/pcscf/ipsec_P_Out_Rpl.sh")
modparam("pcscf","ipsec_P_Out_Req","/opt/OpenIMSCore/ser_ims/
modules/pcscf/ipsec_P_Out_Req.sh")
modparam("pcscf","ipsec_P_Inc_Rpl","/opt/OpenIMSCore/ser_ims/
modules/pcscf/ipsec_P_Inc_Rpl.sh")
modparam("pcscf","ipsec_P_Drop","/opt/OpenIMSCore/ser_ims/
modules/pcscf/ipsec_P_Drop.sh")
```

Výpis 4.26: Implementácia IPsec zabezpečenia v P-CSCF

```
if (!P_is_integrity_protected()){
--------------IPsec je volitelný----------
    #P_remove_security_client();         |
--------------IPsec je povinný------------
    if (!P_remove_security_client()){    |
        route(REGISTER_494);             |
        break;                           |
    }                                    |
    P_add_integrity_protected("no");
}else{
        if (!P_remove_security_verify()||
            !P_remove_security_client()){
            route(REGISTER_494);
            break;
        }
    P_add_integrity_protected("yes");
};
```

Výpis 4.27: Spôsob implementácie IPsec v P-CSCF

Rovnako bolo nutné zmeniť konfiguráciu S-CSCF v súbore `scscf.cfg`, aby bol použitý tretí spôsob implementácie. S-CSCF zároveň rozhoduje, či sa musia klienti

opätovne autentizovať pri ďalších spojeniach alebo bude registrovaným klientom umožnené pripojenie aj bez autentizácie. Nastavenie je zobrazené na výpise 4.29.

```
----------IPsec je povinný-------------------------------------
    S_challenge("oims");
    route(Service_Routes);
    t_reply("401","Unauthorized:Challenging_the_UE");
    break;
}else{
```

Výpis 4.28: Spôsob implementácie IPsec v S-CSCF

```
----------Povinná reautentizácia-----------------------------
#if (!S_is_authorized("oims")){
#   S_challenge("oims");
#   route(Service_Routes);
#   t_reply("401","Unauthorized:Challenging_the_UE");
#   exit;
#}
----------Registrovaní klienti sa nemusia
                          znova autentizovať--------------------
if (S_is_not_registered()){
    if (S_assign_server("oims")){
        route(Service_Routes);
        route(Charging_Function_Addresses);
        t_reply("200",
                "OK:SAR_succesful_and_registrar_saved");
        ISC_match_filter_reg("0");
        exit;
    }
```

Výpis 4.29: Spôsob autentizácie v S-CSCF

4.4.2 TLS zabezpečenie

Pre TLS zabezpečenie boli vygenerované kľúče, pomocou skriptu `tls_prepare.sh`, ktoré sa automaticky uložia do adresáru `PCSCF_CA`. Konfigurácia P-CSCF v súbore `pcscf.cfg` je zobrazená na výpise 4.31, bola definovaná IP adresa, port a implicitné povolenie TLS. Ďalej je nutné načítať certifikáty a TLS modul, ktorý určuje verziu TLS, overovanie certifikátov a kompresiu prenášaných dát.

```
listen=tls:192.168.20.179
tls_port_no=4061
enable_tls=yes
                                -
modparam("pcscf","use_tls", 1)
modparam("pcscf","tls_port", 4061)
```

Výpis 4.30: Základné nastavenie TLS v P-CSCF

