VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2020 Bc. Jiří Ježek

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

OPEN SOURCE IMPLEMENTACE IMS
OPEN SOURCE IMS IMPLEMENTATIONS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — **Bc. Jiří Ježek**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Pavel Šilhavý, Ph.D.**
SUPERVISOR

BRNO 2020

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**

Ústav telekomunikací

Student: Bc. Jiří Ježek ***ID:*** 186523
Ročník: 2 ***Akademický rok:*** 2019/20

NÁZEV TÉMATU:

Open Source implementace IMS

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte technologii IP Multimedia Subsystem (IMS). Porovnejte Open Source implementace technologie IMS. Vytvořte laboratorní úlohu umožňující se podrobně seznámit s vybraným Open Source IMS projektem. Realizujte výkonnostní testování dvojice vybraných Open Source IMS implementací s využitím HW testeru. Zaměřte se rovněž na stabilitu, odolnost vůči útokům a implementované zabezpečení vůči nim.

DOPORUČENÁ LITERATURA:

[1] Chakraborty, S., Peisa, J., Frankkila, T, Synnergren, P. IMS Multimedia Telephony over Cellular Systems : VoIP Evolution in a Converged Telecommunication World. Wiley, 2007. ISBN 978-0-470-05855-8

[2] Baroňák, I., Chamraz, F., Csóka, F., Zafčík, J., Šíp, T., Hartmann, M.. IMS technológie. Bratislava : Spektrum STU, 2017. ISBN 978-80-227-4738-7.

Termín zadání: 3.2.2020 ***Termín odevzdání:*** 1.6.2020

Vedoucí práce: Ing. Pavel Šilhavý, Ph.D.

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

ABSTRAKT

Diplomová práce se zabývá open source implementacemi IMS technologie. V teoretické části je popsáno fungování IMS a její architektura včetně využívaných protokolů, procesu registrace a sestavení spojení a také potenciálních bezpečnostních hrozeb IMS systémům. Byly zvoleny dva open source IMS projekty, konkrétně Open IMS Core a Project Clearwater, na které se práce zaměřuje. Diplomová práce se ve své následující části věnuje popisu vybraných open source IMS projektů. V této části je zahrnut i popis implementovaných zabezpečení obou projektů a podrobný postup instalace systému i s nutnou konfigurací. Diplomová práce se ve své praktické části zabývá výkonnostním testováním zvolených projektů pomocí hardwarového testeru Abacus 5000. Nejdříve byly provedeny inicializační testy, kterými byla ověřena možnost komunikace testeru s testovanými projekty. Následně bylo možné provést výkonnostní testování zvolených projektů. Při výkonnostním testování je kladen důraz na stabilitu systémů, rychlost sestavení hovoru, dobu odezvy SIP zpráv a rychlosti registrace uživatelů do sítě, při různých úrovních zátěže. Systémy byly testovány jak pro sestavení hovorů, tak pro registraci uživatele. Součástí testování je i realizace záplavového DoS útoku inviteflood a reakce systému na zvyšující se intenzitu útoku.

Poslední kapitola je věnována porovnání vybraných IMS projektů, kde jsou přehledně prezentovány výsledky testů. Částečně se v této kapitole vychází i z veřejně dostupných informací, jako jsou poskytnuté technické dokumentace jednotlivých projektů a veřejně dostupné mailing listy.

Součástí diplomové práce je i laboratorní úloha, ve které si student vyzkouší práci s Open IMS Core. Laboratorní úloha se zaměřuje na proces vytvoření uživatele, jeho registrace do IMS sítě, navázání hovoru mezi jednotlivými uživateli a vnitřnímu směrování v IMS sítích. Úloha se zaměřuje zejména na signalizační procesy v rámci realizovaných úkonů.

KLÍČOVÁ SLOVA

bezpečnost IMS, IMS, IP multimedia subsystem, Open IMS Core, Project Clearwater, výkonnostní testování IMS

ABSTRACT

The diploma thesis deals with open source implementations of IMS technology. The theoretical part describes the functioning of IMS and its architecture, including the protocols used, the process of registration and connection establishment, as well as potential security threats to IMS systems.

Two open source IMS projects were selected, Open IMS Core and Project Clearwater, on which the work is focused. In its next part, the diploma thesis deals with the description of selected open source IMS projects, this part also includes a description of the implemented security of both projects and a detailed procedure of system installation with any necessary configuration.

In its practical part, the diploma thesis focused on performance testing of selected projects using the Abacus 5000 hardware tester. First, initialization tests were performed, which verified the possibility of communication between the tester and the tested projects. Subsequently, it was possible to perform performance testing of selected projects. In performance testing, emphasis is placed on system stability, performance in the area of call set-up speed and response time of SIP messages and speed of user registration in the network, at different load levels. The systems have been tested for both call set-up and user registration. Part of the testing is also the implementation of the flood DoS inviteflood attack and the system's response to the increasing intensity of the attack.

The last chapter is devoted to the comparison of selected IMS projects, where the test results are clearly presented. This chapter is partly based on publicly available information, such as the technical documentation of individual projects and possibly publicly available mailing lists.

Part of the diploma thesis is also a laboratory task in which the student tries to work with Open IMS Core. The laboratory task focuses on the process of creating a user and his registration in the IMS network, establishing a call between individual users and internal routing in IMS networks. The task focuses mainly on signaling processes within the implemented tasks.

KEYWORDS

IMS, IMS performance testing, IMS security, IP multimedia subsystem, Open IMS Core, Project Clearwater

JEŽEK, Jiří. *Open Source implementace IMS*. Brno, Rok, 125 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Pavel Šilhavý, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Open Source implementace IMS“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing.Pavlovi Šilhavému, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

Seznam obrázků

Seznam tabulek

Seznam výpisů

Úvod

IP Multimedia Subsystem (IMS) je standardizovaná síťová architektura pro telekomunikační operátory, kteří chtějí poskytovat mobilní a pevné multimediální služby. IMS je specifikovaný v 3GPP (The 3rd Generation Partnership Project) UMTS (Universal Mobile Telecommunication System) vydání 5, které bylo zveřejněno v březnu roku 2002.

Sítě s podporou IMS poskytují přístup ke službám nezávisle na přístupové technologii, ať už jde o sítě paketové nebo o sítě s přepojováním okruhů. Cílem IMS je přenést výhody internetu do 3G celulárních systémů a nadále rozvíjet celulární sítě [1][2].

Nejznámějším zástupcem využití IMS je služba Voice over LTE (VoLTE). VoLTE je technologie pro přenos hlasových hovorů pomocí LTE, čímž je zajištěna vyšší kvalita přenášeného zvuku, rychlejší zahájení hovoru atd.

IMS však není využívána pouze ve službě VoLTE, využití nachází i při textové interaktivní komunikaci (instant messaging), v přenosu médií na vyžádání, jako je například video na vyžádání (Video on Demand) nebo hudba na vyžádání (Music on Demand).

IMS má do budoucna vysoký potenciál v oblasti telekomunikací. Aktuální je velký rozvoj v oblasti internetu věcí (IoT) a cloudových služeb, které mohou plně využít architekturu IMS. Vysoký potenciál IMS v budoucnu je dán i nastupující technologií 5G, která může v IMS plnit roli nejvýznamnější přístupové sítě.

Prvním cílem diplomové práce je nastudovat technologii IMS. Následujícím je prozkoumat vybrané open source implementace technologie IMS. Dalším cílem je zvolené technologie porovnat za pomoci dostupných dat a testování pomocí hardwarového testeru Abacus 5000, se zaměřením na výkonnost a stabilitu systému. Následujícím cílem je zjistit možnosti implementovaného zabezpečení a odolnosti vůči útokům. Posledním cílem bylo vytvořit laboratorní úlohu, umožňující studentům se seznámit s IMS architekturou.

Předkládaná práce se ve své první kapitole zabývá popisem architektury IMS sítě. Tato kapitola obsahuje rozdělení architektury na jednotlivé logické vrstvy a jejich popis. Následuje popis jednotlivých komponent a funkcí které plní. V následující sekci je rozebrána identifikace v rámci IMS sítí. Další sekce obsahuje popis protokolů aplikační vrstvy, které IMS využívá pro svou funkci. Následující sekce se zabývá popisem registrace a navázání spojení v rámci IMS sítí. Poslední sekce se zabývá potenciálními bezpečnostními hrozbami IMS sítě.

V následující kapitole jsou popsána dvě vybraná open source IMS řešení, konkrétně jde o projekty Open IMS Core a Project Clearwater. Tato kapitola obsahuje konkrétní popis architektur vybraných projektů. Součástí kapitoly je i popis implemen-

tovaných možností zabezpečení systému proti potenciálním bezpečnostním hrozbám. Rovněž je zde popsán i postup instalace a základní konfigurace zvolených řešení spolu s realizací zkušebního hovoru pomocí některého z dostupných softwarových IMS klientů.

Následující kapitola se zabývá testováním vybraných IMS projektů pomocí hardwarového testeru Abacus 5000. V první sekci je krátce popsán hardwarový tester Abacus 5000. Následuje sekce ve které jsou shrnuty sledované parametry testů. Dále je popsáno nastavení testeru a i nicializační testování pro oba open source IMS projekty. Následuje sekce o možnostech vytvoření testovacích uživatelských profilů.

Následující sekce se věnují jednotlivým testům a jejich vyhodnocení. Nejdříve jsou v obou systémech testovány parametry sestavení hovoru pro různé zatížení. Následně jsou testovány parametry registrace s různým počtem registrací probíhajících současně. Poslední testování se věnuje reakci systémů na DoS záplavový útok inviteflood a možností zabezpečení proti tomuto útoku.

Poslední kapitola se věnuje porovnání zvolených IMS projektů vycházejících z veřejně dostupných informací a provedených testů.

Součástí diplomové práce je i laboratorní úloha, jejíž cílem je seznámení studentů s projektem Open IMS Core. Laboratorní úloha se zaměřuje na proces registrace uživatele v IMS systému, navázání hovoru mezi jednotlivými uživateli a vnitřnímu směrování v IMS sítích.

1 IMS

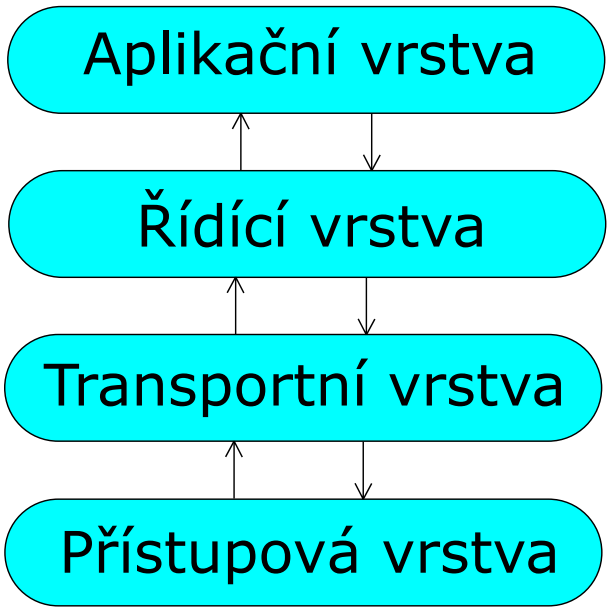
IMS poskytuje platformu pro komunikaci mezi všemi druhy uživatelských zařízení, od analogových telefonů, přes mobilní telefony, po osobní počítače a chytrá zařízení. Podporuje spolupráci s mobilními sítěmi, PSTN (veřejná telefonní síť) a ostatními sítěmi s přepojováním okruhů, s firemními intranety, poskytovateli internetového připojení a internetem.

1.1 Vrstvy

Ze své podstaty je většina síťových architektur, stejně jako IMS, rozdělena do vrstev kvůli přehlednosti a oddělitelnosti jednotlivých funkcionalit.
V IMS architektuře rozlišujeme čtyři vrstvy:

- přístupová vrstva,
- transportní vrstva
- řídící vrstva,
- aplikační vrstva.

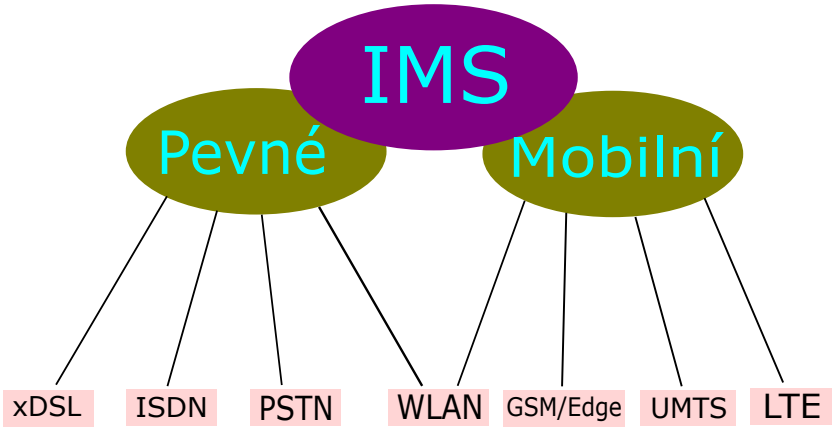
Při pohledu na jednotlivé vrstvy IMS architektury, můžeme pozorovat jistou analogii s modelem TCP/IP ve smyslu rozdělení do vrstev. Rozdělení do logických vrstev pomáhá pochopit síťovou hierarchii a zpřehledňuje komunikační schéma. Jednotlivé vrstvy využívají služby nižších vrstev a poskytují služby vyšším vrstvám. Rozdělení vrstev IMS sítě je znázorněno na obrázku 1.1.



Obr. 1.1: Vrstvy IMS

1.1.1 Přístupová vrstva

IMS umožňuje připojit zařízení nezávisle na přístupové technologii, jediným požadavkem je podpora protokolu IP a SIP (Session Initial Protocol). Lze se připojit i ze systémů nepodporujících zmiňované protokoly prostřednictvím komunikačních bran zajišťujících konverzi dat mezi sítěmi. Nezávislost na přístupové technologii demonstruje obrázek 1.2.



Obr. 1.2: Nezávsilost IMS na přístupové síti.

1.1.2 Transportní vrstva

Jedná se o IP síť, která se skládá z IP směrovačů (okrajové a základní IP směrovače). Zajišťuje konverzi dat přenášených mezi formáty dat přístupové a IMS sítě. Někdy bývá Transportní a přístupová vrstva dohromady označována jako vrstva připojení.

1.1.3 Řídící vrstva

Řídící vrstva obsahuje servery pro řízení a správu hovorů, nebo vytváření a úpravu relací. Dva hlavní prvky této vrstvy jsou soubor funkcí CSCF (call session control function) a HSS (home subscriber server). CSCF (někdy označován jako SIP server) provádí registraci koncových bodů a směrování SIP signalizace konkrétnímu aplikačnímu serveru. Databáze HSS udržuje profil uživatele, to může zahrnovat i informace o poloze, požadované služby, apod.

1.1.4 Aplikační vrstva

Využívá aplikační a obsahové servery k poskytování různých služeb s přidanou hodnotou. Hlavními komponenty jsou AS (aplikační servery), MRFC (multimedia resource function controller) a MRFP (multimedia resource function procesor).

AS je zodpovědný za řízení logiky specifické pro konkrétní službu, například toky volání a interakce uživatelského rozhraní s uživateli.
MRFP (multimedia resource function procesor), který je spíše známý jako mediální server, poskytuje doplňkové zpracování médií pro aplikační vrstvu. Přes MRFP, může poskytovatel služeb dodávat různé netelefonní služby, jako například push-to-talk, služby založené na řeči, video služby a další... [3]

1.2 IMS architektura

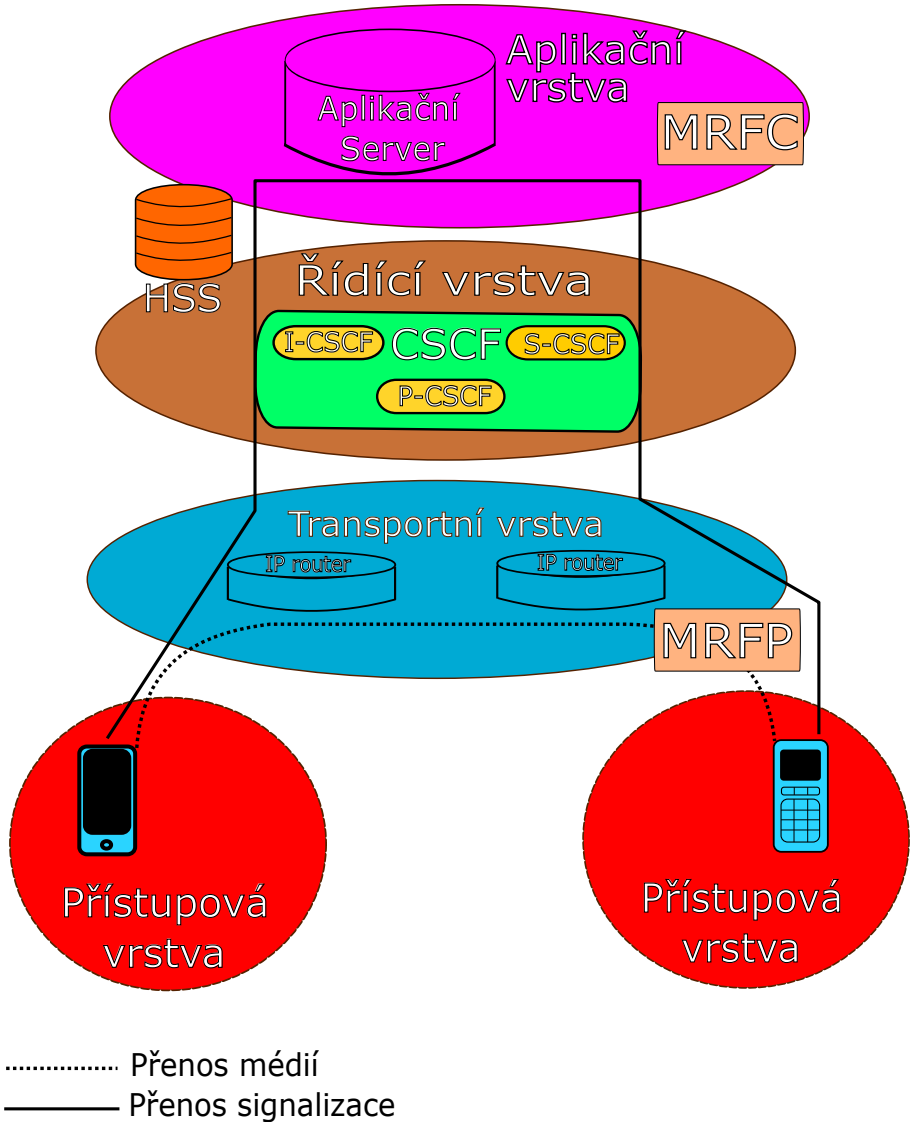
Tato část poskytuje podrobnější pohled na jádro síte IMS a popisuje funkci klíčových komponent jeho logického systému. Uspořádání jednotlivých komponent v rámci hierarchie jednotlivých vrstev znázorňuje obrázek 1.3.

1.2.1 Call Session Control Function

Jedním z hlavních účelů CSCF je signální směrovací funkce. CSCF zajišťuje veškerý signalizační provoz SIP a poskytuje následující síťové služby:

- služby řízení relací včetně účtování, směrování a roamingu,
- kombinace několika různých mediálních nosičů v relaci,
- účtování na základě centrálních služeb,
- bezpečná autentizace a důvěrnost založená na ISIM/USIM,
- řízení kvality služeb (QoS) pomocí policy decision function (PDF).

Po příchodu nového SIP volání, CSCF nejdříve autentizuje uživatele pomocí Home Subscriber Serveru (HSS). Potom je signalizace SIP směrována přes rozhraní IP Multimedia Service Control (ISC) do bloku aplikačních služeb. ISC definuje množinu filtrů, které jsou získány z HSS a přiděleny každému uživateli. V ISC každá SIP zpráva je porovnána s vhodným filtrem, pomocí kterého CSCF rozhodne ke kterým aplikačním službám směruje relaci a v jakém pořadí. Specifika relace jsou rozhodnuta na základě požadavků pro konkrétní službu (např. přítomnost volaného, preference volaného/volajícího, možnosti IMS zařízení nebo aktuální kredit volaného/volajícího, atd.).
Přestože CSCF předává SIP zprávy pomocí rozhraní ISC aplikacím, někdy je nutné zůstat aktivní v řízení relace. Toho je dosaženo přidáním informační hlavičky ve standardní SIP zprávě (REGISTER, INVITE, apod.). Pokud aplikace sama o sobě zvládne vyřídit požadavek na relaci, tak o tom informuje CSCF a směrování si řeší aplikace sama. Když je potřeba zahrnout více služeb do relace, tak aplikace první služby vrátí zprávu SIP do CSCF, kde jsou znovu prozkoumány ISC filtry, identifikuje další službu v řetězci a předá zprávu dál.



Obr. 1.3: Uspořádání jednotlivých komponent v rámci hierarchie jednotlivých vrstev

V UMTS vydání 6 specifikace 3GPP bylo IMS specifikováno jako nezávislé na přístupové technologii. Je důležité, aby poskytovatelé sítí podporující IMS nabízeli připojení k jejich službám z jakékoli cizí sítě. CSCF se dělí do tří oddělených funkcí:

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

Další důležitou rolí CSCF je poskytnutí nezávislosti mezi aplikačními službami. Rozhraní ISC se standardními filtry a SIP signalizací zaručuje nezávislý vývoj a nasazení mnoha aplikací [4], [2].

P-CSCF

P-CSCF je z pohledu uživatele prvním přístupovým bodem do IMS. Z pohledu SIP funguje P-CSCF jako odchozí a příchozí SIP proxy server. Všechny požadavky iniciované IMS zařízením nebo určené pro IMS zařízení prochází přes P-CSCF. Konkrétní P-CSCF je přidělen IMS zařízení během registrace a po dobu registrace se nemění (IMS zařízení komunikuje s jediným P-CSCF po celou dobu kdy je registrováno).
P-CSCF přiřadí identitu uživatele, která platí i pro zbývající uzly v rámci sítě. P-CSCF ověří formát SIP zpráv odeslaných do IMS sítě. Slouží jako filtr SIP zpráv, které nebyli vytvořeny v souladu s pravidly pro jejich vytváření.
P-CSCF také obsahuje kompresor a dekompresor SIP zpráv (IMS zařízení zahrnují taktéž oba). Zatímco zpráva SIP může být přenášena přes širokopásmové připojení za relativně krátký čas, přenos velkých SIP zpráv přes úzkopásmový kanál, může trvat několik sekund. Pro zkrácení doby přenosu SIP zprávy může být SIP zpráva komprimována na straně odesilatele a dekomprimována na straně příjemce.
P-CSCF může zahrnovat funkci PDF. PDF může být integrována přímo v P-CSCF, nebo může být implementována samostatně. PDF autorizuje zdroje a řídí kvalitu služeb v mediální části.
P-CSCF generuje informace potřebné k účtování. Síť IMS může obsahovat několik P-CSCF serverů kvůli škálovatelnosti a redundanci. V tom případě každý P-CSCF obsluhuje několik IMS zařízení, v závislosti na jeho kapacitě [4], [2].

I-CSCF

I-CSCF je komponent umístěný na okraji administrativní domény. Adresa I-CSCF je uvedena v DNS záznamech domény. I-CSCF poskytuje komunikační rozhraní pro SLF a HSS, s kterými komunikuje pomocí protokolu DIAMETER. I-CSCF odešle DIAMETER dotaz do HSS, který zkontroluje zda se uživatel může zaregistrovat v dané síti a poskytne adresu S-CSCF, ke kterému může být uživatel registrován. Po určení S-CSCF serveru je možné směrovat i bez použití I-CSCF.
I-CSCF může volitelně šifrovat části SIP zpráv, které obsahují citlivé informace o doméně, například počet serverů, jejich DNS jména nebo jejich kapacitu. Tato funkce se označuje jako THIG (Topology Hiding Inter-network Gateway). Pokud je THIG aktivní, I-CSCF zůstává zapojena v signalizaci hovoru i po zjištění adresy S-CSCF.
Síť může obsahovat několik I-CSCF serverů kvůli škálovatelnosti a redundanci [4],[2].

S-CSCF

S-CSCF je centrální uzel CSCF, který řídí relace a funguje jako SIP registrátor. Udržuje vazbu mezi lokací uživatele (např. IP adresa zařízení, ke kterému je uživatel

přihlášen) a SIP public user identity.
Stejně jako I-CSCF i S-CSCF implementuje rozhraní komunikující pomocí protokolu DIAMETER s HSS, které plní následující funkce:

- Získání autentizačních vektorů uživatele pokoušejícího se o registraci k IMS síti. S-CSCF následně využije tyto vektory k autentizaci uživatele.
- Získání uživatelského profilu z HSS, který zahrnuje servisní profil– sadu spouštěčů, které mohou směrovat SIP zprávy přes jeden nebo více aplikačních serverů.
- Informování HSS o přiděleném serveru konkrétnímu uživateli po dobu trvání registrace.

Jednou z hlavních funkcí S-CSCF je poskytování směrovacích služeb pomocí protokolu SIP. Pokud uživatel zadá jako identifikátor telefonní číslo namísto SIP URI (Uniform Resource Identifier), S-CSCF poskytne překladové služby obvykle založené na DNS E.164 Number Translation.
S-CSCF také uplatňuje PDF pro konkrétního uživatele, který například nemusí být oprávněn zahájit určité typy relací.
Síť může obsahovat několik S-CSCF. Každý S-CSCF obsluhuje větší množství IMS zařízení, v závislosti na jeho kapacitě [4], [2].

1.2.2 Home Subscriber Server

HSS slouží jako databáze účastnických profilů a centrální úložiště informací o účastnících. Udržuje všechny informace o účastnících, které jsou nezbytné pro navázání relací mezi uživateli a pro poskytování služeb. HSS obsahuje informace o:

- registrovaných účastnících (jméno, adresa, služby, atd.),
- účastnické preference (informace o blokování, nastavení přeposílání),
- polohu účastníka,
- informace o službách.

IMS síť může obsahovat více HSS v případě, že množství uživatelů je příliš vysoké na zpracování jedním HSS. V případě většího množství HSS, jsou všechna data týkající se jednoho konkrétního uživatele uložena v jednom konkrétním HSS. Sítě s větším počtem HSS potřebují SLF (Subscription Locator Function).
SLF je jednoduchá databáze, která mapuje adresy uživatelů na konkrétní HSS. Pomocí dotazu protokolu DIAMETER získá I-CSCF konkrétní HSS, který obsahuje všechny informace související s tímto uživatelem.
Pro komunikaci s CSCF je využíváno rozhraní Sh a protokol DIAMETER. HSS může spojit data dohromady pod jedním rozhraním, místo aby nahrazoval již existující úložiště [4], [2].

1.2.3 MRFC a MRFP

Většina tzv. next-generation služeb vyžaduje zpracování medií, které mohou být poskytovány univerzálními mediálními servery. Tyto mediální servery umožňují přehrávat zvukové výzvy, konvertovat text na řeč, míchat zvuk v konferenčních hovorech, atd.
IMS obsahuje komponentu MRFC, která přijímá instrukce z aplikační služby, následně je předá MRFP, kde jsou na základě poskytnutých instrukcí zpracovávány mediální toky. MRFC řídí MRFP pomocí protokolu H.248/Megaco. MRFC může také provádět účtování MRFP, řízení konferenčních hovorů a řízení účastníkova roamingu [2].

1.2.4 Domácí a navštívené sítě

Podobně jako GSM (Global System for Mobile Communications) a GPRS (General Packet Radio Service) sítě i IMS rozlišuje domácí a navštívené sítě. Když je používána infrastruktura poskytovaná provozovatelem sítě, jedná se domácí síť. Pokud je IMS zařízení mimo oblast pokrytí domácí sítě (typicky návštěva jiného státu) je využívána infrastruktura jiného operátora a jedná se o navštívenou síť. V případě, že chce uživatel použít navštívenou síť, musí navštívený operátor mít podepsánu dohodu s domácím provozovatelem sítě. Obsahem těchto dohod jsou různé aspekty služby poskytované uživateli, jako je například cena, kvalita služby, nebo způsob vyměňování účetních informací.
Většina uzlů v IMS architektuře je vždy umístěna v domácí síti, výjimkou je P-CSCF, který je možné umístit buď do domácí sítě, nebo do navštívené sítě [4].

1.3 Identifikace v IMS

Důležitou funkcí každé sítě je schopnost operátora jednoznačně identifikovat uživatele tak, aby hovory mohly být směrovány na konkrétního uživatele. V PSTN jsou uživatelé identifikováni telefonním číslem. V IMS rozlišujeme dva identifikátory:

- public user identities,
- private user identities.

1.3.1 Public user identities

Uživateli je možné přidělit jednu, nebo více public user identities. Za přidělování těchto identifikátorů je zodpovědný domácí operátor konkrétního uživatele. public user Identity, může být buď SIP URI (definováno v RFC 3261) nebo TEL URI (definováno v RFC 3966). Pokud public user identity obsahuje SIP URI, má obvykle

formát `sip:student@vutbr.cz`. Je možné zahrnout telefonní číslo do SIP URI ve formátu `sip: +1-212-555-0293@operator.com; user=phone`. Protokol SIP při registraci vyžaduje formát typu SIP URI. Není možné zaregistrovat TEL URI v SIP, ale je možné zaregistroval SIP URI, které obsahuje telefonní číslo.
TEL URI jsou potřeba k uskutečnění hovoru z IMS zařízení do PSTN telefonu, protože TEL URI v PSTN jsou reprezentována pouze číslicemi. TEL URI jsou také potřebné pokud chce uživatel PSTN komunikovat s uživatelem IMS, protože PSTN zařízení obvykle umožňují vytáčet pouze číslice. Předpokládáme, že operátoři přidělí alespoň jedno SIP URI a jednoho TEL URI na jednoho uživatele. Existují důvody pro přidělení více než jedné public user identity jednomu uživateli, například schopnost rozlišovat osobní, pracovní a rodinné public user identity.
Každá public user identity, musí být v případě jejího používání registrována pomocí žádosti SIP REGISTER. IMS umožňuje zaregistrovat několik public user identit v jedné zprávě [4].

1.3.2 Private user identities

Každému účastníkovi IMS je přidělena private user identity. Na rozdíl od public user identity, se neskládá z identifikátorů SIP URI nebo TEL URI, místo toho mají formát NAI(Network Access Identifier, specifikovaný v RFC 2486). Formát NAI je například `student@vutbr.cz`.
Na rozdíl od public user identities, nejsou používány pro směrování SIP požadavků, ale používají se výhradně pro účely identifikace a ověření uživatele [4].

1.4 Protokoly využívané v IMS

Následující kapitola obsahuje popis nejdůležitějších protokolů v IMS.

1.4.1 SIP

SIP (Session Initial Protocol), je protokol využívaný pro přenos signalizace a řízení multimediálních relací (video, hlas, instant messaging, online hry,..) v počítačových sítích.
Byl vyvinut skupinou IETF (Internet Engineering Task Force). První specifikace byly vydány v roce 1996 jako součást RFC 2543, specifikace byly následně upravovány a poslední verze je popsána v RFC 3261.
Pro vytvoření a řízení multimediální relace musí SIP zajistit:

- Registraci uživatele – koncové body oznamují SIP proxy jejich umístění v síti.

- Dostupnost uživatele – zjištění dostupnosti uživatele (dostupný, má obsazeno, přesměrování atd.) a jeho schopnosti navázat relaci.
- Uživatelské možnosti – zjištění jaké jsou možnosti účastníka relace (typ kodedu, maximální přenosová rychlost atd.).
- Nastavení relace – zvonění a dohoda atributů relace pro účastníky relace.
- Řízení relace – navázání spojení, reakce na různé události během hovoru a ukončení spojení.

Jedná se o textově orientovaný protokol fungující na modelu žádost/odpověď. Protokol SIP může být používám v IPv4 i IPv6 sítích a využívá transportní protokoly TCP nebo UDP. Nejčastěji je realizován v IPv4 sítích s transportním protokolem UDP.
Každá koncová stanice má přidělený svůj identifikátor SIP URI. Telefony, protože mají většinou numerické klávesy, jsou zodpovědné za překlad čísla na SIP URI, kdy například číslo 1001 bude přeloženo na SIP URI "sip:1001@vutbr.cz"[5],[6].

1.4.2 SDP

SDP (Session Description Protokol), je protokol určený k popisu vlastností multimediálních relací. Protokol nepřenáší vlastní uživatelská data, ale používá se k vyjednávání parametrů relace, například typ kodedu, typ média (audio,video,...), transportní protokol, IP adresy a porty koncových bodů multimediální relace, ...
Často bývá implementován s dalšími protokoly, typicky SIP, SAP, RTP,... Detailně je popsán ve specifikaci RFC 4566.[7]

1.4.3 RTP

RTP (Real-time Transport Protocol) je síťový protokol, který určuje způsob jakým programy řídí přenos multimediálních dat v reálném čase. První verze byla publikována v roce 1996 jako standard RFC 1889, později nahrazeným RFC 3550. Definuje standardizovaný formát paketů pro doručování zvuku a obrazu po síti.
Používá se ve spolupráci s RTCP (Real-time Transport Control Protocol), kvůli syncronizaci více toků médií a zachování kvality službeb (QoS). Důležitou funkcí je číslování sekvencí, které umožňuje sledování doručení paketů. RTP data jsou nejčastěji přenášena pomocí protokolu UDP. Služby pracující s RTP zahrnují určení užitečného zatížení, číslování sekvencí, časové razítkování a sledování přenosu [8].

1.4.4 DIAMETER

Protokol DIAMETER byl vytvořen za účelem tzv. AAA, tedy authentication, authorization a accounting pro různé přístupové aplikace, ať už v lokálním, nebo v roaming režimu. Byl vyvinut z protoklu RADIUS. Patří do aplikační vrstvy síťových protokolů. [9]

1.5 Registrace a sestavení spojení

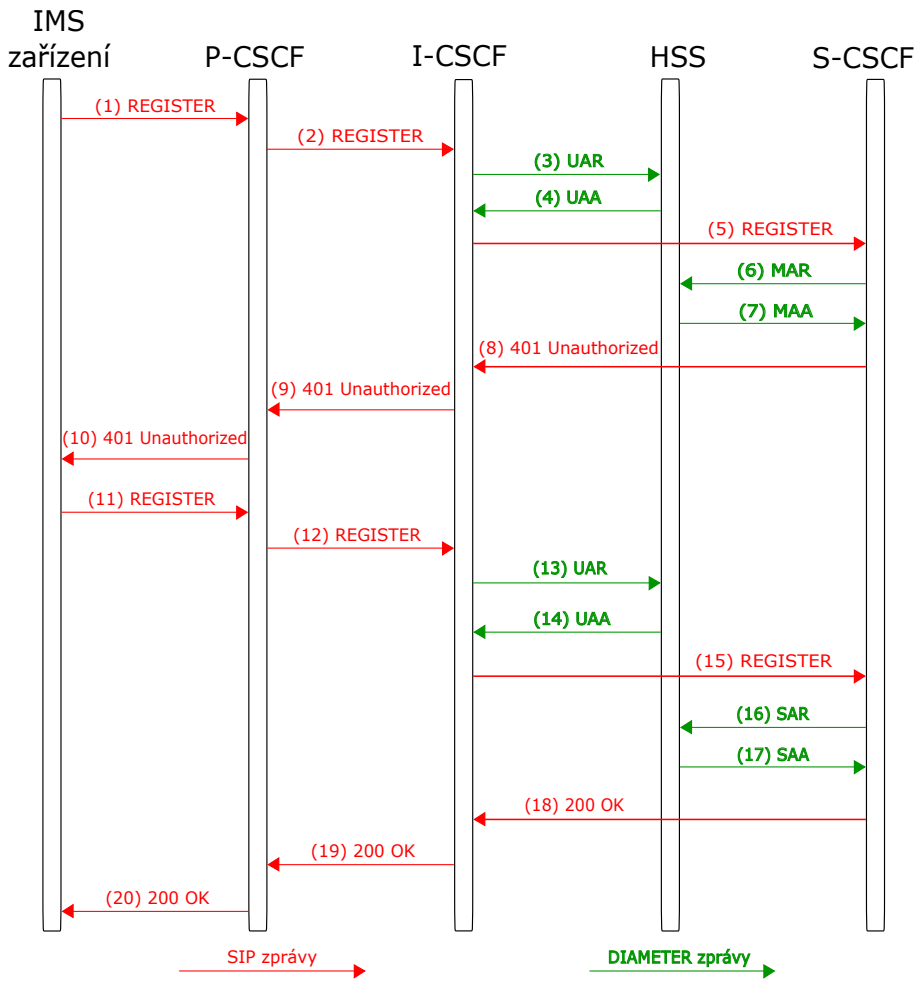
V následující kapitole je popsána registrace uživatele k IMS síti a následné sestavení spojení.

