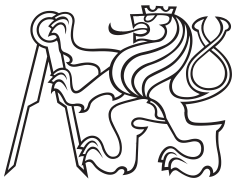
Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra počítačů

Nasazení správy IP adres ve středně velké síti

Jonáš Neuvirt

Vedoucí: Ing. Jan Kubr, Ph.D.
Studijní program: Softwarové inženýrství a technologie
Květen 2021

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Neuvirt** Jméno: **Jonáš** Osobní číslo: **420347**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra počítačů**

Studijní program: **Softwarové inženýrství a technologie**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Nasazení správy IP adres ve středně velké síti

Název bakalářské práce anglicky:

Pilot Implementation of IP Address Management

Pokyny pro vypracování:

1) Analyzujte aktuální nabídku open-source software implementující funkce DHCP, DHCPv6 a DNS serveru.
2) Analyzujte aktuální nabídku open-source software umožňující správu a dohled využívání IP a IPv6 adres v sítích LAN.
3) Navrhněte a realizujte laboratoř pro otestování nasazení zvolených software.
4) Vytvořte doporučení a popis postupu pro implementaci těchto služeb v reálné síti. Po dohodě s vedoucím práce vytvořte pilotní implementaci na části reálné sítě.
5) V případě nutnosti navrhněte, implementujte a otestujte software (nebo jeho část/doplněk) usnadňující správu IP adres a DNS jmen pomocí výše zvolených nástrojů.
6) Navrhněte postup pro otestování správného nasazení správy IP adres.

Seznam doporučené literatury:

Pekárek, Vít. Správa IP adres ve středně velké síti. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2019.
DROMS, Ralph. Dynamic Host Configuration Protocol [online]. 1993. Dostupné z: https://www.rfc-editor.org/rfc/rfc1541.txt. RFC. RFC Editor.
MOCKAPETRIS, Paul. Domain names – implementation and specification [online]. 1987. Dostupné z: https://www.rfc-editor.org/rfc/rfc1034.txt. RFC. RFC Editor.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Jan Kubr, Ph.D., katedra počítačové grafiky a interakce FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **14.02.2020** Termín odevzdání bakalářské práce: **14.08.2020**

Platnost zadání bakalářské práce: **30.09.2021**

Ing. Jan Kubr, Ph.D. podpis vedoucí(ho) práce	podpis vedoucí(ho) ústavu/katedry	prof. Mgr. Petr Páta, Ph.D. podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

. _____
Datum převzetí zadání

_____
Podpis studenta

Poděkování

Rád bych poděkoval panu Ing. Janu Kubrovi, Ph.D. za podporu a schovívavost, kterou projevil během vedení mé bakalářské práce. Zároveň děkuji mé rodině za jejich podporu během celého studia.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškeré použité informační zdroje.

V Praze 21. května 2021

Abstrakt

Cílem práce je navrhnout a otestovat laboratoř pro správu IP adres ve středně velké síti použitím vybraného softwaru pro DHCP, DNS a IPAM služby. Dalším úkolem je navrhnout postup pro implementaci těchto služeb v rámci reálné sítě a nakonec navrhnout postup pro otestování správného nasazení všech vybraných aplikací.

Klíčová slova: PowerDNS, Kea, phpIPAM, DNS, DHCP, IPAM, správa sítě

Vedoucí: Ing. Jan Kubr, Ph.D.

Abstract

The goal of this thesis is to design and test an implementation of IP address management in laboratory conditions using selected software for DHCP, DNS and IPAM services. Additional task is to design a method of deploying selected software in live environment and developing a strategy to test proper software deployment.

Keywords: PowerDNS, Kea, phpIPAM, DNS, DHCP, IPAM, network management

Title translation: Pilot Implementation of IP Adress Management

Obsah

Obrázky

Tabulky

Kapitola 1

Úvod

Správa větších počítačových sítí není jednoduchá, zvláště pokud by daná síť měla být spravována několika správci. V rámci sítě je potřeba mít služby Dynamic Host Configuration Protocol (DHCP) pro automatickou konfiguraci zařízení v síti a Domain Name System (DNS) pro převod doménových jmen na IP adresy. Zároveň je užitečné mít v rámci sítě dostupný IP monitoring system (IPAM), který poskytuje informace o dostupnosti a využití adres v síti.

Na Karlově náměstí se nachází budova Českého vysokého učení technického v Praze (ČVUT). Počítačová síť v této budově je spravována několika správci, ale pouze jeden je administrátorem, který muže provádět změny v konfiguraci. Pokud by některý ze správců chtěl provést změny v konfiguraci pro část sítě, za kterou je zodpovědný, musí požádat administrátora, aby tyto změny implementoval, čímž dochází ke zbytečným prodlevám. Bylo by proto vhodné, aby měli všichni jednotliví správci možnost provádět změny v nastavení sítě. V ideálním případě by měl každý správce mít možnost přístupu pouze k těm serverům či částem sítě, za které je zodpovědný. Použitím vhodných monitorovacích nástrojů pro přehled zapůjčených ip adres v síti pak bude možné zlepšit celkové využití přidělených adresních prostorů.

Přínosem této práce bude modernizace nynější sítě na Karlově náměstí. Použitím aktuálních technologií pro DHCP a DNS servery a přidáním IPAM tak dojde ke usnadnění správy sítě pro jednotlivé správce. Zároveň se tímto urychlí a zpohodlní proces provádění změn v konfiguraci, protože nebude třeba čekat na to, až administrátor implementuje požadované změny v konfiguraci. Využití vhodných monitorovacích nástrojů pak usnadní optimalizaci adresních prostorů v rámci jednotlivých podsítí tak, aby se zvýšil celkový počet dostupných ip adres v rámci celé sítě.

Kapitola 2
Analýza

V této kapitole je popsán nynější a plánovaný zmodernizovaný stav sítě v budově na Karlově Náměstí. Tato práce navazuje na bakalářskou práci předcházejícího studenta [1]. Závěry z této práce jsou zde ověřeny a doplněny o nové informace. Dále jsou zde zmíněny různé možnosti a nastavení jednotlivých vybraných nástrojů.

2.1 Nynější stav

Momentálně je pro DHCP server použitý nástroj ISC DHCP a pro DNS server je použitý nástroj PowerDNS. Nástroj ISC DHCP je poměrně starý, není možné jej spravovat výhradně přes databázové systémy, což způsobuje mnohá omezení. K administraci PowerDNS se používá nástroj PowerDNS Admin. Pro DHCP má momentálně každá katedra vlastní server, IP rozsah a správce. Níže jsou uvedeny možné výhody a vylešení, kterých se dá docílit přechodem na novou sadu nástrojů.

2.2 Požadavky na síť

Na síť jsou kladeny jisté požadavky, aby mohla dobře a spolehlivě fungovat. Jedná se o eliminaci potenciálních SPOF, čehož se dá docílit implementací hlavního a záložního serveru. DHCP i DNS servery musí umět pracovat jak s IPv4 tak s IPv6 adresami.

V rámci sítě je potřeba mít monitorovací nástroje, které umožní zobrazení různých statistik v rámci celé sítě. Především se jedná o přehled vypůjčených adres v rámci různých podsítí, aby se případně mohly optimalizovat konfigurace jednotlivých adresních prostorů těchto podsítí.

Pro efektivnější a bezpečnější správu sítě je požadována možnost omezit správu sítě pro jednotlivé administrátory. Každý správce by měl mít práva k provádění změn pouze v rámci jemu přidělené části sítě.

2.2.1 Seznam požadavků

P1 - Open-source
Navržený systém používá open-source aplikace.
P2 - Vzdálený přístup
Systém umožní adminstrátorům konfigurovat servery vzdáleně.
P3 - Detekce nepoužívaných ip adres v síti
Systém umožní administrátorům identifikovat nepoužívané ip adresy v síti.
P4 - Omezení administrace dle oprávnění
Systém umožní administrátorům spravovat pouze tu část sítě, za kterou jsou zodpovědní.
P5 - DHCP IPv4 a IPv6
Systém musí podporovat DHCP pro IPv4 a IPv6.
P6 - DHCP záložní server
Systém musí podporovat možnost hlavního a záložního DHCP serveru.
P7 - DNS záložní server
Systém musí podporovat master/slave konfiguraci pro DNS servery.
P8 - Migrace
Strategie nasazení musí zohlednit migraci dat mezi konfiguracemi jednotlivých serverů.

2.3 Porovnání DNS implementací

V následující části jsou popsány funkce, výhody a nevýhody několika nejpoužívanějších [2] volně dostupných open-source DNS implementací. Uvedené informace [3] jsou pak pro porovnání zobrazeny v tabulce 2.1 níže.

2.3.1 Bind9

Bind [4] je nejstarší a dodnes nejpoužívanější implementace DNS serveru, která splňuje nároky na všechny očekávané základní funkcionality. Verze Bind9 pak byla rozšířená o pokročilé funkcionality, především DNSSEC a TSIG pro zvýšení bezpečnosti a také přibyla podpora pro IPv6. Bind9 je možné spravovat přes příkazovou řádku nebo za pomocí webového rozhraní. Nevýhodou bindu je, že neumožňuje správu s použitím více administrátorských rolí, kde by se dal omezit přístup ke konfiguračním souborům dle nastavených práv.

