Analýza možností zabezpečeného přístupu do informačního systému

Bc. Jan Kopecký

Diplomová práce
2019

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2018/2019

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Jan Kopecký**
Osobní číslo:	**A17326**
Studijní program:	**N3902 Inženýrská informatika**
Studijní obor:	**Bezpečnostní technologie, systémy a management**
Forma studia:	**kombinovaná**
Téma práce:	**Analýza možností zabezpečeného přístupu do IS**
Téma anglicky:	**An Analysis of the Possibilities of Secure Access to IS**

Zásady pro vypracování:

1. **Formou rešerše objasněte problematiku přístupu do webového informačního systému.**
2. **Zaměřte se na otázky přihlášení, autentizace, autorizace uživatele do IS a možné způsoby zabepečení.**
3. **Na modelovém IS ukažte zabezpečení z pohledu přístupu uživatelů.**
4. **Zhodnoťte jednotlivé možnosti a následně navrhněte bezpečný přihlašovací mechanismus.**

Rozsah diplomové práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. **AJAX a PHP : tvoříme interaktivní webové aplikace profesionálně /. Brno : Zoner Press, 2006. 320 s.**
2. **LACKO, Lubomir. SQL (hotová řešení pro SQL Server, Oracle a MySQL). Computer Press, 2003, ISBN: 80-7226-975-5.**
3. **STEJSKAL, J. Vytváříme WWW stránky pomocí HTML, CSS a JavaScriptu. Computer Press, 2006, ISBN: 80-251-0167-3.**
4. **WELLING, Luke, THOMSON, Laura. MySQL - Průvodce základy databázového systému. Computer Press, 2005, ISBN: 80-251-0671-3.**
5. **ŠPAČEK, David a Jiří ŠPALEK. Informační systémy ve veřejném sektoru: distanční studijní opora. Vyd. 1. Brno: Masarykova univerzita v Brně, Ekonomicko-správní fakulta, 2004, 120.**
6. **RÁBOVÁ, Ivana. Podnikové informační systémy a technologie jejich vývoje. V Tribun EU vyd. 1. Brno: Tribun EU, 2008, 139 s. ISBN 978-80-7399-599-7.**
7. **SODOMKA, Petr. Informační systémy v podnikové praxi. Vyd. 1. Brno: Computer Press, 2006, 351 s. ISBN 80-251-1200-4.**

Vedoucí diplomové práce: **doc. Ing. Jiří Gajdošík, CSc.**
Ústav bezpečnostního inženýrství

Datum zadání diplomové práce: **30. listopadu 2018**

Termín odevzdání diplomové práce: **17. května 2019**

Ve Zlíně dne 14. prosince 2018

L.S.

doc. Mgr. Milan Adámek, Ph.D.
děkan

doc. RNDr. Vojtěch Křesálek, CSc.
ředitel ústavu

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové 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 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 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 práci – nebo poskytnout licenci k jejímu využití jen, připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše), bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové 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 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, dne …….

podpis diplomanta

ABSTRAKT

Práce se zabývá problematikou zabezpečení přístupu do webového informačního systému. Jde o důležitou součást provozování informačních systémů, kdy je třeba hlídat, kdo do systému přistupuje a jaká má oprávnění. Proto se práce zabývá především autentizací a autorizací. V praktické části je zhodnocen stav dvou redakčních systémů a k dosažení požadovaného zabezpečení přístupu jsou využity znalosti uvedené v teoretické části práce.

Klíčová slova: Informační systém, kybernetická bezpečnost, autentizace, autorizace, SSO, OAuth, identity management

ABSTRACT

The thesis deals with the issue of secured access to the web information system. Secured access to information system is very important nowadays. It is necessary to know who is accessing the system and if he has right permissions. The thesis deals mainly with authentication and authorization. In the practical part there are two content management systems used to present their access security. The findings from the theoretical part were used in the practical one to achieve the secured access demanded.

Keywords: Information system, cyber security, authentication, authorization, SSO, OAuth, identity management

Rád bych poděkoval vedoucímu mé diplomové práce panu docentovi Jiřímu Gajdošíkovi za cenný čas, potřebné rady, mnohá doporučení a důvěru, kterou mi poskytl při tvorbě diplomové práce. Velký dík také patří rodině, která mě po celou dobu studia podporovala.

Motto mé práce zní: „Slabé zabezpečení je lepší nežli žádné.“

Prohlašuji, že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD

Tato práce vznikla za účelem prozkoumání současných bezpečnostních technologií, principů a standardů používaných pro přístupu uživatele do nějakého informačního systému.

