Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Diplomová práce

Zabezpečení serverů s operačním systémem Linux

Bc. Jakub Slapnička

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Jakub Slapnička

Systémové inženýrství a informatika
Informatika

Název práce

Zabezpečení serverů s operačním systémem Linux

Název anglicky

Security of servers with operating system Linux

Cíle práce

Cílem této diplomové práce je návrh nové obecné metodiky zabezpečení linuxových serverů vycházející z aktuálně dostupných poznatků a již existujících metodik a postupů.

Součástí práce bude vymezení a popsání linuxových distribucí, využívaných pro servery, definování bezpečnostních rizik serverů a možných postupů správné konfigurace serveru, které tato rizika minimalizují a zvýší tak bezpečnost serverů.

Mezi dílčí cíle práce patří výběr vhodné linuxové distribuce a na základě zjištěných doporučených postupů konfigurace serveru, tyto postupy použít v praxi a následně provést testy zabezpečení serveru a analýzu výsledků.

Metodika

Teoretická část diplomové práce bude vypracována analýzou linuxových distribucí, které se využívají jako operační systémy pro servery. Také bude provedena analýza odborných a vědeckých knih a článků, které se zabývají problematikou ochrany linuxových serverů.

V praktické části diplomové práce bude nejprve provedena analýza pro výběr nejlepší linuxové distribuce a následně budou metodou syntézy aplikovány bezpečnostní postupy a metody na server. Poté bude testována jejich efektivita pomocí simulace útoků a na základě získaných poznatků z aplikování a testování bezpečnostních metod bude provedena analýza výsledků.

Doporučený rozsah práce

60 – 80 stran

Klíčová slova

hacking, server, operační systém, Linux, bezpečnost, zabezpečení, online

---

Doporučené zdroje informací

A. TEVAULT, Donald. Mastering Linux Security and Hardening. Birmingham: Packt Publishing Ltd., 2018. ISBN 978-1-78862-030-7.

D. BAUER, Michael. Linux Server Security. Second Edition. California: O'Reilly Media, 2005. ISBN 0-596-00670-5.

DOSTÁLEK, Libor. Velký průvodce protokoly TCP/IP: bezpečnost. 2. aktualiz. vyd. Praha: Computer Press, 2003. ISBN 80-7226-849-X.

ERDAL OZKAYA Cybersecurity: The Beginner's Guide. Birmingham: Packt Publishing Ltd., 2019. ISBN 9781789616194.

FLICKENGER, Rob. LINUX server hacks. Boston: O'Reilly, 2003. ISBN 0-596-00461-3.

OSTERLOH, Heather. TCP/IP: kompletní průvodce : použitelný pro veškeré operační systémy. Praha: SoftPress, 2003. ISBN 80-86497-34-8.

Předběžný termín obhajoby

2020/21 LS – PEF

Vedoucí práce

Ing. Václav Lohr, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 29. 7. 2020

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 21. 10. 2020

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 29. 03. 2021

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Zabezpečení serverů s operačním systémem Linux" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 29.3.2021 _____

Poděkování

Rád bych touto cestou poděkoval Ing. Václavu Lohrovi, Ph.D. za jeho cenné rady při psané této diplomové práce.

Zabezpečení serverů s operačním systémem Linux

Abstrakt

Tato diplomová práce se zabývá problematikou bezpečnosti linuxových serverů a metodami, jak server bezpečně nastavit a chránit ho tak před útoky a zneužitím. V úvodu práce je představen Linux obecně, včetně jeho historie a charakteristik. Dále budou v teoretické části práce uvedeny různá rizika, která mohou linuxový server ohrozit a metody, jak se před těmito riziky bránit.

V praktické části budou metody a praktiky z teoretické části aplikovány do praxe a následně pak bude provedena analýza získaných výsledků.

Klíčová slova: hacking, server, operační systém, Linux, bezpečnost, zabezpečení, online

Security of servers with operating system Linux

Abstract

This diploma thesis deals with problematics of security of Linux based servers and methods of securely setting up the server and protecting it from attacks and malfeasance. In the introduction of thesis is introduce Linux in general, including its history and characteristics. Furthermore, the theoretical part of the thesis will list various risks that can threaten the Linux server and methods to defend against these risks.

In the practical part of thesis, the methods and practices from the theoretical part will be applied in practice and then the analysis of the obtained results will be performed.

Keywords: hacking, server, operating system, Linux, safety, security, online

Obsah

Seznam obrázků

Seznam tabulek

Seznam použitých zkratek

DNS	Domain Name Server
GNU	GNU není Unix (z angl. GNU ‘s not Unix)
GPL	Obecně veřejná licence (z angl. General Public License)
LTS	Dlouhodobá podpora (z angl. Long-term support)
GUI	Grafické uživatelské rozhraní (z angl. Graphic User Interface)
GNOME	Prostředí síťového modelu GNU (z angl. GNU Network Object Model Environment)
SIG	Speciální Zájmová Skupina (z angl. Special Interest Group)
RHEL	Red Hat Enterprise Linux
DoS	Zamítnutí služby (z angl. Denial of Service)
DDoS	Distribuované zamítnutí služby (z angl. Distributed Denial of Service)
SQL	Strukturovaná dotazovací jazyk (z angl. Structured Query Language)
CPU	Centrální procesorová jednotka (z angl. Central Processing Unit)
HTTP	Hypertext Transfer Protocol
HTTPD	Hypertext Transfer Protocol Daemon
MITM	Člověk uprostřed (z angl. Man in the Middle)
ARP	Protokol pro rozlišování adres (z angl. Address Resolution Protocol)
UID	Identifikátor uživatele (z angl. User Identifier)
GID	Identifikátor skupiny (z angl. Group Identifier)
TLS	Zabezpečení transportní vrstvy (z angl. Transport Layer Security)
SSL	Vrstva bezpečných soketů (z angl. Secure Sockets Layer)
FTP	Protokol pro přenos souborů (z angl. File Transfer Protocol)
SFTP	Bezpečný protokol pro přenos souborů (z angl. Secure File Transfer Protocol)
TCP	Protokol pro řízení přenosu (z angl. Transmission Control Protocol)
IP	Internetový protokol (z angl. Internet protocol)
SSH	Bezpečný shell (z angl. Secure Shell)
UUCP	Kopírovací protokol Unix Unixu (z angl. Unix to Unix Copy Protocol)
NTP	Síťový časový protokol (z angl. Network Time Protocol)
UDP	Protokol uživatelského datagramu (z angl. User Datagram Protocol)
RELP	Spolehlivý protokol pro zaznamenávání událostí (z angl. Reliable Event Logging Protocol)
IPS	Systém prevence průniku (z angl. Intrusion Prevention System)
IDS	Systém detekce průniku (z angl. Intrusion Detection System)
NIDS	Systém detekce průniku sítě (z angl. Network Intrusion Detection System)

LMD	Linuxová detekce malwaru (z angl. Linux Malware Detect)
ISO	ISO Obraz
GB	Giga Byte
VSFTPD	Velmi Bezpečný FTP Démon (z angl. Very Secure FTP Daemon)
GRUB	Unifikovaný Bootloader Grand (z angl. GRand Unified Bootloader)
PAM	Zásuvné ověřovací moduly (z angl. Pluggable Authentication Modules)
ICMP	Internetový Kontrolní Protokol Zpráv (z angl. Internet Control Message Protocol)
IRC	Rozhovor přenášený po internetu (z angl. Internet Relay Chat)
LOIC	Iontové dělo na nízké oběžné dráze (z angl. Low Orbit Ion Cannon)
VPS	Virtuální Osobní Server (z angl. Virtual Private Server)

1 Úvod

V současné době nemusí být server využíván pouze velkými společnostmi, ale servery mohou složit i malým podnikům či pouze obyčejným uživatelům pro správu jejich souborů či hostování jejich individuálních projektů. Avšak na rozdíl od velkých společností, která mají svá vlastní IT oddělení, aby se starali o bezpečnost jejich serverů, malé podniky tyto výsady mít nemusí. I přesto, že se může jednat o malé podniky či jednotlivé uživatele, tak i jejich servery by měli mít zajištěnou jistou úroveň zabezpečení, která je v dnešním digitálním světě důležitá.

Teoretická část práce je zaměřena na definování různých typů dedikovaných serverů. Dále bude představena historie a vznik operačního systému GNU/Linux a také budou představeny jeho linuxové distribuce, které jsou často nabízeny českými poskytovateli serverových služeb, a budou i ukázány jejich základní výhody a nevýhody. Nedílnou součásti práce bude i definování nejznámějších bezpečnostních rizik, které pomáhají tyto hrozby minimalizovat. Též budou i popsány postupy konfigurace často používaných služeb na serveru.

Praktická část práce obsahuje vícekriteriální analýzu, která sloužila pro výběr linuxové distribuce. Na vybrané linuxové distribuci jsou provedeny bezpečnostní konfigurace a nainstalované doplňující programy, které byly probrané v teoretické části, a tato bezpečnostní zabezpečení jsou dále v praktické části testována.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem této diplomové práce je návrh nové obecné metodiky zabezpečení linuxových serverů vycházející z aktuálně dostupných poznatků a již existujících metodik a postupů.

Dalším cílem práce je vymezení a popsání linuxových distribucí, využívané pro servery, definování bezpečnostních rizik serverů a možných postupů správné konfigurace serveru, která tato rizika minimalizují a zvýší tak bezpečnost serverů

Mezi dílčí cíle práce patří výběr vhodné linuxové distribuce a na základě zjištěných doporučených postupů konfigurace serveru, tyto postupy použít v praxi a následně provést testy zabezpečení serveru a analýzu výsledků.

2.2 Metodika

Teoretická část diplomové práce bude vypracována analýzou linuxových distribucích, které se využívají jako operační systémy pro servery. Také zde bude provedena analýza odborných a vědeckých knih a článků, které se zabývají problematikou ochrany linuxových serverů.

V praktické části diplomové práce bude nejprve provedena analýza pro výběr nejvhodnější linuxové distribuce a následně budou metodou syntézy aplikovány bezpečnostní postupy a metody na server. Poté bude testována jejich efektivita pomocí simulace útoků a na základě získaných poznatků z aplikování a testování bezpečnostních metod bude provedena analýza výsledků.

3 Teoretická východiska

3.1 Vymezení pojmů

3.1.1 Server

Server se dá považovat za počítač, který poskytuje jednu nebo více služeb dalším počítačům, kteří se serverem komunikují jako klienti. Servery, které poskytují pouze jednu určitou službu a žádnou další jinou již neposkytují, se označují jako dedikované servery. Ovšem server nelze považovat za klasický stolní počítač. Servery jsou na rozdíl od počítačů navrhovány tak, aby byly spolehlivější, protože většina serverů se nikdy nevypíná, a přesto musí neustále zvládat ukládat a zpracovávat data 24 hodin denně. (BEAL, 2011)

Pro správnou funkčnost serveru, musí být server nakonfigurován tak, aby dokázal zpracovávat požadavky klientů, kteří jsou připojeni na stejné síti. Tato funkcionalita může být zajištěna buď přímo jako součást operačního systému nebo nainstalováním specifického programu anebo jako kombinace těchto dvou daných řešeních. Pokud klient potřebuje ze serveru nějaká data nebo potřebuje využít jeho výpočetní zdroje, tak klient odešle na server požadavek. Pokud například server obdrží požadavek od klienta, na zaslání vybraných dat, tak server nejdříve zkontroluje, zda klient má k daným datům přístupová oprávnění a na základě toho buď server odešle vybraná data klientovi anebo požadavek zamítne. (MITCHELL, 2020)

Existuje mnoho typů serverů a každý provádí nějakou jinou činnost. Mezi základní typy serverů patří například:

- **Webový server.** Webový server je jeden z dalších nejpoužívanějších typů serverů. Jedná se o speciální typ aplikačního serveru, který poskytuje programy a data klientům přes internet nebo intranet. Webové servery zpracovávají požadavky z internetových prohlížečů, které jsou spuštěny na klientských počítačích.

- **Souborový server.** Souborové servery ukládají a poskytují soubory klientům. Tyto servery se dají využít jako snadná varianta pro zálohování dat, protože si

svá data můžete uložit a stáhnout odkudkoliv a kdekoliv. Soubory na serveru mohou být sdíleny mezi vícero klientů.

- **Virtuální server.** Oproti klasickým serverům, které existují jako operační systém nainstalovaný na hardwaru, virtuální servery existují jako součást programu, který se nazývá hypervizor. Na jednom hypervizoru může běžet několik stovek až tisíce virtuálních serverů najednou. (BEAL, 2011)

- **Proxy server.** Proxy server slouží jako prostředník mezi klientským počítačem a serverem. Tento druh serveru se používá pro zvýšení bezpečnosti ať už k izolaci klienta anebo serveru. Jelikož proxy server slouží jako prostředník, tak zde není potřeba, aby klient byl nějak napojený na server, na který chce poslat svůj požadavek.

- **Databázový server.** Velké společnosti zpracovávají velké množství dat, a tyto data jsou ukládána do databází, které musí být přístupné více zaměstnacům najednou a též mohou spotřebovávat spoustu místa na disku. Kvůli těmto podmínkám jsou využívané servery, které jsou přímo dedikované databázím.

- **DNS server.** Jedná se o aplikační server, který zajišťuje převod doménových názvů stránek na IP adresy a naopak. Webové stránky mají svoji doménu (např. www.facebook.com), ale tyto domény slouží pouze pro lepší práci se stránkami pro uživatele, avšak skutečná adresa stránky je pouhá IP adresa, kterou si uživatel snadno nezapamatuje a špatně ji vyhledá. (MITCHELL, 2020)

3.1.2 GNU/Linux

V roce 1983 vznikl projekt GNU, jehož zakladatelem byl Richard Matthew Stallman, jehož úkolem bylo vytvoření nového unixového operačního systému, který měl být postavený pouze na zcela svobodném softwaru. Z GNU se stal operační systém, který byl kompatibilní s různými unixovými systémy. GNU nabízel veškeré důležité aplikace chybělo mu však jádro, které by zajistilo pro systém samostatnou funkčnost, a proto byl zahájen vývoj vlastního jádra Hurd.

Odděleně od projektu GNU pracoval Linus Torvalds na vývoji svého vlastního unixového jádra. Na vývoji unixového jádra pracoval, protože se mu unixové systémy zalíbily, ale většina jich byla příliš drahé, než aby si je mohl dovolit. Toto unixové jádro je později známo pod názvem Linux. (NEWELL, 2020)

Linux si získal mnoho příznivců, kteří pak taky začali pracovat na jeho vývoji. Jelikož se stal Linux oblíbeným, tak Linus zveřejnil zdrojové kódy pod licencí GNU/GPL (svobodná licence, která byla součástí projektu GNU). Protože byl vývoj Linuxu lepší jak vývoj Hurdu, tak se operační systém GNU začal využívat společně s Linuxem a vznikl tak systém GNU/Linux, který se často pouze nazývá jako Linux.

Mnohdy se špatně uvádí, že jádro Linux je součástí systému GNU nebo opačně, že GNU je součástí Linuxu. Pravda je taková, že se jedná o dva samostatné projekty, které dohromady tvoří jednu z nejpoužívanějších kombinací svobodného softwaru. (MOORE, 2020)

Další milný fakt je ten, že neexistuje žádný operační systém Linux, ale systém je publikován v tzv. distribucích. Linuxová distribuce vznikne spojením GNU, Linuxu a dalších systémových balíčků a konfiguračních nástrojů. Všehny distribuce jsou jiné a každá je něcím specifická zaměřena na nějakou činnost a proto je výběr dané distribuce velmi důležitý.

Spousta uživatelů může mít obavy zda bude i nadále moct provádět stejné operace i na jiném operačním systému, a zda i nadále budou moct používat stejné programy anebo aspoň programy jim podobné. V dnešní době je spousta programů multiplatformní, tedy uživatel bude moct používat svoje aktuální programy i na linuxových distribucích, anebo pro Linux existuje řada zástupců známých a používaných programů, mezi které patří např. internetové prohlížeče, přehrávače médií, textové editory, grafické editory a další. (PRAKASH, 2020)

3.1.3 Licence GNU/GPL

Licence GNU/GPL je série licencí pro softwarové produkty, které vznikly pro projekt GNU a jednalo se o tzv. copyleft licence, které umožňovaly uživatelům software svobodně používat, šířit a upravovat. GNU/GPL a svobodný software jsou mnohdy špatně interpretovány, protože svoboda souvisí s užitím, ale ne s cenou za daný software. (DiBONA, a další, 1999) Software je svobodný, pokud umožňuje následující:

- Uživatel může používat software pro své účely
- Uživatel může software upravit pro svoje účely
- Uživatel může sdílet kopie softwaru zdarma anebo za poplatek
  (COTTON, 2016)

3.1.4 Linuxové distribuce

Jak bylo zmíněno výše, tak linuxových distribucí je mnoho a záleží na potřebách uživatele, kterou distribuci se rozhodne používat. Mezi základní distribuce, které se používají jako operační systémy pro servery, se řadí Debian, Ubuntu a CentOS.

Přestože jsou distribuce rozdílné, tak pořád sdílí společné výhody i nevýhody, které platí pro všechny systémy založené na GNU/Linux. Mezi velkou výhodu se může řadit to, že tyto systémy jsou odolnější vůči virům (HOFFMAN, 2016). Nelze říct, že linuxové distribuce jsou plně chráněny proti všem počítačovým virům, ale mají lepší bezpečnost nežli jiné operační systémy. Tato výhoda vzniká i díky faktu, že většina uživatelů osobních počítačů používá jako svůj operační systém Windows nebo MacOS a hackeři se více cílí na tyto systémy. Mezi velkou nevýhodu se řadí nekompatibilita softwarových produktů, protože nejen hackeři, ale i softwarové společnosti vyvíjí své produkty spíše pro Windows a MacOS. (TARAFDER, 2019)

Níže uvedené distribuce byly vybrány na základě poskytovatelů serverů v České republice. Od poskytovatelů jako například Forpsi,Wedos či Active24 byly vybrány ty distribuce, které se opakovaly nejčastěji. V potaz byl brán i seznam nejlepších linuxových distribucí za rok 2020, který byl uveden na stránkách blogu TecMint. (KILI, 2020)

3.1.4.1 Debian

Hlavním úkolem projektu Debian je vytvoření zcela svobodného operačního systému, který se nazývá Debian a jedná se o již výše zmíněnou linuxovou distribuci. Ovšem ne vždy se musí jednat o linuxovou distribuci, jelikož Debian v současné době může používat jako svoje jádro Linux nebo FreeBSD, a i nadále se pracuje na poskytování systému Debian i na další unixová jádra. (PADAMKAR, 2019)

Značná část základních programů systému Debian pochází již z projektu GNU, avšak uživatelé používají Debian pro jeho aplikační software, díky kterému se Debian odlišuje od ostatních linuxových distribucích. Systém Debian obsahuje více jak 59 000 balíčků, což jsou předkompilované programy, které si uživatelé mohou do systému doinstalovat. (KAMENÍK, 2018)

Výhody

1. **Komplexní manuály**

Mezi další výhody by se dali řadit manuály k nabízeným balíčkům systému. Veškeré linuxové distribuce obsahují manuály ke svým balíčkům, ale manuál k balíčkům systému Debian jsou nejvíce komplexní. Manuály obsahují všechny možné detaily, co může daný balíček obsahovat a také jak lze balíček integrovat s jinými balíčky. (BYFIELD, 2017)

2. **Systém na sledování chyb**

Ne každý systém vždy funguje správně, a tak tomu platí i systému Debian. Aby se ovšem zabránilo chybám co nejdříve, tak existuje veřejně dostupný systém na sledování chyb, který umožňuje Debianu reagovat na problémy a chyby co nejdříve. Uživatelé, kteří zadali chybu k opravě, jsou o jejím uzavření informování, a zároveň si mohou přečíst, jak byla chyba vyřešena.

3. **Rychlost systému**

Jiné operační systémy mohou být v jedné nebo ve dvou oblastech rychlejší, ale jelikož je Debian založen na GNU/Linuxu (nebo i na GNU/FreeBSD), tak je schopný zpracovávat složité úlohy efektivně. Softwarový, který je navržený pro operační

systém Windows, lze spustit na Debianu pomocí emulátoru a někdy dokáže běžet rychleji nežli ve svém původním prostředí.

4. **Zákaznická podpora**

Debian má svojí vlastní „zákaznickou podporu“, která je velkou výhodou obzvlášť pro nové uživatele, a navíc absolutně zdarma. Stačí pouze poslat email a svoji odpověď dostanete maximálně do 15 minut, a to přímo od lidí, kteří na systému Debian pracují. (DEBIAN, 2020)

5. **Důraz na bezpečnost**

Hlavní výhodou systému Debian je jeho bezpečnost, kterou vývojáři berou vážně. Vývojáři pracují na všech problémech, které jsou nahlášeny do systému na sledování chyb a snaží se je vyřešit v přiměřeném časové době. Mnoho bezpečnostních opatřeních je koordinováno samotnými dodavateli svobodného softwaru a jsou uveřejňovány ve stejný den, kdy byla nějaká zranitelnost zjištěna a zveřejněna. Debian má i svůj bezpečnostní auditorský tým, který kontroluje, zda nebyly zjištěny nové zranitelnosti systému nebo sám nové zranitelnosti hledá. (DEBIAN, 2020)

Nevýhody

1. **Chybějící populární software**

Jak již bylo zmíněno výše, tak i pro Debian platí skutečnost, že některé populární programy nejsou pro Debian k dispozici, ale pro velkou většinu z nich existují alternativní programy, které dokážou nahradit nejlepší funkce právě chybějících programů.

2. **Konfigurace systému**

Instalace operačního systému Debian je velmi jednoduchá. Někteří lidé tvrdí, že instalace systému Debian je mnohdy lehčí jak instalace systému Windows. U Debianu je problém s pozdější konfigurací, která bývá složitá. Spousta konfigurací připojovaného hardwaru (např. tiskárny) by mohly být jednodušší a mohly by používat skripty, které by uživatele konfigurací prováděli. (DEBIAN, 2020)

3. **Pomalé vydávání nových verzí**

Silnou stránkou je také stabilita systému, která je ovšem náročná na vývoj, a proto je pak spousta softwaru pozadu oproti aktuální verzi systému. Tato cena za stabilitu systému se převážně projevuje v prostředí jádra kernelu a ve stolní počítačové verzi systému.