2.3.2 Knot

Knot [5] je autoritativní DNS server, který je především zaměřen na použití pro domény nejvyššího řádu (anglicky Top Level Domain, zkráceně TLD), proto je kladen důraz na rychlost serveru. Zvlášť je pak vyvíjen Knot Resolver, který plní funkci rekurzivního DNS serveru. Tato implementace je používaná společností Cloudflare pro veřejný DNS server (ip adresa 1.1.1.1). Nevýhodou je malé množství možností pro správu konfigurace, ať už se jedná o podporu uživatelských rolí nebo o správu pomocí webového rozhraní.

Server	Autoritativní server	Rekurzivní server	Záložní server	DNSSEC	TSIG	IPv6	GUI
BIND9	✓	✓	✓	✓	✓	✓	✓
Knot	✓	✓	✓	✓	✓	✓	✗
djbdns	✓	✓	✓	✗	✗	✗	✓
Dnsmasq	✗	✓	✗	✓	✗	✓	✗
PowerDNS	✓	✓	✓	✓	✓	✓	✓

Tabulka 2.1: Tabulka vlastností jednotlivých implementací DNS

2.3.3 djbdns

Djbdns [6] je balíček aplikací, které implementují funkce DNS. Původní verze této aplikace je už zastaralá. Nepodporuje DNSSEC či IPv6 přímo v základu, pro podporu těchto funkcionalit je potřeba nainstalovat patche od různých třetích stran. Aplikaci je možné spravovat přes příkazovou řádku nebo pomocí webového nástroje NicTool.

2.3.4 Dnsmasq

Jedná se o velmi jednoduchou DNS implementaci, která se proto snadno konfiguruje. Dnsmasq je cílená na použití v rámci malé sítě, nenabízí funkcionalitu pro veřejně dostupný autoritativní server a neumožňuje konfiguraci jinak než přes příkazovou řádku. Nemá podporu uživatelských rolí a nepodporuje fungování autoritativního serveru v režimu master/slave.

2.3.5 PowerDNS

PowerDNS [7] patří mezi nejpoužívanější implementace DNS serveru. Autoritativní a rekurzivní servery jsou implementovány zvlášť. Obsahuje všechny funkcionality zmíněné u Bind9 (DNSSEC, TSIG, podpora IPv6), ale navíc přidává podporu pro správu konfigurace za použití administrátorských rolí a práv. Lze spravovat přes příkazovou řádku nebo přes webové rozhraní. PowerDNS nabízí REST API pro integraci s jinými systémy či aplikacemi, což vede k většímu množství implementací pro správu přes webové rozhraní. Jedná se buď o aplikace, které přímo přistupují k PowerDNS databázi (PDNS Manager, Poweradmin), nebo využívají poskytované API (PowerDNS Admin, PowerDNS Manager). Nejvhodnější aplikace pro správu přes webové rozhraní je PowerDNS Admin, jelikož umožňuje správu uživatelských rolí a práv, umí zobrazit statistiky a dovede zalogovat aktivitu jednotlivých uživatelů.

2.3.6 Výhody přechodu na PowerDNS

Hlavní výhodou přechodu na PowerDNS je podpora pro vytvoření libovolného počtu účtů pro správu těchto záznamů tak, aby jednotlivý správci měli přístup

k editaci jenom určitých částí konfigurace. Každý DNS server pak může mít přiřazenou vlastní MariaDB databázi obsahující všechny DNS záznamy s tím, že jednotlivý správci se k této databázi mohou připojit a editovat pouze záznamy ve svých doménách.

Aplikace PowerDNS sama o sobě tyto možnosti nenabízí, je proto nutné použití externích aplikací. Pro tyto účely byla zvolena aplikace PowerDNS Admin. Nabízí totiž všechny požadované funkce, má přívětivé grafické prostředí a je open-source. Zároveň se jedná o aplikaci, která je cílená přímo pro použití s PowerDNS, takže nehrozí případně potíže s podporou či kompatibilitou.

2.4 Porovnání DHCP implementací

V následující sekci jsou uvedeny a popsány dostupné open-source DHCP implementace.

2.4.1 Dnsmasq

Již zmíněný výše, Dnsmasq je schopen poskytnout vedle DNS serveru i DHCP server. Je schopen pracovat jak s DHCPv4 tak DHCPv6. Nepodporuje však funkcionality pro load balancing a failover [8], což ho činí nevhodným pro použití ve středně velké síti.

2.4.2 ISC DHCP

Jedná se o předchůdce Kea DHCP. ISC podporuje DHCPv4 i DHCPv6 a nabízí možnost fungování v režimu failover (nastavení jednoho serveru jako primární a druhého jako sekundární). Kromě toho, že tyto servery fungují jako záloha, ISC rovněž poskytuje funkci load balancingu.

2.4.3 Kea

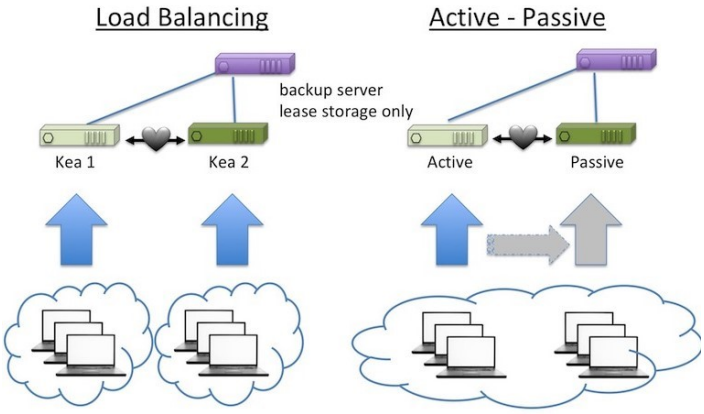
Kea je moderní DHCP server vytvořený autory ISC DHCP. Podporuje DHCPv4 i DHCPv6, nabízí REST API rozhraní a je schopen pracovat s nejpopulárnějšími databázovými systémy (např. MySQL, PostgresSQL či Cassandra). Dále obsahuje funkcionalitu High-Availability (dále HA), která plní stejnou funkci jako failover pro ISC DHCP. Kea rovněž poskytuje nástroj Kea Migration Assistant [9], který je schopen převzít ISC DHCP konfiguraci a převést ji na Kea DHCP konfiguraci.

Pro usnadnění správy pak existuje aplikace Stork, která je vyvíjená přímo pro Kea. Jejím cílem je poskytnout administrátorský nástroj, jenž je schopen sbírat a zobrazovat informace a usnadnit administraci Kea DHCP.

2.4.4 Výhody přechodu na Kea

Obdobně jako PowerDNS, i Kea podporuje ukládání záznamů do databáze. Tato databáze se pak dá využít pro synchronizaci záznamů mezi hlavním a záložním serverem. Existují dva různé způsoby, jak tohoto zálohování a

synchronizace mezi dvěma a více servery docílit. V obou případech servery využívají funkci HA, na obrázku 2.1 jsou tyto dva způsoby znázorněny. Servery pak mají možnost fungovat buď v režimu load balancing, kdy se zátěž rovnoměrně rozděluje mezi oba servery, nebo v režimu hot standby, kdy primární server obsluhuje celou síť sám a sekundární přebere jeho funkci pouze v případě výpadku. V rámci laboratoře byl zvolen režim load balancing.



Obrázek 2.1: Dva způsoby nasazení Kea DHCP serverů v režimu High Availability s použitím jedné databáze *(Zdroj: [10]).*

Kromě HA s použitím jedné databáze existuje i možnost použití více databází najednou v takové konfiguraci, že místo Kea serverů budou sdílení dat zařizovat právě samotné databáze. Obdobně by se toto dalo nakonfigurovat i v případě autoritativních serverů PowerDNS.

V rámci předchozí bakalářské práce se podařilo získat zdarma klíč pro placené rozšíření. Tento klíč je však platný pouze pro verzi 1.5. Podpora pro správu subnetů je rozšířením, které přibylo ve verzi 1.6 [11], konkrétně se jedná o knihovnu cb-cmds hook. V dokumentaci [12] lze vidět všechny placené funkcionality pro práci s databází. Dle [12] není tato knihovna naprosto nutná. Jedná se o modul, který má za cíl zjednodušení nastavování, nenabízí však funkcionality, které by jinak byly nedostupné.

Stejných operací se dá docílit několika dalšími způsoby. První možností je manuální vkládání záznamů do databáze. Tyto záznamy se mohou vkládat jak z terminálu, tak přes jakýkoliv volně dostupný GUI nástroj pro administraci MariaDB databází (např. phpMyAdmin). Druhou možností je pak upravit konfigurační soubor a ten zaslat danému serveru pomocí příkazu config-set,

který je součástí volně dostupného control agenta [13]. Poslední možností je počkat na případnou implementaci podpory v nástroji phpIPAM, který je v aktivním vývoji.

Obdobně jako v případě DNS, i zde jsou jisté požadavky na aplikaci, která usnadní správu sítě. Zde se jedná především o přehledný výpis statistik využití adres v rámci sítě či konfigurace podsítí jednotlivých serverů. Tyto možnosti nabízí aplikace Stork, která je poskytnutá stejnými vývojáři a je určena přímo pro práce s Kea. Stork momentálně nabízí možnost přehledu statistik jednotlivých podsítí (celkový počet adres, počet aktivně přidělených adres, počet volných adres), přehled rezervovaných adres (a zařízení, pro které jsou rezervované), informace o stavu jednotlivých serverů (jejich status, vytížení CPU a paměti) a informace o aktuální konfiguraci podsítí pro daný server.