```
loadmodule "/opt/OpenIMSCore/ser_ims/modules/tls/tls.so"
                                -
modparam("tls","tls_method", "TLSv1")
modparam("tls","private_key",
        "/opt/OpenIMSCore/PCSCF_CA/pcscf_private_key.pem")
modparam("tls","certificate",
        "/opt/OpenIMSCore/PCSCF_CA/pcscf_cert.pem")
modparam("tls","ca_list",
        "/opt/OpenIMSCore/PCSCF_CA/pcscf_ca_list.pem")
modparam("tls","verify_certificate", 1)
modparam("tls","require_certificate", 1)
modparam("tls","tls_disable_compression", 1)
```

Výpis 4.31: Načítanie TLS modulu a certifikátov v P-CSCF

4.4.3 THIG (Topology Hiding Inter-network Gateway)

SIP posiela správy vo forme otvoreného textu, preto je náchylný na útoky typu MitM (Man-in-the-Middle). THIG šifruje polia s citlivými údajmi a zabraňuje tak ich zneužitiu. Funkciu THIG poskytuje I-CSCF, na konfiguráciu slúži samostatný súbor `icscf.thig.cfg` a na spustenie slúži skript `icscf.thig.sh`. Na začiatku bola upravená sieťová konfigurácia a následne THIG parametre (názov, adresa, port, spôsob práce, veľkosť hashu, autentizačný server). Na výpise 4.33 je vidieť Wiresharkom odchytenú SIP komunikáciu bez použitia THIG a pri použití THIG.

```
listen=192.168.20.179
port=5060
alias="icscf.oims"
alias="oims"

modparam("icscf","thig_name","thig@icscf.oims")
modparam("icscf","thig_host","192.168.20.179")
modparam("icscf","thig_port",5060)
modparam("icscf","thig_param","thigenc")
modparam("icscf","hash_size",128)
modparam("icscf","aaa_peer","hss.oims")
```

Výpis 4.32: Konfigurácia THIG v I-CSCF