Hlavním cílem práce bylo vyhodnotit bezpečnost přihlašování uživatelů do informačního systému a navrhnout možná zlepšení. Abych toho mohl docílit, bylo třeba podívat se na celou problematiku zabezpečeného přístupu a informačního systému více detailně.

Práce je dělena typicky na část teoretickou a praktickou. Teoretická část práce se sestává ze tří kapitol. V první kapitole vysvětluji, co je informační systém, jeho funkce a rozdělení. Dále se věnuji především redakčním systémům (CMS) a uvádím některé zástupce. Druhá kapitola je věnována kybernetické bezpečnosti. Od definice, norem a hrozeb přecházím k hlavním tématům a to autentizace a autorizace. A konečně pak k samotným autentizačním standardům a principům ověřování identity uživatele. Ve třetí kapitole se věnuji Identity a Access Managementu. Vysvětluji pojmy identita a přístup a v závěru zmiňuji problematiku bezpečného hesla.

Hlavním cílem práce je zabezpečený přístup do informačního systému. Dále jsem stanovil dílčí cíle:

- zhodnotit aktuální stav zabezpečení jednotlivých systémů
- poukázat na možná rizika
- návrh eliminace jednotlivých rizik

Jejich naplnění je vyhrazena praktická část práce shodou okolností také rozdělena do tří kapitol. Ve čtvrté kapitole vybírám dva zástupce redakčních systému a konkrétně na nich znázorňuji problematiku autentizace a autorizace. U každého pak ukazuji možné slabiny. Pátá kapitola obsahuje možná řešení jednotlivých problému s ukázkami implementace nebo s návody, jak požadovaného zlepšení docílit. V poslední šesté kapitole rozebírám zabezpečení redakčního systému a dávám obecná doporučení pro bezpečný provoz CMS.

I. TEORETICKÁ ČÁST

1 INFORMAČNÍ SYSTÉM

V první kapitole zmíním několik definic a význam samotného informačního systému. Po obecném úvodu se zaměřím na systémy pro správu obsahu. Představím několik zástupců ze světa open source a několik příkladů placených systémů.

Definice informace

„Informace, latinsky zpráva, popřípadě její obsah, smysl. V kybernetice a vědách příbuzných poučení o něčem, sloužící k vzájemnému styku živých a neživých systémů přírody. Je zde chápána v obecném matematickém smyslu.“ [1]

„Informace je název pro obsah toho, co si vyměníme s vnějším světem, když se mu přizpůsobujeme a působíme na něj svým přizpůsobováním." [2]

Definice Informační systém (IS)

Systém sběru, uchování, analýzy a prezentace dat, který poskytuje informace různým uživatelům. Může, avšak nemusí být podporován počítačem. Jedná se o soubor technickoorganizačních opatření. Hlavním účelem je získání, uchování, zpracování a následné přenášení informací za účelem automatizace pracovních postupů. Hlavní cíl podnikového informačního systému je podpora rozhodování v organizaci a sjednocení postupů ekonomických činností. Důležitá je role lidí v systému, kteří na proces zpracování dohlížejí a vybírají relevantní informace pro další zpracování. Přílišná snaha o automatizaci a omezení lidského faktoru může vést k nezdaru projektu a k nepřehlednosti celého systému. [3]

1.1 Co informační systém nevyřeší

Nelze spoléhat na to, že informační systém za nás sám vyřeší všechny problémy:

- špatné řízení společnosti
- špatné fungování společnosti
- chybný podnikatelský záměr
- mezilidské vztahy
- kvalifikace zaměstnanců

1.2 Přínosy informačního systému

Mezi hlavní přínosy informačního systému můžeme řadit:

- centralizace zpracování informací
- zjednodušení evidence
- optimalizace práce
- zvýšení efektivity práce
- snížení nákladů
- rychlost získávání informací
- objem, kvalita a včasnost získávaných informací

1.3 Rozdělení IS podle zaměření

Informační systémy můžeme dělit dle jejich způsobu využití. V diplomové práci se dále budu zabývat pouze systémy pro správu obsahu (CMS).[1]

- Manažerské (EIS - Executive IS)
- Taktické (DSS - Decision Support System)
- Vedení (MIS - Management IS)
- Kancelářské (OIS - Office IS)
- Operativní:
    - TPS - transakční
    - CRM - péče o zákazníka
    - DMS - správa dokumentů
    - CMS - správa obsahu a jeho opětovné užití
    - RIS - rezervační systémy
    - GIS - geografické systémy

[1] Z anglické zkratky Content Management System

1.4 CMS

Systém pro správu obsahu (Content Management System) je v současné době všestranně zaměřeným editačním nástrojem pro tvorbu webu. WEB CMS, česky pak redakční či publikační systém. Původně byl CMS používán pro lokální správu elektronických dokumentů v počítači. Dnes je to snadno dostupný nástroj pro tvorbu webů.