4. **Použití Systemd místo Init**

Systemd a Init jsou softwarové sady pro linuxově orientované operační systémy. Jsou to první procesy, které se spouští při bootování systému a jejich úkolem je sjednocení konfigurace služeb a procesů v systému. I přesto, že velká většina uživatelů akceptovala zavedení Systemd do Debianu, tak pořád existuje spousta lidí, kteří Debian za toto rozhodnutí odsuzují. Vidí Systemd jako příliš mocný administrativní nástroj a radši by nahradili Systemd za Init. Wiki stránka pro Debian nabízí návod jak Systemd nahradit za Init, ale tento proces je příliš složitý a než se touto záměnou zabývat, tak radši zvolí jinou linuxovou distribuce. (BYFIELD, 2017)

5. **Systém není příliš uživatelský přívětivý**

Systém není přímo určený pro nové začínající linuxové uživatele, a tedy ani nemusí být příliš uživatelsky přívětivý. Debian je určený pro vývojáře a administrátory, a proto systém nabízí pouze základní GUI (Graphical User Interface – Grafické Uživatelské Rozhraní), ale většina složitějších procesů se provádí v terminálu a noví uživatelé, kteří s terminálem doposud příliš nepracovali, budou mít se systémem problémy. (HASAN, 2018)

3.1.4.2 Ubuntu

Jako Debian, tak i Ubuntu je dalším volně dostupným operačním systémem z řad linuxových distribucích, který má svoji komunitu uživatelů a profesionální podporu od svých vývojářů. Ubuntu je založeno na myšlenkách, které jsou vepsaná v manifestu Ubuntu. Tyto myšlenky říkají, že jakýkoliv software by měl být volně a zdarma dostupný, že software by měl být dostupný v rodilé řeči, všech jeho uživatelů, a to i navzdory jakýmkoliv komplikacím, které by měli při překladu

softwaru nastat, a že by uživatelé měli mít možnost si svobodně upravit svůj software tak, aby naplňoval jejich potřeby. (PRAKASH, 2020)

Verze systému Ubuntu jsou dodávány uživatelům vždy ve stabilních a pravidelných intervalech, a to tak, že každá nová verze systému bude dodána každých 6 měsíců. Též existuje Ubuntu LTS (Long Term Support – Dlouhodobá Podpora) verze, která vychází každý druhý rok a její podpora trvá celých 5 let. Verze, které vyjdou mezi dvěma LTS verzemi, jsou nazývané non-LTS a jejich podpora trvá 9 měsíců. Ubuntu se plně zavazuje k principům vývoje softwaru jako open source a motivuje uživatele, aby open source software používali, vylepšovali a vylepšené verze sdílely mezi další uživatele.

Opět jako u systému Debian, tak pro Ubuntu existuje mnoho dostupných balíčků a softwarů, které si může uživatel nainstalovat (např. textové a tabulkové editory, internetové prohlížeče atd.) a je vhodný jak pro stolní počítače, tak pro serverové použití. (VERMA, 2018)

Výhody

1. **Pravidelné vydávání nových verzí**

Jak již bylo uvedeno výše, tak systém Ubuntu vydává nové verze pravidelně, a to lze brát jako velkou výhodu oproti ostatním linuxovým distribucím. Nejhorším příkladem ve vydávání nových verzí je již zmíněný Debian.

2. **Šifrování domovských adresářů**

Šifrování domovských adresářů je technika, která zvyšuje bezpečnost systému. Jejich nevýhodou je však to, že potřebuje určitou úroveň znalostí, a tedy ne každý uživatel může tuto výhodu systému využít. Ubuntu však byla první velkou distribucí, která tuto složitost šifrování snížila tím, že do systému přidala možnost šifrování pomocí zašktávacích políček a dala tak méně znalým uživatelům možnost lépe zabezpečit svůj počítač. (VERMA, 2018)

3. **Softwarové centrum**

Další výhodou systému Ubuntu je existence softwarového centra. Toto softwarové centrum nabízí svým uživatelům snadnou a rychlou instalaci nových softwarových aplikacích. Ubuntové softwarové centrum také svým uživatelům usnadňuje cestu k nalezení aplikací, které by vyhovovaly jejich konkrétním potřebám. (HOMEVAGANZA, 2019)

4. **Vhodnější pro začínající uživatele**

Na rozdíl od systému Debian, který je vhodnější pro pokročilejší uživatele, je Ubuntu lepší volba pro uživatele, kteří s linuxovými distribucemi začínají a ještě nejsou tolik zkušení, aby operační systém ovládali pomocí příkazové řádky, kterou na systémech jako Windows nebo MacOS běžný uživatel vůbec nevyužije. (HASAN, 2018)

5. **Kompletnost ovladačů**

Kromě problému s nedostatkem populárního softwaru se může uživatel setkat s dalším problémem a tím jsou ovladače. Některé softwarové ovladače mohou chybět anebo nemusí správně fungovat s uživatelským hadrwarem. Ubuntu má však podporu ovladačů, která se neustále rozšiřuje, a tak se uživatel nemusí obávat, že by jeho hardware nebyl kompatibilní s nainstalovaným ovladačem. (HOMEVAGANZA, 2019)

Nevýhody

1. **Ochrana osobních údajů**

Společnost Canonical, která vydává a sponzoruje Ubuntu, uzavřela dohodu se společností Amazon. Tato dohoda umožní Amazonu sbírat uživatelská data a zajistí funkci, že když uživatel bude vyhledávat na svém lokálním disku, tak se bude moci objevit výsledek vyhledávání z Amazonu. Canonical tvrdí, že by uživatelé měli svěřit svá data věrohodným společnostem jako je Amazon, ale základ ochrany dat spočívá nesvěřovat svoje data žádným společnostem.

2. **Vlastní rozhraní**

Většina linuxových distribucích využívá pro své uživatelské rozhraní GNOME (GNU Network Object Model Environment). I přesto, že se Ubuntu stále zaměřuje na technologii GNOME, tak si Ubuntu vytvořilo vlastní rozhraní Unity. Unity je svobodný software, ale pouze pár dalších distribucí ho přidali do svého repositáře. Ubuntu se tímto krokem více odděluje od linuxové komunity, která společně přispívá na vytvoření plnohodnotného svobodného softwaru. (BYFIELD, 2013)

3. **Vývoj řízený komerční společností**

Některé distribuce jsou spojovány s komerčními společnostmi, stejně jako Ubuntu a Canonical, ale tento vztah je většinou udržován v pozadí. Při vývoji Unity se často objevovaly případy, kdy designová rozhodnutí byla vytvářena Canonicalem, nežli obvyklým komunitním procesem. Díky těmto událostem vznikají obavy o firemních motivech společnosti Canonical.

4. **Reklamy**

Absence reklam je jedna z věcí, které mají stolní počítače s linuxovou distribucí společné. Ubuntu se svým rozhraním Unity však tyto reklamy začíná svým uživatelům předkládat. Tyto reklamy se týkají převážně služeb Ubuntu (např. Ubuntu One), ale začínající se objevovat i reklamy na Amazon a na jeho služby. Tyto reklamy anebo základní ochrana osobních údajů často bývají důvodem, proč lidé přecházejí na linuxové operační systémy.

5. **Pro zkušené uživatele nedostačující**

Mezi výhody systému Ubuntu patří, že je vhodnější pro uživatele, kteří přechází na Linux. Tato výhoda je však zároveň nevýhodou, protože spousta zkušenějších uživatelů tvrdí, že je Ubuntu nedostačující pro složitější úkoly, a raději než Ubuntu by upřednostnili např. zmiňovaný systém Debian. (IVANKOV, 2019)

3.1.4.3 CentOS

Distribuce CentOS je vyvíjena skupinou Projekt CentOS. Tato skupina se skládá z vývojářů, kteří se snaží pomocí volného softwaru vytvořit platformu, která bude

svým uživatelům poskytovat skvělé služby a bude postavena na open source kódu. Distribuce nabízí frameworky pro poskytovatele cloudových služeb, zpracování velkých objemů vědeckých dat a mnoho další. (WOLFSHANT, 2020) Distribuce CentOS je založena na systému Red Hat Enterprise Linux (tzv. RHEL), který je od společnosti Red Hat, se kterou se Projekt CentOS spojil. RHEL je komerčně podporovaná verze linuxového systému a za jeho užívání musí uživatelé platit měsíční předplatné, na rozdíl od systému CentOS, který je pro své uživatele zdarma dostupný. CentOS a RHEL jsou dva velmi podobné systémy, avšak někteří uživatelé, kteří nevyžadují dodatečné komerční služby, se spokojí pouze s CentOS jako s bezplatnou alternativou. (KERNER, 2014)

Výhody

1. **Bezplatná verze Red Hat Enterprise Linuxu**

Jak již bylo uvedeno, tak CentOS je možný přirovnat k systému RHEL jako jeho bezplatná verze. Výhodou toho je, že jako malý podnikatel, anebo jen jako linuxový nadšenec, nemusí uživatel za jeho užívání platit, a i tak se mu dostane podobného produktu jako při zaplacení RHEL. (KERNER, 2014)

2. **Dlouhodobá podpora**

Podpora pro uživatele systému CentOS je na délku 6 let. Systému je také poskytována bezpečnostní podpora až po dobu 10 let od prvního vydání. Dlouhodobá podpora systému je převážně důležité pro softwarové vývojáře, kteří častokrát musí provádět změny v programu pro různé operační systémy.

3. **Stabilita systému**

Podobně jako u ostatních distribucí, tak i CentOS poskytuje stabilní systém a společně se správnou konfigurací a kvalitním hardwarem zajistí svým uživatelům hladký chod serveru. (SESE, 2016)

4. **Poskytované manuály**

Projekt CentOS se snaží přinést svým uživatelům co nejvíce výhod a usnadnit jim tak jejich práci a proto také poskytují knihovnu vlastních manuálů, které za svojí práci

sepsali. Tato knihovna obsajuje například manuál, který uživatele provede celým postupem instalace. Projekt CentOS se tak snaží pomoct novým uživatelům systému.

5. **Rostoucí komunita**

Systém CentOS nemá velmi velkou komunitu jako některé jiné linuxové distribuce, ale i přesto má stále své věrné uživatele a těch neustále přibývá. Tato komunita se spojila a vytvořila Speciální Zájmovou Skupinu, která se často označuje zkráceně SIGs (Special Interest Group – Speciální Zájmová Skupina). Tato skupina se zabývá vylepšováním systému v oblastech grafické rozhraní, virtualizace atd. (ELIOT, 2019)

Nevýhody

1. **Zastaralé balíčky**

Mimo bezpečnostní částí systému je spousta balíčků nabízených v CentOS zastaralá. Na výběr jsou již vyzkoušené a stabilní verze balíčků, ale uživatelovi někdy mohou chybět potřebné funkce. Je zde možnost manuálně softwarové balíčky aktualizovat, ale tento proces je náročný a zdlouhavý.

2. **Nevhodný pro desktopové rozhraní**

CentOS nabízí i vlastní uživatelské rozhraní, které ovšem není tak efektivní jako uživatelská rozhraní u jiných linuxových distribucí. Z tohoto důvodu se CentOS hodí spíše na serverové využití než na stolním počítači.

3. **Slabší uživatelská podpora**

I přestože jsou systémy RHEL a CentOS často spojovány dohromady, tak mají spoustu vlastností, u kterých se rozchází, a to například u poskytované podpory uživatelům. CentOS má vlastní podporu, ale není tak velká jako u RHEL jak by spousta uživatelů čekalo. (SESE, 2016)

4. **Méně vhodný pro začínající uživatele**

Jak již bylo zmíněno, tak CentOS nabízí pouze slabší uživatelské rozhraní, a tedy pokud by chtěl uživatel využít všechny funkcionality, které systém nabízí, musí uživatel umět ovládat systém skrze terminál. (TARAFDER, 2019)

5. Není to RHEL

Jak už bylo uvedeno výše, tak se RHEL a CentOS můžou zdát jako dva podobné systémy, ale v závěru je RHEL lepší systém, než je CentOS. Takže pokud si uživatelé zvyknou na CentOS, ale už jim nebude vystačovat, tak budou muset přejít k placenému systému RHEL anebo se budou muset naučit pracovat s jinou linuxovou distribucí. (KERNER, 2014)

3.2 Bezpečnostní rizika

Za kybernetickou hrozbu lze považovat jakoukoliv zlomyslnou činnost, spáchanou za pomocí výpočetní techniky. Mezi cíle kybernetických útoků patří například ukrást či poničit data, vypnout napadené počítače anebo použít napadené počítače pro další útok. (TUNGGAL, 2020)

V následujících podkapitolách byly popsány nejčastější typy bezpečnostních rizik, které představují pro server hrozbu. Popsané kybernetické útoky a malware byly vybrány na podkladě seznamu o nejčastějších typů útoku. Seznam byl zveřejněn na blogu Netwrix. (MELNICK, 2020)

3.2.1 Kybernetické útoky

Mezi útočníky, kteří se provádí kybernetické útoky, se může řadit velká řada lidí. Může se mezi ně řadit např. teroristické organizace, vlády různých zemí anebo jen obyčejné skupiny lidí či samotní jednotlivci. Tito útočníci jsou často označováni jako hackeři, a někdy i jako crackeři. I když spousta lidí chápe oba výrazy jako sobě rovné, tak existuje rozdíl mezi hackerem a crackerem. (EDUCBA, 2016)

Hacker je člověk, který má dobré znalosti v dané oblasti počítačové bezpečnosti. Neustále hledá různé chyby v systémech a snaží se najít nové způsoby, jak využívat různé technologie. Hacker je spíše technologický dobrodruh a tvůrce nových funkcionalit nežli někdo, kdo se snaží ukrást důvěrná data. Samozřejmě existují hackeři, kteří svoje znalosti pro protizákonné operace využívají.

Cracker je spíše někdo, kdo se snaží prolomit veškerá bezpečnostní opatření za účelem osobního zisku nebo jen poškození cílové osoby (či společnosti). Na rozdíl od hackera, nemá cracker takové znalosti jako hacker a převážně se spoléhá na nástroje, který vytvořil někdo jiný. Veškerá aktivita, kterou cracker provádí, je nelegální. (JELEN, 2019)

3.2.1.1 DoS

Útok DoS (Denial of Service – Zamítnutí služby) má za úkol překazit normální činnost počítače či serveru a zabránit mu tak, aby mohl nadále poskytovat svoje služby svým cíleným uživatelům. Tento typ útoku funguje na principu zadávání velkého množství požadavků na cílený server, dokud není schopen zvládat normální provoz a dojde tak k zamítnutí uživatelských požadavků. Tento typ útoku je charakteristický tím, že se k útoku využívá pouze jedno zařízení.
(MCCOLLIN, 2020)

Útoky DoS jsou nejčastěji prováděny dvěma způsoby. Prvním z nich je **zaplavení vyrovnávací paměti** (Buffer overflow attack). Při tomto útoku se na cíleném serveru spotřebuje veškerá paměť na pevném disku, paměť RAM a CPU čas. To vede k pomalému chování systému serveru anebo přímo k jeho selhání. Druhým způsob je označován jako **povodeň** (flood attack). Útočník nasytí server nadměrným množstvím paketů což vede k zamítnutí služeb. (KEARY, 2020)

Pro provedení DoS útoku lze využít nástroj LOIC (Low Orbit Ion Cannon – Iontové dělo na nízké oběžné dráze), který slouží k testování zatížení sítě. Jedná se o open source nástroj napsaný v jazyce C#. (HUNT, 2013)

3.2.1.2 DDoS

Jedná se o podobný typ útoku jako DoS, s tou výjimkou, že DDoS (Distributed Denial of Service – Distribuované zamítnutí služby) je charakteristický tím, že k útoku se využívá více zařízení najednou, a tato síť zařízení se nazývá botnet. (PETTERS, 2020)

Botnet se skládá z jednotlivých botů, což může být jakékoliv zařízení, které má vlastní výpočetní výkon a může se připojit na internet. Útočník jednotlivé boty může získat různými způsoby. Může například využít vlastní počítače či servery anebo pomocí malwarů napadnout cizí počítače a ty pak zařadit do svého botnetu. (LUTKEVICH, 2019)

Mezi varianty DDoS útoků patří **HTTP Zatopení** (HTTP Flood), které funguje na principu znovuobnovování webové stránky z různých zařízení. (PETTERS, 2020)

3.2.1.3 Prolomení hesla

Jedná se o základní typ útoku, při kterém se útočník snaží získat přístup k uživatelovu účtu. Existuje mnoho způsobů, jak získat uživatelská hesla. Mezi tyto způsoby patří například fyzické sledování cílené osoby a vypozorování jejich hesla, lze využít sociálního inženýrství anebo za pomoci „hrubé síly“ vyzkoušet velké množství hesel (může se jednat o seznam často používaných hesel nebo jen o vygenerované kombinace různých znaků) či se heslo pokusit uhodnout. (MELNICK, 2020)

3.2.1.4 SQL injekce

SQL injekce využívá dotazovacího jazyka SQL, který se používá pro správu relačních databází. Útok je cílený na formulářové prvky internetových stránek, které nemají svoje vstupy z formulářů nijak jištěné.

Princip útoku spočívá v tom, že útočník vloží např. do přihlašovacího formuláře části SQL dotazu, které se po odeslání formuláře přidají do dotazu internetových stránek a vytvoří tak dotaz, který může útočníkovi vypsat veškerá data o uživatelích.

Dotazy pro SQL injekci mají často podobný tvar jako ***‘ or ‘1’=’1***, které budou v SQL dotazu vracet hodnotu *TRUE* a útočník se tak může dostat k potřebným údajům aniž by znal uživatelská jména nebo hesla. (FORMÁNEK, 2018)

3.2.2 Malware

Malware je kombinace slov *malicious software* (přeložena jako *škodlivý software*) a jak tedy název napovídá, tak se jedná o počítačový software, který má za úkol poškodit cílový systém. Na rozdíl od kybernetického útoku, který se ve své podstatě provede ihned, musí malware čekat, než si ho uživatel stáhne a spustí. Způsobů, jak se malware dostane do systému je mnoho počínaje od stáhnutí nelegálních programů až po využití sociálního inženýrství, kdy jsou uživatelé „přemluveni“, aby si malware dobrovolně stáhli a spustili. (REGAN, 2019)

3.2.2.1 Zadní vrátka

Zadní vrátka by se dala považovat za nezdokumentovaný tajný přístup do zabezpečeného počítačového systému bez nutnosti znalosti potřebných přístupových údajů.

Zadní vrátka se do systému mohou dostat jako nechtěný software, tedy formou trojského koně, ale také jsou často zadní vrátka implementována samotnými vyvojáři systému z důvodu možných budoucích oprav a zásahů od systému. To znamená, že ne vždy jsou zadní vrátka vytvářena za účelem poškození počítačového uživatele. Toto ovšem přináší riziko, že nepovolená osoba zadní vrátka najde a zneužije. (ZACKS, 2018)

3.2.2.2 Rootkit

Jedná se o velmi nebezpečný druh počítačového viru. Jedná se o sadu počítačových nástrojů, díky nímž se malwaru daří zůstat skrytý na uživatelovi počítači a umožňuje mu získávat administrátorská práva k softwaru. Pomocí rootkitu může útočník ukrást různé informace o uživateli (např. hesla) anebo vypnout bezpečnostní programy počítače. (ROUSE, 2018)

Mezi typy rootkitů například patří:

- **Kernelový rootkit**, který napadá jádro počítače a umožňuje útočníkovi měnit tak jeho chování.

- **Bootloader (zaváděcí) rootkit** napadne počítač tak, že nahradí původní zaváděcí program svým upraveným programem. To tedy znamená, že rootkit naběhne dříve než uživatelům operační systém.
- **Aplikační rootkit** je zaměřená na aplikace v počítači. Útočník tak může měnit některé funkce napadených programů.

Rootkit se nedokáže šířit sám, a proto se k uživatelovi může dostat např. pomocí phishingových emailů, nakažených dokumentů nebo nakažený programem. (RAFFER, 2019)

3.2.2.3 Spyware

Činností spywaru je v uživatelově počítači sbírat jeho osobní data a data o pohybu na internetu, které malware později předá buď inzerčním společnostem nebo cizím neoprávněným osobám. Hlavním úkolem spywaru ovšem často bývá zachytit uživatelova čísla platebních karet nebo přihlašovací údaje k bankovnictví.

Spyware se do uživatelova počítače nejčastěji dostane např. po stáhnutí neznámé přílohy z phishingového emailu, skrz ilegální stahování filmů, hudby či počítačových her nebo stahování souborů z pochybného internetového zdroje.

Mezi jiné podoby spyware patří také **adware**, který zobrazuje nechtěné reklamy nebo také již zmíněný **trojský kůň**. (REGAN, 2020)

3.2.2.4 Červ

Počítačový červ je typ malwaru, který sám dokáže dělat vlastní kopie a ty šířit mezi ostatní počítače. Červ se může na uživatelův počítač dostat buď jako součást jiného softwaru nebo jako příloha phishingového emailu atd.

Mimo soběstačného šíření, umí červ upravovat či vymazávat soubory na počítači a také v počítači šířit jiný druh nebezpečného malwaru. Existují i případy kdy červ slouží pouze zaplnění a vyčerpání výpočetních zdrojů počítače.
(FRUHLINGER, 2019)

3.3 Zabezpečení serveru

Způsoby, kterými se zabezpečí server, se mohou lišit podle toho na co je daný server využíván. Základní nastavení pro správu uživatelů, konfigurace firewallu a antivirového programu se ovšem tolik mezi jednotlivými typy serverů neliší.

