VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2017 Ivo Procházka

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

LABORATORNÍ ÚLOHA PREZENTUJÍCÍ APLIKAČNÍ FIREWALL
LABORATORY TASK DEMONSTRATES APPLICATION FIREWALL

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

Ivo Procházka

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. Zdeněk Martinásek, Ph.D.

BRNO 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Ivo Procházka ***ID:*** 125293
Ročník: 3 ***Akademický rok:*** 2016/17

NÁZEV TÉMATU:

Laboratorní úloha prezentující aplikační firewall

POKYNY PRO VYPRACOVÁNÍ:

Hlavním cílem bakalářské práce je navrhnout a vypracovat laboratorní úlohu seznamující studenty s principem aplikačních firewallů. Nejprve nainstalujte a zprovozněte experimentální pracoviště (virtualizované F5, OWASP a uživatelská stanice). Prostudujte základní konfiguraci firewallu a navrhněte laboratorní úlohu, která demonstruje ochrany proti útokům cílených na webové aplikace (vyberte nejméně 5 zranitelností). Realizované dílčí kroky včetně zadání laboratorní úlohy budou přehledně zpracovány.

DOPORUČENÁ LITERATURA:

[1] STALLINGS, William. Cryptography and network security: principles and practice. Seventh edition. 731 pages. ISBN 01-333-5469-5.

[2] FADYUSHIN, Vyacheslav a Bruce HYSLOP. Instant penetration testing: Setting up a test lab how-to. 1. vyd. Birmingham: Packt Publishing, 2013, 74 s. ISBN 978-1-84969-412-4.

Termín zadání: 1.2.2017 ***Termín odevzdání:*** 8.6.2017

Vedoucí práce: Ing. Zdeněk Martinásek, Ph.D.
Konzultant:

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

ABSTRAKT

Cílem této práce je návrh laboratorní úlohy prezentující aplikační firewall. Teoretická část úlohy se věnuje formálnímu rozdělení firewallů z historického a technického hlediska. Praktická část obsahuje demonstraci pěti zranitelností ze seznamu OWASP Top 10 společně s návody, jak zabránit útokům na tyto zranitelnosti pomocí platformy F5 BIG-IP. Práce obsahuje také návrh konfigurace ochrany před útoky typu Denial of Service a load balancingu na platformě F5.

KLÍČOVÁ SLOVA

Firewall F5 BIG-IP OWASP laboratorní úloha

ABSTRACT

The aim of this thesis is to design a laboratory exercise presenting an application firewall. The first part describes various types of firewalls from both the historical and technical points of view. The second section presents five vulnerabilities from the OWASP Top 10 list, five vulnerabilities from OWASP Top 10 list are presented including a guide on how prevent these types of attacks using the F5 BIG-IP platform. The thesis also includes a presentation of load balancing and Denial of Service protection on the F5 platform.

KEYWORDS

Firewall F5 BIG-IP OWASP laboratory excercise

PROCHÁZKA, Ivo *Laboratorní úloha prezentující aplikační firewall*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2017. 56 s. Vedoucí práce byl Ing. Zdeněk Martinásek, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Laboratorní úloha prezentující aplikační firewall“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

Brno ....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Zdeňku Martináskovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno ....

podpis autora

Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic

http://www.six.feec.vutbr.cz

PODĚKOVÁNÍ

Výzkum popsaný v této bakalářské práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno . . . . . .

. . . . . .

podpis autora

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

ÚVOD

V současné době dochází k překotnému rozvoji počítačových sítí. Dle odhadů společnosti Gartner můžeme kolem roku 2020 očekávat více než 20 miliard zařízení připojených k síti Internet. Ruku v ruce s tímto rychlým rozvojem rostou i požadavky na poskytovatele připojení, aby zabezpečili svoje sítě proti útokům [4].

V minulosti plnily roli firewallů routery, které na základě hlavičky každého paketu rozhodovaly, zda pakety dorazí do cíle nebo ne. Toto řešení bylo dlouhodobě neudržitelné, a proto na konci 80. let vznikly první firewally, jež mají za úkol zajistit bezpečné oddělení zařízení v sítích [5].

V úvodu je čtenář seznámen s problematikou firewallů, jejich typy a využitím v praxi. Následně rozebereme formální rozdělení firewallů, jak z historického, tak z technického hlediska. U každé technologie stručně zmíníme výhody, nevýhody a případná omezení. Následující kapitola stručně charakterizuje jednotlivé typy útoků na webové aplikace.

Cílem práce je návrh a realizace laboratorní úlohy, která studenty seznámí s funkcí aplikačních firewallů a názorně ukáže možnosti obrany webových aplikací před nejčastějšími typy útoků. První část samotné úlohy tvoří stručný úvod pro studenty, jenž je seznámí se zapojením pracoviště a s použitými technologiemi. Druhou částí je návod pro cvičící k samotné úloze.

1 FIREWALLY

Termín firewall původně pochází ze stavebnictví, kde označoval protipožární zeď. Později se používal obecně pro jakoukoliv překážku, která měla za úkol zastavit šíření požáru (např. u automobilů, v letectví atd.). V informačních technologiích firewall označuje zařízení, jehož úkolem je oddělení dvou či více sítí. První firewally se začaly objevovat na konci 80. let a od té doby se dále vyvíjejí.

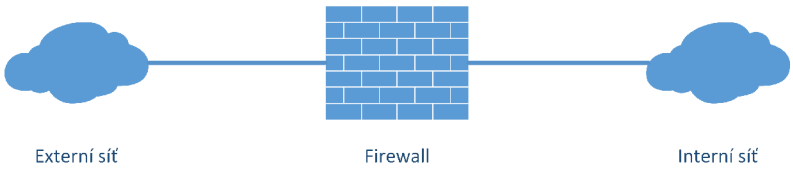
V současné době se firewally využívají jako efektivní způsob ochrany vnitřní počítačové sítě před útoky zvenku a zároveň umožňují připojení k vnější síti. Firewall samotný je prvek, který má za úkol zajistit kontrolu nad síťovým provozem, a to ve formě síťového zařízení či programového vybavení na koncové stanici (viz dále).

Dle [1] jsou cíle firewallu následující:

- Veškerý síťový provoz z vnější sítě do vnitřní (a naopak) musí projít skrze firewall. Toho je dosaženo fyzickým umístěním firewallu mezi vnější a vnitřní síť, jak je vidět na obrázku 1.1.
- Skrze firewall bude propuštěn pouze povolený síťový provoz podle předem daných pravidel. Tato pravidla zároveň implikují, že v rámci sítě je definována určitá bezpečnostní politika.
- Firewall samotný je imunní proti útokům zvenčí. Toto vyžaduje použití zabezpečeného operačního systému a omezení přístupu k zařízení.

V práci [9] jsou definovány následující čtyři obecné techniky, které firewally využívají k vynucení bezpečnostní politiky:

- Kontrola nad službami – podle typu služby firewall rozhodne, zda má být provoz povolen, či zablokován. Tento filtr může být založen na kontrole hlavičky paketů či na typu služby.
- Kontrola nad směrováním – zajišťuje kontrolu nad možností založení spojení klientů mezi vnější a vnitřní sítí.
- Kontrola nad uživatelem – umožňuje zabránit uživatelům v přístupu ke službám, ke kterým nemají autorizaci.
- Kontrola chování – tato technika kontroluje způsob používání jednotlivých služeb.



Obr. 1.1: Schéma zapojení firewallu mezi sítěmi

1.1 Výhody a nevýhody firewallů

Jak bylo naznačeno výše, firewally jsou považovány za přínosná zařízení. Jejich primárním úkolem je zastavení nežádoucího síťového provozu (jak příchozího, tak odchozího) ještě předtím, než dojde k vniknutí do samotné sítě. Samotný firewall slouží jako jediný vstupní a výstupní bod sítě. Síť je tedy chráněna před neautorizovaným vstupem (např. viry, útoky hackerů), ale firewall má také zabránit šíření nákazy počítačovými viry mimo síť.

Firewall také umožňuje sloučení více funkcí v jediném fyzickém zařízení. Je tedy možné, aby plnil roli např. routeru, překladu síťových adres (Network Address Translation – NAT) či analyzátoru využití sítě. Kombinováním těchto funkcí v jednom zařízení je možné snížit cenu za pořízení síťové infrastruktury i cenu za její provoz a údržbu.

Zařízení zároveň slouží k zaznamenávání, analýze a auditování síťového provozu, takže je možné je použít i pro přímou detekci či analýzu bezpečnostních rizik v infrastruktuře. V neposlední řadě je na firewallech možné provozovat Virtual Private Network (VPN) server, který umožňuje uživatelům připojovat se do vnitřní sítě zvenku [1].

Hlavní nevýhodou těchto zařízení je, že jejich použití velice často představuje slabé místo v infrastruktuře. V případě potíží, které způsobí pád či znepřístupnění firewallu, celá síť ztratí kontakt s vnějším světem. Toto je patrné například v případě útoku typu Denial of Service (DoS), kdy zahlcením vnějších firewallů může dojít k znepřístupnění vnější sítě. Stejný efekt má i chybná konfigurace firewallu nebo chyba v softwaru.

Dalším problémem je skutečnost, že použití těchto zařízení má vliv na rychlost připojení. Je to proto, že firewally velice často musí pro správnou funkčnost analyzovat příchozí a odchozí síťový provoz, čímž mohou výrazně ovlivňovat rychlost připojení.

Firewally je třeba brát jako jeden z dostupných nástrojů k zajištění síťové bezpečnosti; samotné použití firewallu nezajistí bezpečnou síť. Firewally nemohou zabránit všem typům útoků a nejsou schopny útoky odrážet bez správné konfigurace [1].

2 TYPY FIREWALLŮ

2.1 Dělení dle typu hostitele

Firewally dělíme podle typu hostitele do následujících skupin:

2.1.1 Hraniční pevnost (bastion host)

Jedná se o bod v síťové infrastruktuře, který byl identifikován síťovým správcem jako klíčový. Jeho zapojení je vidět na obrázku 1.1. Obvykle se tato zařízení instalují přímo mezi vnější a vnitřní síť. Fungují jako aplikační či okruhové brány. Podle [10] má tento typ následující vlastnosti:

- Běží na zabezpečeném hardwaru a softwaru. Na zařízeních funguje verze operačního systému (např. Unix či Linux), která je výrobcem upravena tak, aby bylo dosaženo vyšší úrovně zabezpečení.
- Na těchto zařízeních se provádí pouze služby, které jsou z pohledu administrátora velmi důležité.
- Zařízení obvykle před poskytnutím služby vyžaduje určitou formu autentizace.
- Jednotlivé proxy moduly jsou nakonfigurovány tak, aby podporovaly pouze předem nastavené dílčí služby.
- Zařízení vždy protokoluje veškerý provoz, což je zásadní pro analýzu a detekci průniků do sítě.
- Každý modul je navržen jednoduše, aby byla zajištěna snadná testovatelnost a auditovatelnost kódu.
- Všechny moduly v zařízení jsou vzájemně nezávislé.
- Samotný provoz zařízení se zpracovává v operační paměti zařízení a na disk se přistupuje pouze během inicializace systému. Tím se zvyšuje odolnost zařízení proti infekci.
- Proxy moduly fungují pod uživatelským účtem, který má omezená práva k přístupu do operačního systému daného zařazení.

2.1.2 Hostitelské firewally

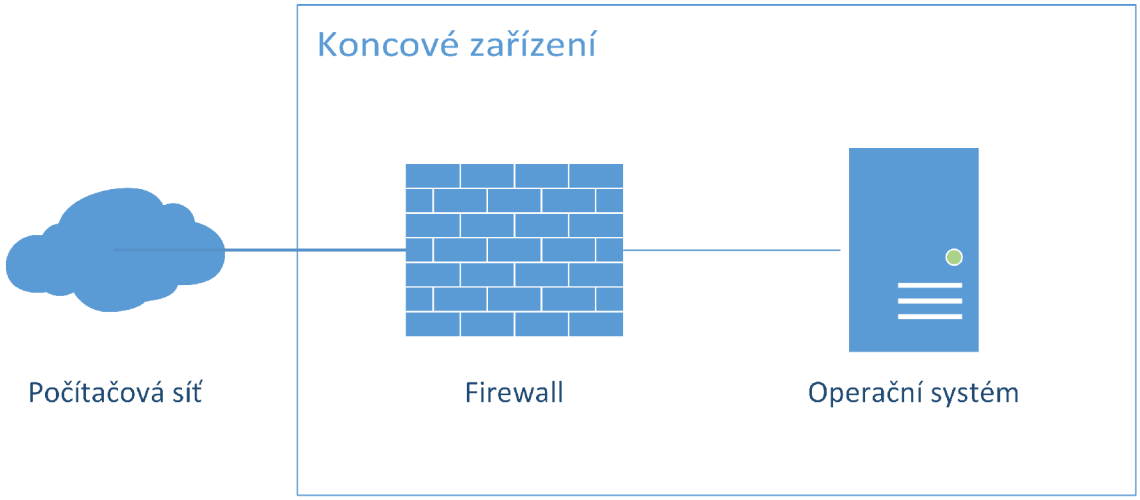
Na rozdíl od bastion hostů běží tyto firewally jako aplikace přímo v hostitelském operačním systému. Schéma jejich použití je zobrazeno na obrázku 2.1. Z pohledu systému, ve kterém pracují, je dělíme na hostitelské firewally a osobní firewally.

Hostitelské firewally se spouštějí na serverech. To znamená, že jejich konfiguraci je možné upravit podle potřeb aplikace, která na serveru běží. Často se jedná o přídavné balíčky do operačního systému, které se po nainstalování nakonfigurují podle

potřeb síťového správce. Hlavní výhody jsou tyto:

- Jak bylo zmíněno výše, jejich konfiguraci lze provést přesně podle potřeb hostované aplikace.
- Jsou nezávislé na síťové topologii.
- Nové stanice lze do sítě připojovat s přednastavenou bezpečnostní politikou, což zvyšuje celkovou úroveň zabezpečení sítě.

Osobní firewally jsou typem softwaru, který běží na stanicích koncových uživatelů. Jejich úkolem je dohlížet na provoz na koncových stanicích a případně ho omezovat. Dříve se jednalo o jednodušší firewally, v současné době se ale rozdíl mezi osobními a hostitelskými firewally stírá (např. Windows Firewall, který je součástí instalace Windows 7, je shodný s firewallem použitým ve verzi Windows Server) [10], [6].

