

Mendelova univerzita v Brně
Provozně ekonomická fakulta

Migrace laboratorního firewallu z platformy Linux PC na platformu MikroTik

Bakalářská práce

Vedoucí práce:
Ing. Martin Pokorný, Ph.D.

Roman Šilhan

Brno 2015

Tímto chci poděkovat vedoucímu této bakalářské práce Ing. Martinu Pokornému, Ph.D. za odborné vedení, trpělivost a užitečné rady. Dále chci poděkovat Bc. Michalovi Šturmovi za pomoc při práci v síťové laboratoři. V neposlední řadě chci také poděkovat mé rodině za podporu nejen při zpracování této práce, ale i během celého studia.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Migrace laboratorního firewallu z platformy Linux PC na platformu MikroTik**

vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*.

Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.

Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

V Brně 16. května 2016

.....

Abstract

ŠILHAN, R. *Migration of laboratory firewall from Linux PC platform to MikroTik platform*. Bachelor thesis. Brno. 2016.

This bachelor's thesis deals with migration of laboratory firewall from Linux PC platform to MikroTik platform. Thesis includes analysis of the current solution and a proposal of a new solution. The new solution is subsequently implemented and tested at the Laboratory of Computer Networking of FBE MENDELU. This thesis also includes scripts for external interface management of the firewall and economic evaluation of the proposal.

Keywords

stateful firewall, firewall migration, MikroTik, iptables

Abstrakt

ŠILHAN, R. *Migrace laboratorního firewallu z platformy Linux PC na platformu MikroTik*. Bakalářská práce. Brno. 2016.

Tato bakalářská práce se zabývá migrací laboratorního firewallu z platformy Linux PC na platformu MikroTik. Práce zahrnuje analýzu stávajícího řešení a návrh nového řešení. Nové řešení je následně implementováno a testováno v Sítové laboratoři PEF MENDELU. Součástí práce jsou i skripty pro správu vnějšího rozhraní firewallu a ekonomické zhodnocení návrhu.

Klíčová slova

stavový firewall, migrace firewallu, MikroTik, iptables

Obsah

1	Úvod a cíl práce	9
1.1	Úvod práce	9
1.2	Cíl práce	9
2	Literární řešerše	10
2.1	Závěrečné práce	10
2.2	Odborné články	11
2.3	Studijní zdroje	12
2.4	Závěrečné zhodnocení	13
3	Analýza uživatelských požadavků	14
3.1	Migrace pravidel	14
	Zajištění stávajících firewallových pravidel	14
	Uživatelské řetězce	14
	Ochranné mechanismy	14
3.2	Výměna linuxového labrouteru za router od firmy MikroTik	14
3.3	Skripty pro správu	15
4	Popis technologického aparátu	16
4.1	Sada protokolů TCP/IP	16
	Aplikační vrstva	16
	Transportní vrstva	16
	Síťová vrstva	17
	Vrstva síťového rozhraní	17
4.2	Iptables	17
4.3	RouterBOARD a RouterOS od firmy MikroTik	19
4.4	VLAN	19
4.5	Bezpečnostní politika	19
4.6	Skript	19
4.7	Protokoly	20
4.8	Protokoly pracující na aplikační vrstvě TCP/IP modelu	20
5	Analýza současného stavu	21
5.1	Topologie laboratorní sítě ÚI PEF MENDELU	21
	VLAN 99	21
	VLAN 30	22
	VLAN 50	22
	VLAN 10 a řešení výukových stanic	22
5.2	Adresace na L3	23
5.3	Fyzické prvky v laboratorní síti ÚI PEF MENDELU	23
5.4	Firewallová politika	24
	Řetězec INPUT	24

	Řetězec FORWARD	26
	Řetězec OUTPUT	28
5.5	Komunikace v laboratorní síti ÚI PEF MENDELU	29
	Komunikace mezi VLAN 30 a VLAN 99	29
	Komunikace mezi VLAN 10 a VLAN 30	31
	Komunikace mezi VLAN 10 a VLAN 99	33
	Komunikace mezi VLAN 10 a Internetem	34
	Komunikace mezi VLAN 30 a Internetem	35
	Komunikace mezi VLAN 50 a Internetem	36
	Komunikace mezi VLAN 99 a Internetem	37
	Řetězec SpatnePakety	38
	Řetězec RFC1918_IN	38
	Řetězec IpAdresy	39
	Řetězce LogDrop	39
5.6	Závěr analýzy	39
6	Návrh řešení	40
6.1	Fyzické zapojení	40
6.2	Základní nastavení labrouteru MikroTik	40
	Název zařízení	40
	Přístupová práva	40
	Synchronizace času	42
	Odesílání logovacích záznamů	42
6.3	Nastavení rozhraní	42
	Vytvoření VLAN	42
	Přidělení IP adres	43
6.4	Směrovací tabulka	43
6.5	DHCP Relay	44
6.6	Návrh nového firewallu	44
	Způsob filtrace paketů	44
	Řazení nových pravidel	45
	Redukce pravidel	45
	Další ochranné mechanismy	46
	Skenování portů	46
	Útok záplavou paketů	47
	Další doporučené ochranné mechanismy	48
	NAT	48
	Způsob migrace	49
	Zavedení firewallu	49
6.7	Skripty pro administraci vnějšího rozhraní firewallu	49
	Způsob spouštění	49
	Algoritmy skriptů	50

7	Řešení	53
7.1	Fyzické zapojení	53
7.2	Základní nastavení labrouteru MikroTik	53
	Název zařízení	53
	Přístupová práva	54
	Synchronizace času	55
	Odesílání logovacích záznamů	55
7.3	Nastavení rozhraní	56
	Vytvoření VLAN	56
	Přidělení IP adres	56
7.4	Směrovací tabulka	57
7.5	DHCP Relay	58
7.6	Firewallová pravidla	58
	Přidané ochranné mechanismy	61
	NAT	62
	Zavedení firewallu	62
7.7	Skripty pro administraci vnějšího rozhraní firewallu	63
	Skript pro povolení/zakázání vnějšího rozhraní	63
	Skript pro zjištění stavu vnějšího rozhraní	64
8	Testování	65
8.1	Vzájemná komunikace	65
8.2	Dostupnost služeb z VLAN 30	66
8.3	Dostupnost služeb z Internetu pro vnitřní síť	67
8.4	Zátěžový test	67
8.5	Testování ochranných mechanismů	68
8.6	Testování skriptů	68
9	Ekonomické zhodnocení	70
10	Závěr	71
10.1	Nedostatky	71
11	Seznam literatury	72
	Přílohy	75
A	Detail fyzického zapojení	76
B	Konfigurace labrouteru MikroTik	77
C	Skript pro povolení vnějšího rozhraní	78
D	Skript pro zakázání vnějšího rozhraní	79

OBSAH	8
E Skript pro zjištění stavu vnějšího rozhraní	80
F Skript pro zavedení firewallových pravidel	81

1 Úvod a cíl práce

1.1 Úvod práce

S rozšířením Internetu nastal problém, jak ochránit vnitřní síť organizací proti nebezpečí přicházející z Internetu. Denně se mnoho hackerů a crackerů snaží proniknout do vnitřních sítí organizací a získat citlivá data. Je nutné mít nástroj, který umožní administrátorům obranu proti takovýmto útokům.

Jedním z řešení tohoto problému je použití firewallu. Firewall se nachází mezi Internetem a vnitřní sítí. Veškerá komunikace prochází skrze něj, a to mu umožňuje filtrovat procházející komunikaci na povolenou a zakázanou.

Tato práce se bude zabývat nahrazením linuxového firewallu v síťové laboratoři ÚI PEF MENDELU za firewall na platformě MikroTik, jehož úkolem bude filtrovat komunikaci nejen mezi vnitřní sítí laboratoře a Internetem, ale i mezi jednotlivými virtuálními sítěmi, do kterých jsou uzly v laboratoři rozděleny.

Síťová laboratoř slouží k výuce předmětů Operační systémy, Počítačové sítě, Inovace pro informatiky a Bezpečnost informačních systémů. Laboratoř dále poskytuje technické zázemí pro zpracování závěrečných prací z oblasti síťových technologií. Mimo výše uvedené slouží laboratoř k výuce síťových akademií. Více o síťové laboratoři a síťových akademiích se lze dočíst na webových stránkách Ústavu informatiky (2013).

Důvodem pro nahrazení současného linuxového firewallu za platformu MikroTik je snížení nákladů na provoz, neboť MikroTik je malý směrovač s minimálními požadavky na odběr elektrické energie.

1.2 Cíl práce

Cílem práce je migrace firewallu síťové laboratoře ÚI PEF MENDELU z platformy Linux na platformu MikroTik, která bude zahrnovat analýzu stávajícího firewallu, návrh nového řešení, implementaci řešení v laboratoři a vytvoření skriptů pro vzdálenou administraci vnějšího síťového rozhraní MikroTik firewallu přes komunikační protokol SSH.

2 Literární rešerše

Literární rešerše je zpracována za následujících kritérií:

- Klíčová slova: MikroTik, Linux, router, RouterOS, firewall, firewall migration, směrování
- Časové rozmezí 2010–2015, výjimkou je odborný článek Miroslava Petříčka (2001), který sice nesplňuje časové kritérium, tento článek je ovšem stále platný a aktuální.

2.1 Závěrečné práce

Haresta (2012) se ve své diplomové práci zabývá zpracováním návrhu a modernizací stávající LAN sítě ve středně velké firmě. V teoretické části Haresta (2012) vysvětluje informace potřebné k vytvoření počítačové sítě. V praktické části, se Haresta (2012) zabývá analýzou stávající sítě a dále návrhem a konfigurací sítě nové. Z práce se lze inspirovat postupem při tvorbě počítačové sítě.

Havlíček (2013) v první části bakalářské práce seznamuje čtenáře s operačními systémy Linux. Dále v teoretické části Havlíček (2013) vysvětluje problematiku zabezpečení operačního systému Linux. V praktické části se Havlíček (2013) zaměřuje na postupy, jak operační systém ochránit a podává čtenáři praktická doporučení v oblasti využívání linuxových distribucí.

Knytl (2014) v bakalářské práci popisuje simulování síťových útoků na jednotlivých vrstvách ISO/OSI modelu. Knytlem (2014) je také popsán postup prováděných simulací, specifikace prostředí, ve kterém byly simulace prováděny, a poskytuje doporučení, jak se takovými útokům bránit. Práce může být užitečná při sestavování testovacího postupu.

Bakalářská práce (Kostelník, 2011) je zaměřena na vytvoření kompletní počítačové sítě pro Dům kultury Vsetín, spol. s r. o. Čtenáře provází celým procesem tvorby podnikové sítě od analýzy přes návrh infrastruktury až po kompletní realizaci nové počítačové sítě. Práce je věnována mimo jiné i vytvoření bezpečnostních řešení a implementaci požadovaných služeb.

Závěrečná práce (Krajča, 2011) řeší problematiku bezpečnosti v síti internet z pohledu ISP. V teoretické části Krajča (2011) popisuje hrozby, které se mohou na internetu vyskytovat, a způsoby jak se těmto hrozbám bránit. V praktické části, Krajča (2011) popisuje konfiguraci firewallu na platformě MikroTik (RouterOS), způsob ochrany koncových uživatelů a ochranu síťového provozu.

Krajča (2013) ve své diplomové práci seznamuje čtenáře se zabezpečením sítě poskytovatele internetových služeb z pohledu administrátora. V práci jsou vysvětleny možná rizika napadení počítačových sítí. Krajča (2013) popisuje konfiguraci základních pravidel firewall v operačním systému RouterOS. V závěru práce je popsán systém testování funkčnosti a účinnosti bezpečnostních funkcí. Práce může sloužit jako návod pro testování funkčnosti bezpečnostních opatření.

Petrák (2013) v diplomové práci navrhuje a vytváří aplikaci v jazyku C++, umožňující filtrovat pakety na druhé a třetí vrstvě ISO/OSI modelu.

Diplomová práce Rypiena (2012) se zaměřuje na realizaci síťového zabezpečení v konkrétní firmě. K zabezpečení je použit linuxový server, který plní funkci firewallu.

Bakalářská práce Tvorba routeru na bázi Linuxu (Suda, 2011) popisuje tvorbu směrovače na 64b linuxové distribuci Debian 6. V práci je vysvětlena problematika směrování a konfigurace routeru. Suda (2011) píše mimo jiné o konfiguraci základních služeb routeru např. vytvoření DHCP serveru, DNS serveru a stavového firewallu. Pomocí firewallu je dále prováděný překlad adres NAT. Závěr bakalářské práce je věnován testování sítě z hlediska rychlosti a propustnosti.

Váňa (2012) ve své práci porovnává různé linuxové distribuce, které mají vyhovovat podnikové prodejně HP Tronic. Práce je zaměřená především na vyhledání optimálního řešení, které sníží náklady na minimum.

Vohník (2013) v závěrečné práci řeší problematiku centrální administrace linuxového firewallu. V sekci Vlastní řešení Vohník (2013) uvádí řešení své práce, spočívající především v návrhu a implementaci aplikace, která vytváří prostředí pro centrální správu sítě.

Hvizdák (2013) ve své diplomové práci popisuje tvorbu firemního firewallu s poštovním serverem na platformě Linux. V teoretické části Hvizdák (2013) vysvětluje náležitosti potřebné k vytvoření poštovního serveru. V praktické části je podrobně popsán proces tvorby poštovního serveru včetně konfigurace firewallu (Iptables).

Práce Generátor základních filtrovacích pravidel pro konfiguraci firewallů na síťových zařízeních (Vyhnátek, 2013) je věnovaná tvorbě mobilní aplikace na platformu Android. Vyhnátek (2013) popisuje analýzu, návrh i implementaci celé aplikace. Aplikace je schopná pomocí GUI prostředí usnadnit tvorbu základních filtrovacích pravidel a to na směrovače od firmy Cisco Systems, Inc, MikroTik i na servery využívající linuxové distribuce. Součástí aplikace je i SSH přístup, který zajišťuje, že jednotlivá pravidla je mobilní zařízení schopno přímo nakonfigurovat do routeru.

2.2 Odborné články

Článek 8 Steps for a Successful Firewall Migration (Besana, 2013) popisuje, 8 kroků potřebných pro úspěšnou migraci firewallu. Besana (2013) v článku rozdělil migraci do 8 etap:

1. Seznámení se s novou technologií – Besana (2013) uvádí, že prvním krokem je seznámit se s novou technologií a jejími funkcemi
2. Prozkoumání stávajícího firewallu – zjistit co a jak funguje ve stávajícím firewallu, zjistit, která pravidla jsou již nepotřebná nebo která naopak chybí

3. Simulace překladu konfigurace – Besana (2013) radí, abychom se na základě ob-
sáhlosti stávajícího firewallu rozhodli, zda stačí pravidla přepsat nebo je vhodné
využít nějaký software pro automatický překlad
4. Testy přípustnosti – naplánování a provedení testů
5. Deklarování zamrzlé zóny – časový úsek, během kterého bychom se měli vyva-
rovat jakýchkoli změn ve stávajícím firewallu
6. Překlad konfigurace – pečlivě zavést pravidla do nového firewallu
7. Migrace – vlastní migrace firewallu, by měla, jak Besana (2013) uvádí, proběh-
nout v čase, kdy je provoz v síti minimální
8. Monitorování – monitorování a údržba firewallu

Stavíme firewall (Petříček, 2001) je série článků, které jsou sice staršího data, ale jejich obsah je stále platný. Petříček (2001) v třídílném seriálu popisuje tvorbu firewallu (Iptables) na platformě Linux. Čtenář je seznámen s problematikou Iptables a je mu vysvětlena základní funkčnost. V prvním díle Petříček (2001) popisuje, jak funguje ověřování paketů bez stavového firewallu. Druhý díl je věnován problematice překladu adres NAT. V posledním díle je čtenář seznámen s funkčností stavového firewallu.

Štrauch (2012) v článku popisuje všechny aspekty tvorby firewallu v operačním systému RouterOS. Začátek článku je věnován problematice toku paketů. V kapitole jsou popsány tři základní řetězce (forward, input, output) a je vysvětleno k čemu slouží. Dále se článek věnuje tvorbě pravidel a popisu práce se stavovým firewallem. Štrauch (2012) pro ukázkou také znázorňuje vzorový příklad. Zbytek článku je věno-
vaný překladu adres pomocí DNAT (cílová NAT) a SNAT (zdrojová NAT).

Článek MikroTik: skriptování (Štrauch, 2012) v RouterOS obsahuje cenné rady pro používání skriptů v operačním systému RouterOS. Štrauch (2012) na začátku článku porovnává rozdíly skriptování na platformě Linux a RouterOS. Z obsahu lze vyčíst, že princip je podobný, ale RouterOS má několik omezení, na která je třeba dávat pozor. Dále lze v článku najít bližší popis jednotlivých příkazů, než je uveden na oficiálním webu. Štrauch (2012) popisuje práci s proměnnými, podmínkami, cykly, práci se stromovou strukturou a spuštění skriptu.

Správa linuxového serveru: Linuxový firewall, základy Iptables (Dočekal, 2010) je obdobně jako Stavíme firewall (Petříček, 2001) seriál z oblasti tvorby firewallových pravidel. Dočekal (2010) na rozdíl od Petříčka (2001) navíc seznamuje čtenářem i s Iptables pro IPv6.

2.3 Studijní zdroje

Oficiální dokumentace vytvořená pro nástroj Iptables od Russella (2002). Iptables umožňuje filtrovat pakety na platformě Linux. Dokumentace obsahuje syntaxi pří-
kazu a popisy přepínačů.

Manuál od výrobce MikroTik směrovačů (MikroTik, 2014) popisuje všechny potřebné prostředky pro vytvoření firewallu na platformě MikroTik.

Iptables od Meitnera (2012) je překlad oficiální komunitní dokumentace vysvětlující práci s Iptables. Meitner (2012) uvedl příkazy, kterými se utilita Iptables ovládá. Příkazy jsou doplněny českým popisem a nechybí ani příklady použití těchto příkazů.

2.4 Závěrečné zhodnocení

V dnešní době je zabezpečení počítačových sítí prioritou každé instituce. Lze tedy najít mnoho zdrojů, které toto téma řeší. Problematika firewallů a zabezpečení počítačů se proto stala častým tématem závěrečných prací. Mnoho z nich se věnuje zejména návrhu popřípadě i implementaci firewallové politiky v podnicích s využitím Iptables na Linuxových serverech.

Je třeba si ale uvědomit, že implementace firewallu je závislá na konkrétní topologii, bezpečnostní politice i konkrétních požadavcích zadavatele. To znamená, že implementace firewallu se od sebe liší. Návrh a implementace firewallu pro laboratorní síť ÚI PEF Mendelu je jedinečná a je zde tedy prostor pro vykonání bakalářské práce. Mimo to nebyly při zpracování literární rešerše odhaleny žádné oficiální návody, jak migrovat firewall z linuxové distribuce na platformu MikroTik.

3 Analýza uživatelských požadavků

V této kapitole budou zanalyzovány a popsány uživatelské požadavky, které jsou nezbytné pro splnění zadání bakalářské práce.

3.1 Migrace pravidel

Zadáním bakalářské práce je migrovat stávající laboratorní firewall z linuxové platformy na platformu MikroTik.

Migrace bude zahrnovat následující činnosti:

- analýzu stávajících firewallových pravidel,
- analýzu rozdělení do uživatelských řetězců,
- upravení a doplnění firewallových pravidel na základě analýzy do nového firewallu,
- přidání ochranných mechanismů,
- nasazení firewallu do nového labrouteru.

Zajištění stávajících firewallových pravidel

Výpis pravidel ze stávajícího firewallu bude poskytnut po podpisu prohlášení o mlčenlivosti. Výpis pravidel bude sloužit pouze k analýze stávajícího firewallu, která je nutná ke splnění bakalářské práce.

Uživatelské řetězce

Požadavek na zachování filtrování paketů do uživatelských řetězců je nutné dodržet zejména kvůli snadné orientaci ve firewallových pravidlech. Dělení do uživatelských řetězců navíc zvyšuje rychlost filtrování paketů.

Ochranné mechanismy

Nové řešení bude zahrnovat přidání ochranných mechanismů, které budou chránit firewall a vnitřní síť proti útokům záplavou paketů a skenování portů.

3.2 Výměna linuxového labrouteru za router od firmy MikroTik

Nový labrouter má zajišťovat nejen funkci firewallu, ale i funkci směrovače. Proto bude nutné analyzovat konfiguraci stávajícího linuxového labrouteru a migrovat tuto konfiguraci na platformu MikroTik. Je nutné migrovat nastavení rozhraní, IP adresaci a směrovací tabulku.

3.3 Skripty pro správu

Pro vzdálenou správu nového labrouteru je nutné vytvořit skripty, které budou schopny vzdáleně povolovat a zakazovat vnější rozhraní labrouteru spojující síťovou laboratoř s Internetem.

Další důležitý skript, bude sloužit ke zjištění stavu vnějšího rozhraní. Skript bude administrátorovi vypisovat, zda je připojení k Internetu povolené nebo zakázané.

4 Popis technologického aparátu

V této kapitole budou shrnuty všechny teoretické znalosti nutné k vypracování této práce.

4.1 Sada protokolů TCP/IP

Síťové protokoly definují komunikační pravidla, která řídí komunikaci v počítačové síti. Horák a Keršláger (2006) uvádějí, že pro funkční síť je nutné, aby všechny síťové stanice používaly stejné protokoly.

Základním modelem, který udává, jak budou síťové prvky komunikovat v počítačové síti, je referenční model ISO/OSI. Tento model slouží zejména pro teoretické účely. V praxi se využívá sada protokolů TCP/IP. Modely využívají vrstvenou architekturu. V modelu TCP/IP jsou jasně definovány 4 vrstvy: vrstva síťového rozhraní, síťová vrstva, transportní vrstva a vrstva aplikační.

Aplikační vrstva

Aplikační vrstva obstarává komunikaci mezi procesy. Procesy jsou vlastně běžící programy na počítačích. V případě síťové laboratoře ÚI PEF MENDELU využívají klienti služeb serveru. Server je hostitel, který umožňuje přijímat od klientů požadavky a poskytovat jim služby. Pokud tedy klient potřebuje od serveru nějakou službu, naváže s ním komunikaci přes procesy. Procesy si mezi sebou vyměňují zprávy. Aby mohl klient odeslat požadavek na určitou službu, musí znát IP adresu hostitele a port procesu, se kterým chce komunikovat. IP adresa společně s portem tvoří socket.

Aby byla zaručená komunikace nejpoužívanějších procesů, vydala organizace IANA standard RFC 6335, ve kterém definuje porty 1–1023 jako well-known, porty 1024–49151 jsou registrované prodejci a porty 49152–65535 pro osobní použití.

Transportní vrstva

Transportní vrstva zajišťuje, jak uvádí Kurose a Ross (2014), přenos zprávy aplikační vrstvy mezi koncovými aplikacemi. Na transportní vrstvě jsou prakticky nejpoužívanější protokoly TCP a UDP.

Protokol UDP je nespojovaný přenos, u kterého nelze zaručit, že paket dorazí na místo určení. Naproti tomu protokol TCP je spojovaný, protože než dojde k přenosu paketu, musí být proveden tzv. handshake mezi účastníky spojení. Transportní vrstva převezme zprávu od aplikační vrstvy a zapouzdří k ní TCP nebo UDP záhlaví. Při spojově orientovanému přenosu může TCP protokol zaručit správné doručení paketu, protože vytvoří virtuální komunikační kanál. Protokol TCP umí také segmentovat zprávy do menších celků, a díky tomu umí poskládat zprávu na přijímaném zařízení i za předpokladu, že segmenty nepřišly sekvenčně.

Protokol TCP používá k vytvoření virtuálního kanálu flagy popsané ve standardu RFC 793. Jak již bylo uvedeno dříve, začátek komunikace je zahájen handshakem. Handshake je vlastně dorozumění se pomocí tří zpráv, které se ohlásí flagy v následujícím pořadí. SYN M slouží k vytvoření spojení, kde M představuje počáteční sekvenční číslo. SYN M posílá klient a na to odpovídá server zprávou s flagy SYN N a ACK M+1, což je odpověď serveru, kde SYN N je flag s počátečním sekvenčním číslem serveru a M+1 číslo sekvence, které je nyní očekáváno od klienta. Poslední zpráva ACK N+1 je odpověď klienta s očekávaným sekvenčním číslem serveru. Nyní je stav spojení established – tedy ustanovené spojení. Podobně vypadá i ukončení spojení, které používá flag FIN. O TCP se lze více dozvědět v knize od Kurose a Rosse (2014) nebo Horáka a Keršlágra (2006).

Síťová vrstva

Kurose a Ross (2014) uvádějí, že síťová vrstva má na starost přenos paketů známé jako datagramy. Datagramy vznikají zapouzdřením segmentu z transportní vrstvy IP protokolem, který umožňuje směrování na 3. vrstvě ISO/OSI. Dnes existují dva druhy IP adresy. Verze 4 je starší 32b číselná adresa s rozsahem 2^{32} tedy přes 4 miliardy adres. IPv4 stále převažuje nad novější verzí 6, přestože má oproti IPv6 menší rozsah adres. IPv6 je 128b adresa s rozsahem 2^{128} adres.

Důvod proč je IPv4 stále oblíbená je maskování neveřejných (privátních) adres za adresu veřejnou. Veřejná IP adresa se totiž musí vyskytovat v síti nejvýše jednou. Proto byl zaveden standard RFC 1918, který určuje, jaké adresy se nesmí používat veřejně.

Vrstva síťového rozhraní

Vrstva síťového rozhraní, též známá jako vrstva linková, slouží k přenosu rámců po jedné lince. Kurose a Ross (2014) uvádějí, že linková vrstva je implementována v síťové kartě (NIC). Adresování na linkové vrstvě se provádí na základě MAC adres, což je číslo síťové karty.

Aby bylo možné přenášet rámce mezi uzly, které jsou adresované pomocí IP adresy, využívá vrstva síťového rozhraní ARP protokol. Tento protokol zajišťuje překlad MAC adres na IP adresy. Pokud chce nějaký uzel komunikovat a nemá ve své ARP tabulce IP adresu cílového uzlu, odešle zprávu ARP Request na broadcastovou MAC adresu. Ve zprávě se ptá, kdo má příslušnou IP, kterou hledá. Pokud ji někdo vlastní, odešle zpět uzlu, který se ptal, ARP zprávu ARP Reply. Uzel si potom zapíše IP a MAC adresu do své ARP tabulky, ve které později primárně vyhledává.

4.2 Iptables

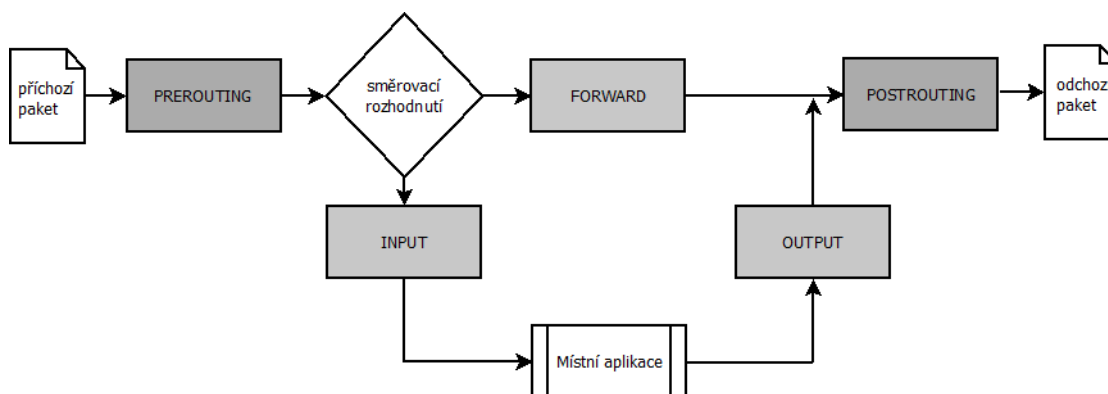
Původní firewall s operačním systémem Linux spravoval firewall pomocí programu Netfilter a jeho nástrojem Iptables. Netfilter není, jak informuje Dočekal (2011), pouze firewall, ale paketový filtr, který umožňuje provádět mnoho akcí. Jednou z těchto

akcí je právě tvorba filtrů, které vykonávají funkci firewallu. Iptables pracují tak, že příchozí paket prochází řetězcem pravidel, dokud některému nevyhoví.