3.3.1 Metodika zabezpečení serveru

Pro zvýšení zabezpečení by se měl server neustále kontrolovat zdali neexistuje nějaká nová aktualizace samotného systému nebo služeb a programů, které jsou na serveru nainstalovány a využívány. Přestože mohou nové aktualizace obsahovat nové zranitelnosti, tak přesto se většin potenciálních útočníků bude pokoušet server napadnout spíše již známými existujícími zranitelnostmi. (MULLINS, 2019)

Pokud se v systému objevují nepotřebné balíky a služby, tak na místo jejich neustálého aktualizování, tak bude lepší je odinstalovat. Pokud se administrátor serveru zpětně rozhodne využívat odinstalované služby, tak je může kdykoliv nově nainstalovat. Musí se však dávat pozor, aby se omylem neodstranila nějaká důležitá služba či balík, které jsou využívány pro správný chod systému. (DAVYDOV, 2020)

3.3.2 Uživatelské účty

Správa uživatelů na serveru zahrnuje, že všichni uživatelé budou mít přístupy ke svým souborům a budou moci vykonávat operace, které jsou důležité pro jejich každodenní činnost. Stejně důležité je i zabránění neoprávněným uživatelům přistupovat k souborům cizích uživatelů nebo spouštět operace, ke kterým jim nebyla udělena patřičná práva. (TEVAULT, 2018)

3.3.2.1 Založení uživatelského účtu

Každý uživatel má své uživatelské jméno a heslo, které slouží pro jeho identifikaci při přihlašování do systému. Při vytváření nového účtu je každému uživateli přidělen UID (User identifier – Identifikátor uživatele) a GID (Group identifier – Identifikátor skupiny). Tyto identifikátory se starají o bezpečnost souborů a ukazují, kteří uživatelé řídí aktuálně spuštěné procesy. (NEWELL, 2020)

Všechny účty v systému mají vlastní položku v souboru **passwd**. Každý tato položka je ve tvaru **uživatel:heslo:UID:GID:komentář:domovský_adresář:shell**, kde:

- **Uživatel**, je uživatelské jméno
- **Heslo**, zašifrované uživatelské heslo
- **UID**, je číslo reprezentující ID uživatele
- **GID**, je číslo reprezentující ID skupiny daného uživatele
- **Komentář**, je informace o uživateli. Často zde bývá uvedeno celé jméno uživatele.
- **Domovský adresář**. V této části je napsaný domovský adresář uživatele.
- **Shell**. V této části je napsaný přihlašovací shell uživatele. (HUNT, 2003)

3.3.2.2 Root a sudoers uživatelé

Hlavní administrátorský účet v linuxových distribucích, který může v systému provádět veškeré operace, se nazývá **root**. Přestože root uživatel má veškerá oprávnění, tak se nedoporučuje ho používat pro každodenní užívání systému, a to z toho důvodu, že lze nedopatřením nebo úmyslně poničit některá data nebo rovnou celý systém. (TEVAULT, 2018)

Z tohoto důvodu existuje funkce zvaná **sudo**. Funkce sudo umožňuje obyčejným uživatelským účtům vykonávat úkoly, které by mohl vykonávat pouze root uživatel. Aby však uživatel mohl tuto funkci využívat, tak musí patřit mezi tzv. **sudoers** uživatele. Existují dva základní způsoby, jak se může uživatelský učet dostat do skupiny sudoers a získat tak práva na úrovni root. (ELLINGWOOD, a další, 2020)

Jedním způsobem je přidáním uživatele za pomocí příkazu v terminálu a druhým způsobem je ruční přidání uživatelského účtu do souboru sudoers. Pro vykonání těchto dvou způsobů je zapotřebí samotný root nebo uživatel, který již je zapsaný mezi sudoers. Sudoers soubor obsahuje seznam všech uživatelů, kteří mohou využít administrátorských práv jako root. Mezi další způsoby patří vytvoření administrátorské skupiny s vysokými práva a poté přidáním vybraných uživatelů. (TERZI, 2018)

3.3.2.3 Metodika zabezpečení uživatelských účtů

Jednou z nejdůležitějších věcí při zakládání nových uživatelských účtů je, aby při tvorbě uživatelského hesla zároveň probíhala kontrola složitosti hesla. Hesla by měla být dostatečné dlouhá, obsahovat speciální znaky a nejlépe by neměla obsahovat žádná obyčejná slova. (BRANDALL, 2018)

Dále by se měl vytvořit již zmíněný sudoer uživatel, který by se staral o správný chod serveru a zároveň se tak omezilo využívání root účtu na minimum (MULLINS, 2019). Správnou metodikou také je zakázání jakéhokoliv vzdálené přihlášení s root účtem. (HESS, 2019)

3.3.3 Firewall

Firewall se dá považovat za systém uvnitř uživatelova systému, který filtruje veškerý síťový provoz, přicházející z internetu do lokální sítě či naopak a nepropustí žádný nežádoucí provoz. O firewall v Linuxovém systému se stará framework **nftables**, který umožňuje překlad síťových adres, filtrování paketů a jejich další manipulaci. Tento framework byl vytvořen jakou součást projektu **Netfilter** a nahrazuje starší technologie jako **iptables**. (TEVAULT, 2018)

3.3.3.1 Seznamy pravidel firewallu

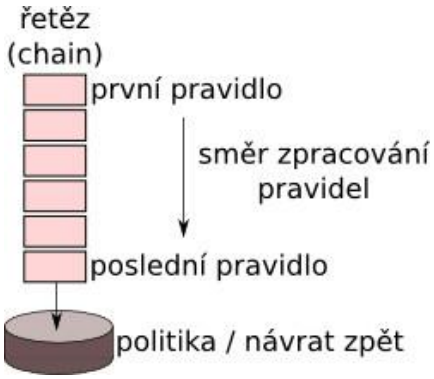
Provoz firewallu se dá rozdělit do tří skupin a na každou tuto skupinu se aplikují jiná filtrovací pravidla:

1. **Vstupní (input).** Nežli jsou datové pakety akceptovány, tak jsou nejdříve otestovaný vstupními pravidly firewallu.
2. **Postupující (forward).** Datové pakety, které dorazí do systému, ale nejsou pro něj určeny, jsou otestovány postupujícími pravidly firewallu.
3. **Výstupní (output).** Výstupní datové pakety, předtím, než jsou skutečně odeslány, jsou testovány výstupními pravidly firewallu. (GANESH, 2020)

Kernelové jádro uchovává pro tyto jednotlivé skupiny seznamy pravidel (V angl. jazyce je tento seznam pravidel nazýván *chain* a proto jsou tyto seznamy někdy označovány jako *řetězy*) a s těmito seznamy lze pracovat pomocí iptables.
(HUNT, 2003)

3.3.3.2 Řetězy

Seznamy pravidel lze představit jako řetězec pravidel, kdy datový paket projde postupně všemi pravidly a je kontrolován, zda pravidlu vyhovuje nebo nevyhovuje.



Obrázek 1 - Řetěz pravidel (DOČEKAL, 2010)

Pokud bude paket některému z pravidel vyhovovat, tak nastane jedna ze čtyř možností. Paket buď bude systémem přijat, odmítnut, zahozen či bude předán jinému řetězu.

Nastane-li situace, že paket nevyhovuje žádnému pravidlu řetězu, tak je paket buď přijat nebo odmítnut, pokud by se jednalo o původní řetěz firewallu. Pokud by se jednalo o řetěz definovaný uživatelem, tak je paket přeposlán zpět odkud do řetězu přišel. (DOČEKAL, 2010)

3.3.3.3 Metodika zabezpečení firewallu

V konfiguraci firewallu by měli být uvedeny porty, které server skutečně ve svých službách využívá. Ostatní porty by měli být uzavřené, aby nenastala situace, že na nějakém otevřeném portu bude zapnutá nezabezpečená služba, která by mohla mít za následek napadení serveru útočníkem. (DAVYDOV, 2020)

Firewall chrání systém před vnějšími útoky, a proto by měl server blokovat veškerá neznámá spojení a povolit spojení jen s ověřenými zdroji. Správné nastavení a využívání anti-IP-spoofing je pro server velmi důležité z důvodu obrany proti hackerský útokům. Mnoho útoků využívá napodobující pakety (z angl. spoofed packets) přičemž nejznámější z nich je útok DoS. Útočník se snaží pomocí

napodobujících paket zamaskovat svůj původ nebo předstírat, že jsou pakety z ověřeného zdroje. (GANESH, 2020)

Pokud nastane situace, že firewall neví, jak naložit s daným paketem a pro paket neexistuje žádné filtrovací pravidlo, tak by měl firewall být nastaven, aby podobné pakety automaticky zahazoval. (NORONHA, 217)

3.3.4 Webový server

Pro webové servery je jedním z nejpoužívanějších softwarů Apache, který je zdarma a snadno se stáhne a nainstaluje. V dnešní době už často bývá, že je Apache součást základní podoby operačního systému. Apache do systému nainstaluje HTTPD (Hypertext Transport Protocol Daemon – Démon Hypertext Transfer Protocol), který slouží pro vytvoření webového serveru. (HUNT, 2003)

Apache se v minulosti potýkala s mnoha zranitelnostmi, ale vždy, co se zranitelnost objevila, byla rychle opravena a následně byla vydaná nová verze softwaru. V posledních letech je většina bezpečnostních rizik způsobena spíše špatnou konfigurací serveru, nežli samotným softwarem. (BAUER, 2005)

3.3.4.1 Apache

Konfigurace Apache je velmi snadná, jelikož software je již po instalaci sám nakonfigurován a připraven k použití. Pouze se musí provést pár úprav v konfiguračním souboru **httpd.conf**, který obsahuje informace o administrátorovi webového serveru, název serveru, umístění konfiguračních souborů, chybová hlášení serveru atd. Dále se musí nastavit, zda na Apache bude hostována pouze jedna nebo více webových domén. Při variantě vícero domén se bude muset nastavení serveru více rozpracovat a bude potřeba využít virtuálních hostů. (ŠENKYŘÍK, 2015)

Konfigurace Apache serveru by se dala rozdělit do tří primárních konfiguračních souborů:

- **httpd.conf**, který byl již zmíněný výše.

- **srm.conf**, který konfiguruje správu požadavků na serveru a určuje umístění HTTP dokumentů a skriptů.
- **access.conf**, který určuje řízení přístupů serveru a jeho poskytovaných informací.

Konfigurace těchto tří souborů se překrývá a jakékoliv nastavení může být uvedeno v jakémkoliv konfiguračním souboru. Proto se běžně stává, že administrátor nevyužívá všech tří souborů, ale pouze konfiguruje soubor httpd.conf. Tento přístup při využití pouze souboru httpd.conf je i doporučovaný. (HUNT, 2003)

3.3.4.2 SSL/TLS

Velká spousta webů v dnešní době uchovává o svých uživatelích osobní data, která je zapotřebí chránit. Pro zvýšení bezpečnosti a důvěryhodnosti webových stránek se využívá šifrovací technologie TLS (Transport Layer Security – Zabezpečení transportní vrstvy) (HUNT, 2003). TLS nahradil zastaralejší SSL (Secure Sockets Layer – Vrstva bezpečných soketů). Přestože TLS nahradil SSL, tak je SSL pořád někde využívána a také lidé tuto technologii stále označují jako SSL. (KINSTA, 2020)

SSL/TLS funguje ne bázi asymetrické kryptografie, aby zprostředkovala šifrovanou a bezpečnou komunikaci mezi klientem a serverem. Takže i kdyby útočník zachytil přeposílané datové pakety, tak je nebude moct nijak přečíst. (FRUHLINGER, 2019)

Začátek šifrované komunikace mezi klientem a serverem je zahájen pomocí funkce *TLS handshake* (TLS podání ruky). Během tohoto podání ruky si klient a server vygenerují relační klíče. Tyto relační klíče šifrují a odšifrují veškerou komunikaci po podání ruky. Při nové relaci klienta a serveru jsou generovány nové klíče. TLS zaručuje klientovi, že skutečně komunikuje s daným serverem a že data nejsou při přenosu nijak modifikována. (DOMANTAS, 2021)

3.3.4.3 Metodika zabezpečení webového serveru

Mezi kybernetické hrozby, které mohou hrozit webovému serveru patří například již zmíněný DoS a DDoS útok, nevědomá instalace trojského koně, ale třeba i zachycení

důležitých paketů, krádež citlivých a osobních údajů uživatelů či hromadné nahrání nevhodného obsahu (např. pornografie).

Webový server také musí zajistit ochranu integrity veškerých informací, které jsou odeslány serverem nebo klienty. Bezpečnost přístupu k těmto informacím je zajištěna řízením přístupu, které se konfigurují v souboru httpd.conf. Lze nastavit přístupy *host-level* (úroveň hostitele) a *user-level* (úroveň uživatele). (KALMAN, 2014)

Pro bezpečnou správu serveru je, jako první věc, důležité při konfiguraci nastavit co nejméně možná práva a snažit se vyhnout variantě, že procesy budou běžet jako root. Je důležité, aby konfigurace byla pro server co nejjednodušší.

Čím více nepotřebných programů, funkcí, účtů atd. bude odstraněno ze serveru, tak tím méně potenciálních bodů průniku bude pro útočníka existovat. A u těch funkcí a programů co na serveru zůstanou je důležité vždy zabezpečit ty, která přijímají jakýkoliv vstup od uživatele, protože každý vstup od uživatele musí být brán jako by šlo o potenciální útok na server. (TAMMANY, 2018)

V poslední řadě je důležité veškeré funkce, programy, správcovské účty, bezpečnostní mechanismy, konfigurační soubory a vlastně veškeré věci, které se na serveru vyskytují, dobře zdokumentovat. Dokumentace je důležitá, protože si uživatel/správce nebude pamatovat vše na čem na serveru dělal anebo pokud přijde nový správce serveru a bude chtít porozumět tomu, jak dané věci fungují, tak se bude moci podívat do vypracované dokumentace. (KUMAR, 2020)

3.3.5 FTP

FTP (File Transfer Protocol – Protokol pro přenos souborů) je jedním z nejstarších protokolů z rodiny TCP/IP protokolů, který se stále využívá pro přenášení souborů mezi počítači po internetu. Tento protokol umožňuje přístup ke vzdáleným adresářům a jeho sub-adresářům. (MARTINDALE, 2020)

3.3.5.1 Metodika zabezpečení FTP

Jelikož se jedná o starší protokol, tak se FTP nedá považovat za velmi bezpečný. Mezi bezpečnostního rizika protokolu se řadí například nešifrování přihlašovacího jména a hesla nebo zranitelnost zachycení přeposílaných datových paketů. Existuje způsob, jak přenos pomocí FTP zabezpečit a to tak, že do něj přidá SSL/TLS šifrování a tím nám vznikne SFTP (Secure File Transfer Protocol – Bezpečný protokol pro přenos souborů). (LORD, 2018)

Samostatné FTP se dá stále využít jako **anonymní FTP**. Jedná se o metodu, která slouží pouze k zobrazení adresáře na serveru a ke stahování souborů z něj. Lze umožnit i nahrávání souborů do přednastavené složky. Anonymní FTP je anonymní, protože uživatelé se nemusejí nijak identifikovat (proto by však měla mít omezená práva pouze na stahování souborů). (BEAL, 2020)

Existuje pár principů, které pomáhají zabezpečit chod FPT:

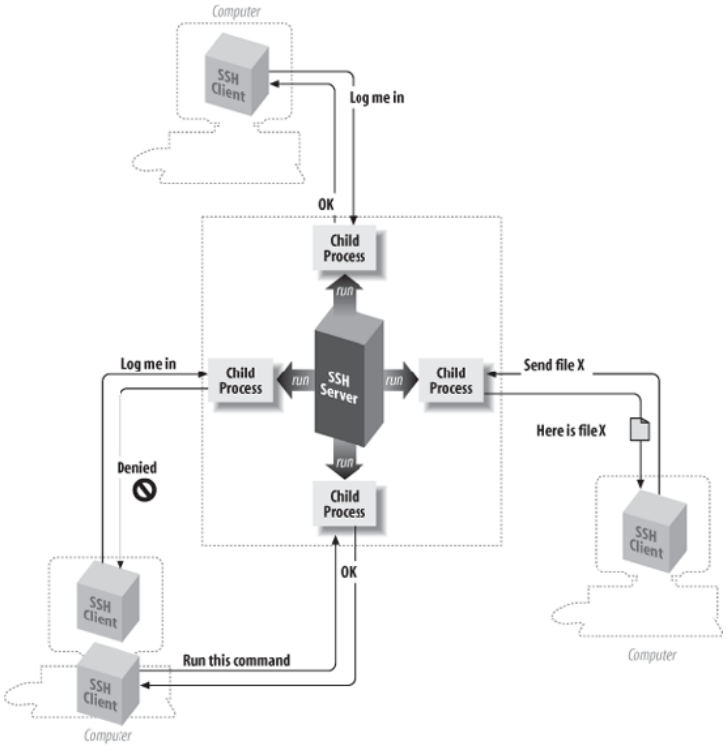
- Pro co největší zabezpečení by měl být FTP, pokud je to možné, spuštěn bez jakýchkoliv práv.
- Pro správu anonymního FTP by měl být využit *nefunkční shell*, který by uživatelům bránil v zasahováním do funkcionality serveru.
- Mělo by být vytvořené *chroot vězení* pro anonymní uživatelé. **Chroot** je proces, který změní používaný kořenový adresář pro aktuálně spuštěné programy. Tyto programy pak nemohou přistupovat k souborům mimo tento uvedený adresář. Takto omezené prostředí se nazývá *chroot vězení*.
- Nedovolujte anonymním uživatelům nahrávat soubory, pokud to není opravdu nezbytné. (BADJATIYA, 2019)

3.3.6 SSH

Jednou z nejčastějších využívaných služeb na serverech je SSH (Secure Shell – Bezpečný shell), která umožňuje uživatelům připojit se na server a komunikovat s ním prostřednictví vzdáleného terminálu. Jedná se o výkonný nástroj pro zabezpečení síťové komunikace. Když uživatel začne posílat nějaká data, tak je SSH zašifruje. Poté co data dorazí na cílený počítač, tak je SSH zase dešifruje. SSH

funguje na architektuře klient/server. **SSH server** přijímá či odmítá příchozí požadavky na server. (BARRETT, a další, 2003)

Uživatel pracuje s programy označované jako **SSH klienti**, ti jsou obvykle umístěné na jiném počítači a pomocí nich posílají požadavky na SSH server.



Obrázek 2 - SSH architektura (BARRETT, a další, 2003)

Co bývá zavádějící, že se v názvu objevuje *shell*, i přestože se o žádný skutečný shell nejedná. To znamená, že SSH neinterpretuje uživatelovi příkazy. Spíše se SSH dá definovat jako bezpečnostní kanál pro ovládání shellu na vzdáleném serveru/počítači. (FLICKENGER, 2003)

3.3.6.1 Protokol SSH

SSH je ve skutečnosti protokol, který se stará o pár věcí, aby zajistil bezpečnou komunikaci přes internet. Protokol se stará o **autentizaci**. Pokud se někdo snaží přihlásit k uživatelskému účtu, tak SSH nejprve vyšle požadavek o potvrzení účastníkovi identity a na základě tohoto testu rozhodne, zda bude přihlášen nebo ne. Jak bylo uvedeno výše, tak se protokol dále stará o **šifrování** dat a zajištění, tak

bezpečné komunikace. V poslední řade protokol zaručuje **integritu**. Zaručuje, že data dorazí od odesílatele k jejich příjemci bez jakékoliv změny. SSH také pozná, zda data byla během přenosu někým zachycena. (LADIA, 2017)

3.3.6.2 Vlastnosti SSH

Výše již bylo uvedeno, že SSH zajišťuje bezpečnou komunikaci, ale to není jediná jeho vlastnost, kterou se SSH vyznačuje. Tento protokol má mnoho vlastností, ale zde pro ukázku byla vybrána pouze ta nejpodstatnější.

1. **Posílání souborů**

Pro přenos souborů byl již uvedený FTP či jeho bezpečnější podoba SFTP. Bezpečné posílání souborů mezi uživateli jde však provést i za pomocí SSH díky funkci bezpečného kopírování. (BARRETT, a další, 2003)

2. **Klíče a agenti**

Při práci s vícero účty není doporučováno používat stejná hesla pro každý účet. Aby si uživatel nemusel všechny přihlašovací údaje pamatovat, tak SSH nabízí autentizační variantu, která je založena na *klíčích*. Klíče jsou krátké posloupnosti bitů, které definitivně rozpoznají každého uživatele SSH. Tyto klíče jsou uloženy v zašifrované podobě a jsou použitelné až po zadaní *přístupové fráze*. Díky klíčům a programu *autentizační agent*, SSH bezpečně autentizuje uživatele bez nutnosti přihlašovacích údajů. (ELLINGWOOD, 2014)

Existují dva druhy klíčů, jeden je soukromý a druhý veřejný. Soukromý je uložený u uživatele na straně klienta a měl by zůstat skrytý před všemi ostatními uživateli. Veřejný klíč je neškodný a každý ho může znát. Veřejný klíč tedy bude uložen na straně serveru a bude šifrovat data. Šifrovaná data dokáže dešifrovat pouze soukromý klíč. (SVERDLOV, 2012)

3. **Řízení přístupu**

Pomocí SSH může majitel konkrétního uživatelského účtu umožnit přistup jinému uživateli na majitelův účet. Uživatel, který obdrží přístup na cizí účet, nemůže

zobrazit ani změnit hesla účtu, ale bude moct spouštět a číst soubory, které by hlavní majitel účtu mohl. (LADIA, 2017)

3.3.6.3 Metodika zabezpečení SSH

Při práci s SSH budou uživatelé chtít využívat pro autentizaci svá uživatelská jména a hesla. Pokud uživatelé dodržují politiku o složitosti hesla, tak by dána metoda autentizace měla být v pořádku avšak i přesto se doporučuje používat pro autentizace již zmíněné klíče. Použití klíčů není tolik uživatelsky přívětivé, ale zaručuje vyšší bezpečnost při používání služby SSH. (DAVYDOV, 2020)

3.4 Monitorování serveru

Pro zajištění bezpečného chodu serveru je zapotřebí ho monitorovat a zaznamenávat různé procesy. Pro správné monitorování systému pomáhají soubory systémových protokolů. Systémové protokoly (z angl. system logs) pomáhají při řešení problémů ať už některého z procesů nebo při opravě samotného systému. Z těchto protokolů se též dají rozpoznat první známky napadení systému nebo jeho zneužití k neoprávněné činnosti. (PLESKY, 2018)

3.4.1 Syslog

Syslog je záznamový protokol, který dokáže pořizovat záznamy a odesílat systémové zprávy z činnost kernelu, všech procesů (až už lokálně nebo na vzdáleném systému). Také je velmi přizpůsobivý a umožňuje administrátorovi serveru nastavit co a kdy se bude v systému zaznamenávat. (ALTVATER, 2017)

3.4.1.1 Kódy závažnosti

Následující tabulka obsahuje seznam kódů závažnosti a co daná závažnost znamená.