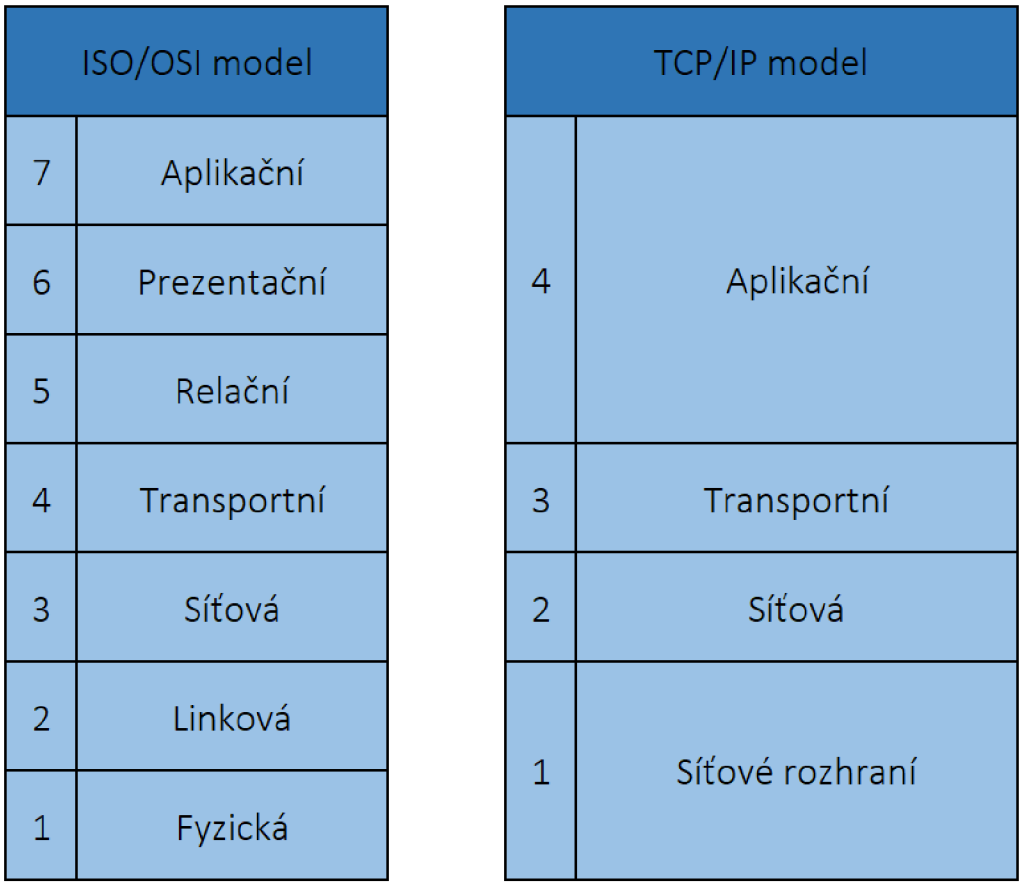


Obr. 2.1: Schéma hostitelského firewallu

2.2 Dělení dle síťových vrstev

Na obrázku 2.2 můžeme vidět jednotlivé vrstvy ISO/OSI modelu a jeho srovnání s TCP/IP modelem. Firewally mohou pracovat v různých vrstvách jednotlivých modelů. Nejnižší možnou vrstvou je 3. vrstva ISO/OSI, respektive 2. vrstva modelu TCP/IP. Na této vrstvě totiž dokáže firewall jedinečně určit zdrojovou a cílovou adresu paketu, a tudíž rozhodnout, zda paket pochází z důvěryhodného zdroje. Firewally pracující na vyšších vrstvách jsou následně o paketu schopné získat více informací, a dokážou ho tedy lépe analyzovat a rozhodnout, zda má být paket vpuštěn do vnitřní sítě. To, že firewall má o paketu více informací, ale nemusí nutně znamenat, že je zajištěna vyšší nebo lepší úroveň zabezpečení [9].

Z historického hlediska lze firewally rozdělit do následujících kategorií:

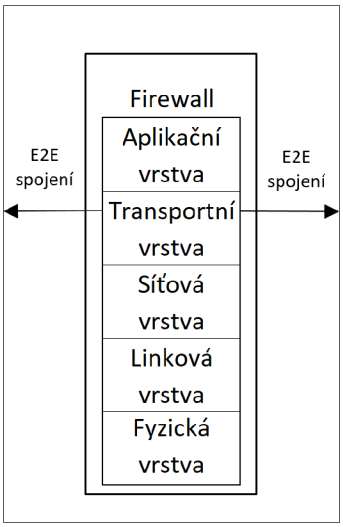


Obr. 2.2: Referenční ISO/OSI model a srovnání s TCP/IP modelem

2.2.1 Paketové filtry

Na obrázku 2.3 je znázorněno schéma paketového filtru, kde je vidět, že spojení mezi koncovými body (takzvané End to End (E2E) spojení) se realizuje na transportní vrstvě. Komunikace je tedy pro aplikace plně transparentní. Jedná se o nejjednodušší typ síťového filtru, který provádí analýzu hlavičky každého datagramu a poté se rozhodne, zda paket přepošle, či zahodí. Zařízení typicky implementují dvě sady pravidel pro příchozí a odchozí komunikaci a analyzují se následující pole v paketu:

- **Zdrojová IP adresa** – adresa zařízení, ze kterého pochází daný paket (např. 10.0.0.1).
- **Cílová IP adresa** – adresa zařízení, ke kterému paket směřuje (např. 10.0.0.2).
- **Zdrojová a cílová adresa transportní vrstvy** – adresa na transportní vrstvě (např. číslo TCP či UDP portu), tyto informace jsou obvykle definovány protokoly vyšších vrstev.
- **Typ IP protokolu** – definuje protokol transportní vrstvy.
- **Rozhraní** – tento parametr se objevuje u zařízení s více rozhraními a definuje, pro které síťové rozhraní dané pravidlo platí [10].



Obr. 2.3: Schéma paketového filtru

Paketové filtry obvykle umožňují definici vlastních pravidel, podle nichž zařízení filtruje vlastní síťový provoz. Tato pravidla jsou obvykle uložena v tabulce či databázi a výše uvedené parametry se porovnávají s uloženými hodnotami. Pokud není v tabulce nalezeno odpovídající pravidlo, použije se výchozí (default) pravidlo. U výchozích pravidel se používají dva přístupy:

- **Výchozí zahození paketu (pozitivní bezpečnostní model)** – síťový provoz, který není explicitně povolen, je zahozen.

- **Výchozí přeposlání paketu (negativní bezpečnostní model)** – síťový provoz, který není zakázán, je povolen [10].

Režim výchozího zahození paketu se považuje za konzervativnější a bezpečnější přístup, a to proto, že po úvodní konfiguraci je veškerý provoz zakázán a konkrétní služby se následně povolují podle potřeby. Pro síťové uživatele je tento přístup transparentní, protože v tomto případě jsou pro ně povoleny pouze předem definované služby a ostatní služby (např. přístup na internet) jsou zakázány. Toto řešení se obvykle používá v sítích, kde je vyžadována vyšší úroveň zabezpečení (soukromé či státní organizace). Výchozí přeposlání paketu je z pohledu správy sítě jednodušší na implementaci. Zakazovány bývají pouze určité adresy (či rozsahy adres), porty, protokoly či služby. Tento přístup bývá obvyklejší v menších či univerzitních sítích, kde se blokují konkrétní služby podle potřeby (obvykle nově objevené zranitelnosti nebo nežádoucí služby) [10].

Výhody a nevýhody paketových filtrů

Paketové filtry mají tyto výhody:

- Jedná se o metodu vyznačující se vysokou rychlostí analýzy paketů. Z toho plyne, že dopad na rychlost sítě je minimální.
- Systémy lze relativně jednoduše konfigurovat a umožňují širokou škálu možných nastavení. Také jsou velice flexibilní ve smyslu implementace bezpečnostní politiky.
- Jedná se o levné řešení, které se tedy často využívá v prostředích, kde je kladen důraz na nízkou cenu [10].

Mezi nevýhody patří:

- I když je samotná konfigurace jednoduchá, správné nastavení politik je mnohdy složité (a drahé). Pravidla také mohou neúmyslně zablokovat nezávadný či důležitý síťový provoz.
- Nedokážou zastavit útoky na vyšších vrstvách (např. na aplikační vrstvě).
- Mohou být zranitelné vůči určitým typům cílených útoků na TCP/IP protokol (např. v případě, že se útočník pokusí podvrhnout hlavičku datagramu, aby byl do sítě vpuštěn nežádoucí síťový provoz).
- Neumožňují ověřování pravosti zdrojových dat [10].

V tabulce 2.1 je vidět ukázka sady pravidel pro paketový filtr. Pravidla A a B jsou ukázkou pravidel blokování konkrétní služby na základě předem známého portu. Pravidlo C slouží jako ukázka blokace všech portů nad určitou hranicí. Pravidla D a E ukazují možnost blokování provozu u zařízení s předem známou IP adresou. V tomto kontextu může PC1 být zařízení, u kterého není připojení do více sítí nutné, zatímco u PC2 je spojení s vnějším světem vyžadováno.

Toto řešení je sice nejstarší, ale stále se aktivně používá. Díky vysoké rychlosti se využívá jako vstupní zařízení do sítě – tzv. perimeter firewall. Protože je levné, používá se i v levnějších routerech, kde zajišťuje základní ochranu. V každém případě je vhodné paketový filtr vždy doplnit o firewall pracující na vyšší síťové vrstvě [10].

Možnosti útoků na paketové filtry

- **Podvržení IP adresy** – útok spočívá v tom, že útočník na firewall pošle paket s podvrženou IP adresou (např. se na externí rozhraní odešle paket se zdrojovou IP adresou z vnitřní sítě). Útočník doufá, že podvržením IP adresy dojde k přeposlání paketu proto, že na firewallu bude existovat pravidlo pro zařízení s podvrženou IP adresou. Těmto útokům lze zabránit kontrolou doručení paketu na rozhraní odpovídající zdrojové IP adrese.
- **Využití zdrojového směrování** – tento typ útoku využívá faktu, že zdrojová stanice určuje cestu, kterou paket použije při cestě přes síť. Tato cesta může byt definována tak, že by paket mohl obejít bezpečnostní prvky, jež analyzují směrovací informace. Obranou proti tomuto typu útoku je zahodit všechny pakety, které cestu takto určují.
- **Útok pomocí malých paketů** – útočník v tomto případě využije možnosti IP fragmentace k vytvoření malých fragmentů – dojde k rozdělení TCP hlaviček do více paketů. Tento typ útoku obchází filtry, které jsou závislé na údajích v TCP hlavičce. Zařízení obvykle rozhodnou o tom, zda paket zahodit, či přeposlat, na základě prvního fragmentu a další fragmenty se již nekontrolují. Útočník doufá, že firewall provede kontrolu pouze prvního fragmentu a všechny následující přepošle dále. Tomuto útoku lze předejít tak, že se na firewallu nakonfiguruje minimální sada informací, které musí hlavička transportní vrstvy obsahovat. Firewall se v takovém případě bude chovat standardně a pokud úvodní pakety obsahují dostatečné množství informací, budou i následující pakety přeposlány nebo zahozeny podle předem definovaných pravidel [10].

Tab. 2.1: Ukázka sady pravidel pro paketový filtr

Pravidlo A					
Akce	Zdr. adresa	Cíl. adresa	Cílový port	Protokol	Komentář
Blokovat	*	*	80	TCP, UDP	Blokace WWW

Pravidlo B					
Akce	Zdr. adresa	Cíl. adresa	Cílový port	Protokol	Komentář
Povolit	*	*	25	TCP, UDP	Povolení SMTP

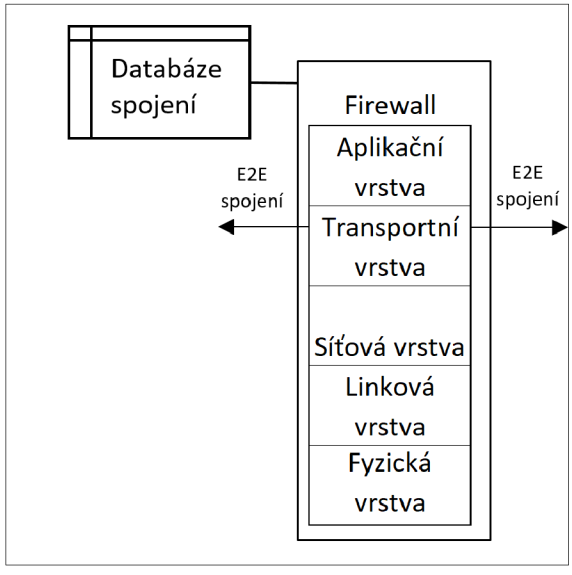
Pravidlo C					
Akce	Zdr. adresa	Cíl. adresa	Cílový port	Protokol	Komentář
Blokovat	*	*	<1024	TCP, UDP	Blokace nerezervovaných portů

Pravidlo D					
Akce	Zdr. adresa	Cíl. adresa	Cílový port	Protokol	Komentář
Blokovat	*	10.0.0.100	*	TCP, UDP	Izolace PC1

Pravidlo E					
Akce	Zdr. adresa	Cíl. adresa	Cílový port	Protokol	Komentář
Povolit	*	10.0.0.101	*	TCP, UDP	Povolení provozu směrem k PC2

2.2.2 Stavový firewall

Na obrázku 2.4 je schéma stavového firewallu. Jednou z hlavních nevýhod paketových filtrů je skutečnost, že zařízení každý paket posuzuje individuálně a nezávisle na předchozím či následujícím paketu. Toto adresují tzv. stavové firewally, které dokážou rozeznat kontext pro daný paket.



Obr. 2.4: Schéma stavového firewallu

Velká většina síťových aplikací v současné době pracuje na principu klient-server.

Tento princip vyžaduje, aby byl před zahájením samotné komunikace definován kanál, na kterém bude klient se serverem komunikovat. Například v případě použití Hypertext Transfer Protocol (HTTP) je spojení iniciováno od klienta směrem k serveru pomocí síťového protokolu TCP. Spojení pomocí protokolu TCP probíhá tak, že klient odešle žádost na daný server na vybraném zdrojovém portu (typicky v rozsahu 1024–65532) a na cílovém portu (pro HTTP protokol obvykle 80 či 8080, ale může být použit i jiný). Server žádost o spojení přijme a odešle zpět odpověď na portu odpovídajícím zdrojovému portu. Tím je TCP spojení mezi zařízeními navázáno a jeho prostřednictvím následně probíhá komunikace pomocí HTTP protokolu. Toto spojení je tvořeno jednotlivými pakety. Kontextem paketu rozumíme jeho příslušnost k danému spojení.