Filtrování paketů probíhá na základě mnoha různých parametrů. Pakety lze například, jak uvádí Russell (2002), filtrovat na základě rozhraní, zdrojové MAC adresy, zdrojové a cílové IP adresy, zdrojového a cílového portu, ale například i na základě limitu, který určuje počet shod paketu s konkrétním pravidlem.

V Iptables jsou tři hlavní tabulky: filter, nat, mangle. Tabulka filter je základní tabulkou v Iptables. Je využívána pro nastavení politiky průchodu do, přes a z počítače. Russell (2002) říká, že tabulka filter je postavena na třech základních řetězcích: INPUT, OUTPUT, FORWARD. Russell (2002) dále uvádí, že nat je tabulka používaná k přesměrování spojení na základě IP adres. Tabulka používá tři řetězce: PREROUTING, POSTROUTING a OUTPUT. Třetí tabulka mangle je určena pro změny paketů a používá všechny již dříve uvedené řetězce.

Pakety prochází jednotlivými řetězci a jsou zde vyhodnocovány na základě pravidel. Schéma průchodu paketů paketovým filtrem je znázorněno na Obr. 1 pomocí Dočekalova (2011) diagramu.



Obrázek 1: Schéma průchodu paketu paketovým filtrem (Dočekal, 2011)

Pravidla Iptables mohou paket vyhodnotit a provést s ním jednu ze čtyř hlavních akcí: ACCEPT – paket je přijat, DROP – paket je zahozen, REJECT – paket je vrácen odesílateli nebo LOG – zápis do logovacího souboru.

Iptables umí filtrovat pakety i na základě jejich stavu. Paket může nabývat jednoho ze čtyř stavů: NEW – paket vytváří nové spojení, ESTABLISHED – paket patří k ustanovenému spojení, RELATED – paket vytváří nové spojení, které ale patří již k některému z existujících spojení, INVALID – paket nevztahující se k žádnému spojení. Všechny výše uvedené informace slouží k úspěšnému vytvoření filtrovacích pravidel na platformě Linux. Pro podrobnější informace se lze informovat v článku Dočekala (2011) nebo Petříčka (2001), kteří ve svých článcích nabízí mimo jiné i příklady jednoduchých pravidel. Dalším zdrojem je oficiální dokumentace Netfilter od Russella (2002).

4.3 RouterBOARD a RouterOS od firmy MikroTik

Jak je uvedeno v cíli, migrace firewallu je prováděna z platformy Linux na platformu MikroTik. Nyní se v síťové laboratoři ÚI PEF MENDELU nachází linuxový server sloužící jako router a firewall, který bude nahrazen směrovačem RouterBOARD od firmy MikroTik.

Firma MikroTik nabízí směrovače ve dvou variantách: integrovaný – kompletně složený router v krytu nebo ve formě základní desky, které jsou, jak říká Discher (2011), připraveny akceptovat bezdrátové síťové karty a lze je zakrýt libovolným krytem.

V tomto případě bude použit integrovaný směrovač RB951-2n s operačním systémem RouterOS, což je systém založený na linuxové technologii. V operačním systému RouterOS lze obdobně jako u Linuxu využít nástroj terminál, a ovládat tak RouterOS pomocí příkazů. Mnohem pohodlnější je ale využít program Winbox, který je MikroTikem (2015) definován jako malá služba, která umožňuje správu MikroTik RouterOS přes jednoduché grafické uživatelské rozhraní. K RouterBOARDU se lze připojit přes IP nebo MAC adresu, což je užitečné zejména v případě, že fyzické rozhraní, do kterého je nutné se připojit, zatím nemá nastavenou IP adresu.

Firewallový filtr implementovaný v rámci RouterOS pracuje na velmi podobném principu jako paketový filtr Iptables na Linuxu. Filtry se liší zejména v syntaxi příkazů.

4.4 VLAN

Ke splnění bakalářské práce je nutné znát také technologii VLAN. Zkratku VLAN vysvětluje Kurose a Ross (2014) jako virtuální lokální síť, která slouží především k logickému oddělení podsítí. VLAN je identifikována pomocí VLAN ID, což je číslo od 0 do 4095. Pokud je port připojen na linku spojující síť v rámci jedné VLAN, takový port je označen jako přístupový port. Přístupový port umožňuje přenášet netagované pakety.

Pokud se mají po lince posílat pakety z více různých VLAN, je třeba nastavit port pro přenášení tagovaných paketů. Více o VLAN technologii lze najít například v knize od Kurose a Rosse (2014).

4.5 Bezpečnostní politika

Jak uvádějí Barker a Morris (2013), bezpečnostní politika je tvořena vedením společnosti a určuje, jakými technologiemi bude chráněna síť. Je podle nich také nutné vyvážit hodnotu chráněných informací a náklady na jejich ochranu.

4.6 Skript

Součástí bakalářské práce je i vytvoření skriptů, přes které bude správce labrouteru moci zakazovat a povolovat vnější rozhraní na směrovači. MikroTik má defaultně

spuštěnou službu SSH, přes kterou se lze připojit ke směrovači a spustit naprogramovaný příkaz. Při spojení SSH protokolem je vyžadováno přihlašovací jméno a heslo. To lze buď zadat až při výzvě nebo při uložení veřejného klíče.

4.7 Protokoly

- IP (Internet Protokol) slouží k adresaci datagramů na 3. vrstvě ISO/OSI.
- ICMP (Internet Control Message Protocol) slouží k odeslání chybových zpráv po síti. Vykonává svoji funkci na síťové vrstvě ISO/OSI modelu.
- TCP (Transmission Control Protocol) je protokol pro vytváření spolehlivého spojení na 4. vrstvě referenčního modelu ISO/OSI.
- UDP (User Datagram Protocol) vytvoří nespolehlivého spojení, které je ale oproti TCP rychlejší. Pracuje na transportní vrstvě ISO/OSI.

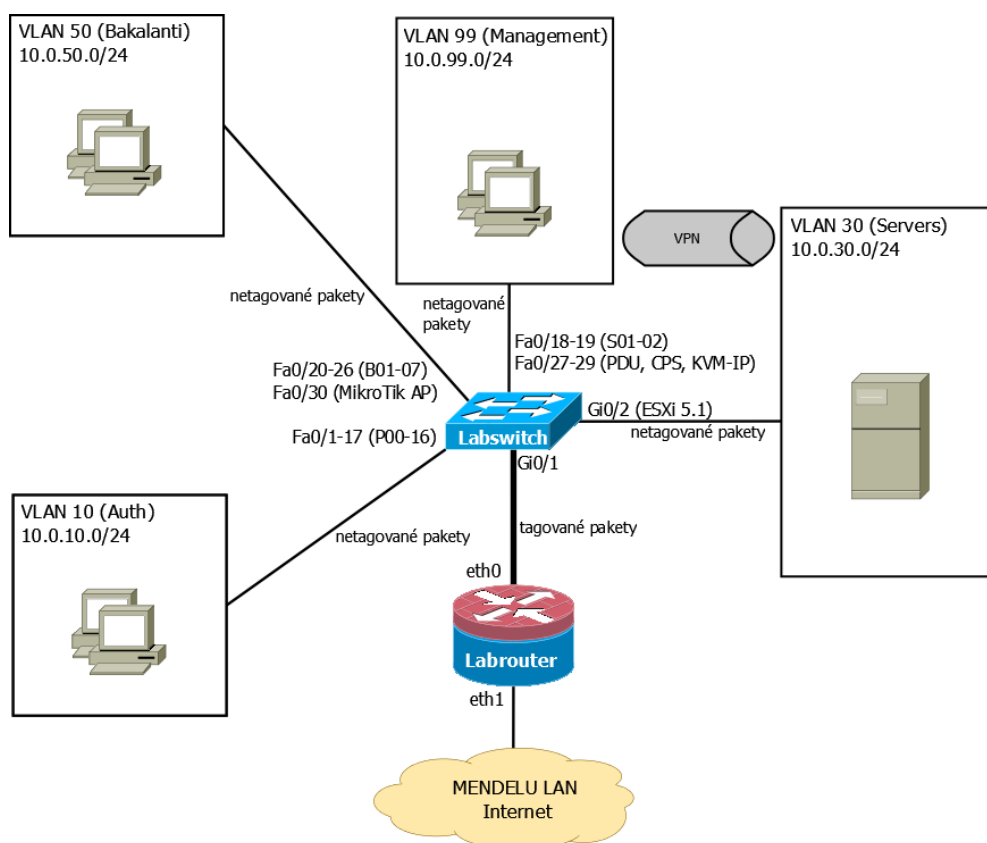
4.8 Protokoly pracující na aplikační vrstvě TCP/IP modelu

- FTP (File Transfer Protokol) slouží k přenosu souborů mezi počítači. FTP využívá dva well-known porty: 20/tcp – pro přenos vlastních dat, 21/tcp – pro přenos řídicích příkazů.
- SSH (Secure Shell) zajišťuje šifrovanou komunikaci mezi dvěma počítači. Používá well-known port 22/tcp.
- DNS (Domain Name Server) je protokol překládající doménová jména na IP adresy a naopak. Využívá well-known port 53/udp nebo 53/tcp.
- DHCP (Dynamic Host Configuration Protocol) je zodpovědný za přidělení IP adresy počítačům v síti. Server operuje s tímto protokolem na standardním portu 67/udp, klient na portu 68/udp.
- HTTP (Hypertext Transfer Protocol) je základním komunikačním protokolem umožňující klientům přístup k webovým službám na portu 80/tcp.
- HTTPS (Hypertext Transfer Protocol Secure) jde o webový komunikační protokol šifrovaný pomocí TLS nebo SSL na portu 443/tcp.
- NTP (Network Time Protocol) protokol sloužící k synchronizaci času přes počítačovou síť. Komunikuje přes well-known port 123/udp.

5 Analýza současného stavu

5.1 Topologie laboratorní sítě ÚI PEF MENDELU

V současné době je síťová laboratoř ÚI PEF MENDELU rozdělena, jak je vidět na Obr. 2, do 4 virtuálních LAN.



Obrázek 2: Zjednodušený diagram síťové topologie (Ústav informatiky, 2013)

VLAN 99

Nativní VLAN má číslo 99, je nazvaná Management a adresa sítě je 10.0.99.0/24. V této VLAN jsou zahrnuty dva administrátorské počítače s operačním systémem Linux (Správce 01 – S01, Správce 02 – S02), PDU master zajišťující zapínání elektrické sítě (IP adresa 10.0.99.3), konzolový server CPS Avocent CCS 16 port (IP adresa 10.0.99.4) a KVM via IP AdderLink iPEPS (IP adresa 10.0.99.5). Z počítače Správce 01 lze přes protokol SSH ovládat laboratorní směrovač, laboratorní přepínač, Linuxový server, ESXi 5.1 hardwarový hostitel nebo počítače ve VLAN 10. Přes Remote Desktop protokol je dále ovládán Windows server.

VLAN 30

Druhá VLAN je vedena pod číslem 30 s IP adresou 10.0.30.0/24 a nese název Servers. Nachází se zde VMware ESXi 5.1 (IP adresa 10.0.30.4) a dva virtuální servery. První virtuální server je linuxová distribuce CentOS (IP adresa 10.0.30.2). Na druhém virtuálním serveru je operační systém Windows Server.

Linuxový server poskytuje klientům v laboratoři služby jako DHCP, DNS, NTP, LDAP, RADIUS, FTP, CUPS, HTTP a Yum repositář. Windowsový server umožňuje připojení přes Rdesktop nebo AVG administraci.

Mezi Správcem 01 a Linuxserverem stejně jako mezi S01 a Winserverem je vytvořen VPN (Virtual Private Network), který slouží například k odesílání syslogu ze serveru na Správce 01.

VLAN 50

Třetí VLAN s číslem 50 a IP adresou 10.0.50.0/24 je nazvána Bakalanti. Slouží především pro zpracovávání závěrečných prací a lze v ní nalézt 8 počítačů. Počítače v této síti mají povolenou komunikaci s MikroTik směrovačem.

VLAN 10 a řešení výukových stanic

Poslední VLAN má číslo 10 a IP adresu 10.0.10.0/24. Jak vyplývá z technické dokumentace z roku 2012 umístěné na webové stránce Ústavu informatiky (2013), každý výukový počítač je vybaven dvěma síťovými kartami (NIC 1 a NIC 2). Jedna síťová karta připojuje počítač do laboratorní sítě – VLAN 10. Přestože hostitelský operační systém a Individuální instalace učitele mají přístup k Internetu povolen, virtuální operační systémy sloužící k výuce mají přístup k Internetu blokován. Druhá síťová karta připojuje počítače do sítě oddělené od laboratorní sítě. Tato síť slouží k výuce předmětu Počítačové sítě, kde je vyžadováno, aby počítač neměl připojení k MENDELU LAN a tudíž i do Internetu.

K připojení do laboratorní sítě se, jak již bylo zmíněno výše, využívá síťová karta (NIC 1). Tuto kartu využívá hostitelský počítač pro hostitelský operační systém Windows 8 a IPI. Hostitelským počítačům jsou DHCP protokolem přiřazovány IP adresy z rozsahu 10.0.10.100–117. Dále je síťová karta využívána virtuálními zařízeními běžícími v programu VirtualBox, které jsou k ní připojeny přes síťový most a jsou jim pomocí protokolu DHCP přiřazovány IP adresy z rozsahu 10.0.10.200–217. Síťový most je, jak vysvětluje dokumentace na serveru VirtualBox (2015), technologie, která využívá hostitelovu síťovou kartu, pro vytvoření nového rozhraní. Nové rozhraní má tedy svou vlastní IP adresu a lze ho adresovat. V dokumentaci na stránkách Ústavu informatiky (2013) je také zmíněn NAT mód, který umožňuje komunikaci virtuálních zařízení nazvaných Individuální instalace učitele. NAT mód popsán v dokumentaci VirtualBox (2015), neumožňuje na rozdíl od síťového mostu virtuálním zařízením vlastnit virtuální rozhraní. Virtuální zařízením je proto,

jak je vysvětleno v dokumentaci VirtualBox (2015), maskována interní IP adresa, za adresu hostitelského počítače.

Labswitch používá technologii Port security, která, jak uvádí dokumentace Cisco Systems (2015), kontroluje MAC adresy připojených síťových karet. Aby byla zabezpečena i ochrana v případě změny MAC adresy, je, jak ukazuje technická dokumentace na internetových stránkách Ústavu informatiky (2013), vyžadována autentizace 802.1x pomocí certifikátu.

Virtuální počítače připojené do výukové sítě využívají síťový most a druhou síťovou kartu (NIC 2). Technická dokumentace na internetových stránkách Ústavu informatiky (2015) ukazuje, že síťové karty jsou připojeny pouze do patch panelu v racku. Zde jsou následně využívány studenty předmětu Počítačové sítě. Do racku jsou připojeny i sériové porty (COM) hostitelských počítačů.

5.2 Adresace na L3

V Tab. 1 jsou uvedeny IP adresy Labrouteru. (Ústav informatiky, 2013)

Tabulka 1: Adresování v Labrouteru

Rozhraní	IP adresa	Síťová maska	VLAN
eth1	195.178.72.189	255.255.255.240	*
eth0	10.0.99.1	255.255.255.0	VLAN 99
eth0.10	10.0.10.1	255.255.255.0	VLAN 10
eth0.30	10.0.30.1	255.255.255.0	VLAN 30
eth0.50	10.0.50.1	255.255.255.0	VLAN 50

5.3 Fyzické prvky v laboratorní síti ÚI PEF MENDELU

Níže uvedená Tab. 2 popisuje fyzické prvky, které souvisejí s výše uvedenou topologií na Obr. 2. Uvedené údaje jsou převzaty z webové stránky Ústavu informatiky (2013).

Tabulka 2: Fyzické prvky (Ústav informatiky, 2013)

Název v laboratoři	Typ	Název	Počet
Labswitch	Přepínač	Cisco Catalyst 2960 (48TT-S)	1
Labrouter	Server	Dell PowerEdge T110 II	1
PDU master	PDU	PDU Server Technology	2
CPS	CPS	CPS Avocent CCS 16 port	1
KVM-IP	KVM-IP	KVM via IP AdderLink iPEPS	1
Výukové počítače	Počítač	HP Compaq 8000	17
Projektové počítače	Počítač	Dell Optiplex 390 MT	7

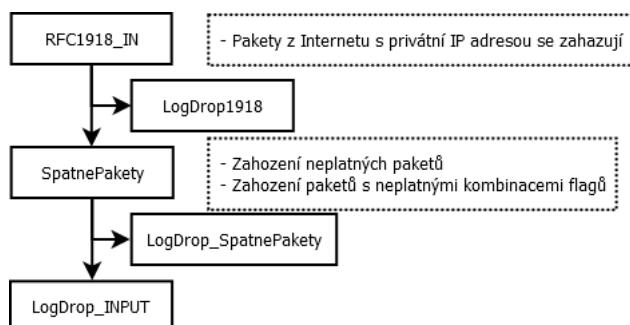
5.4 Firewallová politika

V této části je uvedena současná firewallová politika. Nejprve je třeba uvést, jakým způsobem současný firewall filtruje. V Iptables lze filtrovat na základě parametrů nebo kombinací několika parametrů současně. Současná filtrovací pravidla používají tyto parametry:

1. vstupní rozhraní,
2. výstupní rozhraní,
3. zdrojová IP adresa,
4. cílová IP adresa,
5. stav paketu (new, established, related, invalid),
6. protokol (TCP, UDP, ICMP),
7. zdrojový port,
8. cílový port,
9. typ icmp zprávy,
10. limit, což je, jak uvádí Russell (2002), průměrný počet shody paketu s pravidlem,
11. limit dávky, což je podle Russella (2002) počet shod paketu s pravidlem v jedné dávce.

Řetězec INPUT

Prvním řetězcem je INPUT, který filtruje pakety směřující do Labrouteru. Průchod řetězcem je znázorněn na Obr. 3. Jednotlivá pravidla povolují průchod paketů podle zadaných parametrů. Pokud paket nevyhoví žádnému z pravidel, je předán řetězci LogDrop_INPUT, který jej zaloguje a následně zahodí. Filtrování řetězce INPUT je zachyceno v Tab. 3.



Obrázek 3: Stromový výpis řetězce INPUT

Tabulka 3: Firewallová politika INPUT (provoz směřující na labrouter)

Cíl	Vstup. r.	Výstup. r.	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení všech paketů přicházejících z loopbacku.							
ACCEPT	loopback	*	0.0.0.0/0	0.0.0.0/0	*	*	*
2. Předání řetězci RFC1918, který zkontroluje pakety z vnějšího rozhraní.							
Pokud mají pakety zdrojovou IP adresu z rozsahu privátních adres, jsou zahozeny, jinak jsou vráceny zpět.							
RFC1918_IN	eth1	*	0.0.0.0/0	0.0.0.0/0	*	*	*
3. Předání řetězci SpatnePakety, který zahodí invalid pakety nebo pakety s nesmyslnými kombinacemi flagů.							
SpatnePakety	*	*	0.0.0.0/0	0.0.0.0/0	*	*	*
4. Povolení spojení ve stavu established nebo related.							
ACCEPT	0.0.0.0/0	0.0.0.0/0	*	*	rel., est.		
5. Povolení přístupu z S01 na Labrouter přes SSH spojení.							
ACCEPT	eth0	*	10.0.99.101	10.0.99.1	tcp	22	new
5. Povolení přístupu z S02 na Labrouter přes SSH spojení.							
ACCEPT	eth0	*	10.0.99.102	10.0.99.1	tcp	22	new
5. Povolení komunikace s již vyřazeným uzlem.							
ACCEPT	*	*	10.0.99.100	0.0.0.0/0	*		
6. Povolení pingu z Internetu na Labrouter.							
ACCEPT	eth1	*	0.0.0.0/0	0.0.0.0/0	icmp	*	typ 8
7. Povolení pingu z VLAN10 na Labrouter.							
ACCEPT	eth0.10	*	10.0.10.0/24	10.0.10.1	icmp	*	typ 8
8. Povolení pingu z VLAN30 na Labrouter.							
ACCEPT	eth0.30	*	10.0.30.0/24	10.0.30.1	icmp	*	typ 8
8. Povolení pingu z VLAN50 na Labrouter.							
ACCEPT	eth0.50	*	10.0.50.0/24	10.0.50.1	icmp	*	typ 8
8. Povolení pingu z VLAN99 na Labrouter.							
ACCEPT	eth0	*	10.0.99.0/24	10.0.99.1	icmp	*	typ 8
9. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_INPUT, kde jsou zalogovány a zahozeny.							
LogDrop_INPUT	*	*	0.0.0.0/0	0.0.0.0/0	*	*	*

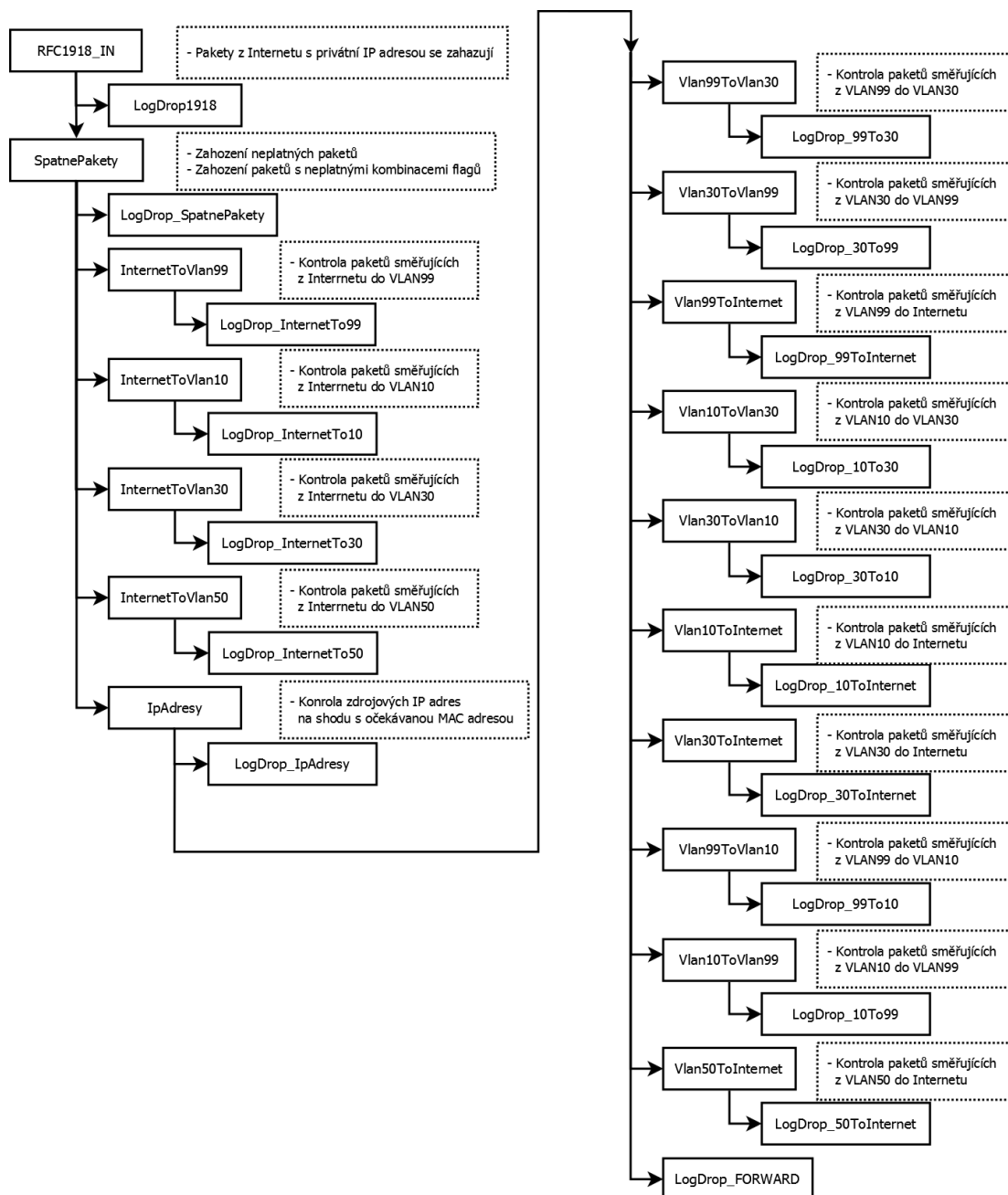
Řetězec FORWARD

Řetězec FORWARD slouží k filtrování tranzitního provozu. Pakety, které jsou filtrovány tímto řetězcem, jsou směrovány z jednoho rozhraní směrovače na jiné rozhraní. Mohou to být pakety, které jsou odeslány z jedné VLAN do jiné, nebo to mohou být pakety směřující z vnitřní sítě do Internetu a naopak.

Pakety jsou propuštěny přes firewall pouze v případě, že vyhovují některému z pravidel. Pakety jsou předávány do uživatelských řetězců podle toho, z jakého rozhraní přišly a přes jaké rozhraní mají být odeslány. Systém předávání je zobrazen na Obr. 4. V jednotlivých uživatelských řetězcích jsou dále definovány cílové porty, na které smí být pakety odeslány.

Pokud paket nevyhoví žádnému z pravidel, je předán řetězci LogDrop_FORWARD, který jej zaloguje a následně zahodí.

Filtrování řetězce FORWARD je zachycená v Tab. 4.