1.5.1 Registrace

Registrace na úrovni IMS je postup kdy uživatel požaduje oprávnění k využívání služeb v IMS síti. IMS síť uživatele autentizuje a autorizuje pro přístup k síti. Registrace je prováděna požadavkem SIP REGISTER. Registrace je v rámci IMS povinná a dokud neproběhne, nemůže uživatel vytvořit relaci. IMS registrace využívá protokol SIP pro směrování mezi CSCF a IMS zařízením a protokol DIAMETER, který je využíván pro komunikaci CSCF s HSS. IMS registrace zajišťuje splnění následujících požadavků:

- Uživatel naváže public user identity na kontaktní adresu, což hlavní účel žádosti SIP REGISTER.
- V domácí síti je autentizován uživatel.
- Uživatel autentizuje domácí síť.
- Domácí síť autorizuje SIP registraci a použití zdrojů, kterými IMS disponuje.
- Když je P-CSCF umístěn v navštívené síti, domácí síť ověří zda existuje dohoda o roamingu mezi domovskou a navštívenou sítí a autorizuje použití P-CSCF.
- Domácí síť informuje uživatele o dalších možných identitách, které domácí provozovatel sítě přidělil výhradně tomuto uživateli.
- IMS zařízení a P-CSCF vyjednávají bezpečnostní mechanismus, který bude v systému použit pro následnou signalizaci.
- P-CSCF a IMS zařízení vytvářejí sadu bezpečnostních asociací, které chrání integritu SIP zpráv odeslaných mezi P-CSCF a IMS zařízením.
- IMS a P-CSCF si navzájem vyměňují algoritmy použité prokompresi SIP zpráv.

Na obrázku 1.4, vidíme signálové toky SIP a DIAMETER zpráv, při registraci uživatele k IMS síti. První zpráva (1) REGISTER je poslána z IMS zařízení na P-CSCF,



Obr. 1.4: Tok zpráv v průběhu registrace uživatele k síti IMS

který může být lokalizován buď v navštívené, nebo domácí síti. Obecně tedy P-CSCF nemusí být v domácí síti, ale musí najít vstupní bod do domácí sítě podle DNS postupů uvedených v RFC 3263. Pomocí těchto postupů je serveru P-CSCF poskytnuta SIP URI I-CSCF serveru v domácí síti. P-CSCF vloží do zprávy položku záhlaví `P-Visited-Network-ID`, která obsahuje identifikátor sítě s použitým P-CSCF.

Domácí síť pomocí tohoto pole záhlaví ověřuje roamingové dohody mezi domovskou a navštívenou sítí. P-CSCF do SIP zprávy také vloží položku záhlaví `Route` s vlastním identifikátorem SIP URI, z důvodu požadavku domácí sítě o přeposílání všech SIP požadavků prostřednictvím konkrétního P-CSCF. Případně P-CSCF přepošle SIP zprávu (2) REGISTER do I-CSCF v domácí síti.

I-CSCF neuchovává stav registrace. I-CSCF nezaznamenávají, zda je S-CSCF přidělen uživateli a jaká je případně jeho IP adresa.

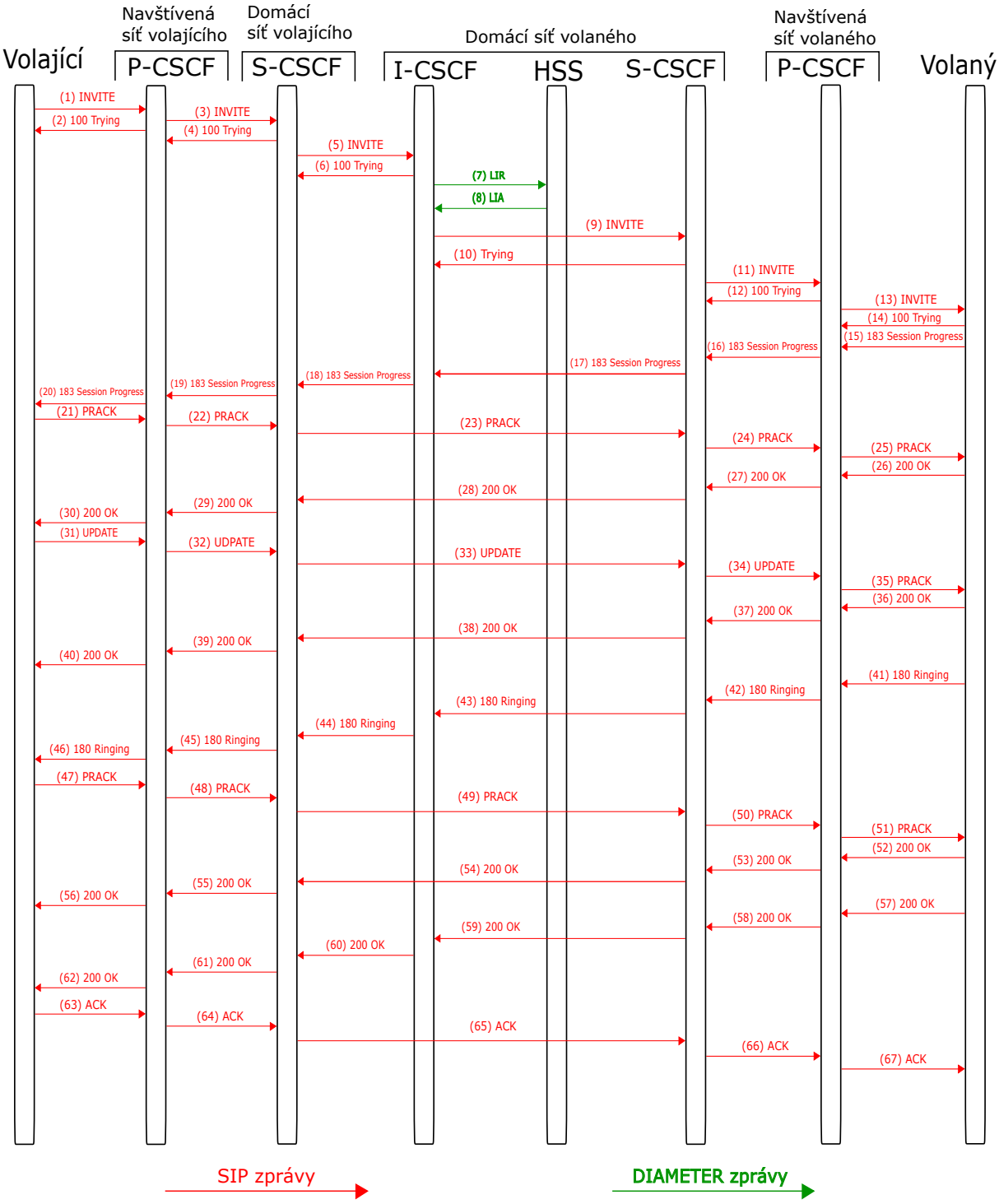
Prvním krokem je zjistit, zda již existuje S-CSCF přidělený uživateli. I-CSCF odešle pomocí protokolu DIAMETER zprávu (3) UAR (User Authentication Request) do

HSS. Informace ve zprávě (3) UAR zahrnují public user identity, private user identity a identifikátor navštívených sítí, které jsou získány ze SIP žádosti REGISTER. HSS autorizuje uživatele k využití roamingu v navštívené síti a potvrdí, že private user identity je přidělena public user identity dle požadavků registrace.
HSS následně odpoví pomocí protokolu DIAMETER zprávou (4) UAA (User Authentication Answer), která v případě že uživateli již byl přidělen S-CSCF, zahrunje SIP URI dříve přiděleného S-CSCF. Pokud je to první registrace (např. poté, co uživatel zapnul IMS zařízení), s největší pravděpodobností nebude uživateli přidělený žádný S-CSCF. V tom případě HSS vrací informace, které jsou pro I-CSCF hlavním vstupem při výběru S-CSCF. Tyto informace jsou rozdělené na povinné a volitelné a jsou použity při volbě cílového S-CSCF. Poté I-CSCF posílá SIP REGISTER požadavek na vybraný S-CSCF.
S-CSCF přijme požadavek (5) REGISTRACE a autentizuje uživatele. Počáteční registrace jsou vždy autentizovány v IMS. Ostatní registrace mohou, ale nemusí být autentizovány v závislosti na bezpečnostních požadavcích. V IMS jsou autentizovány pouze REGISTER žádosti. Další SIP požadavky, jako například INVITE, autentizovány v IMS nejsou, případně jsou autentizovány volitelně za účelem zvýšení bezpečnosti.
S-CSCF pak kontaktuje HSS, protože potřebuje získat autentizační data, aby mohl provést autentizaci pro konkrétního uživatele. Následně musí uložit URI S-CSCF do HSS, aby jakýkoliv další dotaz na HSS pro stejného uživatele vrátil směrovací informace konkrétního S-CSCF. Za tímto účelem S-CSCF odešle DIAMETER požadavek (6) MAR (Multimedia-Auth-Request). HSS si po přijmutí požadavku uloží SIP URI a odpoví DIAMETER zprávou (7) MAA (Multimedia-Auth-Answer).
V IMS jsou uživatelé autentizováni pomocí S-CSCF na základě ověřovacích dat poskytnutých HSS serverem. Tato ověřovací data jsou známé jako autentizační vektory. HSS má jeden nebo více ověřovacích vektorů, které odešle zprávou protokolu DIAMETER (8) MAA. Následně, S-CSCF vytvoří odpověď SIP (9) 401 (Unauthorized), která zahrnuje výzvu k autentizaci obsaženou v poli záhlaví `WWW-Authenticate`, na kterou se od IMS zařízení očekává odpověď.
Odpověď SIP 401 (Unauthorized) je přeposlána přes I-CSCF a P-CSCF k IMS zařízení. Jakmile IMS zařízení přijme odpověď (10) SIP 401 (Unauthorized), detekuje pole záhlaví `WWW-Authenticate` a vytvoří vhodnou odpověď na tuto výzvu. Odpověď na výzvu k autentizaci, se nachází v nové SIP žádosti REGISTER. Podoba autentizačních dat (credentials) závisí na konkrétní IMS síti.
IMS zařízení pošle nový SIP požadavek (11) REGISTER do P-CSCF, který provádí stejnou operaci jako u prvního požadavku REGISTER, tedy najde I-CSCF předá mu (12) REGISTER zprávu. I-CSCF odešle novou zprávu DIAMETER (13) UAR do HSS, ze stejných důvodů jako předcházející zprávu (3) UAR. Tentokrát zpráva

Diameter UAA (14) zahrnuje směrovací informaci SIP URI S-CSCF serveru přiděleného uživateli. HSS si uložil toto URI, když přijal zprávu DIAMETER (6) MAR. Proto bez ohledu na to, zda je I-CSCF ten kterým přišel první požadavek REGISTER, druhý požadavek REGISTER je směrován původnímu S-CSCF.
S-CSCF přijme žádost REGISTER (15), která obsahuje přihlašovací údaje uživatele. S-CSCF poté tyto údaje ověřuje proti autentizačním vektorům poskytnutým HSS ve zprávě protokolu DIAMETER (7) MAA. Pokud je ověření úspěšné, pak S-CSCF odešle DIAMETER zprávu (16) SAR (Server Assignment Request) do HSS, kterou sdělí, že uživatel je nyní zaregistrován a extrahuje si uživatelský profil pomocí DIAMETER zprávy (17) SAA (Server Assignment Answer).
Profil uživatele je důležitá informace, která mimo jiné zahrnuje množinu všech public user identit přidělených k private user identitám.
V této fázi S-CSCF uloží kontaktní URI pro uživatele, obsažené v poli záhlaví `Contact` požadavku SIP REGISTER. Také uloží seznam URI obsažený v poli záhlaví `Route`. Tento seznam vždy obsahuje URI P-CSCF serveru a může volitelně zahrnovat URI I-CSCF serveru. Později S-CSCF směruje SIP požadavky adresované uživateli prostřednictvím seznamu URI obsaženém v poli záhlaví `Route`.
Poté S-CSCF odešle odpověď (18) 200 (OK) na požadavek REGISTER. Odpověď 200 (OK) zahrnuje pole záhlaví `P-Associated-URI`, které obsahuje seznam URI přidělených uživateli. Obsahuje také pole záhlaví `Service-Route`, které zahrnuje seznam URI CSCF serverů. Budoucí SIP požadavky, které odešle IMS zařízení, budou směrovány přes tyto CSCF servery. V IMS hodnota pole záhlaví `Service-Route` vždy obsahuje adresu S-CSCF přiděleného uživateli a může také obsahovat adresu I-CSCF v domácí síti.
Odezva (19) 200 (OK) prochází stejným I-CSCF a P-CSCF jako žádost REGISTER. Nakonec IMS zařízení dostane odpověď (20) 200 (OK). V této fázi je registrační postup dokončen. IMS zařízení je registrováno v IMS síti po dobu uvedenou v parametru `expires` v poli `Contact` [4].

1.5.2 Sestavení spojení

Předpokládáme, že oba uživatelé provádí roaming mimo jejich domácí sítě. Proto jsou v obrázku dvě různé navštívené sítě. Uvažujeme, že každý z uživatelů má jiný obchodní vztah se svým příslušným operátorem a tudíž jsou v obrázku dvě odlišné domácí sítě. Předpokládejme, že P-CSCF je umístěn v navštívené síti. Popisovaný scénář je nejúplnější a nejkomplikovanější možný případ sestavení hovoru.
Signalizace a mediální přenos, jsou odděleny. Signalizace prochází přes CSCF komponenty, ale média jsou přenášeny přímo mezi oběma účastníky hovoru.



Obr. 1.5: Tok zpráv v průběhu sestavení hovoru v případě, že jsou oba účastníci mimo svoji domácí síť

Na obrázku 1.5, jsou zobrazeny signálové toky SIP a DIAMETER zpráv, při sestavení hovoru, kde oba účastníci jsou mimo svoji domácí síť.
Navázaní spojení začíná odesláním SIP požadavku (1) INVITE, v které specifikujeme public user identity volaného pomocí položky záhlaví `Request-URI`. Další důležitou položkou záhlaví je `Via`, která obsahuje IP adresu a číslo portu, na které bude volající přijímat odpovědi na požadavek INVITE a obsahuje informace o použitém protokolu transportní vrstvy. Polem záhlaví `P-Preferred-Identity`, je specifikováno kterou public user identity chce uživatel použít, toto pole je nepovinné.
Další důležité pole je `Route`, které obsahuje P-CSCF server navštívené sítě a S-CSCF domácí sítě, získané při registraci zařízení. Každý hovor má přiřazený jedinečný identifikátor `Call-ID` a položky záhlaví `From` a `To`, které specifikují public user identity volajícího (`From`) a volaného (`To`). Kromě položek záhlaví, zpráva INVITE obsahuje i tělo, v kterém bývá obsažena zpráva protokolu SDP.
P-CSC přijme SIP požadavek INVITE a ověří správnou přítomnost pole záhlaví `Service-Route`, získanou ve zprávě 200 OK při registraci zařízení. Pokud požadavek INVITE neobsahuje požadovanou hodnotu `Service-Route`, P-CSCF buď nahradí záhlaví `Service-Route` standartní hodnotou pro daný P-CSCF, nebo odešle IMS zařízení odpověď 400 (Bad Request) a požadavek INVITE zahodí.
Následně P-CSCF porovná nabídku SDP s místní politikou sítě, podle které můžou být některá média zakázána. Místní politika souvisí s potřebami operátorů, topologií sítě, účtovacími modely, dohodami ohledně roamingu atd. Pokud P-CSCF zjistí parametr média, který je rozdílný od současné místní politiky, vygeneruje odpověď 488 (Not Acceptable Here), která obsahuje tělo SDP a požadavek INVITE zahodí.
Poté P-CSCF hledá, zda přijatá SIP zpráva INVITE obsahuje položku záhlaví `P-Preferred-Identity`. Pakliže obsahuje, P-CSCF ověří, zda hodnota v této hlavičce odpovídá implicitně nebo explicitně registrované public user identity, které vybírá z identit registrovaných uživatelem. Položku záhlaví `P-Preferred-Identity` nahradí položkou záhlaví `P-Asserted-Identity`, která je jedna z registrovaných public user identity uživatele (obvykle obsažená v `P-Preferred-Identity`).
Poté P-CSCF upraví záhlaví, která souvisí se zabezpečením (daná síť může, nebo nemusí používat nějkou formu zabezpečení) a vloží hlavičku týkající se účtování hovoru. V rámci zaznamenání trasy, modifikuje položky záhlaví `Via`, `Route` a `Max-Forwards`.
S-CSCF přidělený volajícímu obdrží požadavek (3) INVITE a pomocí položky záhlaví `P-Asserted-Identity` identifikuje uživatele, který odeslal požadavek INVITE. Při registraci si S-CSCF stáhl uživatelský profil, v kterém jsou mimo jiné obsažena i kritéria filtru obsahující spouštěče určující aplikační server, který bude aktivní při použité službě. S-CSCF vyhodnocuje kritéria filtru, pro počáteční zprávu INVITE nebo SUBSCRIBE z důvodu, že vytvářejí dialog.
S-CSCF, stejně jako P-CSCF, má nastavenou SDP politiku, kterou získává z HSS

a přizpůsobuje ji přímo konkrétnímu uživateli. Pokud požadavek nesouhlasí s politikou S-CSCF, je odeslána zpráva 488 (Not Acceptable Here) směrem k volanému a požadavek INVITE je zahozen.
Následně je třeba najít I-CSCF v domovské síti volaného, jehož adresu nalezne pomocí řady DNS dotazů.
I-CSCF příjme požadavek INVITE (5) a neví ke kterému S-CSCF je volaný uživatel registrován. Pro zjištění konkrétního S-CSCF serveru musí být dotázán HSS, který si během registrace uložil adresu konkrétního S-CSCF pro daného uživatele. Dotaz zprávou LIR (7) (Location-Information-Request) obsahuje hodnotu z `Request-URI` a je realizován protokolem DIAMETER. HSS obdrží žádost LIR, následně ověří AVP (Attribute-Value Pair) pro public user identity a získá uloženou adresu S-CSCF serveru, kterou odešle do I-CSCF pomocí DIAMETER zprávy (8) LIA (Location-Information-Answer).
I-CSCF v této fázi nemodifikuje ani nepřidává žádné pole záhlaví SIP, kromě polí směrování (`Route`, `Max-Forward`, `Via` atd.). Následně I-CSCF pošle zprávu (9) INVITE do přiřazeného S-CSCF.
S-CSCF v domácí síti volaného obdrží žádost (9) INVITE. Nejdříve identifikuje volaného pomocí `Request-URI`, poté vyhodnotí počáteční kritéria filtru volaného.
S-CSCF vytvoří nové `Request-URI` s polem záhlaví `Contact` získaným při registraci volaného uživatele. Také nastaví hodnotu `Route`, získanou během registrace v položce záhlaví `Path`. Kvůli možnosti registrace jednoho uživatele pomocí více public user identit, je využíváno pole záhlaví `P-Called-Party-ID`, jehož hodnota je nastavena na původní `Request-URI` čímž určíme public user identity volajícího. S-CSCF odešle zprávu (11) INVITE P-CSCF serveru získaným při registraci volaného.
P-CSCF volaného příjme požadavek (11) INVITE. Nemusí již provádět rozhodnutí o směrování, protože pole záhlaví zprávy (11) INVITE `Request-URI` již obsahuje SIP URI volaného IMS zařízení. P-CSCF musí znát public user identity volaného, kterou extrahuje z položky záhlaví `P-Called-Party-ID` požadavku INVITE, za účelem nalezení správného typu zabezpečení při komunikaci s IMS zařízením volaného. P-CSCF poté přidá vlastní SIP URI do položky záhlaví `Record-Route`, protože přes tento prvek prochází signalizace vždy. P-CSCF zkontroluje položku záhlaví `Privacy` a pokud je potřeba, tak odstraní položku záhlaví `P-Asserted-Identity`. Pokud není přítomna položka záhlaví `Privacy`, nebo je nastavena na hodnotu `none`, znamená to že žádné akce na podporu soukromí nejsou vyžadovány. Nakonec je požadavek (13) INVITE směrován do IMS zařízení volaného.
Požadavek (13) INVITE je přijat IMS zařízením a obsahuje nabídku SDP generovanou v IMS zařízení volajícího. Nabídka SDP obsahuje IP adresu a čísla portů, kde volající chce přijímat mediální streamy, požadované a podporované kodeky pro každý z mediálních toků atd.

Zpráva (13) INVITE, může obsahovat pole záhlaví `Require`, jeho existence značí nutnost odpovědi zprávou 183 Session Progress ze strany volaného, která obsahuje SDP odpověď. Odpověď SDP obsahuje médiální streamy a kodeky, které je volající schopen přijmout.
IMS zařízení zjistí přítomnost položky záhlaví `P-Asserted-Identity` a případně z ní extrahuje identitu volajícího. IMS zařízení pomocí hodnoty v `P-Called-Party-ID` zjistí komu z registrovaných public user identit je požadavek adresován. IMS zařízení na straně volaného odpoví provizorní zprávou 183 Session Progress obsahující zprávu SDP protokolu.
Odpověď 183 Session Progress prochází postupně stejnou trasou, jako předchozí žádost INVITE směrem k IMS zařízení volajícího. Jakmile P-CSCF v navštívené síti volajícího příjme odpověď (15) 183 Session Progress, ověří zda je formulována správně, což zahrnuje kontrolu položky záhlaví `Via` a `Record-Route`, tak aby odpovídali očekávaným hodnotám. To zamezí tzv. spoofování IMS zařízení, kdy IMS zařízení záměrně nepřidá S-CSCF nebo P-CSCF do plánované trasy, což má za následek to, že se uživatelé v tomto hovoru vyhnou účtování. Když P-CSCF tento pokus o podvod odhalí, tak odpověď zahodí, nebo přepíše hodnoty na správné.
Když P-CSCF přijal požadavek INVITE, uložil hodnotu `P-Called-Party-ID`. Hodnota tohoto pole záhlaví označuje public user identity volaného. P-CSCF do přijaté zprávy 183 Session Progress vloží pole záhlaví `P-Asserted-Identity`, jehož hodnota je stejná jako v `P-Called-Party-ID` v předchozím požadavku INVITE. Následně P-CSCF odešle odpověď (16) 183 Session Progress do S-CSCF v domácí síti volaného.
S-CSCF odstraní pole záhlaví `P-Access-Network-Info` a předá zprávu (17) 183 Session Progress I-CSCF serveru.
I-CSCF neprovádí žádné změny, jen zprávu předá S-CSCF v domácí síti volajícího. S-CSCF přijme zprávu (18) 183 Session Progress a pokud to politika nastavení soukromí vyžaduje, tak odstraní položku záhlaví `P-Asserted-Identity`. S-CSCF odešle zprávu (19) 183 Session Progress P-CSCF v navštívené síti volajícího, který zprávu (20) 183 Session Progress pouze předá k IMS zařízení volajícího.
IMS zařízení volajícího příjme zprávu (20) 183 Session Progress, u které je důležitý obsah zprávy protokolu SPD obsahující IP adresu volaného zařízení, informaci zda volaný přijal relaci s podmínkami poskytnutými volajícím i sítí a seznam podporovaných kodeků. Snaha je aby bylo sjednáno co nejméně kodeků. Když jich je více než jeden, tak si IMS zařízení rezervuje šířku pásma nejnáročnějšího kodeku. To však vede k neefektivnímu využití správy zdrojů a potenciálnímu uživateli bude účtována šířka pásma, která nemusí být ve skutečnosti využita. Z tohoto důvodu IMS zařízení omezí počet kodeků na pouhý jeden v rámci jednoho mediálního streamu. IMS zařízení volajícího proto vytváří novou nabídku protokolu SDP, jejímž

jediným rozdílem oproti původní obsažené v těle SIP zprávy INVITE je odstranění nadbytečných kodeků.
Odpovědí na zprávu 183 (Session Progress), je zpráva (21) PRACK, obsahující novou nabídku protokolu SPD. Vygenerováním požadavku PRACK, IMS zařízení spustí mechanismus rezervace zdrojů. Cesta SIP zprávy PRACK k IMS zařízení volaného je určena polem záhlaví `Route`.
Jakmile IMS zařízení obdrží zprávu (25) PRACK, vygeneruje odpověď (26) 200 OK, obsahující odpověď na SPD žádost. Protože se jedná o druhou výměnu nabídka–odpověď, tak odpověď slouží pouze pro potvrzení mediálních streamů a kodeků relace.
SIP odpověď (26) 200 OK odpověď je poslána volajícímu stejnou cestou jako SIP zpráva PRACK.
IMS zařízení volajícího, které příjme odpověď (30) 200 OK rezervuje požadované zdroje a odešle IMS zařízení volaného SIP požadavek (31) UPDATE stejnou cestou jako zpráva 200 OK. Požadavek (31) UPDATE obsahuje další SDP zprávu s potvrzením rezervace zdrojů na straně volaného.
IMS zařízení volaného obdrží požadavek (35) UPDATE, na který reaguje odpovědí (36) 200 OK zahrnující odpověď SDP, v které jsou finálně dohodnuty potvrzené kodeky pro aktuální relaci a zároveň IMS zařízení volaného rezervuje požadované zdroje. Odpověď 200 OK je odeslána stejnou cestou, jako UPDATE na IMS zařízení volaného.
IMS zařízení volaného začne zvonit, což znamená, že proces rezervace zdrojů je kompletní a odpověď 200 (OK) byla odeslána k IMS zařízení volajícího. IMS zařízení volaného vygeneruje provizorní odpověď (41) 180 Ringing a odešle ji k IMS zařízení volajícího stejnou cestou, jakou byla přijata zpráva INVITE.
Jakmile IMS zařízení volajícího obdrží odpověď (46) 180 Ringing, generuje požadavek (47) PRACK a odešle jej volanému stejnou cestou jako předchozí PRACK zprávu, která tentokrát neobsahuje zprávu protokolu SDP.
IMS zařízení volaného příjme žádost (51) PRACK a odpoví SIP zprávou (52) 200 OK stejnou cestou jako předchozí odpověď 200 OK.
Jakmile IMS zařízení volaného příjme relaci, tak odešle zprávu zprávu (57) 200 OK, kterou se uzavře proces navázání relace.
Zprávu (62) 200 OK příjme IMS zařízení volaného, které začíná generovat mediální provoz a zároveň vygeneruje SIP požadavek (63) ACK, kterým potvrzuje správné přijetí SIP odpovědi 200 OK.
Po příjmu zprávy (67) ACK IMS zařízením volaného, je proces navázání relace úspěšně ukončen. Nyní mohou oba účastníci hovoru mezi sebou generovat mediální streamy.

1.6 Bezpečnost IMS

Zajištění bezpečnosti v IMS sítích je důležité, protože se jedná o komplexní systém zahrnující širokou škálu služeb, protokolů a komponent. Tato komplexnost zvyšuje počet zranitelných míst a rizik pro uživatele IMS zařízení a poskytovatele služeb fungujících na internetu.

1.6.1 Bezpečnostní hrozby

Možnost autentizace uživatele

Ideální je, aby byly uživatelé autentizováni při každém použití IMS zařízení, což není pro uživatele pohodlné a může to vést k většímu vytížení sítě. Uživatelé GSM používají k autentizaci soukromý klíč sdílený mezi sítí operátora a uživatelem. Soukromý klíč je standardně uložen v modulu SIM (Subscriber Identity Module), který je vyjímatelný a lze jej použít v různých typech koncových zařízení.
Pevné VoIP telefony a softwarové telefony nejsou vybaveny žádným SIM modulem, kde by mohl být uložen soukromý klíč. V IMS se proto autentizace provádí pomocí uživatelského jména a hesla, tato implementace je zranitelná útoky typu brute force nebo replay.

Útoky na hranicích IMS sítě

Nejvíce zranitelné jsou v tomto případě IMS brány, jelikož jde o vstupní komponenty z veřejné sítě. Komponenty jako jsou SGW (Signal Gateway), MG (Media Gateway) a MGCF (Media Gateway Control Function) mohou provádět konverzi informací do jiného formátu. Při těchto konverzích je důležité kontrolovat integritu dat z důvodu potenciálního zneužití těchto konverzí útočníkem, který k datům může pomocí reverzní transformace přidat škodlivý skript. Tyto skripty mohou následně poškodit IMS síť.

Odmítnutí služeb (DoS)

Jedná se o útoky s cílem snížení celkové přenosové kapacity IMS sítě, nebo výpadku služby. Příkladem DoS útoku je tzv. INVITE flood, kdy útočník odešle velké množství SIP požadavků INVITE, které jsou uměle vytvořeny za účelem vytížení P-CSCF serveru v IMS síti. Případně lze pro tyto záplavové typy útoků využít jiných zpráv SIP (například REGISTER, SUBSCRIBE a další). P-CSCF musí nějakým způsobem reagovat na tyto falešné přijaté zprávy, což může vést ke snížení kapacity nebo k výpadku služby.
Další ohroženou entitou jsou uživatelské IMS zařízení. IMS zařízení uživatele se

může stát potenciálním šiřitelem záplavových útoků, pokud se do něj dostane nějaký škodlivý kód, který tyto útoky realizuje bez vědomí uživatele. Případně mohou být uživatelské zařízení použita jako server šíření škodlivého obsahu, což může být ještě větším problémem než samotné DoS útoky.

Zjištění topologie sítě

Mnoho poskytovatelů IMS sítí dává přednost jejich vlastní síťové architektuře a možnosti služeb důvěrné a soukromé povahy. SIP ve výchozím nastavení prostřednictvím položek záhlaví (`Via`, `Route`, `Record-Route` a `Path`) odhalí mnoho interních informací o topologii síti. Skrytí vnitřní topologie sítě zajišťuje technologie THIG, která tyto informace šifruje. THIG nemusí být ve všech sítích povolen, nebo může být špatně implementován a informace o vnitřní topologii sítě bude možné odhalit.

Toll fraud

Na základě implementace protokolu SIP, poskytuje IMS oddělený mechanismus signální a mediální cesty dat. To omezuje přenosové kapacity poskytovatelů a také nemůžeme přesně určit zda je ohlašovací mechanismus uživatelského IMS zařízení správně implementován. Uživatel díky tomu má větší možnosti používat neautorizované služby, které vedou k úmyslnému nebo neúmyslnému podvodu při vyúčtování služeb.
Pokud jsou uživatelé připojeni přímo k veřejnému internetu, je možné aby komunikovali přímým adresováním, což by mohlo být v rozporu s poskytovatelovou strukturou cen. V procesu SIP registrace poskytuje HSS mapování mezi IP adresou IMS zařízení a public user identity, avšak není těžké zjistit přiřazenou IP adresu. Ve fázi navazování hovoru získá uživatel navazující hovor IP adresu volaného, následně může odeslat SIP požadavek CANCEL, který navazování relace zruší (uživateli tak nebude účtován žádný poplatek) a následně vytvořit přímé spojení, díky získané znalosti IP adresy volaného bez účasti účtovacího systému.

NAT a IPSec

Protokol SIP předpokládá, že všechny IP adresy obsažené ve zprávě jsou globálně přístupné. Funkce NAT tento základní předpoklad porušuje. Odpověď na požadavek ze soukromé sítě nemusí být vlivem technologie NAT možné směrovat na původní IP adresu. Aby bylo možné směrovat odpovědi na SIP zprávy vždy, je nutné nasadit tzv. SIP-aware NAT zařízení a implementovat správný převodní mechanismus v IMS síti.
Funkce IPSec (Internet Protocol Security) zajišťuje celkovou důvěrnost, integritu a autentičnost každého odeslaného a přijatého paketu. IPSec zprávy nejen šifruje,

ale také zaručuje integritu a autentizaci jednotlivých paketů. Protokol IPSec není kompatibilní se sítí za NAT. Podstatou NAT je úprava hlavičky paketu, změna soukromé IP adresy a portu na globálně směrovatelnou IP adresu a port a přidání certifikátu. Změna obsahu hlavičky paketu porušuje integritu IPSec a počáteční certifikaci, což může vést ke ztrátě důvěrnosti paketu.

Problém zabezpečení serveru ENUM/DNS

Server ENUM/DNS je distribuovaná databáze, která ukládá mapování vztahů mezi formáty TEL URI a SIP URI pro všechna registrovaná telefonní čísla v síti IMS a jména s IP adresami IMS komponent. Každá taková položka se nazývá záznam o prostředku (RR – Resource Record).
V nasazení DNS existují dva hlavní typy dotazů, rekurzivní a iterativní. V IMS jsou všechny servery ENUM / DNS rozděleny na dvě úrovně. Server 1. úrovně je zodpovědný za národní směrování. Servery 2. úrovně jsou zodpovědné za trasu mezi oblastmi. Pokud jsou nastaveny mezioblastní hovory, kdy například uživatelé z oblasti a volají uživatele z oblasti B, každý dotaz musí projít serverem 1. úrovně. Útočník se může pokusit o navázání hovorů s velkým množstvím čísel nacházejících se v různých oblastech, což způsobí velké množství dotazů. To způsobí, že se neustále budou měnit položky v paměti cache a velké množství dotazů přetíží server první úrovně, což může vést až odmítnutí služby [10].

2 Open-source IMS projekty

2.1 Open IMS Core

Projekt byl spuštěn v roce 2006 pro podporu IMS. Od té doby pak Open IMS Core sloužil jako referenční implementace pro testování a prototypování. Projekt je určen pro výzkum a vývoj, využívá jej mnoho poskytovatelů telekomunikačních zařízení, provozovatelů sítí a univerzitních projektů.
Open IMS Core je open source implemetace funkcí CSCF a HSS, které spolu dohromady tvoří základní prvky všech architektur IMS/NGN, dle specifikace v 3GPP, 3GPP2, ETSI TISPAN a iniciativě PacketCable. Všechny čtyři komponenty (I-CSCF, S-CSCF, P-CSCF a HSS) jsou založeny na open source softwaru [11].

2.1.1 Architektura Open IMS Core

Architektura sítě Open IMS Core, je znázorněna na obrázku 2.1, na kterém můžeme vidět, že Open IMS Core obsahuje pouze základní prvky IMS sítě.