V případě paketových filtrů by firewall prováděl analýzu jednotlivých paketů a pro každý paket by se zvlášť rozhodovalo o tom, zda jej firewall propustí či ne. Stavové firewally fungují obdobně, ale navíc si udržují tabulku, respektive databázi, obsahující seznam aktivních spojení.

Ve velké většině případů aplikace, které vytvářejí TCP spojení, využívají jako cílový port některý z rozsahu 1–1023 (tyto porty se označují jako takzvané známé porty) a jako zdrojový port některý z rozsahu 1024–65532. Zdrojový port je pro samotnou aplikaci relevantní pouze po dobu aktivního spojení a po jeho ukončení je opět uvolněn. Tohoto využívají stavové firewally; monitorují spojení pro porty v rozsahu 1–1023 a provoz na vyšších portech povolí pouze v případě, že se dané spojení nachází v databázi [10].

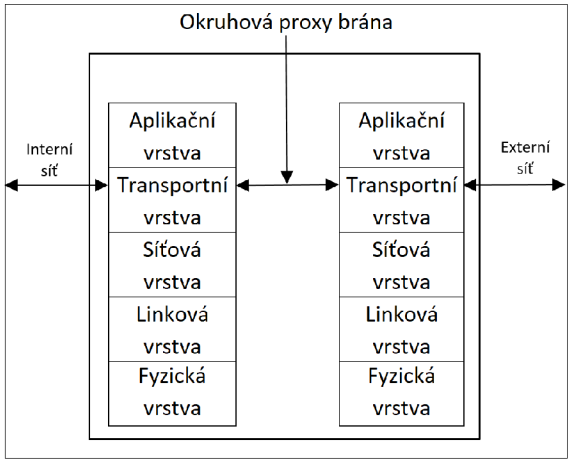
Tab. 2.2: Ukázka tabulky aktivních spojení stavového firewallu

Zdrojová adresa	Zdrojový port	Cílová adresa	Cílový port	Stav spojení
192.168.0.241	50418	77.75.79.39	80	NAVÁZÁNO
192.168.0.241	50452	9.25.247.112	12975	NAVÁZÁNO
192.168.0.241	50418	172.217.23.206	80	NAVÁZÁNO
192.168.0.241	50939	162.125.66.3	443	NAVÁZÁNO
192.168.0.225	54164	172.217.23.206	80	NAVÁZÁNO
192.168.0.225	53655	77.75.79.39	80	NAVÁZÁNO
192.168.0.225	56695	198.47.127.27	443	NAVÁZÁNO

2.2.3 Okruhové proxy brány

Z obrázku 2.5 je patrné, že oproti předchozím firewallům se u okruhové proxy brány již počítá s oddělením dvou či více sítí. Okruhové brány stejně jako paketové filtry a stavové firewally pracují na transportní vrstvě TCP/IP modelu. Okruhová brána,

na rozdíl od dříve uvedených řešení, navazuje dvě spojení – jedno mezi uživatelem a bránou a druhé mezi bránou a cílovým serverem. Samotný síťový provoz se poté přeposílá mezi koncovými zařízeními bránou, ale neprovádí žádnou aktivní analýzu procházejícího provozu. Samotná brána zvyšuje zabezpečení tak, že umožňuje síťovým správcům definovat, jaký typ provozu je v dané síti povolen a jaký ne.



Obr. 2.5: Schéma okruhové brány

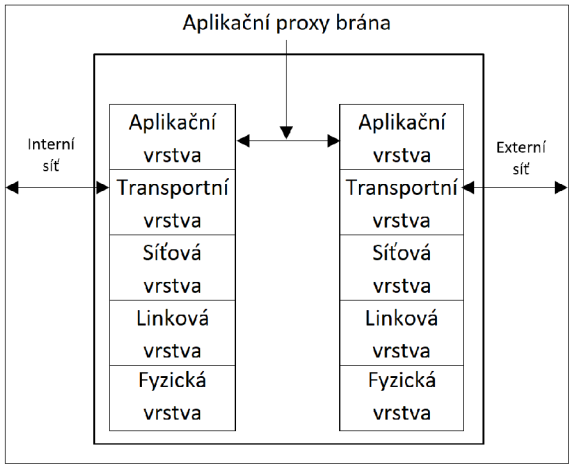
Tento typ se využívá zejména v případech, kdy síťový administrátor důvěřuje vnitřní síti, ale je vyžadována vyšší úroveň zabezpečení sítě před útoky zvenčí. V praxi se tyto filtry používají v kombinací s aplikační bránou (viz dále) – v takovém případě jsou zařízení nakonfigurována tak, že odchozí spojení přeposílají přes okruhovou bránu, zatímco příchozí spojení se zpracovávají pomocí aplikační brány.

Typickým zástupcem tohoto typu firewallu je protokol (respektive balík) SOCKS, který je považován za de facto standard pro okruhové brány [10].

2.2.4 Aplikační proxy brány

Z obrázku 2.6 je patrné, že jako v případě okruhové brány se u aplikační brány (či proxy serveru) již počítá s oddělením dvou či více sítí. Na rozdíl od předchozích případů aplikační brána pracuje s provozem na aplikační vrstvě v TCP/IP modelu. Základem tohoto řešení je, že uživatel kontaktuje server na předem známém protokolu (HTTP, FTP a tak dále), je bránou ověřen a aplikační brána následně odešle žádost na cílový server a přepošle veškerou komunikaci zpět k uživateli.

Je patrné, že toto řešení je bezpečnější. Místo definování kombinací IP adres a portů se totiž v rámci bezpečnostní politiky definují povolené služby. Pokud se uživatel pokouší získat přístup ke službě, která není na aplikační bráně povolena či implementována, není mu přístup k cílové aplikaci či serveru umožněn.



Obr. 2.6: Schéma aplikační brány

Zpracování provozu na aplikační vrstvě také znamená, že síťový provoz lze snadno protokolovat, auditovat či aktivně analyzovat. Možnosti aktivní kontroly průchozího provozu lze využít k přímé ochraně aplikací či uživatelů ve vnitřní síti. Zařízení toho typu zároveň mohou integrovat větší množství síťových funkcí (např. vyvažování síťového provozu, akceleraci aplikací či aktivní antivirovou ochranu a podobně).

Nevýhodou tohoto řešení je, že aktivní analýza síťového provozu může být velmi náročná na výpočetní výkon a může významně omezovat spojení. Pro každé odchozí spojení je navíc třeba navázat dvě různá spojení, která aplikační brána musí udržovat [10].

Web Application Firewall (WAF)

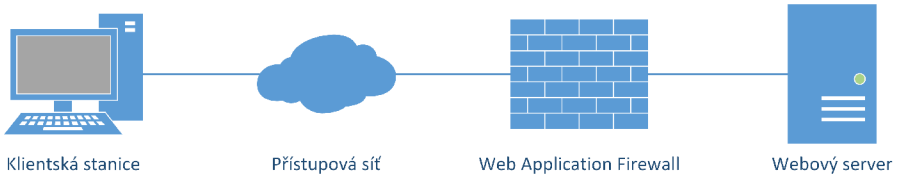
Podmnožinou aplikačních bran jsou takzvané Web Application Firewall (WAF). Rozdíl mezi síťovým firewallem a WAF je, že WAF je navržen tak, aby byl schopen filtrovat provoz pro webové aplikace, zatímco síťový firewall slouží primárně k oddělení provozu mezi servery [13].

Uživatelé k webovým aplikacím obvykle přistupují pomocí HTTP protokolu a WAF pracuje tak, že analyzuje provoz na tomto protokolu. WAF bývají obvykle umístěny logicky mezi uživatele a webové servery a procházející síťový provoz analyzují a porovnávají s databází zranitelností. Případný zjištěný nežádoucí provoz firewall zaznamená nebo přímo zablokuje. Cílem těchto zařízení je preventivně blokovat známé typy útoků (například SQL injection, Cross Site Scripting (XSS)) [13].

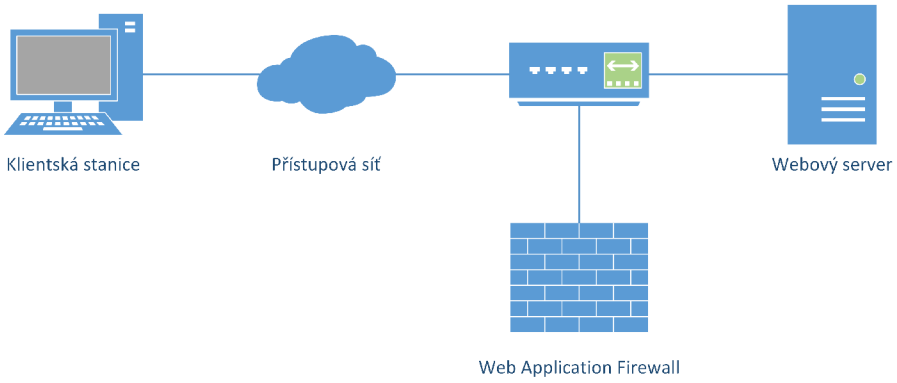
Výhodou těchto zařízení je, že při jejich použití není třeba žádným způsobem upravovat webové aplikace, jež mají být takto chráněny. Tento způsob implementace umožňuje chránit známé zranitelnosti i u aplikací, do jejichž kódu je náročné či nemožné zasahovat. Tato vlastnost je v rámci podnikových sítí velice žádaná.

Nejčastěji se WAF implementují jako:

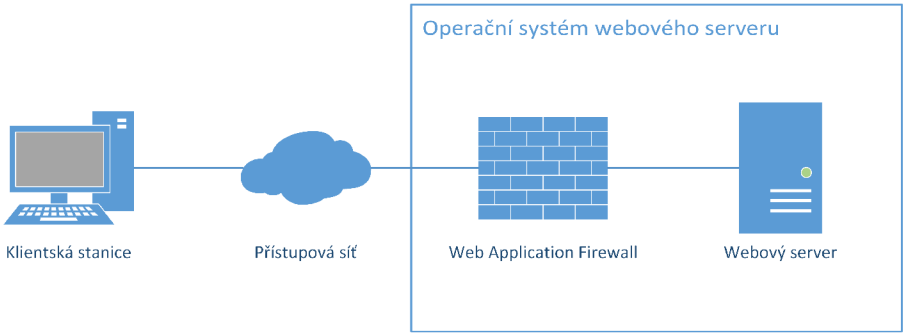
- **Reverzní proxy** – při implementaci WAF jako reverzní proxy je aplikační brána umístěna mezi klienty a webovým serverem. Schéma je vidět na obrázku 2.7. Veškeré požadavky se nejprve odesílají na aplikační bránu, která provede analýzu dat před odesláním na samotný webový server. Práce daty v tomto případě probíhá na 7. vrstvě referenčního ISO/OSI modelu a aplikační brána v případě potřeby provádí šifrování či dešifrování síťového provozu. Tento přístup zajišťuje vysokou úroveň zabezpečení webových aplikací, protože nežádoucí žádosti nikdy nedorazí na samotný webový server.
- **Layer 2 (L2) most** – jedná se o podobný přístup jako v případě reverzní proxy (viz 2.7). Hlavním rozdílem v tomto případě je, že provoz se přeposílá na 2. vrstvě referenčního ISO/OSI modelu. L2 mosty bývají obvykle rychlejší než reverzní proxy a jejich implementace v podnikových sítích je snazší. Nevýhodou je menší rozsah možností konfigurace bezpečnostních politik (např. nemožnost dešifrovat síťový provoz).
- **„Out-of-Band“ řešení** – schéma tohoto typu implementace je zobrazeno na obrázku 2.8. V tomto případě není WAF umístěn přímo mezi webový server a přístupovou síť. Využívá se zde síťového přepínače, který je nakonfigurován tak, aby WAF dostal kopii provozu, který směřuje na webový server. Výhodou tohoto řešení je, že implementace WAF nemá takřka žádný vliv na použitou síťovou infrastrukturu. Nevýhodou je však omezení možností blokování nevyžádaného provozu pouze na odeslání TCP-reset paketů (k přerušení síťového provozu), a proto se toto řešení využívá pouze k monitorování webových aplikací.
- **Implementace na serveru** – WAF je implementován jako dodatečná aplikace nebo plugin na webovém serveru. Toto řešení funguje na stejném principu jako reverzní proxy. Výhodou je nižší cena a odstranění potenciálního slabého místa v síťové infrastruktuře. Nevýhodou je, že tento přístup odčerpává zdroje na webovém serveru a nemusí tedy být vhodný pro všechny typy aplikací. Schéma tohoto typu implementace je zobrazeno na obrázku 2.9.
- **Cloudové řešení** – toto moderní řešení implementace WAF funguje obdobně jako reverzní proxy. Na rozdíl od něj je ale WAF ochrana přenesena na externího dodavatele. Schéma je zobrazeno na obrázku 2.10 a je z něj patrné, že WAF je poskytován jako součást přístupové sítě (např. přesměrováním DNS záznamů aplikace na poskytovatele WAF) [8].



Obr. 2.7: Schéma reverzní proxy a L2 mostu



Obr. 2.8: Schéma „Out-of-Band“ řešení



Obr. 2.9: Schéma implementace WAF v rámci hostitelského operačního systému

Obr. 2.10: Schéma cloudového řešení

3 OPEN WEB APPLICATION SECURITY PROJECT (OWASP)

3.1 Organizace OWASP

Open Web Application Security Project byl založen v roce 2001. V roce 2004 vznikla nadace OWASP (OWASP Foundation), jež zastřešuje práci na tomto projektu. Organizace samotná bylo založena jako mezinárodní nezisková organizace s více než 200 pobočkami po celém světě. Cílem nadace je rozšiřovat povědomí o bezpečnosti webových aplikací a pomáhat organizacím zvyšovat bezpečnost provozovaných webových aplikací. Veškeré materiály publikované pod hlavičkou OWASP Foundation jsou volně dostupné na internetu. Jednotlivé aktivity pod hlavičkou nadace jsou děleny do takzvaných projektů. Mezi nejznámější (tzv. vlajkové) projekty patří:

- Zed Attack Proxy
- Web Testing Environment Project
- OWTF
- Dependency Check
- Security Shepherd
- ModSecurity Core Rule Set Project
- CSRFGuard Project
- AppSensor Project
- Application Security Verification Standard Project
- Software Assurance Maturity Model (SAMM)
- AppSensor Project
- Top Ten Project
- Testing Project

Úplný seznam je možné nalézt na adrese `https://www.owasp.org/index.php/Category:OWASP_Project#tab=Project_Inventory`.