```
--------------Komunikácia s THIG-----------------------------
Message Header
    Via: SIP/2.0/UDP bff575d6adce34e3674f0116679bbc33;branch=0
    Via: SIP/2.0/UDP 9aba453b9f4015a0e3cdf9cacff4d0ca;branch=0
    Via: SIP/2.0/UDP 0e932ec2d551cd88acd93cd35313c5d1;branch=1
From: <sip:0d6db153343c43cd1e2f52c4ef88a56b>;tag=1008
To: <sip:976e5c88e5dbed6ec8d6b338d8fb069f>;tag=1003
--------------Komunikácia bez THIG---------------------------
Message Header
    Via: SIP/2.0/UDP 192.168.20.179:6060;branch=0
    Via: SIP/2.0/UDP 192.168.20.179:4060;branch=0
    Via: SIP/2.0/UDP 192.168.20.120:5062;rport=5062;branch=1
From: <sip:adam@oims>;tag=1008
To: <sip:bob@oims>;tag=1003
```

Výpis 4.33: Výpis komunikácie bez a s použitím THIG

Záver

V prvej časti je obsiahnutá teória, ktorú bolo potrebné naštudovať pre správne porozumenie architektúre IP Multimedia Subsystem. Zahrňuje všetky hlavné prvky IMS architektúry ako CSCF, HSS, identita IMS a iné. Okrem toho okrajovo charakterizuje najpoužívanejšie protokoly v rámci IMS systému. Druhá časť stručne popisuje open source prostredie Open IMS Core, ktoré je použité na implementáciu IMS siete. Tretia kapitola sa venuje doplnkovým službám Emergency-CSCF a IMS presence architektúre, približuje ich časti a spôsob práce.

Praktická časť sa zaoberá využitím Open IMS Core na implementáciu IMS architektúry. Ako prvý bol použitý virtuálny obraz Ubuntu 9.04. Najskôr bolo potrebné overiť funkčnosť všetkých prvkov systému, najjednoduchší spôsob je vykonať hovor medzi užívateľmi, pretože užívateľ sa musí spojiť s CSCF časťou pomocou P-CSCF, následne prebieha komunikácia s I-CSCF, S-CSCF a HSS. Tým sú zapojené do komunikácie všetky hlavné prvky. Pri prvej konfigurácii bola použitá localhost adresa, na ktorej defaultne beží celá architektúra, prvé testovanie sa odporúča vykonať na localhost adrese, pretože môže odhaliť problémy priamo súvisiace s prostredím Open IMS Core. Tým sa zaručí, že prípadné neskoršie problémy budú mať súvis s použitou konfiguráciou a zjednoduší sa ich hľadanie. Jediný odhalený problém spočíval v obsadených portoch, riešením bola úprava nastavenia DIAMETER rozhraní.

Po úspešnom otestovaní funkčnosti prostredia pomocou hovoru odchyteného Wiresharkom bola zmenená IP adresa a názov domény. Záznam o doméne bol uložený na školskom DNS serveri, aby bola možná komunikácia aj zo zariadení mimo IMS doménu. Najskôr bol nakonfigurovaný lokálny DNS server na adresu 192.168.20.179 a doménu `oims`. Potom bola upravená konfigurácia celého Open IMS Core prostredia, konfiguračných súborov HSS a jednotlivých CSCF prvkov, boli zmenené konfiguračné súbory všetkých DIAMETER rozhraní a inicializačné súbory databáz. DNS nastavenie bolo otestované pomocou pingu z VM aj z počítaču mimo doménu. V HSS databáze bol vytvorený nový užívateľ Adam. Komunikácia medzi počítačom mimo doménu `oims` a VM bola zachytená Wiresharkom a spracovaná do formy flow diagramu. Tým bola úspešne otestovaná dostupnosť domény z vonkajšej siete.

Ďalším krokom bola inštalácia rozšírení, ktoré umožňujú vykonávanie núdzových volaní, konrétne prvky E-CSCF a LRF. Nebolo možné stiahnuť rozšírenia priamo na pripravenú VM, pretože Ubuntu 9.04 nespĺňa požadované bezpečnostné parametre zo strany serveru. Po krátkom testovaní na Ubuntu 20.04, 18.04 a 12.04 bola zvolená VM Ubuntu 12.04, pretože vykazovala najlepšiu kompatibilitu s prostredím Open IMS Core. Pri inštalácii bolo prekonaných pár menších problémov spomenutých v praktickej časti práce. Závažný problém sa vyskytol po úspešnej inštalácii s prvkami P-CSCF, E-CSCF a LRF, ktoré priamo súvisia s núdzovou

architektúrou. Prvky sa nespustili, kvôli chybne načítanému modulu `lrf.so`, respektíve neznámej metóde `curl_easy_perform`, ktorú obsahoval skompilovaný kód `lib_lost_client.so`. Kontrola dynamického modulu curl, ktorý je zodpovedný za interpretáciu a prácu s vyššie uvedenými súbormi prebehla úspešne, rovnako aj kontrola knižnice libcurl a kontrola ich vzájomného prepojenia. Problém spočíva pravdepodobne v nedokonalej kompatibilite s novšími verziami prerekvizít, respektíve v úzkom prepojení zdrojového kódu a konkrétnej verzie obsluhujúceho modulu. Posledná možnosť bola preniesť skomprimované zdrojové kódy na pripravenú VM Ubuntu 9.04 a pokúsiť sa o inštaláciu. Na pripravenej VM nakoniec aj tak chýbal modul curl, pravdepodobne je pripravená len na základnú verziu Open IMS Core. Po doplnení modulu curl a úspešnej inštalácii Open IMS Core sa vykytol úplne rovnaký problém aj výpisy chýb. Z predošlého bol vyvodený záver, že problém nesúvisí s kompatibilitou operačného systému, ale pravdepodobne so závislosťou na špecifickej verzii jednotlivých obsluhujúcich modulov alebo priamo v poskytnutom zdrojovom kóde rozšírení Open IMS Core. Napriek tomu, že bol pochopený koncept fungovania, implementácia núdzových volaní dopadla neúspešne.