Do budoucna jsou plánovány další funkcionality, aplikace Stork je v aktivním vývoji. Jedná se například o možnost nastavení alarmů a notifikací při pádu serverů či vyčerpání ip adres (přes email či v kombinaci s jinými aplikacemi, např. Grafana), podporu pro práci s logy nebo přidání testovacího prostředí (možnost simulovat uživatele v různých konfiguracích a scénářích). Dále se pak jedná a možnost přidání více uživatelských rolí a možnost správy a konfigurace serverů přímo přes aplikaci Stork, s tím že přístup k těmto konfiguracím bude možné omezit uživatelskými rolemi.

2.5 Porovnání IPAM implementací

2.5.1 PhpIPAM

PhpIPAM je jedna z nejrobustnějších open-source IPAM aplikací. Nevýhodou je fakt, že pro správnou funkčnost je třeba instalovat velké množství podpůrných aplikací a knihoven, které by mohly u některých verzí a distribucí způsobovat problémy s kompatibilitou.

PhpIpam umožňuje vést seznam zařízení v síti, použitých ip adres, VLAN sítí. Dále umí zobrazit obsazení podsítí a je schopen tyto informace zobrazit i v grafické podobě společně dalšími statistikami.

Hlavním důvodem pro instalaci PhpIpamu je jeho možnost automaticky skenovat podsítě a periodicky kontrolovat dostupnost ip adres. Lze nastavit jak často budou tyto pingy probíhat, případně lze zapnout možnost pro automatické prohledání sítě a přidání nových stanic do seznamu všech přítomných využívaných ip adres.

2.5.2 NetBox

Stejně jako PhpIPAM nabízí možnost správy IP adres, sítí, VLANů či zařízení v daných sítích. Narozdíl od PhpIPAMu ale Netbox nenabízí možnost kontroly dostupnosti strojů v síti či možnost otestovat síť pomocí pingů a přidání nových stanic do seznamu adres.

Další nevýhodou NetBoxu je nutnost instalovat větší množství podpůrných aplikací. Kromě databáze Postgres a webového serveru (možné mít apache

nebo nginx) je nutné instalovat také Redis a Gunicorn. Na druhou stranu je výhodou, že tyto aplikace by měly být funkční hned po instalaci se základní konfigurací.

2.5.3 Stork

Aplikace Stork je již zmíněná výše v sekci Kea. I když je primárně určena k administraci DHCP serverů, některé funkcionality se překrývají s IPAM aplikacemi. V tomto případě se jedná konkrétně o přehled podsítí, informace o jejich obsazenosti (maximální i nynější), a také informace o staticky přidělených adresách v rámci těchto podsítí.

2.6 Zavedení IPAM systému

V síti není momentálně používán žádný IPAM software. PhpIPAM nabízí nejen možnost monitorování IP adres v síti, ale nabízí i integraci s PowerDNS. Dá se tak připojit na databázi autoritativního PowerDNS serveru a provádět změny přímo z grafického rozhraní. Problém zde je v tom, že PhpIPAM neumí vhodně pracovat s přístupovými právy jednotlivých uživatelů. Proto je pro práci s PowerDNS servery vhodnější použít aplikaci PowerDNS Admin, která má tyto funkce implementované mnohem lépe.

V předchozí bakalářské práci je zmíněna plánovaná podpora i pro Kea DHCP, dle [14] ale tato funkcionalita stále není plně implementována[1], momentálně umožňuje pouze čtení, ale ne zápis.

PhpIPAM je velmi robustní aplikace s velkým počtem možností. Hlavní výhodou je možnost běhu na databázi MariaDB, stejně jako Kea a PowerDNS. Jak již bylo zmíněno výše, phpIPAM je schopen se připojit k databázím PowerDNS serverů a provádět změny v jejich konfiguracích.

Další možností je nainstalovat aplikaci Stork, která je primárně určena pro spolupráci s DHCP servery Kea, ale v rámci zobrazení statistik ohledně využívání a půjčování IP adres v síti zároveň poskytuje i určité informace o připojených zařízení (např. MAC adresy).

2.7 Budoucí stav

V ideálním případě by v síti byly nasazeny dva PowerDNS servery v konfiguraci master-slave. Ve skutečnosti by se však jednalo o čtyři servery, jelikož jeden server PowerDNS by využíval dva procesy, rekurzor a autoritativní server. Zároveň by bylo potřeba nainstalovat dva databázové servery, do kterých by se ukládaly jednotlivé DNS záznamy. Tyto databáze se mezi sebou budou vzájemně synchronizovat, aby byly DNS záznamy vždy aktuální.

Aby bylo možné spravovat tyto servery dle požadavků vycházejících ze zadání, bude třeba nainstalovat i aplikaci PowerDNS Admin. Ta umožní

[1] Podpora pro Kea byla prvně zmíněna už v roce 2016, ale dodnes není plně funkční.

správu uživatelů a editaci jejich přístupových práv. Uživatelé tak budou schopni spravovat pouze ty domény, které jim byly přiděleny.

Dále by byly nasazeny dva Kea servery v režimu HA, aby byly pokryty případné výpadky. Zde je opět vhodné podotknout, že by se ve skutečnosti jednalo o celkem čtyři servery, jelikož Kea implementace je rozdělená na DHCPv4 a DHCPv6. K optimálnímu chodu je nutné[2] mít i databáze. Do těchto databází se ukládají informace o aktuálně zapůjčených adresách. Toto je především nutné v případě, že Kea běží v režimu HA s možností load-balancing, kdy si servery rozdělují zátěž rovnoměrně mezi sebe. K synchronizaci vypůjčených adres se pak využívá nativní synchronizace mezi databázemi.

Pro umožnění a usnadnění správy DHCP serverů bude nainstalována aplikace Stork. Ta je schopna monitorovat stav jednotlivých serverů a zároveň podávat statistiky a informace o vypůjčených adresách v rámci sítě, včetně kapacit jednotlivých podsítí. Do budoucna je pak plánováno přidat podporu pro logování a konfiguraci přímo přes tuto aplikaci.

Na závěr by byl ještě nasazen phpIPAM, což znamená, že by byla třeba mít minimálně jeden webový server a jeden databázový server. Dle požadované úrovně zabezpečení a redundance se nabízí hned několik možností.

V první konfiguraci by byl použit jeden z už běžících databázových serverů, který by v ideálním případě byl umístěný na localhostu a zároveň by hostoval i databázi pro jeden z PowerDNS serverů. Nevýhodou je, že se tato konfigurace trochu blíží ke SPOF[3]. Druhou možností by bylo nainstalovat phpIPAM na vlastní stroj, kde by běžel pouze webový server. Pro databázi by se použil jeden z už nainstalovaných a nakonfigurovaných serverů s tím, že by se k němu muselo přistupovat vzdáleně.

Z výše zmíněného je vidět, že k implementaci všech nástrojů bude potřeba mít v síti minimálně jeden webový server, dva databázové servery, dva Kea DHCPv4 a Kea DHCPv6 servery, dva autoritativní PowerDNS servery a dva rekurzory.

[2]Databáze není pro HA chod klíčová, jelikož Kea může běžet i v konfiguraci s tzv. memfile, kdy si každý server ukládá data k sobě na disk a zároveň informuje ostatní serveru o případných změnách.

[3]Na jednom stroji by běželo větší množství aplikací najednou.

Kapitola 3

Návrh a realizace laboratoře

Před jakýmkoliv pokusem o nasazení v reálné síti je vhodné nejdříve instalovat a konfigurovat dané technologie v rámci laboratorního prostředí. V této kapitole je popsáno, jak byla tato testovací laboratoř navržená a realizovaná. Návrh vychází z provedené analýzy, ale je limitován dostupným hardwarem.

3.1 Použité technologie

Níže jsou uvedena všechna reálná i virtuální zařízení, která byla v rámci laboratoře použitá. Dále jsou zde uvedeny i všechny aplikace, které byly na těchto zařízeních nainstalované.

3.1.1 Hardware

Celá laboratoř je realizovaná ve virtuálním prostředí na stroji s operačním systémem Windows 10 s procesorem Intel Core i5-7200U CPU @ 2.5 GHz a 20 GB operační paměti.

Všechny virtuální stroje používají Debian ve verzi 10.4.0 se 4 GB operační paměti a pevnými disky o velikosti 32 GB. Jednalo se o čisté instalace ve VirtualBoxu pomocí příslušného iso obrazu [15].

3.1.2 Software

Většina aplikací byla k instalaci dostupná z debian repozitáře, výjimkou byly aplikace phpIPAM a PowerDNS Admin, které se musely instalovat z git repozitářů. U žádné z aplikací nebyla vyžadována specifická verze, vždy byly instalovány v nejnovější stabilní verzi. Při samotné instalaci nedošlo k žádným velikým potížím, ale vhledem k velkému množství instalovaných nástrojů není možné vyloučit, že se v budoucnu nějaké problémy vyskytnou, a to především při případném updatování již instalovaných nástrojů a knihoven.