CSCF

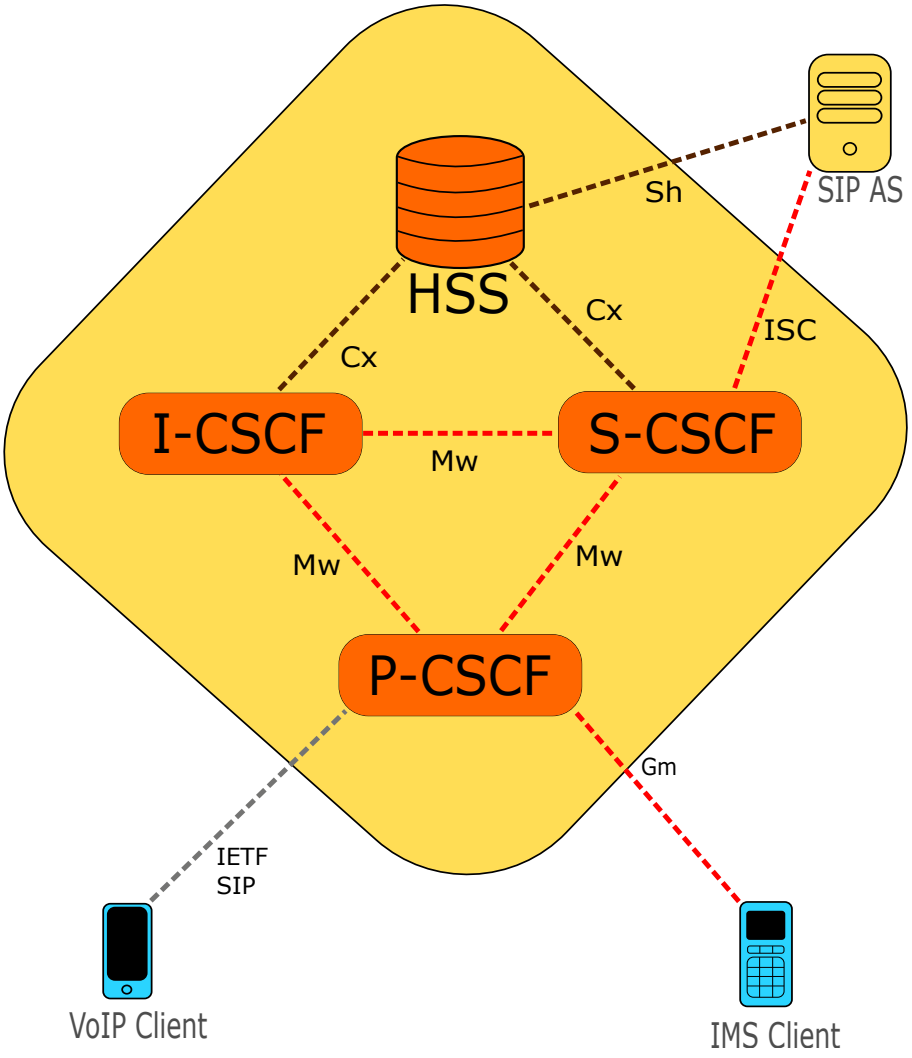
CSCF se skládá z několika modulů:

- modul CDiameterPeer (cdp),
- modul IMS Service Control (isc),
- modul Proxy-CSCF (pcscf),
- modul Interrogating-CSCF (icscf),
- modul Serving-CSCF (scscf).

modul CDiameterPeer Tento modul má umožnit efektivní komunikaci CSCF serverů pomocí protokolu DIAMETER. Je využíván servery I-CSCF a S-CSCF. I-CSCF a S-CSCF mají každý svůj konfigurační soubor ve formátu xml. V těchto souborech je možné nastavit parametry komunikace protokolem Diameter ze strany CSCF serverů.

modul IMS Service Control Tento modul má za úkol poskytovat podporu rozhraní ISC mezi S-CSCF a aplikačními servery. K použití je potřeba mít načtený modul scscf, protože isc využívá registrátor pro ukládání počátečních kritérií filtru. Zprávy předávané aplikačnímu serveru mají následující označení:

```
Route: <AS>, <sip:ifcznačka@[isc_moje_uri];lr;s=xxx;h=xxx;d=xxx>
```



Obr. 2.1: Architektura Open IMS Core

Zpráva je směrována do AS. Pokud AS odpoví na požadavek, bude hodnota 2. záhlaví trasy obsahovat všechny požadované informace o stavu pro identifikaci a obnovení IFC shody.

modul Proxy-CSCF Tento modul poskytuje funkce serveru P-CSCF. Konfigurace P-CSCF je uložena v konfiguračním souboru `pcscf.conf`.

modul Interrogating-CSCF Tento modul poskytuje funkce serveru I-CSCF. K jeho použití je nutný načtený modul CDiameterPerr (cdp), tento modul komunikuje pomocí protokolu DIAMETER s HSS přes rozhraní Cx. Pro správnou funkci I-CSCF, je třeba zajistit v databázi několik tabulek, které jsou definovány v SQL (Structured Query Language) souboru `icscf.sql`.
I-CSCF umožňuji funkci THIG, která poskytuje zvýšené zabezpečení přenosu zpráv,

pomocí šifrování na citlivých položkách záhlaví přenášených zpráv.
Konfigurace I-CSCF je uložena v konfigurační souboru `icscf.conf`.

modul Serving-CSCF Tento modul poskytuje funkce serveru S-CSCF. K jeho použití je nutný načtený modul CDiameterPerr (cdp), tento modul komunikuje pomocí protokolu DIAMETER s HSS přes rozhraní Cx.
Konfigurace S-CSCF je uložena v konfigurační souboru `scscf.conf`.

HSS

FHoSS definuje Java implementaci rozhraní:

- Sh – mohou ho použít aplikační servery pro přístup k HSS.
- Zh – využívá s k případné komunikaci s BSF (Bootstrapping Server Function).
- Cx – které se používá ke komunikaci s I-CSCF a S-CSCF.

K dispozici je také implementace těchto rozhraní, která provádí mapování funkcí rozhraní na požadavky protokolu Diameter. Generování kódu je založeno na schématu xml, které je uvedeno ve specifikaci 3GPP pro související rozhraní.
Jádrem FHoSS je HSSDiameterStac, který využívá DiameterPeer k odesílání požadavků ostatním prvkům a získává požadavky a odpovědi pomocí CommandListeneru.
Provozní data HSS jsou uložena v databázi. Přístupová vrstvy k datům, umožňující měnit databázoví systém byla vytvořena s využitím frameworku Hibernate persistence.
Pro správu a údržbu FHoSS je k dispozici webové rozhraní, které je implementováno pomocí technologie Java Servlet v kombinaci s webovým frameworkem Apache Struts, jež poskytuje jasnou strukturu a oddělení logiky a GUI.

GUI Vrstva GUI (Graphical User Interface) se využívá ke správě a monitorování FHoSS. Implementaci GUI lze nalézt v balíčku `de.fhg.fokus.hss.web`. Vykreslování provádí několik Java Server Pages, které je možné nalézt ve složce ser-web.

DAL Vrstva DAL (Data Acces Layer) byla vytvořena s využitím frameworku Hibernate persistence. Související datové třídy, obsahující také BO třídy poskytující rozšířené funkce datovým třídám, lze najít v balíčku `de.fgh.fokus.hss.db.model`.

Vrstva rozhraní Tato vrstva popisuje vnější chování HSS. Obsahuje tři specifikace rozhraní, obsažená v balíčcích `de.fhg.fokus.hss.cx`, `de.fhg.fokus.hss.sh` a `de.fhg.fokus.hss.zh`. Pro každé rozhraní existuje přímá implementace. Pro všechny metody rozhraní je v operačních balíčcích přidružená provozní třída.

Konfigurace Výchozí síťová konfigurace je spuštěna na adrese localhostu. Konfigurace lze změnit pomocí následujících souborů:

- DiameterPeerHSS.xml – zde lze upravit konfiguraci Diameter peerů.
- hibernate.properties – zde můžete konfigurovat hlavní vlastnosti pro frameworku hibernate. Implicitně je nakonfigurován pro připojení k mysql na adresu localhostu (127.0.0.1:3306).
- hss.properties – zde lze konfigurovat vlastnosti HSS.
- log4j.properties – obsahuje konfiguraci loggeru.

Konfigurační soubory tomcatu můžeme nalézt ve složce `conf`.

Databáze Ke správné funkci FHoSS, je nutné mít vytvořenou databázi. Instalace FHoSS obsahuje dva SQL skripty pro databáze MySQL s defaultními hodnotami.

- hss_db.sql – vytvoří databázi a tabulky.
- userdata.sql – vytvoří dva uživatele a počáteční hodnoty pro servisní profily.

Pokud bychom chtěli využít jiný DBMS (Database Management System), je nutná podpora frameworkem Hibernate (tuto skutečnost nalezneme na webových stránkách frameworku Hibernate) a je nutné modifikovat skripty, tak aby odpovídali zvolenému databázovému systému.

Spuštění FHoSS lze spustit skriptem startup.sh. Před spuštěním skriptu je nutné se ujistit, zda je proměnná JAVA_HOME nastavena správně. Správa FHoSS je prováděna přes webové rozhraní, defaultně nastavené na adrese http://localhost:8080. Defaultně jsou nastaveny dva uživatelské profily:

- hssadmin – heslo:hss
- hss – heslo:hss

Další uživatele lze nakonfigurovat pomocí souboru /conf/tomcat-user.xml.

IMS Core JavaDiameterPeer

JavaDiameterPeer, stejně jako jeho protějšek CDiamterPeer, je implementace protokolu DIAMETER (IETF RFC3588) napsaná v jazyce Java. Poskytuje efektivní způsob využití zásobníku DIAMETER v prostředí Javy a může snadno rozšířit síťové uzly implementované v Javě pomocí DIAMETER rozhraní. V síti Open IMS Core najde využití v HSS.

DiameterPeer představuje DIAMETER uzel, který implementuje protokol DIAMETER a funguje jako buď jako klient, nebo jako server. Nejdůležitější součástí DiameterPeer je PeerManager, který spravuje sadu peerů. Každý peer má Communicator, který udržuje spojení. Peer je implementován na základě RFC 3588, sekce 5 a obsahuje stavový stroj definovaný ve výše uvedené normě. Peery spravované

PeerManagerem lze konfigurovat pomocí konfiguračního souboru. Mohou být také dynamicky přidávány do PeerManageru.
Příchozí a odchozí zprávy DIAMETERU jsou zpracovány pomocí JavaDiameterPeer různým způsobem. Pro odchozí DIAMETER požadavek odešle JavaDiameterPeer tuto zprávu do PeerManageru přímo. PeerManage následně vybere vhodný Peer pro odeslání zprávy.
Příchozí DIAMETER zpráva, která přijatá Communicatorem je nejdříve odeslána do TaskQueue, fronty typu FIFO (First In, First Out). Jakmile je zpráva k dispozici ve frontě, DiameterWorker ji vyřadí z fronty a doručí ji skupině event listenerů definovaných uživatelem.
DIAMETER klient obvykle odesílá požadavek přímo DIAMETER serveru a přijímá odpověď upravenou EventListenerem, který zpracovává příchozí žádosti na straně serveru.
DIAMETER žádosti a odpovědi na straně klienta zpracovává TransactionWorker, který páruje požadavky s příslušnými odpověďmi.
Nastavení peerů je možné provést v souboru DiameterPeer.xml, který nalezneme ve složce s nainstalovaným FHoSS [11].

2.1.2 Implementované zabezpečení

Podpora IPSec protokolu

Výpis 2.1: Konfigurace IPSec v souboru pcscf.cfg v sekci konfigurace modulů

```
modparam ("pcscf","use_ipsec",1)
modparam ("pcscf","ipsec_host","192.168.20.85")
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")
```

Podporu protokolu IPSec pro P-CSCF, lze zapnout v konfiguračním souboru pro P-CSCF `pcscf.cfg`. Ve výpisu 2.1 je zobrazena konfigurace protokolu IPSec obsažená v sekci konfigurace modulů pro P-CSCF.
Defaultně je možnost komunikace přes protokol IPSec zapnuta, ale pouze jako volitelné zabezpečení, kdy P-CSCF umožňuje přijímat registrace i z IMS klientů, kteří nekomunikují přes IPSec protokol.

Výpis 2.2: Povolení registrace pouze klienty používajícími IPSec protokol v souboru pcscf.cfg v sekci směrovací logiky

```
#Variant 2 - accept only IPSec clients
        if (!P_remove_security_client()){
           route(REGISTER_494);
           break;
        }
```

Ve výpisu 2.2 je zobrazena konfigurace omezující povolení registrace pouze klientům používajícím IPSec protokol.
S-CSCF, stejně jako P-CSCF, umožňuje ve svém konfiguračním souboru `scscf.cfg` nastavit přijímání pouze registrací zabezpečených IPSec protokolem, tedy od klientů komunikujících pomocí IPSec protokolu.

Výpis 2.3: Povolení registrace pouze klienty používajícími IPSec protokol v souboru scscf.cfg v sekci směrovací logiky

```
   # Variant 2 - accept only IPSec clients
    S_challenge("open-ims.test");
    route(Service_Routes);
    t_reply("401", "Unauthorized - Challenging the UE");
```

Ve výpisu 2.3 je zobrazena konfigurace, která povoluje pouze žádosti o registraci zabezpečené pomocí protokolu IPSec.

Podpora TLS protokolu

TLS (Transport Layer Security) je kryptografický protokol poskytující autentizaci koncových bodů komunikace. Typicky je autentizován pouze server. Další možností je vzájemná autentizace, kdy jsou autentizovány obě strany, které si vyměňují klíče nejdříve pomocí asymetrické kryptografie a poté šifrují jednotlivé zprávy pomocí symetrické kryptografie.
Konfiguraci TLS provádíme v konfiguračním souboru P-CSCF `pcscf.cfg`, kde v sekci globálních parametrů definujeme IP adresu a port, na kterém bude naslouchat

P-CSCF šifrovanou komunikaci pomocí a následně povolíme TLS komunikaci. Příklad nastavení v povolení komunikace pomocí protokolu TLS je zobrazen ve výpisu 2.4.

Výpis 2.4: Povolení komunikace pomocí protokolu TLS v konfigurační souboru pcscf.cfg v sekci globálních parametrů

```
listen=tls:192.168.20.85
tls_port_no=4061
enable_tls=yes
```

V souboru `pcscf.cfg` je dále nutné nastavit načtení modulů podporujících TLS a jejich konfigurace, příklad defaultní konfigurace, kterou bylo nutné odkomentovat je zobrazen ve výpisu 2.5.

Výpis 2.5: Povolení komunikace pomocí protokolu TLS v konfigurační souboru pcscf.cfg v sekci konfigurace modulů

```
modparam("pcscf","use_tls",1)
modparam("pcscf","tls_port",4061)
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", 0)
modparam("tls", "tls_disable_compression", 1)
```

Pro vygenerování certifikátů a privátního klíče je použit skript `tls_prepare.sh`, který nalezneme ve složce `ser_ims/cfg`. Po spuštění je zobrazen průvodce vytvořením certifikátů a privátního klíče, kde je výpis toho, co skript aktuálně provedl za operace a jsou tam i dotazy potřebné k vytvoření certifikátu.

Skrytí topologie

Protože SIP je textově orientovaný protokol, všechny informace jsou posílány jako prostý text. Některé položky záhlaví SIP protokolu mohou obsahovat citlivé informace o vnitřní topologii, jako třeba IP adresy jednotlivých entit v rámci vnitřní sítě, apod. Konkrétně jsou zašifrovány položky záhlaví `Via`, `Path`, `Record-Route`

a také `Service-Route`. Jako šifrovací algoritmus byl zvolen Twofish. Funkce THIG je realizována v I-CSCF, pro tuto funkci je vytvořen zvlášť konfigurační soubor `icscf.thig-cfg` ve složce `ser_ims/cfg`. Konfigurační soubor je nutné upravit tak, aby odpovídal aktuálnímu síťovému nastavení. Pro spuštění I-CSCF se skrytím topologie je ve složce `ser_ims/cfg` připraven skript `icscf.thig.sh`, který načte konfiguraci přímo ze souboru `icscf.thig-cfg` [12].

2.1.3 Instalace a zkušební hovor

Instalovat Open IMS Core je možné dvěma způsoby

1. Na čisté instalaci Linuxu (doporučená distribuce je Ubuntu), provést instalaci z SVN úložiště.
2. Stáhnout si na webových stránkách projektu již předinstalovaný virtuální obraz Ubuntu verze 9.04, obsahující všechny části Open IMS Core.

Pro účely diplomové práce byla zvolena druhá varianta, tedy předinstalovaný virtuální obraz. Defaultně je nakonfigurován tak, že prvky Open IMS Core mají adresu lokální smyčky s doménovým jménem `open-ims.test`. Pro ověření základní funkcionality je počáteční nastavení dostačující, pro účely přístupu z vnější sítě bylo nutné použít IP adresu virtuální stanice a změnit doménové jméno odpovídající přiřazenému doménovému jménu v testovací síti, v našem případě `openims`.
Pro změnu potřebných síťových parametrů musíme upravit nastavení DNS serveru a souboru `resolv.conf` ve složce etc v kořenovém adresáři. V souboru resolv.conf, konfigurujeme DNS resolver. Příklad pro nastavení je ve výpisu 2.6

Výpis 2.6: Nastavení souboru resolv.conf

```
nameserver 192.168.20.85
search openims
domain openims
```

Po restartu systému může tento soubor být z důvodu nastavení DHCP přepsán. Konfigurační soubory DNS serveru bind, najdeme ve složce `/etc/bind`, bylo třeba vytvořit konfigurační soubory pro DNS a reversní DNS. Následně provést další nastavení pro DNS server, konfigurované soubory byly tedy

- named.conf – Výpis konfiguračního souboru B.1.
- named.conf.options – Výpis konfiguračního souboru B.2.
- open-ims.dnszone – Výpis konfiguračního souboru B.3.
- open-ims-rev.dnszone – Výpis konfiguračního souboru B.4.

Následně je nutné nakonfigurovat jednotlivé IMS komponenty. Konfigurační soubory, spolu se spouštěcími skripty jsou ve složce `ser_ims/cfg` pro cscf a ve složce `FHoSS/deploy` pro HSS. Bylo třeba změnit IP adresu lokální smyčky na IP adresu,

přidělenou virtuální stanici a změnit doménové jméno na `openims`. Pro konfiguraci CSCF byly použity následující soubory:

- icscf.cfg – Výpis změn konfiguračního souboru B.5.
- icscf.xml – Výpis změn konfiguračního souboru B.6.
- icscf.sql – Výpis změn konfiguračního souboru B.7.
- pcscf.cfg – Výpis změn konfiguračního souboru B.9.
- pcscf.xml – Výpis změn konfiguračního souboru B.8.
- scscf.cfg – Výpis změn konfiguračního souboru B.10.
- scscf.xml – Výpis změn konfiguračního souboru B.11.

Poté byly soubory zkopírovány do kořenové složky Open IMS Core, spolu se spouštěcími skripty.

Pro HSS byly konfigurovány tyto soubory:

- DiameterPeerHSS.xml – Výpis změn konfiguračního souboru B.12.
- hss.properties – Výpis změn konfiguračního souboru B.13.

Ve stejné složce nalezneme spouštěcí skript pro HSS `startup.sh`. Z důvodu změny doménového jména musel být přepsán i SQL skript sloužící k vytvoření HSS databáze `userdata.sql`. Protože byly upraveny SQL skripty, bylo třeba je spustit znovu pomocí příkazů:

Výpis 2.7: Příkazy ke spuštění sql skriptů

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

Databáze MySQL běží na adrese lokální smyčky a přistupuje se k ní pomocí HSS. Po úspěšné konfiguraci je možno spustit komponenty pomocí spouštěcích skriptů. Pro co největší přehlednost je doporučeno spustit každý v samostatném okně terminálu, kde můžeme vidět jejich aktuální stav. Následně je možné si vyzkoušet registraci uživatele do sítě a případně realizovat hovor. Přímo ve virtuálním systému jsou předinstalovány dva IMS klienti, Monster a OpenIC, s kterými je možné si vyzkoušet funkčnost nainstalovaného systému [11], [12].

2.2 Project Clearwater

Clearwater je open source implementace IMS navržená pro masivní škálovatelné nasazení v cloudu, poskytování hlasových služeb, videa a instant messaging. Clearwater zachovává architekturu IMS, ale byl navržen pro cloud.

Standardní distribuce Clearwater je navržena pro rychlé nasazení na platformě Amazon Web Services. Clearwater spolupracuje s většinou standartních SIP klientů,

včetně desktopových softwarových telefonů, mobilních softwarových telefonů a standardních telefonů. Vestavěný aplikační server TAS (Telephony Application Server) umožňuje nabízet služby hlasového volání a videohovoru.

2.2.1 Architektura

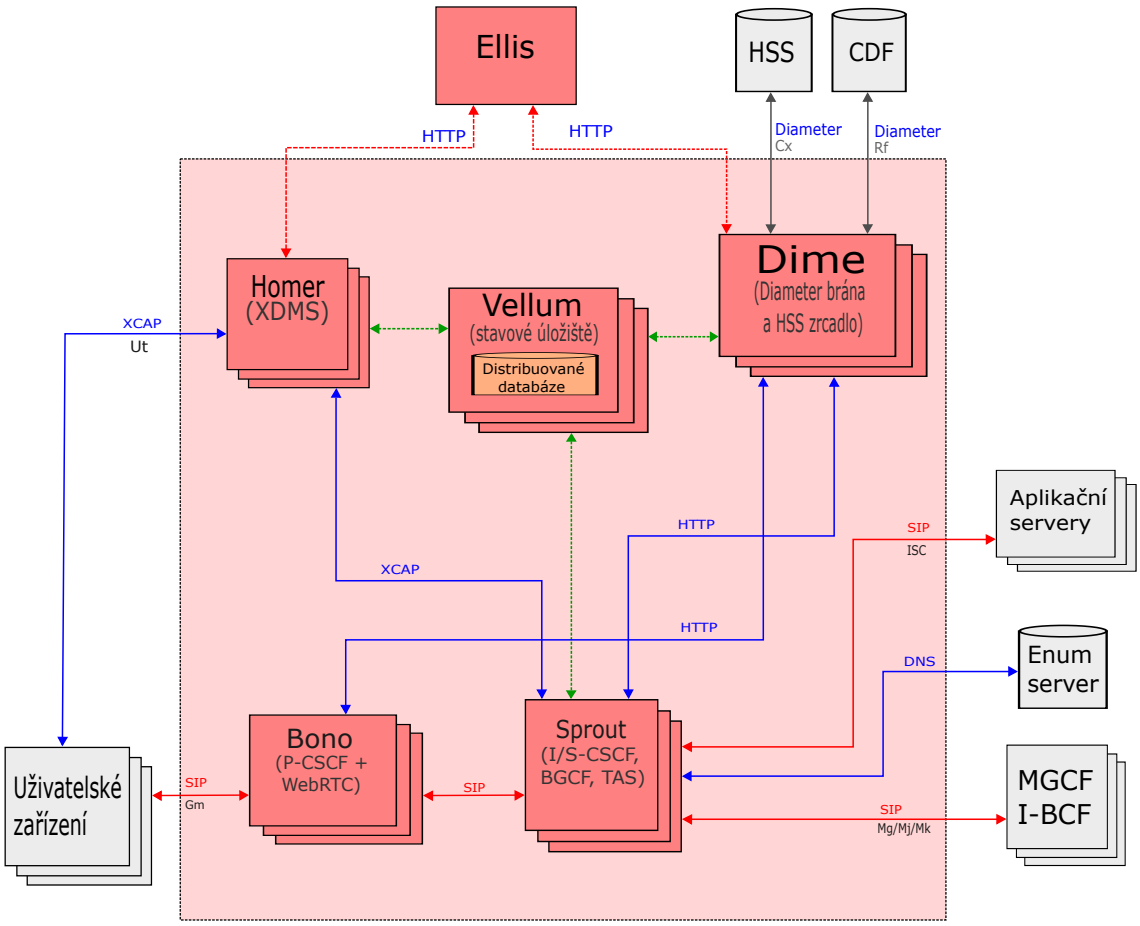
Clearwater byl navržen od základů tak, aby byl optimalizován pro nasazení ve virtualizovaném a cloudovém prostředí. Využívá zavedené návrhové vzory pro vývoj a nasazení rozsáhlých škálovatelných webových aplikací, přizpůsobené vývoji IMS systému. Project Clearwater vycházel z původní IMS architektury, oproti které obsahuje několik změn:

- Všechny komponenty jsou horizontálně škálovatelné pomocí jednoduchého, bezstavového vyvážení zátěže.
- Každý stav s dlouhou životností je uložen pouze na uzlech `Vellum`, které využívají technologie optimalizované pro cloud (např. Cassandra).
- Rozhraní mezi front-endem SIP komponent a back-endem služeb využívají RESTful rozhraní webových služeb.
- Rozhraní mezi různými komponenty využívají sdružování připojení se statistickou recyklací připojení k zajištění rovnoměrného rozložení zátěže.

Architektura Clearwater je popsána na obrázku 2.2.

Bono (Hraniční proxy) Uzly Bono tvoří škálovatelný SIP hraniční proxy poskytující rozhraní kompatibilní se SIP IMS Gm a WebRTC (Web Real-Time Communication) pro koncová zařízení. Uzel Bono poskytuje vstupní bod pro připojení IMS zařízení ke Clearwateru, včetně podpory mechanismů procházení NAT. IMS zařízení využívá konkrétní uzel Bono po celou dobu kdy je registrováno, při vzniku chyby v přiřazeném Bono uzlu může využít jiný Bono uzel. IMS zařízení se mohou připojit k Bonu uzlu pomocí kombinace protokolů SIP/UDP nebo SIP/TCP. Bono podporuje každé WebRTC zařízení, které provádí signalizaci hovoru pomocí protokolu SIP přes WebSocket. Prvek Bono lze nahradit P-CSCF nebo Session Border Controllerem (SBC) implementujícím P-CSCF.

Sprout (SIP router) Uzly Sprout fungují jako horizontálně škálovatelný, kombinovaný SIP registrátor a autoritativní směrovací proxy, zpracovávají autentizaci uživatelů a ISC rozhraní pro aplikační servery. Uzly Sprout také obsahují zabudovaný aplikační server MMTEL (Multimedia Telephony). Transakce SIP jsou zátěžově vyrovnávány ve Sprout clusteru, takže mezi IMS zařízením a konkrétním Sprout uzlem neexistuje dlouhodobé spojení. Sprout neukládá žádná data s dlouhou dobou životnosti, místo toho využívá:



Obr. 2.2: Architektura Clearwater

- Rozhraní webových služeb pro Homestead a Homer k načtení HSS konfigurace, jako autentizační data nebo profily uživatelů a nastavení služeb MMTEL.
- API pro Vellum sloužící k ukládání údajů o registraci účastníka a spuštění časovačů.

Sprout supluje většinu funkcí I-CSCF a S-CSCF, zbytek poskytuje Dime (podpořený dlouhodobými datovými úložišti na Vellum).

Dime (Diameter brána) Uzly Dime jsou využívány k propojení komponent Homestead (HSS Cache) a Ralf (CDF).

Homestead (HSS cache) Homestead poskytuje rozhraní webových služeb pro Sprout sloužící k získávání ověřovacích údajů a informací o uživatelském profilu. Může data získávat sám (v tom případě poskytuje rozhraní pro webové služby), nebo může data stahovat z HSS přes Cx rozhraní. Samostatné uzly Homestead jsou bezstavové, přičemž hlavní data (mastered data) a data vyrovnávací paměti(cached

data) jsou uložena ve Vellum. Hlavní data jsou uložena na úložném clusteru Cassandra, dostupné přes rozhraní Thrift. Data vyrovnávací paměti jsou uloženy v cluesteru Memcached. V IMS architektuře je zrcadlová funkce HSS považována za součást I-CSCF a S-CSCF. V Clearwateru je funkce implementována kombinací clusterů Sprout a Dime.

Ralf (CDF) Ralf poskytuje HTTP API, které mohou Bono a Sprout použít k hlášení účtovacích událostí, ty jsou pomocí fakturačního rozhraní Rf předány do CDF (Charging Data Function). Ralf je bezstavový komponent, využívá Vellum k udržení stavu relace s dlouhou životností a spuštění časovačů potřebných k fakturaci.

Vellum (stavové úložiště) Vellum se používá k udržení všech dlouhodobých stavů. To je realizováno spuštěním mnoha cloudových distribuovaných úložných clusterů.

- Cassandra – je využívána Homesteadem k ukládání ověřovacích údajů a informací o profilu. Je využívána Homerem k ukládání nastavení služeb MMTEL. Vellum poskytuje API Thrift.
- etcd. – využívá Vellum ke sdílení informací o clusterech mezi uzly Vellum a jinými uzly se sdílenou konfiguraci.
- Chronos – je distribuovaná, redundantní a spolehlivá časová služba vyvinutá Clearwaterem. Využívají jej uzly Sprout a Ralf, aby umožnily spuštění časovačů (například pro časové omezenost SIP registrace) nezávislých na operacích konkrétního uzlu (jeden uzel může nastavit časovač a jiný na něj reagovat, když se objeví). Chronos je přístupný přes HTTP API.
- Memcached/Astaire/Rogers – Vellum poskytuje cluster Memcached podporovaný Rogersem a synchronizovaný pomocí Astaire. Astraire je služba, která umožňuje rychlejší škálovatelnost memcached clusterů. Tento cluster využívají Sprout, Homestead a Ralf pro uložení stavu registrace a relace.

Homer (XDMS) Homer je standardní XDMS (XML Data Management Server) používaný k ukládání souborů nastavení služeb MMTEL pro každého uživatele systému. Soubory jsou vytvářeny, čteny, aktualizovány a mazány pomocí standardního rozhraní XCAP. Využívá Vellum jako úložiště pro všechna data s dlouhou životností.

Ellis Ellis je webový portál zajišťující vlastní přihlášení, správu hesel, správu linek a kontrolu nastavení služeb MMTEL. Není určen k tomu aby byl součástí produkčních implementací Clearwater, ale je stavěný tak aby byl snadno použitelný samostatně.

Vyrovnávání zátěže Clearwater využívá vyvažování zátěže DNS, kdy jsou clustery pružně přizpůsobeny celkové zátěži. Pro všechny Sprout uzly je nakonfigurován jeden název domény. Každý Bono uzel udržuje skupinu SIP relací ke Sprout uzlu, přičemž konkrétní uzel pro každé připojení je vybrán náhodně ze seznamu adres vrácených DNS serverem. Připojení jsou v případě poruchy a pravidelně recyklovaná, pokaždé vybírají jinou adresu ze seznamu vráceného DNS serverem. Podobná technika se používá pro HTTP spojení mezi clustery Sprout a Homer/Dime – každý Sprout udržuje seznam zátěžově vyrovnaných připojení napříč clustery Homer/Dime a periodicky vyžaduje recyklaci těchto spojení.

Spolehlivost a redundance Spolehlivost spočívá v dodržování společných návrhových vzorů pro škálovatelné webové služby, kdy je většina prvků bezstavová. Ukládání dat s dlouhou životností probíhá ve speciálně navržených spolehlivých škálovatelných clusterových datových úložištích. Bono i Sprout fungují spíše jako transakčně stavové proxy, než dialogové stavové proxy – stav transakce je ve srovnání s dialogovým stavem krátkodobý. Jako hlavní bod pro klientská připojení přes NAT, zůstává uzel Bono na signální cestě po celou dobu SIP dialogu. Sprout uzel je v signální cestě pouze pro počáteční interakci a následné požadavky jsou směrovány přes Sprout cluster, takže selhání Sprout uzlu negativně neovlivní zavedené SIP dialogy. SIP stavy s dlouhou životností (registrační data a stav předplatného), jsou uloženy v redundantním sdíleném datovém úložišti (memcached jako součást Vellum uzlů) a nejsou tedy vázány na konkrétní Sprout uzel. Podobně si Dime a Homer pouze zachovávají místní stav pro nevyřízené žádosti – všechny stavy s dlouhou životností jsou redundantně uloženy v clusterech datových úložišť realizovaných pomocí Vellum [13].

2.2.2 Implementované zabezpečení

Bezpečnost cloudu

SIP komunikace je rozdělena na důvěryhodnou (pro toky zpráv mezi uzly Sprout a Bono a důvěryhodnými aplikačními servery) a nedůvěryhodnou zónu (pro toky zpráv mezi uzly Bono a externími klienty nebo jinými systémy). Tyto zóny používají různé porty, které umožňují izolovat důvěryhodnou zónu pomocí skupin zabezpečení a pravidel brány firewall, zatímco standardní mechanismy ověřování SIP se používají k ochraně nedůvěryhodných portů. Jiná rozhraní, jako XCAP a Homestead používají pro zabezpečení kombinaci uzamčených portů, standardních schémat ověřování a sdílených tajných API klíčů [13].

Podpora počítání nonce

Počítání nonce umožňuje IMS zařízení uživatele předběžně vypočítat autentizační odpovědi při opětovné registraci, tím se stává opětovná registrace efektivnější. Běžné registrační schéma není moc efektivní, protože potřebuje dva toky registrace – odpověď a generování ověřovací výzvy může být nákladné pro síť (každá výzva může zahrnovat požadavek na HSS).
RFC 2617 umožňuje uživatelskému zařízení pokračovat v používání stejné výzvy a reagovat na ni několikrát. Jako prevence proti replay útokům uživatelské zařízení a síť sledují počet nonce, který určuje kolikrát byla výzva použita k vygenerování odpovědi (první odpověď má hodnotu počtu nonce 1, druhá odpověď 2 atd.).
Počet nonce zahrnuje IMS zařízení ve zprávě SIP REGISTER spolu s autentizační odpovědí. Pokud síť obdrží odpověď s již použitým počtem nonce, tak ji zahodí. Tento mechanismus zabraňuje replay útokům, protože kdyby se útočníkovi podařilo odchytit odeslanou odpověď a on by se ji pokusil zopakovat, byla by zamítnuta z důvodu už použitého počtu nonce. Druhou možností je, že by útočník navýšil počet nonce o 1, kde by však bez znalosti algoritmu, kterým jsou nonce vytvářeny neměl šanci uhodnout správné nonce, tudíž by jeho pokus o podvodnou zprávu byl opět odmítnut IMS sítí.
Pro větší efektivitu a zamezení potenciálním chybám je vhodné mít v systému implementován HSS a I-CSCF.
Defaultně je podpora počítání nonce vypnutá, protože u ní záleží na konkrétní topologii a v některých případech může být nežádoucí, případně špatně implementována. Zapnout se dá pomocí pole `nonce_count_supported=Y` v konfiguračním souboru */etc/clearwater/shared_config*. Následně je nutné aktualizovat nastavení pomocí */usr/share/clearwater/clearwater-config-manager/scripts/upload_shared_config* [14].

Podpora protokolu IPSec

Project Clearwater aktuálně nepodporuje bezpečnostní protokol IPSec, respektive komponent Bono (hraniční proxy) nepodporuje protokol IPSec.

Podpora protokolu TLS

Project Clearwater podporuje zabezpečení protokolem TLS pouze u protokolu DIAMETER, pro zabezpečenou komunikaci pomocí TLS v rámci Dime a procesů Ralf a Homestead.
Nastavení portů zabezpečených pomocí TLS je možno v souboru */etc/clearwater/shared_config* pomocí `ralf_secure_listen_port` a `hs_secure_listen_port`.
Zabezpečení SIP zpráv pomocí protokolu TLS Clearwater nepodporuje, respektive komponent Bono (hraniční proxy) nepodporuje protokol TLS [15].

Chronos

Sprout používá Chronos (horizontálně škálovatelnou, redundantní službu časovače pro cloud) pro zjištění expirace registrace a také k časovému omezení platnosti nonce, to vede k zamezení replay útokům a tím pádem i ke zvýšení bezpečnosti systému [14].

2.2.3 Instalace a zkušební hovor

Pro instalaci jsou k dispozici čtyři známé metody:

1. All-in-one obrazy virtuálních prostředí, buď ve formátu AMI (Amazon Machine Image) běžícím na Amazon EC2, nebo ve formátu OVF běžícím na platformách VMware nebo VirtualBox.
   Tento způsob instalace je velice snadný, neposkytuje žádnou redundanci nebo škálovatelnost a má relativně omezený výkon. Je vhodný pro začátečníky, kteří se chtějí seznámit s Clearwaterem a později přejít na rozsáhlejší nasazení, pomocí jedné z následujících metod.
2. Automatická instalace pomocí systému Chef. Tato instalace je doporučená pro rozsáhlé sítě. V současné době je podporována pouze v cloudu amazonu EC2 a vyžaduje aby byl DNS server realizován pomocí Amazon Route53, který řídí kořenovou doménu. Vyžaduje také přístup k běžícímu serveru Chef. Nastavení je poměrně složité, ale je nutné ho provést jen jednou, čímž je usnadněno vícenásobné nasazení.
3. Manuální instalace pomocí balíčků Debianu a konfigurace každého počítače zvlášť. Doporučená metoda, pokud Chef není podporován na používané virtualizační platformě, nebo pokud DNS není poskytováno přes Amazon Route53. Instalaci lze provést na libovolné kolekci počítačů (je zapotřebí nejméně 5) na nichž je instalován systém Ubuntu 14.04. Vyžaduje manuální konfiguraci všech počítaců, firewall bran a DNS, tudíž to není vhodné řešení pro rozsáhlé nasazení.
4. Instalace ze zdrojového kódu. Je vhodná pokud je použit operační systém, který není založen na Ubuntu, nebo pokud potřebujeme otestovat úpravu kódu, případně různá vylepšení která si uživatel sám doprogramoval.