Nasledujúca práca sa zamerala na implementáciu zabezpečenia na rozhraní medzi UE a IMS sieťou alebo pri spojení IMS siete s vonkajšou sieťou. Zabezpečenie spočíva v používaní IPsec, TLS a THIG pri komunikácii. IPsec zabezpečenie bolo nastavené na P-CSCF a S-CSCF prvkoch, ktoré povolia pripojenie už len užívateľom využívajúcim IPsec. Mód povinnej autentizácie pri každom spojení bol ponechaný v stave, kedy sa registrovaní užívatelia nemusia opätovne autentizovať. Zabezpečenie TLS pridalo možnosť autentizácie pomocou certifikátov. P-CSCF s TLS vyžaduje od užívateľa certifikát a zároveň poskytuje svoj vlastný, použitá je TLS verzia 1.0. THIG funkcionalitu poskytuje I-CSCF, šifrovanie jednotlivých polí prebieha v každej správe smerujúcej mimo IMS sieť. Pomocou Wiresharku bola zachytená komunikácia pri použití THIG, viditeľné zašifrovanie citlivých údajov overilo správnu funkčnosť.

Problémom pri práci s prostredím Open IMS Core je ťažko dostupná kvalitná dokumentácia, väčšina zmien v štruktúre programu v nej nie je zachytená, núdzové rozšírenie sa v nej ani nenachádza. Väčšina odkazov z dokumentácie a oficiálnych stránok nefunguje, odkaz na odporúčané verzie prvkov LoST a LOCSIP bol takisto neplatný. Posledná aktualizácia na stránke je z roku 2014 a v mailing listoch sú odpovede na nové problémy veľmi zriedkavé, posledné 4 roky už žiadne. To veľmi obmedzuje pochopenie zdrojového kódu a hľadanie riešení na vyskytujúce sa problémy. Samostatný problém je zlá kompatibilita s novými verziami operačného systému Ubuntu (verzia 14 a viac), keďže Open IMS Core je úzko spätý s konkrétnymi verziami modulov, či už sa jedná o curl, JDK balíček alebo MySQL databázu. Nové verzie Ubuntu používajú repozitáre, ktoré staršie moduly ani neobsahujú. Pri použití starej verzie Ubuntu (napríklad 9.04) je zase problém s bezpečnostnými ak-

tualizáciami a repozitármi, ktoré neexistujú už približne 10 rokov, takže staré verzie modulov je nutné hladať na neoficiálnych úložiskách, ktoré rovnako trpia zlou alebo žiadnou údržbou. Najlepší nájdený kompromis poskytovalo Ubuntu 12.04.

Literatúra

[1] BEŽILLA, J. *Definovanie pojmu služby sietí NGN*. Žilina, 2009. Bakalárska práca. Žilinská univerzita v Žiline, Elektrotechnická fakulta, Katedra telekomunikácií a médií. Vedoucí práce Prof. Ing. Karol Blunár, DrSc.

[2] ZNATY S.; DAUPHIN J. *IP Multimedia Subsystem: Principles and Architecture*. 2005. Dostupné na internete: `http://www.efort.com/media_pdf/IMS_ENG.pdf`

[3] CAMARILLO G.; GARCÍA-MARTÍN M. A. *The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds*. 2nd edition. Chichester, England: Wiley, 2006. ISBN 0-470-01818-6.

[4] PODHRADSKÝ, P.; MIKÓCZY, E.; DÚHA, J.; TRÚCHLY, P.; BLICHÁR, J. *Siete budúcej generácie: Vybrané kapitoly* [online]. Praha, Česká republika: ČVUT v Prahe, 2013 [cit. 2020-10-25]. ISBN 978-80-01-05295-2. Dostupné na internete: `http://techpedia.fel.cvut.cz/sk/single/?objectId=44`

[5] KOUKAL, M.; BESTAK, R. Architecture of IP Multimedia Subsystem. *Proceedings ELMAR 2006* [online]. IEEE, 2006, s. 323-326 [cit. 2020-10-28]. ISBN 953-7044-03-3. ISSN 1334-2630. Dostupné na internete: `http://ieeexplore.ieee.org/document/4127549`

[6] ASHRAF, S. R.; MOTTAHED B. D. *IMS network architecture witch integrated network elements*. 2008 US. 11/799.944. Uděleno 3. 11. 2008. Zapsáno 3. 5. 2007.

[7] INTERNET ENGINEERING TASK FORCE (IETF). *RFC 3261: SIP: Session Initiation Protocol* [online]. June 2002 [cit. 2020-11-3]. Dostupné na internete: `https://www.rfc-editor.org/rfc/rfc3261.txt`

[8] INTERNET ENGINEERING TASK FORCE (IETF). *RFC 4566: SDP: Session Description Protocol* [online]. July 2006 [cit. 2020-11-3]. Dostupné na internete: `https://www.rfc-editor.org/rfc/rfc4566.txt`