Kód	Závažnost	Význam
0	Nouzová	Systém je nestabilní
1	Poplach	Potřeba okamžitého jednání
2	Kritická	Kritická závažnost
3	Chyba	Chybová zpráva

4	Upozornění	Upozorňující zpráva
5	Oznámení	Významný stav
6	Informace	Informační zpráva
7	Debug	Opravit chybu

Tabulka 1 - Kódy závažnosti (DOOLEY, 2020)

V praxi se však často nestává, že by administrátor obdržel zprávu s nízkým kódem závažnosti, jelikož je systém v příliš špatném stavu na to, aby zprávu odeslal. Zprávu si maximálně uloží do záznamového protokolu, kde si ji administrátor může později přečíst. (DOOLEY, 2020)

3.4.1.2 Kódy vybavení

Tyto kódy by se dali představit jako kategorie. Dřívější verze syslog ukládala všechny zprávy do jednoho záznamového protokolu a *kódy vybavení* sloužily pro jejich roztřídění. V dnešní době jsou systémové zprávy rozdělovány do jednotlivých databází a tyto kódy slouží pouze jako klíčová slova k vyhledávání.
(KERNEL, 2020)

Kód	Vybavení
0	Kernelové zprávy
1	Zprávy na úrovni uživatele
2	Poštovní systém
3	Systémové procesy
4	Bezpečnostní a autorizační zprávy
5	Zprávy generované samotným syslog
6	Subsystém řádkové tiskárny
7	Subsystém síťových novinek
8	Subsystém UUCP (Unix to Unix Copy Protocol)
9	Démon hodin
10	Bezpečnostní a autorizační zprávy
11	Démon FTP

12	Subsystém NTP (Network Time Protocol)
13	Audit protokolu
14	Upozornění protokolu
15	Démon hodin
16-23 **(local0 – local7)**	Lokální použití 0 – Lokální použití 7 (Local use 0 – Local use 7)

Tabulka 2 - Kódy vybavení (DOOLEY, 2020)

3.4.1.3 Nástupci služby syslog

Při správě linuxových serverů se často lze setkat se třemi různými nástroji pro zaznamenávání systémových zpráv. Jedná se o již zmíněný *syslog*, ale i o *syslog-ng* a *rsyslog*. Ve své podstatě jsou všechny 3 nástroje stejné, ale i přesto se jedná o tři rozdílné projekty, kdy se každý projekt snaží vylepšit ten předchozí.

Prvním z těchto projektů byl právě *syslog*, který byl na počátku jednoduchý a podporoval UDP (User Datagram Protocol – Protokol uživatelského datagramu), který nezaručoval úspěšný přenos zpráv. (ECHEVERRI, 2015)
Projekt, který se snažil vylepšit *syslog*, byl projekt *syslog-ng*, který je dnes nejrozšířenějším záznamovým protokolem. Mezi nejdůležitější funkcionality patří využívání TCP, namísto UDP, pro přenos zpráv a využití TLS pro šifrování. Dále zde bylo zavedeno zaznamenávání zpráv do databáze a možnost filtrování obsahu.

Posledním projektem je *rsyslog*, který se též snažil vylepšit *syslog* přidáním nových funkcionalit. Mezi tyto funkcionality patří například využívání protokolu RELP (Reliable Event Logging Protocol – Spolehlivý protokol pro zaznamenávání událostí). (PLESKY, 2018)

3.4.2 Detekce vniknutí

Existuje mnoho hrozeb, které mohou uškodit serveru tím, že napadnou a vniknou do jeho sítě. Pokud taková situace nastane, je důležité být s tímto faktem obeznámen a začít podnikat patřičné kroky. K detekci vniknutí se používají nástroje IPS (Intrusion Prevention System – Systém prevence průniku) a IDS (Intrusion Detection

System – Systém detekce průniku) (též někdy označován jako NIDS (Network Intrusion Detection System – Systém detekce průniku sítě)). (TEVAULT, 2018)

IPS i IDS jsou se v základu podobné, protože oba tyto systémy čtou veškeré datové pakety, které projdou sítí, a zkoumají zda-li se nejedná o některou z průnikových hrozeb. Rozdíl mezi těmito systémy je v tom, jak naloží s pakety, které ohodnotí jako nebezpečné. (PETTERS, 2020)

3.4.2.1 IDS

IDS je označován jako předchůdce IPS. Pokud IDS narazí na některou z předdefinovaných anomálií, tak tento problém nahlásí administrátorovi serveru. Problém tohoto systému je, že je zapotřebí administrátora, který se na nahlášený problém podívá a rozhodne, jak bude daný problém řešen. Problém u IDS přichází, když se se kontroluje větší síťový provoz a administrátor sám nezvládá řešit všechny hlášené problémy. (PATEL, 2020)

3.4.2.2 IPS

Rozdíl mezi IPS a IDS je v tom, jak se systémy zachovají při nalezení problému v síťovém provozu. Oproti IDS má IPS *seznamy pravidel* a podle nich rozhoduje, jak naložit s potenciálně nebezpečnými pakety. U tohoto systému vzniká problém v tom, že musí mít neustále aktualizovanou databázi, která obsahuje co nejvíce informací, ohledně nebezpečných paketů. (PETTERS, 2020)

3.4.3 Detekce malwaru

Jak už bylo výše uvedeno, tak existuje mnoho škodlivého softwaru, který může ohrozit chod serveru či počítače, který obsahuje linuxový operační systém. Naštěstí lze využít z řady nástrojů, které pomáhají svým uživatelům nalézat v systému různé typy malwarů. Většina těchto nástrojů pro detekci malwaru je dostupná zdarma. (GEEKFLARE, 2019)

3.4.3.1 Antivirus

Mezi známé nástroje pro hledání malwaru patří **LMD** (Linux Malware Detect – Linuxová detekce malwaru). Pomocí předdefinovaných databází, které obsahují

informace o škodlivých souborech, je schopen nalézt mnoho různých typů malwaru. Pro rozšiřování své databáze využívá LMD data z IDS/IPS a díky tomu je schopen odhalovat více hrozeb systému.

Mezi další podobné nástroje se může řadit **ClamAV**, který též vyhledává různé druhy malwarů. Spousta uživatelů ho využívá pro soukromé účely, jako například skenování emailů a dalších osobních dokumentů. (BINNIE, 2016)

3.4.3.2 Zranitelnosti

Nástroj **Lynis** se v systému používá v systému nejen pro nalezení malwaru, ale snaží se najít bezpečnostní mezery a rizika, a také hledá slabá místa v konfiguračních souborech. Ovšem Lynis pouze nehledá slabá místa serveru, ale také podává report s návrhy, jak dané slabosti opravit či odstranit a učinit tak server bezpečnějším.

Dalším podobný nástroj na vyhledávání zranitelností je **OpenVAS**. Tento nástroj je určený spíše podnikům, aby nacházel veškeré zranitelnosti v jejich infrastruktuře. (GEEKFLARE, 2019)

Jednoduchým nástrojem je také **Debsecan**. Jedná se o sken systému, který kontroluje všechny nainstalované balíky v systému a porovnává je s již známými zranitelnostmi a také ukáže uživateli jestli již byla zranitelnost opravena anebo o jak závažnou zranitelnost se jedná. (NISSANKA, 2020)

3.4.3.3 Rootkit

Nástroj **Chkrootkit**, jak již název napovídá, slouží pro hledání rootkitů uvnitř systému. Pro svoje fungování využívá linuxových příkazů *strings* a *grep*. Nástroj se snaží v systému nalézt *konfigurační soubory* pro rootkit, *smazané zápisy* či *utajené zápisy* z protokolových souborů nebo *systémové sniffery*. Nevýhodou nástroje je to, že nijak nereaguje na nalezený škodlivý soubor či záznam. Administrátor si proto musí výstup z chkrootkitu uložit a postupně ho projít a všechny systémové hrozby odstranit. (MALENKOVICH, 2013)

Pro vyhledávání rootkitů se dále používá nástroj **rkhunter**. Nástroj zkontroluje všechny soubory, složky, moduly pro správu kernelu a také špatně nastavená oprávnění. (MITCHELL, 2019)

3.4.4 Audit

Pokud chce serverový administrátor mít detailní přehled o tom, co se na serveru děje za události, a který uživatel dané události provádí, tak je zapotřebí mít správně nakonfigurovaný auditní systém. Pro audit systému se využívá démon **auditd**. Auditd zaznamenává veškerou aktivitu, kterou mu administrátor pomocí *auditních pravidel* nakonfiguruje. Může například zaznamenávat, pokud se uživatel pokouší spustit či otevřít jemu neoprávněné soubory, zvládne zaznamenávat kdy se uživatelé přihlásí do systému a také lze zaznamenávat určené druhy příkazů a kdo je provedl (např. zaznamená pokaždé, když uživatel použije příkaz pro kopírování souboru). (JOHN, 2015)

4 Vlastní práce

V následující části diplomové práce budou aplikovány poznatky získané z teoretických východisek. Nejdříve bude vybrána nejvhodnější linuxová distribuce. Po nasazení vybrané distribuce bude server nakonfigurován a otestován, aby bylo ověřeno, že konfigurace systému proběhla správně.

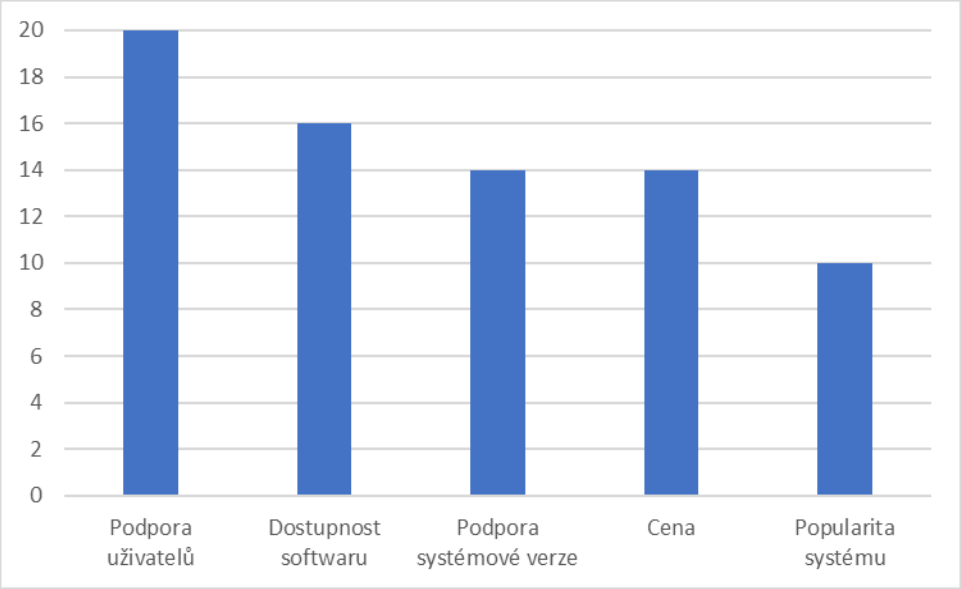
Konfigurace také vycházela z oficiální metodiky zabezpečením operačního systému Debian. Jednalo se o *Securing Debian Manual 3.19*. Tato oficiální metodika byla sepsána na starší verze Debianu a proto byl manuál použit spíše pro orientační postup.

4.1 Výběr linuxové distribuce

Než se proběhla bezpečnostní konfigurace serveru, tak byla provedena více kriteriální analýza pro výběr té nejlepší linuxové distribuce. Do analýzy byly vybrány požadavky na server formou dotazníku na uživatele. Požadavky byly u každé distribuce ohodnoceny a na daném hodnocení byla vybraná jedna linuxové distribuce.

4.1.1 Výběr požadavků

Pro výběr požadavků bylo dotázáno 20 osob, které již mají nějaké zkušenosti se systémem Linux. Jedinou podmínkou dotazníku bylo, že dotazované osoby mohli vypsat maximálně 4 požadavky.



Graf 1 - Požadavky na systém

Prvním požadavkem, který uvedlo všech 20 účastněných, byla *Podpora uživatelů*. Podpora uživatelů je důležitá hlavně pokud se jedná o méně zkušené uživatele. Druhým nejčastějším požadavkem byla *Dostupnost softwaru* a tento požadavek si vybralo 16 účastněných. Dalším požadavkem je *Délka podpory systémové verze*, kterou vybralo 14 účastněných. Předposledním požadavkem byla *Cena*, za systém a tento požadavek vybralo také 14 účastněných. Posledním zařazeným požadavkem byla *Popularita jednotlivých distribucí* a tento požadavek vybralo 10 účastněných.

Požadavky na výběr distribuce:

- Podpora uživatelů
- Dostupnost softwaru
- Podpora systémové verze
- Cena
- Popularita systému

4.1.2 Vyhodnocení analýzy

Pro výběr distribuce bude provedena vícekriteriální analýza za použití bodovací metody. Bodovací metoda funguje na principu bodovací stupnice, která bude v našem případě obsahovat 10 stupňů hodnocení, kdy 1 bod znamená nejhorší hodnocení a 10 bodů naopak nejlepší. (FIALA, a další, 1994)

	Debian	Ubuntu	CentOS
Podpora uživatelů	8	8	6
Dostupnost softwaru	9	8	6
Podpora systémové verze	7	7	9
Cena	9	7	7
Popularita systému	9	7	8
Aritmetický průměr	**42**	**37**	**36**

Tabulka 3 - Vícekriteriální analýza pro výběr distribuce (Zdroj: Vlastní)

Distribuce CentOS má necelých 90 000 balíčků, zatímco Debian a Ubuntu mají obě necelých 120 000. Přestože Debian a Ubuntu mají podobný počet dostupných balíčků, tak Debian obsahuje více tzv. „hlavních“ balíčků (jedná se o časté nástroje, které uživatelé používají). (SMITH, 2021)

Podpora uživatelů je u všech distribucí velmi podobná, kdy všechny distribuce mají fóra s uživatelskými požadavky. (ELIOT, 2019) Ovšem Ubuntu a Debian ještě nabízí aktivnější podporu v podobě IRC (Internet Relay Chat – Rozhovor přenášený po internetu) konzultací. (DEBIAN, 2020)

Debian a Ubuntu mají ve standartní podobě podporu až 5 let, zatímco CentOS může mít až 10 letou podporu verze. (BYFIELD, 2017)

Všechny výše zmíněné distribuce je možné získat zcela zdarma, ale pouze Debian lze stáhnout kompletně bez dalších poplatků, zatímco u distribuce Ubuntu existuje několik placených možností (WALLEN, 2017). CentOS je též jako Debian také zdarma, ovšem CentOS je znám jako neplacená varianta RHEL (KERNER, 2014).

Hodnocení pro distribuce bylo získáno ze stránky DistroWatch, kde uživatelé mohou hodnotit všechny existující linuxové distribuce. Na uvedených stránkách má Debian hodnocení 8,82, CentOS má 8,08 a Ubuntu 7,49.

Pomocí aritmetického průměru, který byl získán z vícekriteriální analýzy, bylo zjištěno, že nejlepší linuxová distribuce pro další postup bude distribuce Debian.

4.2 Příprava serveru

Pro bezpečnostní konfiguraci nebude použitý skutečný fyzický server, ale bude použitý virtuální server, který bude nainstalován v lokálním virtuálním prostředí. Pro vytvoření virtuálního serveru bude využit VirtualBox.

4.2.1 Virtuální prostředí

Jak již bylo zmíněno výše, tak virtuální server bude běžet na softwaru VM VirtualBox. Jedná se o open-source projekt od společnosti Oracle sloužící pro virtualizaci softwaru. VirtualBox je dostupný na všechny základní operační systémy jako je Windows, Linux či MacOS a umožňuje spouštět více operačních systému na jediném zařízení. (WALLEN, 2017)

4.2.2 Instalace systému

Na oficiálních stránkách projektu Debian se nejdříve stáhne ISO obraz (jedná se o soubor, který obsahuje digitální kopii optického disku) s aktuální verzí systému Debian (aktuální verze je Debian 10.7). Ve VirtualBox se vytvoří nový virtuální počítač a nastaví se velikost RAM na 4 GB a přiřadí se virtuální pevný disk s velikostí 50 GB.

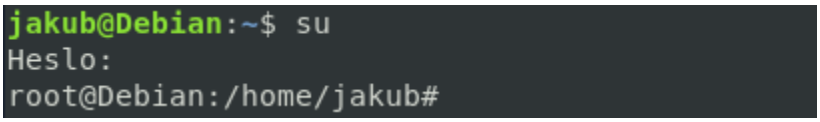
Během instalace se vytvořily dva účty. Root uživatel a obyčejný uživatel *jakub*. U obou účtu bylo vytvořeno heslo o 11 znacích, které neobsahuje žádné skutečné slovo, bylo vytvořeno pomocí malých a velkých písmen, číslic a speciálních znaků.

Poté, co se instalace dokončí, se server restartuje a zobrazí se nainstalovaný operační systém Debian a lze vidět uživatelský účet, který byl vytvořen během instalace. Po přihlášení se uživatelovi zobrazí plocha s grafickým rozhraním GNOME.

4.2.3 Přidání uživatele do sudoers

Při instalaci a konfiguraci bude zapotřebí, aby uživatelský účet, měl root práva, a tedy bude zapotřebí ho přidat mezi **sudoers uživatele.** Nejjednodušší volba pro přidání účtu mezi sudoers bude za pomocí jednoduchého příkazu.

Nejdříve se musí přepnout na root účet. Použijeme tedy příkaz: *su* Tento příkaz slouží k přepnutí mezi účty, ale jelikož se za příkaz nedoplnilo žádné jméno, tak systém předpokládá, že se chceme přihlásit jako root uživatel, a proto se vyplní přihlašovací heslo pro root. Přepnutí účtu proběhlo úspěšně a změnu lze vidět ve jméně účtu v terminálu.



Obrázek 3 - Přepnutí účtů (Zdroj: Vlastní)

Pro přidání použijeme příkaz: */sbin/usermod -aG sudo jakub***.** Funkce *usermod* přidá uživatele *jakub* do skupiny *sudo*. Pokud se přepneme zpátky na účet *jakub* a napíšeme příkaz: *sudo whoami*, tak dostaneme odpověď *root*. Kdyby účet neměl root práva, tak by systém vrátil varovnou hlášku.

```
root@Debian:/home/jakub# /sbin/usermod -aG sudo jakub
root@Debian:/home/jakub# su jakub
jakub@Debian:~$ sudo whoami
[sudo] heslo pro jakub:
root
jakub@Debian:~$
```

Obrázek 4 - Sudoer uživatel (Zdroj: Vlastní)

4.3 Konfigurace serveru

V následujících podkapitolách bude na základě oficiálního manuálu od Debianu zabezpečené jednotlivé části serveru. Použitý zabezpečovací manuál nese verzi 3.19.

4.3.1 Kontrola služeb a balíků

Nyní se zkontrolují spuštěné služby a nainstalované balíky v systému. Je to z důvodu, že ne všechny tyto funkcionality jsou potřeba pro chod serveru a mělo by platit pravidlo, že to, co na serveru není potřeba, tak by mělo být odinstalováno nebo zastaveno. Nejprve se v systému zkontrolují nainstalovaní démoni příkazem:

```
jakub@Debian:~$ systemctl | grep daemon
accounts-daemon.service
avahi-daemon.service
cron.service
fwupd.service
rtkit-daemon.service
avahi-daemon.socket
```

Obrázek 5 - Spuštění démoni (Zdroj: Vlastní)

Z uvedených služeb není zapotřebí **accounts-daemon.service** (umožňuje spravovat uživatelské informace) a **avahi-daemon.service** (umožňuje snadno vyhledávat připojená zařízení). Služby se vypnou příkazem: *systemctl disable <služba>*. (SCHRODER, 2016)

Dále by se měl vypnout **inetd** a jeho služby. **Inetd** dříve kompenzoval některé nedostatečné služby kernelu, ale v dnešní době není tolik zapotřebí.

```
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable telnet
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable echo
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable chargen
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable daytime
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable time
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable talk
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable ntalk
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable rsh
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable rlogin
jakub@Debian:~$ sudo /usr/sbin/update-inetd --disable rcp
```

Obrázek 6 - Inetd služby (Zdroj: Vlastní)

Dále je důležité zkontrolovat balíky. Riziko mohou představovat vývojové nástroje a interpretátor programovacích jazyků. Pokud uživatel plánuje programovat, tak jsou tyto balíky důležité v opačném případě mohou být zneužity útočníkem pro spuštění zákeřného softwaru.

Zde je seznam balíků, které mohou být bezpečně odstraněny ze serveru. Seznam může být trochu zastaralý, a proto je zapotřebí se podívat, jestli od daného balíku nevyšla nová verze.

Package	Size
gdb	2,766,822
gcc-3.3	1,570,284
dpkg-dev	166,800
libc6-dev	2,531,564
cpp-3.3	1,391,346
manpages-dev	1,081,408
flex	257,678
g++	1,384
linux-kernel-headers	1,377,022
bin86	82,090
cpp	29,446
gcc	4,896
g++-3.3	1,778,880
bison	702,830
make	366,138
libstdc++5-3.3-dev	774,982

Obrázek 7 - Seznam nepotřebných balíků (PEÑA, 2017)

V našem systému se nacházel pouze balík **cpp**. Balík ovšem odstraní i uživatelské rozhraní GNOME a ponechá server pouze jako terminál (příkazovou řádku). Přestože

se bude pracovat převážně v terminálu, tak chceme GNOME prostředí ponechat, a tedy balík neodinstalujeme. Mezi balíky, které by se měli odinstalovat, by měl patřit i **perl**, ale ten je využitý v mnoha jiných balících systému, a proto jeho odstranění je náročné (aby se nenarušili jiné programy). Proto **perl** bude také ponechán.