U instalovaných balíčků se konkrétně jedná o phpIPAM v1.3, powerDNS v4.1, Kea v1.6, mariaDB v10.3, nginx 1.14.2, node.js 10.21.0 a php v7.3. Dále pak ještě phpIPAM vyžaduje php-pear v1.10 a řadu php modulů. Některé tyto moduly se nenainstalují automaticky v rámci php instalace a musí být instalovány manuálně. V několika případech nebyly některé požadované

balíčky přímo dostupné z Debian repozitáře, místo toho se nainstalovaly v rámci instalace jiných balíčků. Celý list těchto modulů je dostupný v phpIPAM dokumentaci [16].

Pro vytvoření virtuální sítě byl použit nástroj VirtualBox ve verzi 6.1.

3.2 Architektura sítě

V síti se nachází celkem čtyři virtuální stroje, z nichž jeden funguje čistě jako testovací stanice pro všechny aplikace.

První dva stroje obsahují vlastní instalace MariaDB databáze, autoritativních serverů PowerDNS a serverů KEA DHCPv4 a KEA DHCPv6. Na prvním stroji jsou navíc ještě instalace aplikací phpIPAM a PowerDNS Admin. Obě tyto aplikace fungují na webovém serveru nginx. Servery Kea DHCP jsou konfigurovány v režimu HA, instance MariaDB mají mezi sebou nastaveno nativní replikování. První stroj pak obsahuje server PowerDNS v nastavení master, druhý stroj má instalaci nakonfigurovanou v režimu slave. Třetí stroj pak obsahuje pouze samostatnou instalaci PowerDNS rekurzoru. Celé schéma je znázorněno na obrázku 3.1

3.3 Vytvoření a nastavení testovací sítě

Všechny virtuální stroje měly nastavený síťový adaptér v režimu NAT Network. V nastavení VirtualBoxu v záložce files > preferences > network se musela daná síť nejprve vytvořit a nakonfigurovat. Všechny virtuální stanice běží v tomto režimu a nacházejí se tak v jedné síti.

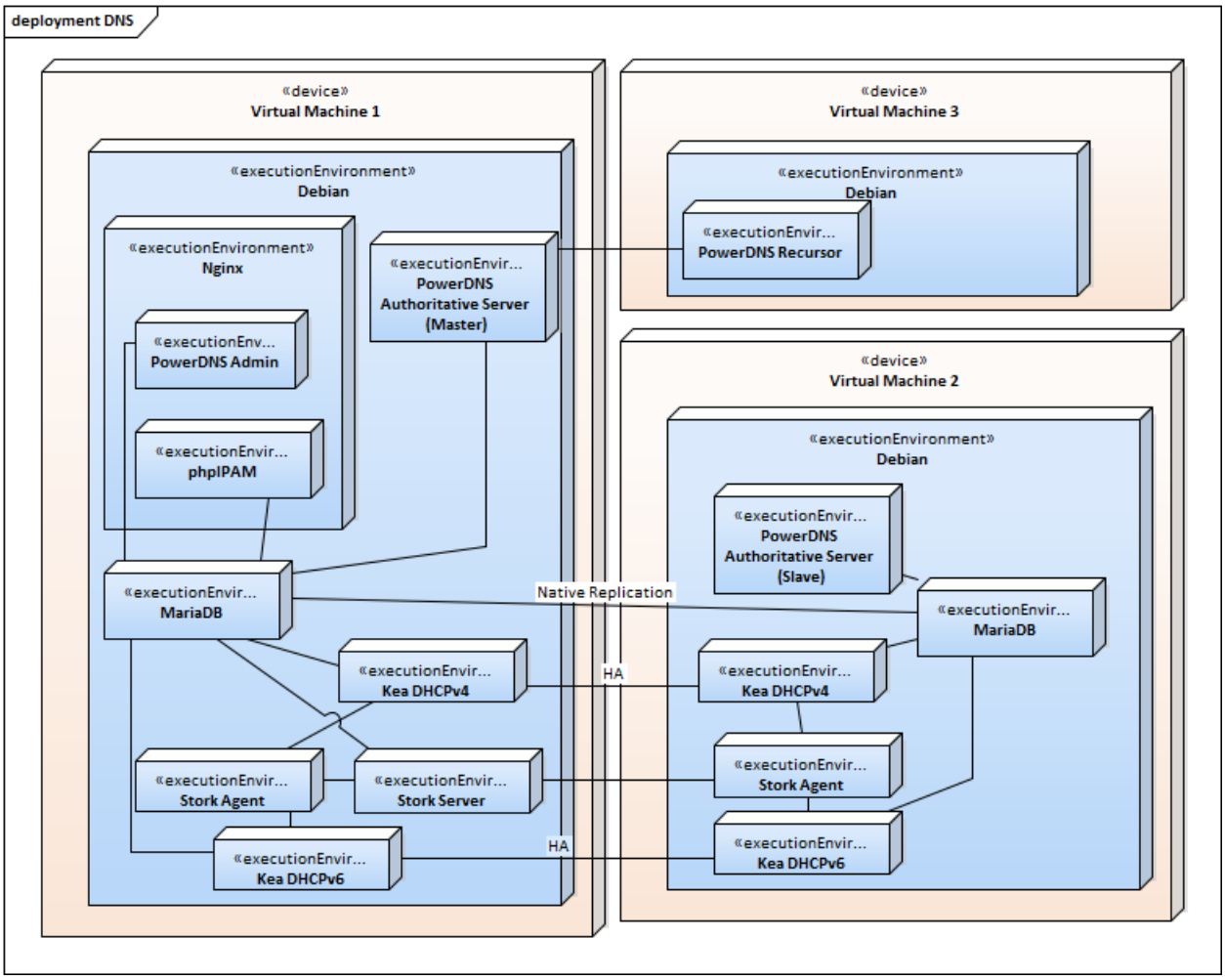
Adresní rozsah IPv4 byl nastaven na 10.0.0.0/8 a záměrně byla v rámci této sítě vypnuta podpora DHCP. Virtuální stroje fungující jako servery měly síťové adaptéry nastaveny manuálně, první měl adresu 10.0.0.10, druhý 10.0.0.20 a obdobně pro další stanice.

Adresy pro IPv6 byly zvoleny z rozsahu 2001:DB8::1:0/64 a jednotlivé virtuální stroje měly přidělené adresy 2001:DB8::1:10, 2001:DB8::1:20 a obdobně pro další stroje.

3.3.1 MariaDB

Jelikož je databáze potřebná pro funkci PowerDNS, Kea i phpIPAM, bylo vhodné ji nainstalovat a nakonfigurovat jako první. V rámci sítě byly nastaveny dva databázové servery. První sloužil jako hlavní databáze, ve které se nacházela data pro PowerDNS master server, phpIPAM, powerDNS Admin. Kea DHCP. Druhý MariaDB server sloužil jako záložní databáze pro PowerDNS slave server a pro Kea DHCP servery.

Pro všechny tři aplikace byly vytvořeny potřebné databáze, jejichž schémata jsou dostupná online nebo v instalačním adresáři jednotlivých aplikací. Tato schémata se pak dají snadno naimportovat, jen je třeba mít na paměti, že se



Obrázek 3.1: Schéma laboratoře.

mohou měnit v rámci vydání nových verzí, např. pro PowerDNS jsou databáze pro verze 4.1 a 4.2 odlišné.

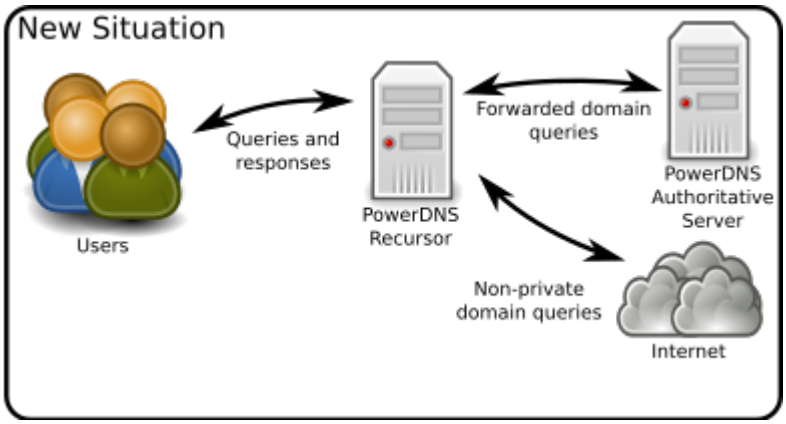
Samotná instalace pak probíhala dle pokynů v dokumentaci [17] a s pomocí online návodu [18]. Vytváření a nastavení jednotlivých databází a uživatelských účtů se pak řídilo pokyny pro instalaci jednotlivých aplikací, které tyto databáze využívaly.

V rámci instalace obou MariaDB databází bylo nastaveno master-slave replikování dle dokumentace [19]. Tato varianta nastavení je doporučená především při použití PowerDNS master-slave konfigurace viz dokumentace [20]. Alternativním řešením pro synchronizaci záznamů v databázi je použití AXFR. Nevýhodou AXFR je to, že pokud dojde k smazání zóny na master serveru, slave server se tomto nikdy nedozví. Jediným řešením je pak nepoužívané zóny odebírat ručně. Tento problém v případě replikace přes databázi nehrozí.

3.3.2 PowerDNS

Instalace PowerDNS probíhala dle příslušných dokumentací pro autoritativní server [20] a rekurzor [21]. PowerDNS byl konfigurován v režimu master-slave. Na prvním stroji se nacházela příslušná primární MariaDB databáze a master autoritativní serverem. Na druhém stroji se pak nacházela záložní MariaDB databáze a k ní patřící slave autoritativní server.