[9] INTERNET ENGINEERING TASK FORCE (IETF). *RFC 3550: RTP: A Transport Protocol for Real-Time Applications* [online]. July 2003 [cit. 2020-11-4]. Dostupné na internete: `https://www.rfc-editor.org/rfc/rfc3550.txt`

[10] INTERNET ENGINEERING TASK FORCE (IETF). *RFC 6733: Diameter Base Protocol* [online]. October 2012 [cit. 2020-11-4]. Dostupné na internete: `https://www.rfc-editor.org/rfc/rfc6733.txt`

[11] JEŽEK, J. *Open Source implementace IMS*. Brno, 2020. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technológií, Ústav telekomunikací. Vedoucí práce Ing. Pavel Šilhavý, Ph.D.

[12] EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE (ETSI). *3GPP TS 23.167 version 13.2.0 Release 13*. 3GPP, ETSI, March 2016. Dostupné na internete: `https://www.etsi.org/deliver/etsi_ts/123100_123199/123167/13.02.00_60/ts_123167v130200p.pdf`

[13] PETRÁŠ, D.; BARONAK, I.; CHROMÝ, E. Presence Service in IMS. *The Scientific World Journal* [online]. Article ID 606790, 2013. Dostupné na internete: `https://doi.org/10.1155/2013/606790`

Zoznam symbolov, veličín a skratiek

3GPP The 3rd Generation Partnership Project

AAA Autentizácia, autorizácia a účtovanie – Authentication, Authorization and Accounting

AH Authentication Header

AS Aplikačný server – Application Server

AVP Attribute-Value Pair

B2BUA Back-to-Back User Agent

BGCF Breakout Gateway Control Function

CAMEL Customized Applications for Mobile network Enhanced Logic

CAP CAMEL Application Part

CSCF Call Session Control Function

DBMS Database Management System

DHCP Dynamic Host Configuration Protocol

DNS Domain Name System

E-CSCF Emergency CSCF

EPS Evolved Packet System

ESP Encapsulating Security Payload

GCC GNU Compiler Collection

GPRS General Packet Radio Service

GSM Global System for Mobile Communications

gsmCF GSM Service Control Function

HSS Home Subscriber Server

HTTP Hypertext Transfer Protocol

I-CSCF Interrogating-CSCF

IETF Internet Engineering Task Force

IMS IP Multimedia Subsystem

IM-SSF IP Multimedia Service Switching Function

IP Internet Protocol

IPsec Internet Protocol Security

IPTV IP televízia – Internet Protocol Television

IPv4 Internet Protocol version 4

IPv6 Internet Protocol version 6

ISC IP Multimedia Service Control

ISIM IP Multimedia Services Identity Module

ISUP ISDN User Part

ITU-T International Telecommunication Union – Telecommunication Standardization Sector

JDK Java Development Kit

LOCSIP Location in SIP/IP

LRF Location Retrieval Function

LoST Location to Service Translation Protocol

MAP Mobile Application Part

MGCF Media Gateway Control Function

MGW Mediálna brána – Media Gateway

MitM Man-in-the-Middle

MRF Media Resource Function

MRFC Media Resource Function Controller

MRFP Media Resource Function Processor

NAI Network Access Identifier

OSA-SCS Open Service Access-Service Capability Server

PCM Pulzná kódová modulácia – Pulse Code Modulation

P-CSCF Proxy-CSCF

PDF Policy Decision Function

PoC Push-to-talk over Cellular

PSAP Public Safety Answering Point

PSTN Public Switched Telephone Network

QoS Kvalita služby – Quality of Service

RADIUS Remote Authentification Dial In User Service

RTCP Real-time Transport Control Protocol

RTP Realtime Transport Protocol

RTSP Real Time Streaming Protocol

S-CSCF Serving-CSCF

SCTP (Stream Control Transmission Protocol

SDP Session Description Protocol

SGW Signaliačná brána – Signaling Gateway

SIGTRAN Signaling Transport over IP

USIM Subscriber Identity Module

SIP Session Initiation Protocol

SLF Subscription Locator Function

SS7 Signalizačný systém č. 7 – Signaling System No. 7

SSF Service Switching Function

TCP Transmission Control Protocol

TEL Telefón – Telephone

THIG Topology Hiding Inter-network Gateway

TLS	Transport Layer Security
UA	Koncové zariadenie – User Agent
UDP	User Datagram Protocol
UI	Identita užívateľa – User Identity
UMTS	Universal Mobile Telecommunications System
URI	Uniform Resource Identifier
USIM	Universal Subscriber Identity Module
VM	Virtuálne zariadenia – Virtual Machine
VMDK	Disk virtuálneho zariadenia – Virtual Machine Disk
VoD	Video na vyžiadanie – Video on Demand
XML	Extensible Markup Language
XDMS	XML Data Management Server

A Skrátené výpisy konfiguračných súborov