3.2 OWASP TOP 10 (2013)

Jedná se o vlajkový projekt pod hlavičkou OWASP Foundation. Projekt byl vytvořen v roce 2003 a v současné době ho vede Dave Wichers. Cílem projektu je zdokumentovat 10 nejzávažnějších a nejčastějších bezpečnostních chyb ve webových aplikacích. Tento seznam se aktualizuje jednou za 3–4 roky a je vytvořen na základě dat od sedmi firem, které se zabývají zabezpečením webových aplikací. Jednotlivé položky v seznamu jsou prioritizovány na základě počtu výskytů u poskytovatelů

v kombinaci s analýzou zneužitelnosti, detekovatelnosti a dopadu jednotlivých zranitelností. Poslední vydání z roku 2013 obsahuje následující zranitelnosti:

- **A1 – Injektování** – K této zranitelnosti dochází vždy, když se v rámci příkazu či dotazu posílají přímo do interpretu nedůvěryhodná data. Data odeslaná útočníkem poté mohou na serveru způsobit nežádoucí chování (např. chybnou autentizaci, umožnění přístupu k citlivým datům atd.). Nejčastějším typem útoku je SQL injektování, v praxi ale je možné provést injektování jakéhokoli programu, který se interpretuje za chodu (operační systém, Lightweight Directory Access Protocol, HTML, JavaScript, atd.).
- **A2 – Chybná autentizace a správa relace** – Většina webových aplikací od uživatelů při používání vyžaduje přihlášení. Funkce sloužící k přihlášení ale často nejsou implementovány správně, a proto může dojít ke kompromitování webové aplikace. Útočník poté může získat přístup k datům, heslům či jiným citlivým údajům jednotlivých uživatelů.
- **A3 – Cross-Site Scripting (XSS)** – K chybám tohoto typu dochází v případě, že aplikace přijme nedůvěryhodná data, která následně odešle prohlížeči bez kontroly či escapování. Tento typ útoku umožňuje útočníkům spouštět skripty v prohlížeči uživatelů. To může vést například k odcizení relace či dat nebo přesměrování oběti na podvodné stránky.
- **A4 – Nezabezpečený přímý odkaz na objekt** – K této zranitelnosti dojde v případě, že vývojář vytvoří pevný odkaz na určitý soubor, který lze následně používat bez jakéhokoli řízení přístupu. V takovém případě hrozí nebezpečí neoprávněného přístupu k datům.
- **A5 – Nezabezpečená konfigurace** – V rámci spuštění webových aplikací je žádoucí mít odpovídající zabezpečení konfiguračních souborů a také změnit výchozí nastavení aplikací. Tato nastavení je nutné pravidelně aktualizovat a udržovat. Při nedodržení těchto pravidel může útočník získat přístup ke konfiguraci webové aplikace a k citlivým datům.
- **A6 – Expozice citlivých dat** – Mnoho aplikací nechrání odpovídajícím přístupem citlivá data (např. čísla kreditních karet, rodná čísla atd.). Tato data lze odcizit a zneužít.
- **A7 – Chyby v řízení úrovní přístupů** – Většina webových aplikací kontroluje, zda má přihlášený uživatel oprávnění pro přístup k funkci, během nahrávání uživatelského rozhraní. Webové aplikace by ale měly kontrolovat tento přístup i v případě, že uživatel danou funkci používá. Při nedodržení tohoto postupu mohou útočníci získat přístup k dané funkci bez řádného povolení.
- **A8 – Cross-Site Request Forgery (CSRF)** – Tento útok spočívá v tom, že útočník přes prohlížeč oběti odešle zranitelné webové aplikaci požadavek na určitý úkon včetně potřebných autentizačních informací. Zranitelná webová

aplikace poté tyto požadavky považuje za platné požadavky oběti.

- **A9 – Použití známých zranitelných komponent** – Veškeré komponenty na webovém serveru (např. web server, databáze, frameworky, knihovny atd.) obvykle běží v privilegovaném režimu. Při použití starších a neaktualizovaných verzí může útočník využít známých zranitelností a získat kontrolu nad aplikací nebo serverem.
- **A10 – Neošetřené přesměrování a předávání** – Webové aplikace obvykle přesměrovávají uživatele na jiné webové stránky. V případě, že toto přesměrování není správně ošetřeno, může být oběť přesměrována na podvodnou stránku [11].

4 NÁVRH LABORATORNÍ ÚLOHY

4.1 Výchozí situace

Cílem této práce je navrhnout a realizovat laboratorní úlohu, jež studenty seznámí s problematikou aplikačních firewallů. Laboratorní úloha bude použita v rámci výuky na Fakultě elektrotechniky a komunikačních technologií VUT v Brně. Při návrhu úlohy bylo nutné vycházet z těchto požadavků:

- Úloha musí být realizována ve virtualizovaném prostředí VMware za použití softwarové platformy F5 BIG-IP.
- Od studentů se očekává základní znalost problematiky zabezpečení webových aplikací.
- V rámci úlohy se očekává, že cvičící bude spolupracovat se studenty na vypracování, proto by rozsah zadání pro studenty měl být na 1–2 strany A4.

Na základě požadavku bylo navrženo testovací pracoviště, jehož schéma je zobrazeno na obrázku 4.1. Úloha ukazuje typické zapojení aplikačního firewallu jako mostu mezi dvěma sítěmi. Jedna síť je považována za interní (síť, v níž jsou umístěny chráněné web servery) a druhá za externí (síť, do které mají koncoví uživatelé přístup). Prakticky se jedná o implementaci reverzní proxy, jež byla popsána výše.

V rámci laboratorní úlohy si studenti vyzkouší:

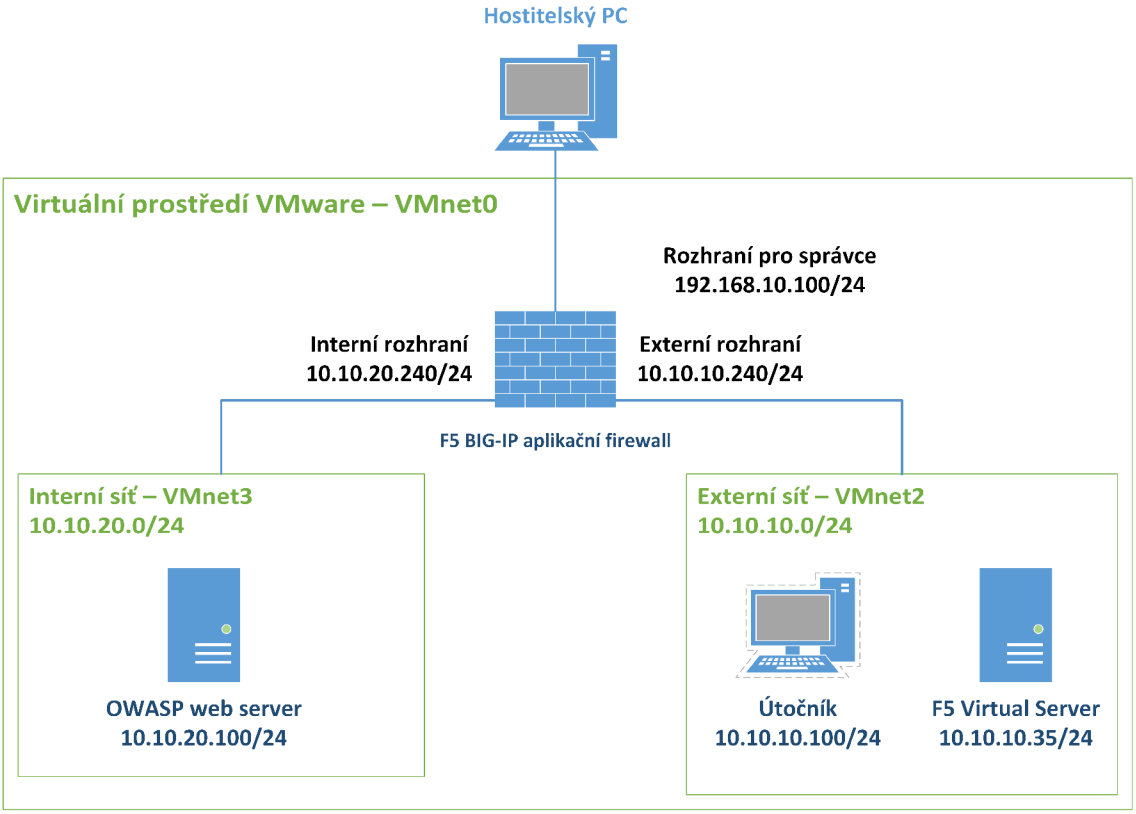
- počáteční konfiguraci aplikačního firewallu,
- vybrané útoky na zranitelné webové aplikace,
- konfigurace WAF,
- ověření funkčnosti aplikačního firewallu,
- procházení záznamů z aplikačního firewallu.

4.2 Přehled použitých nástrojů

4.2.1 F5 BIG-IP

Jedná se o modulární softwarovou platformu od společnosti F5. Řešení pracuje na proprietárním operačním systému Traffic Management Operating System (TMOS). Nad tímto operačním systémem jsou spuštěny jednotlivé moduly poskytující síťové služby. Základní řešení nabízí následující moduly:

- **BIG-IP Local Traffic Manager (LTM)** – Umožňuje inteligentně řídit provoz v sítích.
- **BIG-IP DNS** – Implementace DNS serveru od společnosti F5.
- **BIG-IP Access Policy Manager (APM)** – Umožňuje jednotnou autentizaci uživatelů v rámci interních sítí.



Obr. 4.1: Zapojení pracoviště

- **Secure Web Gateway Services** – Proxy server pro koncové uživatele.
- **BIG-IP Application Security Manager (ASM)** – Aplikační brána pro ochranu webových aplikací.
- **BIG-IP Advanced Firewall Manager (AFM)** – Jedná se o implementaci stavového firewallu bránící před útoky na 3.–4. vrstvě referenčního ISO/OSI modelu.
- **BIG-IP Application Acceleration Manager (AAM)** – Slouží k optimalizaci přístupu ke službám.
- **BIG-IP Link Controller** – Slouží jako inteligentní router.

Tento nástroj je možné používat na hardwaru přímo od výrobce, jako virtualizované řešení či jako cloudové řešení podle modelu Software jako služba (Software-as-a-Service neboli SaaS) [2].

Produkt byl zvolen jako dostupné a dobře dokumentované virtualizované řešení vhodné k použití v rámci výuky.

4.2.2 OWASP Broken Web Applications (BWA)

Jedná se o projekt spadající pod hlavičku OWASP Foundation. Cílem projektu je vytvořit kolekci zranitelných webových aplikací, které jsou publikovány jako součást jiných projektů. Tato kolekce se distribuuje jako virtuální stroj, které je možné volně šířit a používat pro výuku a testování zranitelnosti webových aplikací. V tomto virtuálním prostředí využíváme aplikaci Mutillidae 2, jež nabízí velké množství záměrně zranitelných webových aplikací a dává studentům možnost v praxi otestovat zranitelnosti zdokumentované v rámci seznamu OWASP Top 10. Řešení bylo zvoleno proto, že je přímo určeno pro demonstraci a testování síťových zranitelností, a je tedy pro navrhovanou laboratorní úlohu ideální.

4.2.3 VMware Workstation Player

Jedná se o virtualizační nástroj společnosti VMware, který je pro nekomerční použití dostupný zdarma. Software samotný využívá stejné jádro jako jeho komerční varianta VMware Workstation [12].

Produkt byl zvolen proto, že platforma F5 BIG-IP je pro toto prostředí dostupná jako předpřipravené řešení. Další důležitou vlastností je možnost vytváření virtuálních sítí a jejich konfigurace pomocí dodatečného nástroje.

4.2.4 Burp Suite

Burp Suite je nástroj napsaný v programovacím jazyce Java. Slouží k testování bezpečnosti webových aplikací. Nástroj umožňuje jak automatizované, tak manuální testování a obsahuje velké množství funkcí. Jádrem tohoto nástroje je proxy server, který umožňuje zachytávat, prozkoumávat a upravovat provoz mezi webovým serverem a uživatelskou stanicí. Nástroj je vyvíjen ve dvou variantách:

- Burp Suite Free – zdarma dostupná varianta aplikace
- Burp Suite Professional – placená varianta aplikace

Jak bylo uvedeno výše, aplikace je určena k testování webových zranitelností, a je tedy vhodná pro navrhovanou laboratorní úlohu. V rámci laboratorní úlohy nebudeme používat pokročilé vlastnosti tohoto nástroje, proto nám bude stačit jeho zdarma dostupná varianta.

4.2.5 Klientská stanice

Navržená laboratorní úloha není závislá na operačním systému klientské stanice. Jediným požadavkem na stanici je dostupná verze programovacího jazyka Java a webový prohlížeč. Pro PC simulující roli útočníka byl zvolen operační systém

Linux, a to konkrétně distribuce Xubuntu (z důvodu nízkých hardwarových nároků samotného systému).

5 LABORATORNÍ ÚLOHA

5.1 Zadání úlohy pro studenty

5.1.1 Cíle úlohy

- Seznámení se systémem F5 BIG-IP
- Realizace vybraných útoků na zranitelné webové aplikace
- Konfigurace webového aplikačního firewallu (WAF) v prostředí F5 BIG-IP
- Ověření funkčnosti WAF útokem na webovou aplikaci

5.1.2 Teoretický úvod

Termín firewall původně pochází ze stavebnictví, kde označoval protipožární zeď. Později se používal obecně pro jakoukoliv překážku, která měla za úkol zastavit šíření požáru (např. u automobilů, v letectví atd.). V současné době se firewall ve spojitosti s informačními technologiemi používá jako síťové zařízení či software sloužící ke kontrole přístupu k síti či sítím. Rozlišujeme následující typy firewallů:

1. Paketové brány
2. Stavové firewally
3. Okruhové proxy brány
4. Aplikační proxy brány

V rámci laboratorní úlohy se seznámíme s aplikační bránou. Pro demonstraci funkčnosti aplikačního firewallu využijeme virtuální prostředí VMware Workstation Player, kde budeme virtualizovat dva servery v oddělených sítích a jednu koncovou stanici simulující uživatele či útočníka.