4.3.2 Po instalační opatření

Nyní je server nově nainstalovaný a očištěný od zbytečného softwaru. Teď přichází na řadu skutečná první konfigurace zabezpečení. Co by se mělo hned na začátku provést, je přihlášení k odběru bezpečnostních informačních emailů od vývojářů Debianu. Díky tomu lze rychle získávat upozornění, pokud se objeví nová zranitelnost systému.

Také je zapotřebí ze začátku spustit bezpečnostní aktualizace. Tyto aktualizace se provádí, protože čerstvě nainstalovaný systém nemusí obsahovat všechny dostupné bezpečnostní záplaty, a to hlavně v případě, že instalace probíhala bez připojení internetu (což je také doporučované pravidlo).

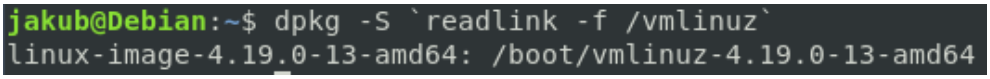
Než spustíme aktualizaci systému, tak do souboru **source.list** (soubor s místy odkud jsou stahovány aktualizace) musí být zapsán řádek
deb http://security.debian.org/ buster/updates main contrib non-free
Buster je „kódové označení“ pro verzi Debian 10.

```
# deb cdrom:[Debian GNU/Linux 10.7.0 _Buster_ - Official amd64 NETINST 20201205-11:16]/ buster main

#deb cdrom:[Debian GNU/Linux 10.7.0 _Buster_ - Official amd64 NETINST 20201205-11:16]/ buster main

deb http://deb.debian.org/debian/ buster main
deb-src http://deb.debian.org/debian/ buster main

deb http://security.debian.org/debian-security buster/updates main
deb-src http://security.debian.org/debian-security buster/updates main
deb http://security.debian.org/ buster/updates main contrib non-free

# buster-updates, previously known as 'volatile'
deb http://deb.debian.org/debian/ buster-updates main
deb-src http://deb.debian.org/debian/ buster-updates main

# This system was installed using small removable media
# (e.g. netinst, live or single CD). The matching "deb cdrom"
# entries were disabled at the end of the installation process.
# For information about how to configure apt package sources,
# see the sources.list(5) manual.
```

Obrázek 8 - source.list (Zdroj: Vlastní)

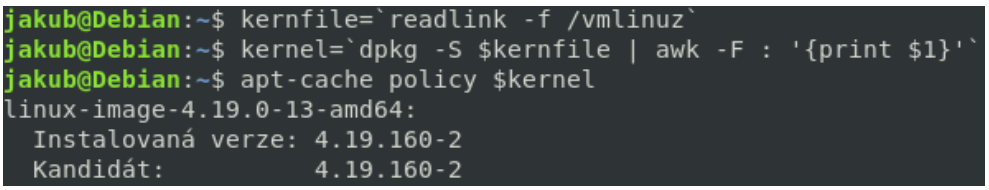
Spuštění aktualizace se spustí příkazy *sudo aptitude update* a následně *sudo aptitude upgrade*.

Též je zapotřebí zkontrolovat, jestli je aktualizovaný kernel. Nejdříve se musí zjistit, zda je kernel spravován pomocí balíkového systému (tedy pomocí příkazů *dpkg*). Pokud by tedy server vyhodil chybnou hlášku, tak víme, že kernel není spravován, jak má, ale v našem případě lze vidět, že je kernel nastaven správně.



Obrázek 9 - kernel verze (Zdroj: Vlastní)

Zkontrolovat se musí i jestli není zapotřebí aktualizovat samotný kernel. Na následujícím obrázku ovšem vidíme, že instalovaná verze je stejná jako verze kandidáta (tedy nejnovější verze kernelu).



Obrázek 10 - kernel verze 2 (Zdroj: Vlastní)

4.3.2.1 Reboot a root práva

Aby se zabránilo fyzickému útoku na server a přebootování nainstalovaného systému, tak se musí nastavit heslo pro bootloader, který je v našem případě GRUB (GRand Unified Bootloader – Unifikovaný Bootloader GRand). Nejprve se vytvoří heslo příkazem *grub-mkpasswd-pbkdf2*. Po zadání hesla se vytvoří jeho hash, který přidáme do souboru */etc/grub.d/00_header* spolu ještě s pár příkazy.

Obrázek 11 - Nastavení hesla pro GRUB (Zdroj: Vlastní)

Kernel také může při bootování a načítání **initramfs** vyhodit chybu, která by uživatele přivedla k příkazové řádce s root právy. Aby k tomu nedocházelo, tak se musí v souboru */etc/default/grub* nastavit GRUB_CMDLINE_LINUX na hodnotu *panic=0*. Všechny tyto změny se následně aplikují příkazem: *sudo update-grub*.

Aby uživatel restartoval či vypnul server, tak může použít vícero příkazů jako **reboot, shutdown, poweroff** a **halt**. Jedná z možností zabezpečení je u všech těchto funkcí odebrat možnost spouštět všemi uživateli. Však všechny tyto funkce jsou pouze symbolickými linky na **/bin/systemctl** jak je vidět na následujícím obrázku.

```
jakub@Debian:~$ ls -l /sbin/reboot
lrwxrwxrwx 1 root root 14 říj 24 20:44 /sbin/reboot -> /bin/systemctl
jakub@Debian:~$ ls -l /sbin/shutdown
lrwxrwxrwx 1 root root 14 říj 24 20:44 /sbin/shutdown -> /bin/systemctl
jakub@Debian:~$ ls -l /sbin/poweroff
lrwxrwxrwx 1 root root 14 říj 24 20:44 /sbin/poweroff -> /bin/systemctl
jakub@Debian:~$ ls -l /sbin/halt
lrwxrwxrwx 1 root root 14 říj 24 20:44 /sbin/halt -> /bin/systemctl
```

Obrázek 12 - Symbolické linky (Zdroj: Vlastní)

Tedy snadnější způsob je odebrat ostatním uživatelům možnost spouštět **systemctl**. To provedeme příkazem: *chmod 755 /bin/systemctl*, kde se práva na spouštění nastaví tak, že **systemctl** bude moc spouštět pouze uživatel s root právy.

4.3.2.2 Uživatelské účty a jejich přístupy

Při práci s uživatelskými účty velmi pomáhá PAM (Pluggable Authentication Modules – Zásuvné ověřovací moduly). Jedná se o moduly, které pomáhají administrátorovi serveru ověřovat uživatele skrz většinu aplikací (aplikace musí mít PAM podporu zabudovanou, což většina Debian aplikací má).

Do souboru */etc/pam.d/common-password* vložíme řádek:
password [success=1 default=ignore] pam_unix.so obscure minlen=8 sha512
Tento řádek nám v systému prosazuje, že nová hesla musí mít minimální délku 8 znaků (minlen=8), umožňuje kontrolovat složitost hesla (obscure) a že musí být zašifrována hash funkcí SHA-512. Bezpečnost hesla se dá i zlepšit instalací balíku **libpam-cracklib**. Ten se nainstaluje příkazem: *sudo apt install libpam-cracklib*.

Pomocí PAM lze i kontrolovat použití příkazu *su*. Příkaz slouží ke změně mezi uživatelskými účty (např. *su john*, přepne na účet *john*), ale také na přepnutí na root účet. Vytvoří se tedy nová skupina **wheel** (pozn.: název může být jakýkoliv), do které se přidají všichni uživatelé, kteří se mohou přepnout na root účet.

```
jakub@Debian:~$ sudo groupadd wheel
jakub@Debian:~$ sudo usermod -aG wheel jakub
```

Obrázek 13 - Vytvoření skupiny (Zdroj: Vlastní)

Poté do souboru */etc/pam.d/su* vložíme řádek:
auth requsite pam_wheel.so group=wheel debug
Tím zabráníme ostatním uživatelům, kteří nejsou ve skupině wheel, používat příkaz *su*.

Jelikož je možné využít zranitelností v dočasných souborech, tak tomu zabráníme přidáním řádku: *session optional pam_tmpdir.so* do souboru */etc/pam.d/common-sesion*.

Je dobré zkontrolovat konfiguraci přihlašování uživatelů, která se nachází v souboru */etc/login.defs*. Zde by se měli zkontrolovat následující řádky:

- *FAILLOG_ENAB yes* – Zaznamenává všechna špatná přihlášení.
- *LOG_UNKFAIL_ENAB no* – Zaznamenává špatná přihlašovací jména. Zde je lepší ponechat vypnuté, protože se uživatel může přepsat a omylem napsat svoje heslo jako přihlašovací jméno, což by byla chyba a jeho heslo by se viditelně uložilo.
- *ENCRYPT_METHOD SHA512* – Podporuje šifrování hash funkcí SHA512.

Soubor */etc/pam.d/login* necháme převážně nepozměněn, pouze zde upravíme řádek: *auth optional pam_faildelay.so delay=10000000*
Tento řádek znamená, že když se uživatel špatně přihlásí, tak musí čekat 10 vteřin (v základním nastavení pouze 3) nežli bude moct se znova pokusit přihlásit. Díky tomuto lze zastavit (zpomalit) útoky hrubou silou.

Do konfiguračního souboru */etc/security/access.conf* přidáme řádek:
-:wheel:ALL EXCEPT LOCAL, pomocí něhož se zabrání vzdálenému přístupu do root účtu.

Nyní se nainstaluje balík **auditd**, který bude sloužit pro zaznamenávání a audit změn v systému. Po jeho instalaci se zkontroluje, zda běží jako služba příkazem: *service auditd status*. Pokud ano, tak lze začít označovat soubory, které mají být sledovány. Přidání souboru či složky mezi sledované se provede příkazem: *sudo auditctl -w <adresa k souboru> -p wrxa*. Tímto se bude zaznamenávat veškerá aktivita, která byla na soubor provedena (*p* v příkazu vyznačuje *oprávnění* a v tomto případě sledujeme, když je soubor přečten **w**, zapsáno do něj **r**, spuštěn **x** nebo k němu něco přidáno **a**).

Jelikož je zapotřebí i zachovat soukromí uživatelů, tak by se měla nastavit přístupová práva i do jejich domovských adresářů. V základním nastavení mají domovské adresáře práva 0755, tedy všichni uživatelé si mohou nahlédnout do svých domovských adresářů navzájem. Nastavení se změní v souboru */etc/adduser.conf,* kde se přepíše proměnná *DIR_MODE* na hodnotu 0750.

V poslední řadě bude na server nainstalován balík **autolog**, který bude pomáhat s neaktivníma uživateli. Po instalaci se otevře soubor */etc/autolog.conf*, ve kterém jsou uložena všechna pravidla. Nyní stačí pouze přidat název skupiny (nebo i název uživatele, protože při vytvoření uživatelského účtu je vytvořená i stejnojmenná skupina, např. uživatel *jakub* je také ve skupině *jakub*) a jak dlouho musí být nečinný,

aby ho systém odhlásil. Nejjednodušší řešení je však vytvoření skupiny **user**, do které přidáme každého nového uživatele. Pravidla vypadají následovně:

```
name=ppp-.*     idle=-1 line=ttyS2
line=pty.*      idle=30 grace=30 nomail nolog
group=games     idle=10 grace=60
group=lynx.*    idle=10 grace=60 clear

# protected users
name=root       idle=-1

# idle - limits
group=student   idle=15 grace=180
group=john      idle=5  grace=120
# session - limits
group=users     idle=10 grace=120      hard ban=1 clear
name=guest      idle=10 grace=120      hard ban=1 clear
```

Obrázek 14 - autolog.conf (Zdroj: Vlastní)

Group udává název skupiny, které se pravidlo týká, *idle* udává, jak dlouho může být uživatel neaktivní (v minutách), *grace* udává, kdy před odhlášením vyskočí varování (ve vteřinách), *ban* udává, jak dlouho bude muset uživatel čekat, než se bude moci znovu přihlásit (v minutách).

4.3.2.3 Logy

Dříve než bude spuštěna konfigurace logů, tak je dobrá výpomoc, vložit následující kód do souboru */etc/hosts.deny*.

```
ALL: ALL: SPAWN ( \
  echo -e "\n\
  TCP Wrappers\: Connection refused\n\
  By\: $(uname -n)\n\
  Process\: %d (pid %p)\n\
  User\: %u\n\
  Host\: %c\n\
  Date\: $(date)\n\
 " | /usr/bin/mail -s "Connection to %d blocked" root) &
```

Obrázek 15 - hosts.deny (Zdroj: Vlastní)

Pokud se někdo bude neúspěšně pokoušet připojit na server, tak admin dostane emailovou zprávu. Dále se na server nainstaluje balík **logcheck**. Balík už se nemusí tolik konfigurovat (ovšem záleží na bezpečnostní politice), pouze jediná změna, která bude provedena je v souboru */etc/logcheck/logcheck.conf*, kde se nastaví hodnota proměnné *SENDMAILTO=“jakub“*. Hodnota nastavuje, která osoba by měla dostávat emaily ohledně logů v systému. V souboru se také nachází proměnná *REPORTLEVEL*, která je v základu nastavená na hodnotu *server*. Pokud bychom spravovali malý server s malým počtem běžících aplikací, tak lze tuto hodnotu nastavit na *paranoid*, tím bude systém sbírat a ukládat více informací o stavu serveru (v opačném případě lze nastavit hodnotu na *workstation*).

Dále se o logy v systému stará služba **rsyslog.** Konfigurační soubor se nachází na */etc/rsyslog.conf*. V dokumentu se nachází funkce, které jsou sledovány a kam jsou tyto záznamy ukládány. Zde od komentujeme některá nastavení, aby soubor vypadal jako na následujícím obrázku.

```
#
# First some standard log files.  Log by facility.
#
auth,authpriv.*                 /var/log/auth.log
*.*;auth,authpriv.none          -/var/log/syslog
cron.*                          /var/log/cron.log
daemon.*                        -/var/log/daemon.log
kern.*                          -/var/log/kern.log
lpr.*                           -/var/log/lpr.log
mail.*                          -/var/log/mail.log
user.*                          -/var/log/user.log
```

Obrázek 16 - rsyslog.conf (Zdroj: Vlastní)

Po aktualizaci souboru je dobré službu restartovat příkazem: *sudo service rsyslog restart*. Nakonec chceme, aby ne všechny soubory byly čitelné ostatními uživateli. Proto použijeme příkaz: *chmod 660 <soubor>* a tím ostatním uživatelům zabráníme v čtení těchto souborů.

4.3.3 Zabezpečení služeb

Na ochranu služeb systému lze použít jeden ze dvou základních způsobů. Jeden způsob je nastavit přístupy ke službám pouze oprávněným uživatelům. Druhý způsob

je zamezení služby, aby byla dostupná pouze na vybraných přístupových bodech, na kterých je služba potřebná. (PEÑA, 2017)

4.3.3.1 SSH

Jelikož se v systému stále nachází **telnet**, který je v podstatě nezabezpečený **SSH**, tak bude lepší ho odinstalovat. Použije se příkaz: *sudo apt remove telnet*. Následně použijeme příkaz *sudo apt install ssh*, který nainstaluje SSH. Bude zapotřebí upravit některé hodnoty v konfiguračním souboru */etc/ssh/sshd_config*.

- *ListenAddress 10.0.2.15* - jedná se o rozhraní (síťová karta), na kterém chceme, aby SSH naslouchalo.
- *Port 666* – změnou portu nebude potenciální útočník schopen říct, zda na serveru běží SSH démon.
- *PermitRootLogin no* – Znemožní přihlášení jako root skrz SSH.
- *AllowGroups wheel* – Pouze uživatelé ve skupině *wheel* budou schnopni používat SSH.
- *Protocol 2* – Vynucuje použití *protokolu 2* jelikož *protokol 1* obsahuje zranitelnosti ohledně snazšího prolomení hesla.
- *PasswordAuthentication no* – Nepovoluje přihlášení za pomocí hesla. Musí být pro připojení využit SSH klíč.

Jak již bylo zmíněno, tak autentizace u SSH by se měla provádět na základě vygenerovaných soukromých a veřejných klíčů. Generování klíčů se bude provádět na uživatelském počítači, kterým se uživatel bude přihlašovat na server. Klíče se vygenerují příkazem: *ssh-keygen*. Pokud na počítači již byly existující klíče, tak je možné jej přepsat. Klíče se uloží do složky */home/<uživatelské_jméno>/.ssh*. Během generování lze nastavit tzv. přístupovou frázi, která slouží podobně jako heslo. Nahrání veřejného klíče na server se provede příkazem: *ssh-copy-id jakub@10.0.2.15 -p 666*. V terminálu vyskočí zpráva o úspěšném přidání klíče.

Dalšími důležitými soubory jsou */etc/hosts.allow* a */etc/hosts.deny*. Do těchto souborů se mohou zadat IP adresy, které se mohou připojit na server. Přidání IP

adres pro SSH v těchto souborech se provede přidáním řádku do konfiguračního souboru:
sshd: X.X.X.X nebo *sshd: ALL*.

4.3.3.2 FTP

Na místo původního **FTP** bude na server požitá jeho bezpečnější verze **VSFTPD**. VSFTPD není součástí původního systému a nainstaluje se příkazem: *sudo apt install vsftpd*. Pro účely FTP se vytvoří jedinečný účet, který se pojmenuje **ftpuser**. Vytvoříme FTP složku příkazem: *sudo mkdir /home/ftpuser/ftp*, které se nastaví „prázdný“ vlastník: *sudo chown nobody:nogroup /home/ftpuser/ftp*. Ze složky se odebere právo zápisu: *sudo chmod a-w /home/ftpuser/ftp*. Poté se v této složce vytvoří ještě jedna složka, která bude sloužit pro nahrávání souborů: *sudo mkdir /home/ftpuser/ftp/soubory*. Následně se přiřadí vlastník jako **ftpuser**: *sudo chown ftpuser:ftpuser /home/ftpuser/ftp/soubory*. V uvedené složce se vytvoří textový soubor *Hello.txt*, který bude sloužit při testování připojení FTP.

Po přidání FTP účtu se otevře soubor */etc/vsftpd.conf*, kde se nastaví následující hodnoty:

- *anonymous_enable=NO* – zakazuje anonymní přihlášení přes FTP.
- *local_enable=YES* – umožňuje přihlášení lokálním uživatelům.
- *write_enable=YES* – umožňuje lokálním uživatelům zapisovat data do systému.
- *chroot_local_user=YES* – bude uzavírat uživatele do chroot vězení.
- *user_sub_token=$USER* – uživatelský token pro vložení uživatelského jména
- *local_root=/etc/$USER/ftp* – nastavení cesty pro uživatele
- *userlist_enable=YES* – aktivuje službu *userlist*
- *userlist_file=/etc/vsftpd.userlist* – ukazuje na soubor *userlist*
- *userlist_deny=NO* – pokud je hodnota nastavena na *NO*, tak bude ftp povolovat uživatele, kteří jsou uložení v souboru *vsftpd.userlist*.

Nakonec konfiguračního souboru se připíšou ještě následující parametry, které by měly zajišťovat bezpečné šifrování komunikace.

```
#rsa_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
#rsa_private_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
rsa_cert_file=/etc/ssl/private/vsftpd.pem
rsa_private_key_file=/etc/ssl/private/vsftpd.pem
ssl_enable=YES
allow_anon_ssl=NO
force_local_data_ssl=YES
force_local_logins_ssl=YES
ssl_tlsv1=YES
ssl_sslv2=NO
ssl_sslv3=NO
require_ssl_reuse=NO
ssl_ciphers=HIGH
```

Obrázek 17 - vsftpd.conf (Zdroj: Vlastní)

Na předchozím obrázku jsou dva řádky za komentované a to z toho důvodu, že se bude vytvářet nový SSL certifikát. Certifikát s platností 1 rok se vytvoří příkazem: *sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/vsftpd.pem -out /etc/ssl/private/vsftpd.pem*. Následně se systém doptá na doplňující informace ohledně.

```
Generating a RSA private key
.....................................................................+++++
..............+++++
writing new private key to '/etc/ssl/private/vsftpd.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:CZ
State or Province Name (full name) [Some-State]:Czech republic
Locality Name (eg, city) []:Prague
Organization Name (eg, company) [Internet Widgits Pty Ltd]:CZU
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:10.0.2.15
Email Address []:
```

Obrázek 18 - Doplňující informace o certifikátu (Zdroj: Vlastní)

Po uložení konfiguračního souboru se vytvoří soubor */etc/vsftpd.userlist*, do kterého se vloží jeden řádek: *ftpuser*. Nakonec se služba restartuje příkazem: *sudo service vsftp restart*.

4.3.3.3 Apache

Než se začne s konfigurací **Apache** (služba nese v systému název **apache2**), tak je dobré zkontrolovat je nainstalovaná nejnovější verze. Z následujícího obrázku je zřejmé, že je Apache na nejaktuálnější verzi.

```
jakub@Debian:~$ sudo apt install apache2
Načítají se seznamy balíků… Hotovo
Vytváří se strom závislostí
Načítají se stavové informace… Hotovo
apache2 je již nejnovější verze (2.4.38-3+deb10u4).
Následující balík byl nainstalován automaticky a již není potřeba:
  analog
Pro jeho odstranění použijte „sudo apt autoremove“.
0 aktualizováno, 0 nově instalováno, 0 k odstranění a 2 neaktualizováno.
```

Obrázek 19 - verze apache2 (Zdroj: Vlastní)

Nastavení služby se provádí v souboru */etc/apache2/apache2.conf* (v jiných systémech se tento soubor jmenuje *httpd.conf*). Do tohoto souboru se přidají následující hodnoty:

- *ServerSignature Off* – odebere verzi systému z vygenerovaných stránek od Apache (jedná se například o neexistující stránky s HTTP kódem 404).
- *ServerTokens Prod* – změní hlavičku stránek na produkci. Tedy výše uvedené vygenerované stránky budou pouze obsahovat název *Apache*.
- *FileETag None* – odebere Etag hlavičku, která by útočníkům mohla prozradit informace jako např. čísla i-uzlu či běžící podřízené procesy
- *TraceEnable Off* – odebere možnost TRACE požadavků na webu. Tyto požadavky mohou útočníkovi dovolit provést Cross Site Tracing útok a ukrást tak informace o cookie.