```
zone "oims" IN{
    type master;
    file "/etc/bind/open-ims.dnszone";
};
```

Výpis A.1: Nastavenie DNS v súbore `named.conf`

```
options {
directory "/var/cache/bind";
forwarders {192.168.20.179;};
auth-nxdomain no;
listen-on {192.168.20.179; };
listen-on-v6 { ::; };
};
```

Výpis A.2: Nastavenie DNS v súbore `named.conf.options`

```
$ORIGIN oims.
$TTL 1W
@       1D IN SOA       oims. root.ims.(
        2006101001      ; serial
        3H              ; refresh
        15M             ; retry
        1W              ; expiry
        1D )            ; minimum
        1D IN NS        oims
pcscf   1D IN A         192.168.20.179
scscf   1D IN A         192.168.20.179
icscf   1D IN A         192.168.20.179
oims.   1D IN A         192.168.20.179
hss     1D IN A         192.168.20.179
oims. 1D IN NAPTR 10 50 "s" "SIP+D2U" "" _sip._udp
oims. 1D IN NAPTR 20 50 "s" "SIP+D2T" "" _sip._tcp
_sip       1D SRV 0 0 5060 icscf
_sip._udp  1D SRV 0 0 5060 icscf
_sip._tcp  1D SRV 0 0 5060 icscf
ue         1D IN A      192.168.20.179
presence   1D IN A      192.168.20.179
```

Výpis A.3: Nastavenie DNS v súbore `open-ims.dnszone`

```
host=192.168.20.179
port=8008
```

Výpis A.4: Zmeny v súbore `hss.properties`

```
INSERT INTO 'preferred_scscf_set'
    VALUES (1,1,'scscf1','sip:scscf.oims:6060,0');
INSERT INTO 'visited_network' VALUES (1,'oims');
```

Výpis A.5: Zmeny v súbore `userdata.sql`

```
listen=192.168.20.179
alias="pcscf.oims":4060
                          -
modparam("pcscf","name","sip:pcscf.oims:4060")
modparam("pcscf","ipsec_host","192.168.20.179")
modparam("pcscf","rtpproxy_socket","udp:192.168.20.179:34999")
modparam("pcscf","icid_gen_addr","192.168.20.179")
modparam("pcscf","orig_ioi","oims")
modparam("pcscf","term_ioi","oims")
modparam("pcscf","ecscf\_uri","sip:ecscf.oims:7060")
                          -
P_add_p_visited_network_id("oims");
P_access_network_info("oims");
```

Výpis A.6: Konfigurácia P-CSCF v súbore `pcscf.cfg`