Obrázek 4: Stromový výpis řetězce FORWARD

Tabulka 4: Firewallová pravidla FORWARD (tranzitní provoz)

Cíl	Vstupní rozhraní	Výstupní rozhraní	Zdrojová IP	Cílová IP
1. Předání řetězci RFC1918, který zkontroluje pakety z vnějšího rozhraní. Pokud mají pakety zdrojovou IP adresu z rozsahu privátních adres, jsou zahozeny, jinak jsou vráceny zpět.				
RFC1918_IN	eth1	*	0.0.0.0/0	0.0.0.0/0
2. Předání řetězci SpatnePakety, který zahodí invalid pakety nebo pakety s nesmyslnými kombinacemi flagů.				
SpatnePakety	*	*	0.0.0.0/0	0.0.0.0/0
3. Předání paketu odeslaného z Internetu do VLAN99 řetězci InternetToVlan99, který určuje povolené služby.				
InternetToVlan99	eth1	eth0	0.0.0.0/0	10.0.99.0/24
4. Předání paketu odeslaného z Internetu do VLAN10 řetězci InternetToVlan10, který určuje povolené služby.				
InternetToVlan10	eth1	eth0.10	0.0.0.0/0	10.0.10.0/24
5. Předání paketu odeslaného z Internetu do VLAN30 řetězci InternetToVlan30, který určuje povolené služby.				
InternetToVlan30	eth1	eth0.30	0.0.0.0/0	10.0.30.0/24
6. Předání paketu odeslaného z Internetu do VLAN50 řetězci InternetToVlan50, který určuje povolené služby.				
InternetToVlan50	eth1	eth0.50	0.0.0.0/0	10.0.50.0/24
7. Předání řetězci IpAdresy na kontrolu shody MAC adres s IP adresou.				
IpAdresy	*	*	0.0.0.0/0	0.0.0.0/0
8. Předání paketu odeslaného z VLAN99 do VLAN30 řetězci Vlan99ToVlan30, který určuje povolené služby.				
Vlan99ToVlan30	eth0	eth0.30	10.0.99.0/24	10.0.30.0/24
9. Předání paketu odeslaného z VLAN30 do VLAN99 řetězci Vlan30ToVlan99, který určuje povolené služby.				
Vlan30ToVlan99	eth0.30	eth0	10.0.30.0/24	10.0.99.0/24
10. Předání paketu odeslaného z VLAN99 do Internetu řetězci Vlan99ToInternet, který určuje povolené služby.				
Vlan99ToInternet	eth0	eth1	10.0.99.0/24	0.0.0.0/0
11. Předání paketu odeslaného z VLAN10 do VLAN30 řetězci Vlan10ToVlan30, který určuje povolené služby.				
Vlan10ToVlan30	eth0.10	eth0.30	10.0.10.0/24	10.0.30.0/24
12. Předání paketu odeslaného z VLAN30 do VLAN10 řetězci Vlan30ToVlan10, který určuje povolené služby.				
Vlan30ToVlan10	eth0.30	eth0.10	10.0.30.0/24	10.0.10.0/24
13. Předání paketu odeslaného z VLAN10 do Internetu řetězci Vlan10ToInternet, který určuje povolené služby.				
Vlan10ToInternet	eth0.10	eth1	10.0.10.0/24	0.0.0.0/0
14. Předání paketu odeslaného z VLAN30 do Internetu řetězci Vlan30ToInternet, který určuje povolené služby.				
Vlan30ToInternet	eth0.30	eth1	10.0.30.0/24	0.0.0.0/0
15. Předání paketu odeslaného z VLAN99 do VLAN10 řetězci Vlan99ToVlan10, který určuje povolené služby.				
Vlan99ToVlan10	eth0	eth0.10	0.0.0.0/0	0.0.0.0/0
16. Předání paketu odeslaného z VLAN10 do VLAN99 řetězci Vlan10ToVlan99, který určuje povolené služby.				
Vlan10ToVlan99	eth0.10	eth0	0.0.0.0/0	0.0.0.0/0
17. Předání paketu odeslaného z VLAN50 do Internetu řetězci Vlan50ToInternet, který určuje povolené služby.				
Vlan50ToInternet	VLAN50	eth1	10.0.50.0/24	*
18. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_FORWARD, kde jsou zalogovány a zahozeny.				
LogDrop_FORWARD	*	*	0.0.0.0/0	0.0.0.0/0

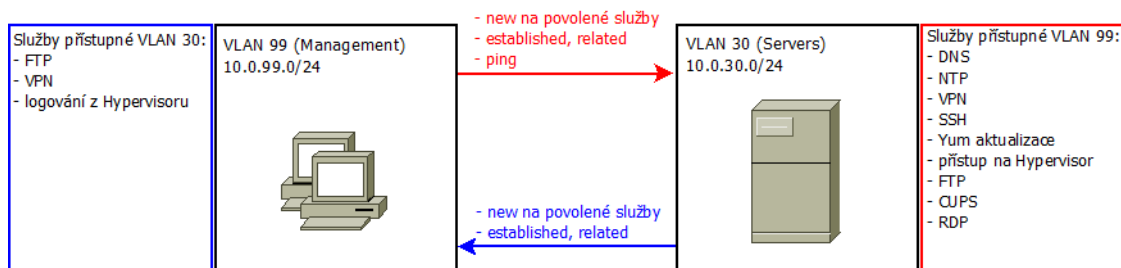
Řetězec OUTPUT

Do řetězce OUTPUT jsou filtrovány pakety, které jsou odesílány samotným labrouterem. V tomto řetězci se nenachází žádná pravidla. Řetězec má nastavenou defaultní politiku ACCEPT, tudíž cokoliv odesílá labrouter, je povoleno.

5.5 Komunikace v laboratorní síti ÚI PEF MENDELU

Komunikace mezi VLAN 30 a VLAN 99

VLAN 30 poskytuje VLAN 99 služby, které jsou popsány na Obr. 5. Druhý směr, tedy od VLAN 30 do VLAN 99, slouží především pro zápis syslogu nebo FTP spojení.



Obrázek 5: Komunikace mezi VLAN 30 a VLAN 99

V Tab. 5 jsou popsána spojení, která jsou povolena z VLAN 30 do VLAN 99. Tab. 6 dále popisuje pravidla, která povolují komunikaci z VLAN 99 do VLAN 30.

Tabulka 5: Řetězec Vlan30ToVlan99

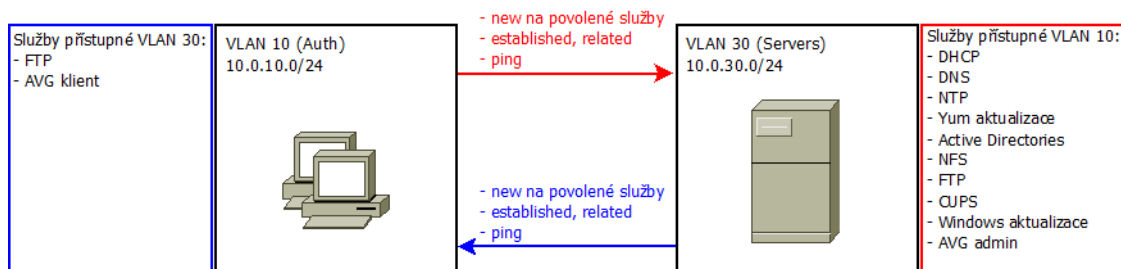
Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	10.0.99.101	*	*	est., rel.
2. Povolení spojení s již vyřazeným uzlem, který měl IP adresu 10.0.99.100.					
ACCEPT	0.0.0.0/0	10.0.99.100	*	*	*
1. Povolení VPN tunelu z Linuxového serveru na Správce 01.					
ACCEPT	10.0.30.2	10.0.99.101	udp	VPN 50001	new
2. Povolení VPN tunelu z Windowsového serveru na Správce 01.					
ACCEPT	10.0.30.3	10.0.99.101	udp	VPN 50002	new
3. Povolení logování z hostitelského serveru na Správce 01.					
ACCEPT	10.0.30.4	10.0.99.101	udp	514	new
4. Povolení FTP spojení z Linuxového serveru na Správce 01.					
Povolení řízení FTP (cílový port 21/tcp) i přenos dat (cílový port 20/tcp.)					
ACCEPT	10.0.30.2	10.0.99.101	tcp	21	new
5. Povolení FTP spojení z Linuxového serveru na Správce 02.					
Povolení řízení FTP (cílový port 21/tcp) i přenos dat (cílový port 20/tcp.)					
ACCEPT	10.0.30.2	10.0.99.102	tcp	21	new
6. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_30To99, kde jsou zalogovány a zahozeny.					
LogDrop_30To99	0.0.0.0/0	0.0.0.0/0	*	*	*

Tabulka 6: Řetězec Vlan99ToVlan30

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	0.0.0.0/0	*	*	est., rel.
2. Povolení spojení s již vyřazeným uzlem, který měl IP adresu 10.0.99.100.					
ACCEPT	10.0.99.100	0.0.0.0/0	*	*	*
3.–5. Povolení pingu ze Správce 01 na Linserver. Následují dvě pravidla, která stejným způsobem povolují ping i na hostitelský server a Winserver.					
ACCEPT	10.0.99.101	10.0.30.2	icmp	*	typ 8 (echo)
6.–8. Povolení pingu ze Správce 02 na Linserver. Následují dvě pravidla, která stejným způsobem povolují ping i na hostitelský server a Winserver.					
ACCEPT	10.0.99.101	10.0.30.2	icmp	*	typ 8 (echo)
9.–12. Povolení DNS ze Správce 01 do Linserveru. Následují tři pravidla, která stejným způsobem povolují průchod DNS paketů i ze Správce 02, Labswitche a CPS.					
ACCEPT	10.0.99.101	10.0.30.2	udp	53	new
13.–18. Povolení NTP pro Správce 01. Následuje 5 pravidel, která stejným způsobem povolují NTP i pro Správce 02, PDU master, KVM-IP, Labswitch a CPS.					
ACCEPT	10.0.99.101	10.0.30.2	udp	123	new
19. Povolení VPN tunelu ze Správce 01 na Linuxový server.					
ACCEPT	10.0.99.101	10.0.30.2	udp	VPN 50001	new
20. Povolení VPN tunelu ze Správce 01 na Windowsový server.					
ACCEPT	10.0.99.101	10.0.30.3	udp	VPN 50002	new
21. Povolení VPN tunelu ze Správce 01 na hostitelský server.					
ACCEPT	10.0.99.101	10.0.30.4	udp	VPN 50000	new
22.–23. Povolení přístupu ze Správce 01 na Linserver přes SSH spojení. Následuje pravidlo, které stejným způsobem povoluje SSH spojení ze Správce 02.					
ACCEPT	10.0.99.101	10.0.30.2	tcp	22	new
24.–25. Povolení přístupu ze Správce 01 na hostitelský server přes SSH spojení. Následuje pravidlo, které stejným způsobem povoluje SSH spojení ze Správce 02.					
ACCEPT	10.0.99.101	10.0.30.4	tcp	22	new
26. Povolení Remote Desktop spojení ze Správce 01 na Winserver.					
ACCEPT	10.0.99.101	10.0.30.3	tcp	3389	new
27. Povolení Remote Desktop spojení ze Správce 02 na Winserver.					
ACCEPT	10.0.99.102	10.0.30.3	tcp	3389	new
28. Povolení HTTP spojení ze Správce 01 na Linserver.					
ACCEPT	10.0.99.101	10.0.30.2	tcp	80	new
29. Povolení HTTP spojení ze Správce 02 na Linserver.					
ACCEPT	10.0.99.102	10.0.30.2	tcp	80	new
30.–31. Povolení HTTP spojení ze Správce 01 na hostitelský server. Následuje pravidlo, které stejným způsobem povoluje HTTPS spojení (cílový port 443/tcp).					
ACCEPT	10.0.99.101	10.0.30.2	tcp	80	new
32. Povolení přístupu na AVG admina na Winserveru ze Správce 01.					
ACCEPT	10.0.99.101	10.0.30.3	tcp	4158	new
33. Povolení přístupu na WSUS na Winserveru ze Správce 01.					
ACCEPT	10.0.99.101	10.0.30.3	tcp	8530	new
34. Povolení komunikace Labswitche s RADIUS serverem na Linserveru.					
ACCEPT	10.0.99.2	10.0.30.2	udp	1812	new
35. Povolení VMvare Server Console ze Správce 01 do hostitelského serveru.					
ACCEPT	10.0.99.101	10.0.30.4	tcp	902	new
36. Povolení FTP spojení ze Správce 01 na Linserver. Povolení řízení FTP (cílový port 21/tcp) i přenos dat (cílový port 20/tcp.)					
ACCEPT	10.0.99.101	10.0.30.2	tcp	21	new
37. Povolení FTP spojení ze Správce 02 na Linserver. Povolení řízení FTP (cílový port 21/tcp) i přenos dat (cílový port 20/tcp.)					
ACCEPT	10.0.99.102	10.0.30.2	tcp	21	new
38.–39. Povolení komunikace CUPS. Následuje pravidlo, které povoluje komunikaci CUPS i přes tcp protokol.					
ACCEPT	0.0.0.0/0	10.0.30.2	udp	631	new
40. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_99To30, kde jsou zalogovány a zahozeny.					
LogDrop_99To30	0.0.0.0/0	0.0.0.0/0	*	*	*

Komunikace mezi VLAN 10 a VLAN 30

Komunikace z VLAN 10 do VLAN 30 slouží k využívání služeb na serverech ve VLAN 30. Přehled služeb je k vidění na Obr. 6. Směr z VLAN 30 do VLAN 10 slouží především pro aktualizaci AVG klientů.



Obrázek 6: Komunikace mezi VLAN 10 a VLAN 30

Tab. 7 obsahuje firewallová pravidla, která se nacházejí v řetězci Vlan30ToVlan10. Pokud je nějaký paket odeslán z VLAN 30 do VLAN 10, je o jeho povolení rozhodnuto právě těmito pravidly.

Řetězec Vlan10ToVlan30, který rozhoduje o propuštění paketů ve směru z VLAN 10 do VLAN 30, je naplněn pravidly popsanými v Tab. 8.

Tabulka 7: Řetězec Vlan30ToVlan10

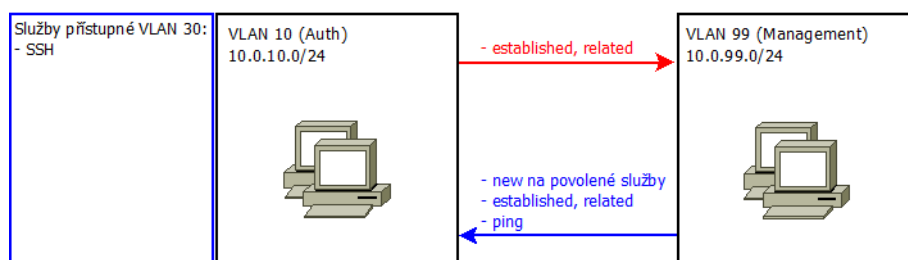
Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	0.0.0.0/0	*	*	est., rel.
2.–4. Povolení komunikace mezi AVG správcem na Winserveru a AVG klienty z VLAN 10.					
ACCEPT	10.0.30.3	0.0.0.0/0	tcp	6051	new
ACCEPT	10.0.30.3	0.0.0.0/0	tcp	6054	new
ACCEPT	10.0.30.3	0.0.0.0/0	tcp	4158	new
5. Povolení pingu z Winserveru do VLAN 10.					
ACCEPT	10.0.30.3	0.0.0.0/0	icmp	*	typ 8 (echo)
6.–7. Povolení FTP spojení z VLAN 30 do VLAN 10.					
Povolení řízení FTP (cílový port 21/tcp) i přenos dat (cílový port 20/tcp.)					
ACCEPT	0.0.0.0/0	0.0.0.0/0	tcp	21	new
8.–9. Povolení Microsoft-DS z VLAN 30 do VLAN 10.					
ACCEPT	0.0.0.0/0	0.0.0.0/0	tcp	445	new
ACCEPT	0.0.0.0/0	0.0.0.0/0	udp	445	new
9. Povolení HTTP z VLAN 30 do VLAN 10.					
ACCEPT	0.0.0.0/0	0.0.0.0/0	tcp	80	new
55. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_30To10, kde jsou zalogovány a zahozeny.					
LogDrop_30To10	0.0.0.0/0	0.0.0.0/0	*	*	*

Tabulka 8: Řetězec Vlan10ToVlan30

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	0.0.0.0/0	*	*	est., rel.
2.-3. Povolení pingů z VLAN 10 na Linserver a Winserver.					
ACCEPT	0.0.0.0/0	10.0.30.2	icmp	*	typ 8 (echo)
ACCEPT	0.0.0.0/0	10.0.30.3	icmp	*	typ 8 (echo)
4. Povolení spojení na DNS server na Linserveru.					
ACCEPT	0.0.0.0/0	10.0.30.2	udp	53	new
5.-6. Povolení spojení na DNS server na Winserver.					
Šesté pravidlo je podobné jako páté, ale povoluje komunikaci na cílový port 53/tcp.					
ACCEPT	0.0.0.0/0	10.0.30.3	udp	53	new
7. Povolení spojení na DHCP server na Linserveru.					
ACCEPT	0.0.0.0/0	10.0.30.2	udp	67	new
8. Povolení spojení na NTP server na Linserveru.					
ACCEPT	0.0.0.0/0	10.0.30.2	udp	123	new
9. Povolení HTTP z VLAN 10 na Linserver.					
ACCEPT	0.0.0.0/0	10.0.30.2	tcp	80	new
10.-11. Povolení FTP spojení z VLAN 10 na Linserver.					
Povolení řízení FTP (cílový port 21/tcp) i přenos dat (cílový port 20/tcp.)					
ACCEPT	0.0.0.0/0	10.0.30.2	tcp	21	new
12.-13. Povolení HTTP z VLAN 10 na Winserver.					
Třinácté pravidlo stejným způsobem povoluje HTTPS (port 443/tcp).					
ACCEPT	0.0.0.0/0	10.0.30.2	tcp	80	new
14. Povolení WSUS z VLAN 10 do Linserveru.					
ACCEPT	0.0.0.0/0	10.0.30.2	tcp	8530	new
15.-17. Povolení komunikace mezi AVG klienty z VLAN 10 a AVG správcem na Winserveru.					
ACCEPT	0.0.0.0/0	10.0.30.3	tcp	6051	new
ACCEPT	0.0.0.0/0	10.0.30.3	tcp	6054	new
ACCEPT	0.0.0.0/0	10.0.30.3	tcp	4158	new
18.-29. Povolení portů pro komunikaci s NFS serverem.					
V dalších pravidlech jsou povoleny následující cílové porty:					
892/tcp, 892/udp, 2049/tcp, 2049/udp, 32803/tcp, 32803/udp, 662/udp, 662/tcp, 32769/udp, 32769/tcp					
ACCEPT	0.0.0.0/0	10.0.30.2	tcp	111	new
ACCEPT	0.0.0.0/0	10.0.30.2	udp	111	new
30. Povolení LDAP z VLAN 10 do Linserveru.					
ACCEPT	0.0.0.0/0	10.0.30.2	tcp	389	new
31.-32. Povolení komunikace CUPS pro protokol udp i tcp.					
ACCEPT	0.0.0.0/0	10.0.30.2	udp	631	new
33. Otevřené porty pro FTP na Linserveru.					
ACCEPT	0.0.0.0/0	10.0.30.2	tcp	30000:30100	new
34.-54. Povolení portů pro komunikaci s Active Directory na Winserveru.					
V dalších pravidlech je povolena komunikace přes následující cílové porty:					
3268/tcp, 138/udp, 445/tcp, 445/udp, 636/tcp, 3269/tcp, 123/udp, 135/tcp, 464/tcp, 464/udp, 9389/udp, 67/udp, 2535/udp, 88/udp, 137/udp, 88/tcp, 25/tcp, 49152:65535/tcp					
ACCEPT	0.0.0.0/0	10.0.30.3	tcp	389	new
ACCEPT	0.0.0.0/0	10.0.30.3	udp	389	new
55. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_10To30, kde jsou zalogovány a zahozeny.					
LogDrop_10To30	0.0.0.0/0	0.0.0.0/0	*	*	*

Komunikace mezi VLAN 10 a VLAN 99

Toky mezi VLAN 99 a VLAN 10 tvoří pouze SSH spojení navázané z VLAN 99. Z VLAN 10 lze odesílat pouze pakety, které patří k již ustanovenému spojení nebo s ním souvisí. Povolená komunikace je popsána v Obr. 7.



Obrázek 7: Komunikace mezi VLAN 10 a VLAN 99

Firewallová pravidla, která povolují komunikaci ve směru z VLAN 10 do VLAN 99, jsou popsána v Tab. 9. Opačný směr toku paketů tedy z VLAN 99 do VLAN 10 je filtrován pomocí pravidel, která lze vidět v Tab. 10.

Tabulka 9: Řetězec Vlan10ToVlan99

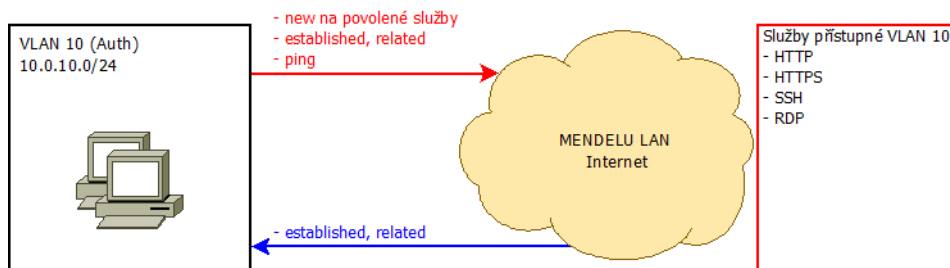
Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení z VLAN 10 na Správce 01 ve stavu established nebo related.					
ACCEPT	10.0.10.0/24	10.0.99.101	*	*	est., rel.
2. Všechny ostatní pakety, jsou předány řetězci LogDrop_10To99, kde jsou zalogovány a zahozeny.					
LogDrop_10To99	0.0.0.0/0	0.0.0.0/0	*	*	*

Tabulka 10: Řetězec Vlan99ToVlan10

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení ze Správce 01 do VLAN 10 ve stavu established nebo related.					
ACCEPT	10.0.99.101	10.0.10.0/24	*	*	est., rel.
2. Povolení SSH spojení ze Správce 01 do VLAN 10.					
ACCEPT	10.0.99.101	10.0.10.0/24	tcp	22	new
3. Povolení pingu ze Správce 01 do VLAN 10.					
ACCEPT	10.0.99.101	10.0.10.0/24	icmp	*	typ 8 (echo)
4. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_99To10, kde jsou zalogovány a zahozeny.					
LogDrop_99To10	0.0.0.0/0	0.0.0.0/0	*	*	*

Komunikace mezi VLAN 10 a Internetem

V následujícím Obr. 8 jsou zobrazeny služby, které jsou dostupné pro VLAN 10 z Internetu. Veškerá komunikace z Internetu do VLAN 10 smí být pouze ve stavu ustanovené nebo související komunikace.



Obrázek 8: Komunikace mezi VLAN 10 a Internetem

Pravidla, která povolují komunikaci z VLAN 10 do Internetu, jsou shrnuta v Tab. 11. Povolená spojení z Internetu do VLAN 10 jsou popsána v Tab. 12.

Tabulka 11: Řetězec Vlan10ToInternet

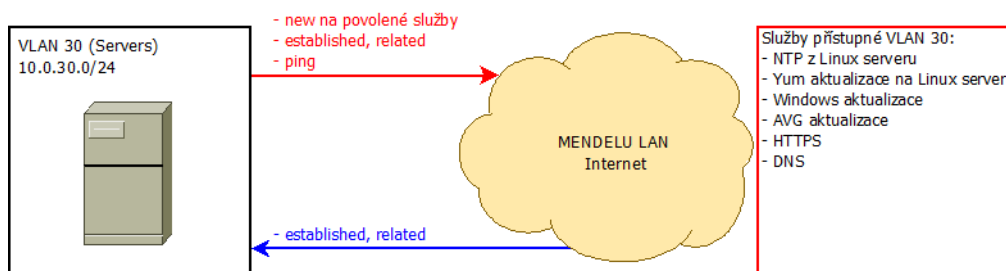
Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1.–16. Zakázání přístupu na Internet z virtuálních počítačů, které slouží pro výuku Operačních systémů. Pokud paket shodně s jedním z těchto pravidel je mu odeslána ICMP zpráva port-unreachable. Stejná pravidla jsou i pro pakety se zdrojovou IP adresou 10.0.10.202–216.					
REJECT	10.0.10.201	0.0.0.0/0	*	*	port-unreach.
17.–18. Povolení HTTP spojení z VLAN 10 do Internetu. Následuje pravidlo, které stejným způsobem povoluje HTTPS spojení (cílový port 443/tcp).					
ACCEPT	10.0.10.0/24	0.0.0.0/0	tcp	80	*
19. Povolení SSH spojení z VLAN 10 do Internetu.					
ACCEPT	10.0.10.0/24	0.0.0.0/0	tcp	22	*
20. Povolení Remote Desktop z VLAN 10 do Internetu.					
ACCEPT	10.0.10.0/24	0.0.0.0/0	tcp	3389	*
21. Povolení pingu z VLAN 10 do Internetu.					
ACCEPT	10.0.10.0/24	0.0.0.0/0	icmp	*	typ 8 (echo)
22. Povolení komunikace mezi KMS klientem a KMS server.					
ACCEPT	10.0.10.0/24	0.0.0.0/0	tcp	1688	*
23. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_10ToInternet, kde jsou zalogovány a zahozeny.					
LogDrop_10ToInternet	0.0.0.0/0	0.0.0.0/0	*	*	*

Tabulka 12: Řetězec InternetToVlan10

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení z Internetu do VLAN 10 pouze ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	0.0.0.0/0	*	*	est., rel.
2. Všechny ostatní pakety, jsou předány řetězci LogDrop_InternetTo10, kde jsou zalogovány a zahozeny.					
LogDrop_InternetTo10	0.0.0.0/0	0.0.0.0/0	*	*	*

Komunikace mezi VLAN 30 a Internetem

Obr. 9 popisuje, jaké služby jsou dostupné pro VLAN 30 z Internetu. Komunikace z Internetu do VLAN 30 je povolena pouze ve stavu ustanovená nebo související spojení.



Obrázek 9: Komunikace mezi VLAN 30 a Internetem

Všechny povolené služby popsané na Obr. 9 jsou povoleny v řetězci Vlan30ToInternet. Pravidla v tomto řetězci jsou popsána v Tab. 13. Paket, který byl odeslán z Internetu do VLAN 30, je filtrován v řetězci InternetToVlan30, který je popsán Tab. 14.

Tabulka 13: Řetězec Vlan30ToInternet

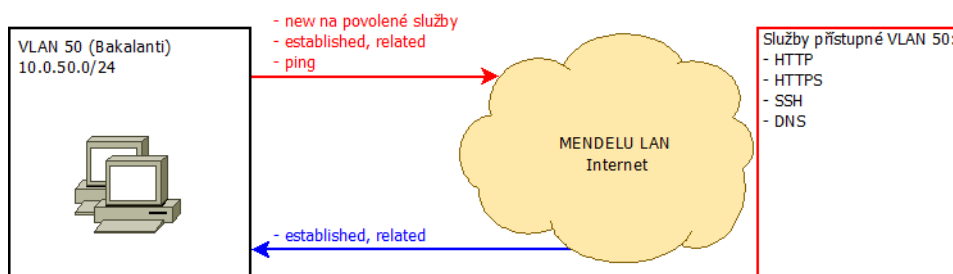
Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení komunikace DNS z Linserveru do Internetu.					
ACCEPT	10.0.30.2	0.0.0.0/0	udp	53	*
2. Povolení pingu z Linserveru do Internetu.					
ACCEPT	10.0.30.2	0.0.0.0/0	icmp	*	typ 8 (echo)
3. Povolení pingu z Winserveru do Internetu.					
ACCEPT	10.0.30.3	0.0.0.0/0	icmp	*	typ 8 (echo)
4. Povolení komunikace NTP z Linserveru do Internetu.					
ACCEPT	10.0.30.2	0.0.0.0/0	udp	123	*
5. Povolení RSYNC kvůli aktualizaci Yum z Linuxserveru do Internetu.					
ACCEPT	10.0.30.2	0.0.0.0/0	tcp	873	*
6.–7. Povolení HTTP spojení z Winserveru do Internetu. Následuje pravidlo, které stejným způsobem povoluje HTTPS spojení (cílový port 443/tcp).					
ACCEPT	10.0.30.3	0.0.0.0/0	tcp	80	*
8.–9. Povolení HTTP spojení z hostitelského serveru do Internetu. Následuje pravidlo, které stejným způsobem povoluje HTTPS spojení (cílový port 443/tcp).					
ACCEPT	10.0.30.4	0.0.0.0/0	tcp	80	*
10. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_30ToInternet, kde jsou zalogovány a zahozeny.					
LogDrop_30ToInternet	0.0.0.0/0	0.0.0.0/0	*	*	*

Tabulka 14: Řetězec InternetToVlan30

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení z Internetu do Linserveru ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	10.0.30.2	*	*	est., rel.
2. Povolení spojení z Internetu do Winserveru ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	10.0.30.3	*	*	est., rel.
3. Všechny ostatní pakety, jsou předány řetězci LogDrop_InternetTo30, kde jsou zalogovány a zahozeny.					
LogDrop_InternetTo30	0.0.0.0/0	0.0.0.0/0	*	*	*

Komunikace mezi VLAN 50 a Internetem

Služby z Internetu, ke kterým mají přístup počítače z VLAN 50, jsou popsány na Obr. 10.