Dále se v souboru upraví tag <*Directory /var/www*>, tak aby měl následující podobu:

```
<Directory /var/www/>
        Options None
        AllowOverride None
        Require all granted
</Directory>
```

Obrázek 20 - Directory tag (Zdroj: Vlastní)

Tento tag zabraňuje, aby server ukazoval adresáře a jejich obsah. Vytvoří se tedy složka */var/www/html/soubory*, do které se vloží soubor *test.txt*. Tato složka a soubor se vytváří pouze pro otestování konfigurace.

V souboru by se také měl nastavit uživatelský účet s nízkými oprávněními. Tedy nejdříve se vytvoří skupina příkazem: *groupadd apache-user*. Poté se použije příkaz pro přidání uživatele s následujícími atributy: *useradd -p <heslo> -g apache-user -d /var/www apache-user*. Tento příkaz vytvoří uživatele *apache-user*, který bude přidán do již vytvořené skupiny *apache-user* a jeho domovský adresář bude nastaven na */var/www* neboli adresář Apache serveru. Opět se otevře soubor *apache2.conf*, kde se nastaví uživatel a skupina na *apache-user*.

Poslední změnou bude nahrazení defaultní stránky, která je na serveru hostována. Defaultní webová stránka může být ponechána, ale je zvykem defaultní stránku nezobrazovat. Stránka může být nahrazena jakýmikoliv jinými stránkami, v případě této diplomové práce bude stránka nahrazena pouze obyčejným nadpisem „Hello World“. Defaultní webová stránka je umístěna */var/www/html/index.html*.

Po provedení všech uvedených změn se služba restartuje příkazem: *sudo service apache2 restart*.

4.3.3.4 Firewall

Jako firewall bude na server využita služba **nftables**, která však není součástí základní instalace systému a je potřeba doinstalovat. Po instalaci je potřeba službu zapnout a popřípadě zkontrolovat, zda **nftables** běží.

```
jakub@Debian:~$ sudo service nftables start
jakub@Debian:~$ sudo service nftables status
● nftables.service - nftables
   Loaded: loaded (/lib/systemd/system/nftables.service; disabled; vendor preset: enabled)
   Active: active (exited) since Tue 2021-01-26 19:28:18 CET; 2s ago
     Docs: man:nft(8)
           http://wiki.nftables.org
  Process: 3806 ExecStart=/usr/sbin/nft -f /etc/nftables.conf (code=exited, status=0/SUCCESS)
 Main PID: 3806 (code=exited, status=0/SUCCESS)

led 26 19:28:18 Debian systemd[1]: Starting nftables...
led 26 19:28:18 Debian systemd[1]: Started nftables.
```

Obrázek 21 - Kontrola nftables (Zdroj: Vlastní)

Nastavení se bude provádět v hlavním konfiguračním souboru */etc/nftables.conf*. V původní podobě by měl konfigurační soubor obsahovat základní nastavení:
type filter hook <typ pravidla> priority 0;
Může se ale stát, že soubor takové nastavení nemá a bude ho potřeba doplnit. Tedy v základu by měl soubor vypadat jako na následujícím obrázku.

```
#!/usr/sbin/nft -f

flush ruleset

table inet filter {
        chain input {
                type filter hook input priority 0;
        }
        chain forward {
                type filter hook forward priority 0;
        }
        chain output {
                type filter hook output priority 0;
        }
}
```

Obrázek 22 - nftables.conf (Zdroj: Vlastní)

Nejdříve se nastaví řetěz *input*, do kterého se přidají následující pravidla:

- *iifname lo accept*. – Akceptuje spojení z *loopback* adres (jedná se o rozmezí adres 127.0.0.0 – 127.255.255.255, tedy lokální adresy systému).
- *ct state established,related accept* – Akceptuje spojení, která pochází z našeho systému.
- *ct state invalid drop* – Zahození neplatných příchozích paketů.
- *ip protocol icmp limit rate 2/second accept* – Akceptuje ICMP (Internet Control Message Protocol – Internetový Kontrolní Protokol Zpráv) pakety na IPv4. Obsahuje limit, aby se zabránilo pokusům o útok zaplavením.
- *ip6 nexthdr icmpv6 limit rate 2/second accept* – Akceptuje ICMP pakety na IPv6. Obsahuje limit, aby se zabránilo pokusům o útok zaplavení.
- *tcp dport {21,666} accept* – Akceptuje použití FTP a SSH na portech 21 a 666 (viz kapitola *4.3.3.1 SSH*).
- *tcp dport {http,https} accept* – Akceptuje veškeré HTTP a HTTPS požadavky.
- *policy drop* – Zahození všeho co nespadá pod uvedená pravidla *input* řetězu.

Řetěz *forward* se moc upravovat nebude. Pouze se zde přidá *policy drop*.

Do řetězu *output* se přidají následující pravidla:

- *tcp dport ssh reject with icmp type host-unreachable* – Zamezuje server se pomocí SSH připojit na jiné zařízení.
- *policy accept* – Akceptuje jakákoliv jiná výstupová data.

Výsledný konfigurace **nftables** by měla vypadat jako na následujícím obrázku. Dále se soubor uloží a služba se restartuje, aby se pravidla uplatnila.

```
#!/usr/sbin/nft -f

flush ruleset

table inet filter {
        chain input {
                type filter hook input priority 0;

                # Povolit loopback adresy
                iifname lo accept;

                # Povolit pripojeni pochazejici z naseho systemu
                ct state established,related accept;

                # Zahodit neplatne pakety
                ct state invalid drop;

                # Povolit ICMP - ochrana pred utokem zaplaveni
                ip6 nexthdr icmpv6 limit rate 2/second accept;
                ip protocol icmp limit rate 2/second accept;

                # Povolit FTP na portu 21, SSH na portu 666
                tcp dport {21,666} accept;

                # Povolit veskere HTTP, HTTPS pozadavky
                tcp dport {http,https} accept;

                # Zahodit cokoliv jineho
                policy drop;
        }
        chain forward {
                type filter hook forward priority 0;

                # Zahodit cokoliv jineho
                policy drop;
        }
        chain output {
                type filter hook output priority 0;

                # Zamezeni serveru se pripojit pomoci SSH na jine zarizeni
                tcp dport ssh reject with icmp type host-unreachable;

                # Akceptovat jakokoliv jina vystupova data
                policy accept;
        }
}
```

Obrázek 23 - Upraveny nftables.conf (Zdroj: Vlastní)

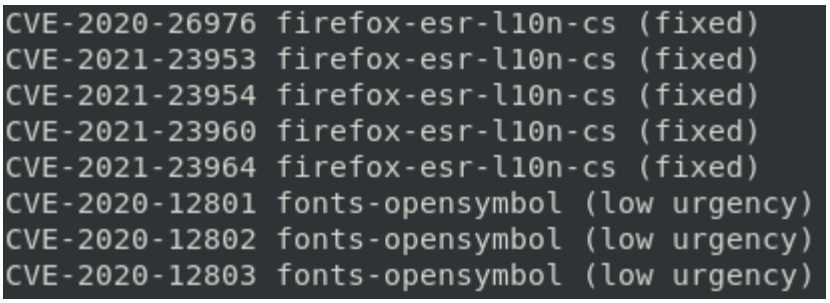
4.3.4 Doplňující nástroje

V následujících podkapitolách budou do systému nainstalované doplňující nástroje, které budou pomáhat v zabezpečování serveru.

4.3.4.1 Debsecan

Debsecan je program, který analyzuje veškeré nainstalované balíky v systému a následně vyhodnocuje potenciální zranitelnosti. Některé tyto zranitelnosti jsou i označené, tak aby uživatel věděl, jestli se jedná **opravenou**, **nezávažnou** anebo **závažnou** chybu. Zranitelnosti se posuzují na základě oficiální bezpečností databáze Debianu.

Neprovádí se téměř žádná konfigurace. Nejdříve se program stáhne příkazem: *sudo apt install debsecan* a poté se pouze použije příkaz: *debsecan –suite=<kódové označení systémové verze>* (v tomto případě *buster*). Poté debsecan vypíše seznam CVE, který by se měl projít a posléze aktualizovat, opravit či nahradit rizikové balíky. V případě našeho systému neobsahoval výsledek analýzy žádné velké zranitelnosti.



Obrázek 24 - ukázka výstupu debsecan (Zdroj: Vlastní)

4.3.4.2 Rkhunter

Pro odhalování rootkitů v systému bude použit nástroj **rkhunter**. Nástroj se nejdříve nainstaluje příkazem: *sudo apt install rkhunter*. První věcí, která by se měla provést po instalaci (a měla by se provádět i na pravidelné bázi) je aktualizace databáze vlastností, díky kterým rkhunter hledá škodlivé programy. Aktualizace se provede příkazem: *sudo rkhunter –propupd*.

Než bude spuštěn první sken, tak se otevře konfigurační soubor */etc/rkhunter.conf*, do kterého se přidají následující řádky.

```
SCRIPTWHITELIST=/usr/bin/egrep
SCRIPTWHITELIST=/usr/bin/fgrep
SCRIPTWHITELIST=/usr/bin/which
SCRIPTWHITELIST=/usr/bin/lwp-request
```

Obrázek 25 - rkhunter.conf (Zdroj: Vlastní)

Pokud by konfigurační soubor neupravil, tak by nástroj vrátil varovná upozornění na tyto uvedené skripty. Je to z důvodu, že některé původní služby byly právě nahrazeny těmito skripty. Po uložení souboru se spustí sken systému příkazem: *sudo rkhunter --check --sk*. Po dokončení skenu získáme seznam všech systémových příkazů a jestli jsou v pořádku a také získáme seznam známých rootkitů a jestli se v systému vyskytují. V našem případě se v systému žádný rootkit nevyskytuje. Veškeré výsledky skenu lze později zobrazit v souboru */var/log/rkhunter.log*.

```
Checking for rootkits...

  Performing check of known rootkit files and directories
    55808 Trojan - Variant A                                 [ Not found ]
    ADM Worm                                                 [ Not found ]
    AjaKit Rootkit                                           [ Not found ]
    Adore Rootkit                                            [ Not found ]
    aPa Kit                                                  [ Not found ]
    Apache Worm                                              [ Not found ]
```

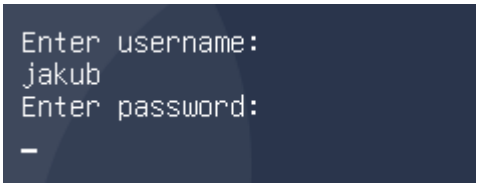
Obrázek 26 - Ukázka výsledku rkhunter (Zdroj: Vlastní)

4.4 Testování zabezpečení

V následujících podkapitolách bude provedeno testování a kontrola zavedených bezpečnostních opatření na serveru. Pro otestování některých konfigurací bude použit další virtuální počítač, kterým se budeme snažit přihlásit pomocí SSH na server. Bude se jednat o Linux Kali 2020.3, který je postaven na systému Debian.

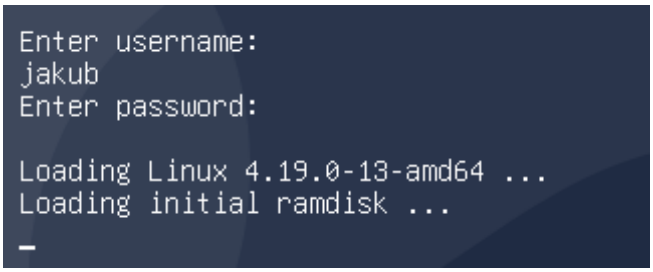
4.4.1 Bootloader

Po instalaci systému bylo nastaveno heslo pro GRUB spolu s uživatelským účtem. Při startu serveru se tedy systém automaticky nenačte, ale nejdřív musíme zadat uživatelské jméno spolu s jedinečným heslem (nejedná se o heslo k účtu).



Obrázek 27 - Přihlášeni GRUB (Zdroj: Vlastní)

Po zadání správného přihlašovacího jména a hesla lze vidět, že přihlášení bylo úspěšné a systém se začne načítat. Po dokončení načítání se budeme moci přihlásit do systému.



Obrázek 28 - Úspěšné přihlášení GRUB (Zdroj: Vlastní)

4.4.2 Uživatelské účty

Při vytváření nového účtu by měl být aplikován důraz na složitost hesla. V našem systému je nastavena politika hesla na délku minimálně 8 znaků a heslo by nemělo být příliš jednoduché. Na následujícím obrázku vidíme, že v případě, kdy zadáme heslo pouze jako **123**, tak nám vyskočí upozornění, že heslo je příliš krátké a jednoduché. V dalším případě zkusíme zadat heslo **12345678**, tedy splníme podmínku počtu znaků, ale heslo je stále jednoduché. Zde je tedy vidět, že upozornění na délku se již nezobrazují, ale ukazuje se, že heslo je stále jednoduché. V posledním pokusu se zadá již složité heslo a systém tedy neukazuje žádná upozornění.

```
jakub@Debian:~$ sudo adduser tester
Přidávám uživatele „tester“…
Přidávám novou skupinu „tester“ (1004)…
Přidávám nového uživatele „tester“ (1003) se skupinou „tester“…
Domovský adresář „/home/tester“ již existuje. Nekopíruji z „/etc/skel“.
Nové heslo:
ŠPATNÉ HESLO: je PŘÍLIŠ krátké
ŠPATNÉ HESLO: je příliš jednoduché
Opakujte nové heslo:
Hesla se neshodují.
Nové heslo:
ŠPATNÉ HESLO: je příliš jednoduché nebo systematické
Opakujte nové heslo:
Hesla se neshodují.
Nové heslo:
Opakujte nové heslo:
passwd: heslo bylo úspěšně změněno
Měním informace o uživateli tester
```

Obrázek 29 - Tvorba hesla (Zdroj: Vlastní)

Tato upozornění jsou pouze varovná. Tedy pokud uživatel bude ignorovat upozornění, tak může založit účet se slabým heslem.

Kromě vybraných uživatelů, by ostatní uživatelé neměli být schopni používat příkazy *sudo* a *su*. Na následujícím obrázku je vidět, že nově vytvořená účet **tester** není zapsán v souboru **sudoers** a ani nemůže použít příkaz pro změnu účtu.

```
tester@Debian:~$ sudo cat /etc/shadow

Věříme, že jste od správce místního systému obdrželi obvyklé školení.
Obvykle se jedná o tyto tři věci:

    1. Respektujte soukromí druhých.
    2. Přemýšlejte, než začnete psát.
    3. S velkými právy přichází velká zodpovědnost.

[sudo] heslo pro tester:
tester není v souboru sudoers. Tento událost bude ohlášena.
tester@Debian:~$ su
su: Přístup zamítnut
```

Obrázek 30 - Pravomoc uživatelů (Zdroj: Vlastní)

Dále systém úspěšně zaznamenává všechna špatná přihlášení, které si lze zobrazit v souboru */var/log/auth*.log. V poslední řade bylo otestováno automatické odhlášení. Nejdříve ponecháme učet „nehybně“ stát a systém nás po 5 minutách sám odhlásí.

4.4.3 SSH

Nejdříve se do souboru */etc/hosts.allow* přidá IP adresa počítače, které se bude připojovat na server. Příkaz pro připojení pomocí ssh bude: *ssh 10.0.2.15 -l jakub*. Příkaz obsahuje IP adresu serveru a login **jakub**, na který se chceme připojit. Bez uvedení portu se bude zařízení připojovat na port 22. Na serveru je nastaven port pro SSH na 666 a tedy se počítač na server nepřipojí.

```
kali@kali:~$ ssh 10.0.2.15 -l jakub
ssh: connect to host 10.0.2.15 port 22: Connection refused
```

Obrázek 31 - Nesprávný SSH port (Zdroj: Vlastní)

Pokud do příkazu přidáme parametr *-p 666*, tak se budeme moc na server přihlásit. Pro autentizaci za pomocí klíčů se nezadává heslo, ale přístupová fráze, která se vytvářela při generování klíčů.

```
kali@kali:~$ ssh 10.0.2.15 -p 666 -l jakub
Enter passphrase for key '/home/kali/.ssh/id_rsa':
Linux Debian 4.19.0-13-amd64 #1 SMP Debian 4.19.160-2 (2020-11-28) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have new mail.
Last login: Sun Mar 21 14:47:11 2021 from 10.0.2.9
jakub@Debian:~$
```

Obrázek 32 - SSH připojení (Zdroj: Vlastní)

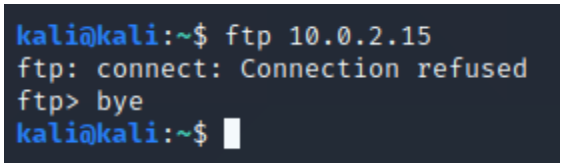
Na serveru je nastaveno, že pouze uživatelé ve skupině **wheel** se mohou přihlásit za pomocí SSH a že přihlášení na **root** je zakázané. Pokud se tedy počítač zkusí přihlásit jako **root** nebo jako obyčejný uživatel, tak server připojení zamítne.

```
kali@kali:~$ ssh 10.0.2.15 -p 666 -l tester
tester@10.0.2.15: Permission denied (publickey).
kali@kali:~$ ssh 10.0.2.15 -p 666 -l root
root@10.0.2.15: Permission denied (publickey).
```

Obrázek 33 - Neoprávněné SSH přihlášení (Zdroj: Vlastní)

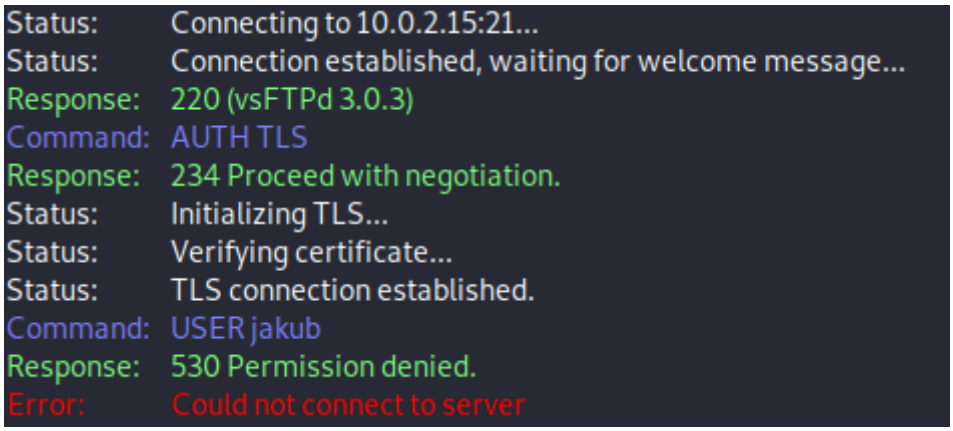
4.4.4 FTP

Aktuální konfigurace serveru nám brání používat obyčejného FTP klienta v terminálu počítače. Pokud se počítač pokus připojit přes terminál, tak server připojení zamítne.



Obrázek 34 - Zamítnutí FTP připojení (Zdroj: Vlastní)

Pro otestování bude využit FTP klient **FileZilla**. Pokud se počítač pokusí na server přihlásit jako jiný uživatel než je před-vytvořený **ftpuser**, tak opět server zamítne připojení.

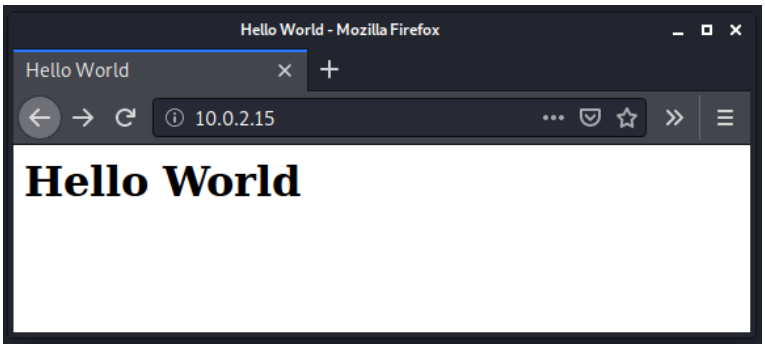


Obrázek 35 - Zamítnutí FTP připojení 2 (Zdroj: Vlastní)

Pouze v případě, že se počítač přihlašuje jako **ftpuser**, tak server povolí připojení. Nežli server pustí uživatele na server, tak je ukázán certifikát, který byl na serveru vytvořen. Po úspěšném přihlášení lze vidět výpis složky, která byla výše nastavena a že se zde nachází testovací soubor *Hello.txt*.

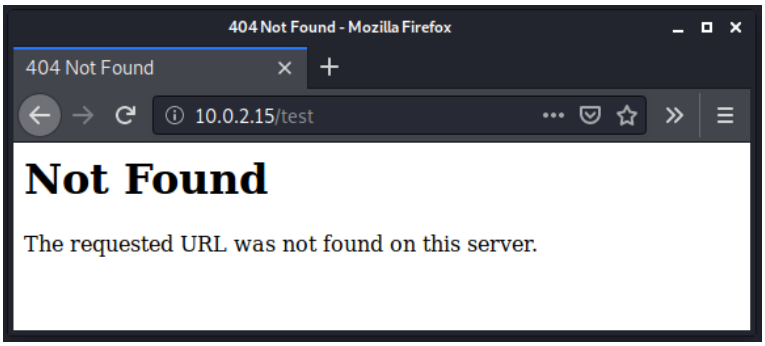
4.4.5 Apache

Pokud se v počítači otevře prohlížeč a zadá se zde IP adresa serveru, tak se správně objeví prázdná stránka, která pouze obsahuje nadpis „Hello World“.