Pro nasazení byla zvolena konfigurace s rekurzorem, znázorněno na obrázku 3.2. Tato konfigurace je v praxi akceptována jako nejvhodnější. Má výhody především v oblasti bezpečnosti (např. zvýšená odolnost proti DoS útokům) a výkonnosti serverů (lepší rozložení zátěže), více informací např. zde [22]. Tento rekurzor byl nainstalován na třetím stroji. V reálném prostředí by pak bylo vhodné přidat záložní rekurzor na čtvrtý stroj, ať se eliminuje SPOF.



Obrázek 3.2: Znázornění konfigurace s rekurzorem. *(Zdroj: [23]).*

U rekurzorů je potřeba nastavit domény, u kterých dojde k přeposílání všech DNS dotazů na autoritativní server, viz obrázek 3.3. Zároveň je u rekurzoru dobré omezit adresní rozsahy, ze kterých bude přijímat dotazy. Implicitně jsou povoleny pouze privátní adresní rozsahy definované RFC 1918 a loopbacky. Pro master a slave rekurzor je konfigurační soubor stejný, výjimkou je pouze lokální adresa serveru. U autoritativních serverů je kromě lokální adresy potřeba změnit i přihlašovací údaje pro připojení do databáze, příklad takové konfigurace je zobrazen na obrázku 3.4 .

```
local-address=10.0.0.10, 2001:DB8::1:10
local-port=53
allow-from=127.0.0.0/8, 10.0.0.0/8, 001:DB8::1:0/64

forward-zones=example.com=127.0.0.1:5300
forward-zones+=pokus.example.com=127.0.0.1:5300
```

Obrázek 3.3: Ukázka části konfiguračního souboru master rekurzoru.

Dále je nutné nakonfigurovat webserver a api, aby bylo možné použít PowerDNS Admin aplikaci pro správu jednotlivých uživatelů, jejich práv, domén a zón. Údaje uvedené v těchto konfiguračních souborech se poté

```
launch=gmysql
gmysql-host=127.0.0.1
gmysql-user=admin
gmysql-password=debian
gmysql-dbname=powerdns

local-address=127.0.0.1
local-ipv6=::1
local-port=5300
```

Obrázek 3.4: Ukázka části konfiguračního souboru autoritativního serveru.

musí nakonfigurovat i v PowerDNS Admin aplikaci, jinak nebude databázové spojení správně fungovat. V úspěšné konfiguraci byla použita adresa autoritativního serveru. Aby bylo možné navázat spojení s databází přes PowerDNS Admin, musela být do konfigurace přidána adresa, ze které mohou aplikace přes api k databázi přistupovat. Jedná se o řádek webserver-allow-from, viz ukázka konfigurace na obrázku 3.5.

Relevantní část konfigurace autoritativního serveru

```
api=yes
api-key=api
webserver=yes
webserver-address=10.0.0.10
webserver-allow-from=0.0.0.0/0,::/0,10.0.0.10
webserver-port=8081
```

Relevantní část konfigurace rekurzoru

```
webserver=yes
webserver-address=10.0.0.10
webserver-port=8082
webserver-allow-from=0.0.0.0/0,::/0,10.0.0.10
```

Obrázek 3.5: Ukázka konfigurace api webového rozhraní.

3.3.3 PowerDNS Admin

Hlavní informace k instalaci pocházely z wiki stránky projektu na gitlabu [24]. Tyto informace ale neobsahují všechny potřebné kroky pro správné nastavení a nakonfigurování aplikace. Proto úplné zprovoznění tak byly použité i návody z webových stránek a blogů nacházejících se zde [25] a [26] a [27].

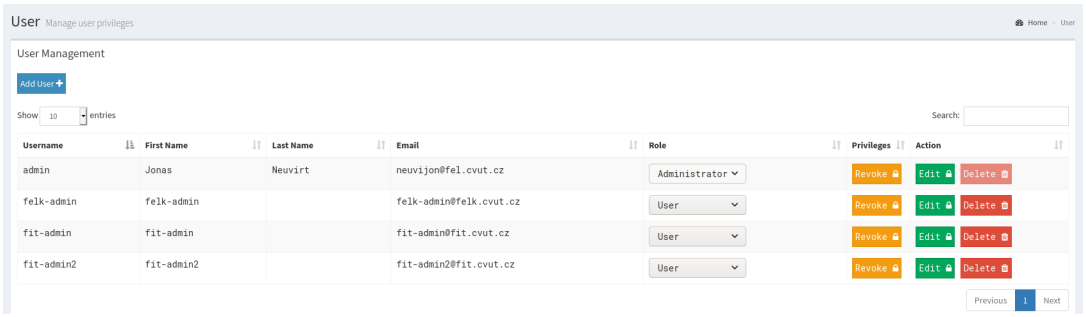
Mnoho komplikací se správnou funkčností spočívalo ve správném nastavení práv pro určité soubory a některé adresáře. Bylo nutné správně nastavit vlastníka pro kořenový webový adresář aplikace (v mém případě www-data:www-data). Zároveň ale bylo třeba u některých podadresářů nastavit vlastníka a skupinu tak, aby odpovídaly hodnotám setuid a setegid uvedeným v konfiguračních souborech pro aplikaci PowerDNS.

Další komplikací bylo zprovoznění socketu, na kterém PowerDNS Admin naslouchal. Při konfiguraci se samostatně nastavuje PowerDNS-Admin.service a PowerDNS-Admin.socket viz [28]. Správně by se měla zapnout PowerDNS-Admin.service, která pak spustí i příslušný socket. Pokud ale dojde k restartování této service, dojde i k restartování socketu. Z nějakého důvodu (který dodnes není znám) se tento socket po restartování nejdříve otevře, ale poté se ihned uzavře. Proces PowerDNS-Admin.service si i přesto myslí, že je vše

v pořádku. Výsledkem je, že aplikace není schopná se spojit s PowerDNS databází.

Nalezené řešení je, že před každým restartováním PowerDNS-Admin.service je nejprve nutné manuálně zastavit PowerDNS-Admin.socket. Až poté je možné nastartovat proces PowerDNS-Admin.service, který si zároveň při inicializaci otevře i svůj příslušný funkční socket.

Při prvním spuštění PowerDNS-Admin je nutné se nejprve zaregistrovat. První registrovaný uživatel se pak stává administrátorem s plnými přístupovými právy, všichni následující registrovaní uživatelé začínají v roli user. Admin uživatel poté může tyto role libovolně měnit, ukázka na obrázku 3.6.



Obrázek 3.6: Ukázka správy uživatelů v aplikaci PowerDNS Admin.

Přiřazení jednotlivých uživatelů ke správě domén pak probíhá pomocí tzv. accounts (účtů). Po registraci uživatel nemůže nic spravovat. Nejprve je nutné ho přiřadit do skupiny uživatelů, která má právo editovat všechny domény, které jsou vlastněny daným účtem. Na obrázku 3.7 je ukázána konfigurace pro účet jménem fit. Na levé straně je okno se všemi registrovanými uživateli v rámci celé aplikace. Na pravé straně jsou uživatelé fit-admin1 a fit-admin2, kteří spadají pod účet fit. Tito uživatelé pak mohou spravovat všechny domény, které rovněž spadají pod tento účet.

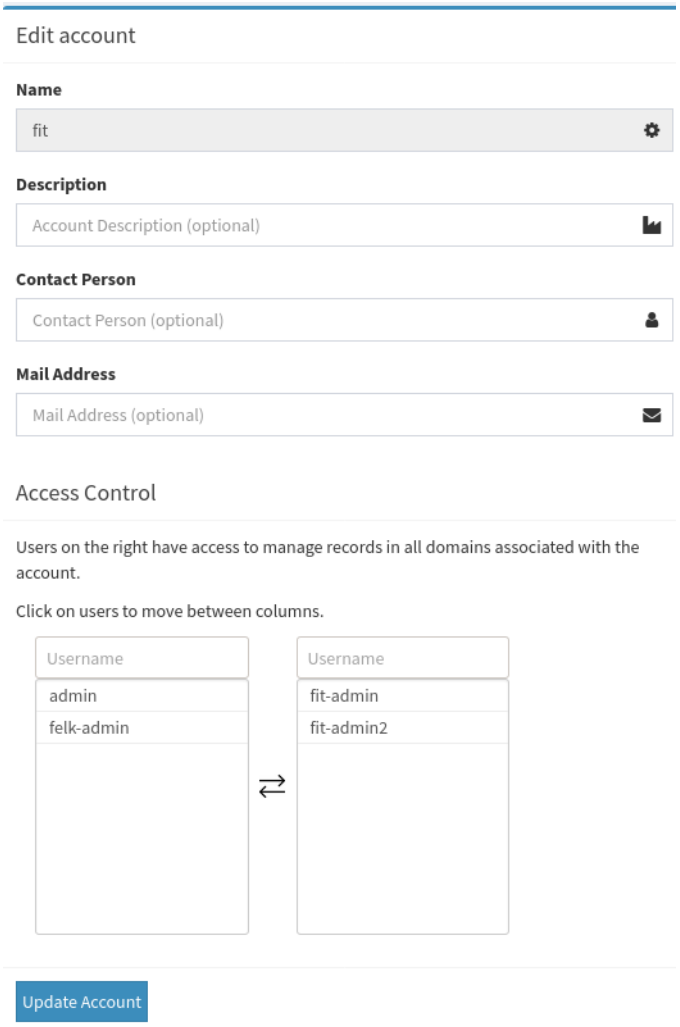
Na obrázku 3.8 je ukázka pro vytvoření domény feld.cvut.cz. Při jejím vytváření je nutno uvést účet, pod který tato doména spadá (zde se jedná o účet feld). Všichni uživatelé, kteří jsou přiřazení k účtu feld, pak mohou provádět změny v konfiguraci této domény. Na obrázku 3.9 je pak ukázka vytváření nového záznamu pro doménu felk.cvut.cz, kde lze rovněž vidět i všechny ostatní záznamy spadající pod tuto doménu.