Systém pro správu obsahu je softwarový nástroj, který umožňuje vytvářet, upravovat a publikovat obsah. Zatímco původní software CMS byl používán ke správě dokumentů a souborů lokálně, většina nynějších CMS je určena výhradně ke správě webového obsahu. Kdykoliv v práci zmíním CMS, bude se vždy jednat o webový CMS, neuvedu-li jinak. [4]

Zde uvádím vhodnou definici CMS: „*Internetová služba provozovaná na serveru za pomoci skriptovacích jazyků a databází za účelem zjednodušit funkci konkrétního systému, případně dozvědět se více o návštěvnících daného webu. Ne druhořadou funkcí je jistě i zpříjemnění pobytu návštěvníkům, kteří mohou pomocí systému diskutovat a jinak se interaktivně zapojovat do dění, čímž pomáhají ve vývoji portálu.“* [5]

Cílem CMS je poskytnout intuitivní uživatelské rozhraní pro tvorbu a úpravu obsahu webových stránek. Každý CMS také poskytuje nástroj pro publikování na webu, který umožňuje jednomu nebo více uživatelům tvořit obsah a vyvíjet web. Existují i softwarové programy CMS, například Adobe Contribute. V poslední době jsou ale nahrazeny webovými CMS. Většina uživatelů dává přednost webovému rozhraní, protože zjednodušuje proces aktualizace webových stránek a obecně práce je snazší a dostupná z jakéhokoliv zařízení s připojením k internetu. Většina webových redakčních systémů je navíc automaticky aktualizována, takže je neustále k dispozici v aktuální verzi.

Správně nastavený a spuštěný redakční systém dovoluje svému uživateli takřka bez jakékoliv znalosti HTML či PHP vytvářet, upravovat nebo mazat webový obsah. Často uživateli stačí „naklikat“ obsah, ať se jedná o text nebo grafický obsah. Uživatelská znalost redakčního systému na úrovni editace obsahu je velice jednoduchá. I naprostému začátečníkovi nebude trvat seznámení se s redakčním systémem déle než 15 minut. Samozřejmě složitější úpravy grafického rozvržení webu nebo instalace doplňujících funkcionalit již budou vyžadovat uživatele znalého. Stále se ovšem obejde bez nutnosti znát programovací jazyky. Jakmile bude třeba upravit útroby redakčního systému, pak už se bez znalosti HTML, PHP či Javascriptu neobejde.

Široká nabídka CMS může být nepřehledná. Každý systém se snaží poskytnout to nejlepší, aby zaujal co nejvíce potenciálních uživatelů. Pokusím se představit moderní trendy a důležité vlastnosti, které by redakční systém neměl postrádat.

Kontrola nad obsahem

Uživatel musí mít naprostou kontrolu nad obsahem i formou systému. Musí být schopen přistoupit do různých úrovní systému:

- obsah oddělen od designu
- dostupný zdrojový kód
- editace URL struktury
- tvorba, editace, mazání obsahu

Přizpůsobitelnost

Pod pojmem přizpůsobitelnost dnes budeme především hledat responzivní design, tedy schopnost optimálně zobrazit obsah na různých koncových zařízeních:

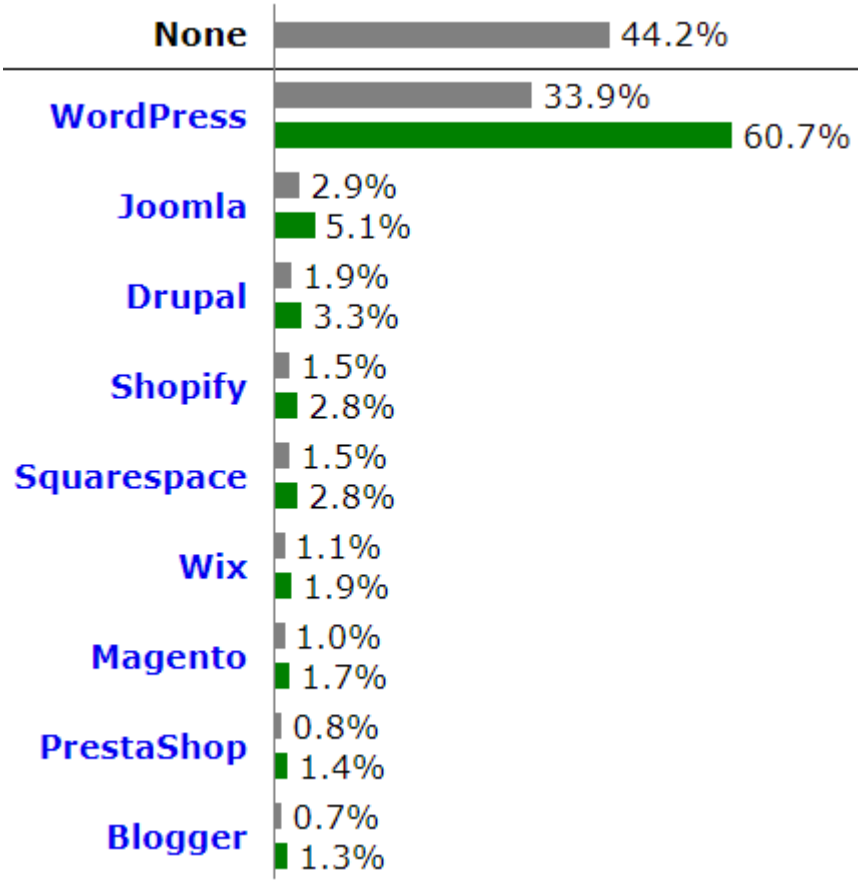
- responzivní design
- přístup ke všem funkcím v administračním rozhraní
- administrace uživatelských oprávnění
- instalace šablon a dalších doplňků

Bezpečnost

Zabezpečení redakčního systému budu věnovat celou vlastní kapitolu. Jen tedy krátce především k aktuálnosti systému. Stejně jako Microsoft aktualizuje svůj OS Windows, tak i komunity jednotlivých CMS dohlížejí na to, aby jejich produkty reagovaly na nejnovější hrozby a trhliny v kódu. Úkolem administrátora systému pak je tyto bezpečnostní záplaty přijímat a systém pravidelně aktualizovat.

1.4.1 CMS na internetu

Obrázek číslo 1 zobrazuje zastoupení redakčních systému na internetu. Méně než polovina webů, celých 44 %, není postavena na žádném redakčním systému. Zbylých 56% webů používá nějaký redakční systém.[2]



Obrázek 1. *Použití redakčních systémů na internetu*, převzato z [6]

Z obrázku číslo 1 je patrné, že s obrovskou převahou vítězí redakční systém WordPress. V následující kapitole vyberu nejpoužívanější a z mého pohledu nejoblíbenější redakční systémy a krátce o nich pohovořím.

[2] Úplný seznam redakčních systémů je k nalezení zde [6]

Obrázek číslo 2 ukazuje základní schéma redakčního systému.

Obrázek 2. *Schéma* redakčního *systému*, převzato z [7]

1.4.2 WordPress

Obrázek 3. *Logo redakčního systému WordPress*, převzato z [8]

Jak už vyplynulo z textu výše, WordPress je nejrozšířenějším, nejpoužívanějším a nejpopulárnějším volně šiřitelným redakčním systémem na světě. Hlavními vlastnostmi, kterými dosáhl WordPress tak velké popularity, jsou intuitivní ovládání a velmi estetický/líbivý vzhled na vysoké grafické úrovni, díky kterým získal WordPress široký okruh vývojářů. Pro využití všech nástrojů/možností, které WordPress nabízí, je nakonec znalost programování v PHP nezbytná. Přesto je tento redakční systém vhodný i pro úplné začátečníky. Rozvržením je WordPress spíše vhodný pro prezentaci firem či (osobních) blogů.

Historie

WordPress byl vytvořen Mattem Mullenwegem a Mikem Littlem v roce 2003 jako komplexní publikační systém vystavěný na základech b2 cafelogu, který byl ovladatelný pomocí PHP a MySQL a licencovaný v rámci GPL[3]. WordPress je aktivní a rozvíjející se systém. Zajímavost ze zákulisí tvorby programu/redakčního systému je, že tvůrci rádi poslouchají jazzovou hudbu, a tak je každá verze odlišena číslem a nese název podle slavného jazzového hudebníka. První pojmenovanou verzí byl 1.0 Miles Davis, ačkoliv se jednalo o v pořadí pátou verzi programu, sedmou byla verze 1.0.2 Art Blakey, následovala 1.2 Charles Mingus, za zmínku stojí verze 2.1 Ella Fitzgerald anebo verze 4. 7 Sarah "Sassy" Vaughan. Celkem vývojáři pojmenovali 37 verzí z celkových 366, a to včetně nejnovější 5.2 Jaco Pastorius[4]. [8]