```
listen=192.168.20.179
alias="scscf.oims":6060
                          -
modparam("scscf","name","sip:scscf.oims:6060")
modparam("isc","my_uri","scscf.oims:6060")
                          -
S_assign_server_unreg("oims", "orig");
if (uri=~"sip:(.*)@oims(.*)" || uri=~"tel:.*"){
S_assign_server_unreg("oims", "term");
if (uri=~"sip:(.*)oims(.*)"){
if (!S_is_integrity_protected("oims")){
if (!S_is_authorized("oims")) {
S_challenge ("oims");
                          -
if (S_assign_server("oims")){
if (S_assign_server("oims")){
if (S_assign_server("oims")){
if (S_assign_server("oims")){
                          -
if (S_check_visited_network_id("oims")){
S_add_service_route("sip:thig@icscf.oims");
                          -
if (S_check_visited_network_id("oims")){
                          -
t_relay_to_udp("192.168.20.179",6060);
t_relay_to_udp("192.168.20.179",6060);
                          -
if (uri=~"sip:(.*)oims(.*)"){
if (uri=~"sip:(.*)oims(.*)"){
if (uri=~"sip:\+[0-9]+@oims.*user=phone.*"){
t_relay_to_udp("192.168.20.179",6060);
```

Výpis A.7: Konfigurácia S-CSCF v súbore `scscf.cfg`

```
listen=192.168.20.179
alias="icscf.oims"
alias="oims"
                        -
modparam("icscf","name","icscf.oims")
modparam("icscf","forced_hss_peer","hss.oims")
modparam("icscf","icid_gen_addr","192.168.20.179")
modparam("icscf","orig_ioi","oims")
modparam("icscf","term_ioi","oims")
                        -
t_relay_to_udp("192.168.20.179", "9060");
                        -
if (! uri=~".*@oims.*")
if (uri=~".*@oims.*")
```

Výpis A.8: Konfigurácia I-CSCF v súbore `icscf.cfg`

```
INSERT INTO 'nds_trusted_domains' VALUES (1,'oims');
                        -
INSERT INTO 's_cscf' VALUES
    (1,'First_and_only_S-CSCF','sip:scscf.oims:6060');
```

Výpis A.9: Konfigurácia I-CSCF v súbore `icscf.sql`