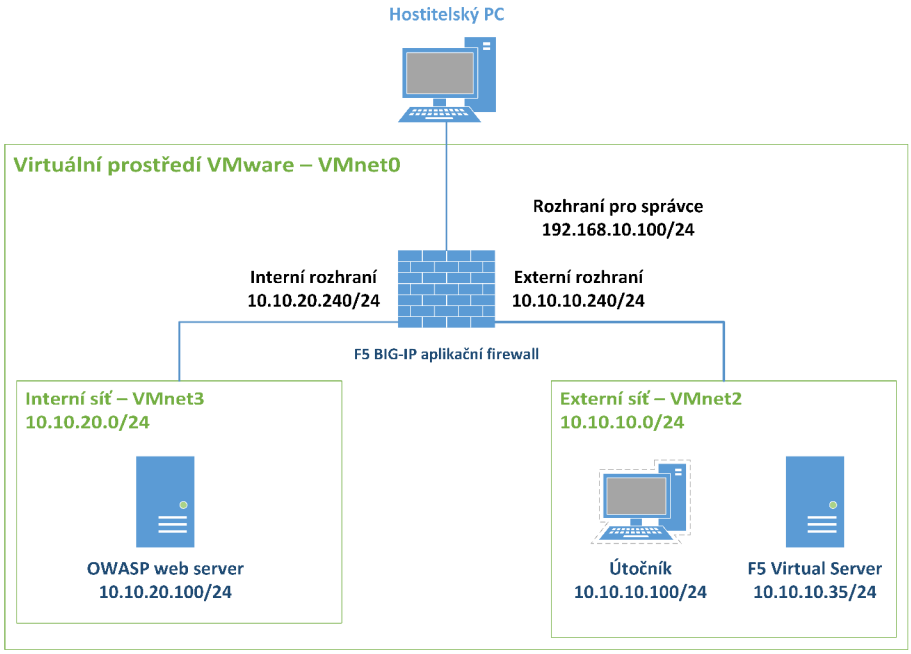
Jako server se zranitelnou webovou aplikací nám poslouží virtuální stroj vyvinutý v rámci projektu OWASP. Tento komunitní projekt se zabývá bezpečností webových aplikací jak z technologického, tak i uživatelského hlediska. Projekt mimo jiné vydává seznam 10 nejvíce zneužívaných chyb ve webových aplikacích. Tyto zranitelnosti si poté v praxi vyzkoušíme.

Jako aplikační firewall použijeme virtuální stroj s platformou F5 BIG-IP. Tato platforma poskytuje nepřeberné množství funkcí, jejichž popis by vystačil na samostatnou knihu.

Jako klientské/útočnické PC nám poslouží stanice s linuxovou distribucí Xubuntu.

5.1.3 Zapojení pracoviště

Zapojení pracoviště je zobrazeno na obrázku 5.1.



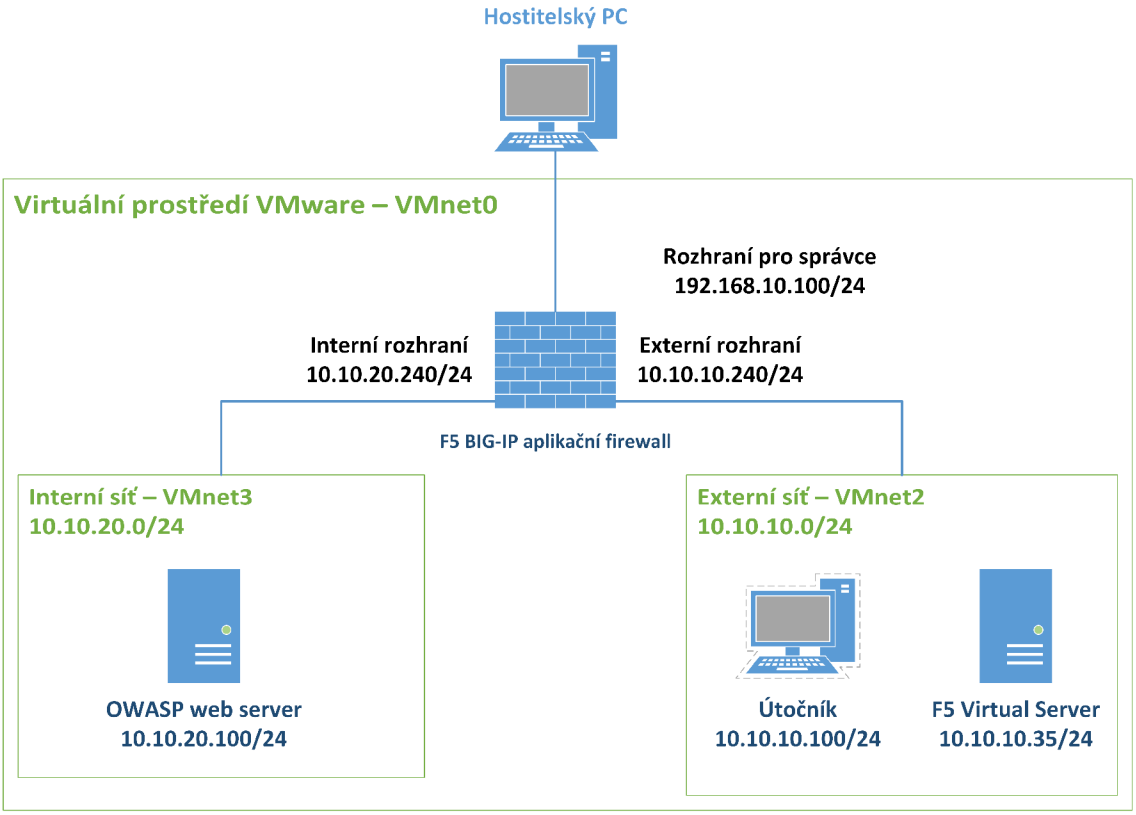
Obr. 5.1: Zapojení pracoviště

5.1.4 Otázky k úloze

1. Na které vrstvě v referenčním ISO/OSI modelu pracuje použitý aplikační firewall?
2. K čemu slouží virtuální server použitý v úloze?
3. Proč je v rámci konfigurace firewallu nutné vybrat správnou znakovou sadu?
4. Z jakého důvodu je OWASP server umístěn v izolované síti?
5. Bylo by možné pomocí použitých nástrojů umístit všechna zařízení do jedné podsítě? Pokud ano, jak by vypadalo schéma sítě?

5.2 Pracovní postup - informace pro cvičící

5.2.1 Nastavení pracoviště

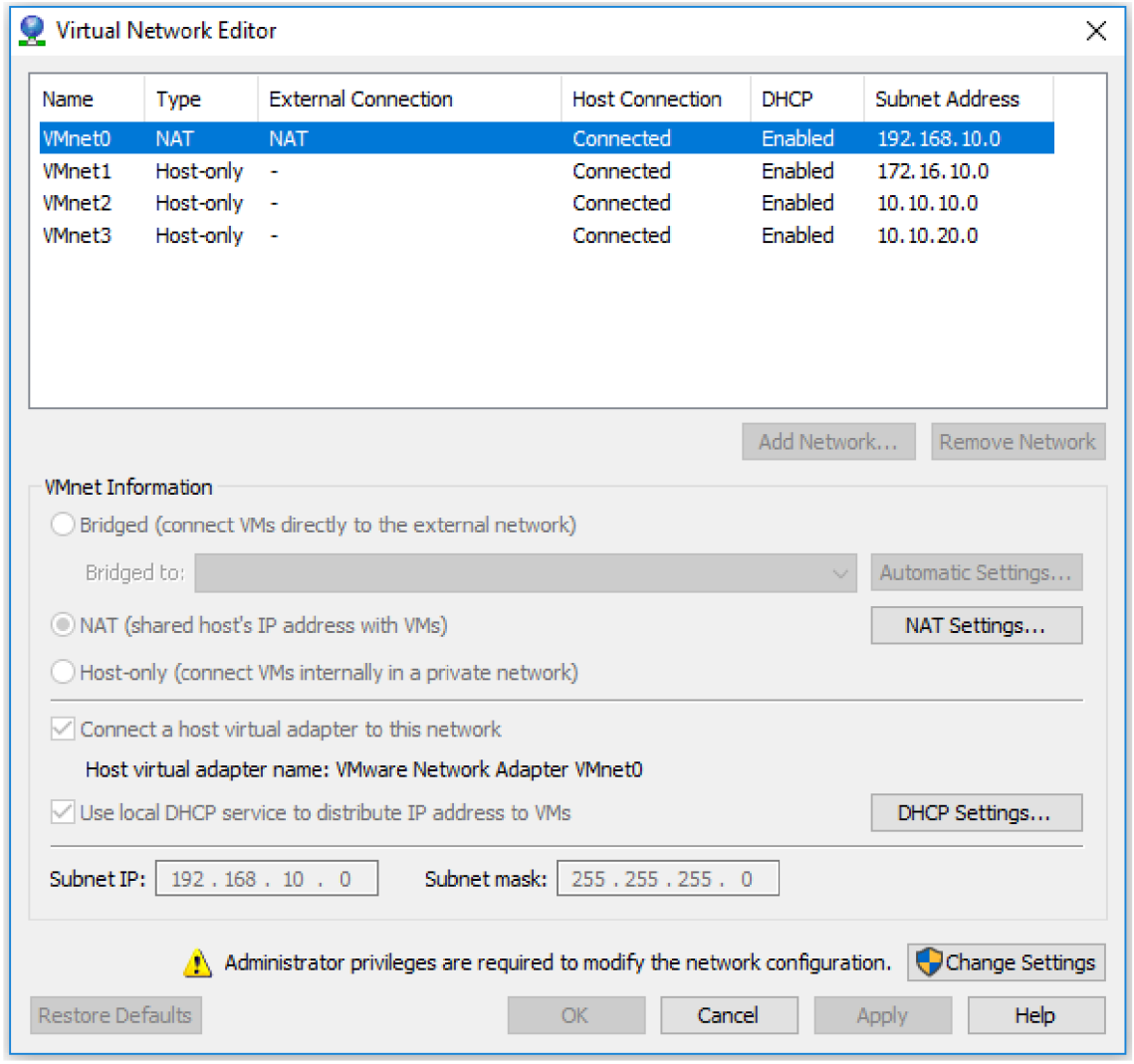


Obr. 5.2: Zapojení pracoviště

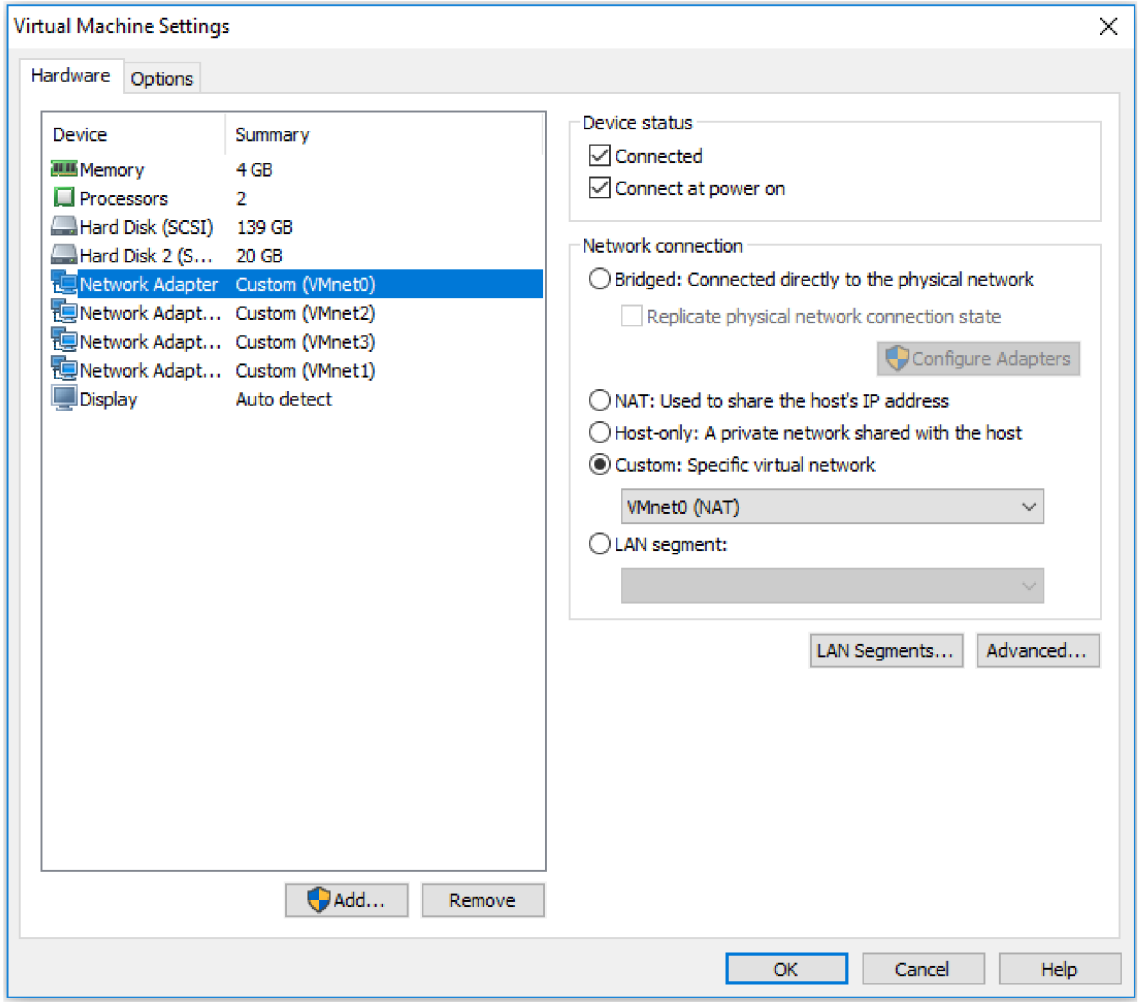
Na obrázku 5.1 je vidět zapojení pracoviště. Před konfigurací stanic je třeba nejdříve zkontrolovat nastavení síťových karet ve virtuálním prostředí VMware. K tomu využijeme nástroj *Virtual Network Editor*. Tento nástroj není standardní součástí nástroje VMware Player a je třeba ho ručně doinstalovat. Nástroj stačí zkopírovat do instalační složky nástroje VMware Player (obvykle *C:\Program Files(x86)\VMware\VMware Player*). Na USB disku přiloženém k bakalářské práci je tento nástroj uložen ve složce *\\VNE_archiv*. Po spuštění v nástroji upravíme jednotlivé virtuální sítě podle obrázku 5.3. DHCP rozsahy pro všechny prostředí nastavíme vždy 10.10.x.100 – 10.10.x.120 (toto nastavení je možné najít po kliknutí na položku *DHCP setting* v nástroji *Virtual Network Editor*). Po správném nastavení sítí importujeme do nástroje VMware Player jednotlivé virtuální stroje. Import provedeme v hlavním okně programu VMware Player kliknutím na položky *Player* > *File* > *Open*. Dalším krokem je přiřazení virtuálních sítí jednotlivým adaptérům v prostředí VMware Player. Na obrázku 5.4 jsou zobrazena nastavení jednot-

livých virtuálních strojů. Rozdělení virtuálních sítí pro jednotlivé stroje je vypsáno v tabulce 5.1. Následně můžeme spustit virtuální stroje. Na stanice se přihlásíme pomocí následujících přihlašovacích údajů:

1. Xubuntu
   Jméno: **Student**
   Heslo: **vutbrno**
2. F5 BIG-IP
   Jméno: **root**
   Heslo: **default**
3. OWASP BWA
   Jméno: **root**
   Heslo: **owaspbwa**



Obr. 5.3: Nastavení virtuálních sítí v nástroji Virtual Network Editor



Obr. 5.4: Přiřazení virtuálních sítí jednotlivým rozhraním

Tab. 5.1: Konfigurace síťových karet pro jednotlivé servery

F5 BIG-IP Image	OWASP	Xubuntu
Network adapter 1: VMnet0 Network adapter 2: VMnet2 Network adapter 3: VMnet3 Network adapter 4: VMnet1	Network Adapter: VMnet3	NetworkAdapter: VMnet2

Po přihlášení na stanice je nutné provést kontrolu pomocí terminálu a nástroje ifconfig. Zjistíme, zda zařízení obdržela IP adresy ze správných rozsahů podle adresy sítě v tabulce 5.1. V případě, že se tak nestalo, zkontrolujeme správnost nastavení. Konfigurace firewallu se skládá z těchto kroků:

1. Úvodní nastavení aplikačního firewallu
2. Vytvoření monitorovací služby
3. Vytvoření skupiny serverů
4. Vytvoření virtuálního serveru
5. Test funkčnosti virtuálního serveru bez aplikační brány
6. Konfigurace aplikačního firewallu
7. Test funkčnosti aplikační brány

Úvodní nastavení aplikačního firewallu

Po startu virtuálního serveru F5 BIG-IP zjistíme pomocí systémové konzole a nástroje ifconfig IP adresu firewallu. Zařízení by mělo obdržet adresu z DHCP serveru. Poté bude možné přímo z hostitelského operačního systému získat přístup k webovému rozhraní pomoci adresy https://192.168.10.xyz (kde xyz je adresa na rozhraní eth0, tuto adresu získáme pomocí příkazu **ifconfig | more**; protokol HTTPS je nutný) z hostitelského stroje. Pro přístup do webového rozhraní použijeme následující údaje:

- *Username*: **admin**
- *Password*: **admin**

Ve webovém rozhraní provedeme úvodní konfiguraci takto:

1. Na úvodní obrazovce klikneme na *Next*.
2. Následující krok obsahuje údaje o licenci – na stroji by měla být aktivní licence, a proto pokračujeme kliknutím na tlačítko *Next*.
3. Dalším krokem je aktivace potřebných modulů. Pro úlohu je třeba aktivovat možnosti *Local Traffic (LTM)* a *Application Security (ASM)*. U všech možností nastavíme ve sloupci *Provisioning* hodnotu *Nominal* nebo *Minimal* (pro potřeby cvičení hodnota nerozhoduje). Pokračujeme stisknutím tlačítka *Next*. Po změně je třeba restartovat moduly. Tuto akci potvrdíme a počkáme na dokončení operace.
4. V dalším kroku dojde k nastavení certifikátů na zařízení. Nastavení pouze potvrdíme kliknutím na tlačítko *Next*.
5. Následuje nastavení názvu zařízení a uživatelských účtů. V tomto kroku nastavíme název zařízení včetně domény (například **bigip1.local**), *Time Zone* na *Europe/Prague*. V sekci *User administration* nastavíme heslo pro uživatele root na **default** a údaje pro přihlášení Administratora do webového rozhraní

nastavíme na **admin**. Pokračujeme stisknutím tlačítka *Next*. V tomto kroku je také nutné se znovu přihlásit do zařízení.

6. V následujícím kroku nastavíme síťová rozhraní. Pokračujeme stisknutím tlačítka *Next*.
7. Prvním krokem je nastavení redundance zařízení. Tuto možnost nevyužíváme, proto zrušíme zaškrtnutí políčka *Config Sync* a stiskneme tlačítko *Next*.
8. V dalším kroku nastavíme interní síť. Použijeme IP adresu **10.10.20.240** a masku podsítě **255.255.255.0**. *Port Lockdown* ponecháme na výchozí hodnotě. V části *VLAN Configuration* zvolíme možnost *VLAN interface 1.2* a režim *Untagged*, následně stiskneme tlačítko *Add* a pokračujeme na další stránku stisknutím tlačítka *Next*.
9. Nastavíme externí síť. Použijeme IP adresu **10.10.10.240** a masku podsítě **255.255.255.0**. *Port Lockdown* ponecháme na výchozí hodnotě. Pro *VLAN configuration* potom zvolíme *VLAN interface 1.1* a režim *Untagged*, následně klikneme na tlačítko *Add* a na tlačítko *Finished*.
10. Konfiguraci následně ověříme příkazem *ping* v konzoli firewallu, jak na klientskou stanici, tak na virtuální stroj, kde běží webový server.

Vytvoření monitorovací služby

V hlavní nabídce zvolíme možnosti *Local Traffic > Monitors (+)* a vytvoříme monitor s následujícími parametry:

1. *Name:* **owasp_monitor**
2. *Type:* **HTTP**
3. *Send string: GET /index.php\r\n*
4. Ostatní nastavení ponecháme na výchozích hodnotách.

Vytvoření skupiny serverů

V hlavní nabídce vybereme možnost *Local Traffic > Pools > Pool List (+)* a vytvoříme novou skupinu serverů s následujícími vlastnostmi:

1. *Name*: **owasp_pool**
2. *Health monitor*: **owasp_monitor**
3. *Members*:
   Address: **10.10.20.100**
   Port: **80**
4. Ostatní nastavení ponecháme na výchozích hodnotách.

Vytvoření virtuálního serveru

V hlavní nabídce zvolíme možnost *Local Traffic > Virtual Servers > Virtual Server List (+)* a vytvoříme novou skupinu serverů s následujícími parametry:

1. *Name*: **owasp_vs**
2. *Destination Address*: **10.10.10.35**
3. *Service Port*: **443**
4. *HTTP Profile*: **HTTP**
5. *SSL Profile (client)*: **clientssl**
6. *Source Address Translation*: **Auto Map**
7. *Default Pool*: **owasp_pool**
8. Ostatní nastavení ponecháme na výchozí hodnotě a klikneme na tlačítko *Finished*.

Po nastavení virtuálního serveru je ještě nutné nastavit profil pro záznam událostí.

1. V nabídce *Virtual Servers > Virtual Server List* vybereme virtuální server **owasp_vs**.
2. V horní liště vybereme položku *Security > Policies*. V nabídce *Log Profile* vybereme položku *Log illegal requests* a přesuneme ji do pole *Selected*.
3. Volbu potvrdíme tlačítkem *Update*.

Burp Suite

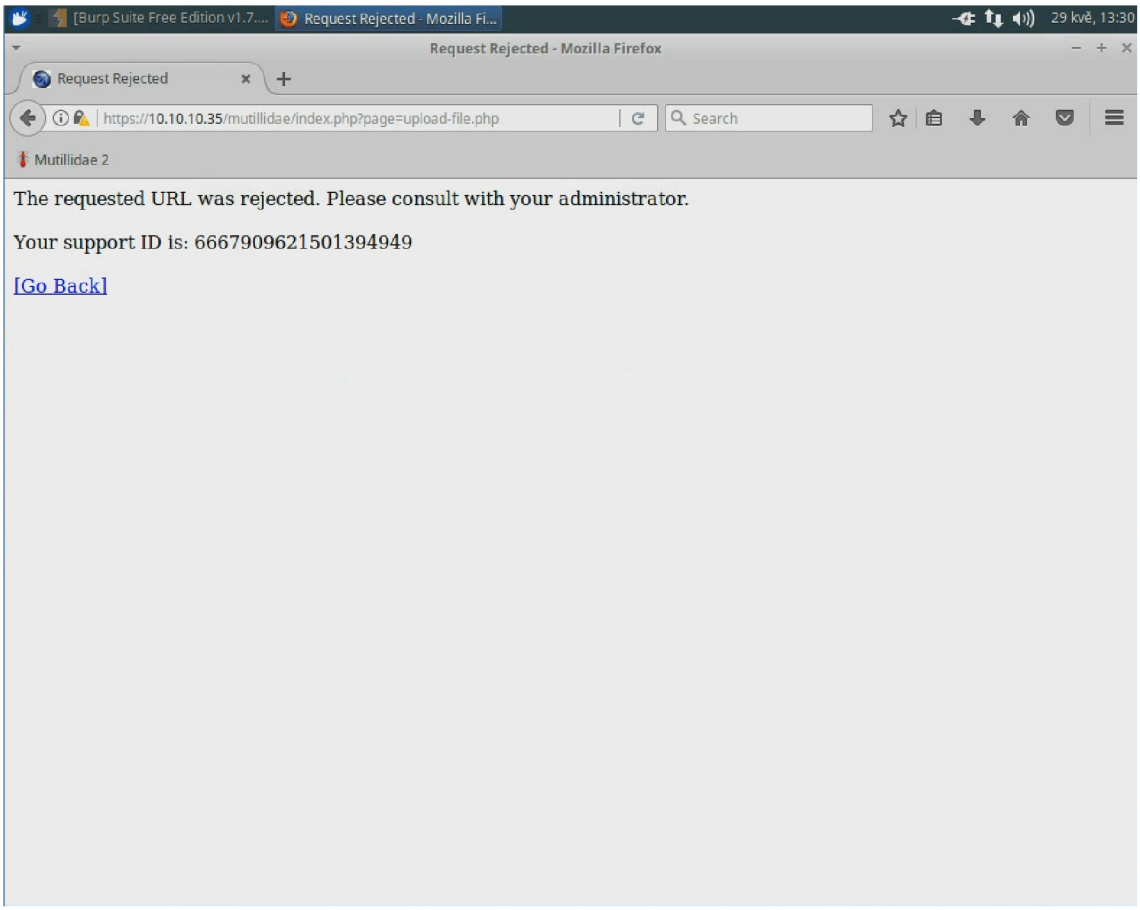
Pro testování zranitelností je také nutné spustit nástroj Burp Suite Free. Tento nástroj již je předinstalován v hostitelském operačním systému a stačí jej pouze spustit odkazem na ploše. Po spuštění se zobrazí průvodce:

1. Vybereme položku *Temporary project*, klikneme na tlačítko *Next*.
2. Vybereme položku *Use Burp defaults*, klikneme na tlačítko *Start Burp*.
3. Po spuštění přejdeme v horní nabídce na záložku *Proxy* a poté na podzáložku *Intercept*, kde přepneme položku *Intercept is on* na *Intercept is off* (tento nástroj nebudeme v tuto chvíli potřebovat).

Tímto máme pracoviště nastaveno a můžeme začít konfigurovat aplikační firewall.

5.2.2 Testování webových zranitelností

Testování zranitelností probíhá tak, že vždy nejdříve provedeme popsaný postup. Tím ověříme, že webová aplikace je daným útokem zranitelná. Následně nastavíme aplikační firewall a celý postup opakujeme. S aktivním aplikačním firewallem by se měla zobrazit chybová zpráva, kterou můžeme vidět na obrázku 5.5.



Obr. 5.5: Zobrazená stránka v případě útoku na chráněnou stránku

A1 – Injektování

První předvedenou zranitelností je zranitelnost A1 – Injektování. Test provedeme tak, že na klientském PC zadáme do webového prohlížeče IP adresu virtuálního serveru owasp_vs, tedy https://10.10.10.35/mutillidae/ a následně:

1. Na webové stránce vybereme možnost *OWASP 2013 > A1 – Injection (SQL) > SQLi Bypass Authentication > Login*.
2. Zobrazí se přihlašovací obrazovka, kde zadáme tyto údaje:
   Jméno: **admin**
   Heslo: „**' or 1=1 --** "(na konci je mezera)
3. Pokud je aplikační firewall vypnutý, aplikace by vás měla přihlásit.

Konfigurace WAF

V hlavní nabídce vybereme volbu *Security > Application Security > Security Policies (+)* a spustíme průvodce vytvořením nové bezpečnostní politiky.

V průvodci postupujeme takto:

1. *Select Local Traffic Deployment Scenario: Existing Virtual Server*. Klikneme na tlačítko *Next*.
2. Na následující stránce zvolíme tyto možnosti:
   Type of Protocol: *HTTPS*
   HTTPS virtual server: **owasp_vs**. Klikneme na tlačítko *Next*.
3. *Deployment Scenario: Create a security policy automatically (recommended)*. Zrušíme zaškrtnutí položek *Security Policy is case sensitive* a *Differentiate between HTTP and HTTPS URLs* a stiskneme tlačítko *Next*.
4. Zde použijeme následující hodnoty:
   Security Policy Name: **owasp_a1**
   Application Language: Unicode (utf-8)
   Klikneme na tlačítko *Next*.
5. Ze seznamu *Available Systems* přesuneme do seznamu *Assigned systems* položku *MySQL*.
6. Na následující stránce zvolíme tyto možnosti:
   Možnost *Policy type* nastavíme na *Comprehensive*.
   Policy Builder Learning Speed nastavíme na *Slow* a klikneme na tlačítko *Next*.
7. V dalším kroku zkontrolujeme nastavení jednotlivých položek a klikneme na tlačítko *Finish*.
8. Na následující stránce potvrdíme nastavení stisknutím tlačítka *Apply Policy*.

Útok by po aktivaci WAF měl být neúspěšný, což můžeme ověřit v sekci *Security > Event Logs > Application > Requests*.