Při přihlášení je vidět na úvodní straně přehled všech hostovaných domén. Ukázka je na obrázku 3.10, kde vybraná záložka Hosted Domains, která zobrazuje domény IPv4.

3.3.4 PhpIPAM

K instalaci phpIPAM byla použitá dokumentace [16] a návod [29].

PhpIPAM byl instalován na stejném stroji, kde byl instalován PowerDNS master server a PowerDNS Admin. Důvodem je přístup k příslušné databázi pouze z localhost, a to z důvodu bezpečnosti. V případě, že by se vzdálený přístup k databázi nepavažoval za bezpečností riziko, bylo by možné instalovat



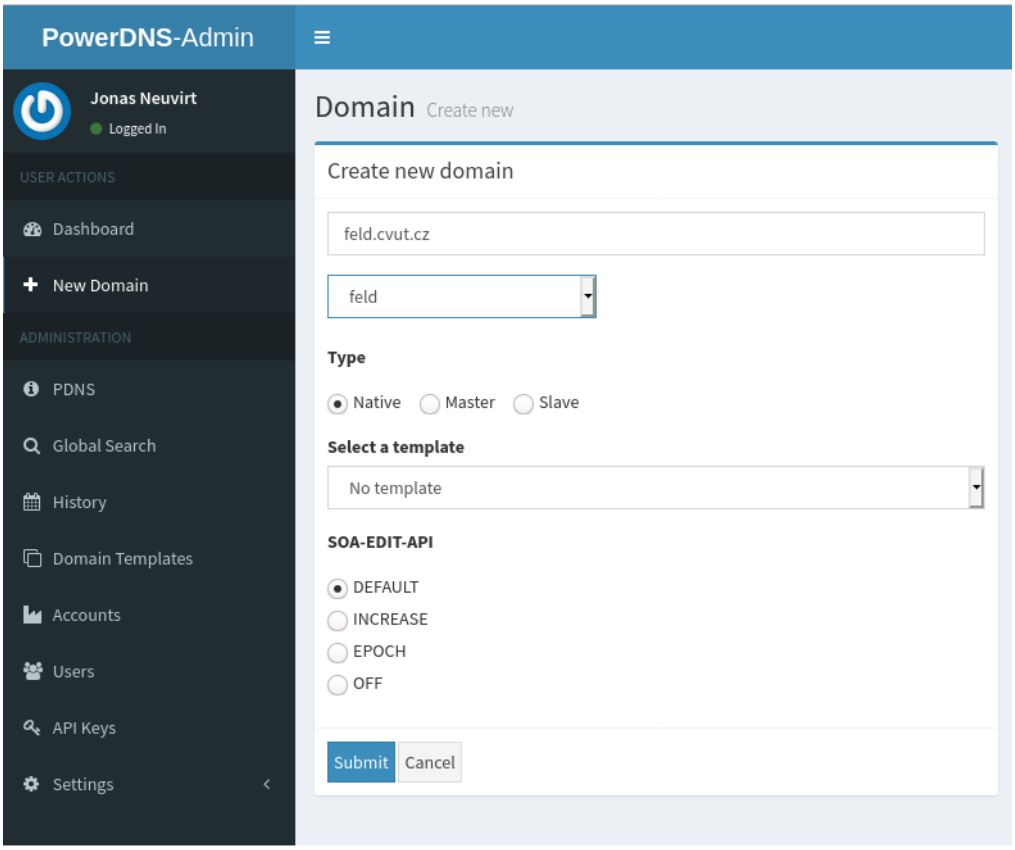
Obrázek 3.7: Ukázka přiřazení uživatelů fit-admin1 a fit-admin2 pod účet fit.

phpIpam na libovolný stroj. Pokud by pak bylo potřeba přistupovat k databázi mimo localhost, stačilo by vytvořit uživatele s oprávněním přístupu z venčí. Lze nastavit i přístup z konkrétní ip adresy, takže se tím dá limitovat přístup na konkrétní stanici v síti.[1]

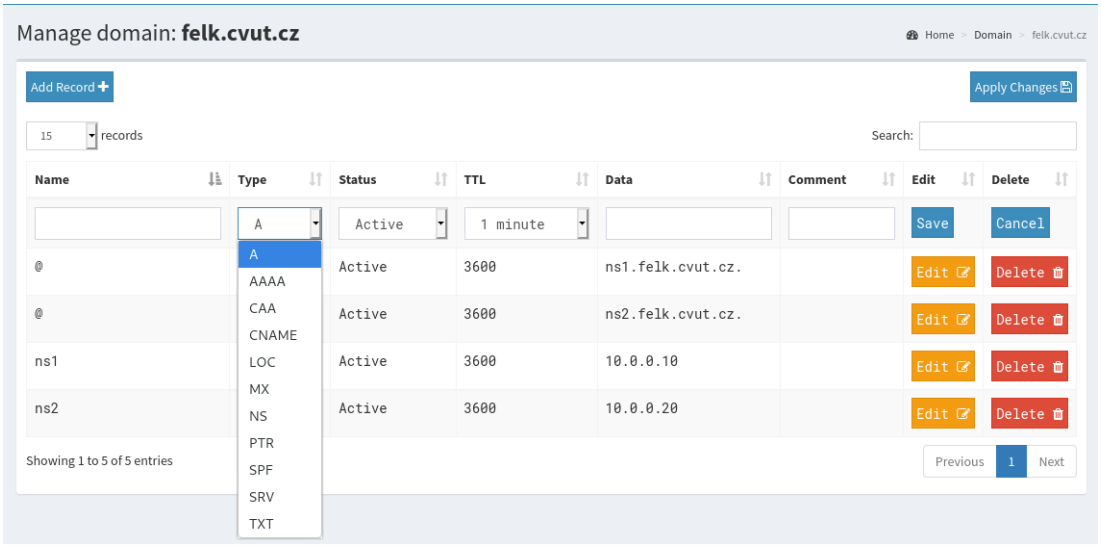
V případě, že je požadováno, aby zde nevznikal SPOF, nabízí se několik možností. V případě selhání webového serveru by bylo nutné mít nainstaloaný záložní webový server s druhou instalací phpIPAMu. V případě selhání primární databáze je jedinou možností změnit konfiguraci phpIPAMu tak, aby se připojil na záložní databázi. Tato akce nejde automatizovat, takže v případě výpadku se tato změna v konfiguraci musí provést manuálně.

Po instalaci je nutné integraci s PowerDNS zapnout v záložce nastavení. Aby bylo možné spravovat DNS záznamy, musí se nastavit přihlašovací údaje pro připojení k databázi PowerDNS serveru. K tomu je potřeba vytvořit pro phpIPAM uživatelský účet se správně nastavenými přístupovými právy.

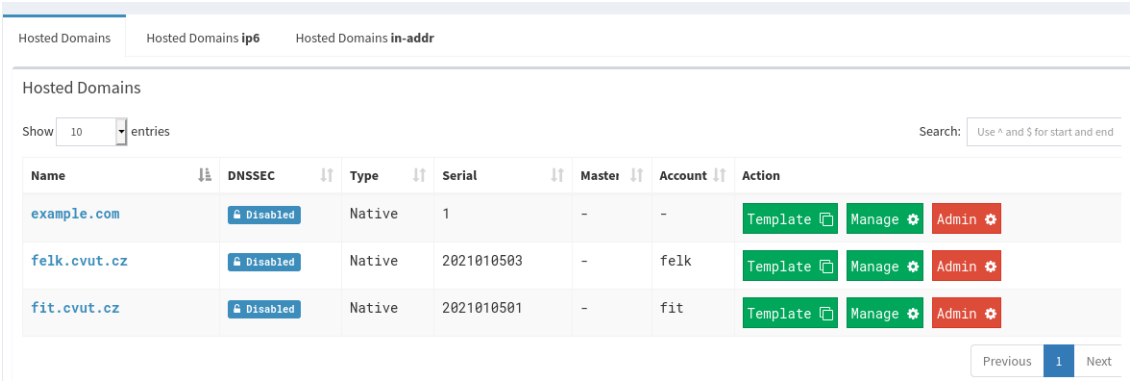
[1] Za předpokladu, že daná ip adresa je nakonfigurována staticky.