Obrázek 10: Komunikace mezi VLAN 50 a Internetem

Tab. 15 obsahuje pravidla, kterými je naplněn řetězec Vlan50ToInternetu. Tento řetězec slouží k filtraci paketů, které jsou odeslány z VLAN 50 do Internetu.

Pro pakety směřující z Internetu do VLAN 50 je připraven uživatelský řetězec InternetToVlan50. Tento řetěze je popsán pomocí Tab. 16.

Tabulka 15: Řetězec Vlan50ToInternet

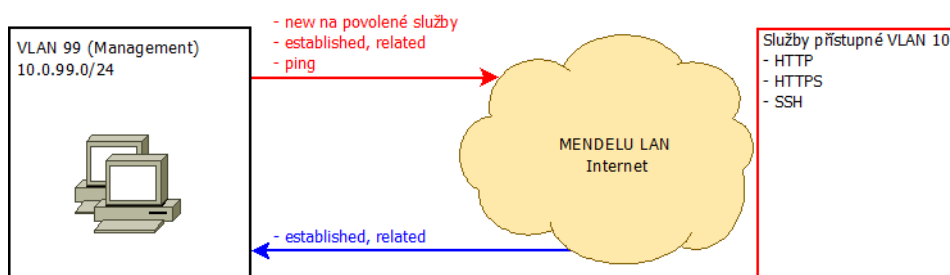
Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1.–2. Povolení HTTP spojení z VLAN 50 do Internetu.					
Následuje pravidlo, které stejným způsobem povoluje HTTPS spojení (cílový port 443/tcp).					
ACCEPT	10.0.50.0/24	0.0.0.0/0	tcp	80	*
3. Povolení SSH spojení z VLAN 50 do Internetu.					
ACCEPT	10.0.50.0/24	0.0.0.0/0	tcp	22	*
4. Povolení komunikace DNS z VLAN 50 do Internetu.					
ACCEPT	10.0.50.0/24	0.0.0.0/0	udp	53	*
5. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_50ToInternet, kde jsou zalogovány a zahozeny.					
LogDrop_50ToInternet	0.0.0.0/0	0.0.0.0/0	*	*	*

Tabulka 16: Řetězec InternetToVlan50

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení z Internetu VLAN 50 pouze ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	0.0.0.0/0	*	*	est., rel.
2. Všechny ostatní pakety, jsou předány řetězci LogDrop_InternetTo50, kde jsou zalogovány a zahozeny.					
LogDrop_InternetTo50	0.0.0.0/0	0.0.0.0/0	*	*	*

Komunikace mezi VLAN 99 a Internetem

Poslední směry, kterými mohou být pakety odeslány, jsou z VLAN 99 do Internetu a z Internetu do VLAN 99. Povolené služby jsou popsány pomocí Obr. 11.



Obrázek 11: Komunikace mezi VLAN 99 a Internetem

Konkrétní pravidla, která se vyskytují v řetězci Vlan99ToInternet, jsou specifikována v Tab. 17. Pakety směřující z Internetu do VLAN 99 jsou filtrována do uživatelského řetězce InternetToVlan99, jehož popis se nachází v Tab. 18.

Tabulka 17: Řetězec Vlan99ToInternet

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení veškerého spojení ze Správce 01 do Internetu.					
ACCEPT	10.0.99.101	0.0.0.0/0	*	*	*
2.–3. Povolení HTTP spojení ze Správce 02 do Internetu. Následuje pravidlo, které stejným způsobem povoluje HTTPS spojení (cílový port 443/tcp).					
ACCEPT	10.0.99.102	0.0.0.0/0	tcp	80	*
4. Povolení SSH spojení ze Správce 02 do Internetu.					
ACCEPT	10.0.99.102	0.0.0.0/0	tcp	22	*
5. Všechny pakety, které nebyly filtrovány v některém z předešlých pravidel, jsou předány řetězci LogDrop_99ToInternet, kde jsou zalogovány a zahozeny.					
LogDrop_99ToInternet	0.0.0.0/0	0.0.0.0/0	*	*	*

Tabulka 18: Řetězec InternetToVlan99

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav
1. Povolení spojení z Internetu do Správce 01 ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	10.0.99.101	*	*	est., rel.
2. Povolení spojení z Internetu do Správce 02 ve stavu established nebo related.					
ACCEPT	0.0.0.0/0	10.0.99.102	*	*	est., rel.
3. Všechny ostatní pakety, jsou předány řetězci LogDrop_InternetTo99, kde jsou zalogovány a zahozeny.					
LogDrop_InternetTo30	0.0.0.0/0	0.0.0.0/0	*	*	*

Řetězec SpatnePakety

Tento uživatelský řetězec slouží k odfiltrování vadných paketů. Pakety mohou být označeny za vadné v případě, že jsou ve stavu invalid.

Vadné pakety jsou také ty, které obsahují nesmyslné nastavení flagů. Tato nesmyslná nastavení slouží k rozpoznání operačního systému koncového počítače, což slouží především k přípravě síťových útoků.

Firewallová pravidla z řetězce SpatnePakety jsou popsána v Tab. 19.

Tabulka 19: Řetězec SpatnePakety

Cíl	Zdrojová IP	Cílová IP	Protokol	Cílový port	Stav	Flag
1. Zalogování a zahození paketů ve stavu invalid.						
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	*	*	invalid	*
2.–8. Zalogování a zahození paketů s nesmyslnými kombinacemi flagů. Zahozením těchto paketů docílíme toho, že neoprávněná osoba nezíská informace o operačním systému uzlu, na který se snaží zaútočit.						
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	tcp	*	*	0x3F/0x29
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	tcp	*	new	!0x17/0x02
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	tcp	*	*	0x3F/0x00
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	tcp	*	*	0x3F/0x3F
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	tcp	*	*	0x3F/0x37
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	tcp	*	*	0x06/0x06
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	tcp	*	*	0x03/0x03
9.–13. Zalogování a zahození paketů mířících z VLAN 99 s IP adresou broadcast. Stejná pravidla jsou i pro pakety mířící z Internetu, VLAN 10, VLAN 30 a VLAN 50.						
LogDrop_SpatnePakety	255.255.255.255	*	*	*	*	*
14.–18. Zalogování a zahození paketů mířících z VLAN 99 s IP adresou z rozsahu 127.0.0.0/8. Stejná pravidla jsou i pro pakety mířící z Internetu, VLAN 10, VLAN 30 a VLAN 50.						
LogDrop_SpatnePakety	127.0.0.0/8	*	*	*	*	*
19. Zalogování a zahození paketů mířících na cílový port 137/udp.						
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	udp	137	*	*
20. Zalogování a zahození paketů mířících na cílový port 138/udp.						
LogDrop_SpatnePakety	0.0.0.0/0	0.0.0.0/0	udp	138	*	*
21. Všechny ostatní pakety, jsou vráceny zpět nadřazenému řetězci.						
RETURN	0.0.0.0/0	0.0.0.0/0	*	*	*	*

Řetězec RFC1918_IN

Řetězec kontroluje zdrojové adresy paketů přicházejících z WAN portu eth1. Pokud je zdrojová adresa paketu privátní (192.168.0.0/16, 10.0.0.0/8, 172.16.0.0/12), je

paket předán řetězci LogDrop1918, jinak je paket vrácen nadřazenému řetězci.

Řetězec IpAdresy

V tomto uživatelském řetězci se testují zdrojové IP adresy paketů na shodnost s očekávanou MAC adresou. Navíc se zde kontrolují pakety z VLAN 50, zda jejich zdrojové IP adresy odpovídají povoleným IP adresám.

Řetězce LogDrop

Pokud je splněn limit pro logování, což je v případě Iptables v laboratoři ÚI PEF MENDELU průměrně 10 shod paketů s pravidlem za hodinu, je paket zalogován a následně zahozen. V případě že není limit naplněn, jsou pakety zahazovány ihned.

5.6 Závěr analýzy

1. Předávání paketů v základním řetězci uživatelským řetězcům je logické a díky vhodnému pojmenování uživatelských řetězců se lze snadno orientovat v pravidlech. Proto je autorem doporučeno tento způsob filtrování ponechat.
2. Změna bude provedena v přesunutí pravidla, které kontroluje pakety patřící k ustanovenému spojení nebo s ním souvisí, na začátek každého ze základních řetězců. V nynějším firewallu jsou tato pravidla v každém z uživatelských řetězců, což zpomaluje filtrování paketů.
3. Dále bude provedena změna pořadí pravidel v uživatelských řetězcích tam, kde to neovlivní funkcionalitu. Je důležité přesunout pravidla, která povolují nejvíce paketů, na začátek řetězce, protože čím výše pravidlo je, tím rychleji je paket povolen. Rychlejší povolení paketů také méně zatěžuje procesor směrovače, který filtrování provádí.
Tato změna je navržena na základě sledování provozu v síťové laboratoři ÚI PEF MENDELU v běžném studijním týdnu. Toto sledování začalo v 12:50 h 23. listopadu 2015 a skončilo v 18:15 h 2. prosince 2015. Pro sledování byla využita počítačidla splnění jednotlivých pravidel, která Iptables umožňují měřit.
4. Dále se v nynějším firewallu vyskytovala některá duplicitní pravidla, která budou v novém návrhu vypuštěna. Ve firewallu byla nalezena pravidla související s IP adresou 10.0.99.100, která však nyní již nepatří žádnému prvku v síti, a proto budou tato pravidla v novém návrhu zrušena. Některá pravidla, která sloužila pro povolení přístupu k určitým službám například k AVG na Windows serveru, nebudou do nového návrhu zahrnuta, protože se jedná o již nepoužívané služby.
5. Při penetračním testování pod dohledem správce síťové laboratoře bylo zjištěno, že se firewall nedokáže účinně bránit útokům záplavou paketů tzv. DoS. Ochranu proti těmto útokům tedy navrhuji přidat do nového firewallu.

6 Návrh řešení

Návrh nového řešení se silně opírá o analýzu současného stavu. Nejprve bude představena nová topologie laboratorní sítě ÚI PEF MENDELU. Hlavní a podstatná změna nastává ve změně laboratorního firewallu (směrovače). Stávající labrouter bude nahrazen novým směrovačem od firmy MikroTik. Stávající labrouter je linuxový server s povolenou technologií IP forwarding, která umožňuje posílat pakety mezi rozhraními. Nový routeru MikroTik RouterBOARD 951-2n s verzí RouterOS 6.32.3 je malý směrovač s pěti Fast ethernetovými porty. V budoucnu je plánováno pořízení výkonnějšího směrovače s Gigabit ethernetovými porty.

6.1 Fyzické zapojení

Do Fast ethernetového konektoru s označením ether1 bude zapojena komunikace z vnitřní sítě. Linka vedoucí do portu ether1 bude umožňovat forwardovat tagované pakety, protože je potřeba zajistit komunikaci z jedné VLAN do jiné, což umožňuje pouze komunikace na třetí vrstvě ISO/OSI.

Komunikace z Internetu bude směřovat do Fast ethernetového portu ether2, který se tím stává tzv. WAN portem. Přes tento port bude celá laboratoř komunikovat s Internetem.

Nákres topologie na Obr. 12 je upravený nákres topologie z webové stránky Ústavu informatiky (2015).

6.2 Základní nastavení labrouteru MikroTik

Po fyzickém zapojení přichází na řadu základní nastavení routeru. Jde o nastavení nezbytné pro správu a ovládání routeru.

Název zařízení

Jako první je důležité změnit název samotného routeru. To je důležité zejména pro orientaci při ovládání routeru z jiného počítače například přes SSH spojení.

Protože je v laboratoři využíván interní DNS server na linuxovém serveru, má labrouter (10.0.99.1) vlastní záznam v doméně netlab.local.

Je tedy logické zachovat pojmenování nového routeru podle stávajícího labrouteru tedy Labrouter. Bude tak souhlasit nejen název zařízení (hostname), který je vidět při ovládání labrouteru přes příkazovou řádku nebo Winbox, ale bude správně i překládané volání pomocí doménového jména.

Přístupová práva

Na MikroTik routerech je defaultně nastaven administrátorský účet admin, který je při prvním spuštění bez hesla. Je tedy nutné při prvotním nastavení přidělit tomuto účtu heslo, aby neoprávněné osoby neměly možnost labrouter ovládat. Vlastní heslo

bude přiděleno správcem laboratoře, který se bude tímto heslem autentizovat. Další ochranou je přidání povolených IP, ze kterých se lze na labrouter přihlásit. Kvůli bezpečnosti budou do povolených adres zařazeny pouze adresy z VLAN 99.

Účet admin patří do skupiny Full, která má právo měnit veškerá nastavení routeru.

Protože je běžné provádět administraci linuxových zařízení pod účtem root, bude přidán tento účet do Labrouteru také. Heslo k účtu bude vytvořeno správcem sítě laboratoře. Přihlášení pod tímto účtem bude možné pouze z VLAN 99, stejně jako je to v případě účtu admin. Účet root bude také zařazen do skupiny full a bude sloužit zejména ke spouštění skriptů.

Synchronizace času

MikroTik umožňuje nastavení časového pásma v sekci System a podsekci Clock. Nastavení časové zóny na automatickou detekci umožní automatické zjištění, ve kterém časovém pásmu se router nachází.

Pro správné fungování komunikace je nutné, aby byl čas synchronizovaný s ostatními prvky v síti. Přidáním balíčku ntp získá MikroTik router možnost synchronizovat svůj čas pomocí NTP protokolu. V laboratoři se všechna zařízení synchronizují s linuxovým serverem, proto bude primární server pro synchronizaci právě linuxový server (10.0.30.2).

Odesílání logovacích záznamů

Aby měl správce labrouteru přístup k informacím, jaké pakety byly zahozeny, je nutné nastavit logování. Protože se v síťové laboratoři využívá Syslog server ve Správci 01 (10.0.99.101), budou logovací záznamy z firewallu přeposílány do tohoto centrálního syslogu.

6.3 Nastavení rozhraní

Dalším krokem je nastavení rozhraní routeru. Protože router musí forwardovat pakety mezi virtuálními lokálními sítěmi (VLAN), je třeba na rozhraní ether1 navázat všechny virtuální LAN z vnitřní sítě.

Vytvoření VLAN

VLAN se v RouterOS vytvoří v sekci Interface a v podsekci VLAN. Zde se nejprve vytvoří všechny virtuální LAN, které se poté přidají na určité fyzické rozhraní.

Jako první VLAN bude vytvořena VLAN pro studentské počítače, která bude mít číslo 10. Tato VLAN ponese označení VLAN10 a bude mít VLAN ID také 10. Této VLAN bude následně přiděleno fyzické rozhraní ether1. Stejným způsobem budou vytvořeny i VLAN 30 a 50. VLAN 99 pro management laboratoře je nativní VLAN. MikroTik funkci nativní VLAN svazuje přímo s fyzickým rozhraním ether1,

a proto se nebude virtuální lokální síť 99 vytvářet jako VLAN. Vytvořené virtuální LAN jsou vidět v Tab. 20.

Tabulka 20: Seznam VLAN

Rozhraní	Název VLAN	VLAN ID
ether1	VLAN10	10
ether1	VLAN30	30
ether1	VLAN50	50

Přidělení IP adres

Jako první bude vytvořena nativní VLAN 99 a to tak, že se fyzickému rozhraní ether1 přidělí IP adresa 10.0.99.1/24. Další IP adresy budou přiděleny přímo pro jednotlivé VLAN. Jako poslední bude přidána adresa pro vnější rozhraní (ether2), která byla získána z analýzy tabulky IP adres stávajícího labrouteru. Přidělení adres znázorňuje Tab. 21.

Tabulka 21: Seznam IP adres

Rozhraní	IP adresa	IP adresa sítě
ether1	10.0.99.1/24	10.0.99.0
VLAN10	10.0.10.1/24	10.0.10.0
VLAN30	10.0.30.1/24	10.0.30.0
VLAN50	10.0.50.1/24	10.0.50.0
ether2	195.178.72.189/28	195.178.72.176

6.4 Směrovací tabulka

Směrovací tabulka slouží k rozhodování, kam má být datagram, který se objeví ve směrovači, odeslán. Labrouter využívá statickou směrovací tabulku, protože se záznamy ve směrovací tabulce nemění.

Některé části směrovací tabulky se naplní samy, jiné je třeba ručně přidat. Záznamy pro přímo připojené sítě se přidávají po zadání IP adresy některému z rozhraní. Statický záznam, který bude třeba do směrovací tabulky přidat, je defaultní brána, na kterou je datagram odeslán, pokud není v tabulce vhodnější cíl. Obsah směrovací tabulky Labrouteru je znázorněn v Tab. 22.

Tabulka 22: Směrovací tabulka

Stav	Cílová IP adresa	Brána	IP adresa brány
dynamická, aktivní, přímo připojená	10.0.99.0/24	dostupné přes ether1	10.0.99.1
dynamická, aktivní, přímo připojená	10.0.10.0/24	dostupné přes VLAN10	10.0.10.1
dynamická, aktivní, přímo připojená	10.0.30.0/24	dostupné přes VLAN30	10.0.30.1
dynamická, aktivní, přímo připojená	10.0.50.0/24	dostupné přes VLAN50	10.0.50.1
dynamická, aktivní, přímo připojená	195.178.72.176/28	dostupné přes ether2	195.178.72.189
statická, aktivní	0.0.0.0/0	195.178.72.177	

6.5 DHCP Relay

V laboratoři je využíván RADIUS server spuštěný na linuxovém serveru, který na základě certifikátů přiděluje výukovým stanicím ve VLAN 10 IP adresy, IP adresu DNS serveru a IP adresu výchozí brány. Aby bylo možné přesměrovat DHCP pakety na linuxový server, je nezbytné nastavit na novém labrouteru technologii DHCP Relay. Jak uvádí MikroTik (2015), DHCP Relay je pouze jakýsi zástupce DHCP serveru, který přijme DHCP pakety a přeopíše je skutečnému DHCP serveru.

Nový labrouter je tedy nutné nastavit tak, že pokud se na rozhraní VLAN 10 objeví DHCP paket, labrouter jej přesměruje na linuxový server ve VLAN 30.

6.6 Návrh nového firewallu

Nový firewall bude mít stejnou výchozí politiku jako ten stávající. Veškerá odchozí komunikace (řetězec OUTPUT) bude povolena, protože tuto komunikaci bude řídit administrátor na samotném firewallu.

Výchozí politika, která se bude uplatňovat pro komunikaci směřující do firewallu (řetězec INPUT), bude povolení komunikace, kterou vymezení správce síťové laboratoře ÚI PEF MENDELU, veškerá ostatní komunikace bude zakázána.

Výchozí politika pro tranzitní provoz (řetězec FORWARD) bude nastavena na stejném principu jako výchozí politika pro řetězec INPUT. Správce laboratoře určí, která komunikace v jakém směru má být propuštěna. Komunikace z Internetu do vnitřní sítě může být pouze v ustanoveném nebo souvisejícím spojení. Povolena komunikace mezi jednotlivými VLAN je určena v uživatelských řetězcích. Ostatní pakety, které se neshodnou s žádným pravidlem, které by je propustilo přes firewall, budou zahazovány.

Způsob filtrace paketů

Jak bylo poznamenáno v prvním bodu závěru analýzy, v novém firewallu bude ponecháno označení a způsob filtrování do uživatelských řetězců. V novém firewallu zůstanou i všechny kontrolní řetězce. Kontrolními řetězci se rozumí řetězec RFC1819_IN, který slouží pro zahození paketů z Internetu s privátními adresami, dále řetězec SpatnePakety definující vadné pakety, které nesmějí být propuštěny, a řetězec IpAdresy, který kontroluje shodu IP adresy s nastavenou MAC adresou.

Řazení nových pravidel

Pravidla v novém firewallu budou seřazena jinak, než ve stávajícím. Tento krok je reakce na druhý a třetí bod závěru analýzy. Reakcí na druhý bod je přesunutí pravidla, které povoluje ustanovené a související spojení (established a related), z jednotlivých uživatelských řetězců na první pozici v základních řetězcích FORWARD a INPUT.

Pokud je paket, který přichází do firewallu nový, je zařazen do nějakého uživatelského řetězce, kde je dále vyhodnocen buď jako povolená nebo nepovolená komunikace. V případě, že je paket vyhodnocen jako povolený, zapíše si firewall informaci o povolení komunikace do tabulky Conntrack, kterou lze na platformě MikroTik najít v sekci Firewall a v záložce Connection. Další pakety patřící do komunikace, která má již záznam v tabulce Conntrack, jsou v ustanoveném spojení a shodují se právě s pravidlem povolujícím ustanovené spojení. Těchto paketů je tedy velice mnoho a je vhodné co možná nejméně zatěžovat procesor firewallu jejich filtrováním, což právě přesunutí pravidla na první pozici řeší.

Přesunutím pravidla povolující ustanovené a související spojení na začátek řetězců FORWARD a INPUT navíc vzniká další optimalizace. Ve stávajícím firewallu se nacházely řetězce, které řešily, jaká komunikace je povolena z Internetu do jednotlivých VLAN. Prakticky jedinými pravidly v těchto řetězcích byla pravidla povolující komunikaci v ustanoveném a souvisejícím spojení. Tyto řetězce se tedy v novém firewallu nacházet nebudou a není třeba se jimi dále zabývat.

Tam, kde to neovlivní funkcionalitu pravidel, tedy tam, kde se filtrují pravidla podle stejných parametrů, budou pravidla seřazena podle počtu shod s pakety. Informace o počtu shod byly sledovány od 12:50 h 23. listopadu 2015 do 18:15 h 2. prosince 2015. Pravidla s nejvyšším počtem shod s pakety budou přesunuta na vrchol řetězce, protože pravidla se procházejí sekvenčně a čím výše pravidlo s častými shodami je, tím nižší je zatížení procesoru firewallu. Ovšem není vhodné seřadit pravidla pouze podle počtu shod. Při řazení pravidel bude přihlíženo i k logickému seskupení pravidel podle toho, jaké služby obsluhují. Není vhodné mít pravidla, která spolu logicky souvisí, na různých pozicích, protože by byla administrace takového firewallu nepřehledná a složitě by se vyhledávaly a řešily problémy.

Redukce pravidel

První redukcí, která bude provedena, je již zmiňované přesunutí pravidla povolujícího ustanovené a související spojení na začátek řetězců FORWARD a INPUT. Další redukce pravidel je reakcí na čtvrtý bod závěru analýzy. V původním firewallu se nacházela pravidla, která filtrovala pakety z nebo případně do uzlu s IP adresou 10.0.99.100. Po odzkoušení v laboratoři bylo zjištěno, že v síti již žádný uzel s touto IP adresou neexistuje, a proto mohou být všechna pravidla související s tímto uzlem v novém firewallu úplně vypuštěna.

Po konzultaci se správcem síťové laboratoře bylo dohodnuto, že nový firewall nebude obsahovat povolení komunikace se službou AVG, jelikož se jedná o již nepoužívanou službu. Proto budou oproti stávajícímu firewallu vypuštěna veškerá pravidla

dla týkající se právě komunikace mezi AVG klientem z VLAN 10 a AVG správcem na Winserveru.

Další redukce se bude týkat komunikace pomocí FTP protokolu. V původním firewallu se nacházela vždy dvojice pravidel, která povolovala FTP komunikaci. Pravidla vždy povolovala novou komunikaci směřující na port 21/tcp a 20/tcp. Po otestování funkčnosti je navrženo v novém firewallu vynechat pravidlo, které povoluje novou komunikaci na port 20/tcp.

Pravidlo již nebude potřeba, protože, jak uvádí Kurose a Ross (2014), první paket týkající se FTP směřuje na port 21/tcp. Přes tento port se provádí řízení spojení. V pasivním režimu, který je v síťové laboratoři ÚI PEF MENDELU využíván, je následně z portu 21/tcp FTP serveru odeslána odpověď klientovi s číslem portu, který je pro samotný přenos dat otevřen. Aby firewall věděl, které pakety přenáší data, využije FTP conntrack, který rozezná odpověď z FTP serveru s číslem portu, a poté označí pakety s daty jako související (related) spojení, které je firewallem povoleno v pravidle pro ustanovená a související spojení.

Další ochranné mechanismy

V reakci na čtvrtý bod závěru analýzy budou přidány ochranné mechanismy proti útokům záplavou paketů a ochrana proti s těmito útoky souvisejícím skenování portů.

V publikaci od Juniper Networks (2012) je popsáno mnoho síťových útoků. Ne všechny mechanismy fungující v prvcích z dílny Juniper Networks je MikroTik schopn aplikovat. Z publikace byly vybrány pouze některé užitečné ochranné mechanismy proti častým síťovým útokům, které jsou aplikovatelné v zařízeních MikroTik.

Skenování portů

Jako první byl vybrán útok zvaný TCP Port scan. Jak Juniper Networks (2015) vysvětluje, jde o TCP SYN segmenty, které jsou odesílány z jedné zdrojové IP adresy na různé porty jedné cílové IP adresy s tím, že útočník doufá, že některý port odpoví. Přes tento port bude následně veden útok.

MikroTik (2015) poskytuje návod, jak se tomuto útoku bránit. Jak MikroTik (2015) uvádí, je nutné přesunout IP adresu útočníka do speciálního seznamu tzv. Address List, a poté pakety se zdrojovou IP adresou shodnou s některou v seznamu blokovat. MikroTik umožňuje filtrovat pakety na základě parametru psd, který, jak popisuje MikroTik (2015), slouží k detekci TCP a UDP skenování portů. V dokumentaci MikroTik (2015) se lze navíc dočíst, jak nastavit filtrování podle tohoto parametru. Nastavení tohoto pravidla je vidět v Tab. 23.

Pokud tedy bude IP adresa z vstupního rozhraní ether2 detekována jako IP adresa útočníka, bude zařazena do Address Listu nazvaného BlackList a zůstane v tomto seznamu 30 minut.

Útok záplavou paketů

Další možné útoky jsou podle Juniper Networks (2012) zaplavení TCP SYN, UDP nebo ICMP pakety. Jak uvádí Juniper Networks (2012), cílem těchto útoků je zaplnění tabulky relací (Conntrack), čímž se firewall prakticky vyřadí z provozu, protože již není schopný udržovat informace o legitimním spojení.

MikroTik (2015) i v tomto případě poskytuje rady, jak se bránit. První varianta je stejně jako v případě skenování portů, odstavit útočnicka přesunutím do Address Listu BlackList. MikroTik (2015) uvádí, že na základě parametru limit lze zjistit, kolik spojení je aktuálně vytvořeno z jedné IP adresy. Aby bylo možné filtrovat podle parametru limit, je nutné mu nastavit dvě hodnoty. První hodnota představuje počet spojení, druhá síťovou masku. Pokud je síťová maska nastavena na číslo 32, znamená to v případě IPv4 počet spojení z jedné konkrétní IP adresy. Pravidlo je ukázáno v Tab. 23.

Druhá varianta, kterou MikroTik (2015) uvádí, je použití SYN Cookie technologie. Kurose a Ross (2014) uvádějí, že SYN Cookie je technologie, která pomocí hashovací funkce vytvoří zašifrované počáteční číslo TCP spojení. V tabulce relací se tedy neudrží informace o napůl otevřeném spojení. Kurose a Ross (2015) dále popisují, že pokud se jedná o legitimní spojení, vrátí klient segment ACK, který bude shodný s hodnotou cookie plus jedna. Lze tak jednoduše zamezit útokům TCP SYN Flood.

Nový firewall bude využívat kombinaci těchto dvou technologií, protože při zkoušce funkčnosti SYN Cookie bylo zjištěno, že výpočet je mírně náročný, což by při velkém počtu SYN segmentů v jeden okamžik vedlo k většímu zatížení Labrouteru.