```
#modparam("scscf","registration_default_algorithm",
    "AKAv1-MD5")
#modparam("scscf","registration_default_algorithm",
    "AKAv2-MD5")
modparam("scscf","registration_default_algorithm","MD5")
#modparam("scscf","registration_default_algorithm",
    "CableLabs-Digest")
#modparam("scscf","registration_default_algorithm",
    "3GPP-Digest")
modparam("scscf","registration_default_algorithm",
    "TISPAN-HTTP_DIGEST_MD5")
```

Výpis A.10: Zmena autentizačného algoritmu v `scscf.cfg`

B Konfiguračné súbory Open IMS Core

V elektronickej prílohe sú všetky súbory použité pri konfigurácii Open IMS Core. Súbory sú rozdelené do adresárovej štruktúry uvedenej na obrázku B.1. V nasledujúcich troch podkapitolách je uvedený stručný popis jednotlivých súborov.

B.1 Konfiguračné súbory DNS

Konfiguračné súbory I-CSCF:

- `named.conf` – vytvorenie DNS záznamu pre doménu `oims`.
- `named.conf.options` – nastavenie adresy, na ktorej počúva DNS server.
- `open-ims.dnszone` – priradenie adries ku názvom prvkov.
- `resolv.conf` – nastavenie DNS resolveru na názov domény `oims` a IP adresu 192.168.20.179.

B.2 Konfiguračné súbory CSCF

Konfiguračné súbory I-CSCF:

- `icscf.cfg` – úprava konfigurácie I-CSCF (zmena IP adresy, zmena parametrov modulov a funkcí).
- `icscf.sql` – úprava dát vkladaných do databáze.
- `icscf.xml` – nastavenie DIAMETER rozhrania.

Konfiguračné súbory P-CSCF:

- `pcscf.cfg` – úprava konfigurácie P-CSCF (zmena IP adresy, zmena parametrov modulov a funkcí).
- `pcscf.xml` – nastavenie DIAMETER rozhrania.

Konfiguračné súbory S-CSCF:

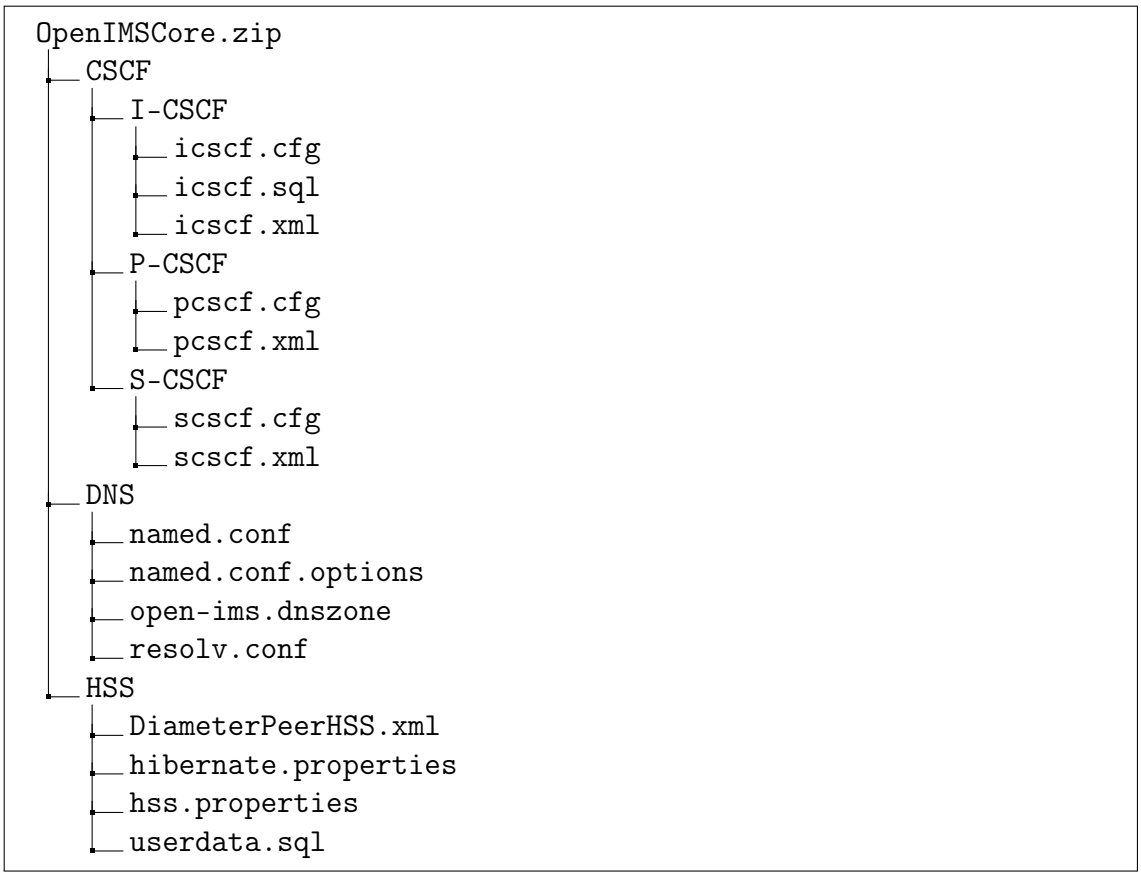
- `scscf.cfg` – úprava konfigurácie S-CSCF (zmena IP adresy, zmena parametrov modulov a funkcí).
- `scscf.xml` – nastavenie DIAMETER rozhrania.

B.3 Konfiguračné súbory HSS

Konfiguračné súbory HSS:

- `DiameterPeerHSS.xml` – nastavenie DIAMETER rozhrania.
- `hibernate.properties` – nastavenie frameworku hibernate.

- `hss.properties` – nastavenie IP adresy a portu.
- `userdata.sql` – úprava dát vkladaných do databáze.



Obr. B.1: Adresárová štruktúra elektronickej prílohy