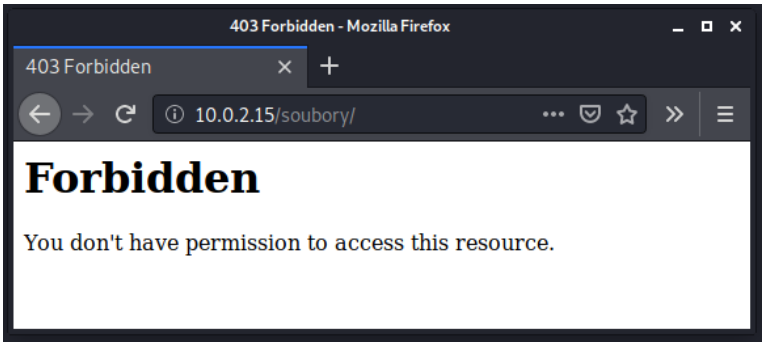
Obrázek 36 - Webová stránka (Zdroj: Vlastní)

Při nedostatečném opatření se může stát, že server na neexistující stránky (např. stránky s chybovou hláškou 404) ukazuje informace o serveru jako je název služby či její verze. V případě, že zadáme nějakou neexistující adresu, např. *10.0.2.15/test*, tak na následujícím obrázku lze vidět, že server o sobě neukazuje žádné informace.



Obrázek 37 - Webová stránka 2 (Zdroj: Vlastní)

Na serveru však existuje soubor *test.txt*, který se nachází na adrese *10.0.2.15/soubory*. Výpis složky, a tedy i zobrazení souboru by nemělo fungovat. Na následujícím obrázku je vidět, že adresa je správná, ale prohlížeč k ní nemá udělen přístup.



Obrázek 38 - Webová stránka 3 (Zdroj: Vlastní)

4.4.6 Sken serveru

Jak již bylo uvedeno výše, tak na serveru byly spuštěny 2 skenovací programy, **Debsecan** a **Rkhunter**. Ty mají za úkol oskenovat systém a najít na něm zákeřný software nebo systémovou zranitelnost. V obou případech byly výsledky skenů úspěšně dokončené a nebyly v systému nalezené žádné zranitelnosti.

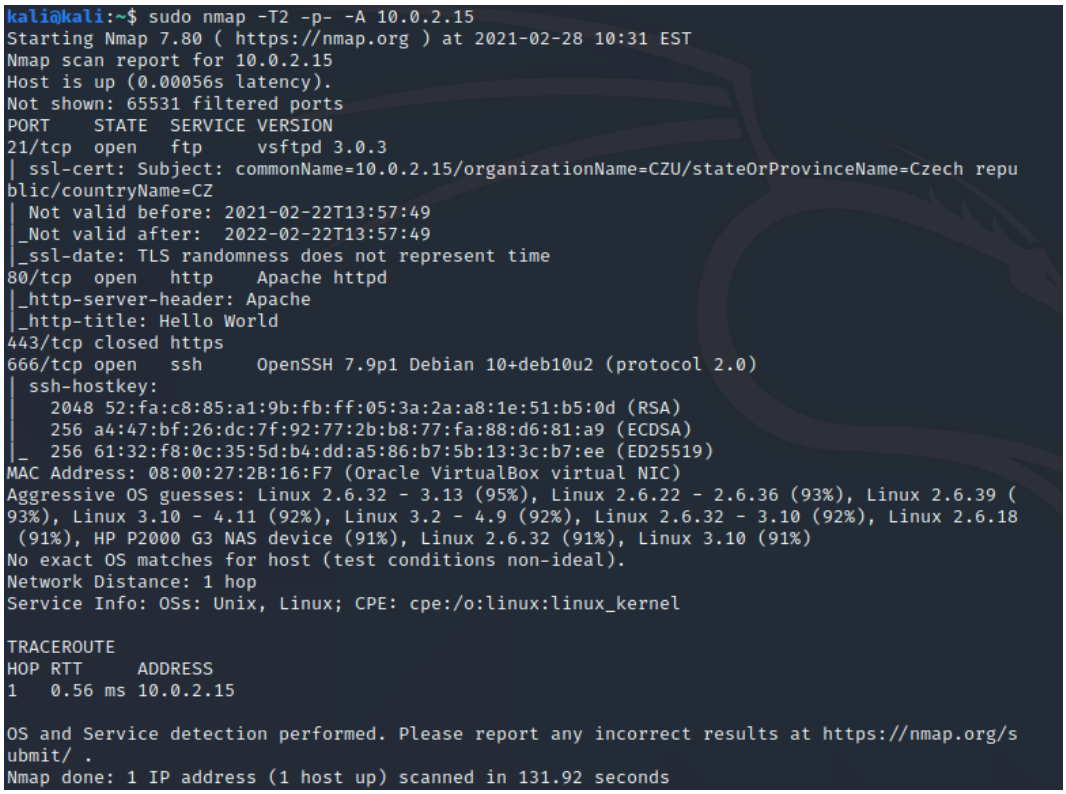
Další sken se provede za použití nástroje **nmap**. Jedná se o bezpečnostní skener, který se snaží zjistit operační systém a otevřené porty na serveru. Pokud **nmap** nalezne otevřený port, tak se dále pokusí zjistit jaká služba na daném portu běží a posléze zjistit i verzi dané služby. Sken se provedl z Kali počítače a spustil se příkazem: *sudo nmap -T4 -p- -A 10.0.2.15*.

```
kali@kali:~$ sudo nmap -T4 -p- -A 10.0.2.15
Starting Nmap 7.80 ( https://nmap.org ) at 2021-02-28 09:31 EST
Nmap scan report for 10.0.2.15
Host is up (0.00066s latency).
Not shown: 65533 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
| ssl-cert: Subject: commonName=10.0.2.15/organizationName=CZU/stateOrProvinceName=Czech republic/countryName=CZ
| Not valid before: 2021-02-22T13:57:49
|_Not valid after:  2022-02-22T13:57:49
|_ssl-date: TLS randomness does not represent time
80/tcp open  http    Apache httpd
|_http-server-header: Apache
|_http-title: Hello World
MAC Address: 08:00:27:2B:16:F7 (Oracle VirtualBox virtual NIC)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=2/28%OT=21%CT=1%CU=30370%PV=Y%DS=1%DC=D%G=Y%M=080027%T
OS:M=603BA977%P=x86_64-pc-linux-gnu)SEQ(SP=101%GCD=1%ISR=10A%TI=Z%CI=Z%II=I
OS:%TS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O
OS:5=M5B4ST11NW7%O6=M5B4ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6
OS:=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%O=M5B4NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O
OS:%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=
OS:0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%
OS:S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(
OS:R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=
OS:N%T=40%CD=S)

Network Distance: 1 hop
Service Info: OS: Unix

TRACEROUTE
HOP RTT     ADDRESS
1   0.66 ms 10.0.2.15

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.78 seconds
```

Obrázek 39 - Výsledek nmap (Zdroj: Vlastní)

Z výsledku skenu lze vidět pouze 2 otevřené porty. Jedná se o port 21, na kterém běží FTP port je tedy správně otevřený. Dalším otevřeným portem je port 80, na kterém běží Apache server a také tady je port správně otevřený. Na serveru běží služba SSH, ale ta běží na portu 666 a ten se ve výsledku neukazuje. SSH normálně běží na portu 22, ale na serveru byl nastaven na port 666 právě kvůli zmatení před podobnými skeny systému. Výsledek dále ukazuje, že **nmap** nebyl schopný zjistit operační systém.

Nástroj **nmap** má mnoho možností skenování a jedna z nich je určení rychlosti skenu. Tato možnost se nastavuje pomocí parametru *T0* až *T5*, kdy *T0* je nejpomalejší, avšak nejdůkladnější sken. *T5* je naopak nejrychlejší a „nejhlasitější“ sken. Jelikož nemusí být sken vždy přesný, tak byl proveden ještě jeden sken pomocí příkazu: *sudo nmap -T2 -p- -A 10.0.2.15*.

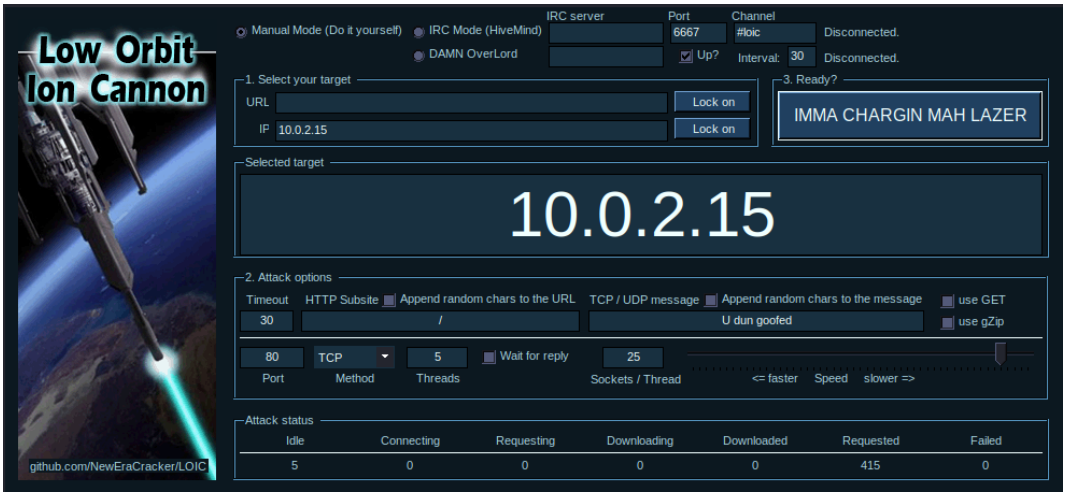


Obrázek 40 - Výsledek nmap 2 (Zdroj: Vlastní)

Na rozdíl od předchozího skenu je zde vidět otevřený port 666, na kterém běží SSH. Také je zde vidět, že **nmap** se snaží zjistit jaká je na serveru verze Linuxového jádra.

4.4.7 Denial of Service

Pro provedení DoS útoku byl použit již zmíněný nástroj LOIC (viz kapitola *3.2.1.1 DoS*). Nástroj je možné stáhnout na zde uvedeném GitHub repositáři: https://github.com/NewEraCracker/LOIC.



Obrázek 41 – LOIC (Zdroj: Vlastní)

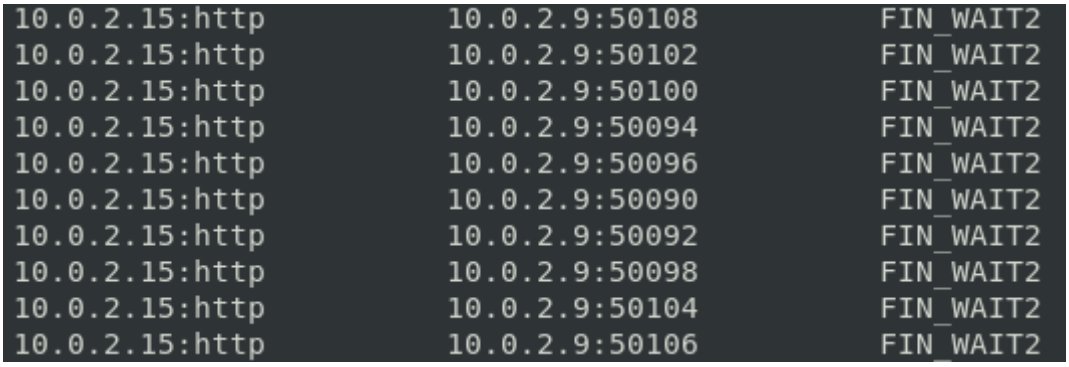
Nástroj je velmi intuitivní a nepotřebuje se zde nic upravovat. V první sekci se nastavuje IP adresa serveru, která se následně uzamkne. V druhé sekci se nastavuje, jakou metodou bude zaplavení probíhat, cílový port, počet vláken, rychlost zasílání či popřípadě zpráva, která bude v požadavcích zasílána. Ve třetí sekci už se spouští samotný útok. Pokud se útok spustí s 5 vlákny a metodou TCP, tak lze na serveru tato spojení vidět. Spojení lze zobrazit pomocí příkazu: *netstat -at*, který ukazuje TCP spojení.

```
jakub@Debian:~$ netstat -at
Aktivní Internetová spojení (servery a navázaná spojení)
Proto Přích-F Odch-F Místní Adresa          Vzdálená Adresa         Stav
tcp        0      0 localhost:ipp           0.0.0.0:*               NASLOUCHÁ
tcp        0      0 localhost:smtp          0.0.0.0:*               NASLOUCHÁ
tcp6       0      0 [::]:http               [::]:*                  NASLOUCHÁ
tcp6       0      0 [::]:ftp                [::]:*                  NASLOUCHÁ
tcp6       0      0 localhost:ipp           [::]:*                  NASLOUCHÁ
tcp6       0      0 localhost:smtp          [::]:*                  NASLOUCHÁ
tcp6       0      0 10.0.2.15:http          10.0.2.9:49788          SPOJENO
tcp6       0      0 10.0.2.15:http          10.0.2.9:49782          SPOJENO
tcp6       0      0 10.0.2.15:http          10.0.2.9:49786          SPOJENO
tcp6       0      0 10.0.2.15:http          10.0.2.9:49790          SPOJENO
tcp6       0      0 10.0.2.15:http          10.0.2.9:49780          SPOJENO
```

Obrázek 42 - Netstat tabulka (Zdroj: Vlastní)

DoS útok s pěti vlákny není na serveru poznat, hlavně i z důvodu že server daná spojení po čase sám uzavře. Aby byl útok efektivnější, tak bude muset být nastaveno více vláken a větší rychlost. Provede se útok ještě jednou se stejným nastavením a pouze se změní rychlost na 75% a 50 vláken. Pokud se uživatel pokusí nastavit velký počet vláken (např. 100), tak ho sám program upozorní, že je počet vláken moc velký a útok neumožní spustit.

Nyní je útok na serveru poznatelný, zasekává se a příkazy se vykonávají čím dál více pomaleji jelikož se postupně navazuje všech 50 spojení. Pokud se na serveru opět použije příkaz pro zobrazení spojení, tak je vidět, že některá spojení jsou již pozastavená, ale server je nestíhá uzavírat všechny.



Obrázek 43 - Uzavřená spojení (Zdroj: Vlastní)

Jelikož je útok nyní efektivní, tak si ho administrátor dokáže všimnout a jelikož se jedná o DoS útok (tedy útok pouze z jedné IP adresy), tak lze veškerá spojení na dané IP adrese zastavit příkazem: *sudo route add 10.0.2.9 reject*. Veškerá spojení z dané adresy poté zmizí z výše uvedené tabulky a v programu LOIC lze vidět, že se žádné další pakety nezasílají.

5 Výsledky a diskuse

V praktické části práce byla ukázána metodika a postup zabezpečení linuxového serveru. Tato nová metodika má za cíl sjednotit již existující metodiky a ukázat sjednocený obecný postup pro základní zabezpečení serveru.

Bylo zajištěno, aby bootloader, který se stará o spuštění systému, byl zabezpečen přihlašovacím jménem a heslem. Díky tomuto opatření se zmenší riziko, že se potencionální útočník bude pokoušet napadnout server jeho přebootováním. Přebootování by mohlo mít za následek ztrátu dat anebo ztrátu konfigurací, která zabezpečují data. Ztráta těchto zabezpečení by znamenala, že data uložená na serveru by najednou byla jednoduše přístupná.

Na serveru byl přidán sudoer uživatel, který měl za úkol zastupovat root účet a též byla zavedena přísnější politika, která apelovala na složitost při tvorbě hesla při zakládání nových uživatelských účtů. Zabezpečení přihlašování by se mohlo rozšířit o využití dvoufázové ověření, které by mohlo probíhat např. doplňujícím kódem, který přijde uživatelovi na soukromý telefon, či by se dalo využít generátoru hesla, který by zajistil jeho dostatečnou složitost. Ovšem při generování hesla vzniká pravděpodobnost, že si uživatel heslo někam poznamená pro zapamatování a to vytváří novu zranitelnost. Uživatelé mají též omezené možnosti, které by jim mohli dovolit přepnutí se na root účet, a také bylo provedeno opatření, které uživatelům nedovoluje být neaktivní a dát tak některým jedincům příležitost se zmocnit neaktivního účtu. Server byl aktualizován spolu se všemi jeho nainstalovanými balíky, aby se zabránilo zneužití nedokonalosti balíků či samotných funkcionalit systému. V systému jsou nyní zaznamenávaný incidenty, které jsou ve formě logů ukládány a také zasílány na administrátorský účet.

Dále byly na severu provedeno zabezpečení služeb SSH, FTP, Apache a Firewall. I přesto, že služba SSH je jednou z nejčastějších používaných služeb na serveru obecně, tak uživatelé tuto službu nemohou automaticky používat, ale musí jim administrátorem být přidělena práva. Zde je i možné provést rozhovor, mezi administrátorem a uživatelem, a zjistit jestli uživatel možnost vzdáleného přístupu

opravdu potřebuje. Dále bylo u SSH byla zakázaná možnost přihlásit se do systému jako root uživatel a v poslední řadě bylo zrušeno přihlášení za pomocí uživatelského jména a hesla. Nyní se pro využití SSH musí využívat bezpečnější varianta v podobě soukromých a veřejných klíčů.

U služby FTP bylo zakázáno anonymní přihlašování a pokud neexistuje patřičný důvod pro jeho existenci, tak by mělo toto přihlašování být vždy zakázáno. FTP je již starší služba, a proto zde muselo být vynucené šifrované spojení, které ovšem zabraňuje využití FTP klienta v příkazovém terminálu. Na místo toho byl využit bezpečnější FTP klient FilleZilla.

V praktické části byla provedena i konfigurace webového serveru Apache, avšak jednalo se pouze o velmi jednoduchou konfiguraci, jelikož součástí této diplomové práce nebyla nahrávána na server žádná webová prezentace či aplikace. V zabezpečení se tedy jednalo pouze o zabránění zobrazení uložených složek a dokumentů na webovém serveru a schování informací, obsahující informace o názvu a verzi webového serveru.

V poslední řadě byl změněn nástroj pro správu firewallu. Pravidla firewallu byla aplikována nástrojem iptables, ten byl nahrazen nástrojem nftables. Do firewallu byla přidána pravidla, která se snaží zabránit pokusům o zaplavení serveru. Jsou akceptovány pouze ty pakety a otevřené porty, které jsou potřebné pro správné fungování serveru a jeho aktivních služeb.

Byl proveden sken, který měl za úkol najít nežádoucí zákeřné programy či odhalit neošetřené známe existující slabiny systému. Na serveru se žádné tyto programy a ani slabiny nevyskytovaly, to však z toho důvodů. že server nevyužívají jiní uživatelé a server byl nově nainstalován a aktualizován na nejnovější verze.

V poslední části praktické práce byl proveden DoS útok, který server úspěšně ustál. Sám server se snažil veškerá špatné spojení ukončit a jelikož se jednalo o DoS útok přicházející pouze z jedné IP adresy, tak aktivní administrátor by byl schopen

zakázat danou IP adresu velmi rychle. Problém by však přicházel, pokud by se jednalo o DDoS útok, který by přicházel z vícero IP adres. Pokud by server a ani administrátor nestíhal blokovat veškerá špatná spojení, tak je možné že by musel být server vypnut anebo by v průběhu času sám spadl.

6 Závěr

V dnešní době nemusí být server něco, co využívají pouze malé či velké firmy, ale i obyčejní lidé mohou využívat služeb, které osobní server nabízí. Server se stal dostupnější široké veřejnosti nejspíše z toho důvodu, že se nemusí pořizovat drahý hardware, ale pouze stačí si od velkého množství poskytovatelů pronajmout VPS (Virtual Private Server – Virtuální Osobní Server) a využívat tak pouze tolik výpočetních zdrojů kolik uživatel skutečně potřebuje.

Cílem práce bylo sestavení nové ucelené obecné metodiky ohledně zabezpečení linuxových serverů, která vycházela z již existujících metodik a postupů. Jedná se tedy o sjednocení metodik správy systému a jeho uživatelů. Byly vybrány základní služby, které se na serverech nejčastěji vyskytují a byla popsána metodika jak tyto služby zabezpečit. Tyto metodiky byly aplikovány na vybraný server a následně byla otestována jejich efektivita.

Mezi další cíle práce patřilo definování vybraných linuxových distribucí, k čemuž byla ještě přidána definice serveru a jednotlivých typů dedikovaných serverů. V práci byla provede vícekriteriální analýza, která sloužila pro výběr linuxové distribuce, na které byla následně provedena veškerá bezpečnostní opatření. Za pomocí výsledků z analýzy byla vybraná distribuce Debian, avšak musí být bráno v potaz, že analýza může mít různé výsledky v závislosti na tom, kdo analýzu provádí, na co bude server využíván a tedy jaká budou na server stanovena kritéria.

Dalšími cíli bylo využití nové obecné metodiky pro zabezpečení a uplatnit ji na systém Debian. Jelikož se jedná o obecnou bezpečnostní metodiku, tak byly postupy konfigurace provedeny pouze jako základní opatření, která zvyšují bezpečnost systému od jeho původního nastavení po nové instalaci. Existuje zde možnost konfiguraci rozšířit na základě toho na co by měl být server využitý a jaké služby a technologie na něm poběží.

Konfigurace byly otestovány a výsledky ukázaly, že všechna nová opatření se správně aplikovala a server neukazoval žádné známky nechtěného chování. Server

obstál i DoS útok, přičemž by se nejspíše mohli objevovat problémy při DDoS útoku, ale jelikož se jedná o menší server, který by mohl sloužit pro menší podniky či jednotlivce, tak je nepravděpodobné, že by se tak to velký útok měl na uvedeném serveru provádět.

7 Seznam použitých zdrojů

ALTVATER, Alexandra. 2017. Syslog Tutorial: How It Works, Examples, Best Practices, and More. *Stackify.* [Online] 30. Červen 2017. [Citace: 17. Leden 2021.] https://stackify.com/syslog-101/.

BADJATIYA, Pinkesh. 2019. Linux Virtualization - Chroot Jail. *GeeksforGeeks.* [Online] 30. Leden 2019. [Citace: 21. Srpen 2020.] https://www.geeksforgeeks.org/linux-virtualization-using-chroot-jail/.

BARRETT, Daniel J. a SILVERMAN, Richard E. 2003. *SSH: Kompletní průvodce.* [překl.] Martin Blažík. Brno : Computer Press, 2003. 9788072268528.

BAUER, Michael D. 2005. *Linux Server Security.* 2nd Edition. Kalifornie : O'Reilly Media, 2005. ISBN: 0-596-00670-5.