Z praktických důvodů byla zvolena první možnost, tedy instalace pomocí all-in-one obrazu na virtualizační platformě VMWare.
Virtuální obraz využívá DHCP k získání IP konfigurace, takže virtualizační platforma musí nativně podporovat DHCP, nebo musí být připojena k síti pomocí DHCP serveru.
Nejdříve je nutné stáhnout obraz disku z oficiálních stránek projektu, pak stačí spustit OVF soubor a program VMware Player si virtuální kopii importuje, spustí

a přidá do seznamu virtuálních počítačů. Přihlašovací údaje k systému jsou defaultně nastaveny jako `ubuntu` s heslem `cw-aio`. OVF poskytuje tři síťové služby:

1. SSH – uživatelské jméno je `ubuntu` a heslo `cw-aio`.
2. HTTP rozhraní pro správu uživatelů Ellis – registrační kód je `secret`, lze však kdykoliv změnit v konfiguračních souborech.
3. SIP rozhraní na Bono pro signalizaci hovorů. Potřebné údaje jsou poskytována prostřednictvím Ellis.

VMware Player vytvoří síť mezi počítačem a virtuální strojem a následně přidělí IP adresu virtuálního počítače. Zadáním této adresy do webového prohlížeče, se dostaneme do webového rozhraní Ellis. IP adresu virtuálního počítače zjistíme pomocí příkazu `hostname -I`. Pro registraci uživatele přes některého SIP klienta je nutné nastavit odchozí proxy na přidělenou IP adresu s portem 5060. Instalace pomocí `all-in-one` má přiděleno doménové jméno example.com.

Pro používání Eliss je nutné se registrovat, `signup_key` je defaultně nastaven jako `secret`. Po přihlášení vytvoří Ellis automaticky jedno číslo a přiřadí k němu heslo (zobrazí se pouze jednou, ale lze kdykoliv restartovat). Další čísla lze přidávat přes `Add Number` v uživatelském rozhraní Ellis [15].

3 Testování IMS řešení

V této kapitole bude popsána praktická část diplomové práce, kde budou otestovány jednotlivá IMS řešení s otevřeným kódem. Pro oba virtuální stroje byly nastaveny stejné hardwarové parametry, tedy 4 GB operační paměti a 2 jádra procesoru Intel Core i5 3570 3.40 GHz.

3.1 Abacus 5000

Abacus 5000 je integrovaný testovací systém IP a PSTN telefonie s analogovými, TDM (časový multiplex) a ethernet rozhraními. Systém generuje skutečné hlasové streamy pro simulaci reálného zatížení a umožňuje měření kvality hlasu v reálném čase.
Abacus 5000 dále umožňuje testování pokročilé signalizace kanálu, funkční testování, kapacitu, výkon, interoperabilitu, robustnost, distribuované testování s více systémy Abacus, VoIP a IMS zátěžové testování, VoIP bezpečnostní testování a emulaci softswitche pro testování nové generace signalizačních a trunkových bran. [16]. Abacus 5000 může fungovat jako generátor hovorů, nebo jako přepínač hovorů. Abacus 5000 je modulární systém, který může poskytovat řadu rozhraní, která vyhovují požadavkům většiny testů. Subsystémy CG (Circuit Generator) jsou k dispozici pro celou řadu rozhraní a funkcí, lze mezi je sebou libovolně kombinovat a přidávat dle požadavků na testování. Samotný systém obsahuje minimálně systémový kontrolér a jeden CG subsystém.
Abacus 5000 poskytuje GUI, díky čemuž lze tester ovládat přímo z počítače, ke kterému je Abacus 5000 připojen buď pomocí sériového portu nebo LAN. Systém Abacus 5000 umožňuje měnit nastavení jednotlivých protokolů dle potřeb konkrétních testů pomocí vestavěného editoru. Dále umožňuje zobrazit statistiky testu a rozsáhlé nastavení jednotlivých testů, které je možné ukládat a opakovaně používat [17].

3.2 Sledované parametry

Pro účely výkonového testování byly sledovány následující parametry:

- Okno statistics
  - Počet pokusů o navázání hovoru. Jsou započítávány jak pokusy směrem od volajícího k volanému, tak od volaného k volajícímu (jeden hovor je započítán dvakrát).
  - Průměrný počet pokusů o navázaní hovoru za sekundu (Call atempts per second (average)).
  - Procento úspěšných hovorů (% Call completions)

  - Počet chyb – důležité pro určení, kolik hovorů bylo zahozeno.
- Okno Summary(Variance)
  - Průměrná doba odezvy SIP zpráv (response time)
  - Průměrná doba zahájení hovoru (call setup)
  - Průběh doby odezvy (response time) během testu
  - Průběh doby sestavení hovoru (call setup) během testu
  - Průměrná doba registrace – tento parametr je sledován pouze u měření parametrů registrace.
- Parametry změřené ve virtálních strojích
  - Průměrné vytížení CPU (Centrální procesorová jednotka)
  - Průměrný počet přerušení za sekundu
  - Graf vytížení CPU
  - Graf počtu přerušení za sekundu
- Skutečné délka testu – reálná délka testu bez inicializační fáze, ve které probíhala registrace. Odregistrace je v tomto čase započtena.

Tester Abacus 5000 umožňuje změřit velké množství parametrů, ale neumožňuje změřit hardwarové parametry na virtuálních strojích s IMS.
U virtuálního stroje s Open IMS Core bylo měření vytížení cpu a množství přerušení za sekundu získáváno pomocí informací ze souboru `/proc/stat`, který ukazuje aktuální informace o CPU. Pro účely získávání těchto informací byl napsán skript 3.1. Skript je možné kdykoliv ukončit stisknutím kláves `Ctrl` a `C`. Skript potřebné informace zapisuje do souborů `cpu.txt` a `intr.txt`, které po každém spuštění skriptu přemazává, tudíž je nutné si vždy naměřené hodnoty zálohovat pro další zpracování.

Výpis 3.1: Skript pro získání informací o CPU

```
grep 'cpu ' /proc/stat >cpu.txt
grep 'intr' /proc/stat >intr.txt
sleep 1
for(( ; ;))
do
grep 'cpu ' /proc/stat >>cpu.txt
grep 'intr' /proc/stat >>intr.txt
sleep 1
done
```

Získaná data byla následně zpracována v Excelu, kde byly také vytvořeny grafy naměřených hodnot. Varianta výpočtu ze souboru `/proc/stat`, byla zvolena kvůli problémům s programem `apt-get`, které znemožnili instalaci sofistikovanějšího nástroje `collectl`, který byl využit při měření vybraných parametrů u Clearwateru.
U virtuálního stroje s Clearwaterem bylo měření vytížení CPU a množství přerušení

za sekundu realizováno pomocí programu `collectl`, který zobrazuje různé informace o systému, například potřebné informace o CPU, disku, paměti, síti atd. Zaznamenávání parametrů pomocí programu `collectl` spustíme příkazem `collectl -all -P -f clearwater`, příkaz nám zaznamená všechny parametry do komprimovaného souboru `clearwater-cw-aio-<datum>.tab.gz`. Pomocí Excelu jsou následně vytvořeny grafy naměřených hodnot.

3.3 Inicializační testování

Před započetím hlubších testů je důležité vyzkoušet, zda jsou testované technologie vůbec schopny komunikovat s testerem, což je i smyslem tohoto testování.
Nejdříve bylo nutné definovat SUT (System Under Test) nastavení (cesta k nastavení je *Protocol Selection -> SUT -> SUT Config*), kde v kartě *Network Setting* byly nastaveny údaje proxy serveru daného systému:

- Name – jméno (vyplněno jako název domény) proxy serveru.
- Address – IP adresa na které se nachází proxy server.
- Port number – číslo portu na kterém komunikuje proxy server.

Následně byly vyplněny údaje pro registrační server:

- Name – jméno (vyplněno jako název domény) registračního serveru.
- Address – IP adresa na které se nachází registrační server.
- Port number – číslo portu na kterém komunikuje registrační server.
- Register phones on pre-start – zaškrtnuto, aby se uživatelé registrovali k síti v inicializační části testu.

Ostatní parametry můžou zůstat tak jak jsou, pouze v případě delších testů by museli být upraveny.
Nejvíce konfigurace bylo prováděno v sekci *Partition and Timing*, v první kartě *Association* bylo nastaveno pole *Configuration* na *OT OT OT...*, čímž bylo definováno, že hovory budou probíhat mezi sousedními kanály (uživatel 1 volá uživateli 2, který hovor přijímá) a potvrzení cesty *Path Confirmation* na hodnotu *Default*.
Dále je nutné nastavit časový scénář testu (*Timing -> Load Profile*, kde je možnost si definovat scénář pomocí zadaných údajů:

- Load Type – typ zátěže
  - Call rate – je definován určitý počet hovorů generovaný za sekundu.
  - Call volume – je definován určitý počet hovorů, které proběhnou v určitém čase.
  - SIP Registrations Only – je definován určitý počet pokusů o registraci za sekundu.
- Guard time, sec – čas mezi koncem existujícího hovoru a začátkem dalšího hovoru v rámci jednoho kanálu.

- Profile duration – ukazuje celkovou délku testu, vypočtenou dle nastavení zátěže.
- Call length – délka jednoho hovoru.
- Repetition, count – počet opakování definované fáze.
- Pattern – průběh zatížení
  - SawTooth – pilovitý průběh
  - Rectangle – obdélníkový průběh
  - SawTooth – lichoběžníkový průběh
  - SawTooth – průběh s poissonovým rozdělením (jen pro typ zátěže call rate)
  - SawTooth – schodovitý průběh
- Nastavení průběhu zatížení je pro každý průběh jiný a jsou tam specifikovány doby trvání a počty hovorů v určitých fázích průběhu.
- Phase duration – doba trvání jedné fáze.

V kartě *Scripts* byly nastaveny položky *Script originate* a *Script terminate* jako *Turbo RTP*, z důvodu velkého počtu kanálů ve výkonostním testování.
Bylo třeba nastavit koncové body, kdy v případě IMS bylo nutné vyplnit tyto údaje u každého uživatele v kartě *Phones Endpoint -> Endpoints*):

- User Name – uživatelské jméno
- Domain URL – doménové jméno
- Realm Name – stejné jako doménové jméno
- Realm auth Name – private user identity uživatele
- Realm auth Password – heslo k účtu uživatele
- VLAN1 – nastavení VLAN
- IPv4 – IP adresa testeru
- Gateway IPv4 – IP adresa výchozí brány
- Subnet Mask IPv4 – maska sítě IP adresy testeru

Počet uživatelů v každém testu lze nastavit jednoduše v jakékoliv kartě *Partition and Timing* upravením pole *Total*.
Následně kvůli nekompatibilitě původních SIP zpráv nadefinovaných v testeru, bylo třeba tyto zprávy mírně upravit, tak aby je testovaná síť přijala a vyhodnotila je jako správné. Ke zjištění správného formátu SIP zpráv bylo třeba analyzovat testovací pokusy pomocí programu Wireshark a porovnávat je s komunikací, která byla korektní pomocí některého z IMS klientů. Konfigurace SIP zpráv byla provedena v sekci *Configure -> Protocol Development*.

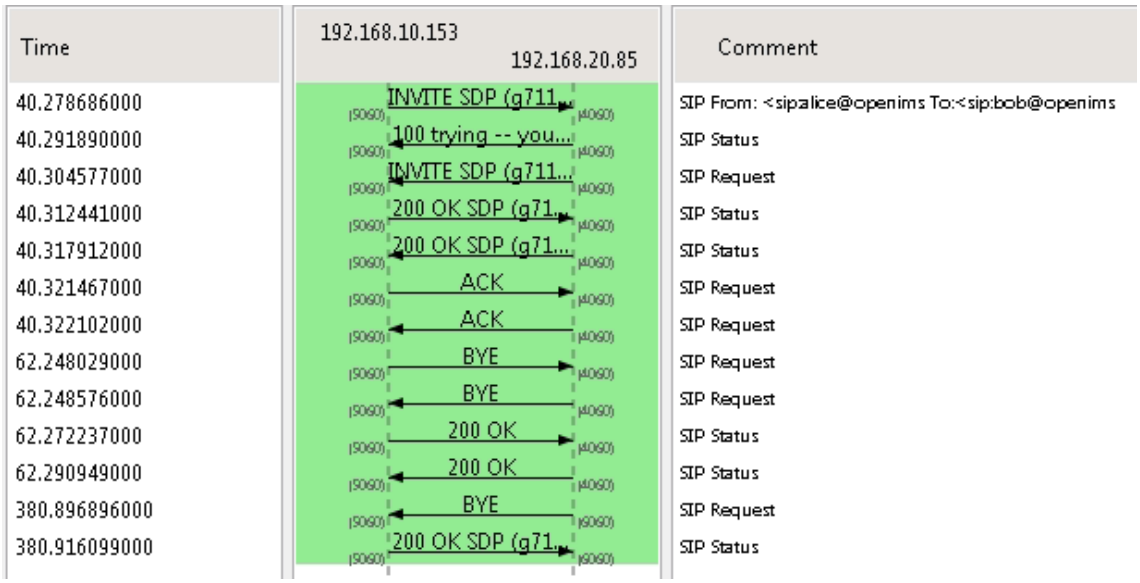
3.3.1 Open IMS Core

Nastavení koncových bodů použitých pro účely inicializačního testu je na obrázku 3.1. Následně bylo definováno SUT nastavení, kde byly vyplněny informace o proxy

	From - To	EP Count	User Name	Domain URL	Realm Name	Realm Auth Name	Realm Auth Password	VLAN 1	IP v4	Gateway IPv4	Subnet Mask IPv4
	1	1	alice	openims	openims	alice@openims	alice	0x8100 5 0 303	192.168.10.153	192.168.10.1	255.255.255.0
	2	1	bob	openims	openims	bob@openims	bob	0x8100 5 0 303	192.168.10.153	192.168.10.1	255.255.255.0

Obr. 3.1: Nastavení koncových bodů na testeru pro Open IMS Core

a registračním serveru.
Výsledkem byl kompletní test, který byl nastaven na 6 minut. V programu Wireshark byla zachycena kompletní komunikace, kde tok i formát SIP zpráv odpovídá teoretickým předpokladům o korektním zahájení a ukončení hovoru. Tok zpráv můžeme vidět na obrázku 3.2 Tímto testem byla vyzkoušena funkčnost komunikace testeru



Obr. 3.2: Tok SIP zpráv v Open IMS Core zachycený wiresharkem

Abacus 5000 se systémem Open IMS Core, čímž byl položen základ pro zátěžové testování.

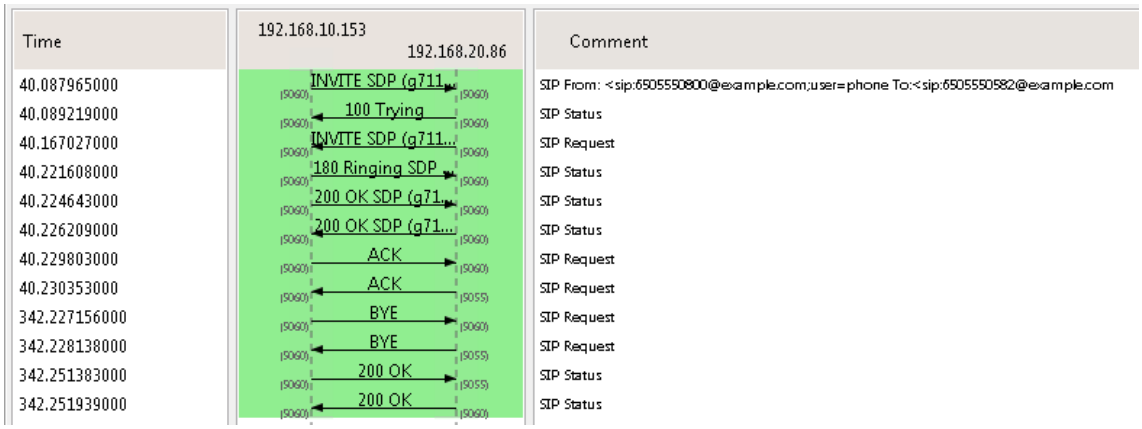
3.3.2 Project Clearwater

Nastavení koncových bodů použitých pro účely inicializačního testu je na obrázku 3.3. Následně bylo definováno SUT nastavení, kde byly vyplněny informace o proxy a registračním serveru.
Výsledkem byl kompletní test, který byl nastaven na 6 minut. V programu Wireshark byla zachycena kompletní komunikace, kde tok i formát SIP zpráv odpovídá

	From - To	EP Count	User Name	Domain URL	Realm Name	Realm Auth Name	Realm Auth Password	VLAN 1	IP v4	Gateway IPv4	Subnet Mask IPv4
	1	1	6505550800	example.com	example.com	6505550800@example.com	papWhKWyN	0x8100 5 0 303	192.168.10.153	192.168.10.1	255.255.255.0
	2	1	6505550582	example.com	example.com	6505550582@example.com	mw4Gbz56F	0x8100 5 0 303	192.168.10.153	192.168.10.1	255.255.255.0

Obr. 3.3: Nastavení koncových bodů na testeru pro Clearwater

teoretickým předpokladům o korektním zahájení a ukončení hovoru. Tok zpráv můžeme vidět na obrázku 3.4. Tímto testem byla vyzkoušena funkčnost komunikace



Obr. 3.4: Tok SIP zpráv v Clearwateru zachycený wiresharkem

testeru Abacus 5000 se systémem Clearwater, čímž byl položen základ pro zátěžové testování.

3.4 Vytvoření testovacích uživatelských profilů

Pro zátěžové testování je nutné mít vytvořené větší množství uživatelských profilů v obou IMS sítích. Vytvářet uživatelské profily po jednom pomocí webových rozhraní by bylo zdlouhavé a neefektivní, proto bylo nutné najít způsob jak vytvořit rychle větší množství uživatelských profilů.

Open IMS Core

U Open IMS Core byl využit skript `multiuser.sh` umožňující vytvořit definované množství uživatelských profilů. Syntaxe skriptu `multiuser.sh` je uvedena ve výpisu 3.2

Výpis 3.2: Syntaxe skriptu a příklad použití skriptu

```
./multiuser.sh -s <start_index> -e <start_index> [-a|-d]
...
./multiuser.sh -s 1 -e 11 -a
```

Kde

- -s – začátek indexace uživatelského jména,
- -e – konec indexace uživatelského jména,
- -a – přidat uživatele,
- -d – smazat uživatele.

Uvedený příklad z výpisu 3.2 vytvoří 10 uživatelů, začínajících jménem subs0001 a končících jménem subs0011 (prefix čísel je definován ve skriptu `multiuser.sh` a lze jej změnit přepsáním původní hodnoty) [18].

Clearwater

Homestead-Prov poskytuje rozhraní API nejen pro Ellis, ale také pro příkazový řádek, kde je možné spravovat databázi uživatelskcýh profilů (přidávat, měnit, mazat a zobrazit). Ve výchozím nastavení jsou nástroje nainstalovány pouze v uzlech Dime (součást balíčku clearwater-prov-tools) v adresáři `/usr/share/clearwater/bin`. Zde se nabízela možnost vytvořit si skript, který by pomocí daných metod vytvořil několik uživatelů s definovanými vlastnostmi. Existuje ale ještě jedna rychlejší a pohodlnější možnost, která byla vytvořena právě k účelu zátěžového testování. Stačí jen spustit následující skript na kterémkoli uzlu Homestead Cassandra, jak je uvedeno ve výpisu 3.3

Výpis 3.3: Příklady použití skriptu stress_provision.sh

```
/usr/share/clearwater/crest-prov/src/metaswitch/crest
/tools/stress_provision.sh
...
/usr/share/clearwater/crest-prov/src/metaswitch/crest
/tools/stress_provision.sh 20000
```

Skript bez argumentu vytvoří 50 0000 uživatelů, jejichž číslování začíná od 201000000 a s každým vytvořeným uživatelem je inkrementováno. Pro všechny uživatele nastaví heslo 77kkzTyGW. Pro vytvoření jiného počtu čísel lze použít argument, kterým jej specifikujeme, v uvedeném příkladu je to 20 000.

3.5 Schodovitý nárůst zátěže do 100 hovorů

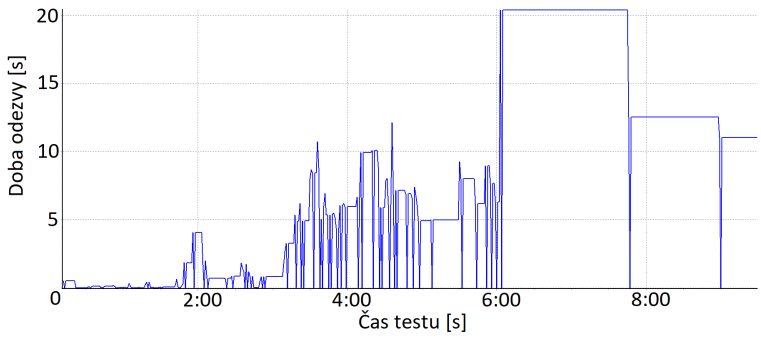
Časový scénář znázorňuje tabulka 3.1. Časový scénář testu je pro všechny měřené IMS systémy stejný, takže je uveden pouze jednou.

Tab. 3.1: Časový scénář testu

Parametr	Hodnota
Délka hovoru	10 s
Průběh	Schodovitý
Hodnota prvního schodku	2 hovory
Výška schodku	2 hovory
Počet schodků	50 hovorů
Délka testu	8:20 min

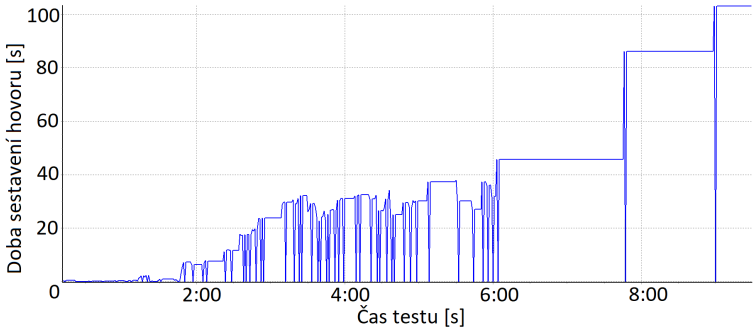
3.5.1 Open IMS Core se zapnutým logováním u cscf

V čase testu 1:50 dosáhla doba odezvy SIP zpráv hodnoty větší než 1 s, pro zatížení 20 probíhajících hovorů. Od zmiňovaného času doba odezvy rostla, svého maxima dosáhla v čase 6:05, kdy byla odezva 20,4 s. Poté doba odezvy zhruba o třetinu klesla. Celý průběh je zobrazen v grafu 3.5 Vliv zvyšujícího se zatížení můžeme

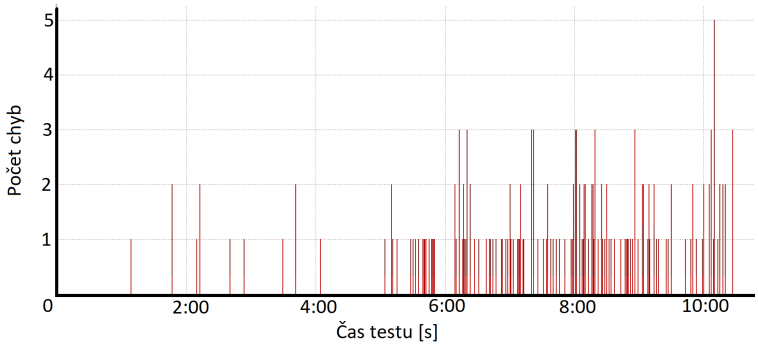


Obr. 3.5: Průběh doby odezvy (response time) během testu

sledovat i v nárůstu doby potřebné k zahájení hovoru. Zhruba v čase 1:50 byla doba potřebná k zahájení hovoru větší než 6 s. Postupně se navyšovala a ve svém maximu v závěru testu dosahovala hodnoty 103 s. Celý průběh je zobrazen v grafu 3.6. V čase 1:07 od začátku testu byl zahozen první hovor – chyba `connect failed`. Čas 1:07 odpovídá zátěži 12 hovorů, se zvyšující se zátěží se zvýšila i četnost chyb. Zejména od 5. minuty testu četnost chyb prudce stoupla a docházelo k častějšímu zahazování hovorů. Četnost chyb v průběhu testu je zobrazena v grafu 3.7.



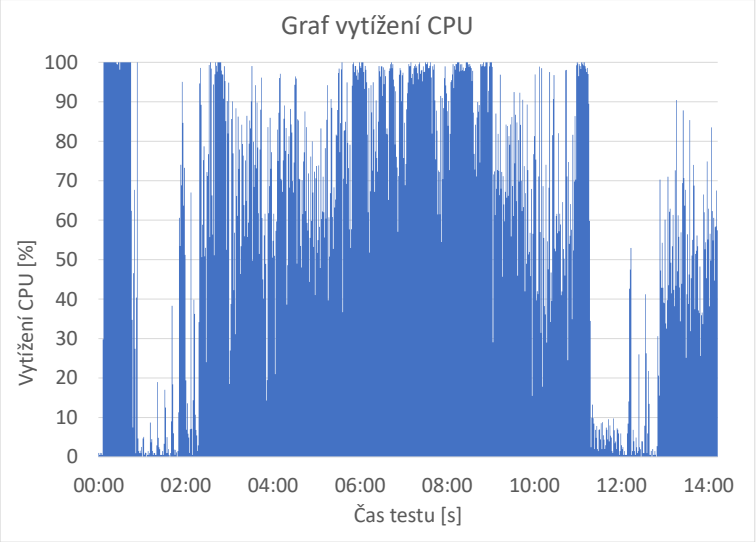
Obr. 3.6: Průběh doby sestavení hovoru (call setup) během testu



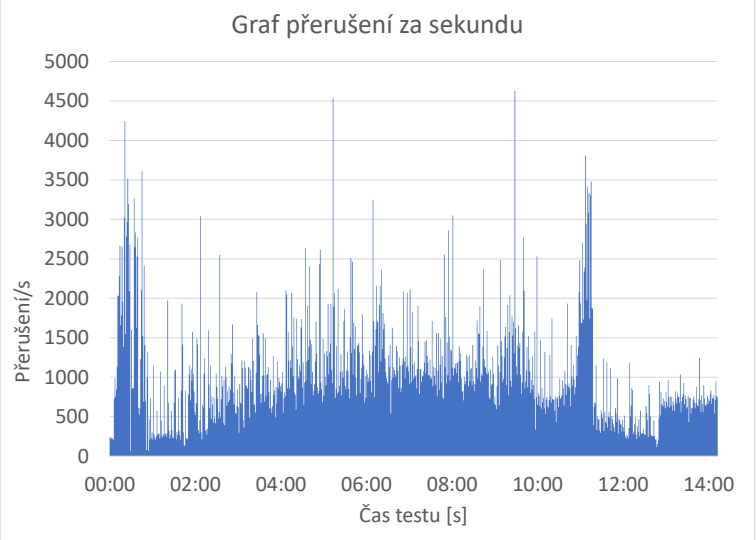
Obr. 3.7: Graf četnosti chyb během testu

Ze zaznamenaných informací o procesoru byl v Excelu vytvořen graf procentuálního vytížení CPU, kdy na počátku bylo maximální vytížení kvůli probíhající registraci uživatelů. Vytížení bylo po celou dobu testu poměrně vysoké, uprostřed testu bylo nejvyšší. Po skončení testu dobíhaly signalizace hovorů s velkým zpožděním, v tu dobu byl procesor vytížený málo a následovalo další vetší vytížení na konci testu, při odregistraci uživatelů. Celý průběh je zobrazen v grafu 3.8 Následně byl vytvořen graf přerušení za sekundu, kdy v inicializační části testu bylo zaznamenáno více přerušení za sekundu kvůli probíhající registraci. Téměř po celou dobu testu bylo rovnoměrně rozložené, na konci testu vzrostlo. Celý průběh je zobrazen v grafu 3.9.

Z tabulky změřených hodnot 3.2 vidíme, že úspěšně navázaných hovorů bylo méně než 80%. Z důvodu rostoucí doby potřebné k navázání hovoru, bylo provedeno méně pokusů o zahájení hovoru, než dle předpokladu vyplývajícího z nastavení scénáře testeru. Můžeme vidět, že skutečná délka testu byla výrazně vyšší, než teoretická délka testu. Délku testu negativně ovlivnila dlouhá doba odezvy SIP zpráv a dlouhá doba sestavení hovorů zejména v pozdější fázi testu. Vysoké vytížení procesoru a s tím související chyby, byly způsobeny zapnutím logováním u jednotlivých CSCF serverů.



Obr. 3.8: Graf vytížení CPU



Obr. 3.9: Graf přerušení za sekundu

Tab. 3.2: Změřené parametry

Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	733
Průměrný počet pokusů o navázaní hovoru za sekundu	0,88
Procento úspěšných hovorů	76,12 %
Počet chyb	177
Průměrná doba odezvy SIP zpráv	2,72 s
Průměrná doba sestavení hovoru	13,48 s
Průměrné zatížení CPU	61,58 %
Průměrný počet přerušení za sekundu	1036
Skutečná délka testu	13:53 min

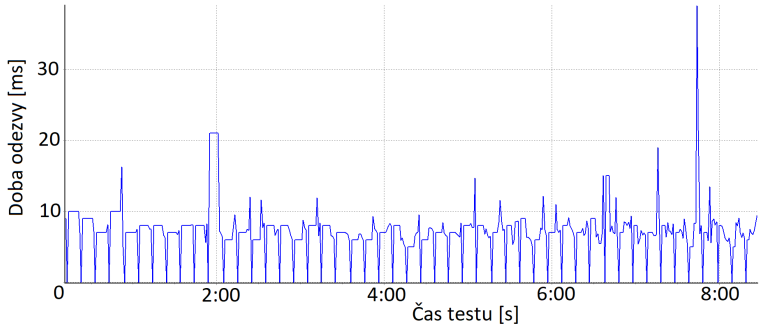
3.5.2 Open IMS Core s vypnutým logováním serveru

Velmi nízká stabilita Open IMS Core, byla způsobena zapnutým logováním u CSCF, které bylo vypnuto v konfiguračních souborech CSCF serverů. K vypnutí logování byla použita konfigurace uvedená ve výpisu 3.4.

Výpis 3.4: Vypnutí logování u cscf

```
debug=0
log_stderror=no
memlog=0
sip_warning=no
```

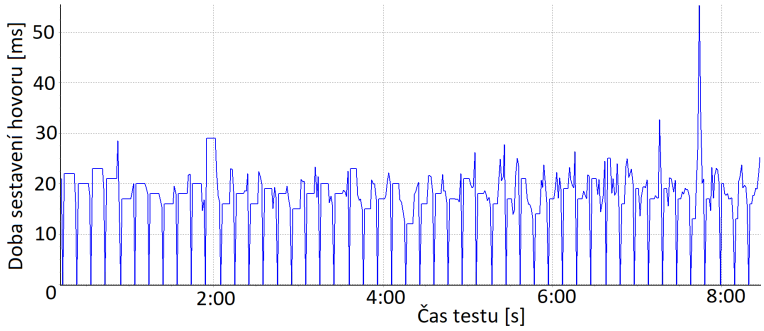
Doba odezvy se po celou dobu testu pohybovala mezi 8 ms a 10 ms s občasnými výkyvy k vyšším hodnotám, které nepřesáhly 40 ms. Doba odezvy byla stabilní po celou dobu testu. Celý průběh je zobrazen v grafu 3.10.



Obr. 3.10: Průběh doby odezvy (response time) během testu

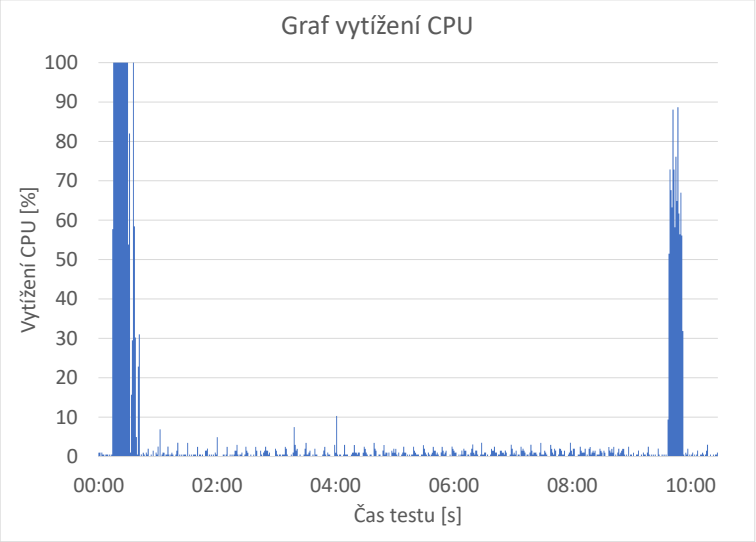
Průběh doby sestavení hovoru byl po celou dobu testu stabilní. Doba sestavení hovoru se po celou dobu testu pohybuje mezi 15 a 25 ms s občasnými výkyvy k nižším

nebo vyšším hodnotám, které nepřesahují 60 ms. Celý průběh je zobrazen v grafu 3.11.



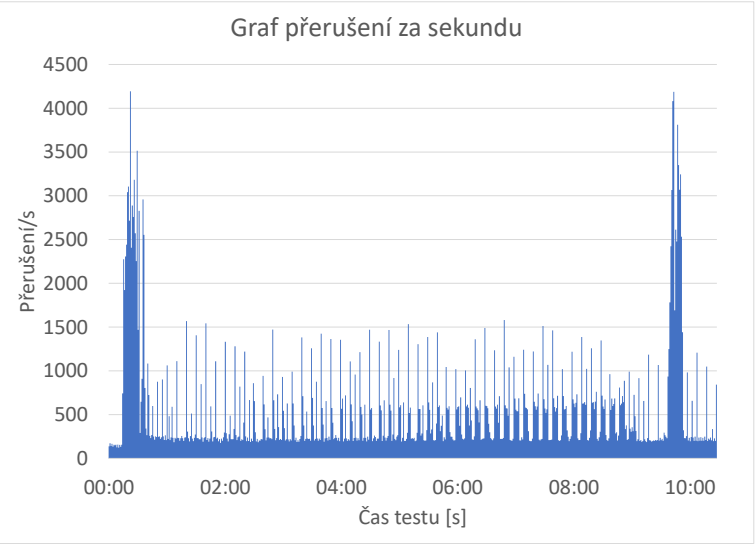
Obr. 3.11: Průběh doby sestavení hovoru (call setup) během testu

Ze zaznamenaných informací o procesoru, byl v Excelu vytvořen graf procentuálního vytížení CPU. V inicializační fázi testu bylo vytížení CPU maximální, kvůli probíhající registraci uživatelů. Vytížení bylo po celou dobu testu téměř zanedbatelné a nepřesáhlo úroveň 10 %. Na konci testu probíhala odregistrace uživatelů, což zapříčinilo prudký nárůst vytížení procesoru. Celý průběh je zobrazen v grafu 3.12.



Obr. 3.12: Graf vytížení CPU

Následně byl vytvořen graf přerušení za sekundu, podobně jako u vytížení CPU v inicializační fázi byl zaznamenán větší počet přerušení. Hodnoty počtu přerušení za sekundu byly po celou dobu testu rovnoměrně rozložené a nebyly vyšší než 1600 přerušení za sekundu. Na konci testu pozorujeme velký nárůst počtu přerušení za sekundu, což je způsobeno probíhající odregistrací. Celý průběh je zobrazen v grafu 3.13.