Do Access Listu BlackList budou zařazeny i IP adresy těch, kteří se budou snažit o UDP Flood, což je zaplnění tabulky relací UDP pakety. Detekce tohoto útoku se provádí také pomocí parametru limit, stejně jako první varianta ochrany proti TCP SYN Floodu.

Pravidlo, které zajistí, že budou pakety od útočníků zahazovány, bude přidáno do uživatelského řetězce SpatnePakety. Po 30 minutách od přidání zdrojové IP adresy do seznamu BlackList je záznam vymazán a pakety jsou znovu propouštěny. Konkrétní parametry pravidla jsou uvedeny v Tab. 23.

Do nového firewallu bude zavedena také ochrana proti Ping Flood, což je vlastně zaplavení oběti ICMP Echo request datagramy. Pro ochranu proti Ping Flood bude do nového firewallu zařazeno pravidlo pro protokol ICMP s parametrem limit, které po naplnění limitu shod nepovolí další průchod paketů. Nastavení limitu bude povolení průměrně jednoho paketu za sekundu a povolení maximálně 5 paketů v jedné dávce. Po naplnění limitu bude již paket s ICMP volbou Echo Request zahazován. Celou kontrolu limitů pro ICMP datagramy budou uzavřeny do uživatelského řetězce nazvaného PingLimit.

Další doporučené ochranné mechanismy

Juniper Networks (2012) dále navrhuje několik pravidel, které budou aplikovány v novém firewallu. Přidáno bude pravidlo pro zahazování paketů s cílovým portem 139/udp. Dále budou přidána pravidla, která zahazují pakety s IPv4 volbou loose-source-routing a strict-source-routing, protože, jak uvádí Juniper Networks (2012), tato volba umožňuje falšovat pravou zdrojovou IP adresu. A jako poslední radí Juniper Networks (2012), přidat pravidlo pro zahazování fragmentovaných SYN paketů, protože tyto pakety bývají často malé a jejich fragmentace je tudíž podezřelá. Nastavení pravidel je zobrazeno v Tab. 24.

Tabulka 23: Ochranná pravidla proti skenování portů a útokům záplavou paketů

Akce	Vstup. r.	Protokol/Flag	Limit	Poznámka
Ochrana proti skenování portů přidáním IP adresy útočnicka na 30 min do BlackListu.				
IP do BlackList -- 30 min	řer2	TCP	300 spojení/ 1 IP	psd=21,3s,3,1
Ochrana proti SYN Floodu přidáním IP adresy útočnicka na 30 min do BlackListu.				
IP do BlackList -- 30 min	řer2	TCP/SYN	300 spojení/ 1 IP	*
Ochrana proti UDP Floodu přidáním IP adresy útočnicka na 30 min do BlackListu.				
IP do BlackList -- 30 min	řer2	UDP	300 spojení/ 1 IP	*
Ochrana proti Ping Floodu. Níže je ukázán řetězec PingLimit do něhož jsou předávány ICMP pakety.				
ACCEPT	*	1 paket/s, v dávce 5	*	v PingLimit
DROP	*	*	*	v PingLimit

Tabulka 24: Přidaná doporučená ochranná pravidla

Cíl	Protokol/Flag	Cílový port	Poznámka
Zahození paketů směřující na port 139/udp.			
LogDrop_SpatnePakety	UDP	139	*
Zahození fragmentovaných TCP-SYN paketů.			
LogDrop_SpatnePakety	TCP/SYN	*	fragmentovaný
Zahození paketů s nastavenou volbou loose-source-routing a strict-source-routing.			
LogDrop_SpatnePakety	*	*	loose-source-routing
LogDrop_SpatnePakety	*	*	strict-source-routing
Zahození paketů, jejichž zdrojová IP adresa je shodná s IP adresou v BlackList.			
LogDrop_SpatnePakety	*	*	Shoda s IP v BlackList

NAT

Jak uvádí MiktoTik (2015) v oficiální dokumentaci, NAT (Network Address Translation) je standard, který povoluje používat jiné IP adresy pro vnitřní a vnější komunikaci. V případě síťové laboratoře se NAT využívá pro změnu IP adres z vnitřní sítě za IP adresu WAN rozhraní (ether2). Ideální řešení pro tento problém nabízí RouterOS v podobě přidání pravidla masquerade do tabulky NAT, které změní IP adresu všech paketů odcházejících z rozhraní ether2 za IP adresu toho rozhraní (195.178.72.189/28).

Pomocí jedné IP adresy tak mohou všechna zařízení z laboratoře komunikovat ven z vnitřní sítě.

Způsob migrace

Migrovat firewall lze v zásadě dvěma způsoby. Pokud je pravidel mnoho, vyplatí se konvertovat data pomocí skriptu. Pokud je pravidel málo a je rychlejší pravidla přepsat ručně, vyplatí se provést konverzi ručně. Protože stávající firewall obsahuje přibližně 300 pravidel a parametry, podle kterých se pakety filtrují, jsou často rozdílné, rozhodl se autor práce provést konverzi ručně.

Zavedení firewallu

Pro zavedení pravidel do firewallu bylo zvoleno vytvoření skriptu, který všechna pravidla zavede pomocí SSH spojení. Skript lze spouštět z linuxového operačního systému. Pro zavedení pravidel je třeba administrátorské heslo firewallu. Ve skriptu jsou jednotlivá pravidla uložena jako řetězec příkazů v syntaxi RouterOS do jedné proměnné. Příkaz v proměnné je na konci skriptu vykonán. Zdrojový kód skriptu je uveden v příloze F.

6.7 Skripty pro administraci vnějšího rozhraní firewallu

Správce síťové laboratoře požadoval několik skriptů pro management Labrouteru. Prvními dvěma jsou skripty, které budou povolovat, respektive zakazovat připojení celé laboratoře do MENDELU LAN a odtud dál do Internetu. Další skript bude sloužit ke zjištění, zda je momentálně síťová laboratoř připojena do MENDELU LAN nebo ne.

Způsob spouštění

Skripty jsou koncipovány tak, aby je bylo možné jednoduše spouštět z ovládacích počítačů ve VLAN 99 (Správce 01 nebo Správce 02). Skripty budou využívat SSH spojení s Labrouterem kvůli šifrované komunikaci.

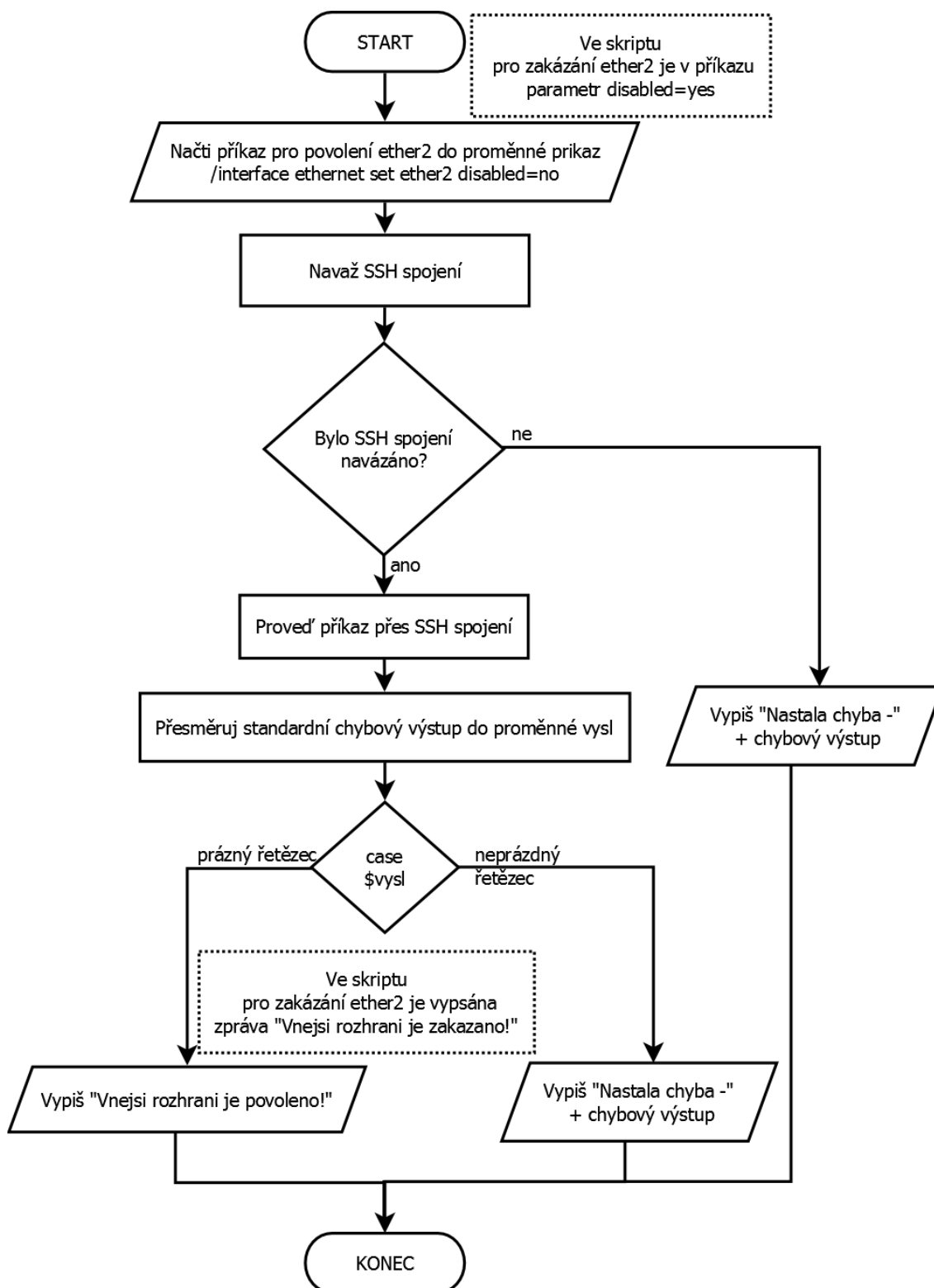
SSH autentizaci lze řešit dvojím způsobem. Lze se autentizovat pomocí hesla nebo pomocí DSA klíčů.

Nyní bude popsána autentizace pomocí DSA klíčů. Jak uvádí MikroTik (2015), díky klíčům lze jednoduše spouštět skripty bez nutnosti zadávat heslo.

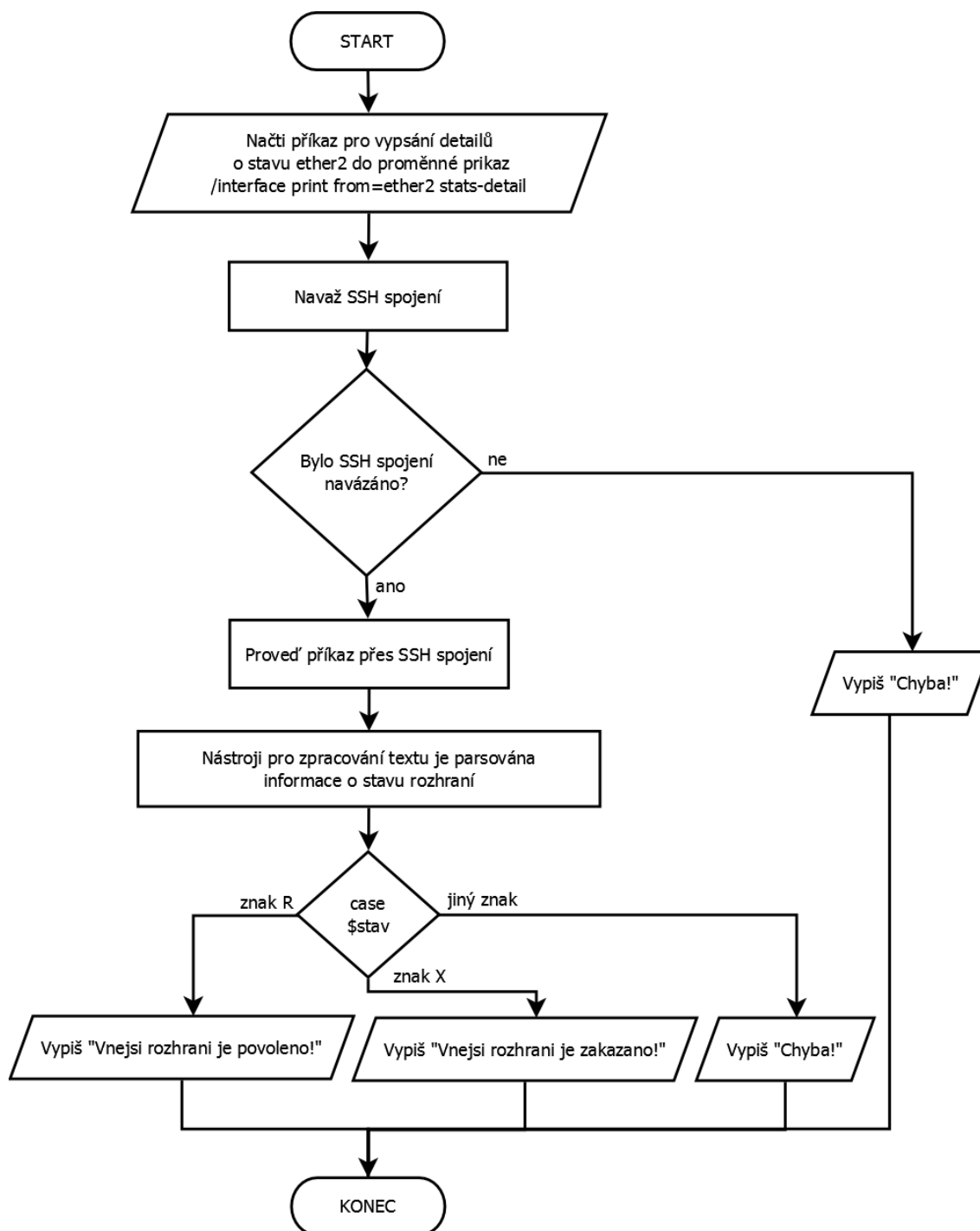
MikroTik (2015) poskytuje návod, co je nutné vykonat na klientovi i na SSH serveru. Jako první je nutné vygenerovat veřejný a privátní klíč na klientovi. Poté je nutné přesunout veřejný klíč klienta na SSH server, což je v tomto případě Labrouter. Pokud není zadána kontrolní fráze, je možné spouštět skripty jednoduše bez zadávání jakýchkoliv hesel. Je zřejmé, že tento typ připojení na server je možný pouze z klienta, který má svůj veřejný klíč uložen v Labrouteru.

Algoritmy skriptů

Na následujících Obr. 13 a Obr. 14 jsou nakresleny vývojové diagramy skriptů pro správu vnějších rozhraní. Jak je možné na Obr. 13 a Obr. 14 vidět, skripty uživateli zobrazují informaci o tom, zda byl příkaz proveden v pořádku nebo zda došlo k chybě. Pokud dojde k nějaké chybě, je uživateli zobrazena chybová hláška se specifikací chyby.



Obrázek 13: Skript pro povolení/zakázání vnějšího rozhraní



Obrázek 14: Skript pro zjištění stavu vnějšího rozhraní

7 Řešení

V této kapitole bude prezentováno výsledné řešení, provedené na základě návrhu v předchozí kapitole.

7.1 Fyzické zapojení

Fyzické zapojení odpovídá nákresu topologie na Obr. 12 uvedeného v kapitole Návrh řešení. Do rozhraní ether1 je zapojena komunikace z vnitřní sítě. Do rozhraní ether2 směřuje komunikace z Internetu.

Rozdíl mezi původním a novým labrouterem je vidět na Obr. 15.



Obrázek 15: Fyzické zapojení v síťové laboratoři ÚI PEF MENDELU

7.2 Základní nastavení labrouteru MikroTik

Nastavení lze provádět dvěma způsoby a to pomocí dialogových oken v programu Winbox nebo v terminálu. Dále budou uvedeny pouze příkazy v terminálu, neboť ty lze přenést pomocí skriptu na jiné zařízení MikroTik, a tím celý směrovač nastavit.

Název zařízení

Název zařízení se nastavuje v sekci System a podsekci Identity. Název je Labrouter. Výsledek změny názvu je vidět na Obr. 16.

- `/system identity set name=Labrouter`

```

Terminal
MMM      MMM      KKK      TTTTTTTTTT      KKK
MMMM     MMMM     KKK      TTTTTTTTTT      KKK
MMM MMMM MMM III KKK KKK RRRRRR 000000 TTT III KKK KKK
MMM MM  MMM III KKKKK RRR RRR 000 000 TTT III KKKKK
MMM     MMM III KKK KKK RRRRRR 000 000 TTT III KKK KKK
MMM     MMM III KKK KKK RRR RRR 000000 TTT III KKK KKK

MikroTik RouterOS 6.32.3 (c) 1999-2015      http://www.mikrotik.com/

[?]          Gives the list of available commands
command [?]  Gives help on the command and list of arguments

[Tab]       Completes the command/word. If the input is ambiguous,
            a second [Tab] gives possible options

/           Move up to base level
..         Move up one level
/command   Use command at the base level
[admin@MikroTik] > /system identity set name=Labrouter
[admin@Labrouter] >

```

Obrázek 16: Přiřazení názvu směrovači

Přístupová práva

Účty a příslušné skupiny se nastavují a vytváří v sekci User. Nejprve je nutné vytvořit účtu admin heslo a povolený rozsah IP adres, z kterých se lze pod tímto účtem přihlásit. Jak bylo zmíněno výše v kapitole Návrh řešení, heslo je vytvářeno přímo správcem síťové laboratoře. Uveden je tedy pouze příkaz, kterým se heslo nastaví bez samotného hesla. Povolené IP adresy jsou všechny IP adresy z rozsahu 10.0.99.0/24, což odpovídá virtuální lokální síti č. 99 (VLAN 99).

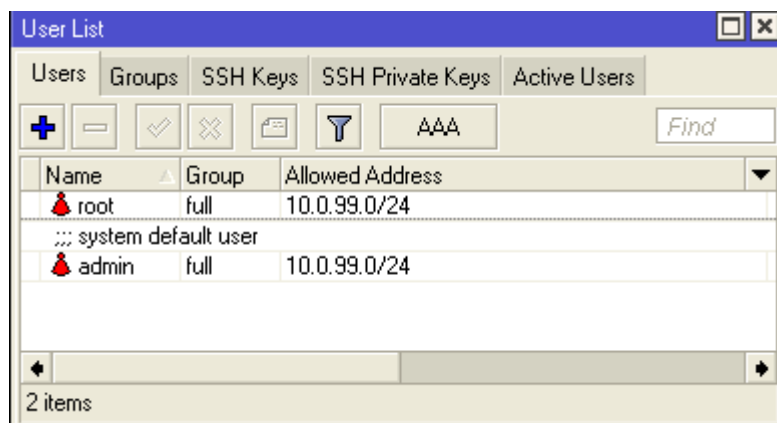
Protože účet admin je implicitně ve skupině Full, není nutné skupinu nastavovat.

- `/user set admin password=***** address=10.0.99.0/24`

Dále je vytvořen účet root. Účet root je přidán do skupiny Full, a jsou mu přiděleny IP adresy z rozsahu 10.0.99.0/24 stejně jako v případě účtu admin. Tomuto účtu je také vytvořeno heslo správcem laboratoře.

- `/user add group=full name=root address=10.0.99.0/24 password=*****`

Oba vytvořené účty jsou vidět na Obr. 17.



Obrázek 17: Vytvořené účty

Synchronizace času

V sekci System a podsekci Clock se nastaví automatická detekce časového pásma. To zajistí, že Labrouter sám zjistí aktuální časové pásmo.

- `/system clock set time-zone-autodetect=yes`

Pro synchronizaci času je využít NTP server na linuxovém serveru s IP adresou 10.0.30.2. Aby mohl Labrouter pracovat jako NTP klient, musí mu být přidán balíček ntp do sekce System a podsekcce Packages. Tyto balíčky jsou zdarma ke stažení z oficiální webové stránky <http://www.mikrotik.com/download>. Po nainstalování balíčku a restartu Labrouteru lze provádět synchronizaci času s linuxovým serverem.

- `/system ntp client set primary-ntp=10.0.30.2 enabled=yes`

Odesílání logovacích záznamů

Pro nastavení odesílání logů na centrální Syslog server (Správce 01) je nutné nejdříve vytvořit akci, která bude určovat, jak logy spravovat. Vytvoření této akce se provádí v sekci System, v podsekci Logging a v sekci třetí úrovně s názvem Action. Vytvoření akce vyžaduje přidělení názvu. Dále je přidělen cíl, který určuje, co se s logovacím záznamem stane. Pro přeposílání na Syslog server je cílem remote. Posledními parametry jsou cílový port a cílová IP adresa. Standardní port na serveru pro příjem logů je 514/udp.

- `/system logging action add name=ToSyslog target=remote remote-port=514 remote=10.0.99.101`

Když je akce vytvořena, je přidělena určitému typu logovacích záznamů tzv. topics. Typ logu určuje, čeho se logy týkají. Jak popisuje MikroTik (2015), pro odeslání logů, které byly označeny ve firewallu akcí log, je potřebné vybrat typ firewall.

- `/system logging add topics=firewall action=ToSyslog`

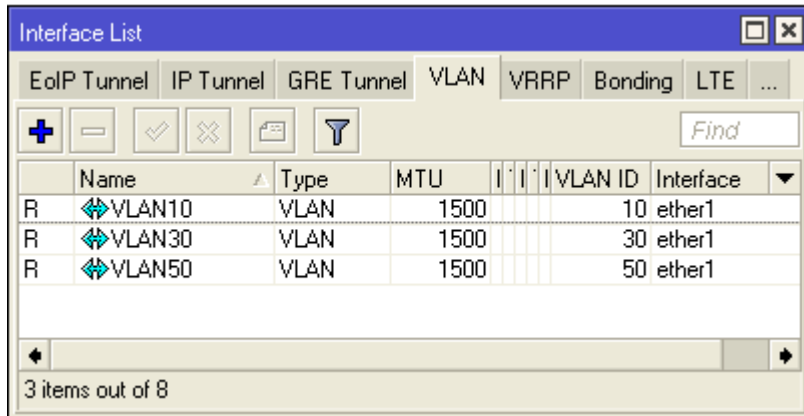
7.3 Nastavení rozhraní

Cílem toho kroku je připravit rozhraní routeru do stavu, aby byl schopen forwardovat pakety. Forwardování je nezbytné mezi jednotlivými VLAN – pro vnitřní komunikaci i mezi oběma fyzickými rozhraními – pro přístup k Internetu.

Vytvoření VLAN

Vytvoření VLAN probíhá v sekci Interface v podsekcí VLAN. Při vytváření VLAN je nastaveno rozhraní ether1. Na toto rozhraní bude VLAN navázána. Dále je nutné VLAN pojmenovat a zadat jim VLAN-ID, které je přidáváno tagovaným paketům. VLAN se budou jmenovat podle čísla VLAN-ID. Výsledek příkazů je zobrazen na Obr. 18.

- `/interface vlan add interface=ether1 name=VLAN10 vlan-id=10 mtu=1500`
- `/interface vlan add interface=ether1 name=VLAN30 vlan-id=30 mtu=1500`
- `/interface vlan add interface=ether1 name=VLAN50 vlan-id=50 mtu=1500`



	Name	Type	MTU	VLAN ID	Interface
R	VLAN10	VLAN	1500	10	ether1
R	VLAN30	VLAN	1500	30	ether1
R	VLAN50	VLAN	1500	50	ether1

3 items out of 8

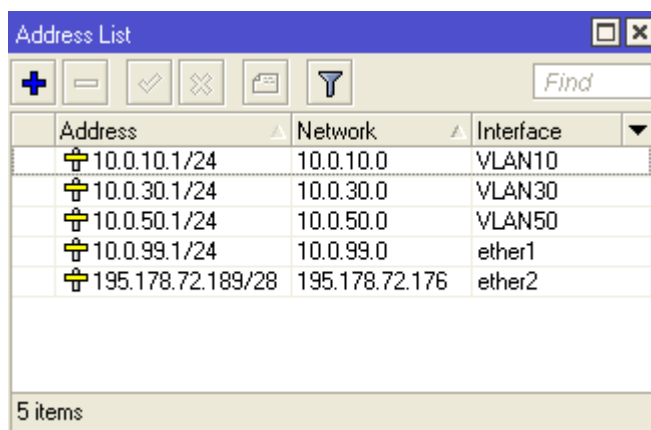
Obrázek 18: Vytvořené virtuální LAN

Přidělení IP adres

Když jsou již VLAN vytvořené, mohou jim být přiděleny IP adresy. Kromě VLAN budou přidány i IP adresy fyzickým rozhraním ether1 a ether2. Funkce nativní VLAN je v MikroTiku přiřazena přímo jako IP adresa fyzického rozhraní ether1. Rozhraní ether2 má funkci WAN portu, a proto je jemu nastavená IP adresa z rozsahu IP adres v MENDELU LAN a je veřejná.

IP adresy se nastavují v sekci IP a podsekci Address. Při nastavení IP adresy je nutné specifikovat IP adresu, síťovou masku ve tvaru prefixu, rozhraní a IP adresu sítě. Nastavení všech IP adres jsou viditelné na Obr. 19.

- `/ip address add address=10.0.99.1/24 interface=ether1 network=10.0.99.0`
- `/ip address add address=10.0.10.1/24 interface=VLAN10 network=10.0.10.0`
- `/ip address add address=10.0.30.1/24 interface=VLAN30 network=10.0.30.0`
- `/ip address add address=10.0.50.1/24 interface=VLAN50 network=10.0.50.0`
- `/ip address add address=195.178.72.189/28 interface=ether2 network=195.178.72.176`



Address	Network	Interface
10.0.10.1/24	10.0.10.0	VLAN10
10.0.30.1/24	10.0.30.0	VLAN30
10.0.50.1/24	10.0.50.0	VLAN50
10.0.99.1/24	10.0.99.0	ether1
195.178.72.189/28	195.178.72.176	ether2

5 items

Obrázek 19: Výpis IP adres

7.4 Směrovací tabulka

Jak bylo již uvedeno v kapitole Návrh řešení, většina záznamů ve směrovací tabulce se vytvoří automaticky po přidání IP adresy rozhraní. Záznam, který je nezbytné přidat, je defaultní brána (default gateway). Přes tuto bránu jsou forwardovány datagramy, pro které není záznam s nižším počtem skoků. Jde o datagramy, které směřují do vnější sítě.

Přidání záznamu do směrovací tabulky se provádí v sekci IP a podsekci Route. IP adresa defaultní brány bude IP adresa rozhraní ether2. Celá směrovací tabulka je vypsána na Obr. 20.

- `/ip route add distance=1 gateway=195.178.72.177`

	Dst. Address	Gateway	Distance	Pref. Source
AS	0.0.0.0/0	195.178.72.177 reachable ether2	1	
DAC	10.0.10.0/24	VLAN10 reachable	0	10.0.10.1
DAC	10.0.30.0/24	VLAN30 reachable	0	10.0.30.1
DAC	10.0.50.0/24	VLAN50 reachable	0	10.0.50.1
DAC	10.0.99.0/24	ether1 reachable	0	10.0.99.1
DAC	195.178.72.176/28	ether2 reachable	0	195.178.72.189

6 items

Obrázek 20: Směrovací tabulka

7.5 DHCP Relay

V kapitole Návrh řešení bylo zmíněno, že je nutné nastavit na MikroTiku technologii DHCP Relay. Tato technologie se nastavuje v sekci IP a podsekci DHCP Relay. Při nastavení je nutné určit reálný DHCP server, což je v případě síťové laboratoře ÚI PEF MENDELU linuxový server (10.0.30.2). Dále je třeba nastavit lokální IP adresu, rozhraní a název, který slouží především pro orientaci administrátora.