BEAL, Vangie. 2020. anonymous FTP. *Webopedia.* [Online] 24. Červen 2020. [Citace: 21. Srpen 2020.] https://www.webopedia.com/TERM/A/anonymous_FTP.html.
—. 2011. Server. *Webopedia.* [Online] 2011. [Citace: 10. Červen 2020.] https://www.webopedia.com/TERM/S/server.html.

BINNIE, Chris. 2016. *Linux Server Security: Hack and Defend.* 1. místo neznámé : John Wiley & Sons, Incorporated, 2016. 9781119283096.

BRANDALL, Benjamin. 2018. 34 Linux Server Security Tips & Checklists for Sysadmins . *process.st.* [Online] 21. Březen 2018. [Citace: 21. Březen 2021.] https://www.process.st/server-security/.

BYFIELD, Bruce. 2017. 7 Reasons to Use Debian (and 3 Reasons Not To). *Datamation.* [Online] 14. Březen 2017. [Citace: 24. Červen 2020.] https://www.datamation.com/open-source/7-reasons-to-use-debian-and-3-reasons-not-to.html.
—. 2017. Bruce Byfield. *Datamation.* [Online] 24. Říjen 2017. [Citace: 7. Březen 2021.] https://www.datamation.com/author/bruce-byfield/.
—. 2013. Ubuntu Pros and Cons. *Datamation.* [Online] 19. Únor 2013. [Citace: 25. Červen 2020.] https://www.datamation.com/open-source/ubuntu-pros-and-cons-1.html.

COTTON, Ben. 2016. What is copyleft? *Opensource.* [Online] 12. Srpen 2016. [Citace: 30. Červen 2020.] https://opensource.com/resources/what-is-copyleft.

DAVYDOV, Yevgeniya. 2020. Linux Server Security: 10 Linux Hardening and Security Best Practices. *Security Boulevard.* [Online] 8. Srpen 2020. [Citace: 21. Březen 2021.] https://securityboulevard.com/2020/08/linux-server-security-10-linux-hardening-security-best-practices/.

DEBIAN. 2020. About Debian. *Debian.* [Online] 14. Srpen 2020. [Citace: 18. Září 2020.] https://www.debian.org/intro/about.en.html.
—. 2020. Reasons to Choose Debian. *Debian.* [Online] 24. Srpen 2020. [Citace: 15. Září 2020.] https://www.debian.org/intro/why_debian.en.html.

—. 2020. Security Information. *Debian.* [Online] 16. Září 2020. [Citace: 17. Září 2020.] https://www.debian.org/security/index.en.html.

DiBONA, Chris, OCKMAN, Sam a STONE, Mark. 1999. *Open Sources: Voices from the Open Source Revolution.* Sebastopol, CA : O'Reilly, 1999. 0898-1221.

DOČEKAL, Michal. 2010. Správa linuxového serveru: Linuxový firewall, základy iptables. *Linux expres.* [Online] 1. Červenec 2010. [Citace: 17. Srpen 2020.] https://www.linuxexpres.cz/praxe/sprava-linuxoveho-serveru-linuxovy-firewall-zaklady-iptables.

DOMANTAS, G. 2021. What is SSL/TLS and HTTPS? *Hostinger.* [Online] 9. Březen 2021. [Citace: 15. Březen 2021.] https://www.hostinger.com/tutorials/what-is-ssl-tls-https.

DOOLEY, Kevin. 2020. What is Syslog adn How Does It Work? *Auvik.* [Online] 5. Květen 2020. [Citace: 23. Srpen 2020.] https://www.auvik.com/franklyit/blog/what-is-syslog/.

DOSTÁLEK, Libor. 2003. *Velký průvodce protokoly TCP/IP: Bezpečnost.* 2. aktualizované vydání. Praha : Computer Press, 2003. ISBN: 80-7226-849-X.

EDUCBA. 2016. Hackers vs Crackers: Easy to Understand Exclusive Difference. *Educba.* [Online] 16. Květen 2016. [Citace: 20. Září 2020.] https://www.educba.com/hackers-vs-crackers/.

ECHEVERRI, Amy. 2015. Syslog for newbies. *Loggly Blog.* [Online] 13. Červenec 2015. [Citace: 28. Srpen 2020.] https://www.loggly.com/blog/syslog-for-newbies/.

ELIOT, Hadley. 2019. Everything You Ever Wanted to Know about CentOS. *ServerPronto.* [Online] 14. Květen 2019. [Citace: 26. Červen 2020.] https://www.serverpronto.com/spu/2019/05/everything-you-ever-wanted-to-know-about-centos/.

ELLINGWOOD, Justin a BOUCHERON, Brian. 2020. How To Edit the Sudoers File. *DigitalOcean.* [Online] 7. Červenec 2020. [Citace: 12. Srpen 2020.] https://www.digitalocean.com/community/tutorials/how-to-edit-the-sudoers-file.

ELLINGWOOD, Justin. 2014. How To Configure SSH Key-Based Authentication on a Linux Server. *DigitalOcean.* [Online] 20. Říjen 2014. [Citace: 21. Březen 2021.] https://www.digitalocean.com/community/tutorials/how-to-configure-ssh-key-based-authentication-on-a-linux-server.

FIALA, Petr, MAŇAS, Miroslav a JABLONSKÝ, Josef. 1994. *Vícekriteriální rozhodování.* Praha : Vysoká škola ekonomická, 1994. 80-7079-748-7.

FLICKENGER, Rob. 2003. *Linux server hacks.* Boston : O'Reilly Media, 2003. ISBN: 0-596-00461-3.

FORMÁNEK, David. 2018. Zranitelnosti typu injekce: SQL injekce . *Root.* [Online] 11. Říjen 2018. [Citace: 10. Srpen 2020.] https://www.root.cz/clanky/zranitelnosti-typu-injekce-sql-injekce/.

FRUHLINGER, Josh. 2019. What is a computer worm? How this self-spreading malware wreaks havoc. *CSOonline.* [Online] 6. Srpen 2019. [Citace: 11. Srpen 2020.] https://www.csoonline.com/article/3429569/what-is-a-computer-worm-how-this-self-spreading-malware-wreaks-havoc.html.
—. 2018. What is SSL, TLS? And how this encryption protocol works. *CSO Online.* [Online] 4. Prosinec 2018. [Citace: 20. Březen 2021.] https://www.csoonline.com/article/3246212/what-is-ssl-tls-and-how-this-encryption-protocol-works.html.

GANESH, Bala. 2020. What is the Linux Firewall? How to Enable Packet Filtering With Open Source Iptables Firewall? *Cyber Security News.* [Online] 21. Září 2020. [Citace: 29. Únor 2021.] https://cybersecuritynews.com/linux-firewall-iptables/.

GEEKFLARE. 2019. 11 Tools to Scan Linux Server for Security Flaws and Malware. *Geekflare.* [Online] 19. Leden 2019. [Citace: 2. Září 2020.] https://geekflare.com/linux-security-scanner/.

HASAN, Mehedi. 2018. Debian vs Ubuntu: Top 15 Things To Know Before Choosing the Best One. *Ubuntu pit.* [Online] 28. Červenec 2018. [Citace: 26. Červen 2020.] https://www.ubuntupit.com/debian-vs-ubuntu-top-15-things-to-know-before-choosing-the-best-one/.

HESS, Ken. 2019. 5 tips for getting started with Linux server security. *Red Hat.* [Online] 25. Říjen 2019. [Citace: 21. Březen 2021.] https://www.redhat.com/sysadmin/getting-started-linux-security.

HOFFMAN, Chriss. 2016. What Is a Linux Distro, and How Are They Different from One Another? *How-To Geek.* [Online] 23. Září 2016. [Citace: 18. Březen 2021.] https://www.howtogeek.com/132624/htg-explains-whats-a-linux-distro-and-how-are-they-different/.

HOMEVAGANZA. 2019. 9 Advantages and disadvantages of Ubuntu Linux Must Be Known. *Homevaganze.* [Online] 30. Prosinec 2019. [Citace: 25. Červen 2020.] https://www.homevaganza.com/9-advantages-and-disadvantages-of-ubuntu-linux-must-be-known/.

HUNT, Craig. 2003. *Linux: síťové servery.* Praha : SoftPress, 2003. 80-86497-59-3.

HUNT, Troy. 2013. What is LOIC and can I be arrested for DDoS'ing someone? *Troy Hunt.* [Online] 29. Leden 2013. [Citace: 27. Březen 2021.] https://www.troyhunt.com/what-is-loic-and-can-i-be-arrested-for/.

IVANKOV, Alex. 2019. Ubuntu Operating System: Advantages and Disadvantages. *Profolus.* [Online] 23. Srpen 2019. [Citace: 29. Červen 2020.] https://www.profolus.com/topics/ubuntu-operating-system-advantages-and-disadvantages/.

JELEN, Sara. 2019. Hacker vs Cracker: Main Differences Explained. *Security Trails.* [Online] 11. Červenec 2019. [Citace: 20. Září 2020.] https://securitytrails.com/blog/hacker-vs-cracker.

JOHN, Veena K. 2015. How To Use the Linux Auditing System on CentOS 7. *DigitalOcean.* [Online] 16. Červenec 2015. [Citace: 4. Září 2020.] https://www.digitalocean.com/community/tutorials/how-to-use-the-linux-auditing-system-on-centos-7.

KALMAN, Gergely. 2014. 10 Most Common Web Security Vulnerabilities. *Toptal.* [Online] 29. Květen 2014. [Citace: 20. Březen 2021.] https://www.toptal.com/security/10-most-common-web-security-vulnerabilities.

KAMENÍK, Pavel. 2018. Debian - distribuce pro konzervativní uživatele. *Linux EXPRES.* [Online] 13. Červen 2018. [Citace: 16. Září 2020.] https://www.linuxexpres.cz/distro/debian-distribuce-pro-konzervativni-uzivatele.

KEARY, Tim. 2020. Dos vs DDoS Attacks: The Differences and How To Prevent Them. *comparitech.* [Online] 9. Červenec 2020. [Citace: 6. Srpen 2020.] https://www.comparitech.com/net-admin/dos-vs-ddos-attacks-differences-prevention/#What_is_a_DoS_Attack.

KERNEL, Jane. 2020. What Is Syslog? Everything You Need to Know. *XPLG.* [Online] 3. Duben 2020. [Citace: 20. Březen 2021.] https://www.xplg.com/what-is-syslog-everything-you-need-to-know/.

KERNER, Sean M. 2014. CentOS 7 Comes on the Heels of Red Hat Enterprise Linux 7. *eWeek.* [Online] 7. Červenec 2014. [Citace: 18. Červenec 2020.] https://www.eweek.com/enterprise-apps/centos-7-comes-on-the-heels-of-red-hat-enterprise-linux-7.

KILI, Aaron. 2020. 10 Best Linux Server Distributions of 2020. *TecMint.* [Online] 27. Červenec 2020. [Citace: 21. Březen 2021.] https://www.tecmint.com/10-best-linux-server-distributions/.

KINSTA. 2020. How SSL Works and Why It's Important. *Kinsta.* [Online] 2. Březen 2020. [Citace: 17. Září 2020.] https://kinsta.com/knowledgebase/how-ssl-works/.

KUMAR, Chandan. 2020. 8 Essential Tips to Secure Web Application Server. *GEEKFLARE.* [Online] 29. Únor 2020. [Citace: 18. Srpen 2020.] https://geekflare.com/secure-web-application-server/.

LADIA, Aman. 2017. How Does SSH work. *Hostinger.* [Online] 3. Červen 2017. [Citace: 17. Září 2020.] https://www.hostinger.com/tutorials/ssh-tutorial-how-does-ssh-work.

LORD, Nate. 2018. What is FTP Security? Securing FTP Usage. *Digital Guardian.* [Online] 7. Září 2018. [Citace: 21. Srpen 2020.] https://digitalguardian.com/blog/what-ftp-security-securing-ftp-usage.

LUTKEVICH, Ben. 2019. [Online] 29. Leden 2019. [Citace: 20. Březen 2021.] https://searchsecurity.techtarget.com/definition/botnet.

MALENKOVICH, Serge. 2013. What is a rootkit and how to remove it. *Kaspersky daily.* [Online] 28. Březen 2013. [Citace: 15. Březen 2021.] https://www.kaspersky.com/blog/rootkit/1508/.

MARTINDALE, Jon. 2020. What is FTP? *digitaltrends.* [Online] 14. Duben 2020. [Citace: 21. Srpen 2020.] https://www.digitaltrends.com/computing/what-is-ftp-and-how-do-i-use-it/.

MCCOLLIN, Rachel. 2020. [Online] 26. Říjen 2020. [Citace: 20. Březen 2021.] https://kinsta.com/blog/what-is-a-ddos-attack/.

MELNICK, Jeff. 2020. Top 10 Most Common Types of Cyber Attacks. *netwrix.* [Online] 10. Březen 2020. [Citace: 7. Srpen 2020.] https://blog.netwrix.com/2018/05/15/top-10-most-common-types-of-cyber-attacks/#Password%20attack.

MITCHELL, Bradley. 2020. What Is a Server? *Lifewire.* [Online] 24. Červen 2020. [Citace: 10. Září 2020.] https://www.lifewire.com/servers-in-computer-networking-817380.

MITCHELL, Jeff. 2019. How to Scan a Linux Server for Malware and Rootkit. *maketecheasier.* [Online] 16. Srpen 2019. [Citace: 2. Září 2020.] https://www.maketecheasier.com/scan-linux-server-for-malware-rootkit/.

MOORE, Ben. 2020. What is GNU/Linux. *PCMag.* [Online] 11. Červen 2020. [Citace: 15. Září 2020.] https://www.pcmag.com/news/what-is-gnulinux.

MULLINS, Patrick H. 2019. 7 steps to securing your Linux server. *opensource.com.* [Online] 8. Říjen 2019. [Citace: 21. Březen 2021.] https://opensource.com/article/19/10/linux-server-security.

NEWELL, Gary. 2020. How to Create Users in Linux Using the 'useradd' Command. *Lifewire.* [Online] 22. Prosinec 2020. [Citace: 15. Březen 2021.] https://www.lifewire.com/create-users-useradd-command-3572157.
—. 2020. The Difference Between Linux and GNU/Linux. *Lifewire.* [Online] 11. Září 2020. [Citace: 17. Září 2020.] https://www.lifewire.com/what-is-linux-2201940.

NISSANKA, Ruwantha. 2020. Debsecan: You will not miss another security update. *Linuxnix.* [Online] 28. Prosinec 2020. [Citace: 7. Únor 2021.]

NORONHA, Tanya. 217. A Step-By-Step Guide on How to Configure a Firewall in Linux. *Reseller Club.* [Online] 19. Prosinec 217. [Citace: 17. Září 2020.] https://blog.resellerclub.com/a-step-by-step-guide-on-how-to-configure-firewall-in-linux/.

OZKAYA, Erdal. 2019. *Cybersecurity: The Beginner's Guide.* Birmingham : Pack Publishing Ltd., 2019. ISBN: 9781789616194.

PADAMKAR, Priya. 2019. What is Debian? *Educba.* [Online] 1. Září 2019. [Citace: 26. Leden 2021.] https://www.educba.com/what-is-debian/.

PAESSLER. IT Explained: Server. *Paessler.* [Online] [Citace: 10. Červen 2020.] https://www.paessler.com/it-explained/server.

PATEL, Pankaj. 2020. Intrusion Detection System (IDS). *GeeksforGeeks.* [Online] 16. Leden 2020. [Citace: 30. Srpen 2020.] https://www.geeksforgeeks.org/intrusion-detection-system-ids/.

PEÑA, Javier Fernández-Sanguino. 2017. Securing Debian Manual 3.19. *Debian.* [Online] 2017. [Citace: 13. Leden 2021.] https://www.debian.org/doc/manuals/securing-debian-manual/index.en.html.

PETTERS, Jeff. 2020. IDS vs. IPS: What is the Difference? *Varonis.* [Online] 29. Březen 2020. [Citace: 30. Srpen 2020.] https://www.varonis.com/blog/ids-vs-ips/.
—. 2020. What is a DDoS Attack? Identifying Denial-of-Service Attacks. *Varonis.* [Online] 29. Březen 2020. [Citace: 6. Srpen 2020.] https://www.varonis.com/blog/what-is-a-ddos-attack/.

PLESKY, Elvis. 2018. Linux Logs Explained. *plesk.* [Online] 20. Listopad 2018. [Citace: 17. Září 2020.] https://www.plesk.com/blog/featured/linux-logs-explained/.

PRAKASH, Abhishek. 2020. What is a Linux Distribution. *It's FOSS.* [Online] 6. Září 2020. [Citace: 17. Září 2020.] https://itsfoss.com/what-is-linux-distribution/.
—. 2020. Which Ubuntu install? *It's FOSS.* [Online] 3. Září 2020. [Citace: 17. Září 2020.] https://itsfoss.com/which-ubuntu-install/.

RAFFER, Dan. 2019. What is a rootkit? And how to stop them. *Norton.* [Online] 27. Březen 2019. [Citace: 11. Srpen 2020.] https://us.norton.com/internetsecurity-malware-what-is-a-rootkit-and-how-to-stop-them.html.

REGAN, Joseph. 2019. What is a trojan? *AVG.* [Online] 10. Prosinec 2019. [Citace: 12. Srpen 2020.] https://www.avg.com/en/signal/what-is-a-trojan.
—. 2019. What Is Malware? How Malware Works & How to Remove It. *AVG.* [Online] 11. Červenec 2019. [Citace: 20. Září 2020.] https://www.avg.com/en/signal/what-is-malware.
—. 2020. What is spyware? *AVG.* [Online] 2. Leden 2020. [Citace: 11. Srpen 2020.] https://www.avg.com/en/signal/what-is-spyware.

ROUSE, Margaret. 2018. What is rootkit? *SearchSecurity.* [Online] 1. Duben 2018. [Citace: 11. Srpen 2020.] https://searchsecurity.techtarget.com/definition/rootkit.

SABIH, Zaid. 2018. *Learn Ethical Hacking from Scratch.* Birmingham : Packt Publishing Ltd., 2018. 978-1-78862-205-9.

SESE, Roseanne Chiara. 2016. CentOS. *Prezi.* [Online] 25. Duben 2016. [Citace: 24. Červen 2020.] https://prezi.com/vhzzpdqnix6b/centos/.

SCHRODER, Carla. 2016. Linux. *Cleaning Up Your Linux Startup Process.* [Online] 18. Květen 2016. [Citace: 21. Leden 2021.] https://www.linux.com/topic/desktop/cleaning-your-linux-startup-process/.

SMITH, Jesse. 2021. Compare Packages Between Distributions. *DistroWatch.* [Online] Atea Ataroa Limited, 1. Leden 2021. [Citace: 7. Březen 2021.] https://distrowatch.com/dwres.php?firstlist=debian&secondlist=ubuntu&firstversions=2&secondversions=1&resource=compare-packages&view=major&refresh=Obnovit.

SSTABINSKAS, Edward. 2020. How to List Installed Packages on Ubuntu 20.04. *Hostinger.* [Online] 14. Září 2020. [Citace: 4. Leden 2021.] https://www.hostinger.com/tutorials/how-to-list-installed-packages-on-ubuntu/.

SVERDLOV, Etel. 2012. How To Set Up SSH Keys. *Digital Ocean.* [Online] 22. Červen 2012. [Citace: 13. Březen 2021.] https://www.digitalocean.com/community/tutorials/how-to-set-up-ssh-keys-2.

ŠENKYŘÍK, Daniel. 2015. Seznámení a konfigurace Apache. *Interval.* [Online] 30. Červenec 2015. [Citace: 19. Srpen 2020.] https://www.interval.cz/clanky/seznameni-a-konfigurace-s-apache/.

TAMMANY, Joyce. 2018. What is website security? *Sitelock.* [Online] 11. Říjen 2018. [Citace: 16. Leden 2021.] https://www.sitelock.com/blog/what-is-website-security/.

TARAFDER, Avik. 2019. Advantages and Disadvantages of Linux Operating System. *The Core World.* [Online] 25. Září 2019. [Citace: 25. Červen 2020.] https://www.thecoderworld.com/advantages-and-disadvantages-of-linux-operating-system/.

TERZI, Rob. 2018. How to enable sudo on Red Hat Enterprise Linux. *Red Hat.* [Online] 15. Srpen 2018. [Citace: 3. Leden 2021.] https://developers.redhat.com/blog/2018/08/15/how-to-enable-sudo-on-rhel/.

TEVAULT, Donald D. 2018. *Mastering Linux Security and Hardening.* Birmingham : Pack Publishing Ltd., 2018. ISBN: 978-1-78862-030-7.

TUNGGAL, Abi Tyas. 2020. What is a Cyber Threat? *UpGuard.* [Online] 15. Leden 2020. [Citace: 5. Srpen 2020.] https://www.upguard.com/blog/cyber-threat.

UBUNTU. What is Ubuntu? *Ubuntu Installation Guide.* [Online] [Citace: 25. Červen 2020.] https://help.ubuntu.com/lts/installation-guide/s390x/ch01s01.html.

VERMA, Adarsh. 2018. 10 Reasons To Use Ubuntu Linux. *Fossbytes.* [Online] 6. Duben 2018. [Citace: 17. Září 2020.] https://fossbytes.com/reasons-to-use-ubuntu-linux-advantage/.

WALLEN, Jack. 2017. *TechRepublic.* [Online] TechRepublic, 16. Říjen 2017. [Citace: 1. Leden 2021.] https://www.techrepublic.com/article/virtualbox-everything-the-pros-need-to-know/.
—. 2020. Ubuntu Server: A cheat sheet. *TechRepublic.* [Online] 10. Prosinec 2020. [Citace: 8. Březen 2021.] https://www.techrepublic.com/article/ubuntu-server-the-smart-persons-guide/.

WOLFSHANT, Manuel. 2020. Frequently Asked Questions about CentOS in general. *CentOS.* [Online] 2. Červen 2020. [Citace: 18. Červenec 2020.] https://wiki.centos.org/FAQ/General.

ZACKS, Aviva. 2018. What is a Backdoor and How to Protect Against it. *SafetyDetectives.* [Online] 2. Září 2018. [Citace: 12. Srpen 2020.] https://www.safetydetectives.com/blog/what-is-a-backdoor-and-how-to-protect-against-it/.