Obr. 3.13: Graf přerušení za sekundu

Tab. 3.3: Změřené parametry

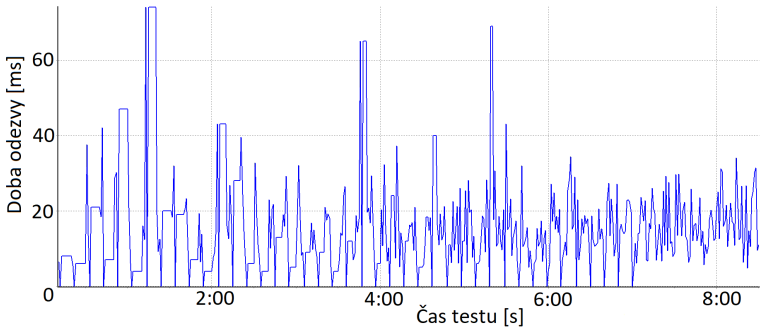
Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	5100
Průměrný počet pokusů o navázaní hovoru za sekundu	8,66
Procento úspěšných hovorů	100 %
Počet chyb	0
Průměrná doba odezvy SIP zpráv	8 ms
Průměrná doba sestavení hovoru	20 ms
Průměrné zatížení CPU	5,47 %
Průměrný počet přerušení za sekundu	555
Skutečná délka testu	9:51 min

Z tabulky změřených hodnot 3.3 vyplývá, že všechny hovory byly úspěšně navázány. Z naměřených hodnot vyplývá, že systém je pro zvolené zatížení stabilní.

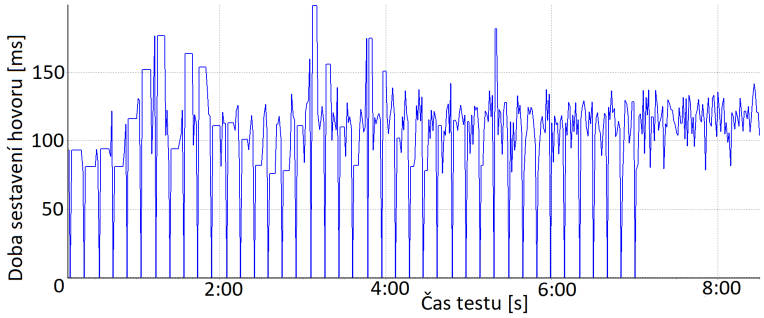
3.5.3 Project Clearwater

Hodnota doby odezvy se po celou dobu testu pohybovala do 50 ms, se třemi výjimkami kde doba odezvy nepřesáhla 80 ms. Doba odezvy byla pro zvolené zatížení stabilní. Celý průběh je zobrazen v grafu 3.14.

Hodnota doby sestavení hovoru byla nižší než 200 ms. Nebyly zaznamenány žádné větší výchylky, většina naměřených hodnot se pohybovala mezi 75 ms a 150 ms. Doba sestavení hovoru byla pro zvolené zatížení stabilní . Celý průběh je zobrazen v grafu 3.15.



Obr. 3.14: Průběh doby odezvy (response time) během testu



Obr. 3.15: Průběh doby sestavení hovoru (call setup) během testu

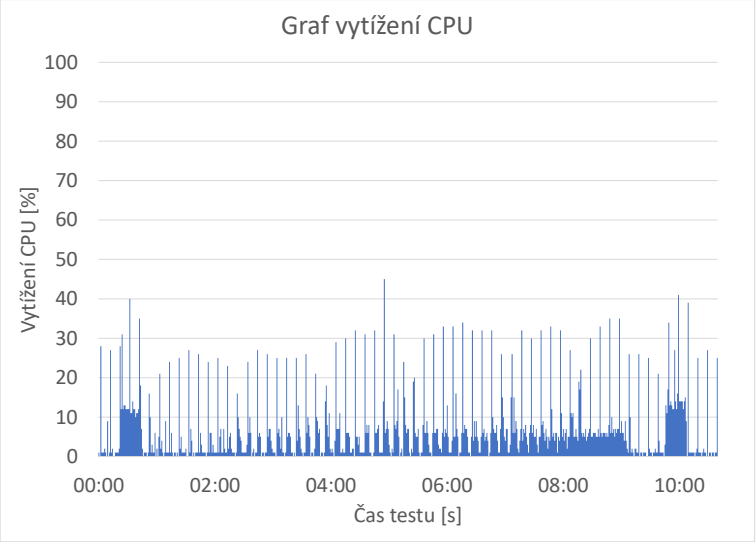
Ze zaznamenaných informací o procesoru, byl v Excelu vytvořen graf procentuálního vytížení CPU. Po celou dobu testu bylo vytížení procesoru na nízkých úrovních a nepřekročilo hranici 50 %. Rozložení skokových navýšení odpovídá délce jednotlivých schodů testu, takže je nutné brát v úvahu právě toto rozložení. Celý průběh je zobrazen v grafu 3.16.

Následně byl vytvořen graf přerušení za sekundu, v době registrace byl zaznamenán větší počet přerušení. Hodnoty přerušení po dobu testu nebyly vyšší než 2500 přerušení za sekundu. V závěru testu pozorujeme velký nárůst počtu přerušení za sekundu, což je způsobeno probíhající odregistrací. Celý průběh je zobrazen v grafu 3.17.

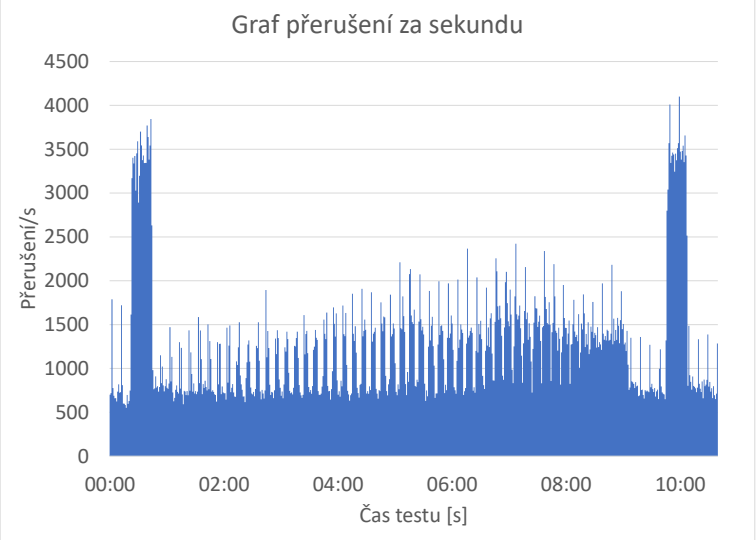
Z tabulky změřených hodnot 3.4 vyplývá, že všechny hovory byly úspěšně navázány. Z výsledků vyplývá, že se systém choval pro zvolené zatížení stabilně.

3.5.4 Shrnutí a porovnání výsledků

Open IMS Core byl testován dvakrát, jednou se zapnutým logováním CSCF serverů a podruhé s vypnutým logováním CSCF serverů. Clearwater byl testován ve své výchozí konfiguraci.
Testy ukázali, že zapnuté logování u Open IMS Core výrazně ovlivňuje výkon a sta-



Obr. 3.16: Graf vytížení CPU



Obr. 3.17: Graf přerušení za sekundu

Tab. 3.4: Změřené parametry

Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	5100
Průměrný počet pokusů o navázaní hovoru za sekundu	8,6
Procento úspěšných hovorů	100 %
Počet chyb	0
Průměrná doba odezvy SIP zpráv	16 ms
Průměrná doba sestavení hovoru	117 ms
Průměrné zatížení CPU	6,8 %
Průměrný počet přerušení za sekundu	1280
Skutečná délka testu	9:55 min

bilitu systému. Systém byl stabilní do 12 současně probíhajících hovorů, poté se stabilita začala zhoršovat. Následovalo postupné zahazování hovorů, prodloužení doby odezvy SIP zpráv dosahující maxima 20 s. Patrné bylo i prodloužení času potřebného k sestavení hovoru, které dosahovalo 100 s. Vysoké zatížení systému bylo pozorováno i na zachycených statistikách CPU.

Open IMS Core s vypnutým logováním u CSCF serverů se choval stabilně. Vypnutí logování u CSCF serverů mělo výrazný pozitivní vliv na výkon celého systému. Doba odezvy SIP zpráv byla po celou dobu testu stabilní a průměrně dosahovala hodnot 8 ms, což bylo ze všech testů prováděných pro schodové zatížení do 100 hovorů nejnižší. Rovněž doba sestavení hovoru je stabilní, průměrně dosahovala hodnot 20 ms a je rovněž nejnižší ze všech testů prováděných pro schodové zatížení do 100 hovorů. Zatížení systému bylo nepatrné, což můžeme pozorovat v zachycených statistikách CPU.

Project Clearwater se po celou dobu testu choval stabilně. Průměrná doba odezvy SIP zpráv byla 16 ms a průměrná doba sestavení hovoru byla 117 ms. Zatížení systému bylo nepatrné, což pozorujeme v zachycených statistikách CPU.

Nejlepších výsledků dosahoval Open IMS Core s vypnutým logováním CSCF serverů. Druhý v pořadí byl Project Clearwater, který dosahoval horších výsledků zejména u průměrné doby sestavení hovoru, která byla v porovnání s Open IMS Core s vypnutým logováním CSCF serverů téměř šestinásobná. Nejhůře na tom byl Open IMS Core se zapnutým logováním CSCF serverů, který byl pro zvolené zatížení prakticky nepoužitelný.

3.6 Schodovitý nárůst zátěže do 1000 hovorů

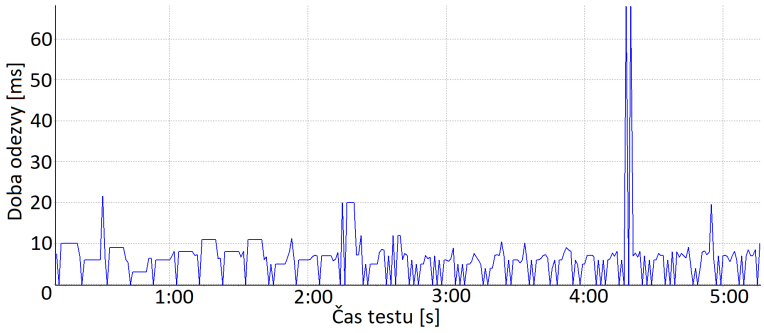
Časový scénář znázorňuje tabulka 3.5. Na základě výsledků předchozích testů, byla zvýšena zátěž 10 krát. Open IMS Core, byl nyní testován pouze s vypnutým logováním CSCF serverů.

Tab. 3.5: Časový scénář testu

Parametr	Hodnota
Délka hovoru	10s
Průběh	Schodovitý
Hodnota prvního schodku	10 hovorů
Výška schodku	10 hovorů
Počet schodků	100 hovorů
Délka testu	16:40 min

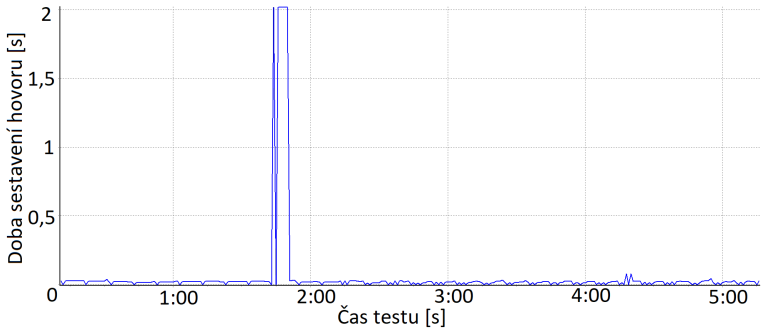
3.6.1 Open IMS Core

Test byl ukončen předčasně z důvodu pádu S-CSCF, v čase 5:30 při zátěži 330 hovorů. Důvod pádu nebyl zjištěn. Hodnoty byly změřeny do pádu S-CSF a krátký čas po pádu S-CSCF.
Hodnota doby odezvy se do doby pádu S-CSCF pohybovala kolem 10 ms s občasnými výkyvy k vyšším hodnotám dosahujících 20 ms, s výjimkou jedné větší výchylky k 70 ms. Z grafu vyplývá, že do doby pádu S-CSCF v čase 5:30, byla doba odezvy stabilní. Průběh do předčasného ukončení testu je zobrazen v grafu 3.18.



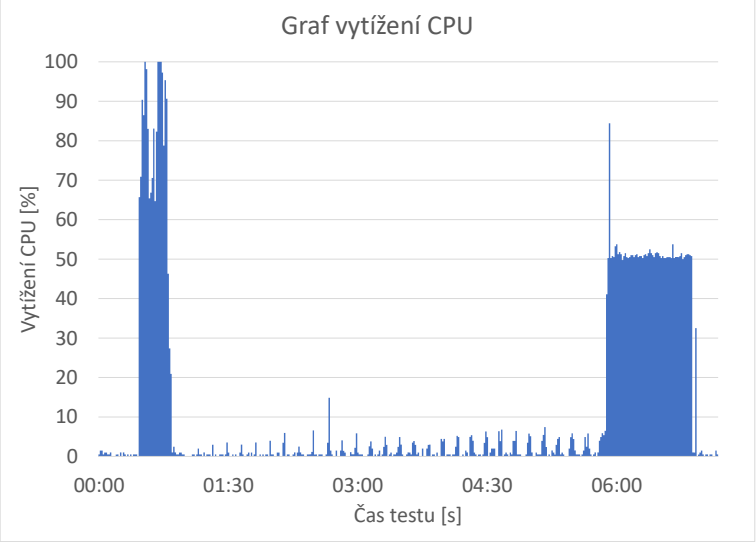
Obr. 3.18: Průběh doby odezvy (response time) během testu

Hodnota doby sestavení hovoru se do pádu S-CSCF pohybuje kolem 25 ms s ojedinělým výkyvem přesahujícím úroveň 2000 ms. Doba sestavení hovoru byla do pádu S-CSCF stabilní. Průběh do předčasného ukončení testu je zobrazen v grafu 3.19.



Obr. 3.19: Průběh doby sestavení hovoru (call setup) během testu

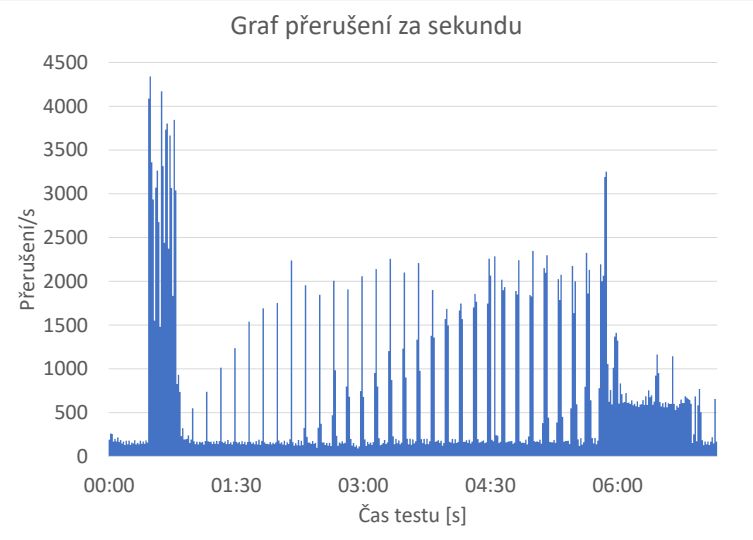
Ze zaznamenaných informací o procesoru, byl v Excelu vytvořen graf procentuálního vytížení CPU. Na počátku testu vytížení dosahuje až 100 % kvůli probíhající registraci uživatelů. Vytížení bylo do času pádu S-CSCF téměř zanedbatelné a pouze jednou přesáhlo úroveň 10 %. Po pádu S-CSCF vystoupalo zatížení mírně nad 50 %. Průběh do předčasného ukončení testu je zobrazen v grafu 3.20.



Obr. 3.20: Graf vytížení CPU

Následně byl vytvořen graf přerušení za sekundu, podobně jako u vytížení CPU v době registrace byl zaznamenán větší počet přerušení za sekundu. Počet přerušení za sekundu postupně stoupal k hodnotám kolem 2000 přerušení za sekundu, na kterých se v čase testu 1:40 ustálil. Před pádem S-CSCF pozorujeme velký nárůst a následný pokles vlivem pádu S-CSCF. Průběh do předčasného ukončení testu je zobrazen v grafu 3.21

Z tabulky změřených hodnot 3.6 vyplývá, že test byl ukončen předčasně. Příčinou předčasného ukončení testu byl pád S-CSCF, přes který procházela signalizace, tudíž



Obr. 3.21: Graf přerušení za sekundu

Tab. 3.6: Změřené parametry

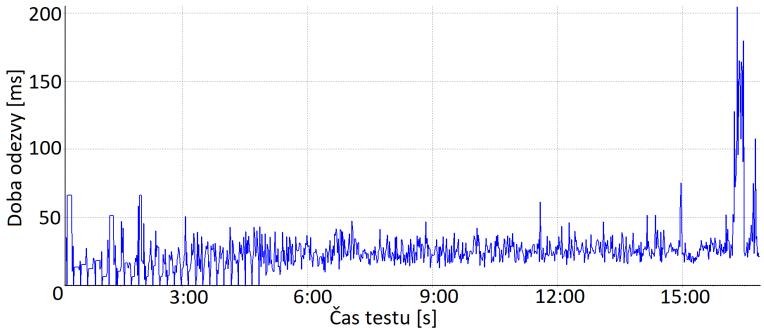
Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	11569
Průměrný počet pokusů o navázaní hovoru za sekundu	26,6
Procento úspěšných hovorů	85,14 %
Počet chyb	1454
Průměrná doba odezvy SIP zpráv	7 ms
Průměrná doba sestavení hovoru	27 ms
Průměrné zatížení CPU	12,37 %
Průměrný počet přerušení za sekundu	657
Skutečná délka testu	7:15 min

by dokončení testu nemělo smysl. Byla zjištěna maximální zátěž pro S-CSC a zároveň celý systém Open IMS Core, dosahující hodnoty 320 hovorů zahajovaných současně. Do doby pádu S-CSCF se systém choval stabilně a dosahoval dobrých výsledků v rychlosti sestavení hovoru.

3.6.2 Project Clearwater

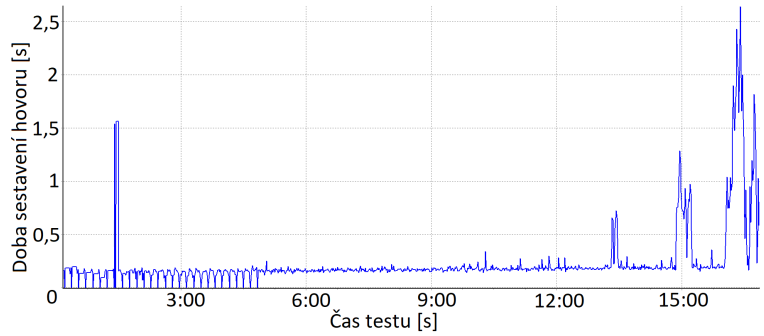
Doba odezvy SIP zpráv byla po většinu času testu stabilní a pohybovala se okolo hodnoty 30 ms, s výkyvy až k hladinám 80 ms. V závěru testu v čase od 16:00 doba odezvy dosahovala úrovní až 200 ms. Celý průběh je zobrazen v grafu 3.22.

V první polovině testu v čase 1:30 byla zaznamenána jedna výchylka přesahující 1500 ms. V závěru testu byly zaznamenány 3 větší výchylky, první z nich dosahovala



Obr. 3.22: Průběh doby odezvy (response time) během testu

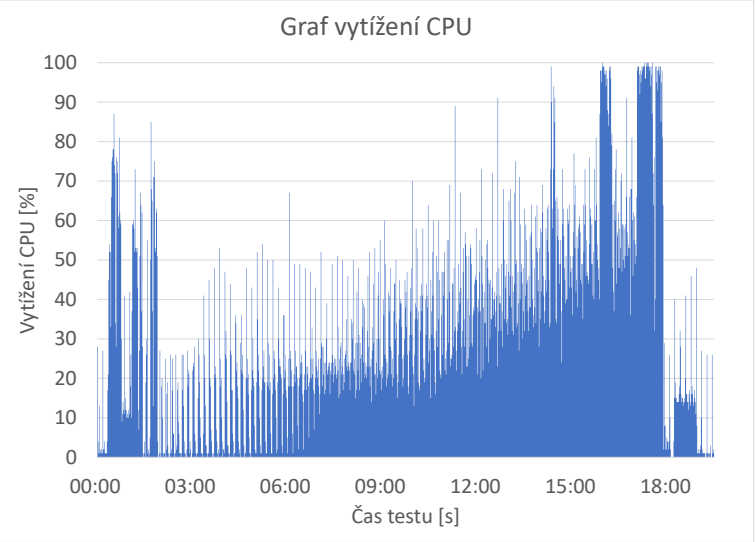
hodnot 700 ms, druhá 1300 ms a třetí 2500 ms. Bylo zaznamenáno menší množství výchylek, které nepřesáhli 350 ms. Po celou dobu testu byla doba sestavení hovoru téměř stabilníí, vyjímkou je závěrečná část testu. Celý průběh je zobrazen v grafu 3.23.



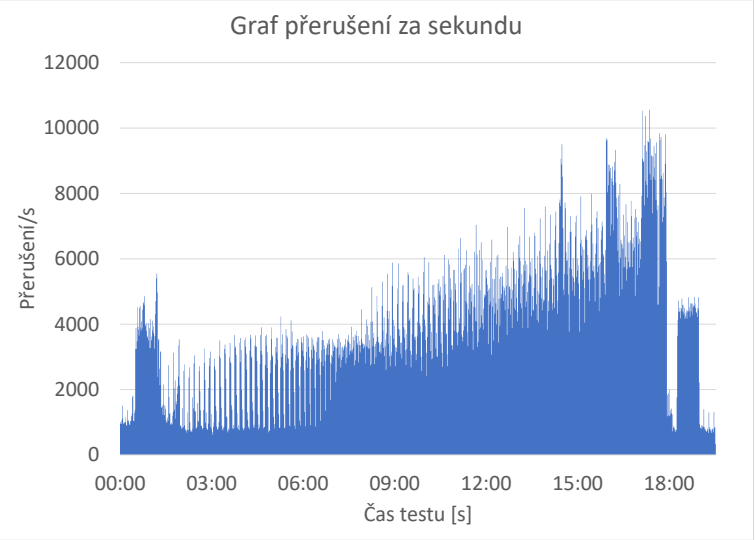
Obr. 3.23: Průběh doby sestavení hovoru (call setup) během testu

Ze zaznamenaných informací o procesoru, byl v Excelu vytvořen graf procentuálního vytížení CPU. Na počátku testu je patrné vysoké zatížení, dosahující 85 %, z důvodu registrace uživatelů. Následně se vytížení procesoru postupně zvyšovalo s rostoucí zátěží, na počátku testu asi do 3. minuty se pohybovalo kolem 30 %, postupně se zvyšovalo a na konci testu dosahovalo 100 %. Celý průběh je zobrazen v grafu 3.28. V době registrace byl zaznamenán větší počet přerušení přesahující hranici 5000 přerušení/s. Hodnoty počtu přerušení za sekundu se zvyšovali s rostoucí zátěží. Na konci testu byl zaznamenán vysoký nárůst podobně jako u grafu vytížení CPU. Celý průběh je zobrazen v grafu 3.25.

Z tabulky změřených hodnot 3.7 vyplývá, že všechny hovory byly úspěšně navázány a nevznikla žádná chyba. Z naměřených hodnot CPU je zřejmé, že 1000 realizovaných hovorů je blízko maximální zátěži, pro kterou Clearwater nezahazuje hovory.



Obr. 3.24: Graf vytížení CPU



Obr. 3.25: Graf přerušení za sekundu

Tab. 3.7: Změřené parametry

Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	97942
Průměrný počet pokusů o navázaní hovoru za sekundu	89.3
Procento úspěšných hovorů	100 %
Počet chyb	0
Průměrná doba odezvy SIP zpráv	28 ms
Průměrná doba sestavení hovoru	297 ms
Průměrné zatížení CPU	34,43 %
Průměrný počet přerušení za sekundu	3964
Skutečná délka testu	18:17 min

3.6.3 Shrnutí a porovnání výsledků

Open IMS Core byl nyní testován pouze s vypnutým logováním CSCF serverů. Clearwater byl testován ve své výchozí konfiguraci.

Test u Open IMS Core byl ukončen dříve z důvodu pádu S-CSCF serveru při zátěži 330 hovorů. Maximální zjištěná zátěž tedy odpovídá 320 hovorům. Příčina pádu S-CSCF nebyla blíže analyzována. Do doby pádu S-CSCF byl systém stabilní. Průměrná doba odezvy SIP zpráv byla 7 ms a průměrná doba sestavení hovoru byla 27 ms. V době pádu S-CSCF vidíme prudký nárůst vytížení CPU na 50 %, což jsou pravděpodobně pokusy o zachování funkce S-CSCF serveru.

Project Clearwater se téměř celou dobu testu choval stabilně, výjimkou je závěr testu, kdy se doba odezvy SIP zpráv a doba sestavení hovoru výrazně zvýšili. V závěru testu také pozorujeme vysokou zátěž CPU a počet přerušení za sekundu. Průměrná doba odezvy SIP zpráv byla 28 ms a průměrná doba sestavení hovoru byla téměř 300 ms.

Pro Open IMS Core byla zvolená zátěž příliš vysoká. Clearwater měl se zvolenou zátěží potíže až v závěru testu. Pro zvolené zatížení dosahoval lepších výsledků Clearwater, protože nedošlo k pádu žádného z komponentů, což je pro funkci systému klíčové. Tímto testováním bylo zjištěno, že Open IMS Core je pro testovanou zátěž nevyhovující i když dosahoval dobrých výsledků do pádu S-CSCF serveru. Clearwater je pro testovanou zátěž vyhovující.

3.7 Schodovitý nárůst zátěže do 2000 hovorů

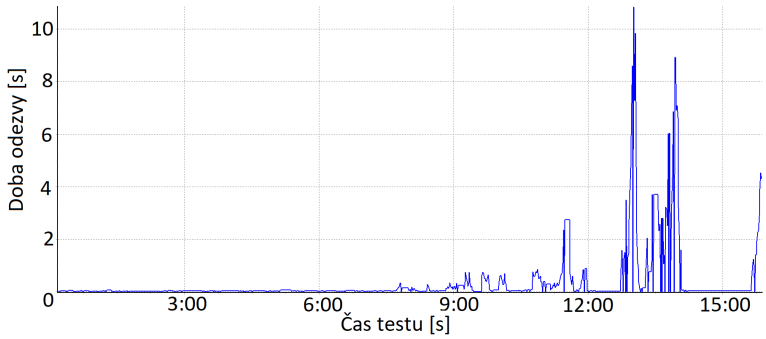
Časový scénář znázorňuje tabulka 3.8. Test byl realizován pouze u Clearwateru, protože u Open IMS Core by neměl smysl vzhledem k výsledkům předchozího testu.

Tab. 3.8: Časový scénář testu

Parametr	Hodnota
Délka hovoru	10 s
Průběh	Schodovitý
Počáteční hodnota schodku	20 hovorů
Výška schodku	20 hovorů
Počet schodků	100 hovorů
Délka testu	16:40 min

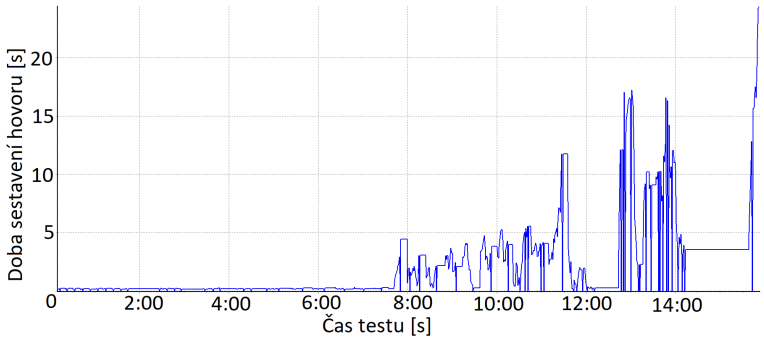
3.7.1 Project Clearwater

Doba odezvy SIP zpráv byla v první polovině testu stabilní, pohybovala se okolo hodnoty 30 ms, s menšími výkyvy. V druhé polovině testu se začala doba odezvy SIP zpráv zvyšovat, největší úroveň které dosáhla byla 13 s. Od času 11:30 je doba odezvy SIP zpráv příliš velká, což odpovídá i počátku zahazování hovorů v čase 11:37. Celý průběh je zobrazen v grafu 3.26.



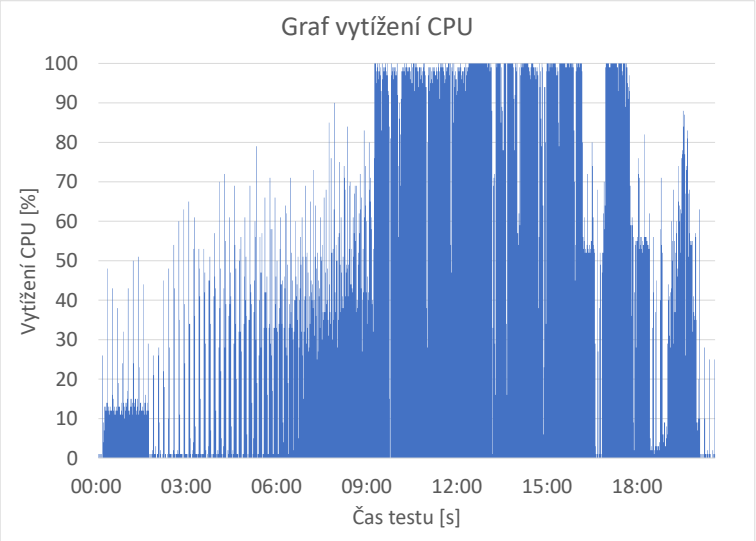
Obr. 3.26: Průběh doby odezvy (response time) během testu

Doba sestavení hovoru byla v první polovině testu nízká, pohybovala se kolem 150 ms. V druhé polovině testu od času 7:30 začala doba sestavení hovoru narůstat, ve svém maximu dosahovala hodnot 29 s. Na grafu je možno vypozorovat i zahazování hovorů, které dobu sestavení hovoru na chvíli stabilizovalo, ale následně vlivem přetížení systému a neustálého nárůstu zátěže zase roste, nejvýrazněji kolem času testu 12:00. Celý průběh je zobrazen v grafu 3.27.



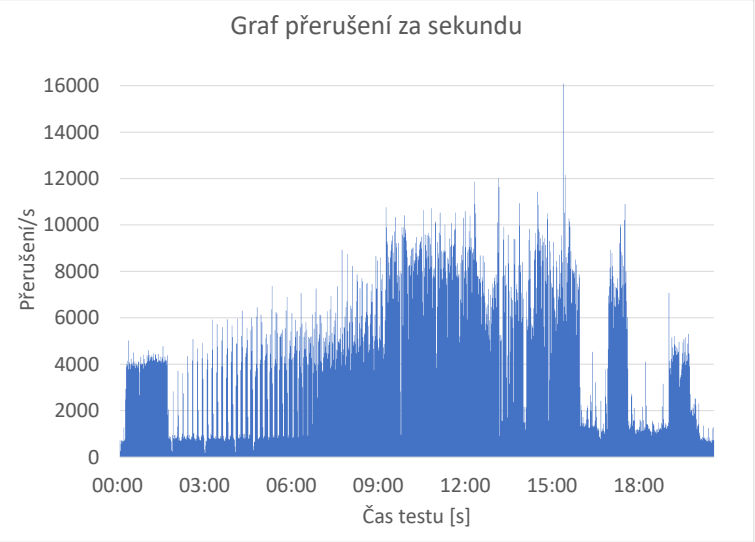
Obr. 3.27: Průběh doby sestavení hovoru (call setup) během testu

Ze zaznamenaných informací o procesoru, byl v Excelu vytvořen graf procentuálního vytížení CPU. Na počátku testu lze pozorovat vytížení dosahující až 50 % z důvodu registrace uživatelů. Následně se vytížení procesoru postupně zvyšovalo s rostoucí zátěží. Od času testu 9:00 se drželo blízko hodnotě 100 %, na této úrovni se drželo téměř po celý zbytek testu, v závěru testu kleslo k úrovním 85 %. Celý průběh je zobrazen v grafu 3.28.



Obr. 3.28: Graf vytížení CPU

V době registrace byl zaznamenán vyšší počet přerušení přesahující hranici 5000 přerušení za sekundu. Hodnoty počtu přerušení za sekundu se zvyšovali s rostoucí zátěží. V první polovině testu do času 8:00 minut počet přerušení dosahoval úrovní kolem 4000 přerušení za sekundu. Následně se počet přerušení za sekundu postupně zvyšoval a v závěru testu dosahoval úrovní přesahujících 11000 přerušení za sekundu. V závěru testu pozorujeme větší počet přerušení dosahujících téměř 5000 přerušení za sekundu při odregistraci uživatelů. Celý průběh je zobrazen v grafu 3.29.



Obr. 3.29: Graf přerušení za sekundu

Tab. 3.9: Změřené parametry

Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	147518
Průměrný počet pokusů o navázaní hovoru za sekundu	132,5
Procento úspěšných hovorů	69,32 %
Počet chyb	45256
Průměrná doba odezvy SIP zpráv	424 ms
Průměrná doba sestavení hovoru	953 ms
Průměrné zatížení CPU	54,83 %
Průměrný počet přerušení za sekundu	4831
Skutečná délka testu	18:33 min

Od času testu 11:37 dochází k zahazování hovorů – chyba `connect failed`, čas 11:37 odpovídá zátěži 1380 hovorů. Od doby prvního zahození se snížil i počet realizovaných hovorů, který se držel mezi 700 až 1000 hovory. Tímto testem byla zjištěna maximální zátěž pro Clearwater, dosahující hodnoty 1380 hovorů. Naměřené hodnoty testu jsou zobrazeny v tabulce 3.9.

3.7.2 Shrnutí

Na základě minulého testu byl v případě schodovitého zvyšování zátěže do 2000 hovorů testován pouze Clearwater.
Project Clearwater se v první polovině testu choval stabilně. Od zátěže odpovídající 1040 hovorům se začíná zvyšovat doba potřebná k sestavení hovoru a také doba

odezvy SIP zpráv. Od zátěže odpovídající 1380 hovorům dochází i k zahazování hovorů. Naměřené parametry CPU potvrzují, že jsme překonali limitní zátěž, kterou Clearwater ještě zvládne obsloužit a pro kterou je stabilní.

3.8 Zátěž s Poissonovým rozdělením

Poissonovo rozdělení zátěže nejlépe simuluje reálnou zátěž systému, rozložení hovorů je náhodné s průměrným počtem hovorů za sekundu. Nastavujeme zde call rate (počet hovorů za sekundu) s Poissonovým rozdělením zátěže a délku hovoru. Časový scénář obou testů je zaznačen v tabulce 3.10. Zátěž je zvolena taková, aby nebyla příliš velká pro oba testované systémy. Pro demonstraci rozložení zátěže byly přidány grafy s rozložením pokusů o navázání hovoru v čase testu.