- `/ip dhcp-relay add dhcp-server=10.0.30.2 interface=VLAN10 local-address=10.0.10.1 name=DHCP_Relay_VLAN10 disabled=no`

7.6 Firewallová pravidla

Způsob přidání pravidel do nového labrouteru byl popsán v kapitole Návrh řešení. Samotná firewallová pravidla jsou podobná pravidlům v původním firewallu. Vytváření pravidel se provádí v sekci IP, v podsekci Firewall a v sekci třetí úrovně Filter.

Pro ukázkou jsou uvedeny příkazy pro vložení pravidel do základních řetězců OUTPUT, INPUT a FORWARD. Kvůli velkému množství pravidel budou příkazy, které jednotlivá pravidla přidávají do uživatelských řetězců, uvedeny až v příloze této práce.

Pro vytvoření pravidel se v základních řetězcích používají následující parametry:

- `chain` – řetězec, do kterého má být pravidlo přidáno;
- `protocol` – protokol paketu (icmp, tcp, udp);
- `connection-state` – stav paketu (new, established, related);
- `action` – akce udávající, co se má s paketem stát (jump, drop, log);

- jump-target – jméno uživatelského řetězce, kterému je paket předán;
- in-interface – rozhraní, z kterého paket přichází;
- out-interface – rozhraní, na jaké paket směřuje;
- src-address – IP adresa, z které paket přišel;
- dst-address – IP adresa, na který je paket odeslán.

OUTPUT

#####

```
#Povolení spojení, které iniciuje Labrouter
/ip firewall filter add chain=output;
```

INPUT

#####

```
#Povolení spojení ve stavu established nebo related
/ip firewall filter add chain=input protocol=!icmp connection-state=established,related;

#Ochrana Labrouteru před Ping Flood
/ip firewall filter add chain=input action=jump jump-target=PingLimit protocol=icmp
connection-state=established,related;

#Povolení všech paketů přicházejících z loopbacku
/ip firewall filter add chain=input in-interface=loopback0 action=accept;

#Kontrola paketů z vnějšího rozhraní na privátní IP adresu
/ip firewall filter add chain=input action=jump in-interface=ether2 jump-target=RFC1918_IN;

#Předání řetězci SpatnePakety, který zahodí vadné pakety
/ip firewall filter add chain=input action=jump jump-target=SpatnePakety;

#Povolení přístupu z S01 na Labrouter přes SSH spojení
/ip firewall filter add chain=input dst-address=10.0.99.1 dst-port=22 in-interface=ether1
protocol=tcp src-address=10.0.99.101 connection-state=new;

#Povolení přístupu z S02 na Labrouter přes SSH spojení
/ip firewall filter add chain=input dst-address=10.0.99.1 dst-port=22 in-interface=ether1
protocol=tcp src-address=10.0.99.102 connection-state=new;

#Povolení přístupu z S01 na Labrouter přes Winbox
/ip firewall filter add chain=input dst-address=10.0.99.1 dst-port=8291 in-interface=ether1
protocol=tcp src-address=10.0.99.101 connection-state=new;

#Povolení pingu z vnějšího rozhraní na Labrouter
/ip firewall filter add chain=input icmp-options=8 in-interface=ether2 protocol=icmp;

#Povolení pingu z vnitřní sítě na rozhraní VLAN10
/ip firewall filter add chain=input dst-address=10.0.10.1 icmp-options=8
in-interface=VLAN10 protocol=icmp src-address=10.0.10.0/24;

#Povolení pingu z vnitřní sítě na rozhraní VLAN30
/ip firewall filter add chain=input dst-address=10.0.30.1 icmp-options=8
in-interface=VLAN30 protocol=icmp src-address=10.0.30.0/24;

#Povolení pingu z vnitřní sítě na rozhraní VLAN50
/ip firewall filter add chain=input dst-address=10.0.50.1 icmp-options=8
in-interface=VLAN50 protocol=icmp src-address=10.0.50.0/24;

#Povolení pingu z vnitřní sítě na rozhraní ether1 (VLAN99)
/ip firewall filter add chain=input dst-address=10.0.99.1 icmp-options=8
in-interface=ether1 protocol=icmp src-address=10.0.99.0/24;
```

```
#Povolení DHCP protokolu kvůli správné funkci DHCP Relay
/ip firewall filter add chain=input in-interface=VLAN10 dst-port=67 protocol=udp
comment="\kvuli DHCP Relay\" connection-state=new;

/ip firewall filter add chain=input in-interface=VLAN30 dst-port=67 protocol=udp
comment="\kvuli DHCP Relay\" connection-state=new;

#Zahození všech paketů, které nebyly povoleny předchozími pravidly
/ip firewall filter add chain=input action=jump jump-target=LogDrop_INPUT;

FORWARD
#####
#Povolení spojení ve stavu established nebo related
/ip firewall filter add chain=forward protocol=!icmp connection-state=established,related;

#Ochrana vnitřní sítě před Ping Flood
/ip firewall filter add chain=forward action=jump jump-target=PingLimit protocol=icmp
connection-state=established,related;

#Kontrola paketů z vnějšího rozhraní na privátní IP adresu
/ip firewall filter add chain=forward action=jump in-interface=ether2 jump-target=RFC1918_IN;

#Předání řetězci SpatnePakety, který zahodí vadné pakety
/ip firewall filter add chain=forward action=jump jump-target=SpatnePakety;

#Předání řetězci IpAdresy na kontrolu shody MAC adres s IP adresou
/ip firewall filter add chain=forward action=jump jump-target=IpAdresy;

#Předání paketu odeslaného z VLAN10 do Internetu řetězci Vlan10ToInternet
/ip firewall filter add chain=forward action=jump in-interface=VLAN10 jump-target=Vlan10ToInternet
out-interface=ether2 src-address=10.0.10.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN50 do Internetu řetězci Vlan50ToInternet
/ip firewall filter add chain=forward action=jump in-interface=VLAN50 jump-target=Vlan50ToInternet
out-interface=ether2 src-address=10.0.50.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN10 do VLAN30 řetězci Vlan10ToVlan30
/ip firewall filter add chain=forward action=jump dst-address=10.0.30.0/24 in-interface=VLAN10
jump-target=Vlan10ToVlan30 out-interface=VLAN30 src-address=10.0.10.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN30 do VLAN10 řetězci Vlan30ToVlan10
/ip firewall filter add chain=forward action=jump dst-address=10.0.10.0/24 in-interface=VLAN30
jump-target=Vlan30ToVlan10 out-interface=VLAN10 src-address=10.0.30.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN30 do VLAN99 řetězci Vlan30ToVlan99
/ip firewall filter add chain=forward action=jump dst-address=10.0.99.0/24 in-interface=VLAN30
jump-target=Vlan30ToVlan99 out-interface=ether1 src-address=10.0.30.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN99 do VLAN30 řetězci Vlan99ToVlan30
/ip firewall filter add chain=forward action=jump dst-address=10.0.30.0/24 in-interface=ether1
jump-target=Vlan99ToVlan30 out-interface=VLAN30 src-address=10.0.99.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN30 do Internetu řetězci Vlan30ToInternet
/ip firewall filter add chain=forward action=jump in-interface=VLAN30 jump-target=Vlan30ToInternet
out-interface=ether2 src-address=10.0.30.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN99 do Internetu řetězci Vlan99ToInternet
/ip firewall filter add chain=forward action=jump in-interface=ether1 jump-target=Vlan99ToInternet
out-interface=ether2 src-address=10.0.99.0/24 connection-state=new;

#Předání paketu odeslaného z VLAN99 do VLAN10 řetězci Vlan99ToVlan10
/ip firewall filter add chain=forward action=jump in-interface=ether1 jump-target=Vlan99ToVlan10
out-interface=VLAN10 connection-state=new;
```

```
#Předání paketu odeslaného z VLAN10 do VLAN99 řetězci Vlan10ToVlan99
/ip firewall filter add chain=forward action=jump in-interface=VLAN10 jump-target=Vlan10ToVlan99
out-interface=ether1 connection-state=new;

#Předání paketu odeslaného z VLAN99 do VLAN50 řetězci Vlan99ToVlan50
/ip firewall filter add chain=forward action=jump in-interface=ether1 jump-target=Vlan99ToVlan50
out-interface=VLAN50 connection-state=new;

#Zahození všech paketů, které nebyly povoleny předchozími pravidly
/ip firewall filter add chain=forward action=jump jump-target=LogDrop_FORWARD;
```

Přidané ochranné mechanismy

Přidané ochranné mechanismy slouží k zabránění útoku záplavou paketů (DoS) a skenování portů. Jak již bylo uvedeno v kapitole Návrh řešení, jedním z opatření obrany proti útoku TCP SYN Flood je použití technologie SYN Cookie. Zapnutí SYN Cookie se provádí v sekci IP a podsekcí Settings.

- `/ip settings set tcp-syncookies=yes`

Protože výpočet SYN Cookie je mírně náročný a řeší pouze TCP/SYN Flood útok, budou přidána ještě další firewallová pravidla. Tato pravidla zahrnují ochranu proti TCP/SYN Flood, UDP Flood a skenování portů. Struktura pravidel je znázorněna v předchozí kapitole v Tab. 23.

Níže uvedené příkazy přidávají do firewallu pravidla, která řeší ochranu proti útoku záplavou paketů. Pakety se filtrují podle parametru `connection-limit=300,32`, který přesune útočnickovu IP adresu do BlackListu. Přesunutí do BlackList se provádí příkazem `action=add-src-to-address-list` a `address-list=BlackList`.

- `/ip firewall filter add chain=SpatnePakety protocol=tcp
in-interface=ether2 tcp-flags=syn action=add-src-to-address-list
address-list=BlackList address-list-timeout=30m
connection-limit=300,32 comment="SYN Flooder"`
- `/ip firewall filter add chain=SpatnePakety protocol=udp
in-interface=ether2 action=add-src-to-address-list
address-list=BlackList address-list-timeout=30m
connection-limit=300,32 comment="UDP Flooder"`

Další příkaz přidá do firewallu pravidlo, které dokáže identifikovat pokus o skenování portů.

- `/ip firewall filter add chain=SpatnePakety protocol=tcp
in-interface=ether2 action=add-src-to-address-list
address-list=BlackList address-list-timeout=30m
psd=21,3s,3,1 comment="Port Scanner"`

Další ochranná pravidla zobrazená v Tab. 24 jsou přidána společně s firewallovými pravidly pomocí skriptu.

NAT

Nastavení NAT se provádí v sekci Ip, podsekci Firewall a v sekci třetí úrovně s názvem Nat. Je nutné přidat pravidlo do řetězce srcnat. Pravidlu je nastavena akce masquerade a výstupní rozhraní ether2, za jehož IP adresu budou lokální IP adresy měněny. Výsledek příkazu je vidět na Obr. 21.

- `/ip firewall nat add chain=srcnat action=masquerade out-interface=ether2`

#	Action	Chain	Out. Interface	Bytes	Packets
0	masquerade	srcnat	ether2	2070 B	15

Obrázek 21: Natovací tabulka

Zavedení firewallu

Firewallová pravidla jsou do Labrouteru zaváděna pomocí SSH spojení ze Správce 01. Jak bylo již uvedeno výše, pravidel je mnoho, a proto bude celý skript uveden pouze v příloze této práce. Pro demonstraci skriptu, jímž je zavádění řešeno, bude použito symbolické pojmenování pravidel.

Pravidla jsou nejdříve načtena do proměnné `prikazy` jako řetězec. Kód na konci skriptu provede odeslání firewallových pravidel do Labrouteru a v případě špatného zavedení vypíše chybovou hlášku.

```
#!/bin/bash
```

```
#načtení IP adres
```

```
#načtení MAC adres
```

```
prikazy="

#pravidlo OUTPUT

#pravidla INPUT

#pravidla FORWARD

#pravidla uživatelských řetězců

#pravidlo NAT

"

vysl=$((ssh root@10.0.99.1 "$prikazy") 2>&1);
echo $vysl;

case $vysl in
'' )
    echo "====="
    echo "|Firewall byl uspesne nasazen|"
    echo "====="; ;
*)
    echo "!!!!!!!!!!!!!!"
    echo "Nastala chyba - "$vysl;;

esac

sleep 1
```

7.7 Skripty pro administraci vnějšího rozhraní firewallu

Pro administraci vnějšího rozhraní byly autorem bashovské skripty. Aby bylo možné skripty spustit, je nutné jim přidělit oprávnění spouštět.

Skript pro povolení/zakázání vnějšího rozhraní

Vývojový diagram skriptů byl uveden v kapitole Návrh řešení na Obr. 13. Následující zdrojový kód řeší zakázání vnějšího rozhraní. Skript pro povolení vnějšího rozhraní

se liší v parametru `disabled=no` a při výpisu do terminálu se vypisuje `|Vnejsi rozhrani je zakazano!|`.

```
#!/bin/bash

prikaz="/interface ethernet set ether2 disabled=no;"

vysl=$((ssh root@10.0.99.1 "$prikaz") 2>&1);

case $vysl in
'')    echo "====="
        echo "|Vnejsi rozhrani je povoleno!|"
        echo "=====";;
*)     echo "!!!!!!!!!!!!!!"
        echo "Nastala chyba - "$vysl;;
esac
```

Skript pro zjištění stavu vnějšího rozhraní

Další skript, který byl zmíněn v kapitole Návrh řešení na Obr. 14, slouží ke zjištění stavu vnějšího rozhraní respektive ke zjištění, zda je síťová laboratoř připojena k Internetu.

```
#!/bin/bash

prikaz="/interface print from=ether2 stats-detail;"

stav=$(ssh root@10.0.99.1 "$prikaz" | grep name | sed 's/^[ 0 ]*//'
| cut -d ' ' -f 1)

case $stav in
R)  echo "====="
     echo "|Vnejsi rozhrani je povoleno!|"
     echo "====="
     ;;
X)  echo "====="
     echo "|Vnejsi rozhrani je zakazano!|"
     echo "====="
     ;;
*)  echo "Chyba!"
     exit
     ;;
esac
```


8 Testování

Po fyzické implementaci je nutné provést otestování dostupnosti služeb, zátěžový test, otestování ochrany proti síťovým útokům a ověření funkčnosti skriptů. Testování se řídí navrženým testovacím scénářem.

1. Komunikace mezi VLAN a Internetem
2. Komunikace mezi virtuálními LAN
3. Dostupnost služeb z VLAN 30 pro VLAN 10
4. Dostupnost služeb z VLAN 30 pro VLAN 99
5. Dostupnost služeb z Internetu pro VLAN 10
6. Dostupnost služeb z Internetu pro VLAN 30
7. Dostupnost služeb z Internetu pro VLAN 50
8. Dostupnost služeb z Internetu pro Správce 02 (VLAN 99)
9. Zátěžový test
10. Testování ochranných mechanismů
11. Testování skriptů

Před samotným testováním byly vynulovány počítače (counter) ve firewallu. Při testování bylo poté sledováno, zda je testovaná komunikace povolena správným pravidlem. Pokud byla komunikace úspěšná a pakety byly povoleny správným pravidlem, je takový stav v následujících tabulkách označen odškrtnutím (✓). V případě neúspěšné komunikace nebo pokud byly pakety povoleny nesprávným pravidlem, je takový stav označen křížkem (X).

8.1 Vzájemná komunikace

Testování vzájemné komunikace bylo provedeno pomocí nástroje ping. V Tab. 25 je vyhodnoceno, která komunikace byla úspěšná.

Tabulka 25: Vzájemná komunikace

Iniciátor spojení	Cílová síť	✓
VLAN 10	Internet	✓
VLAN 30	Internet	✓
VLAN 50	Internet	✓
VLAN 99	Internet	✓
VLAN 10	VLAN 30	✓
VLAN 99	VLAN 10	✓
VLAN 99	VLAN 30	✓

Z Tab. 25 je patrné, že komunikace byla úspěšná.

8.2 Dostupnost služeb z VLAN 30

Následné výsledky testování zobrazují, které služby jsou dostupné z VLAN 30 pro VLAN 10. Ve VLAN 30 se nachází dva virtualizované servery Linuxserver a Winserver, které poskytují klientům z VLAN 10 služby uvedené v Tab. 26.

Tabulka 26: Dostupnost služeb z VLAN 30 pro VLAN 10

z Linuxserveru	✓	z Winserveru	✓
DHCP	✓	MS Update	✓
DNS	✓		
NTP	✓		
Yum	✓		
LDAP	✓		
NFS	✓		
FTP	✓		
CUPS	X		

Počítadlo paketů u pravidla, které slouží k povolení CUPS, bylo navýšeno o jedna, ale protože se tiskový systém momentálně nevyužívá, komunikace nebyla úspěšná.

Pro VLAN 99 jsou k dispozici také služby na Linuxserveru i Winserveru. Mimo tyto dva je také nutné ovládat ESXi server, který vytváří prostředí pro virtuální servery.

Tabulka 27: Dostupnost služeb z VLAN 30 pro VLAN 99

z Linuxserveru	✓	z Winserveru	✓	z ESXi serveru	✓
DNS	✓	Rdesktop	✓	vSphere	✓
NTP	✓				
VPN	✓				
SSH	✓				
Yum	✓				
FTP	✓				
CUPS	X				

Stejně jako pro VLAN 10 je i pro VLAN 99 nedostupná služba CUPS.

8.3 Dostupnost služeb z Internetu pro vnitřní síť

V Tab. 28 je vyhodnocena dostupnost služeb z Internetu pro klienty ze síťové laboratoře. Sloupce Tab. 28 jsou rozděleny podle jednotlivých VLAN.

Tabulka 28: Dostupnost služeb z Internetu

VLAN 10	✓	VLAN 30	✓	VLAN 50	✓	Správce 02	✓
HTTP	✓	DNS	✓	HTTP	✓	HTTP	✓
HTTPS	✓	NTP	✓	HTTPS	✓	HTTPS	✓
SSH	✓	Yum	✓	SSH	✓	SSH	✓
		MS Update	✓	DNS	✓		

8.4 Zátěžový test

Zátěžový test byl proveden pomocí stahování ISO souboru z webové stránky <https://www.centos.org/download/>. Na této webové stránce byl vybrán mirror z Fakulty informačních studií VUT v Brně http://merlin.fit.vutbr.cz/mirrors/centos/7/isos/x86_64/CentOS-7-x86_64-Everything-1511.iso. Simultánně byla sledována aktuální přenosová rychlost a vytížení CPU labrouteru při různém počtu stahujících klientů. Testování probíhalo v následujícím pořadí:

1. Stahování ISO souboru z jednoho počítače,
2. Stahování ISO souboru z dvou počítačů zároveň,
3. Stahování ISO souboru ze tří počítačů zároveň,
4. Stahování ISO souboru ze čtyř počítačů zároveň,
5. Stahování ISO souboru z pěti počítačů zároveň,
6. Stahování ISO souboru z deseti počítačů zároveň.

Naměřené hodnoty jsou uvedeny v Tab. 29.

Tabulka 29: Rychlost stahování dat z Internetu

Počet stahujících počítačů	Přenosová rychlost (stahování)	Využití CPU pro firewall	Využití CPU pro směrování
1	88,80 Mb/s	0 %	7 %
2	42,56 Mb/s	0 %	18 %
3	38,48 Mb/s	1 %	24 %
4	19,20 Mb/s	1 %	39 %
5	17,04 Mb/s	1 %	48 %
10	8,32 Mb/s	2 %	65 %

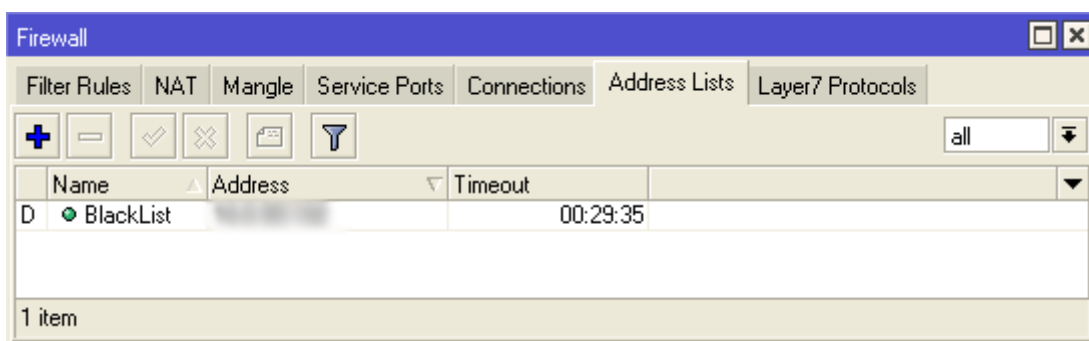
Z Tab. 29 je patrné, že při vyšším počtu stahujících počítačů najednou, razantně klesá přenosová rychlost. Využití CPU pro směrování dosáhlo v případě 10 klientů až na 65 %. Filtrování paketů ve firewallu nemělo na přenosovou rychlost žádný vliv.

8.5 Testování ochranných mechanismů

Při testování skenování portů byly pakety určené ke skenování úspěšně odchyčeny příslušným pravidlem, což způsobilo, že IP adresa útočníka byla umístěna do seznamu BlackList. Veškerá komunikace z IP adresy útočníka tím byla poté na 30 minut blokována.

Při testování ochrany proti útoku záplavou paketů (proveden test UDP Flood) byly pakety odchyčeny v příslušnými pravidly a i v tomto případě byla IP adresa útočníka přesunuta do seznamu BlackList.

Výsledek přesunutí IP adresy útočníka do seznamu BlackList je vidět na Obr. 22.

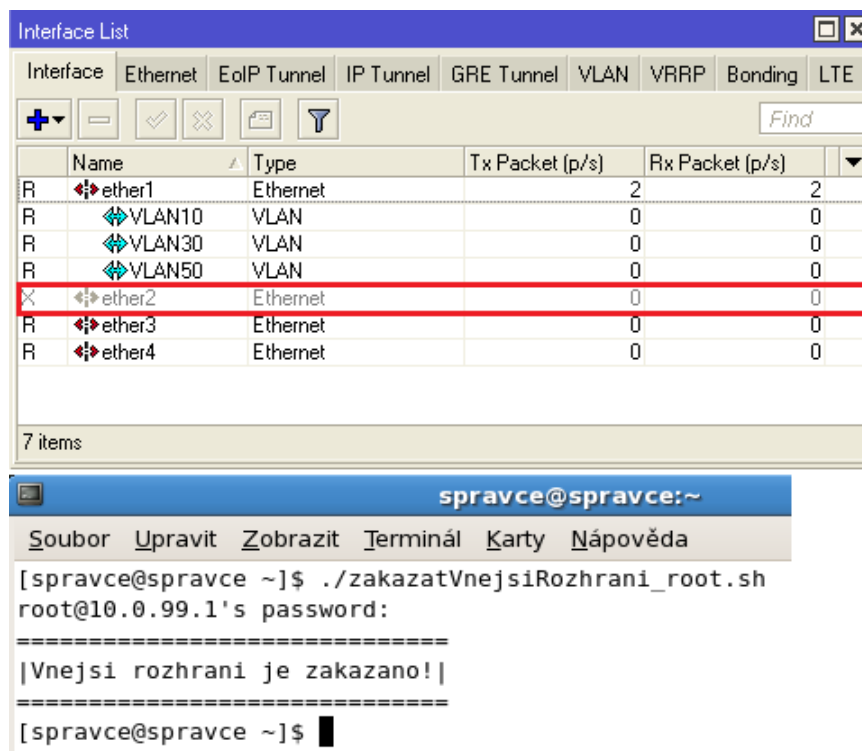


Obrázek 22: IP adresa útočníka v seznamu BlackList

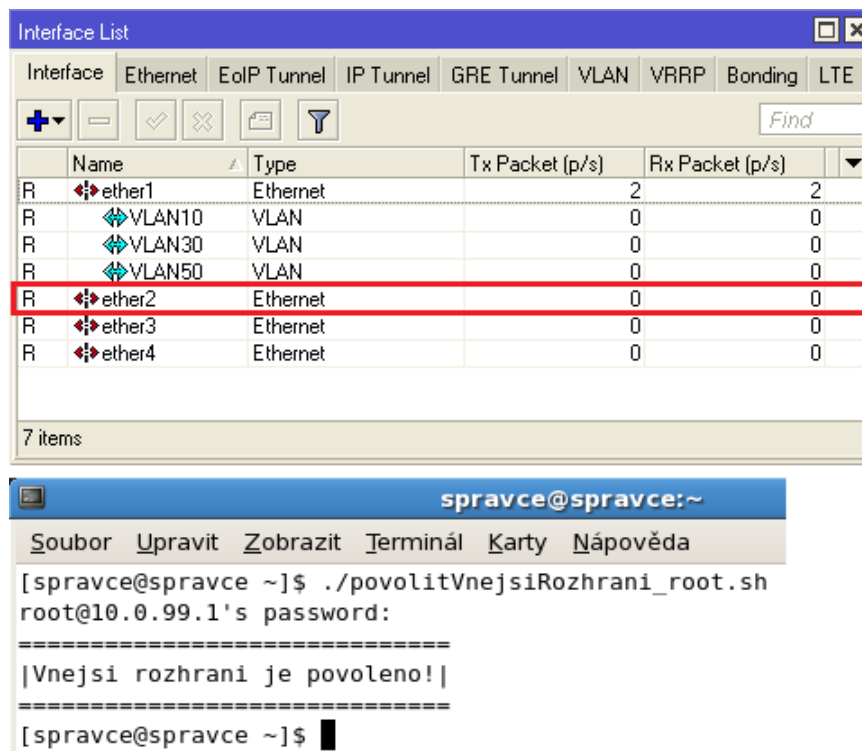
8.6 Testování skriptů

Jak je vidět na Obr. 23, skript pro zakázání vnějšího rozhraní firewallu úspěšně deaktivoval rozhraní ether2 a vypsalo zprávu se stavem do terminálu.

Po spuštění skriptu, který povoluje vnější rozhraní firewallu, bylo rozhraní ether2 opět aktivní. Výsledek povolení vnějšího rozhraní je vidět na Obr. 24.



Obrázek 23: Zakázání vnějšího rozhraní firewallu



Obrázek 24: Povolení vnějšího rozhraní firewallu

9 Ekonomické zhodnocení

Tato kapitola je zaměřena na ekonomickou stránku výsledků této práce. Hlavním důvodem migrace firewallu je snížení nákladů na elektrickou energii. Původní linuxový labrouter byl tower server, který potřeboval velké množství elektrické energie. Původní labrouter obsahoval napájecí zdroj s maximálním výkonem 305 W. Maximální výkon nového labrouteru od firmy MikroTik je 3 W.

Jak uvádějí Nygrýn a Čapek (2007), je u skupiny výkonných počítačů příkon v rozmezí 125–170 W. Pro výpočet je zvolen zaokrouhlený průměr naměřených hodnot tedy 150 W.

Pro porovnání nákladů na elektrickou energii vychází autor z článku Poncarové (2016), z kterého je vyvozena odhadovaná cena 4,02 Kč za 1 kWh elektrické energie. Výsledné porovnání původního a nového řešení je vyhodnoceno v Tab. 30.

Tabulka 30: Porovnání provozních nákladů

Labrouter	W	kW/h	kW/den	kW/měsíc	cena za měsíc
původní Linux	150,0	0,150	3,600	108,00	434,2 Kč
nový MikroTik	3,0	0,003	0,072	2,16	8,7 Kč
Rozdíl nákladů za el. energii za měsíc					425,5 Kč

Z Tab. 30 lze vyvodit, že úspora při provozu směrovače MikroTik RB951-2n je 425,5 Kč měsíčně oproti původnímu řešení.

Dále budou vyčísleny pořizovací náklady. Je obtížné zjistit odpovídající cenu routeru MikroTik RB951-2n, protože se nyní již nenachází na trhu. Cenu nového labrouteru lze odvodit od modelu s podobnými parametry, jako je například MikroTik RouterBOARD RB750r2, jehož cena se nyní pohybuje ve výši 900 Kč bez DPH.