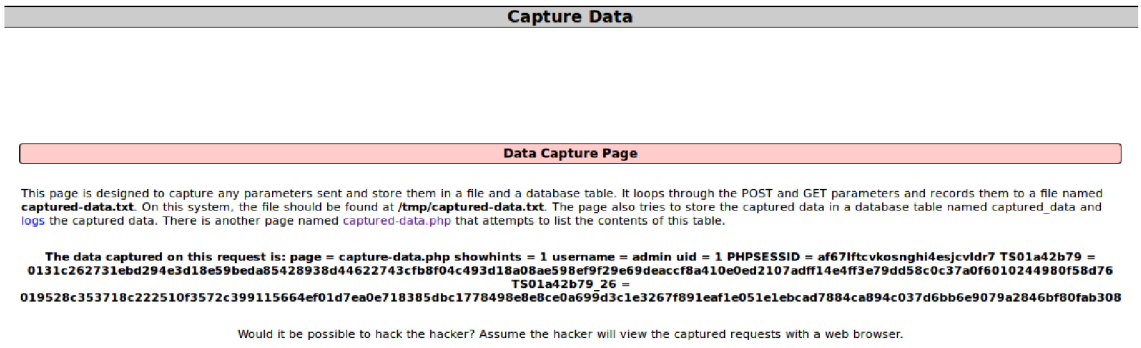
Aplikační firewall poté můžeme aktivovat nebo deaktivovat v nabídce *Policies* u virtuálního serveru:

1. V nabídce *Virtual Servers* > *Virtual Server List* vybereme virtuální server **owasp_vs**.
2. V horní liště vybereme položku *Security* > *Policies*. V nabídce *Application Security Policy* vybereme podle potřeby *Enable* či *Disable*.
3. Volbu potvrdíme tlačítkem *Update*.

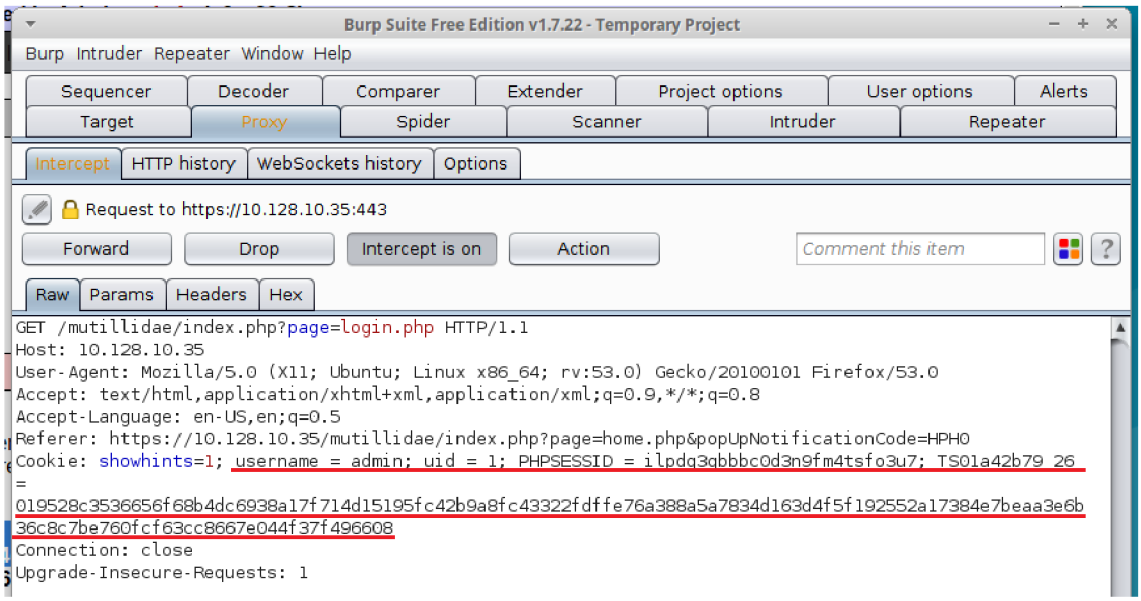
A2 – Chybná autentizace a správa relace

Druhou zranitelností je odcizení relace za pomoci podvržení cookie souborů. K tomuto účelu budeme potřebovat prohlížeč Firefox (nakonfigurovaný tak, aby používal Burp Suite) a druhý prohlížeč (v našem případě Chromium).

1. V prvním kroku se přihlásíme do webové aplikace v prohlížeči Chromium. Jako přihlašovací údaje použijeme:
   Username: **admin**
   Password: **admin**
2. Následně se přihlásíme v prohlížeči Firefox pomocí těchto přihlašovacích údajů::
   Username: **jeremy**
   Password: **password**
3. V pravé nabídce prohlížeče Chromium vybereme možnosti *Other* > *Data Capture Pages* > *Data Capture*. Na stránce se zobrazí informace o současné relaci. Formát dat je vidět na obrázku 5.6. Takto získaná data nyní využijeme pro odcizení relace.
4. V nástroji Burp Suite aktivujeme *Proxy* > *Intercept* > *Intercept is on* a následně vložíme data z prohlížeče Chromium do sekce *Cookie*. V tomto kroku je třeba jednotlivá data ještě oddělit středníkem (viz obrázek 5.7).
5. Po stisknutí tlačítka *Forward* nás webový server přihlásí jako administrátora.



Obr. 5.6: Stránka obsahující informace o relaci



Obr. 5.7: Okno programu Burp Suite s upravenými daty o relaci

Konfigurace WAF

V hlavní nabídce vybereme volbu *Security* > *Application Security* > *Security Policies (+)* a spustíme průvodce vytvořením nové bezpečnostní politiky.

V průvodci postupujeme takto:

1. *Select Local Traffic Deployment Scenario: Existing Virtual server*. Klikneme na tlačítko *Next*.
2. Na následující stránce zvolíme tyto možnosti:

   Type of Protocol: *HTTPS*

   HTTPS virtual server: **owasp_vs**. Klikneme na tlačítko *Next*.
3. *Deployment Scenario: Create a security policy automatically (recommended)*. Zrušíme zaškrtnutí položek *Security Policy is case sensitive* a *Differentiate between HTTP and HTTPS URLs* a stiskneme tlačítko *Next*.
4. Zde použijeme následující hodnoty:

   Security Policy Name: **owasp_a2**

   Application Language: Unicode (utf-8)

   Klikneme na tlačítko *Next*.
5. V tomto kroku není třeba přidávat další položky do seznamu, tudíž pokračujeme tlačítkem *Next*.
6. Na následující stránce zvolíme tyto možnosti:

   Možnost *Policy type* nastavíme na *Comprehensive*.

   Policy Builder Learning Speed nastavíme na *Slow* a klikneme na tlačítko *Next*.
7. V dalším kroku zkontrolujeme nastavení jednotlivých položek a klikneme na

tlačítko *Finish*.

8. Na následující stránce potvrdíme nastavení stisknutím tlačítka *Apply Policy*.

Útok by po aktivaci WAF měl být neúspěšný, což můžeme ověřit v sekci *Security > Event Logs > Application > Requests*.

A3 – Cross-Site Scripting (XSS)

1. Na webové stránce vybereme možnost *OWASP 2013 > A3 – Cross-Site Scripting (XSS) > Reflected (First Order) - DNS Lookup*.
2. Zobrazí se pole pro vložení Hostname/IP. Do tohoto pole vložíme řetězec **google.com && ls**
3. Stránka vypíše, že požadavek pro danou doménu vypršel. Následuje ale seznam souborů ve složce, což znamená, že útok byl úspěšný.

Konfigurace WAF

V hlavní nabídce vybereme volbu *Security > Application Security > Security Policies (+)* a spustíme průvodce vytvořením nové bezpečnostní politiky.

V průvodci postupujeme takto:

1. *Select Local Traffic Deployment Scenario: Existing Virtual server* Klikneme na tlačítko *Next*.
2. Na následující stránce zvolíme tyto možnosti:
   Type of Protocol: *HTTPS*
   HTTPS virtual server: **owasp_vs**. Klikneme na tlačítko *Next*.
3. *Deployment Scenario: Create a security policy automatically (recommended)*. Zrušíme zaškrtnutí položek *Security Policy is case sensitive* a *Differentiate between HTTP and HTTPS URLs* a stiskneme tlačítko *Next*.
4. Zde použijeme následující hodnoty:
   Security Policy Name: **owasp_a3**
   Application Language: Unicode (utf-8)
   Klikneme na tlačítko *Next*.
5. Ze seznamu *Available Systems* přesuneme do seznamu *Assigned systems* položky *Unix\Linux* a *PHP*.
6. Na následující stránce zvolíme tyto možnosti:
   Možnost *Policy type* nastavíme na *Comprehensive*.
   Policy Builder Learning Speed nastavíme na *Slow* a klikneme na tlačítko *Next*.
7. V dalším kroku zkontrolujeme nastavení jednotlivých položek a klikneme na tlačítko *Finish*.
8. Na následující stránce potvrdíme nastavení stisknutím tlačítka *Apply Policy*.

Útok by po aktivaci WAF měl být neúspěšný, což můžeme ověřit v sekci *Security > Event Logs > Application > Requests*.

A5 – Nezabezpečená konfigurace

1. Na webové stránce vybereme možnost *OWASP 2013 > A5 - Security Misconfiguration > Unrestricted File Upload*.
2. Zobrazí se pole, kde můžeme vybrat soubor pro nahrání na webový server. V nabídce vybereme soubor *A5_ukazka*, který se nachází na ploše, a stiskneme tlačítko *Upload File*.
3. V následujícím okně vidíme cestu, kam byl soubor uložen. Tento soubor můžeme rovnou zneužít, a to tak, že do prohlížeče vložíme zobrazenou absolutní cestu jako parametr pro soubor index.php (ukázka v poli níže). Zobrazí se stránka, kam můžeme vkládat příkazy, jež jsou skriptem předány přímo operačnímu systému.

```
https://10.10.10.35/mutillidae/index.php?page=/tmp/A5_ukazka
```

Konfigurace WAF

V hlavní nabídce vybereme volbu *Security > Application Security > Security Policies (+)* a spustíme průvodce vytvořením nové bezpečnostní politiky.

V průvodci postupujeme takto:

1. *Select Local Traffic Deployment Scenario: Existing Virtual server*. Klikneme na tlačítko *Next*.
2. Na následující stránce zvolíme tyto možnosti:
   Type of Protocol: *HTTPS*
   HTTPS virtual server: **owasp_vs**. Klikneme na tlačítko *Next*.
3. *Deployment Scenario: Create a security policy automatically (recommended)*. Zrušíme zaškrtnutí položek *Security Policy is case sensitive* a *Differentiate between HTTP and HTTPS URLs* a stiskneme tlačítko *Next*.
4. Zde použijeme následující hodnoty:
   Security Policy Name: **owasp_a5**
   Application Language: Unicode (utf-8)
   Klikneme na tlačítko *Next*.
5. Ze seznamu *Available Systems* přesuneme do seznamu *Assigned systems* položky *Unix\Linux* a *PHP*.
6. Na následující stránce zvolíme tyto možnosti:
   Možnost *Policy type* nastavíme na *Comprehensive*.
   Policy Builder Learning Speed nastavíme na *Slow* a klikneme na tlačítko *Next*.

7. V dalším kroku zkontrolujeme nastavení jednotlivých položek a klikneme na tlačítko *Finish*.
8. Na následující stránce potvrdíme nastavení stisknutím tlačítka *Apply Policy*.

Útok by po aktivaci WAF měl být neúspěšný, což můžeme ověřit v sekci *Security > Event Logs > Application > Requests*.

A8 – Cross-Site Request Forgery (CSRF)

1. Na webové stránce vybereme možnost *OWASP 2013 > A8 – Cross-Site Request Forgery (CSRF) > Add to your blog*.
2. Do pole na stránce vložíme obsah souboru *CSRF Ukazka*, který se nachází na ploše, a stiskneme tlačítko *Save Blog Entry*.
3. V seznamu na stránce se objeví nový záznam. Když na nový záznam umístíme kurzor, začnou se vkládat nové záznamy, což znamená, že útok byl úspěšný.

Konfigurace WAF

V hlavní nabídce Vybereme volbu *Security > Application Security > Security Policies (+)* a spustíme průvodce vytvořením nové bezpečnostní politiky.

V průvodci postupujeme takto:

1. *Select Local Traffic Deployment Scenario: Existing Virtual server*. Klikneme na tlačítko *Next*.
2. Na následující stránce zvolíme tyto možnosti:
   Type of Protocol: *HTTPS*
   HTTPS virtual server: **owasp_vs**. Klikneme na tlačítko *Next*.
3. *Deployment Scenario: Create a security policy automatically (recommended)*. Zrušíme zaškrtnutí položek *Security Policy is case sensitive* a *Differentiate between HTTP and HTTPS URLs* a stiskneme tlačítko *Next*.
4. Zde použijeme následující hodnoty:
   Security Policy Name: **owasp_a8**
   Application Language: Unicode (utf-8)
   Klikneme na tlačítko Next.
5. V tomto kroku není třeba přidávat další položky do seznamu, pokračujeme tlačítkem *Next*.
6. Na další stránce nastavíme tyto možnosti:
   Možnost *Policy type* nastavíme na *Comprehensive*.
   Policy Builder Learning Speed nastavíme na *Slow* a klikneme na tlačítko *Next*.
7. V dalším kroku zkontrolujeme nastavení jednotlivých položek a klikneme na tlačítko *Finish*.
8. Na následující stránce potvrdíme nastavení stisknutím tlačítka *Apply Policy*.

Útok by po aktivaci WAF měl být neúspěšný, což můžeme ověřit v sekci *Security > Event Logs > Application > Requests*.

5.2.3 Další vlastnosti aplikační brány

Ochrana před Denial of Service (DoS) útoky

Před nastavením samotné ochrany před útokem DoS je třeba v sekci *System > Resource Provisioning* deaktivovat modul *Application Security (ASM)* a aktivovat položku *Advanced Firewall (AFM)*. *Provisioning* může být nastaven na *Minimum* či *Nominal* (pro potřeby úlohy toto nehraje roli). Po restartování modulů postupujeme takto:

Nastavení log profilu:

1. V hlavní nabídce klikneme na položku *System > Logs > Configurations > Log Publisher (+)*.
2. Log Publisher pojmenujeme **log_for_dos**. V části *Destinations* přesuneme položku *local-db* do seznamu *Selected* a potvrdíme tlačítkem *Finished*.
3. Následně v sekci *Security > Event Logs > Logging Profiles (+)* vytvoříme nový profil, který pojmenujeme **dos_logging**. Na stránce vybereme položku *DoS Protection* a v sekci *Network DoS Protection* zvolíme námi vytvořený **log_for_dos**. Průvodce ukončíme tlačítkem *Finished*.
4. Volbu potvrdíme tlačítkem *Update*.
5. V nabídce *Virtual Servers > Virtual Server List* vybereme virtuální server **owasp_vs**.
6. V horní liště vybereme položku *Security > Policies*. V nabídce *Log Profile* přesuneme vytvořený profil do pole *Selected*.
7. Volbu potvrdíme tlačítkem *Update*.