Tab. 3.10: Časový scénář testu

Parametr	Hodnota
Délka hovoru	10 s
Průběh	Poissonovo rozdělení
Průměrný počet hovorů za sekundu	5
Délka testu	10:00

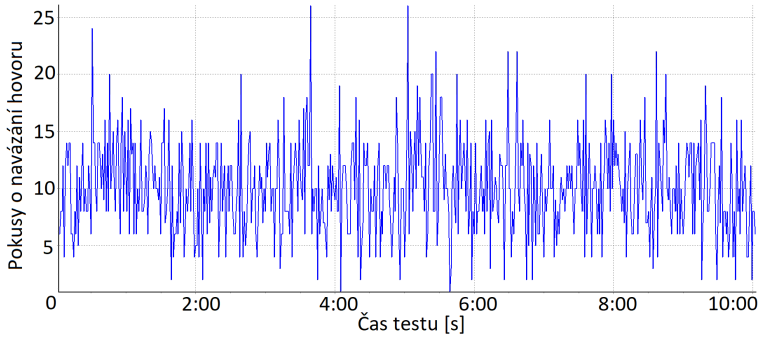
3.8.1 Open IMS Core

Rozložení počtu pokusů o navázání hovoru se drželo kolem zvolené hodnoty 10 pokusů o hovor za sekundu, kdy byl jeden celý hovor je tvořen dvěma pokusy o navázání hovoru (jeden ze strany volajícího a druhý ze strany volaného). Zaznamenané hodnoty se pohybovali od 26 pokusů o hovor za sekundu k 0 pokusům o hovor za sekundu. Celý průběh je zobrazen v grafu 3.30.

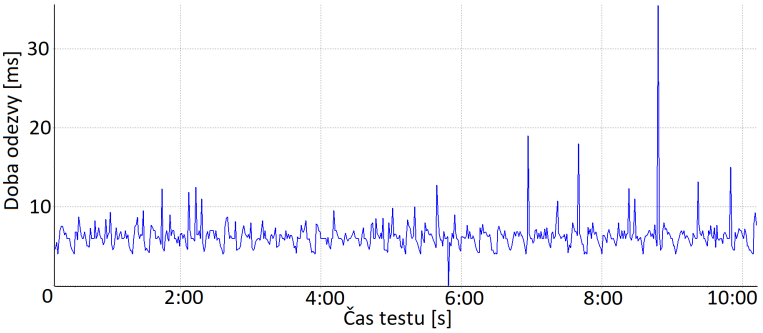
Hodnota doby odezvy se po celou dobu testu pohybovala kolem 6 ms. Větší výkyvy byly zaznamenány převážně v druhé polovině testu, kdy maximální hodnota doby odezvy byla 65 ms. Celý průběh je zobrazen v grafu 3.31.

Doba potřebná k sestavení hovoru se pohybovala kolem 16 ms s jedním výkyvem dosahujícím 120 ms. Doba potřebná k sestavení hovoru byla po celou dobu testu stabilní. Celý průběh je zobrazen v grafu 3.32.

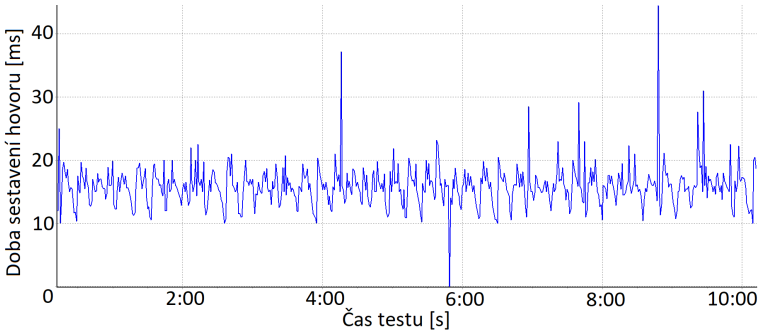
Ze zaznamenaných informací o procesoru, byl v Excelu vytvořen graf procentuálního vytížení CPU. Na počátku testu vytížení dosahuje až 100 %, kvůli probíhající registraci uživatelů. Vytížení bylo po dobu testu téměř zanedbatelné a pouze jednou přesáhlo úroveň 10 %. V závěru testu dosahovalo hodnot téměř 100 %, kvůli probíhající odregistraci. Celý průběh je zakreslen v grafu 3.33.



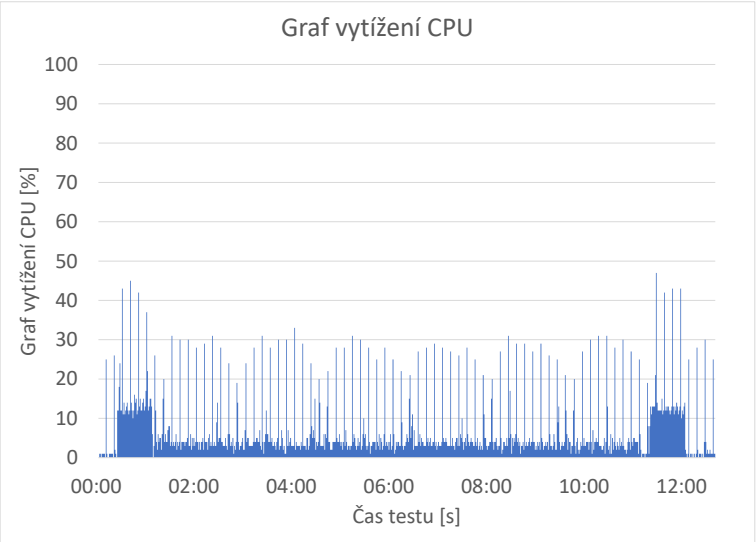
Obr. 3.30: Průběh generace počtu pokusů o navázání hovoru (call attempts) během testu



Obr. 3.31: Průběh doby odezvy (response time) během testu

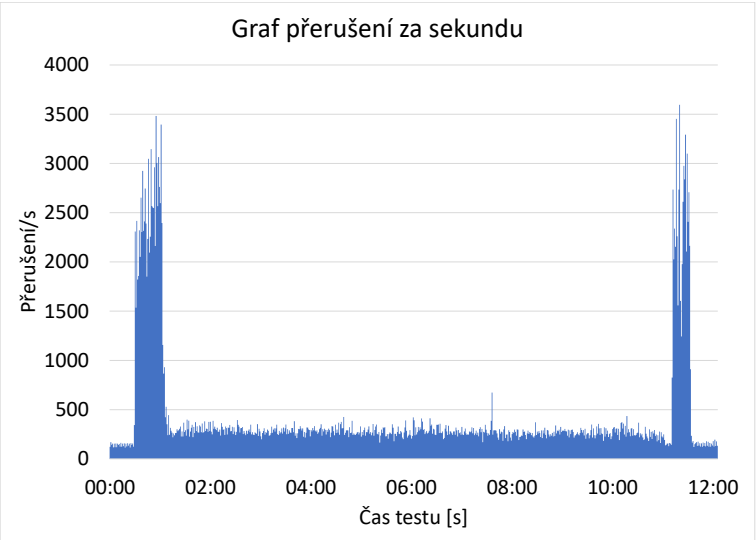


Obr. 3.32: Průběh doby sestavení hovoru (call setup) během testu



Obr. 3.33: Graf vytížení CPU

Následně byl vytvořen graf přerušení za sekundu, podobně jako u vytížení CPU v době registrace byl zaznamenán větší počet přerušení za sekundu. Počet přerušení za sekundu se po dobu generování hovorů držel na nízkých úrovních a pouze jednou hodnota přerušení za sekundu přesáhla 500 přerušení. V závěru testu byl zaznamenán zvýšený počet přerušení za sekundu, kvůli probíhající odregistraci. Celý průběh je zakreslen v grafu 3.34



Obr. 3.34: Graf přerušení za sekundu

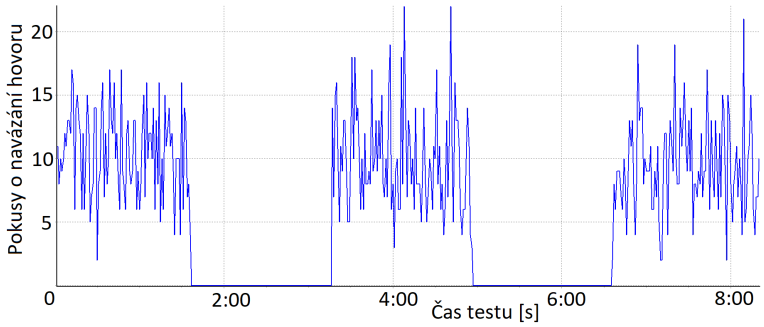
Z tabulky změřených hodnot 3.11 vyplývá, že všechny hovory byly úspěšně navázány. Z naměřených hodnot vyplývá, že systém je pro zvolené zatížení stabilní.

Tab. 3.11: Změřené parametry

Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	6098
Průměrný počet pokusů o navázaní hovoru za sekundu	9,2
Procento úspěšných hovorů	100 %
Počet chyb	0
Průměrná doba odezvy SIP zpráv	6 ms
Průměrná doba sestavení hovoru	16 ms
Průměrné zatížení CPU	7,35 %
Průměrný počet přerušení za sekundu	435
Skutečná délka testu	11:05 min

3.8.2 Project Clearwater

Rozložení počtu pokusů se drželo kolem zvolené hodnoty 10 pokusů o hovor za sekundu, byly zaznamenány periodické výpadky v generaci hovorů, které vždy trvali téměř dvě minuty. Jeden celý hovor je tvořen dvěma pokusy o navázání hovoru (jeden ze strany volajícího a druhý ze strany volaného). Zaznamenané hodnoty se pohybovali od 24 pokusů o hovor za sekundu k 0 pokusům o hovor za sekundu. Celý průběh je zobrazen v grafu 3.35.

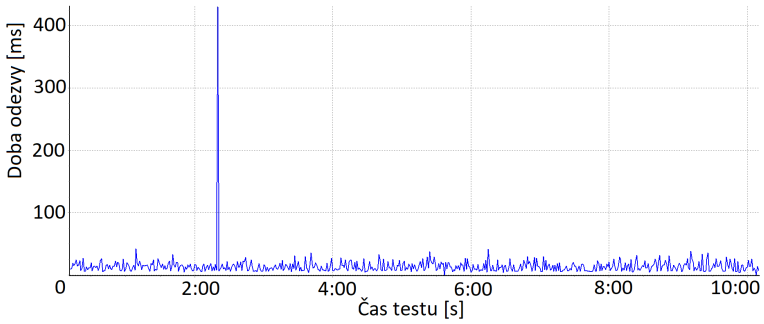


Obr. 3.35: Průběh generace počtu pokusů o navázání hovoru (call attempts) během testu

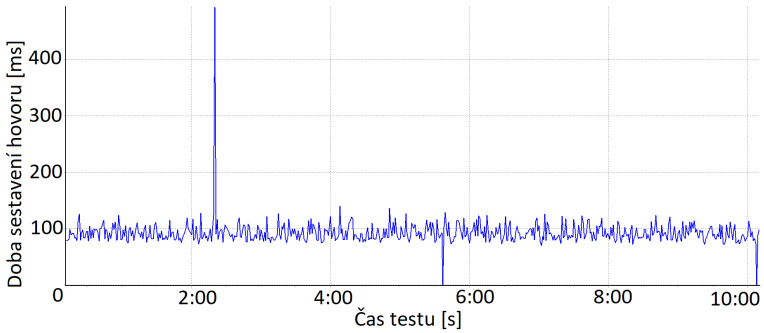
Doba odezvy SIP zpráv byla stabilní a pohybovala se okolo hodnoty 15 ms, s jedním výkyvem k hladině 420 ms. Celý průběh je zobrazen v grafu 3.36.

Doba sestavení hovoru se pohybovala kolem 100 ms. Hodnoty doby sestavení hovoru byly po celou dobu testu stabilní, s výjimkou jedné výchylky k hodnotám 480 ms. Celý průběh je zobrazen v grafu 3.37.

Ze zaznamenaných informací o procesoru, byl v Excelu vytvořen graf procentuálního vytížení CPU. Na počátku testu je zaznamenáno vyšší vytížení CPU z důvodu



Obr. 3.36: Průběh doby odezvy (response time) během testu



Obr. 3.37: Průběh doby sestavení hovoru (call setup) během testu

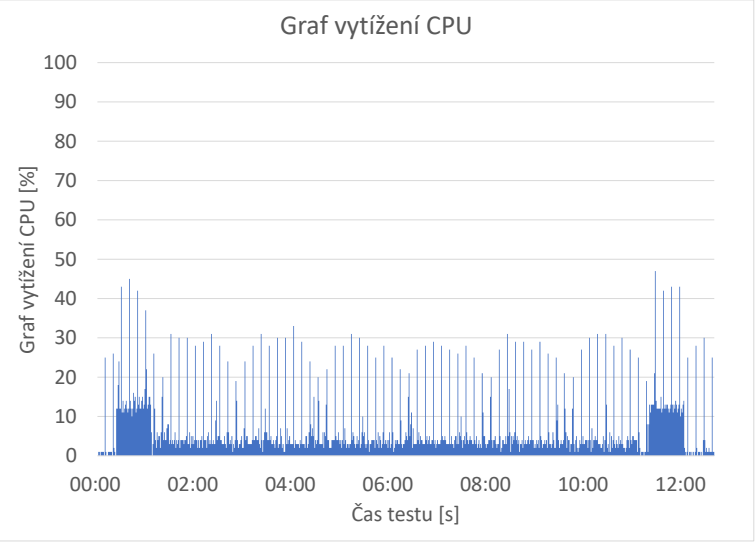
probíhající registrace nepřesahující 50 %. Po dobu testu bylo vytížení stabilní a dosahovalo hodnot nižších než 35 %. Na konci testu bylo opět vyšší vytížení kvůli probíhající odregistraci. Celý průběh je zobrazen v grafu 3.38.

V době registrace byl zaznamenán větší počet přerušení nepřesahující hranici 5000 přerušení/s. Hodnoty přerušení za sekundu byli po celou dobu testu stabilní a pohybovali se od 700 do 2200. Na konci testu byl zaznamenán vyšší počet přerušení kvůli probíhající odregistraci. Celý průběh je zobrazen v grafu 3.39.

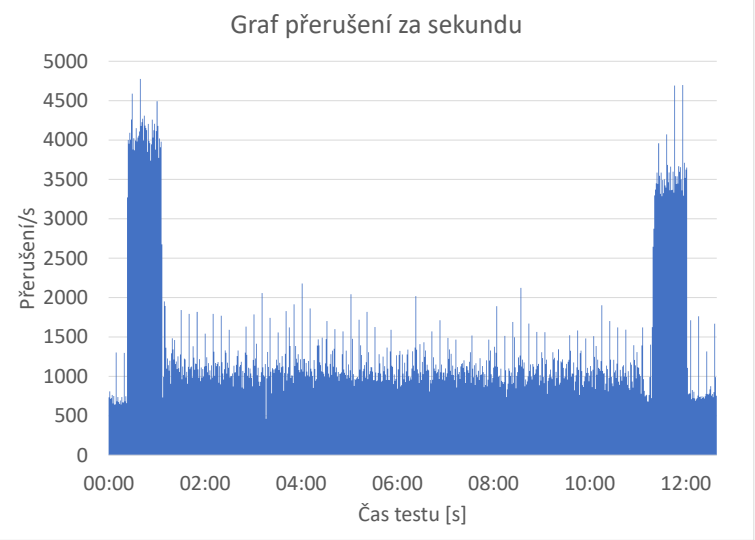
Z tabulky změřených hodnot 3.12, vidíme že všechny hovory byly úspěšně navázány a nevznikla žádná chyba. Systém se pro dané zatížení choval stabilně.

3.8.3 Shrnutí a porovnání výsledků

Open IMS Core byl nyní testován s vypnutým logováním CSCF serverů. Clearwater byl testován ve své výchozí konfiguraci.
Open IMS Core byl po celou dobu testu stabilní. Průměrná doba odezvy SIP zpráv byla 6 ms, což je nejnižší ze všech proběhlých testů. Průměrná doba sestavení hovoru byla 16 ms, což je rovněž nejnižší hodnota ze všech proběhlých testů. Zaznamenané hodnoty CPU se pohybovali na nízkých úrovních.
Project Clearwater byl po celou dobu testu stabilní. Průměrná doba odezvy SIP



Obr. 3.38: Graf vytížení CPU



Obr. 3.39: Graf přerušení za sekundu

Tab. 3.12: Změřené parametry

Změřený parametr	Hodnota
Počet pokusů o navázaní hovoru	5920
Průměrný počet pokusů o navázaní hovoru za sekundu	8,6
Procento úspěšných hovorů	100 %
Počet chyb	0
Průměrná doba odezvy SIP zpráv	15 ms
Průměrná doba sestavení hovoru	95 ms
Průměrné zatížení CPU	7,1 %
Průměrný počet přerušení za sekundu	1409
Skutečná délka testu	11:27 min

zpráv byla 15 ms. Průměrná doba sestavení hovoru byla 95 ms. Zaznamenané hodnoty CPU se pohybovali na nízkých úrovních.
Pro zvolené zatížení dosahoval lepších výsledků Open IMS Core, kdy signalizace byla rychlejší. Oba testované systémy jsou vyhovující pro zvolenou zátěž.

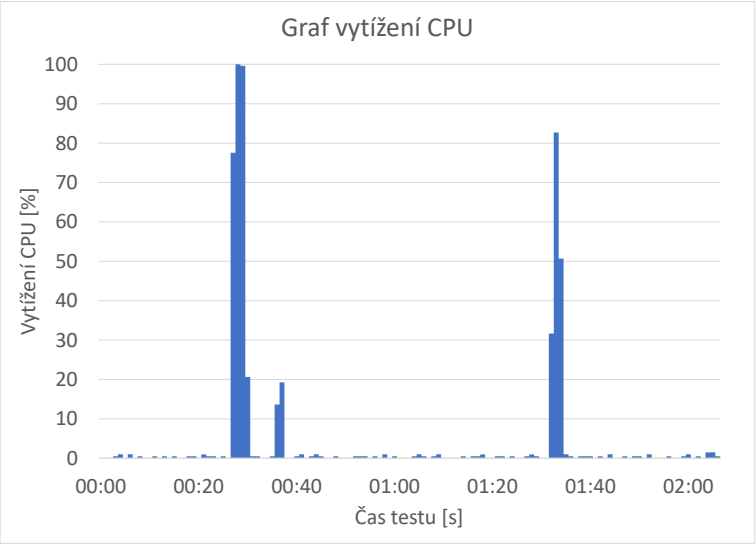
3.9 Měření parametrů registrace

V následujících testech bude porovnávána doba potřebná k úspěšné registraci, při různém počtu registrovaných uživatelů. Sledován bude také vliv registrace na vytížení procesoru a počet přerušení procesoru za sekundu. Test bude probíhat pouze jako počáteční registrace stanoveného počtu uživatelů, nebudou generovány žádné hovory. Tímto testem bude sledován vliv určitého počtu pokusů o registraci na testovaný systém a průměrnou dobu registrace.

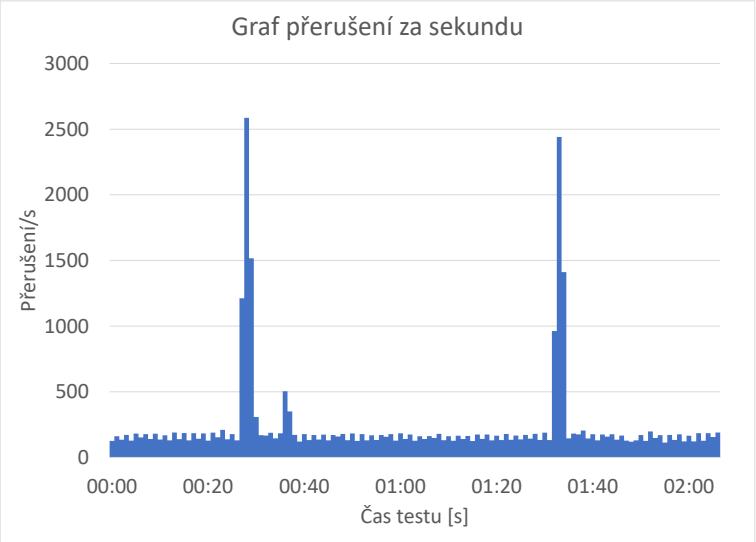
3.9.1 Open IMS Core 100 registrací

Z grafu vytížení CPU vidíme, že při registraci 100 uživatelů bylo vytížení procesoru 100 %. Při odregistraci dosahovala úroveň vytížení procesoru 82 %. Procesor byl vytížen pouze v krátkých časových úsecích, kdy Open IMS Core zpracovával registrace. Při pohledu na graf vidíme, že zpracování registrace i odregistrace probíhalo pouze krátký časový úsek, při kterém bylo značné vytížení procesoru. Celý průběh je zobrazen v grafu 3.40.

Z grafu přerušení za sekundu vidíme, že při registraci 100 uživatelů byl maximální počet přerušení procesoru 2580. Při odregistraci dosahoval počet přerušení za sekundu hodnot 2450. Celý průběh je zobrazen v grafu 3.40.



Obr. 3.40: Graf vytížení CPU



Obr. 3.41: Graf přerušení za sekundu

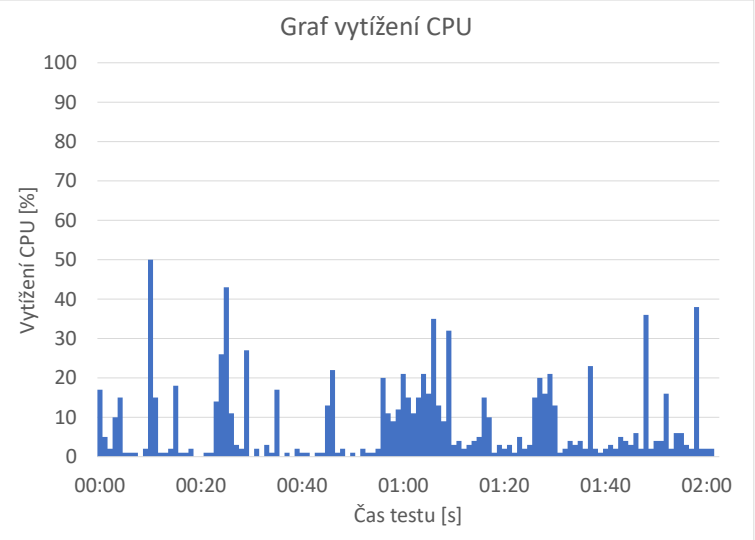
Tab. 3.13: Změřené parametry

Změřený parametr	Hodnota
Průměrná doba registrace	657 ms
Průměrné zatížení CPU	4,2 %
Průměrný počet přerušení za sekundu	234
Skutečná délka testu	1:35 min

Z vynesených grafů lze vyčíst, že zátěž systému je velmi krátká, ale je vysoká i při takhle nízkém počtu registrací. Průměrná délka registrace jednoho uživatele je 657 ms. Změřené parametry testu jsou zapsány v tabulce 3.13.

3.9.2 Clearwater 100 registrací

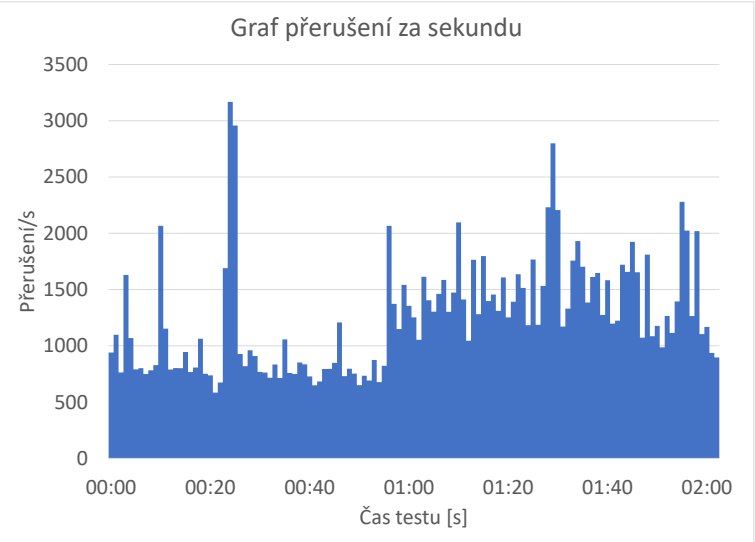
Maximální hodnota vytížení procesoru dosahovala 50 %, která byla zaznamenána při registraci uživatelů. Při odregistraci dosahovalo vytížení procesoru nejvyšší úrovně 38 %. Celý průběh je zobrazen v grafu 3.42.



Obr. 3.42: Graf vytížení CPU

Maximální hodnota dosáhla úrovně blížící se 3200 přerušení za sekundu, byla zaznamenána při registraci uživatelů. Při odregistraci dosahoval počet přerušení za sekundu maximální hodnoty blížící se 2800. Celý průběh je zobrazen v grafu 3.43.

Z vynesených grafů lze vyčíst, že zátěž systému je nízká a je rozložena do delšího časového úseku. Průměrná délka registrace jednoho uživatele je 35 ms. Změřené parametry jsou zapsány v tabulce 3.14.



Obr. 3.43: Graf přerušení za sekundu

Tab. 3.14: Změřené parametry

Změřený parametr	Hodnota
Průměrná doba registrace	35 ms
Průměrné zatížení CPU	7,59 %
Průměrný počet přerušení za sekundu	1248
Skutečná délka testu	1:35 min

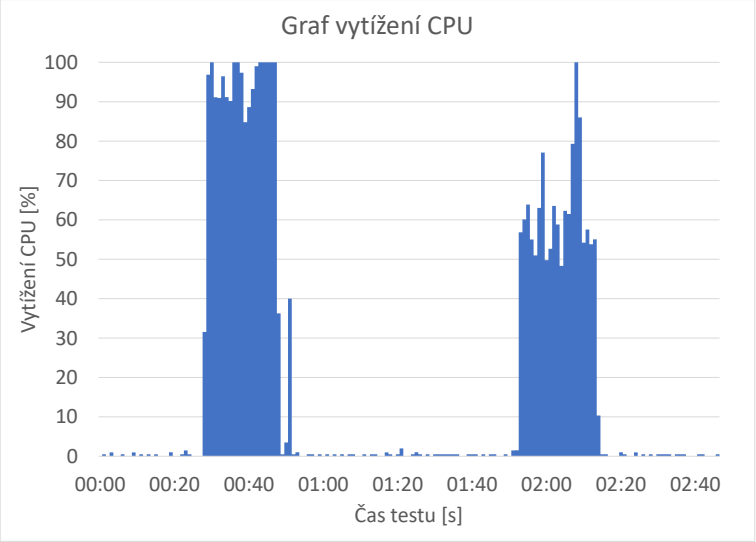
3.9.3 Open IMS Core 1000 registrací

Z grafu vytížení CPU vidíme, že při registraci 1000 uživatelů bylo vytížení procesoru 100 %, nebo blížící se 100 % po dobu 20 s. Při odregistraci dosáhla úroveň vytížení procesoru 100 % ve svém maximu. Celý průběh je zobrazen v grafu 3.44.

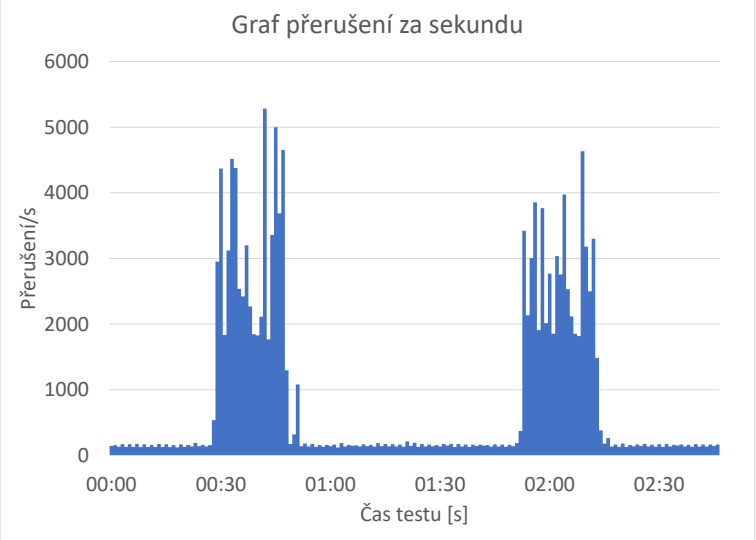
Z grafu přerušení za sekundu vidíme, že při registraci 1000 uživatelů byl maximální počet přerušení procesoru 5350. Při odregistraci počet přerušení za sekundu dosáhl maximální hodnoty 4650. Celý průběh je zobrazen v grafu 3.44.

Tab. 3.15: Změřené parametry

Změřený parametr	Hodnota
Průměrná doba registrace	623 ms
Průměrné zatížení CPU	19,72 %
Průměrný počet přerušení za sekundu	848
Skutečná délka testu	1:55 min



Obr. 3.44: Graf vytížení CPU

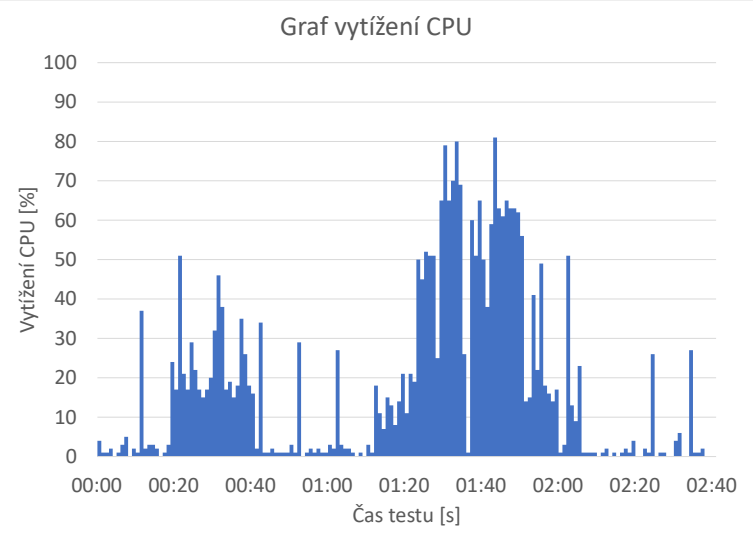


Obr. 3.45: Graf přerušení za sekundu

Z vynesených grafů lze vyčíst, že zátěž systému je vysoká po dobu registrace a odregistrace. Průměrná délka registrace jednoho uživatele je 623 ms. Změřené parametry jsou zapsány v tabulce 3.15.

3.9.4 Clearwater 1000 registrací

Maximální hodnota vytížení procesoru dosahovala 51 % při registraci uživatelů. Při odregistraci dosahovalo vytížení procesoru nejvyšší úrovně 81 %. Celý průběh je zobrazen v grafu 3.46.



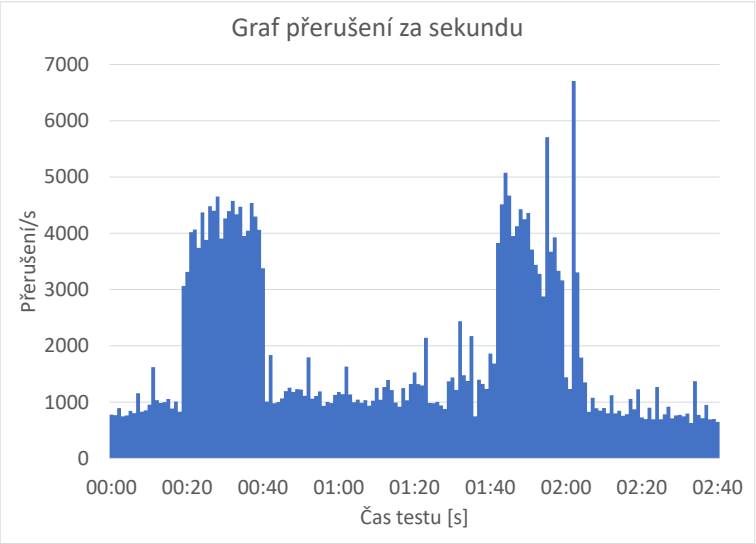
Obr. 3.46: Graf vytížení CPU

Při registraci dosahovaly hodnoty počtu přerušení za sekundu 4500. Při odregistraci dosahoval počet přerušení za sekundu maximální hodnoty blížící se 6800. Celý průběh je zobrazen v grafu 3.47.

Tab. 3.16: Změřené parametry

Změřený parametr	Hodnota
Průměrná doba registrace	18 ms
Průměrné zatížení CPU	17,7 %
Průměrný počet přerušení za sekundu	1874
Skutečná délka testu	1:59 min

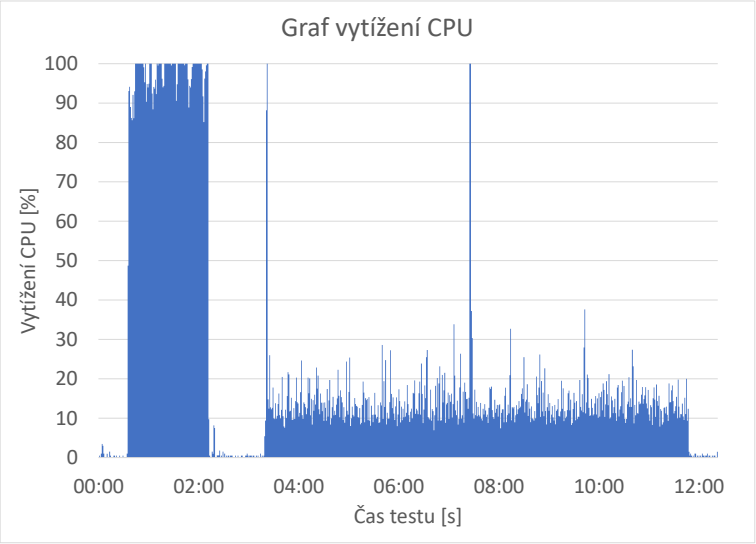
Z vynesených grafů lze vyčíst, že zátěž systému je vysoká a je rozložena do poměrně krátkého časového úseku. Průměrná délka registrace jednoho uživatele je 18 ms. Změřené parametry jsou zapsány v tabulce 3.16.



Obr. 3.47: Graf přerušení za sekundu

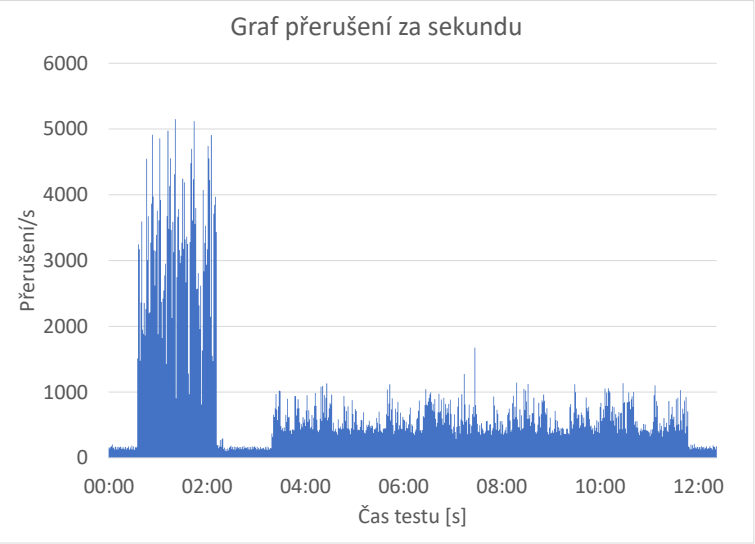
3.9.5 Open IMS Core 5000 registrací

Z grafu vytížení CPU vidíme, že při registraci 5000 uživatelů bylo vytížení procesoru 100 %, nebo blížící se 100 % po dobu 96 s. Při odregistraci dosáhla úroveň vytížení procesoru 100 % ve svém maximu, ale šlo pouze o ojedinělé výkyvy. Vytížení procesoru při odregistraci se pohybovalo kolem 15 % a trvalo 8,5 minuty. Celý průběh je zobrazen v grafu 3.48.



Obr. 3.48: Graf vytížení CPU

Z grafu přerušení za sekundu vidíme, že při registraci 5000 uživatelů byl maximální počet přerušení procesoru 5200. Při odregistraci počet přerušení za sekundu dosáhl maximální hodnoty 1650. Celý průběh je zobrazen v grafu 3.48.