Tabulka 31: Celková cena za nasazení nového firewallu

Položka	Cena/jednotku bez DPH	Množství	Cena celkem bez DPH	Cena celkem vč. DPH
MikroTik RB951-2n	900 Kč	1 ks	900 Kč	1 089 Kč
Návrh firewallu	500 Kč	20 h	10 000 Kč	12 100 Kč
Konfigurace směrovače	300 Kč	5 h	1 500 Kč	1 815 Kč
Nasazení firewallu	500 Kč	3 h	1 500 Kč	1 815 Kč
Testování firewallu	500 Kč	8 h	4 000 Kč	4 840 Kč
Celkem za nasazení nového firewallu			17 900 Kč	21 659 Kč

Celková cena za nasazení nového firewallu je, jak ukazuje Tab. 31, 21 659 Kč vč. DPH.

10 Závěr

Cílem této práce bylo zanalyzovat linuxový firewall (Iptables), který odděloval síťovou laboratoř ÚI PEF MENDELU od Internetu, a na základě této analýzy navrhnout a implementovat firewall na novém směrovači od firmy MikroTik. Mimo to bylo zapotřebí navrhnout a vytvořit skripty pro správu vnějšího rozhraní.

Výsledky autorovy práce jsou shrnuty v kapitole Návrh řešení a Řešení. Po analýze linuxového firewallu byly navrženy změny, které vedou k rychlejšímu průchodu paketů firewallem. Při migraci z původního firewallu byla odstraněna nepotřebná nebo duplicitní pravidla.

Výhodou nového řešení je přesunutí pravidla, které povoluje ustanovená a související spojení, na začátek řetězce Forward, čímž je dosaženo nižšího vytížení procesoru.

Nový labrouter je nyní díky přidání nových ochranných pravidel a technologie SYN Cookie imunní vůči DoS útokům a útokům skenováním portů.

Pro ovládání vnějšího rozhraní byly vytvořeny skripty, které lze jednoduše spouštět z ovládacích počítačů ve VLAN 99.

Bylo dosaženo snížení spotřeby elektrické energie, což byl hlavní důvod pro nahrazení původního linuxového labrouteru.

10.1 Nedostatky

Hlavním nedostatkem je nízký výkon současného labrouteru. Momentální vybavení síťové laboratoře ÚI PEF MENDELU nabízí pouze MikroTik RB951-2n. Tento směrovač disponuje pouze Fast ethernetovými porty. Pokud je v jeden okamžik aktivní mnoho spojení, což například nastává v případě, kdy všechny studentské počítače potřebují stáhnout určitý software, dochází k velkému zpomalení přenosové rychlosti.

Problém je v nízké maximální přenosové rychlosti 100 Mb/s, která je v případě velkého počtu aktivních spojení rozdělena, a dochází tak k rapidnímu snížení reálné přenosové rychlosti.

Podíl na nedostatku výkonu lze také přisoudit nízkému výkonu CPU a malé operační paměti. Taktovací frekvence procesoru směrovače RB951-2n je 400 MHz a operační paměť disponuje kapacitou pouhých 32 MB.

Bude tedy nutné do budoucna zajistit výkonnější MikroTik směrovač, který bude obsahovat výkonnější CPU a minimálně 2 Gigabit ethernetové porty, čímž vzroste maximální možná přenosová rychlost na 1 Gb/s, což je desetinásobný nárůst oproti současnému řešení.

11 Seznam literatury

- BARKER, K.; MORRIS, S. *CCNA security 640-554 official cert guide*. Indianapolis, IN: CISCO Press, 2013. ISBN 15-872-0446-0.
- BESANA, D. 8 Steps for a Successful Firewall Migration. *Router Freak* [online]. 2013, , 1 [cit. 2015-03-23]. Dostupné z: www.routerfreak.com/8-steps-successful-firewall-migration/.
- CISCO SYSTEMS. *Cisco Systems* [online]. San Jose (USA): Cisco Systems, Inc., 2015 [cit. 2016-02-28]. Dostupné z: <http://www.cisco.com/>.
- DISCHER, S. *RouterOS by example: understanding MikroTik RouterOS through real life applications*. College Station, Texas: MikroTik, 2011. ISBN 978-061-5547-046.
- DOČEKAL, M. Správa linuxového serveru: Linuxový firewall, základy iptables. *Linuxexpress* [online]. 2010, (12), 4 [cit. 2015-10-13]. Dostupné z: <http://www.linuxexpres.cz/praxe/sprava-linuxoveho-serveru-linuxovy-firewall-zaklady-iptables>.
- HARESTA, P. *Návrh a realizace lokální počítačové sítě ve střední firmě* [online]. Zlín, 2012 [cit. 2015-03-20]. Dostupné z: <https://stag.utb.cz/StagPortletsJSR168/KvalifPraceDownloadServlet?typ=1&adipidno=26508>. Diplomová práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky.
- HAVLÍČEK, J. *Zabezpečení systému Linux* [online]. Zlín, 2013 [cit. 2015-03-20]. Dostupné z: <https://stag.utb.cz/StagPortletsJSR168/KvalifPraceDownloadServlet?typ=1&adipidno=31668>. Bakalářská práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky.
- HORÁK, J.; KERŠLÁGER, M. *Počítačové sítě pro začínající správce*. 3., aktualiz. vyd. Brno: Computer Press, 2006. Bestseller (Computer Press). ISBN 80-251-0892-9.
- HVIZDÁK, M. *Firemný firewall s poštovým serverom v operačnom systéme Linux, distribúcia Red Hat* [online]. Zlín, 2013 [cit. 2015-03-23]. Dostupné z: <https://stag.utb.cz/StagPortletsJSR168/KvalifPraceDownloadServlet?typ=1&adipidno=31142>. Diplomová práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky.
- JUNIPER NETWORKS. *JNCIS-SEC Study Guide—Part 1*. 1. USA: Juniper Networks, Inc., 2012.
- KNYTL, R. *Simulace útoků na síťovou infrastrukturu* [online]. Pardubice, 2014 [cit. 2015-03-24]. Dostupné z: <http://dSPACE.upce.cz/handle/10195/55701>. Bakalářská práce. Univerzita Pardubice, Fakulta elektrotechniky a informatiky.

- KOSTELNÍK, P. *Reengineering počítačové sítě Domu kultury Vsetín, spol. s r. o.* [online]. Brno, 2011 [cit. 2015-03-20]. Dostupné z: https://is.mendelu.cz/zp/portal_zp.pl?prehled=vyhledavani;podrobnosti_zp=28527;zp=28527. Bakalářská práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta.
- KRAJČA, T. *Bezpečnostní rizika v síti internet z pohledu poskytovatele internetových služeb* [online]. Zlín, 2011 [cit. 2015-03-21]. Dostupné z: <https://stag.utb.cz/StagPortletsJSR168/KvalifPraceDownloadServlet?typ=1&adipidno=21168>. Bakalářská práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky.
- KRAJČA, T. *Návrh systému zabezpečení sítě poskytovatele internetových služeb* [online]. Zlín, 2013 [cit. 2015-03-21]. Dostupné z: <https://stag.utb.cz/StagPortletsJSR168/KvalifPraceDownloadServlet?typ=1&adipidno=31333>. Diplomová práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky.
- KUROSE, J. F.; ROSS K. W. *Počítačové sítě*. Brno: Computer Press, 2014. ISBN 978-80-251-3825-0.
- MEITNER, J. Iptables. In: *Ubuntu.cz* [online]. Praha, 2012 [cit. 2015-03-28]. Dostupné z: <http://wiki.ubuntu.cz/bezpe%C4%8Dnost/firewall/iptables>.
- MIKROTIK. *MikroTik Wiki* [online]. Lotyšsko: MikroTik, 2014 [cit. 2015-10-19]. Dostupné z: <http://wiki.mikrotik.com/wiki>.
- NYGRÝT, P.; ČAPEK, J. Elektřina: Kolik vás to stojí? In: *Extrahardware.cz* [online]. Brno: Extra Publishing, s. r. o., 2007 [cit. 2016-05-07]. Dostupné z: <http://www.cnews.cz/elektrina-kolik-vas-stoji>.
- PETRÁK, M. *Softwarový firewall pro filtrování na síťové a linkové vrstvě* [online]. Plzeň, 2013 [cit. 2015-03-25]. Dostupné z: <https://portal.zcu.cz/StagPortletsJSR168/KvalifPraceDownloadServlet?typ=1&adipidno=51124>. Diplomová práce. Západočeská univerzita v Plzni, Fakulta aplikovaných věd.
- PETŘÍČEK, M. Stavíme firewall. *Root.cz* [online]. 2001, 4, 3 [cit. 2015-03-21]. ISSN 1212-8309. Dostupné z: <http://www.root.cz/clanky/stavime-firewall-1/>.
- PONCAROVÁ, J. Cena kWh elektřiny 2016: Zjistěte, kolik a za co platíte. In: *Penize.cz* [online]. Praha: Partners media, s.r.o., 2016 [cit. 2016-05-04]. Dostupné z: <http://www.penize.cz/bydleni/308044-cena-kwh-elektriny-2016-zjistete-kolik-a-za-co-platite>.
- RUSSELL, P. Linux 2.4 NAT HOWTO. In: *Netfilter* [online]. 2002 [cit. 2015-11-05]. Dostupné z: <http://netfilter.org/documentation/HOWTO//NAT-HOWTO.html>.

- RUSSELL, P. Linux 2.4 Packet Filtering HOWTO. In: *Netfilter* [online]. 2002 [cit. 2015-11-05]. Dostupné z: <http://netfilter.org/documentation/HOWTO//packet-filtering-HOWTO.html>.
- SUDA, J. *Tvorba routeru na bázi Linuxu* [online]. České Budějovice, 2011 [cit. 2015-03-21]. Dostupné z: http://theses.cz/id/oly1w1/Ji_Suda_-_Tvorba_routeru_na_bzi_Linuxu.pdf. Bakalářská práce. Jihočeská univerzita, Přírodovědecká fakulta.
- ŠTRAUCH, A. Konfigurace firewallu na RouterOS od Mikrotiku. *Root.cz* [online]. 2011, 14, 1 [cit. 2015-03-23]. ISSN 1212-8309. Dostupné z: <http://www.root.cz/clanky/konfigurace-firewallu-na-routeros-od-mikrotiku/>.
- ŠTRAUCH, A. Mikrotik: skriptování v RouterOS. *Root.cz* [online]. 15, 1 [cit. 2015-03-23]. ISSN 1212-8309. Dostupné z: <http://www.root.cz/clanky/mikrotik-skriptovani-v-routeros/>.
- UBUNTU. *Ubuntu Wiki* [online]. Praha, 2014 [cit. 2015-03-28]. Dostupné z: <http://wiki.ubuntu.cz/>.
- VÁŇA, R. *Router na bázi linuxové distribuce* [online]. Zlín, 2012 [cit. 2015-03-26]. Dostupné z: <https://stag.utb.cz/StagPortletsJSR168/KvalifPraceDownloadServlet?typ=1&adipidno=26706>. Bakalářská práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky.
- VOHNÍK, R. *Centrální správa linuxového firewallu* [online]. Pardubice, 2013 [cit. 2015-03-23]. Dostupné z: <http://dspace.upce.cz/handle/10195/53911>. Diplomová práce. Univerzita Pardubice, Fakulta elektrotechniky a informatiky.
- VYHNÁTEK, J. *Generátor základních filtrovacích pravidel pro konfiguraci firewallů na síťových zařízeních* [online]. Brno, 2013 [cit. 2015-03-21]. Dostupné z: http://is.muni.cz/th/374138/fi_b/Bakalarska_prace_Jan_Vyhnanek.pdf. Bakalářská práce. Masarykova univerzita, Fakulta informatiky.
- VIRTUALBOX. *VirtualBox* [online]. Redwood Shores (USA): Oracle, 2015 [cit. 2015-11-30]. Dostupné z: <https://www.virtualbox.org/>.
- ÚSTAV INFORMATIKY. *Síťové technologie* [online]. Brno: Provozně ekonomická fakulta, 2013 [cit. 2015-9-30]. Dostupné z: <https://ui.pefka.mendelu.cz/>.

Přílohy

A Detail fyzického zapojení



Obrázek 25: Detail fyzického zapojení v síťové laboratoři ÚI PEF MENDELU

B Konfigurace labrouteru MikroTik

```
/system logging action add name=ToSyslog remote=10.0.99.101 syslog-facility=syslog target=remote
/system logging add action=ToSyslog prefix=labrouter topics=firewall

/ip settings set tcp-syncookies=yes

/interface vlan add interface=ether1 l2mtu=1596 name=VLAN10 vlan-id=10
/interface vlan add interface=ether1 l2mtu=1596 name=VLAN30 vlan-id=30
/interface vlan add interface=ether1 l2mtu=1596 name=VLAN50 vlan-id=50

/ip address add address=10.0.99.1/24 interface=ether1 network=10.0.99.0
/ip address add address=10.0.10.1/24 interface=VLAN10 network=10.0.10.0
/ip address add address=10.0.30.1/24 interface=VLAN30 network=10.0.30.0
/ip address add address=10.0.50.1/24 interface=VLAN50 network=10.0.50.0
/ip address add address=195.178.72.189/28 interface=ether2 network=195.178.72.176

/ip route add distance=1 gateway=195.178.72.177

/ip dhcp-relay add dhcp-server=10.0.30.2 disabled=no interface=VLAN10
local-address=10.0.10.1 name=DHCP_Relay_VLAN10

/interface bridge add name=loopback0

/ip dns set servers=10.0.30.2

/system clock set time-zone-name=Europe/Prague

/ip firewall nat add action=masquerade chain=srcnat out-interface=ether2
```

C Skript pro povolení vnějšího rozhraní

```
#!/bin/bash

prikaz="/interface ethernet set ether2 disabled=no;"

vysl=$((ssh root@10.0.99.1 "$prikaz") 2>&1);

case $vysl in
  '')
    echo "======"
    echo "|Vnejsi rozhrani je povoleno!|"
    echo "======";;
  *)
    echo "!!!!!!!!!!!!!!!"
    echo "Nastala chyba - "$vysl;;
esac

sleep 5
```

D Skript pro zakázání vnějšího rozhraní

```
#!/bin/bash

prikaz="/interface ethernet set ether2 disabled=yes;"

vysl=$((ssh root@10.0.99.1 "$prikaz") 2>&1);

case $vysl in
  '')  echo "======"
        echo "|Vnejsi rozhrani je zakazano!|"
        echo "======";;
  *)   echo "!!!!!!!!!!!!!"
        echo "Nastala chyba - "$vysl;;
esac

sleep 5
```

E Skript pro zjištění stavu vnějšního rozhraní

```
#!/bin/bash

prikaz="/interface print from=ether2 stats-detail;"

stav=$(ssh root@10.0.99.1 "$prikaz" | grep name | sed 's/^[ 0 ]*//' | cut -d ' ' -f 1)

case $stav in
R)
    echo "======"
    echo "|Vnejsi rozhrani je povoleno!|"
    echo "======"
    ;;
X)
    echo "======"
    echo "|Vnejsi rozhrani je zakazano!|"
    echo "======"
    ;;
*)
    echo "Chyba!"
    exit
    ;;
esac

sleep 5
```