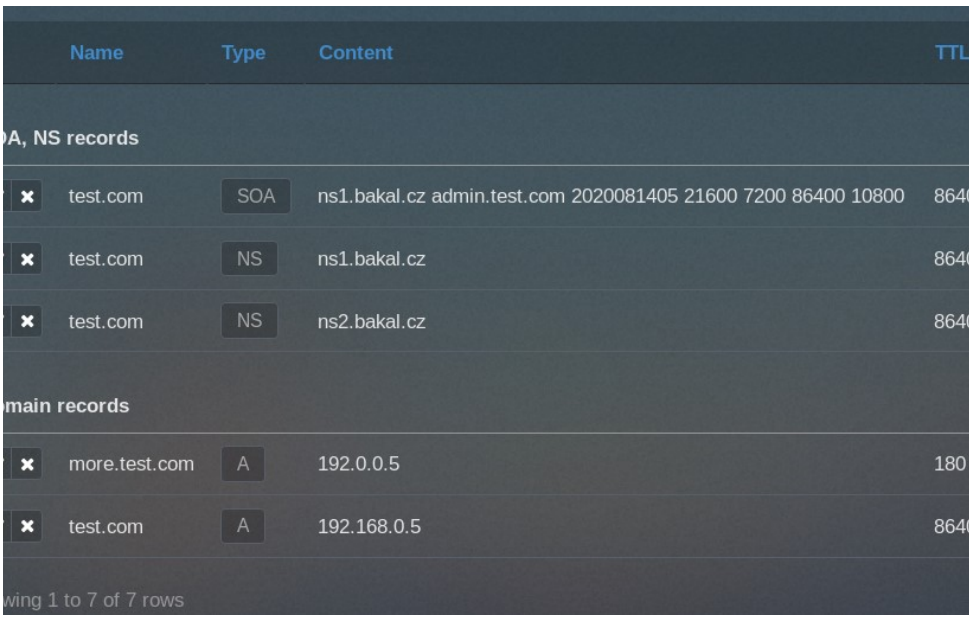
Obrázek 3.8: Ukázka vytvoření nové domény feld.cvut.cz.



Obrázek 3.9: Ukázka vytvoření nového záznamu pro doménu felk.cvut.cz.



Obrázek 3.10: Přehled všech hostovaných domén.



Obrázek 3.11: Ukázka správy doménových záznamů PowerDNS v phpIPAM.

3.3.5 Kea

Podle dokumentace [30] přibyla od verze 1.6 možnost instalovat Kea pomocí balíčků. Konfigurace pak vycházela z online dokumentace [31] a záznamu z webináře [32]. Zároveň zde byla nainstalována i MariaDB databáze. Kea je implementovaná zvlášť pro DHCPv4 a DHCPv6, takže nakonec v rámci sítě musí běžet celkem čtyři Kea servery a dva MariaDB servery.

Tento server má povolený vzdálený přístup, který je nutný v případě, že požadujeme, aby v síti mohlo být více DHCP serverů najednou. Tyto servery jsou pak nastaveny v konfiguraci High Availability, přičemž každý přistupuje do své vlastní databáze, které se synchronizují mezi sebou samostatně bez zásahu Kea serverů.

Zároveň je dobré nastavit heartbeat interval na rozumnou hodnotu, aby při výpadku jednoho serveru netrvalo příliš dlouho, než se o tom sekundární

server dozví. V dokumentaci je ještě popsaná alternativní metoda nasazení, kdy oba servery sdílejí stejnou databázi, synchronizaci řeší Kea servery mezi sebou. Aby zde nevznikal SPOF, je nutné mít databáze umístěné v clusteru, který poté přebírá zodpovědnost za poskytování přístupu k databázi.

3.3.6 Stork

Instalace probíhala z balíčků dostupných pro debian dle online dokumentace [33]. Stork je složený ze dvou částí - Stork Server a Stork Agent. Na celou síť stačí jeden Stork server, který pak komunikuje s jednotlivými Stork agenty. Tito agenti se nacházejí na každém stroji, kde běží nějaká instance kea-dhcp serverů. Tito agenti pak monitorují jednotlivé stroje (CPU, RAM, status) a informace posílají Stork Serveru. Momentálně je podporována pouze databáze PostgreSQL, takže musela být prozatímně nainstalována v rámci laboratoře. Do budoucna je plánována i podpora dalších databázových systémů. Aby byl Stork Server schopen číst statistiky o zapůjčených ip adresách, je třeba mít nakonfigurovanou knihovnu Stat Commands. Tato knihovna je volně dostupná všem uživatelům. Pro čtení rezervovaných ip adres z databáze je třeba mít knihovnu Host Commands, která je však placená.

Po instalaci je automaticky vytvořen uživatel admin. Ten pak může přidat další uživatele. Pro sledování statistik je nejprve nutné zaregistrovat jednotlivé stroje v rámci Stork aplikace. Přidávají se přes grafické rozhraní za použití ip adres a portů všech strojů, na kterých běží jednotlivý agenti. Po připojení agentů je pak Stork schopen zobrazovat informace i jednotlivých strojích a jejích nakonfigurovaných podsítí včetně jejich obsazení.

Kapitola 4

Strategie nasazení

Strategie pro nasazení všech aplikací v rámci v reálné sítě by se výrazně nelišila od postupu nasazování během laborek, jelikož návrh laboratoře bral v úvahu nasazení v reálném provozu. Ideálně by se jednotlivé servery připravili mimo aktivní síť. Konkrétně by se jednalo o nastavení konfiguračních souborů v jednotlivých virtuálech. Po otestování funkčnosti v rámci uzavřeného prostředí by se pak tyto servery zpřístupnily pro celou síť. Metodika byla inspirována několika články, které diskutovali vhodné praktiky ([34], [35], [36], [37], [38], [39])

4.0.1 Nasazení DNS

Pro nasazení DNS by bylo vhodné nasadit hlavní a záložní DNS server tak, že běží zároveň s aktivními DNS servery. Jelikož však nejsou nabízeny v rámci DHCP konfigurace, uživatelé v síti je nebudou využívat. V této době dojde k migraci konfiguračního souboru z BIND do PowerDNS databáze, dokumentace nabízí hned několik možností [**dnsmil1**]. Jakmile se otestuje správná funckionalita (konfigurace je v pořádku), je možné upravit adresy DNS serverů tak, aby byly nabízeny v rámci sítě i nové DNS servery.

4.0.2 Nasazení DHCP

Autoři Kea DHCP vytvořili nástroj Kea Migration Assistant **??**, který je schopen zkonvertovat konfigurační soubor z ISC DHCP na Kea DHCP. Výjimkou jsou rezervované adresy, ty musí administrátoři převést manuálně. Společně s ním sepsali i doporučený postup pro migraci. Po konverzi ISC konfiguračního souboru na Kea konfigurační soubor je nutné jej nejprve otestovat v laboratorním prostředí.

Jakmile je Kea konfigurační soubor připraven, je možné zahájit přechod. Momentálně neexistuje žádný nástroj na převedení již zapůjčených adres, což znamená, že výpadkům se zde vyhnout nedá. Nové servery totiž neví, které adresy jsou už zapůjčeny, takže by pak mohli nabízet ještě obsazené adresy. Jednotlivé stanice si tudíž musí po přechodu na Kea vyžádat nové adresy.

Tato nepříjemnost se dá zmírnit dvěma opatřeními. Za prvé je možné tyto přechody provést v méně aktivních hodinách. Za druhé je možné snížit dobu, na kterou se jednotlivé adresy zapůjčují. Jakmile uplyne doba krátkého lease

time, víme, že v síti už nejsou žádné obsazené adresy, a můžeme tak přejít na nové DHCP servery s tím, že není šance, aby nastaly případné kolize.

Ve větších sítích ale může nastat problém v případě, že se nastaví velké adresní rozsahy s příliš krátkým lease time, jelikož by se v síti mohl výrazně zvýšil počet DHCP dotazů. Pokud byl původní lease time velmi dlouhý, bude třeba nejprve lease time zkrátit a poté počkat, než vyprší (v případě, že je několik dnů dlouhý).

4.0.3 Nasazení databází

MariaDB databáze lze do sítě zapojit bez výraznějších problémů. Jednotlivá databázová schémata jsou dostupná u instalačních souborů či v instalačních adresářích, při samotném importování je velmi nepravděpodobné, že by došlo k chybě, která by mohla způsobit výpadek databáze. V případě, že by se v síti již nacházela běžící databáze, není problém do ní za běhu vložit nové tabulky, které by byly nutné pro instalaci nových aplikací.

Kapitola 5

Postup testování

V této kapitole bude rozebráno a navrženo, jak je možné síť otestovat. Testování výsledné konfigurace je možné rozdělit do dvou hlavních kategorií. Jedná se nejprve o testování funkčnosti všech instalovaných nástrojů a poté o zátěžové testování.

Druhou testovací kategorií by pak byly testy zátěžové, kde by se kladl důraz na rychlost a výkonnost jednotlivých serverů. Do této části testování by bylo zahrnuto i testování sítě na výpadky různých jejích částí, a testovalo by se, jak dobře se s těmito nečekanými situacemi síť vypořádá.

5.1 Testování funkčnosti

V rámci první kategorie by se jednalo o testování funkcionality všech implementovaných nástrojů. Toto testování by se soustředilo především na testování základních činností jednotlivých aplikací v kontrolovaném prostředí, kdy nehrozí, že by nastaly nějaké nenadálé situace. Pokud by se toto testování dokončilo úspěšně, bylo by možné přejít do následující fáze testování. V této fázi lze i simulovat výpadky jednotlivých serverů a testování toho, jestli správně fungují záložní servery.