Obr. 3.49: Graf přerušení za sekundu

Tab. 3.17: Změřené parametry

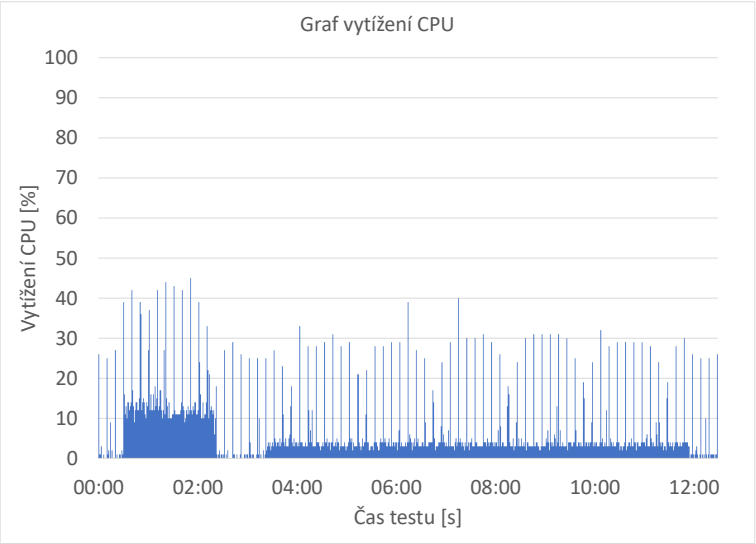
Změřený parametr	Hodnota
Průměrná doba registrace	1054 ms
Průměrné zatížení CPU	22,59 %
Průměrný počet přerušení za sekundu	850
Skutečná délka testu	10:09 min

Z vynesených grafů lze vyčíst, že zátěž systému je vysoká po dobu registrace, odregistrace byla rozprostřena ve větším časovém úseku – 8,5 minuty. Průměrná délka registrace jednoho uživatele je 1054 ms. Změřené parametry jsou zapsány v tabulce 3.17.

3.9.6 Clearwater 5000 registrací

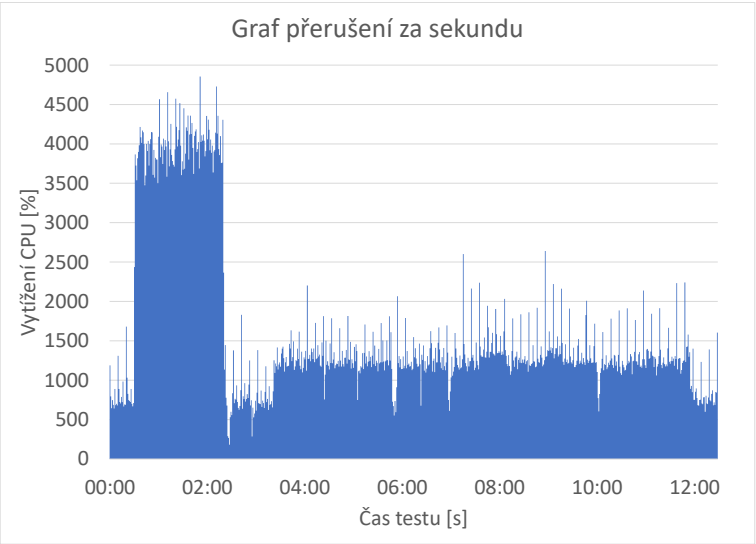
Maximální hodnota vytížení procesoru dosahovala 45 % při registraci uživatelů, která trvala asi 2 minuty. Při odregistraci dosahovalo vytížení procesoru nejvyšší úrovně 40 %. Celý průběh je zobrazen v grafu 3.50.

Při registraci dosahovaly hodnoty počtu přerušení za sekundu až 4700 a průměrně se pohybovali kolem 4000 přerušení za sekundu. Při odregistraci dosahoval



Obr. 3.50: Graf vytížení CPU

počet přerušení za sekundu maximální hodnoty blížící se 2700 a průměrně dosahoval hodnoty 1300 přerušení az sekundu. Celý průběh je zobrazen v grafu 3.51.



Obr. 3.51: Graf přerušení za sekundu

Z vynesených grafů lze vyčíst, že zátěž systému je poměrně nízká a je rozložena do delšího časového úseku. Průměrná délka registrace jednoho uživatele je 16 ms. Změřené parametry jsou zapsány v tabulce 3.18.

Tab. 3.18: Změřené parametry

Změřený parametr	Hodnota
Průměrná doba registrace	16 ms
Průměrné zatížení CPU	7,22 %
Průměrný počet přerušení za sekundu	1616
Skutečná délka testu	10:07 min

3.9.7 Shrnutí a porovnání výsledků

Open IMS Core byl testován s vypnutým logováním CSCF serverů. Clearwater byl testován ve své výchozí konfiguraci.
Sadou testů registrační části bylo zjištěno, že registrace u Open IMS Core průměrně trvala pro 100 a 1000 registrací téměř stejnou dobu (657 ms a 623 ms) a pro 5000 registrací dosahovala 1,054 s. Do určité hranice počtu registrací má doba registrace stabilní délku asi 0,6 s a od hranice 1000 registrovaných uživatelů se doba registrace zvyšuje. Změřené vlastnosti procesoru ukazují, že probíhající registrace značně vytíží Open IMS Core.
Sadou testů registrační části bylo zjištěno, že registrace u Clearwateru průměrně trvala pro 100, 1000 a 5000 registrací téměř stejnou dobu (35 ms, 18 ms a 16 ms). Změřené vlastnosti procesoru ukazují, že probíhající registrace výrazně nevytíží Project Clearwater.
Průměrná registrace jednoho uživatele trvala u Clearwateru výrazně nižší dobu, než u Open IMS Core a také tolik nezatěžovala systém.

3.10 Realizace DoS útoku

Jako typ útoku byl zvolen INVITE flood, jehož princip spočívá ve velkém množství žádostí o navázání hovoru pomocí SIP zprávy INVITE poslaných na proxy server napadeného systému. Velké množství přijímaných falešných žádostí o spojení vede k zahlcení proxy serveru systému a následnému odepření služby (DoS). Pro realizaci DoS útoku byl zvolen nástroj inviteflood z webových stránek *hackingexposedvoip.com*. Pro vývoj nástroje inviteflood byly použity tyto open-source knihovny:

1. Libnet v1.1.2.1
2. Hack_library

Nástroj inviteflood byl napsán v jazyce C a je spustitelný z příkazového řádku operačního systému s linuxový jádrem.
Příklad syntaxe příkazu je zobrazen ve výpisu 3.5.

Výpis 3.5: Konfigurace IPSec protokolu v souboru pcscf.cfg v sekci konfigurace modulů

```
./inviteflood  eth0  Bob  openims  192.168.20.85  1000
```

Povinné atributy jsou:

- Interface – rozhraní z kterého jsou odesílány útoky (např. eth0).
- Target user – uživatelské jméno které je cílem útoku (např. Bob).
- Target domain – doménové jméno systému (např. openims).
- IPv4 addr of flood targer – IPv4 adresa IMS systému, na který je prováděn útok (např. 192.168.10.85).
- flood stage – počet vyslaných INVITE žádostí (např. 1000).

Volitelné atributy jsou:

- -a obsah položky záhlaví `From` se jménem uživatele (např. Alice).
- -i zdrojová IPv4 adresa, defaultní je IP adresa použitého rozhraní).
- -S zdrojový port (0 – 65535), defaultní je port 9.
- -D cílový port (0 – 65535), defaultní je port 5060.
- -l linka použitá pro SNOM, defaultně je prázdný.
- -s doba rozestupu mezi jednotlivými INVITE zprávami (v mikrosekundách), defaultně asi 20 mikrosekund.
- -h help – popis všech atributů.
- -v podrobný výstupní režim.

Útok probíhal na systémy u který byla generována stabilní zátěž pomocí hardwarového testeru Abacus 5000 a postupně byla zvyšována intenzita útoku. Sledovaným parametrem je čas odezvy (response time) v čase testu, na kterém je transparentně vidět reakce systému na příjem falešných zpráv generovaných nástrojem inviteflood. Časový scénář testu je zobrazen v tabulce 3.19. Test je rozdělen na 4 fáze, kdy v první

Tab. 3.19: Časový scénář testu

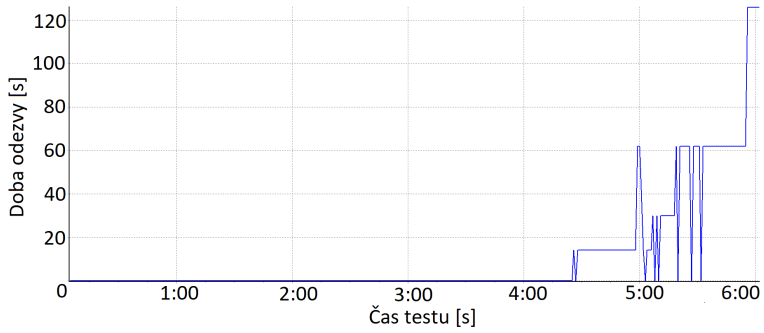
Parametr	Hodnota
Délka hovoru	2 s
Průběh	Obdélníkové rozdělení
Délka jednoho obdélníku	1:00 min
Délka pauzy	15 s
Počet fází	4
Počet hovorů za sekundu	5
Délka testu	5:00 min

fázi není prováděn žádný útok. Ve druhé fázi je prováděn útok s frekvencí 100 INVITE zpráv za sekundu. Ve třetí fázi je prováděn útok s frekvencí 1000 INVITE

zpráv za sekundu. V poslední, čtvrté fázi nebyl použit parametr -s, který určuje dobu rozestupu mezi INVITE zprávami, kde frekvence byla asi 50 000 INVITE žádostí za sekundu. Každá fáze trvá 1:15 minut, z čehož 1 minutu je generován provoz s vytížením 5 hovorů za sekundu a 15 sekund je pauza, po jejíž dobu není generován žádný hovor [19].

3.10.1 Open IMS Core

Z průběhu testu vyplývá, že systém se choval stabilně během útoků se silou 100 INVITE a 1000 INVITE zpráv odeslaných k P-CSCF serveru za sekundu. K odepření služby došlo až při třetím útoku, který odpovídal síle asi 50 000 INVITE žádostí odeslaných k P-CSCF serveru za sekundu. Průběh doby odezvy je zobrazen v grafu



Obr. 3.52: Průběh doby odezvy (response time) během testu

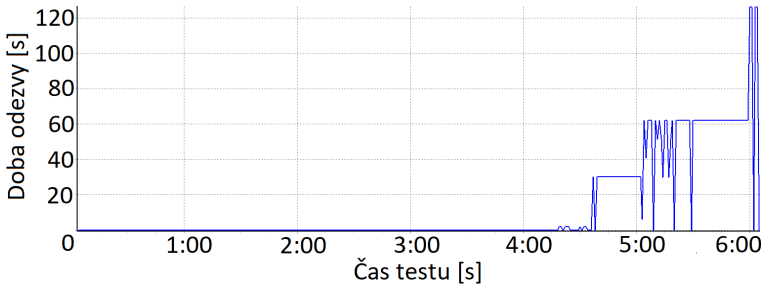
3.52, kde vidíme, že nejvyšší doba odezvy přesahovala 120 s a vzhledem k velkému nárůstu doby odezvy ve čtvrté fázi testu, můžeme považovat DoS útok za úspěšný.

3.10.2 Clearwater

Z průběhu testu vyplývá, že systém se choval stabilně během útoků se silou 100 INVITE a 1000 INVITE zpráv odeslaných k Bono serveru za sekundu. K odepření služby došlo až v třetím útoku, který odpovídal síle asi 50 000 INVITE žádostí odeslaných k Bono serveru za sekundu. Průběh doby odezvy je zobrazen v grafu 3.53, kde vidíme, že nejvyšší doba odezvy přesahovala 120 s a vzhledem k velkému nárůstu doby odezvy ve čtvrté fázi testu můžeme považovat DoS útok za úspěšný.

3.10.3 Shrnutí a možnosti obrany

Z provedených testů vyplývá, že testované systémy reagovali na zvyšující se intenzitu stejně. V obou porovnávaných systémech došlo k odepření služby (DoS) při záplavovém útoku inviteflood s intenzitou 50 000 odeslaných INVITE zpráv za sekundu.



Obr. 3.53: Průběh doby odezvy (response time) během testu

Menší intenzity útoku (100 a 1000 INVITE zpráv za sekundu) neměly výrazný vliv na stabilitu systému a poskytované služby u obou porovnávaných systémů.
Možnost ochrany proti DoS útokům nabízí Sesion Border Controller (SBC). SBC je síťová funkce pomáhající zabezpečit VoIP infrastruktury a současně zajišťuje konverzi mezi nekompatibilními signalizačními zprávami a mediálními toky z koncových zařízení, nebo aplikačních serverů.
Pro ochranu proti DoS útokům SBC využívá Access Control Lists (ACLs) a omezení datového toku. Také analyzuje každou správu, aby byla eliminována možnost zneužití nesprávně formátovaných paketů [20].

4 Porovnání Open IMS Core a Clearwater

Z pohledu studenta, který na jednotlivých řešeních bude dělat testy, bylo velké téma dostupnost a kvalita technické dokumentace. Open IMS Core, má technickou dokumentaci dostupnou z vlastních webových stránek, je však poměrně neucelená a zastaralá, mimo to oficiální webové stránky projektu často odkazují na již neexistující url. Naštěstí však existují mailing listy na webu sourceforge.net, kde lze najít řešení na některé problémy. U projektu Clearwater je situace odlišná, na oficiálním webu je odkaz na github se články týkajícími se systému. Technická dokumentace je kvalitně zpracovaná a využívá službu read the docs, která poskytuje prostor open source projektům pro technickou dokumentaci. Technická dokumentace je poměrně obsáhlá a popisuje průvodce instalací, realizaci zkušebního hovoru, další rozšířené funkce projektu, testování, atd. . .

Další věcí, která byla porovnávána je samotná architektura jednotlivých řešení. Open IMS Core striktně vychází z konceptu IMS, jak je popsán v teoretické části diplomové práce věnované IMS architektuře. K Open IMS Core architektuře je možné přidávat různé aplikační servery a další komponenty VoIP architektury jako rozšíření.

Architektura Projectu Clearwater se od původního konceptu IMS liší, což je dáno faktem, že IMS byl od základu navržen pro cloudové řešení. Project Clearwater ve své základní konfiguraci neobsahuje HSS, ale pouze Homestead Prov spolu s Homeastead, kteří si uživatelská data obstarávají z Vellum v databázi Cassandra. Volitelně je možné přidat externí HSS, například z Open IMS Core. Volitelně je také možné přidávat i další komponenty VoIP architektury, jako například aplikační servery, apod.

Obě porovnávané řešení podporují jak IPv4, tak i IPv6, ale neumožňují souběh protokolu IPv4 a IPv6 (dualstack).

Obě porovnávané řešení umožňují manuální instalaci na systému s linuxovým jádrem. Vývojáři systému poskytují i předinstalovaný virtuální obraz obou projektů, kterých bylo využito v této diplomové práci. Díky předinstalovanému virtuálnímu obrazu byla eliminována možnost nekompatibilního operačního systému a usnadnila se tím prováděná konfigurace.

Na základě prováděných testů bylo zjištěno, že Open IMS Core je vhodný spíše pro malé systémy, kdy při nízkém zatížení dosahoval lepších výsledků v rychlosti navazování hovorů, než Clearwater. Naopak pro větší systémy je Open IMS Core prakticky nepoužitelný, neboť při vetším zatížení docházelo k pádu S-CSCF serveru při zátěži 330 probíhajících hovorů. Do hranice 320 probíhajících hovorů byl Open IMS Core stabilní. Clearwater byl stabilní asi do zátěže 1000 probíhajících hovorů současně a maximálně zvládl obsloužit 1380 hovorů, kde však již docházelo k zpožďování při

Tab. 4.1: Shrnutí výsledků testů při navazování hovorů

Open IMS Core	100 hov.	1000 hov. (do pádu S-CSCF)	2000 hov.	5 hov/s
Průměrná doba odezvy SIP zpráv	8 ms	7 ms	–	6 ms
Průměrná doba sestavení hovoru	20 ms	27 ms	–	16 ms
Průměrné zatížení CPU	5,47 %	12,37 %	–	7,35 %
Project Clearwater	100 hov.	1000 hov.	2000 hov.	5 hov/s
Průměrná doba odezvy SIP zpráv	16 ms	28 ms	424 ms	15 ms
Průměrná doba sestavení hovoru	117 ms	297 ms	953 ms	95 ms
Průměrné zatížení CPU	6,8 %	34,43 %	54,83 %	7,1 %

Tab. 4.2: Shrnutí výsledků testů při registraci uživatelů

Open IMS Core	100 registrací	1000 registrací	5000 registrací
Průměrná doba registrace	657 ms	623 ms	1054 ms
Project Clearwater	100 registrací	1000 registrací	5000 registrací
Průměrná doba registrace	35 ms	18 ms	16 ms

zahajování hovorů v řádu několika sekund. Na základě změřených parametrů bylo zjištěno, že Open IMS Core zvládá navazovat hovory řádově rychleji, než Project Clearwater. Project Clearwter se naopak ukázal jako stabilnější a zvládá vyšší zatížení, což je pro reálné nasazení klíčové. Shrnutí výsledků testů s klíčovými parametry při navazování hovorů jsou zobrazeny v tabulce 4.1.

Sadou testů registrace bylo zjištěno, že doba registrace u Open IMS Core průměrně trvá delší dobu, než u Project Clearwater. Značný rozdíl v době registrace uživatele k síti je pravděpodobně způsoben rozdílným řešením úložiště uživatelských profilů. Přehledné shrnutí průměrné doby registrace uživatelů, které byly zjištěny v testu jsou zobrazeny v tabulce 4.2.

Odolnost proti DoS útoku inviteflood nemá ve své základní konfiguraci implementovanou ani jedna z obou testovaných systémů. U obou systémů došlo k odmítnutí

Tab. 4.3: Porovnávací tabulka vybraných open source IMS řešení

Parametry	Open IMS Core	Project Clearwater
Vývoj	Fraunhofer FOKUS Institute	Metaswitch Network
Škálovatelnost	ne	ano
IP protokoly	IPv6, IPv4	IPv6, IPv4
Architektura	Čisté IMS	Cloudové řešení IMS
Rok vydání	2006	2013
Vhodnost nasazení	Malé sítě, testovací sítě	Větší sítě
Limitní počet hovorů	320	1380

služby při intenzitě útoku 50 000 IVNITE zpráv za sekundu.
Více možností implementovaného zabezpečení nabízí Open IMS Core, který podporuje komunikaci zabezpečenou pomocí protokolů IPSec a TLS. Clearwater podporuje pouze zabezpečení protokolu DIAMETER pomocí TLS při komunikaci s procesy Ralf a Homestead. Open IMS Core také podporuje funkci skrytí topologie, která umožňuje zašifrovat důležité informace o topologii sítě. Clearwater komunikaci rozděluje na důvěryhodnou a nedůvěryhodnou zónu, které jsou navzájem odděleny pomocí bezpečnostních skupin a nebo pravidly firewallu. Clearwater také umožňuje počítání nonce, což může mít při správné implementaci pozitivní bezpečnostní dopad a také pozitivní vliv na výkon celého systému. Oba systémy využívají standardní autentizační mechanismy (SIP digest) [15], [11].
Porovnávané parametry shrnuje tabulka 4.3.

Závěr

Na počátku bylo nutné si nastudovat technologii IMS, zejména jaké komponenty obsahuje a jakým způsobem mezi sebou dané komponenty komunikují. V důsledku toho vznikla celá první kapitola, kde je popsána IMS architektura. IMS architektura byla pro větší přehlednost rozdělena do logických vrstev. Následně byly popsány jednotlivé komponenty IMS architektury. Následuje krátká sekce o identifikaci v IMS sítích. První kapitola obsahuje rovněž krátké seznámení s protokoly využívanými IMS, většina z nich je notoricky známá, takže nebylo nutné při jejich popisu zacházet příliš do hloubky. V další sekci je detailně popsán proces registrace a sestavení spojení. V poslední sekci této kapitoly jsou popsány různé možnosti bezpečnostních hrozeb pro IMS.

Po shromáždění teoretických informací bylo nutné vybrat alespoň dvě open source implementace technologie IMS, které budou následně popsána a porovnána. Prvním vybraná implementace byla Open IMS Core, který zachovává původní architekturu IMS jak je popsána v teoretické části. Open IMS Core je hojně využíván pro testování IMS, například v akademické sféře. Druhou vybranou implementací je Project Clearwater řídící se principy IMS, které jsou však implementovány odlišně, než u původního modelu IMS architektury. Project Clearwater byl od základu navržen jako cloudové řešení IMS a je více zaměřen na výkon celého systému. Vybrané systémy byly detailně prostudovány a popsány. Popis vybraných implementací se zaměřuje zejména na konkrétní architekturu zvolených projektů, jejich instalaci s vyzkoušením základních funkcí a možnosti implementovaného zabezpečení proti různým bezpečnostním hrozbám.

Ve své praktické části se diplomová práce soustředí na výkonnostní testování zvolených projektů pomocí hardwarového testeru Abacus 5000. Pro realizaci výkonnostních testů bylo nutné se nejdříve podrobně seznámit s hardwarovým testerem Abacus 5000 a provést inicializační testování na 2 uživatelích. Tímto testováním byla zjištěna možnost kompatibility testeru a testovaných projektů. Pomocí inicializačního testování byl odladěn formát SIP zpráv a nastavení testeru, tak aby byli kompatibilní s testovanými systémy. Následně bylo možné realizovat výkonnostní testování zvolených implementací.

Byly realizovány výkonnostní testy pro zvolené implementace s různou definovanou zátěží. Sledována byla rychlost sestavení hovoru, doba odezvy SIP zpráv a vliv aktuální zátěže na vytížení systému. U Open IMS Core, bylo nutné vypnout logování CSCF serverů, z důvodu zvýšení výkonosti systému. Srovnání výkonosti Open IMS Core se zapnutým a vypnutým logováním CSCF serverů je k dispozici hned v prvním testu, kde byla zátěž schodově zvyšována do výše 100 hovorů. Open IMS Core se zapnutým logováním CSCF serverů byl pro zvolené zatížení prakticky nepoužitelný

a docházelo tam k několikasekundovým zpožděním při navazování hovorů a později také k zahazování hovorů. Open IMS Core s vypnutým logováním CSCF serverů dosahoval lepších výsledků než Project Clearwater a oba IMS projekty byly pro dané zatížení stabilní. Další testy byl proto realizován pouze s vypnutým logováním CSCF serverů.

Následujícím testovacím scénářem bylo schodovité navyšování zátěže do výše 1000 hovorů. U Open IMS Core nemohl být test dokončen z důvodu pádu S-CSCF serveru. Příčina pádu nebyla blíže analyzována ani přesně určena. Pád S-CSCF serveru nastal při pokusu o navázání hovoru 330 uživatelů, do té doby se systém choval stabilně. U Projectu Clearwater žádná chyba nenastala, takže byl test dokončen. V závěru testu u Projectu Clearwter docházelo k výraznějšímu zpoždění při sestavení hovorů a vysokému vytížení CPU virtuálního stroje, což poukazuje na blízkost maximální hodnoty obsluhovaných hovorů u Projectu Clearwater.

Dalším testovaným scénářem bylo schodovité navyšování zátěže do 2000 hovorů, které bylo realizováno pouze pro Project Clearwater z důvodu předchozího výsledku testování Open IMS Core. Tímto testováním byl zjištěn maximální počet hovorů, které dokáže Project Clearwater obsloužit bez zahazování hovorů, který byl 1380 hovorů. Realizovaný test ukázal, že Project Clearwater je stabilní asi do 1000 hovorů probíhajících současně.

Následující testovaný scénář simuluje reálnou zátěž systému poissonovým rozložením hovorů po dobu testu. Počet hovorů byl stanoven na průměrnou hodnotu 5 pokosů o navázání hovoru za sekundu. Průměrný počet hovorů za sekundu byl zvolen takový, aby jej oba systémy bez problémů zvládli a bylo možné změřit důležité parametry při sestavení hovorů. Lepších výsledků v rychlosti sestavení hovoru a době odezvy SIP zpráv dosáhl Open IMS Core.

V dalším testování byly sledovány parametry registrace uživatelů k IMS sítím. Byl testován různý počet registrací probíhajících současně. Testováno bylo 100, 1000 a 5000 pokusů o registraci k IMS systému. Sledována byla průměrná rychlost registrace uživatele a vliv registrace na vytížení systému. U Open IMS Core trvala registrace řádově déle než u Open IMS Core a také měla větší vliv na vytížení systému, registrace trvala více než 0,6 sekundy. U Projectu Clearwater naopak byla doba registrace nízká a tolik nevytěžovala systém, doba registrace se pohybovala v řádech desítek milisekund.

Posledním testováním s hardwarovým testerem Abacus 5000, byla realizace DoS útoku s pomocí nástroje inviteflood, který je schopen realizovat záplavový útok pomocí SIP žádostí o spojení INVITE. Byla sledována reakce systémů, kde byla pomocí hardwarového testeru Abacus 5000 generována zátěž 5 hovorů za sekundu s délkou hovoru 2 sekundy. Intenzita útoku se posupně zvyšovala, na počátku byla 100 INVITE zpráv za sekundu, v další fázi 1000 INVITE zpráv za sekundu a v poslední

fázi 50 000 INVITE zpráv za sekundu. K odepření služby došlo u obou testovaných systémů až při třetí fázi útoku. Po vyhodnocení realizovaného útoku následuje část věnovaná možné ochraně proti DoS útokům.
Poslední kapitola se zabývá porovnáním zvolených open source IMS implementací vycházejících z provedených testů a z veřejně dostupných informací.
Jedním z cílů diplomové práce bylo zpracovat laboratorní úlohu, která seznámí studenty s funkcí IMS sítě. Pro konkrétní realizaci byl zvolen projekt Open IMS Core, který zachovává standardní architekturu IMS. Studenti v navržené laboratorní úloze vyzkouší vytvoření uživatelského profilu v rámci HSS, registraci uživatele do sítě a navázání hovoru s ostatními uživateli. Signalizační procesy, které stojí za těmito úkony prozkoumá zachycením komunikace. Úloha se rovněž zaměřuje na vnitřní směrování v IMS síti.

Literatura

[1] E. Gałczyńska, W. Zabierowski, and A. Napieralski. Ip multimedia subsystem and its protocols: A step to convergence. In *2008 International Conference on "Modern Problems of Radio Engineering, Telecommunications and Computer Science"(TCSET)*, pages 485–488, Feb 2008.

[2] M. Koukal and R. Bestak. Architecture of ip multimedia subsystem. In *Proceedings ELMAR 2006*, pages 323–326, June 2006. doi:10.1109/ELMAR.2006.329576.

[3] Simon ZNATY and Jean-Louis DAUPHIN. Ip multimedia subsystem : Principles and architecture. 18.6.2005. URL: http://www.efort.com/media_pdf/IMS_ENG.pdf.

[4] Gonzalo Camarillo and Miguel A. García-Martín. *The 3G IP multimedia subsystem (IMS)*. Wiley, Chichester, 2.edice edition, 2006.

[5] What is sip?, 2004. URL: https://www.networkworld.com/article/2332980/lan-wan-what-is-sip.html.

[6] Voip thing. URL: http://www.en.voipforo.com/SIP/SIP_architecture.php.

[7] *SDP: Session Description Protocol*. University of Glasgow, rfc4566 edition, Červenec 2006.

[8] Real-time transport protocol (rtp). URL: https://www.techopedia.com/definition/4755/real-time-transport-protocol-rtp.

[9] *Diameter Base Protocol*. Nokia Research Center, rfc6733 edition, 2012.

[10] H. Meng. A preliminary research on security issues in ip multimedia subsystem. In *2012 9th International Conference on Fuzzy Systems and Knowledge Discovery*, pages 2560–2564, May 2012. doi:10.1109/FSKD.2012.6234000.

[11] The open source ims core project, 2008. URL: http://openimscore.sourceforge.net/.

[12] Sourceforge. URL: https://sourceforge.net/p/openimscore/mailman/.

[13] Project clearwater. URL: https://www.projectclearwater.org/.

[14] Nonce count support, 2019. URL: https://github.com/Metaswitch/clearwater-website-archive/blob/master/blog_posts/Nonce_Count_Support.md.

[15] Project clearwater 1.0 documentation, 2016. URL: https://clearwater.readthedocs.io/en/stable/index.html#.

[16] Abacus 5000—ip telephony migration test system, 2010.

[17] Software manual, 2009.

[18] Saliha Mallem. How to add multipleusers in hss of open ims core?, 2015. URL: https://www.researchgate.net/post/How_to_add_MultipleUsers_In_HSS_of_OPEN_IMS_Core.

[19] Hacking exposed voip, 2006. URL: http://www.hackingexposedvoip.com/index.php.

[20] Session border control in ims and volte / v2olte. URL: https://www.metaswitch.com/knowledge-center/reference/session-border-control-in-ims-and-volte-/-v2olte.

[21] L. N. Saleem and S. Mohan. An analysis of ip multimedia subsystems (ims). In *2007 First International Symposium on Advanced Networks and Telecommunication Systems*, pages 1–2, Dec 2007. doi:10.1109/ANTS.2007.4620219.

Seznam symbolů, veličin a zkratek

3GPP	Partnerský Projekt 3. Generace – the 3rd Generation Partnership Project
AAA	authentication, authorization and accounting
ACLs	Access Control Lists
AMI	Amazon Machine Image
API	Application Programming Interface
AS	Aplikační servery – Aplication Servers
AVP	Attribute-Value Pair
BSF	Bootstrapping Server Function
CDF	Charging Data Function
cdp	CDiameterPeer modul
CG	Circuit Generator
CPU	Centrální procesorová jednotka – Central Processing Unit
CSCF	call session control function
DAL	Data Acces Layer
DBMS	Database Management System
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	Odmítnutí služeb – Denial of Service
ETSI	European Telecommunications Standards Institute
FIFO	First In, First Out
FQDN	Fully Qualified Domain Name
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
GUI	Grafické uživatelské rozhraní – Graphical user interface
HSS	home subscriber server
HTTP	hypertext transfer protocol
I-CSCF	Interrogating-CSCF
IBCF	Interconnect Border Control Function
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IoT	internet věcí – Internet of Things
IPSec	Internet Protocol Security
IPv4	IP protokol verze 4
IPv6	IP protokol verze 6
ISC	IP Multimedia Service Control
ISIM	IMS Subscriber Identity Module

LIA	Location-Information-Answer
LIR	Location-Information-Request
MAA	Multimedia-Auth-Answer
MAR	Multimedia-Auth-Request
MG	Media Gateway
MGFC	Media Gateway Control Function
MMTEL	Multimedia telephony
MRFC	multimedia resource function controller
MRFP	multimedia resource function procesor
NAI	Network Access Identifier
NAPT	Network Addressand Port Translation
NAT	Network Address Translation
NGN	Next Generation Network
P-CSCF	Proxy-CSCF
PDF	policy decision function
PSTN	Veřejná telefonní síť – public switched telephone network
QoS	Kvalita služeb – Quality of Service
RTCP	Real-time Transport Control Protocol
RTP	Real-time Transport Protocol
S-CSCF	Serving-CSCF
SAA	Server Assignment Answer
SAP	Service Advertising Protocol
SAR	Server Assignment Request
SBC	Session Border Controller
SDP	Session Description Protokol
SGW	Signal Gateway
SIM	Subscriber Identity Module
SIP	protokol pro inicializaci relací – Session Initiation Protocol
SLF	Subscriber location function
SQL	Structured Query Language
SSH	Secure Shell
SUT	System Under Test
TAS	Telephony Application Server
TCP	Transmission Control Protocol
TDM	časový multiplex – Time-division multiplexing
THIG	Topology Hiding Internetwork Gateway
TLS	Transport Layer Security
TISPAN	Telecoms Internet converged Services Protocols for Advanced Networks

UAR	User Authentication Request
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunication System
USIM	Universal Subscriber Identity Module
URI	Uniform Resource Identifier
VoIP	Voice over Internet Protocol
VoLTE	Voice over LTE
VXML	Voice XML
WebRTC	Web Real-Time Communication
XCAP	XML Configuration Access Protocol
XDMS	XML Data Management Server
XML	rozšiřitelný značkovací jazyk – eXtensible Markup Language

Seznam příloh

A Laboratorní úloha

5 IP Multimedia Subsystem – Open IMS Core

5.1 Úvod

Cílem IMS je přenést výhody internetu do 3G celulárních systémů a nadále rozvíjet celulární sítě. Mezi služby podporované IMS patří například konferenční hovory, videokonference, chat, instant messaging.

IMS poskytuje platformu pro komunikaci mezi všemi druhy terminálů od starých telefonů přes mobilní telefony po osobní počítače a chytrá zařízení.

V případě sítí IP, IMS podporuje hovory pomocí protokolu SIP z koncových zařízení přes veřejné nebo soukromé sítě.

IMS umožňuje připojit zařízení nezávisle na přístupové technologii, jediným požadavkem je protokol IP a SIP, avšak lze se připojit i ze systémů nepodporujících zmiňované protokoly prostřednictvím bran.

5.2 Komponenty IMS

Call Session Control Function Ústřední systémovou součástí síťové infrastruktury IMS je funkce Call Session Control Function (CSCF). Jedním z hlavních účelů CSCF je signální směrovací funkce.

Po příchodu nového SIP volání, CSCF nejdříve autentizuje uživatele pomocí Home Subscriber Serveru (HSS). Přestože CSCF předává zprávy SIP rozhraní ISC aplikacím, někdy je potřeba zůstat aktivní v řízení relace. Toho je dosaženo přidáním informační hlavičky ve standardní SIP zprávě (REGISTER, INVITE, atd.).

Pokud aplikace sama o sobě zvládne vyřídit požadavek na relaci, tak informuje CSCF, které již nepoužívá žádné další SIP směrování. Naopak, když je potřeba zahrnout více služeb do relace, tak aplikace první služby vrátí zprávu SIP CSCF, která znovu prozkoumá ISC filtry a podívá se jaká je další služba v řetězci a finálně předá zprávu dál.

Při hlubším pohledu na IMS architekturu zjistíme, že CSCF se dělí do tří ovládacích funkcí:

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

P-CSCF, je z pohledu uživatele prvním přístupovým bodem do IMS. Mezi jeho hlavní funkce patří zajištění, že registrace uživatele je předána do správné domácí sítě a že SIP zprávy jsou předány správnému S-CSCF, jakmile dojde k registraci. Kontakt s domovskou sítí během registrace probíhá prostřednictvím I-CSCF v domácí síti a počáteční nastavení relace SIP probíhá přes tzv. party I-CSCF. P-CSCF se také stará o alokování zdrojů pro mediální toky, generuje data pro vyúčtování a poskytuje ochranu proti signalizačním útokům SIP.

I-CSCF je důležitý pro relace z peer-to-peer sítí. Určuje S-CSCF, u kterého by se měl uživatel zaregistrovat. Toho je dosaženo dotazem na Home Subscriber Server (HSS), který zkontroluje, zda se uživatel může zaregistrovat v původní síti a vrátí jméno a přidělení paměti. Jakmile bude určeno, který S-CSCF je používám, tak je možno odstranit I-CSCF z cesty.

I-CSCF může mít funkci Topology Hiding Internetwork Gateway (THIG), která umožňuje skrýt topologie operátora z peer to peer sítí. Pokud je THIG aktivní, I-CSCF zůstává zapojena v signalizaci hovoru.

S-CSCF je centrálním bodem CSCF. Operace S-CSCF jsou řízeny dle politiky uložené v HSS. Tato logika je zodpovědná za autentizaci, registraci a autorizaci uživatelů, pro zpracování komunikace včetně získání informací o spouštění služby a uživatelský profilů z HSS a pro směrování komunikace do aplikací. S-CSCF udržuje čítače relací a poskytuje fakturační informace systémům zprostředkování faktur.