F Skript pro zavedení firewallových pravidel

```
#!/bin/bash

##### IP ADRESY #####

#####
#IP adresy VLAN
#####
VLAN10_NET=10.0.10.0/24;
VLAN30_NET=10.0.30.0/24;
VLAN50_NET=10.0.50.0/24;
VLAN99_NET=10.0.99.0/24;

#####
#IP adresy portů na labrouteru
#####
VLAN10_IP=10.0.10.1;
VLAN30_IP=10.0.30.1;
VLAN50_IP=10.0.50.1;
VLAN99_IP=10.0.99.1;

#####
#IP adresy zařízení z VLAN10
#PXX_IP - IP adresy pokustonů
#PXX_OS_IP - IP adresy pokustonů sloužících k výuce Operačních systémů
#####
P00_IP=10.0.10.100;
P01_IP=10.0.10.101;
P02_IP=10.0.10.102;
P03_IP=10.0.10.103;
P04_IP=10.0.10.104;
P05_IP=10.0.10.105;
P06_IP=10.0.10.106;
P07_IP=10.0.10.107;
P08_IP=10.0.10.108;
P09_IP=10.0.10.109;
P10_IP=10.0.10.110;
P11_IP=10.0.10.111;
P12_IP=10.0.10.112;
P13_IP=10.0.10.113;
P14_IP=10.0.10.114;
P15_IP=10.0.10.115;
P16_IP=10.0.10.116;
P00_OS_IP=10.0.10.200;
P01_OS_IP=10.0.10.201;
P02_OS_IP=10.0.10.202;
P03_OS_IP=10.0.10.203;
P04_OS_IP=10.0.10.204;
P05_OS_IP=10.0.10.205;
P06_OS_IP=10.0.10.206;
P07_OS_IP=10.0.10.207;
P08_OS_IP=10.0.10.208;
P09_OS_IP=10.0.10.209;
P10_OS_IP=10.0.10.210;
P11_OS_IP=10.0.10.211;
P12_OS_IP=10.0.10.212;
P13_OS_IP=10.0.10.213;
P14_OS_IP=10.0.10.214;
P15_OS_IP=10.0.10.215;
P16_OS_IP=10.0.10.216;

#####
#IP adresy zařízení z VLAN30
#####
```

```
HOSTSE_IP=10.0.30.4;
WINSE_IP=10.0.30.3;
LINSE_IP=10.0.30.2;
```

```
#####
#IP adresy zařízení z VLAN99
#####
S01_IP=10.0.99.101;
S02_IP=10.0.99.102;
SWITCH_IP=10.0.99.2;
PDU_IP=10.0.99.3;
CPS_IP=10.0.99.4;
KVM_IP=10.0.99.5;
```

```
##### MAC ADRESY #####
```

```
#
```

```
#MAC adresy (čísloVlan_IPadresaPočítače)
```

```
#####
```

```
MAC99_1=00:11:3B:14:5C:A8;
MAC99_2=00:26:0A:5F:59:41;
MAC99_3=00:0A:9C:51:82:9B;
MAC99_4=00:E0:86:0A:21:C4;
MAC99_5=00:0F:58:01:36:E5;
#MAC99_100=08:00:27:D3:C8:C4;
MAC99_101=D0:67:E5:03:CF:DB;
MAC99_102=1C:C1:DE:5E:68:E3;
MAC30_1=00:11:3B:14:5C:A8;
MAC30_2=00:0C:29:03:E6:19;
MAC30_3=00:0C:29:4D:91:DC;
MAC30_4=BC:30:5B:E5:E9:DF;
MAC10_1=00:11:3B:14:5C:A8;
MAC50_1=00:11:3B:14:5C:A8;
MAC10_100=B4:B5:2F:C2:64:6B;
MAC10_101=B4:B5:2F:C8:8C:BC;
MAC10_102=B4:B5:2F:C2:64:7E;
MAC10_103=B4:B5:2F:BA:3D:81;
MAC10_104=24:BE:05:1A:28:DB;
MAC10_105=B4:B5:2F:C2:64:61;
MAC10_106=B4:B5:2F:C2:64:51;
MAC10_107=B4:B5:2F:BA:3D:82;
MAC10_108=B4:B5:2F:C8:8C:C0;
MAC10_109=B4:B5:2F:BA:3D:87;
MAC10_110=B4:B5:2F:C8:8C:CB;
MAC10_111=B4:B5:2F:BA:3D:6A;
MAC10_112=B4:B5:2F:C8:8C:C8;
MAC10_113=B4:B5:2F:C8:8C:C2;
MAC10_114=B4:B5:2F:C2:64:71;
MAC10_115=B4:B5:2F:BA:3D:8D;
MAC10_116=24:BE:05:14:BE:C4;
MAC10_200=08:00:27:C8:0A:00;
MAC10_201=08:00:27:C8:0A:01;
MAC10_202=08:00:27:C8:0A:02;
MAC10_203=08:00:27:C8:0A:03;
MAC10_204=08:00:27:C8:0A:04;
MAC10_205=08:00:27:C8:0A:05;
MAC10_206=08:00:27:C8:0A:06;
MAC10_207=08:00:27:C8:0A:07;
MAC10_208=08:00:27:C8:0A:08;
MAC10_209=08:00:27:C8:0A:09;
MAC10_210=08:00:27:C8:0A:10;
MAC10_211=08:00:27:C8:0A:11;
MAC10_212=08:00:27:C8:0A:12;
MAC10_213=08:00:27:C8:0A:13;
MAC10_214=08:00:27:C8:0A:14;
MAC10_215=08:00:27:C8:0A:15;
```

```
MAC10_216=08:00:27:C8:0A:16;

neniICMP="!icmp";
##### FIREWALLOVÁ PRAVIDLA #####

prikazy="
#odebrani vseh pravidel z firewallu
/ip firewall filter remove [/ip firewall filter find];
/ip firewall nat remove [/ip firewall nat find];
#
#####
#OUTPUT
#####
/ip firewall filter add chain=output;

#####
#INPUT
#####
#Povolení spojení ve stavu established nebo related
/ip firewall filter add chain=input protocol=$neniICMP connection-state=established,related;

#Ochrana Labrouteru před Ping Flood
/ip firewall filter add chain=input action=jump jump-target=PingLimit protocol=icmp
connection-state=established,related;

#Povolení všech paketů přicházejících z loopbacku
/ip firewall filter add chain=input in-interface=loopback0 action=accept;

#Kontrola paketů z vnějšího rozhraní na privátní IP adresu
/ip firewall filter add chain=input action=jump in-interface=ether2 jump-target=RFC1918_IN;

#Předání řetězci SpatnePakety, který zahodí vadné pakety
/ip firewall filter add chain=input action=jump jump-target=SpatnePakety;

#Povolení přístupu z S01 na Labrouter přes SSH spojení
/ip firewall filter add chain=input dst-address=$VLAN99_IP dst-port=22 in-interface=ether1 protocol=tcp
src-address=$S01_IP connection-state=new;

#Povolení přístupu z S02 na Labrouter přes SSH spojení
/ip firewall filter add chain=input dst-address=$VLAN99_IP dst-port=22 in-interface=ether1 protocol=tcp
src-address=$S02_IP connection-state=new;

#Povolení přístupu z S0 na Labrouter přes Winbox
/ip firewall filter add chain=input dst-address=$VLAN99_IP dst-port=8291 in-interface=ether1
protocol=tcp src-address=$S01_IP connection-state=new;

#Povolení pingu z vnějšího rozhraní na Labrouter
/ip firewall filter add chain=input icmp-options=8 in-interface=ether2 protocol=icmp;

#Povolení pingu z vnitřní sítě na rozhraní VLAN10
/ip firewall filter add chain=input dst-address=$VLAN10_IP icmp-options=8 in-interface=VLAN10
protocol=icmp src-address=$VLAN10_NET;

#Povolení pingu z vnitřní sítě na rozhraní VLAN30
/ip firewall filter add chain=input dst-address=$VLAN30_IP icmp-options=8 in-interface=VLAN30
protocol=icmp src-address=$VLAN30_NET;

#Povolení pingu z vnitřní sítě na rozhraní VLAN50
/ip firewall filter add chain=input dst-address=$VLAN50_IP icmp-options=8 in-interface=VLAN50
protocol=icmp src-address=$VLAN50_NET;

#Povolení pingu z vnitřní sítě na rozhraní ether1 (VLAN99)
/ip firewall filter add chain=input dst-address=$VLAN99_IP icmp-options=8 in-interface=ether1
protocol=icmp src-address=$VLAN99_NET;
```

```
#Povolení DHCP protokolu kvůli správné funkci DHCP Relay
/ip firewall filter add chain=input in-interface=VLAN10 dst-port=67 protocol=udp
comment="\kvuli DHCP Relay\" connection-state=new;

/ip firewall filter add chain=input in-interface=VLAN30 dst-port=67 protocol=udp
comment="\kvuli DHCP Relay\" connection-state=new;

#Zahození všech paketů, které nebyly povoleny předchozími pravidly
/ip firewall filter add chain=input action=jump jump-target=LogDrop_INPUT;
#####
#FORWARD
#####
#Povolení spojení ve stavu established nebo related
/ip firewall filter add chain=forward protocol=$neniICMP connection-state=established,related;

#Ochrana vnitřní sítě před Ping Flood
/ip firewall filter add chain=forward action=jump jump-target=PingLimit protocol=icmp
connection-state=established,related;

#Kontrola paketů z vnějšího rozhraní na privátní IP adresu
/ip firewall filter add chain=forward action=jump in-interface=ether2 jump-target=RFC1918_IN;

#Předání řetězci SpatnePakety, který zahodí vadné pakety
/ip firewall filter add chain=forward action=jump jump-target=SpatnePakety;

#Předání řetězci IpAdresy na kontrolu shody MAC adres s IP adresou
/ip firewall filter add chain=forward action=jump jump-target=IpAdresy;

#Předání paketu odeslaného z VLAN10 do Internetu řetězci Vlan10ToInternet
/ip firewall filter add chain=forward action=jump in-interface=VLAN10 jump-target=Vlan10ToInternet
out-interface=ether2 src-address=$VLAN10_NET connection-state=new;

#Předání paketu odeslaného z VLAN50 do Internetu řetězci Vlan50ToInternet
/ip firewall filter add chain=forward action=jump in-interface=VLAN50 jump-target=Vlan50ToInternet
out-interface=ether2 src-address=$VLAN50_NET connection-state=new;

#Předání paketu odeslaného z VLAN10 do VLAN30 řetězci Vlan10ToVlan30
/ip firewall filter add chain=forward action=jump dst-address=$VLAN30_NET in-interface=VLAN10
jump-target=Vlan10ToVlan30 out-interface=VLAN30 src-address=$VLAN10_NET connection-state=new;

#Předání paketu odeslaného z VLAN30 do VLAN10 řetězci Vlan30ToVlan10
/ip firewall filter add chain=forward action=jump dst-address=$VLAN10_NET in-interface=VLAN30
jump-target=Vlan30ToVlan10 out-interface=VLAN10 src-address=$VLAN30_NET connection-state=new;

#Předání paketu odeslaného z VLAN30 do VLAN99 řetězci Vlan30ToVlan99
/ip firewall filter add chain=forward action=jump dst-address=$VLAN99_NET in-interface=VLAN30
jump-target=Vlan30ToVlan99 out-interface=ether1 src-address=$VLAN30_NET connection-state=new;

#Předání paketu odeslaného z VLAN99 do VLAN30 řetězci Vlan99ToVlan30
/ip firewall filter add chain=forward action=jump dst-address=$VLAN30_NET in-interface=ether1
jump-target=Vlan99ToVlan30 out-interface=VLAN30 src-address=$VLAN99_NET connection-state=new;

#Předání paketu odeslaného z VLAN30 do Internetu řetězci Vlan30ToInternet
/ip firewall filter add chain=forward action=jump in-interface=VLAN30 jump-target=Vlan30ToInternet
out-interface=ether2 src-address=$VLAN30_NET connection-state=new;

#Předání paketu odeslaného z VLAN99 do Internetu řetězci Vlan99ToInternet
/ip firewall filter add chain=forward action=jump in-interface=ether1 jump-target=Vlan99ToInternet
out-interface=ether2 src-address=$VLAN99_NET connection-state=new;

#Předání paketu odeslaného z VLAN99 do VLAN10 řetězci Vlan99ToVlan10
/ip firewall filter add chain=forward action=jump in-interface=ether1 jump-target=Vlan99ToVlan10
out-interface=VLAN10 connection-state=new;
```

```
#Předání paketu odeslaného z VLAN10 do VLAN99 řetězci Vlan10ToVlan99
/ip firewall filter add chain=forward action=jump in-interface=VLAN10 jump-target=Vlan10ToVlan99
out-interface=ether1 connection-state=new;

#Předání paketu odeslaného z VLAN99 do VLAN50 řetězci Vlan99ToVlan50
/ip firewall filter add chain=forward action=jump in-interface=ether1 jump-target=Vlan99ToVlan50
out-interface=VLAN50 connection-state=new;

#Zahození všech paketů, které nebyly povoleny předchozími pravidly
/ip firewall filter add chain=forward action=jump jump-target=LogDrop_FORWARD;

#####
#IpAdresy
#####
/ip firewall filter add chain=IpAdresy action=return src-address=$VLAN99_IP src-mac-address=$MAC99_1;
/ip firewall filter add chain=IpAdresy action=return src-address=$SWITCH_IP src-mac-address=$MAC99_2;
/ip firewall filter add chain=IpAdresy action=return src-address=$PDU_IP src-mac-address=$MAC99_3;
/ip firewall filter add chain=IpAdresy action=return src-address=$CPS_IP src-mac-address=$MAC99_4;
/ip firewall filter add chain=IpAdresy action=return src-address=$KVM_IP src-mac-address=$MAC99_5;
/ip firewall filter add chain=IpAdresy action=return src-address=$S01_IP src-mac-address=$MAC99_101;
/ip firewall filter add chain=IpAdresy action=return src-address=$S02_IP src-mac-address=$MAC99_102;
/ip firewall filter add chain=IpAdresy action=return src-address=$VLAN30_IP src-mac-address=$MAC30_1;
/ip firewall filter add chain=IpAdresy action=return src-address=$LINSE_IP src-mac-address=$MAC30_2;
/ip firewall filter add chain=IpAdresy action=return src-address=$WINSE_IP src-mac-address=$MAC30_3;
/ip firewall filter add chain=IpAdresy action=return src-address=$HOSTSE_IP src-mac-address=$MAC30_4;
/ip firewall filter add chain=IpAdresy action=return src-address=$VLAN10_IP src-mac-address=$MAC10_1;
/ip firewall filter add chain=IpAdresy action=return src-address=$VLAN50_IP src-mac-address=$MAC50_1;
/ip firewall filter add chain=IpAdresy action=return src-address=127.0.0.1;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.101;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.102;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.103;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.104;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.105;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.106;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.107;
/ip firewall filter add chain=IpAdresy action=return src-address=10.0.50.110;
/ip firewall filter add chain=IpAdresy action=return src-address=$P00_IP src-mac-address=$MAC10_100;
/ip firewall filter add chain=IpAdresy action=return src-address=$P01_IP src-mac-address=$MAC10_101;
/ip firewall filter add chain=IpAdresy action=return src-address=$P02_IP src-mac-address=$MAC10_102;
/ip firewall filter add chain=IpAdresy action=return src-address=$P03_IP src-mac-address=$MAC10_103;
/ip firewall filter add chain=IpAdresy action=return src-address=$P04_IP src-mac-address=$MAC10_104;
/ip firewall filter add chain=IpAdresy action=return src-address=$P05_IP src-mac-address=$MAC10_105;
/ip firewall filter add chain=IpAdresy action=return src-address=$P06_IP src-mac-address=$MAC10_106;
/ip firewall filter add chain=IpAdresy action=return src-address=$P07_IP src-mac-address=$MAC10_107;
/ip firewall filter add chain=IpAdresy action=return src-address=$P08_IP src-mac-address=$MAC10_108;
/ip firewall filter add chain=IpAdresy action=return src-address=$P09_IP src-mac-address=$MAC10_109;
/ip firewall filter add chain=IpAdresy action=return src-address=$P10_IP src-mac-address=$MAC10_110;
/ip firewall filter add chain=IpAdresy action=return src-address=$P11_IP src-mac-address=$MAC10_111;
/ip firewall filter add chain=IpAdresy action=return src-address=$P12_IP src-mac-address=$MAC10_112;
/ip firewall filter add chain=IpAdresy action=return src-address=$P13_IP src-mac-address=$MAC10_113;
/ip firewall filter add chain=IpAdresy action=return src-address=$P14_IP src-mac-address=$MAC10_114;
/ip firewall filter add chain=IpAdresy action=return src-address=$P15_IP src-mac-address=$MAC10_115;
/ip firewall filter add chain=IpAdresy action=return src-address=$P16_IP src-mac-address=$MAC10_116;
/ip firewall filter add chain=IpAdresy action=return src-address=$P00_OS_IP src-mac-address=$MAC10_200;
/ip firewall filter add chain=IpAdresy action=return src-address=$P01_OS_IP src-mac-address=$MAC10_201;
/ip firewall filter add chain=IpAdresy action=return src-address=$P02_OS_IP src-mac-address=$MAC10_202;
/ip firewall filter add chain=IpAdresy action=return src-address=$P03_OS_IP src-mac-address=$MAC10_203;
/ip firewall filter add chain=IpAdresy action=return src-address=$P04_OS_IP src-mac-address=$MAC10_204;
/ip firewall filter add chain=IpAdresy action=return src-address=$P05_OS_IP src-mac-address=$MAC10_205;
/ip firewall filter add chain=IpAdresy action=return src-address=$P06_OS_IP src-mac-address=$MAC10_206;
/ip firewall filter add chain=IpAdresy action=return src-address=$P07_OS_IP src-mac-address=$MAC10_207;
/ip firewall filter add chain=IpAdresy action=return src-address=$P08_OS_IP src-mac-address=$MAC10_208;
/ip firewall filter add chain=IpAdresy action=return src-address=$P09_OS_IP src-mac-address=$MAC10_209;
/ip firewall filter add chain=IpAdresy action=return src-address=$P10_OS_IP src-mac-address=$MAC10_210;
/ip firewall filter add chain=IpAdresy action=return src-address=$P11_OS_IP src-mac-address=$MAC10_211;
```

```
/ip firewall filter add chain=IpAdresy action=return src-address=$P12_OS_IP src-mac-address=$MAC10_212;
/ip firewall filter add chain=IpAdresy action=return src-address=$P13_OS_IP src-mac-address=$MAC10_213;
/ip firewall filter add chain=IpAdresy action=return src-address=$P14_OS_IP src-mac-address=$MAC10_214;
/ip firewall filter add chain=IpAdresy action=return src-address=$P15_OS_IP src-mac-address=$MAC10_215;
/ip firewall filter add chain=IpAdresy action=return src-address=$P16_OS_IP src-mac-address=$MAC10_216;
/ip firewall filter add chain=IpAdresy action=jump jump-target=LogDrop_IpAdresy;

#####
#RFC 1918
#####
/ip firewall filter add chain=RFC1918_IN action=jump jump-target=LogDrop1918 src-address=192.168.0.0/16;
/ip firewall filter add chain=RFC1918_IN action=jump jump-target=LogDrop1918 src-address=10.0.0.0/8;
/ip firewall filter add chain=RFC1918_IN action=jump jump-target=LogDrop1918 src-address=172.16.0.0/12;
/ip firewall filter add chain=RFC1918_IN action=return;

#####
#Spatne pakety
#####
/ip firewall filter add chain=SpatnePakety action=jump dst-port=137 jump-target=LogDrop_SpatnePakety
protocol=udp;

/ip firewall filter add chain=SpatnePakety action=jump dst-port=138 jump-target=LogDrop_SpatnePakety
protocol=udp;

/ip firewall filter add chain=SpatnePakety action=jump connection-state=invalid
jump-target=LogDrop_SpatnePakety;

#PORT SCAN
/ip firewall filter add chain=SpatnePakety protocol=tcp in-interface=ether2 psd=21,3s,3,1
action=add-src-to-address-list address-list=BlackList address-list-timeout=30m comment="\Port Scanner\";

/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
protocol=tcp tcp-flags=urg,!ack,psh,!rst,!syn,fin;
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
protocol=tcp tcp-flags=!syn,!fin,!rst,!ack connection-state=new;
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
protocol=tcp tcp-flags=!urg,!ack,!psh,!rst,!syn,!fin;
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
protocol=tcp tcp-flags=urg,ack,psh,rst,syn,fin;
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
protocol=tcp tcp-flags=urg,ack,!psh,rst,syn,fin;
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
protocol=tcp tcp-flags=rst,syn;
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
protocol=tcp tcp-flags=syn,fin;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=ether1
jump-target=LogDrop_SpatnePakety src-address=255.255.255.255;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=VLAN10
jump-target=LogDrop_SpatnePakety src-address=255.255.255.255;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=VLAN30
jump-target=LogDrop_SpatnePakety src-address=255.255.255.255;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=VLAN50
jump-target=LogDrop_SpatnePakety src-address=255.255.255.255;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=ether2
jump-target=LogDrop_SpatnePakety src-address=255.255.255.255;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=ether1
jump-target=LogDrop_SpatnePakety src-address=127.0.0.0/8;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=VLAN10
jump-target=LogDrop_SpatnePakety src-address=127.0.0.0/8;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=VLAN30
jump-target=LogDrop_SpatnePakety src-address=127.0.0.0/8;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=VLAN50
jump-target=LogDrop_SpatnePakety src-address=127.0.0.0/8;
/ip firewall filter add chain=SpatnePakety action=jump in-interface=ether2
jump-target=LogDrop_SpatnePakety src-address=127.0.0.0/8;
```

```
/ip firewall filter add chain=SpatnePakety action=jump in-interface=ether2
jump-target=LogDrop_SpatnePakety src-address=0.0.0.0;

#PING OF DEATH
/ip firewall filter add chain=SpatnePakety action=jump fragment=yes jump-target=LogDrop_SpatnePakety
protocol=icmp;

/ip firewall filter add chain=SpatnePakety action=jump dst-port=139 jump-target=LogDrop_SpatnePakety
protocol=udp;

#SYN FLOOD
/ip firewall filter add chain=SpatnePakety protocol=tcp in-interface=ether2 tcp-flags=syn
action=add-src-to-address-list address-list=BlackList address-list-timeout=30m comment=\"SYN Flooder\"
connection-limit=300,32;

#UDP FLOOD
/ip firewall filter add chain=SpatnePakety protocol=udp in-interface=ether2
action=add-src-to-address-list address-list=BlackList address-list-timeout=30m comment=\"UDP Flooder\"
connection-limit=300,32;

#IP SOURCE ROUTE OPTIONS
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
ipv4-options=strict-source-routing;
/ip firewall filter add chain=SpatnePakety action=jump jump-target=LogDrop_SpatnePakety
ipv4-options=loose-source-routing;

#SYN FRAGMENTED
/ip firewall filter add chain=SpatnePakety action=jump protocol=tcp tcp-flags=syn fragment=yes
jump-target=LogDrop_SpatnePakety;

#Zahození paketů, které mají IP adresu shodnou s IP adresou v BlackListu
/ip firewall filter add chain=SpatnePakety in-interface=ether2 action=jump
jump-target=LogDrop_SpatnePakety src-address-list=BlackList comment=\"Shoda s Blacklistem\";
/ip firewall filter add chain=SpatnePakety action=return;

#####
#LogDrop1918
#####
/ip firewall filter add chain=LogDrop1918 action=log limit=10/1h,3 log=yes
log-prefix=\"RFC 1918 zahozeno: \";
/ip firewall filter add chain=LogDrop1918 action=drop;

#####
#LogDrop_10To30
#####
/ip firewall filter add chain=LogDrop_10To30 action=log limit=10/1h,3 log=yes
log-prefix=\"No match in 10to30: \";
/ip firewall filter add chain=LogDrop_10To30 action=drop;

#####
#LogDrop_10To99
#####
/ip firewall filter add chain=LogDrop_10To99 action=log limit=10/1h,3 log=yes
log-prefix=\"No match in 10to99: \";
/ip firewall filter add chain=LogDrop_10To99 action=drop;

#####
#LogDrop_10ToInternet
#####
/ip firewall filter add chain=LogDrop_10ToInternet action=log limit=10/1h,3 log=yes
log-prefix=\"No match in 10toNet: \";
/ip firewall filter add chain=LogDrop_10ToInternet action=drop;

#####
#LogDrop_30To10
```

```
#####
/ip firewall filter add chain=LogDrop_30To10 action=log limit=10/1h,3 log=yes
log-prefix="No match 30to10: \";
/ip firewall filter add chain=LogDrop_30To10 action=drop;

#####
#LogDrop_30To99
#####
/ip firewall filter add chain=LogDrop_30To99 action=log limit=10/1h,3 log=yes
log-prefix="No match in 30to99: \";
/ip firewall filter add chain=LogDrop_30To99 action=drop;

#####
#LogDrop_30ToInternet
#####
/ip firewall filter add chain=LogDrop_30ToInternet action=log limit=10/1h,3 log=yes
log-prefix="No match in 30toNet: \";
/ip firewall filter add chain=LogDrop_30ToInternet action=drop;

#####
#LogDrop_50ToInternet
#####
/ip firewall filter add chain=LogDrop_50ToInternet action=log limit=10/1h,3 log=yes
log-prefix="No match in 50toNet: \";
/ip firewall filter add chain=LogDrop_50ToInternet action=drop;

#####
#LogDrop_99To10
#####
/ip firewall filter add chain=LogDrop_99To10 action=log limit=10/1h,3 log=yes
log-prefix="No match in 99to10: \";
/ip firewall filter add chain=LogDrop_99To10 action=drop;

#####
#LogDrop_99To30
#####
/ip firewall filter add chain=LogDrop_99To30 action=log limit=10/1h,3 log=yes
log-prefix="No match in 99to30: \";
/ip firewall filter add chain=LogDrop_99To30 action=drop;

#####
#LogDrop_99To50
#####
/ip firewall filter add chain=LogDrop_99To50 action=log limit=10/1h,3 log=yes
log-prefix="No match in 99to50: \";
/ip firewall filter add chain=LogDrop_99To50 action=drop;

#####
#LogDrop_99ToInternet
#####
/ip firewall filter add chain=LogDrop_99ToInternet action=log limit=10/1h,3 log=yes
log-prefix="No match in 99toNet: \";
/ip firewall filter add chain=LogDrop_99ToInternet action=drop;

#####
#LogDrop_FORWARD
#####
/ip firewall filter add chain=LogDrop_FORWARD action=log limit=10/1h,3 log=yes
log-prefix="No match in FORWARD: \";
/ip firewall filter add chain=LogDrop_FORWARD action=drop;

#####
#LogDrop_INPUT
#####
/ip firewall filter add chain=LogDrop_INPUT action=log limit=10/1h,3 log=yes
```



```
log-prefix="No match in INPUT: \";
/ip firewall filter add chain=LogDrop_INPUT action=drop;

#####
#LogDrop_IpAdresy
#####
/ip firewall filter add chain=LogDrop_IpAdresy action=log limit=10/1h,3 log=yes
log-prefix="Invalid ip address: \";
/ip firewall filter add chain=LogDrop_IpAdresy action=drop;

#####
#LogDrop_SpatnePakety
#####
/ip firewall filter add chain=LogDrop_SpatnePakety action=log dst-port=137 protocol=udp limit=10/1h,3
log=yes log-prefix="Netbios zahozeno: \";
/ip firewall filter add chain=LogDrop_SpatnePakety action=drop dst-port=137 protocol=udp;

/ip firewall filter add chain=LogDrop_SpatnePakety action=log dst-port=138 protocol=udp limit=10/1h,3
log=yes log-prefix="Netbios zahozeno: \";

/ip firewall filter add chain=LogDrop_SpatnePakety action=drop dst-port=138 protocol=udp;

/ip firewall filter add chain=LogDrop_SpatnePakety action=log limit=10/1h,3 log=yes
log-prefix="Spatny paket: \";

/ip firewall filter add chain=LogDrop_SpatnePakety action=drop;

#####
#Vlan10ToInternet
#####
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P01_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P02_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P03_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P04_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P05_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P06_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P07_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P08_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P09_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P10_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P11_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P12_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P13_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P14_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P15_OS_IP;
/ip firewall filter add chain=Vlan10ToInternet action=reject reject-with=icmp-port-unreachable
src-address=$P16_OS_IP;

#HTTPS
/ip firewall filter add chain=Vlan10ToInternet dst-port=443 protocol=tcp src-address=$VLAN10_NET;
```

```
#HTTP
/ip firewall filter add chain=Vlan10ToInternet dst-port=80 protocol=tcp src-address=$VLAN10_NET;

#SSH
/ip firewall filter add chain=Vlan10ToInternet dst-port=22 protocol=tcp src-address=$VLAN10_NET;

#Remote Desktop Protocol
/ip firewall filter add chain=Vlan10ToInternet dst-port=3389 protocol=tcp src-address=$VLAN10_NET
disabled=yes;

#Povolení komunikace mezi KMS klientem a KMS server
/ip firewall filter add chain=Vlan10ToInternet dst-port=1688 protocol=tcp src-address=$VLAN10_NET
dst-address=195.178.72.13;

#PING
/ip firewall filter add chain=Vlan10ToInternet icmp-options=8 protocol=icmp src-address=$VLAN10_NET;

#FTP
/ip firewall filter add chain=Vlan10ToInternet dst-port=21 protocol=tcp src-address=$VLAN10_NET;

#NTP
/ip firewall filter add chain=Vlan10ToInternet dst-port=123 protocol=udp src-address=$VLAN10_NET;
/ip firewall filter add chain=Vlan10ToInternet dst-port=123 protocol=tcp src-address=$VLAN10_NET;

/ip firewall filter add chain=Vlan10ToInternet action=jump jump-target=LogDrop_10ToInternet;

#####
#Vlan30ToInternet
#####

#DNS
/ip firewall filter add chain=Vlan30ToInternet protocol=udp src-address=$LINSE_IP dst-port=53;

#NTP
/ip firewall filter add chain=Vlan30ToInternet protocol=udp src-address=$LINSE_IP dst-port=123;

#Aktualizace YUM
/ip firewall filter add chain=Vlan30ToInternet protocol=tcp src-address=$LINSE_IP dst-port=873;

#Aktualizace HTTP a HTTPS
/ip firewall filter add chain=Vlan30ToInternet protocol=tcp src-address=$WINSE_IP dst-port=80;
/ip firewall filter add chain=Vlan30ToInternet protocol=tcp src-address=$WINSE_IP dst-port=443;

#Hypervisor přístup k aktualizacím na internetu HTTP a HTTPS
/ip firewall filter add chain=Vlan30ToInternet protocol=tcp src-address=$HOSTSE_IP dst-port=80;
/ip firewall filter add chain=Vlan30ToInternet protocol=tcp src-address=$HOSTSE_IP dst-port=443;

#PING
/ip firewall filter add chain=Vlan30ToInternet protocol=icmp src-address=$LINSE_IP icmp-options=8;
/ip firewall filter add chain=Vlan30ToInternet protocol=icmp src-address=$WINSE_IP icmp-options=8;

#SSH
/ip firewall filter add chain=Vlan30ToInternet protocol=tcp src-address=$LINSE_IP dst-port=22;

/ip firewall filter add chain=Vlan30ToInternet action=jump jump-target=LogDrop_30ToInternet;

#####
#Vlan50ToInternet
#####

#HTTP
/ip firewall filter add chain=Vlan50ToInternet protocol=tcp src-address=$VLAN50_NET dst-port=80;
```

```
#SSH
/ip firewall filter add chain=Vlan50ToInternet protocol=tcp src-address=$VLAN50_NET dst-port=22;

#HTTPS
/ip firewall filter add chain=Vlan50ToInternet protocol=tcp src-address=$VLAN50_NET dst-port=443;

#DNS
/ip firewall filter add chain=Vlan50ToInternet protocol=udp src-address=$VLAN50_NET dst-port=53;

#PING
/ip firewall filter add chain=Vlan50ToInternet protocol=icmp src-address=$VLAN50_NET icmp-options=8;

/ip firewall filter add chain=Vlan50ToInternet action=jump jump-target=LogDrop_50ToInternet;

#####
#Vlan99ToInternet
#####

#Z S01 na Internet
/ip firewall filter add chain=Vlan99ToInternet src-address=$S01_IP;

#Z S02 na HTTP, HTTPS, SSH
/ip firewall filter add chain=Vlan99ToInternet protocol=tcp src-address=$S02_IP dst-port=80;
/ip firewall filter add chain=Vlan99ToInternet protocol=tcp src-address=$S02_IP dst-port=443;
/ip firewall filter add chain=Vlan99ToInternet protocol=tcp src-address=$S02_IP dst-port=22;

/ip firewall filter add chain=Vlan99ToInternet action=jump jump-target=LogDrop_99ToInternet;

#####
#Vlan10ToVlan99
#####
/ip firewall filter add chain=Vlan10ToVlan99 action=jump jump-target=LogDrop_10To99;

#####
#Vlan10ToVlan30
#####

#PING
/ip firewall filter add chain=Vlan10ToVlan30 protocol=icmp dst-address=$LINSE_IP icmp-options=8;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=icmp dst-address=$WINSE_IP icmp-options=8;

#DNS
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=53;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=53;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=53;

#Active Directory protocols
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=389;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=88;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=389;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=445;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=49152-65535;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=123;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=135;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=3268;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=138;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=445;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=636;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=3269;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=464;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=464;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=9389;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=67;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=2535;
```

```
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=88;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$WINSE_IP dst-port=137;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=25;

#DHCP
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=67;

#NTP
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=123;

#YUM
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=80;

#FTP
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=21;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=30000-30100;

#WSUS
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=80;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=443;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$WINSE_IP dst-port=8530;

#LDAP
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=389;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=111;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=111;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=892;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=2049;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=32803;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=662;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=662;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=892;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=2049;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=32803;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=32769;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=32769;

#CUPS
/ip firewall filter add chain=Vlan10ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=631;
/ip firewall filter add chain=Vlan10ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=631;

/ip firewall filter add chain=Vlan10ToVlan30 action=jump jump-target=LogDrop_10To30;

#####
#Vlan30ToVlan10
#####

#PING z WINSERVERU
/ip firewall filter add chain=Vlan30ToVlan10 protocol=icmp src-address=$WINSE_IP icmp-options=8;

#FTP
/ip firewall filter add chain=Vlan30ToVlan10 protocol=tcp dst-port=21;

#Active Directory
/ip firewall filter add chain=Vlan30ToVlan10 protocol=tcp dst-port=445;
/ip firewall filter add chain=Vlan30ToVlan10 protocol=udp dst-port=445;
/ip firewall filter add chain=Vlan30ToVlan10 protocol=tcp dst-port=80;

/ip firewall filter add chain=Vlan30ToVlan10 action=jump jump-target=LogDrop_30To10;

#####
#Vlan30ToVlan99
#####

#VPN
```

```
/ip firewall filter add chain=Vlan30ToVlan99 protocol=udp src-address=$LINSE_IP dst-address=$S01_IP
src-port=50001 dst-port=50001;

#LOG
/ip firewall filter add chain=Vlan30ToVlan99 protocol=udp src-address=$HOSTSE_IP dst-address=$S01_IP
dst-port=514;

#FTP z LINSERVERU na S01
/ip firewall filter add chain=Vlan30ToVlan99 protocol=tcp src-address=$LINSE_IP dst-address=$S01_IP
dst-port=21;

#FTP z LINSERVERU na S02
/ip firewall filter add chain=Vlan30ToVlan99 protocol=tcp src-address=$LINSE_IP dst-address=$S02_IP
dst-port=21;

#SSH
/ip firewall filter add chain=Vlan30ToVlan99 protocol=tcp src-address=$LINSE_IP dst-address=$S01_IP
dst-port=22;

/ip firewall filter add chain=Vlan30ToVlan99 action=jump jump-target=LogDrop_30To99;

#####
#Vlan99ToVlan10
#####

#PING
/ip firewall filter add chain=Vlan99ToVlan10 protocol=icmp src-address=$S01_IP dst-address=$VLAN10_NET
icmp-options=8;

#SSH
/ip firewall filter add chain=Vlan99ToVlan10 protocol=tcp src-address=$S01_IP dst-address=$VLAN10_NET
dst-port=22;

/ip firewall filter add chain=Vlan99ToVlan10 action=jump jump-target=LogDrop_99To10;

#####
#Vlan99ToVlan30
#####

#NTP
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$SWITCH_IP dst-address=$LINSE_IP
dst-port=123;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$S01_IP dst-address=$LINSE_IP
dst-port=123;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$CPS_IP dst-address=$LINSE_IP
dst-port=123;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$KVM_IP dst-address=$LINSE_IP
dst-port=123;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$PDU_IP dst-address=$LINSE_IP
dst-port=123;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$S02_IP dst-address=$LINSE_IP
dst-port=123;

#DNS
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$S01_IP dst-address=$LINSE_IP
dst-port=53;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$S02_IP dst-address=$LINSE_IP
dst-port=53;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$CPS_IP dst-address=$LINSE_IP
dst-port=53;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$SWITCH_IP dst-address=$LINSE_IP
dst-port=53;

#RADIUS
```

```
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$SWITCH_IP dst-address=$LINSE_IP
dst-port=1812;

#VMware Server Console
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$HOSTSE_IP
dst-port=902;

#VPN
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$S01_IP dst-address=$LINSE_IP
src-port=50001 dst-port=50001;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp src-address=$S01_IP dst-address=$HOSTSE_IP
src-port=50000 dst-port=50000;

#PING
/ip firewall filter add chain=Vlan99ToVlan30 protocol=icmp src-address=$S01_IP dst-address=$LINSE_IP
icmp-options=8;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=icmp src-address=$S01_IP dst-address=$WINSE_IP
icmp-options=8;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=icmp src-address=$S01_IP dst-address=$HOSTSE_IP
icmp-options=8;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=icmp src-address=$S02_IP dst-address=$LINSE_IP
icmp-options=8;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=icmp src-address=$S02_IP dst-address=$WINSE_IP
icmp-options=8;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=icmp src-address=$S02_IP dst-address=$HOSTSE_IP
icmp-options=8;

#SSH
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$LINSE_IP
dst-port=22;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S02_IP dst-address=$LINSE_IP
dst-port=22;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$HOSTSE_IP
dst-port=22;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S02_IP dst-address=$HOSTSE_IP
dst-port=22;

#RDP
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$WINSE_IP
dst-port=3389;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S02_IP dst-address=$WINSE_IP
dst-port=3389;

#HTTP a HTTPS
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$LINSE_IP
dst-port=80;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S02_IP dst-address=$LINSE_IP
dst-port=80;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$HOSTSE_IP
dst-port=80;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$HOSTSE_IP
dst-port=443;

#Přístup na WSUS z vSphere Clienta
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$WINSE_IP
dst-port=8530;

#FTP z S01 na LINSERVER
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S01_IP dst-address=$LINSE_IP
dst-port=21;

#FTP z S02 na LINSERVER
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp src-address=$S02_IP dst-address=$LINSE_IP
dst-port=21;
```

```
#Komunikace CUPS
/ip firewall filter add chain=Vlan99ToVlan30 protocol=udp dst-address=$LINSE_IP dst-port=631;
/ip firewall filter add chain=Vlan99ToVlan30 protocol=tcp dst-address=$LINSE_IP dst-port=631;

/ip firewall filter add chain=Vlan99ToVlan30 action=jump jump-target=LogDrop_99To30;

#####
#Vlan99ToVlan50
#####
#SSH
/ip firewall filter add chain=Vlan99ToVlan50 protocol=tcp src-address=$S01_IP dst-address=10.0.50.110
dst-port=22;
/ip firewall filter add chain=Vlan99ToVlan50 action=jump jump-target=LogDrop_99To50;

#####
#PingLimit
#####
/ip firewall filter add chain=PingLimit limit=10/1s,5;
/ip firewall filter add chain=PingLimit action=drop;

#####
#NAT
#####

/ip firewall nat add chain=srcnat out-interface=ether2 action=masquerade;

"

vysl=$((ssh root@10.0.99.1 "$prikazy") 2>&1);
echo $vysl;

case $vysl in
  '') echo "====="
      echo "|Firewall byl uspesne nasazen|"
      echo "=====";
  *) echo "!!!!!!!!!!!!!!"
      echo "Nastala chyba - "$vysl;;
esac

sleep 1
```