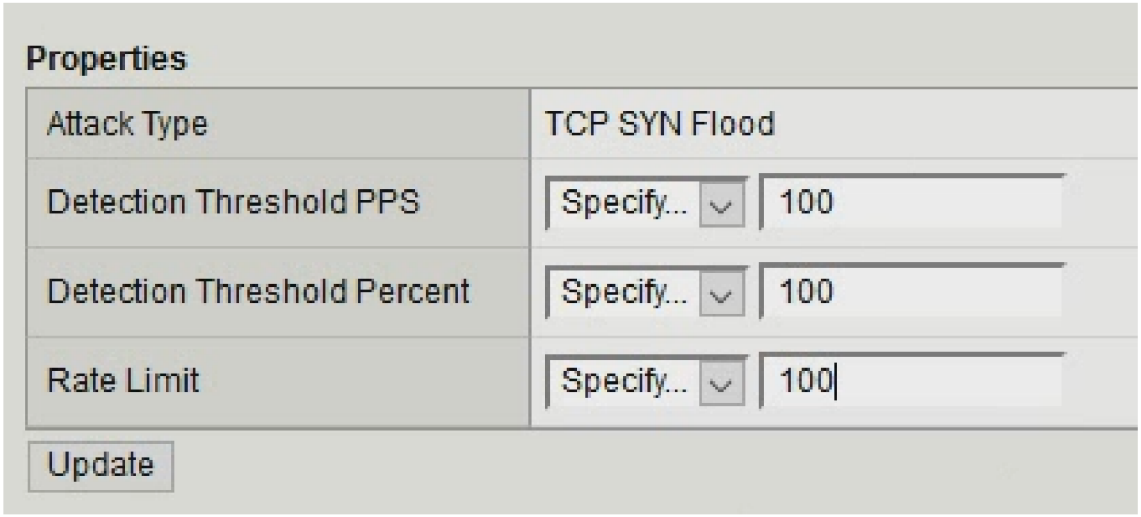
Následně je nutné nastavit limity pro útok typu DoS:

1. V hlavní nabídce vybereme položku *Security > DoS Protection > Device Configuration*.
2. V této sekci nastavíme jako *Log Publisher* vytvořený profil **log_for_dos**.
3. Nastavíme limity pro jednotlivé typy útoků. To provedeme v sekci *Flood*: upravíme zde nastavení položek *TCP Syn Flood* a *TCP Syn oversize*. Limity nastavíme podle obrázku 5.8.

Pro otestování funkčnosti spustíme útok nástrojem *hping3*. V konzoli klientské stanice spustíme následující příkaz:

```
sudo hping3 -c 10000 -d 120 -S -w 64 -p 443 –flood 10.10.10.35
```

Kontrolu funkce je možné provést v nabídce *Statistics > DoS Overview*, jde zde během několika minut zjistit, že začal útok na webový server. Kliknutím na probíhající



Obr. 5.8: Nastavení DoS profilu

útok lze zobrazit informace o tomto útoku a také reakci aplikační brány.

Load balancing

Při nastavení load balancingu je postup obdobný jako u vytváření skupiny serverů výše. Jediným rozdílem je nutnost přidat do seznamu více serverů, mezi které se bude provoz dělit. Také je nutné do prostředí importovat druhý virtuální server OWASP BWA podle postupu popsaného výše.

Vytvoření skupiny serverů

V hlavní nabídce vybereme možnost *Local Traffic* > *Pools* > *Pool List (+)* a vytvoříme novou skupinu serverů s následujícími vlastnostmi:

1. *Name*: **owasp_lb**
2. *Health monitor*: **owasp_monitor**
3. *Members*:
   Name: **10.10.20.100**
   Address: **10.10.20.100**
   Port: **80**
   Name: **10.10.20.101**
   Address: **10.100.20.101**
   Port: **80**
4. Ostatní nastavení ponecháme na výchozích hodnotách.

Vytvoření virtuálního serveru

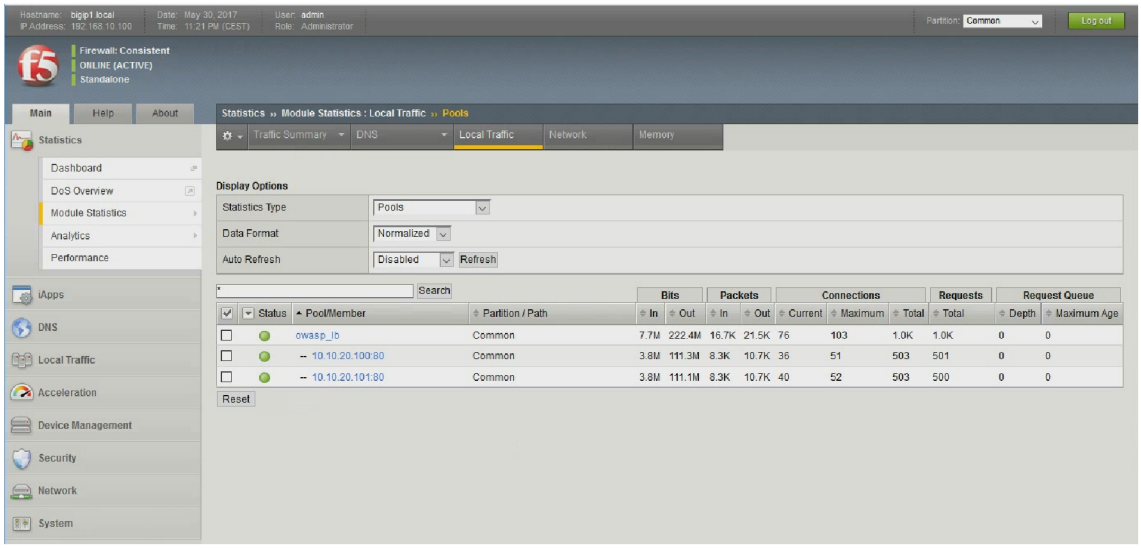
V hlavní nabídce zvolíme možnost *Local Traffic* > *Virtual Servers* > *Virtual Server List (+)* a vytvoříme novou skupinu serverů s následujícími parametry:

1. *Name*: owasp_lb
2. *Destination Address*: **10.10.10.35**
3. *Service Port*: **443**
4. *HTTP Profile*: **HTTP**
5. *SSL Profile (client)*: **clientssl**
6. *Source Address Translation*: **Auto Map**
7. *Default Pool*: **owasp_pool**
8. Ostatní nastavení ponecháme na výchozí hodnotě a klikneme na tlačítko *Finished*.

Pro otestování funkčnosti vytvoříme síťový provoz pomocí nástroje Apache Bench. V konzoli klientské stanice spustíme následující příkaz:

```
ab -n 10000 -c 100 https://10.10.10.35/
```

Funkčnost lze ověřit v nabídce *Local Traffic* > *Pools* > *Statistics* (viz obrázek 5.9). Zde je vidět, že zařízení rovnoměrně dělí provoz mezi virtuální servery.



Obr. 5.9: Kontrola funkčnosti load balancingu

5.2.4 Obnovení platformy BIG-IP do továrního nastavení

Firewall obnovíme do továrního nastavení takto:

1. V příkazovém řádku virtuálního stroje aktivujeme nástroj TMSH
   tmsh

2. Nahrajeme tovární nastavení pomocí příkazu
   load /sys config default
3. Operaci potvrdíme klávesou **y**
4. Uložíme změny následujícím příkazem
   save /sys config partitions all
5. Ukončíme nástroj tmsh
   quit
6. Restartujeme virtuální zařízení následujícím příkazem
   full_box_reboot

6 ZÁVĚR

V rámci bakalářské práce byla navržena laboratorní úloha, jež studenty seznámí s vlastnostmi webového aplikačního firewallu. Pro potřeby úlohy byla navržena jednoduchá síť ve virtuálním prostředí, jež studentům umožňuje se seznámit se se základní konfigurací webového aplikačního firewallu. V rámci úlohy si také studenti otestují následujícími typy útoků na webové aplikace:

- SQL injektování
- Odcizení relace
- Cross-Site Scripting (XSS)
- Zabránění uploadu nebezpečných souborů
- Cross-Site Request Forgery (CSRF)

Vybrané útoky byly vybrány ze seznamu deseti nejčastějších a nejvíce zneužívaných zranitelností, který vydává organizace OWASP. Prezentované zranitelnosti byly zvoleny pro svoji názornost.

Kromě testování webových zranitelností mají studenti možnost vyzkoušet si ochranu proti útoku typu DoS, jež bývá běžně používána jako součást webových aplikačních firewallů. Také je ukázána konfigurace a využití load balancingu webových aplikací, což může být pro studenty přínosné v budoucím zaměstnání.

Součástí úlohy je zadání pro studenty a návod pro cvičící. Úloha byla navržena jako interaktivní a od cvičícího se očekává, že spolu se studenty provede konfiguraci firewallu. Z tohoto důvodu je zadání pro studenty krátké, zatímco návod pro cvičící obsahuje detailní popis nastavení. V rámci úlohy byly také zpracovány video návody pro cvičícího. Tato videa lze nalézt na přiloženém flash disku.

LITERATURA

[1] ALAM, M. Afshar, Tamanna SIDDIQUI a K. R. SEEJA. *Recent Developments in Computing and its Applications*. Nové Dillí: IK International Publishing House, 2009. ISBN 978-9380026787.

[2] BIG-IP Platform. *F5 Networks* [online]. USA: F5 Networks, 2016 [cit. 2016-12-08]. Dostupné z: `https://f5.com/products/big-ip`

[3] CANAVAN, John E. *Fundamentals of network security*. Boston: Artech House, c2001. ISBN 15-805-3176-8.

[4] Gartner Says 6.4 Billion Connected "Things"Will Be in Use in 2016, Up 30 Percent From 2015. In: *Gartner.com* [online]. STAMFORD: Gartner, 2015 [cit. 2016-11-17]. Dostupné z: `http://www.gartner.com/newsroom/id/3165317`

[5] INGHAM, Kenneth a Stephanie FORREST. *A History and Survey of Network Firewalls* [online]. Albuquerque, 2002 [cit. 2016-11-18]. Dostupné z: `http://www.cs.unm.edu/~treport/tr/02-12/firewall.pdf. TechnicalReport.TheUniversityofNewMexico.`

[6] Overview of Windows Firewall with Advanced Security. In: *Microsoft* [online]. USA: Microsoft TechNet, 2009 [cit. 2016-12-07]. Dostupné z: `https://technet.microsoft.com/cs-cz/library/dd448535(v=ws.10).aspx`

[7] OWASP top 10. *OWASP top 10* [online]. Open Web Application Security Project, 2013, **2013**(1), 1-22 [cit. 2016-12-08]. Dostupné z: `https://www.owasp.org/images/f/f3/OWASP_Top_10_-_2013_Final_-_Czech_V1.1.pdf`

[8] PUBAL, Jason, FILKINS, Barbara, ed. Web Application Firewalls: Enterprise techniques. In: *SANS Institute* [online]. USA: SANS Institute, 2015, s. 29 [cit. 2017-05-22]. Dostupné z: https://www.sans.org/reading-room/whitepapers/application/web-application-firewalls-35817

[9] SMITH, Richard E. *Internet cryptography*. Reading, Mass.: Addison-Wesley, c1997. ISBN 02-019-2480-3.

[10] STALLINGS, William. *Cryptography and network security: principles and practice*. 5th ed. Boston: Prentice Hall, c2011. ISBN 01-360-9704-9.

[11] THE OWASP FOUNDATION. *OWASP Top 10 - 2013: Deset nejkritičtějších rizik webových aplikací*. The OWASP Foundation, online, 2013. Dostupné také z: https://www.owasp.org/images/d/d4/OWASP_Top_10_-_2013_-_Czech_V1.0.pdf

[12] VMware Workstation Player. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2016-12-08]. Dostupné z: `https://en.wikipedia.org/wiki/VMware_Workstation_Player`

[13] Web application firewall. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-05-22]. Dostupné z: https://en.wikipedia.org/wiki/Web_application_firewall

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

BWA Broken Web Applications

CSRF Cross-Site Request Forgery

DoS Denial of Service

HTTP Hypertext Transfer Protocol

HTTPS Hypertext Transfer Protocol Secure

HW Hardware

IP Internet Protocol

ISO International Organization for Standardization

NAT Network Address Translation

OSI Open Systems Interconnection

OWASP Open Web Application Security Project

SAMM Software Assurance Maturity Model

SOCKS Socket Secure

SOHO Small Office, Home Office

SW Software

TCP Transmission Control Protocol

TMOS Traffic Management Operating System

UDP User Datagram Protocol

VPN Virtual Private Network

WAF Web Application Firewall

XSS Cross-Site Scripting

OBSAH PŘILOŽENÉHO FLASH DISKU

/
- `BP` ..... Složka s elektronickou verzí práce
  - `BP_Ivo_Prochazka.pdf` ..... Elektronická verze práce
- `obrazy` ..... Složka s obrazy jednotlivých virtuálních strojů
  - `F5_image.zip` ..... Obraz platformy F5
  - `OWASP_BWA_VM_1.2.zip` ..... Obraz webového serveru OWASP
  - `Xubuntu.zip` ..... Klientská stanice
- `videonavody` ..... Složka s videonávody
  - `01-nastaveni_pracoviste.mp4` ..... Nastavení pracoviště
  - `02-nastaveni_F5.mp4` ..... Základní konfiguraci platformy F5
  - `03-nast_vs.mp4` ..... Nastavení virtuálního serveru
  - `04-injektovani.mp4` ..... Ukázka SQL Injektování
  - `05-autentizace.mp4` ..... Ukázka odcizení relace
  - `06-xss.mp4` ..... Ukázka XSS
  - `07-security_misc.mp4` ..... Ukázka nezabezpečené konfigurace
  - `08-csrf.mp4` ..... Ukázka CSRF
  - `09-dos.mp4` ..... Ukázka ochrany před DoS útokem
  - `10-lb.mp4` ..... Ukázka load balancingu
- `VNE_archiv` ..... Složka s nástrojem Virtual Network Editor
  - `VMware-player-12.5.6-5528349.zip` Instalační soubor VMware Playeru
  - `vmnetcfg.zip` ..... Archiv s nástrojem Virtual Network Editor