Hlavními vhodnými nástroji pro toto testování jsou ping (test dostupnosti jednotlivých aplikací v rámci sítě) a nástroj dig, který je dostupný v balíčku dnsutils a umožňuje vytvářet DNS dotazy. Výpadky jednotlivých serverů se dají simulovat zastavením a spouštěním příslušných procesů.

5.2 Zátěžové testování

Druhou testovací kategorií by pak byly testy zátěžové, kde by se kladl důraz na rychlost odezvy a vyřízení jednotlivých požadavků. Poté by přišlo na řadu výkonnostní testování jednotlivých serverů, kde by se dalo zjistit maximální možně množství požadavků, které servery zvládnou vyřídit najednou.

Kea poskytuje vhodný nástroj pro toto zátěžové testování nazvaný perfdhcp, dokumentace a příklady použití jsou dostupné online [40] a i jako video zde [41]. Tento nástroj simuluje zařízení v síti a zároveň sbírá data a počítá

statistiky (např. počty přijatých, odeslaných a ztracených paket či rychlost odezvy).

Kapitola 6

Závěr

V rámci vypracovávání bakalářské práce jsem potýkal se síťařskými problémy, které se velice pravděpodobně v praxi běžně vyskytují. V rámci práci jsou adresované všechny požadavky, které byly uvedeny v analýze. Osobně jsem vůbec nepočítal s tím, jak náročné by mohlo být navrhnout a nakonfigurovat laboratorní síť obsahující DHCP, DNS a IPAM nástroje od úplného počátku. Jedná se například o problémy typu neviditelných chyb v konfiguračních souborech, kterých se jen velmi těžko zbavuje. Konkrétně bych zmínil třeba tečku na konci DNS záznamu, která se dostala do databáze narušila funkčnost PowerDNS serverů. Odhalena byla až po jednom dni hledání. Rovněž se mi povedlo při instalaci databáze vybrat špatnou verzi databázového schématu. Trvalo poměrně dlouho, než jsem přišel na to, proč mi nefunguje správně ukládání dat do databáze.

Celkově jsem se naučil mnoho nových věcí a především jsem získal mnoho cenných zkušeností v oblasti konfigurace počítačových sítí, tvorby a virtualizace strojů včetně práce s virtuálními sítěmi. Zároveň jsem výrazně zlepšil v rámci práce s linuxovými operačními systémy. Navíc pak ještě mnohem lépe rozumím tomu, jak přistupovat ke konfiguraci a instalaci více než jedné aplikace v rámci jedné sítě.

Bibliografie

[1] Vít Pekárek. *Správa IP adres ve středně velké síti.* Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2019. Dostupné z: https://dspace.cvut.cz/handle/10467/83239.

[2] Don Moore. *DNS server survey.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: http://mydns.bboy.net./survey/.

[3] the free encyclopedia Wikipedia. *Comparison of DNS server software.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://en.wikipedia.org/wiki/Comparison_of_DNS_server_software.

[4] Joseph Caudle. *The Top DNS Servers And What They Offer.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://blog.dnsimple.com/2015/02/top-dns-servers/.

[5] knot-dns.cz. *Knot DNS.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.knot-dns.cz/.

[6] ntchosting.com. *DNS Software.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.ntchosting.com/encyclopedia/dns/software/.

[7] Jonas Lejon. *Top DNS Server Software.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://blog.hostdns.com/top-dns-server-software/.

[8] the free encyclopedia Wikipedia. *Comparison of DHCP server software.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://en.wikipedia.org/wiki/Comparison_of_DHCP_server_software.

[9] Suzanne Goldlust. *Migrating from ISC DHCP to Kea DHCP using the Migration Assistant.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://kb.isc.org/docs/migrating-from-isc-dhcp-to-kea-dhcp-using-the-migration-assistant.

[10] Vicky Risk. *Kea 1.4 Adds High-Availability Mode.* [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.isc.org/blogs/kea-1-4-adds-high-availability-mode/.

[11] Vicky Risk. *Kea 1.6 Adds Configuration Database*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://www.isc.org/blogs/kea-1-6/.

[12] Internet Systems Consortium. *API Reference*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://kea.readthedocs.io/en/latest/api.html.

[13] Internet Systems Consortium. *API Reference*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://kea.readthedocs.io/en/kea-1.6.1/api.html?highlight=config-set.

[14] Miha Petkovsek. *Kea DHCP management functional?* [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://github.com/phpipam/phpipam/issues/777.

[15] debian.org. *Debian CD/DVD images*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://cdimage.debian.org/debian-cd/current-live/amd64/iso-hybrid/.

[16] phpipam.net. *Phpipam Installation Guide*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://phpipam.net/documents/installation/.

[17] mariadb.com. *Where to Download MariaDB*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://mariadb.com/kb/en/getting-installing-and-upgrading-mariadb/.

[18] Justin Ellingwood, Brian Boucheron a Hanif Jetha. *How To Install the Latest MySQL on Debian 10*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://www.digitalocean.com/community/tutorials/how-to-install-the-latest-mysql-on-debian-10.

[19] MariaDB. *Setting up a Replication Slave with Mariabackup*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://mariadb.com/kb/en/setting-up-replication/.

[20] powerdns.com. *PowerDNS Authoritative Nameserver*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://doc.powerdns.com/authoritative/index.html.

[21] powerdns.com. *PowerDNS Recursor*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://docs.powerdns.com/recursor/indexTOC.html.

[22] Whalebone. *4 crucial reasons why ISPs should separate their authoritative and recursive DNS servers*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.whalebone.io/separate-dns-servers/.

[23] powerdns.com. *Migrating from using recursion on the Authoritative Server to using a Recursor*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://doc.powerdns.com/authoritative/guides/recursion.html.

[24] Khanh Ngo. *Home*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://github.com/ngoduykhanh/PowerDNS-Admin/wiki.

[25] Hitesh Jethva. *How to Install PowerDNS Server and PowerDNS Admin on Ubuntu 20.04*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.howtoforge.com/how-to-install-powerdns-admin-on-ubuntu-20-04/.

[26] Josphat Mutai. *Install PowerDNS and PowerDNS-Admin on Ubuntu 18.04 / Debian 9 with MariaDB Backend*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://computingforgeeks.com/install-powerdns-and-powerdns-admin-on-ubuntu-18-04-debian-9-mariadb-backend/.

[27] koromicha. *Easily Install and Setup PowerDNS Admin on Ubuntu 20.04*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://kifarunix.com/easily-install-and-setup-powerdns-admin-on-ubuntu-20-04/.

[28] Patrik Cevela. *Running PowerDNS Admin with Systemd, Gunicorn and Nginx*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://github.com/ngoduykhanh/PowerDNS-Admin/wiki/Running-PowerDNS-Admin-with-Systemd,-Gunicorn--and--Nginx.

[29] Josphat Mutai. *Install and Configure phpIPAM on Ubuntu 20.04/18.04 / Debian 10 Linux*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://computingforgeeks.com/install-and-configure-phpipam-on-ubuntu-debian-linux/.

[30] Vicky Risk, Suzanne Goldlust a Michal Nowikowski. *ISC Packages for Kea DHCP*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://kb.isc.org/docs/isc-kea-packages.

[31] Internet Systems Consortium. *Kea Administrator Reference Manual*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://kea.readthedocs.io/en/kea-1.6.2/index.html.

[32] Eddy Winstead. *Getting Started with Kea*. [online]. Naposledy navštíveno: 14.8.2020. Dostupné z: https://www.youtube.com/watch?v=446Ew0POkTw.

[33] Internet Systems Consortium. *Stork Administrator Reference Manual*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://kea.readthedocs.io/projects/Stork/en/v0.14.0/index.html.

[34] Robert Allen. *Top 16 DHCP Best Practices: The Ultimate Guide*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://activedirectorypro.com/dhcp-best-practices/.

[35] Tomek Mrugalski. *Migrating from ISC DHCP to Kea DHCP*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://blog.apnic.net/2020/03/06/how-to-getting-started-with-kea-dhcp-for-ipv4-ipv6/.

[36] Neally. *Server Migration Best Practices*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://community.spiceworks.com/topic/1476001-server-migration-best-practices.

[37] microsoft.com. *DHCP Server Migration: Preparing to Migrate*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/dn495423(v=ws.11).

[38] Ram Mohan. *DNS Migration: How To Minimize Problems When Switching DNS Providers*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.securityweek.com/dns-migration-how-minimize-problems-when-switching-dns-providers.

[39] Jeff Edwards. *Managing Network Configuration Changes: Five Best Practices*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.whatsupgold.com/blog/best-practices-in-network-configuration-and-change-management.

[40] Marcin Siodelski Tomasz Mrugalski. *DHCP Performance Guide*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://users.isc.org/~tomasz/perfdhcp/dhcp-perf-guide.html.

[41] Prowse Tech. *Kea DHCP4 Basic Performance Testing*. [online]. Naposledy navštíveno: 28.12.2020. Dostupné z: https://www.youtube.com/watch?v=IW3eXTM9skc.

Příloha A

Seznam zkratek

AXFR	Asynchronous Zone Transfer Protocol
ČVUT	České vysoké učení technické
DHCP	Dynamic Host Configuration Protocol
DoS	Denial of Service
DNS	Domain Name System
HA	High Availability
IPAM	IP Address Management
ISC	Internet Systems Consortium
RFC	Request for Comments
SPOF	Single Point of Failure