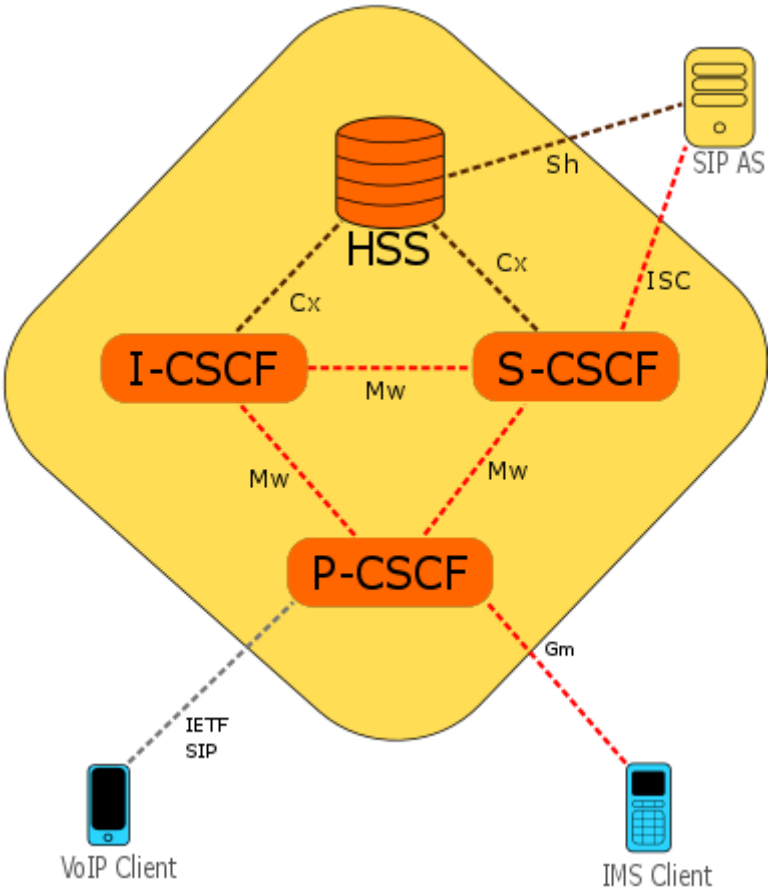
Home Subscriber Server (HSS)

IMS specifikace obsahují definici databáze účastnických profilů, centrální úložiště informací o účastnících, kterému se říka Home Subscriber Server (HSS). Udržuje všechny informace o účastnících , které jsou nezbytné pro navázání relací mezi uživateli a pro poskytování služeb. Rozhraní HSS je označeno Sh a obsahuje standarty pro protokol komunikační protokol s HSS, data uložená v HSS a mechanismus na bázy XML pro ukládání dat o účastnických službách. Sh rozhraní vychází z existujícího standardu DIAMETER používaného pro přístup k informacím o účastnících. Integrace s existujícími databázemi - IMS zohledňuje investice předchozích operátorů do jejich síťových infrastruktur a v tomto rozsahu zmírňuje budoucí nasazení technologií. Na HSS může být nahlíženo jako na server, který poskytuje pevné rozhraní pro více zdrojů informací.

Open IMS Core

Projekt byl spuštěn v roce 2006 na podporu IMS. Od té doby pak projekt Open IMS Core sloužil jako referenční implementace a zkušebna pro testování a prototypování. Projekt je určen pro výzkum a vývoj, využívá jej mnoho poskytovatelů telekomunikačních zařízení, provozovatelů sítí a univerzitních projektů. Otevřená zkušební stanoviště jsou důležitým prostředkem pro poskytování povolující infrastruktury ke zrychlení inovací prostřednictvím výzkumných a vývojových činností.

Open IMS Core je open source implemetace funkce CSCF a HSS, které spolu dohromady tvoří základní prvky všech architektur IMS/NGN, dle specifikace v 3GPP, 3GPP2, ETSI TISPAN a iniciativě PacketCable. Všechny čtyři komponenty (ICSCF, S-CSCF, P-CSCF a HSS) jsou založeny na open source softwaru.



Obr. 1 Architektura Open IMS Core

5.3 Zadání úlohy – Spuštění a testování Open IMS Core

🛠 Úkol 1: Vytvoření uživatelských účtů ve webovém rozhraní HSS

Do internetového prohlížeče zadáme IP adresu 192.168.10.182 s portem 8080, čili výsledná url bude mít následující tvar: *http://192.168.10.182:8080* , přístupové údaje pro administrátorskou sekci jsou:

- Uživatelské jméno*: hssAdmin*
- Heslo: *hss*

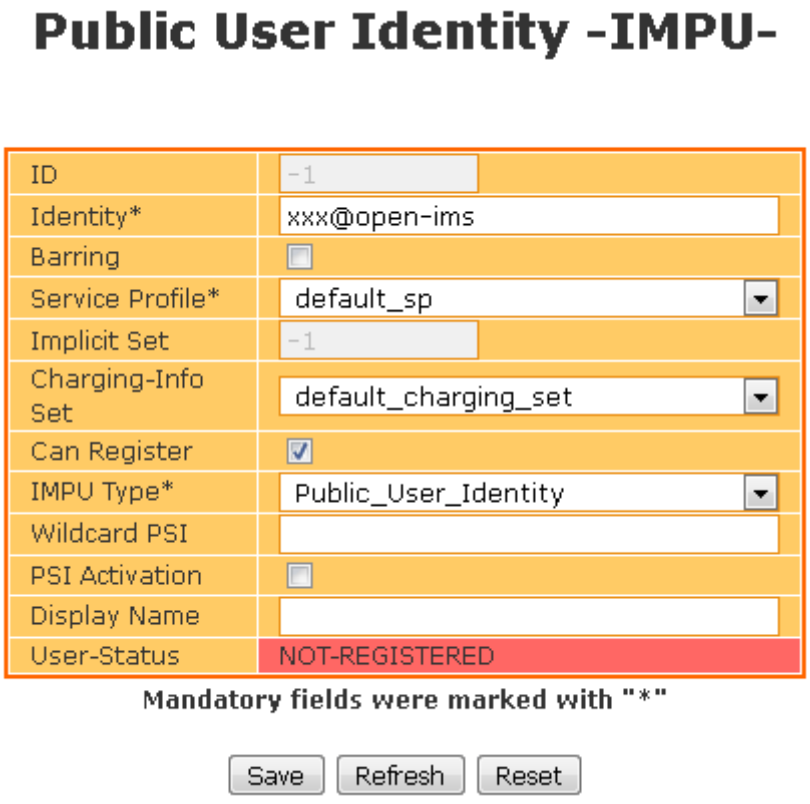
Po úspěšném přihlášení v horním menu přejdeme do sekce *User Identities*, kde jsou na výběr podsekce:

- IMS Subsciprion
- Private Identity
- Public User Identity

Přesuneme se do podsekce *Public User Identity*, kde pomocí tlačítka *Create* přejdeme k formuláři pro vytvoření nové IMPU. Do pole Identity vložíme sipovou url ve tvaru *sip:xxx@open-ims* za xxx dosadíme jméno účtu z tabulky 1, dle čísla pracoviště. Profil služeb *Service profile* zvolíme *default_sp,* informace o účtování *Charging-Info Set* nastavíme jako *default_charging_set, IMPU Type* nastavíme na *Public_User_Identity* dále již nic nenastavuje a uložte vytvořenou IMPU. Poté musíme přidat položku open-ims do navštívených sítí, vybráním položky v poli Add Visited-Networks.

Tabulka 1: Rozdělení pracovišť

Pracoviště	IP adresa PC	Jméno účtu (za X dosadíme libovolnou číslici)
1	192.168.10.163	60X
2	192.168.10.164	61X
3	192.168.10.165	62X
4	192.168.10.167	63X
5	192.168.10.168	64X
6	192.168.10.169	65X
7	192.168.10.170	66X
8	192.168.10.171	67X
9	192.168.10.172	68X
10	192.168.10.173	69X



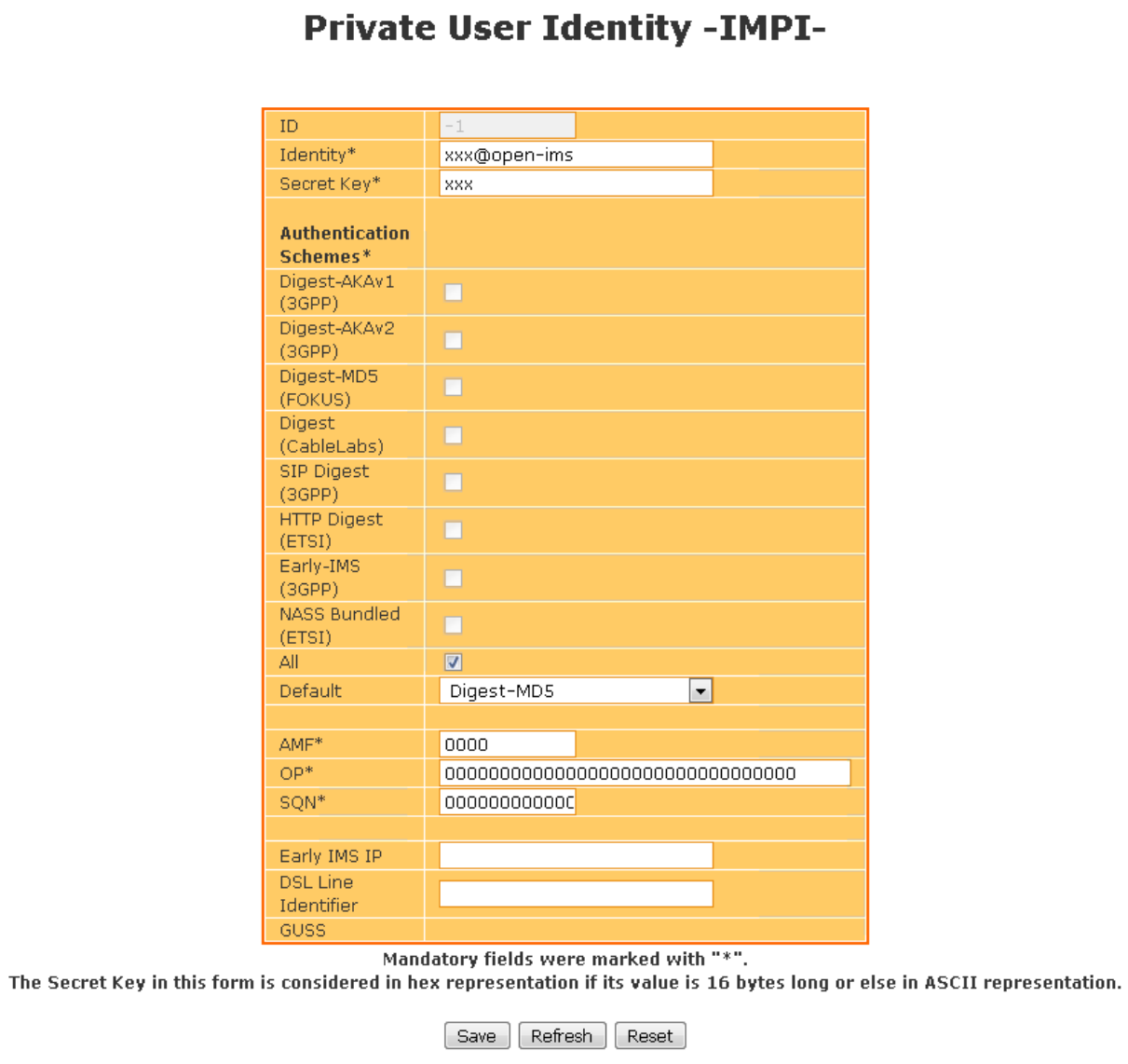
Obr.2 Založení nové IMPU

Následně se přesuneme do podsekce *Private Identity – Create*, kde vytvoříme IMPI, do položky *Identity* vyplníme ve tvaru: *xxx@open-ims* kde za xxx dosadíme jméno účtu z tabulky 1, dle čísla pracoviště. Dále nastavíme heslo pomocí položky *Secret Key*, protože je to laboratorní úloha tak nejvhodnější bude, aby bylo krátké a dobře se pamatovalo, autentizační schéma v části *Authentication Schemes* zaklikneme *All* a v položce výchozí

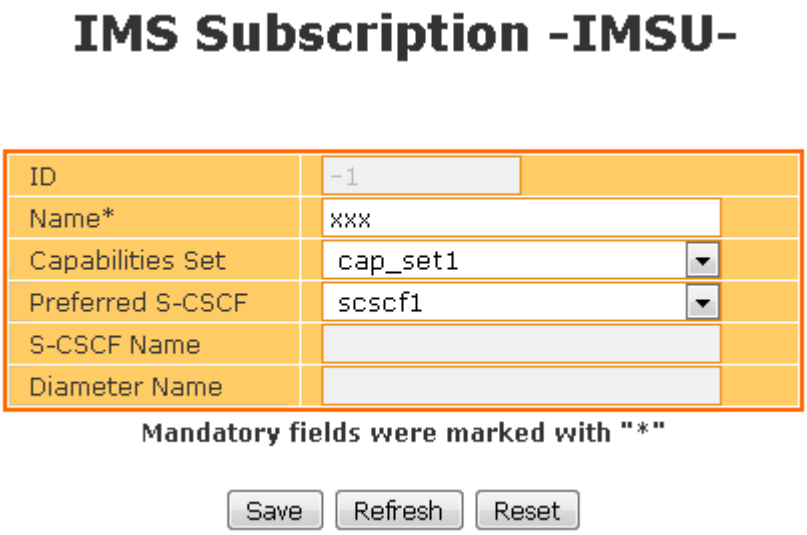
Default zaklikneme *Digest-MD5*. Zbylé položky necháme tak jak jsou a uložíme. Následně je nutné asociovat IMPI a IMPU, toho docílíme tak, že v položce *Associate IMPU(s)* zadáme vytvořené IMPU a klikneme na *Add*.

Následně vytvoříme *IMS Subscription*. V dané podsekci klikneme na *Create*. Jméno *Name* zvolíme stejné jako jméno uživatele, tj. *xxx*, kde za xxx dosadíme jméno účtu z tabulky 1, dle čísla pracoviště. *Capabilities set* zvolíme *cap_set1*, *Prefered S-CSCF* zvolíme *scscf1*, a uložte. Nakonec položce *Associate IMPI(s)* zadáme námi vytvořené *IMPI* a potvrdíme.

Stejný postup opakujeme pro dalšího uživatele se jménem xxx, kde za xxx dosadíme jméno účtu z tabulky 1, dle čísla pracoviště.



Obr.3 Založení nové IMPI



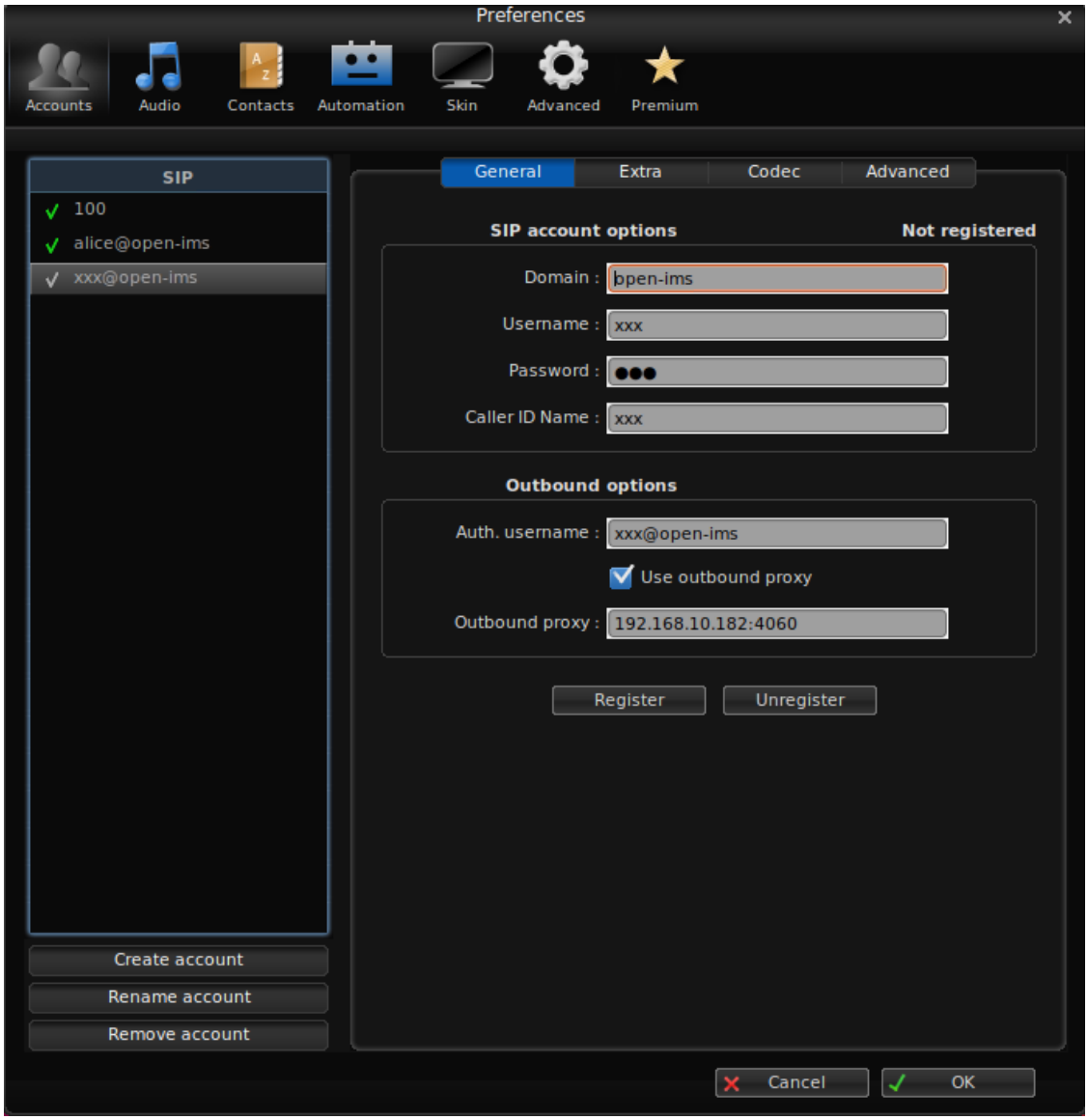
Obr.4 Založení nové IMSU

🛠 Úkol 3: Konfigurace klientů

Komunikaci budeme testovat na softwarovém klientovi Zoiper, kde si vytvoříme nový účet, v menu *Settings-Create a new account,* zvolíme typ účtu – *SIP,* poté vyplníme údaje *user / user@host,* kde zadáme IMPI ve tvaru *xxx@open-ims* . Do kolonky *Password* zadáme heslo dle vytvořeného účtu. Do poslední položky *Domain / Outbound proxy* zadáme IP adresu počítače a port 4060 ve tvaru *192.168.10.182:4060,* jako adresu P-CSCF. V dalším kroku odškrtneme *Skip auto-detection* a následující okno zavřeme. V aktuálně otevřeném okně s uživatelskými účty nastavíme položku v podsekci *Outbound options Auth. username* jako naši IMPI, čili např. *xxx@open-ims* a do položky *Caller ID Name* vložíme jméno účtu jako *xxx*.

V této fázi si již můžeme klienta zaregistrovat tlačítkem *Register*, pokud byla registrace úspěšná tak se vlevo u vytvořeného účtu objeví zelená fajfka.

Totéž opakujeme i pro druhý účet, na jiné verzi Zoiperu.



Obr.4 Konfigurace účtu v programu Zoiper

🛠 Úkol 4: Odchytněte registraci klienta k IMS a analyzujte jednotlivé zprávy pomocí programu Wireshark.

Odregistrujte si jeden účet pomocí menu tlačítka *Unregister*.
Otevřete si terminál a pomocí ssh se připojte na k virtuálnímu stroji s Open IMS Core příkazem:

```
ssh 192.168.10.182
```

Tímto příkazem se připojíte ke vzdálenému pc, jako heslo použijte *student*.
Následně zapněte zachytávání paketů pomocí programu tcpdump příkazem:

```
sudo tcpdump -i any portrange 4060-6060 -s 65535 -vvv -w ~/<VUT login>Registrace
```

Tímto příkazem zapnete zachytávání paketů ve virtuálním počítači s IMS. Následně se opět registrujte a zastavte zachytávání komunikace pomocí stisku kláves CTRL+C. Ukončete spojení ssh a pomocí příkazu *exit*.
Poté zkopíruje vytvořený soubor pomocí programu scp:

```
scp -r 192.168.10.182:~/ <VUT login>Registrace ~/Desktop/<VUT login>Registrace
```

Přenesený soubor obsahující zachycenou komunikaci si otevřete v programu wireshark a zanalyzujte. Pro větší přehlednost je doporučené si vyfiltrovat provoz pouze z vašeho pracoviště.

Zaměřte se na zprávy *401 unauthorized* na pole *WWW-Authenticate,* v které scscf žádá o identifikaci uživatele, který se pokouší registrovat. Na tuto zprávu je odpovězeno SIP požadavkem REGISTER, který obsahuje pole *authorization,* v kterém IMS client odpovídá na výzvu o autentifikaci.

Zapište si, přes které komponenty probíhalo vnitřní směrování registrace, v tom Vám může pomoci tabulka 2, v které jsou vyznačeny jednotlivé komponenty, jejich IP adresy a porty. Rovněž se zaměřte na to, jak souvisí pole záhlaví *Record-Route* a *Via* s probíhající signalizací.

Vyhledejte kód *nonce* v poli *WWW-Authenticate* ve zprávě *401 unauthorized* a porovnejte jej s *nonce* kódem v odpovědi SIP požadavkem REGISTER rovněž v poli *WWW-Authenticate*. Nakonec S-CSCF odpoví zprávou *200 OK*, že uživatel byl úspěšně registrován.

🛠 Úkol 5: Realizujte hovor mezi dvěma uživateli, SIP zprávy analyzujte pomocí programu Wireshark.

Stejným způsobem jako ve 4. úkolu zachytíme komunikaci SIP klientů, tentokrát pro realizaci hovoru mezi dvěma uživateli.

Utvořte dvojice a zavolejte si mezi sebou, stačí zadat jen jméno účtu na který chcete zavolat v programu Zoiper.

V programu Wireshark pomocí *statistics – VoIP Calls*, vybereme realizovaný hovor, tam je možno zobrazit hovor v přehledné grafice, pomocí tlačítka *Graph*. Výsledný graf by měl být podobný jako na obr.6.

Prozkoumejte zprávu INVITE, která obsahuje zprávu protokolu SDP, která slouží jako nabídka parametrů relace.

Další důležitou zprávou je odpověď 200 OK, která obsahuje SDP odpověď a dohodnutí parametrů relace.

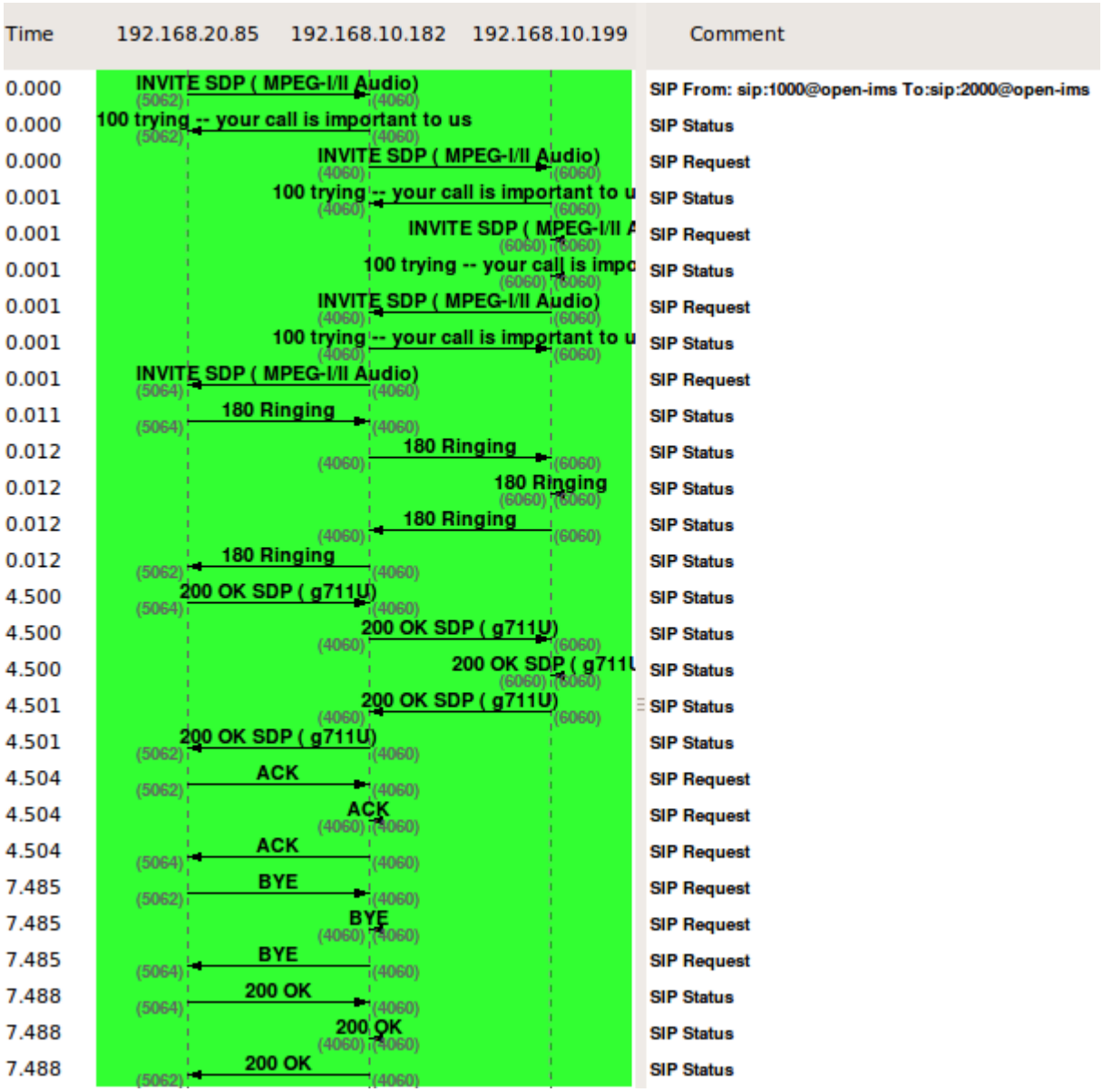
Volající potvrdí zprávu 200 OK, pomocí ACK a tím započne hovor. Ukončení jednou stranou je realizováno zprávou BYE, na kterou druhá strana odpoví zprávou 200 OK a hovor se ukončí.

Zapište si, přes které komponenty probíhalo vnitřní směrování hovoru, v tom Vám může pomoci tabulka 2, v které jsou vyznačeny jednotlivé komponenty, jejich IP adresy a porty. Rovněž se zaměřte na to, jak souvisí pole záhlaví *Record-Route* a *Via* s probíhající signalizací.

Zapsané výsledky ukažte vyučujícímu.

Tabulka 2: Porty a IP adresy, na kterých komunikují jednotlivé prvky

Port	IP adresa	Prvek
4060	192.168.20.182	P-CSCF
5060	192.168.20.189	I-CSCF
6060	192.168.20.199	S-CSCF



Obr.6 Zachycená signalizační část hovoru programem wireshark

🛠 Úkol 6: Uvedení pracoviště do původního stavu

Připojte se pomocí webového rozhraní k HSS a smažte vytvořené profily v částech *IMS Subscription*, *Private Identity* a *Public User Identity*.

Seznam použité literatury

[1] L. N. Saleem and S. Mohan. An analysis of ip multimedia subsystems (ims). In *2007 First International Symposium on Advanced Networks and Telecommunication Systems*, pages 1–2, Dec 2007. doi:10.1109/ANTS.2007.4620219.

[2] E. Gałczyńska, W. Zabierowski, and A. Napieralski. Ip multimedia subsystem and its protocols: A step to convergence. In *2008 International Conference on "Modern Problems of Radio Engineering, Telecommunications and Computer Science"(TCSET)*, pages 485–488, Feb 2008.

[3] M. Koukal and R. Bestak. Architecture of ip multimedia subsystem. In *Proceedings ELMAR 2006*, pages 323–326, June 2006. doi:10.1109/ELMAR.2006.329576.

[4] The open source ims core project, 2008. URL: http://openimscore.sourceforge.net/.

B Konfigurační soubory OpenIMS Core

Výpis B.1: Výpis nastavení dns, ze souboru named.conf

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/rndc.key";
include "/etc/bind/zones.rfc1918";
controls{
inet 127.0.0.1 port 953
allow{127.0.0.1;}keys{"rndc-key";};
};
zone "openims" {
type master;
file "/etc/bind/open-ims.dnszone";
};
zone "20.168.192.in-addr.arpa" IN{
type slave;
file "/etc/bind/open-ims.rev.dnszone";
allow-query{any;};
masters {192.168.20.85;};
notify no;
};
zone "." {
type hint;
file "/etc/bind/db.root";
};
```

Výpis B.2: Výpis nastavení dns, ze souboru named.conf.options

```
options {
directory "/var/cache/bind";
forward first;
forwarders {
  192.168.20.85;
};
auth-nxdomain no;
listen-on {192.168.20.85; };
listen-on-v6 { ::; };
};
```

Výpis B.3: Výpis nastavení dns ze souboru openims.dnszone

```
$ORIGIN openims.
$TTL 1W
@       1D IN SOA       openims.  root.openims. (
        2006101001      ; serial
        3H              ; refresh
        15M             ; retry
        1W              ; expiry
        1D )            ; minimum
        1D IN NS        openims
pcscf                   1D IN A         192.168.20.85
_sip.pcscf    1D SRV 0 0 4060 pcscf
_sip._udp.pcscf   1D SRV 0 0 4060 pcscf
_sip._tcp.pcscf   1D SRV 0 0 4060 pcscf
icscf     1D IN A   192.168.20.85
_sip      1D SRV 0 0 5060 icscf
_sip._udp   1D SRV 0 0 5060 icscf
_sip._tcp       1D SRV 0 0 5060 icscf
openims.1D IN A         192.168.20.85
openims.1D IN NAPTR 10 50 "s"
"SIP+D2U" ""  _sip._udp.openims.
openims.1D IN NAPTR 20 50 "s"
"SIP+D2T" ""  _sip._tcp.openims.
scscf                   1D IN A         192.168.20.85
_sip.scscf              1D SRV 0 0 6060 scscf
_sip._udp.scscf         1D SRV 0 0 6060 scscf
_sip._tcp.scscf         1D SRV 0 0 6060 scscf
hss                     1D IN A         192.168.20.85
ue                      1D IN A         192.168.20.85
presence    1D IN A   192.168.20.85
opense      1D IN A   192.168.20.85
sipsee      1D IN A   192.168.20.85
anubis      1D IN A   192.168.20.85
xdms      1D IN A   192.168.20.85
xmldb     1D IN A   192.168.20.85
openpe      1D IN A   192.168.20.85
interceptor   1D IN A   192.168.20.85
omaco     1D IN A   192.168.20.85
```

Výpis B.4: Výpis nastavení dns, ze souboru openimsrev.dnszone

```
$TTL 86400
@ IN SOA    openims.openims. root.openims. (
        2006101001      ; serial
        3H              ; refresh
        15M             ; retry
        1W              ; expiry
        1D )            ; minimum
IN  NS  openims.openims
85    PTR pcscf.openims
85    PTR icscf.openims
85    PTR scscf.openims
85    PTR hss.openims
85  IN  PTR openims.openims
120 IN  PTR pc420.openims
```

Výpis B.5: Výpis řádků se změnou parametrů v souboru icscf.cfg

```
...
listen=192.168.20.85
...
alias=icscf.openims
alias=openims
...
modparam("icscf","name","icscf.openims")
...
modparam("icscf","forced_hss_peer","hss.openims")
...
modparam("icscf","icid_gen_addr","192.168.20.85")
modparam("icscf","orig_ioi","openims")
modparam("icscf","term_ioi","openims")
...
t_relay_to_udp("192.168.20.85", "9060");
...
if ( ! uri=~".*@openims.*")
...
if ( uri=~".*@openims.*")
...
```

Výpis B.6: Výpis řádků se změnou parametrů v souboru icscf.xml

```
...
FQDN="icscf.openims"
Realm="openims"
...
<Peer FQDN="hss.openims" Realm="openims" port="3868"/>
<Acceptor port="3869" bind="192.168.20.85"/>
...
<DefaultRoute FQDN="hss.openims" metric="10"/>
...
```

Výpis B.7: Výpis řádků se změnou parametrů v souboru icscf.sql

```
...
INSERT INTO ‘nds_trusted_domains‘ VALUES (1,’openims’);
...
INSERT INTO ‘s_cscf‘ VALUES
(1,’First␣and␣only␣S-CSCF’,’sip:scscf.openims:6060’);
...
```

Výpis B.8: Výpis řádků se změnou parametrů v souboru pcscf.xml

```
...
FQDN="pcscf.openims"
Realm="openims"
...
<Peer FQDN="clf.openims" Realm="openims" port="3868"/>
<Acceptor port="3867" bind="192.168.20.85"/>
...
<DefaultRoute FQDN="clf.openims" metric="10"/>
```

Výpis B.9: Výpis řádků se změnou parametrů v souboru pcscf.cfg

```
...
listen=192.168.20.85
...
alias=openims:4060
...
modparam("pcscf","name","sip:pcscf.openims:4060")
...
modparam("pcscf","ipsec\_host","192.168.20.85")
...
modparam("pcscf","rtpproxy\_socket",
"udp:192.168.20.85:34999")
...
modparam("pcscf","icid\_gen\_addr","192.168.20.85")
modparam("pcscf","orig\_ioi","openims")
modparam("pcscf","term\_ioi","openims")
...
modparam("pcscf","ecscf\_uri", "sip:ecscf.openims:7060")
...
P\_add\_p\_visited\_network\_id("openims");
...
P\_access\_network\_info("openims");
```

Výpis B.10: Výpis řádků se změnou parametrů v souboru scscf.cfg

```
...
listen=192.168.20.85
...
alias=scscf.openims:6060
...
modparam("scscf","name","sip:scscf.openims:6060")
...
modparam("isc","my_uri","scscf.openims:6060")
...
S\_assign_server\_unreg("openims", "orig");
...
if (uri=~"sip:(.*)@openims(.*)" || uri=~"tel:.*"){
...
S\_assign\_server\_unreg("openims", "term");
...
```

```
if (uri=~"sip:(.*)openims(.*)"){
...
if (!S_is_integrity_protected("openims")){
...
if (!S_is_authorized("openims")) {
S_challenge("openims");
...
if (S_assign_server("openims")){
...
if (S_assign_server("openims")){
...
if (S_assign_server("openims")){
...
if (S_assign_server("openims")){
...
if (S_check_visited_network_id("openims")){
...
t\_relay\_to\_udp("192.168.20.85",6060);
...
t\_relay\_to\_udp("192.168.20.85",6060);
...
if (uri=~"sip:(.*)@openims(.*)") {
...
if (uri=~"sip:(.*)@openims(.*)") {
...
if (uri=~"sip:\+[0-9]+@openims*user=phone.*"){
...
t\_relay\_to\_udp("192.168.20.85",6060);
...
```

Výpis B.11: Výpis řádků se změnou parametrů v souboru scscf.xml

```
...
FQDN="scscf.openims"
Realm="openims"
...
<Peer FQDN="hss.openims" Realm="openims" port="3868"/>
<Acceptor port="3870" bind="192.168.20.85"/>
...
<DefaultRoute FQDN="hss.openims" metric="10"/>
```

Výpis B.12: Výpis řádků se změnou parametrů v souboru DiameterPeerHSS.xml

```
...
FQDN="scscf.openims"
Realm="openims"
...
<Peer FQDN="icscf.openims" Realm="openims" port="3869" />
<Peer FQDN="scscf.openims" Realm="openims" port="3870" />
<Acceptor port="3868" bind="192.168.20.10" />
```

Výpis B.13: Výpis řádků se změnou parametrů v souboru hss.properties

```
host=192.168.20.85
```