V současnosti patří WordPress mezi nejpoužívanější redakční systémy s rozsáhlým zázemím (podporou), je celosvětově rozšířený s více než 67 miliony uživatelů. [9]

1.4.3 Joomla

Obrázek 4. *Logo redakčního systému Joomla*, převzato z [10]

Jumla znamená svahilsky „dohromady, celkem, suma" volně přeloženo autory, jako „Všichni společně". Vývojáři tím udávají směr, jakým chtějí, aby se redakční systém ubíral. Joomla je vhodný redakční systém pro středně velké a větší projekty jako jsou chaty a diskuzní fóra, e-shopy, apod. Joomla je rovněž licencována pod GPL.

Historie

Joomla navazuje na software Mambo od australské společnosti Miro Corporation z roku 2000, který sloužil výhradně pro potřeby firmy Miro Corporation. Software byl sdílen s veřejností ve snaze jejrychleji rozvíjet, což se ukázalo být dobrý tahem, a software si na-

[3] Licence pro svobodný software z angličtiny General Public License

[4] Aktuální k 10.5.2019

šlo mnoho příznivců z celého světa. První oficiální verze Joomly byla zveřejněna v roce 2005 jako dílo několika vývojářů, kteří se oddělili od původní skupiny kolem Mambo softwaru. Joomla se stejně jako Mambo nadále vyvíjí, ale je zřejmé, že si projekt Joomla vede výrazně lépe, jako druhý nejpoužívanější redakční systém, i když za WordPresem výrazně zaostává. [11]

1.4.4 Drupal

Obrázek 5. *Logo redakčního systému Drupal*, převzato z [12]

Drupal je svou náročností na užívání vhodný pro zkušenějšího vývojáře, který se v redakčních systémech již orientuje a navíc ovládá PHP a HTML alespoň na základní úrovni. Uživateli přijde vhod i základní znalost programovacích jazyků. Drupal se hodí na větší a středně velké projekty, tedy pro mezinárodní společnosti, velké e-shopy, renomované zpravodaje, apod.

Historie

Drupal byl, podobně jako mnoho jiných komunikačních kanálů, původně navržen jen pro komunikaci mezi studenty na koleji. Navrhl jej Dries Buytaert, holandský student. Odtud se vzal i název, který vznikl zkomolením z holandského označení pro vesnici - drop s výslovností [druppel]. Systém byl dále rozvíjen a v roce 2003 byl použit jako komunikační kanál a ke sdílení materiálů ve společnosti DeanSpace. Systém byl nadále rozvíjen vývojáři z původní komunity spolu se zakladatelem Driesem Buytaertem. [13]

1.4.5 Sharepoint

Obrázek 6. *Logo redakčního systému SharePoint*, převzato z [14]

Na rozdíl od ostatních redakčních systémů, které jsem zmínil výše, Microsoft SharePoint není software, který byl primárně určen pro tvorbu webových stránek. Jeho hlavní role spočívá ve správě obsahu, dokumentů a projektů, i když je poslední dobou stále více přizpůsobován právě tvorbě webových stránek, což v podstatě splňuje definici CMS, kdy se jedná převážně o publikování různých dokumentů. Jedná se o produkt z rodiny Microsoft a jde o robustní placený nástroj pro velké organizace.

Často se můžeme setkat s tím, že SharePoint je vhodným řešením pro intranet. Perfektní integrace produktů Microsoft Office a dalších kancelářských aplikací usnadňuje každodenní práci jeho uživatelům. Přestože je pro intranetové použití populárnější, neznamená to nutně, že s jeho pomocí nelze vytvořit externí web. Ovšem pokud se rozhodnete použít Sharepoint jako hlavní webovou prezentaci, bez zkušených programátorů se v tomto případě neobejdete. [15]

Historie

Služba SharePoint byla spuštěna v roce 2001. Microsoft původně nabídl SharePoint ve dvou verzích. On-premise, tedy lokálně hostovaný systém, a dnes již mnohem častější cloudové řešení nabízené mnohdy společně s Office 365. [16]

1.4.6 SDL Tridion

Obrázek 7. *Logo redakčního systému SharePoint*, převzato z [17]

Po úspěchu v Evropě se společnost SDL Tridion zaměřila na trh v USA, kde působí od roku 2006. CMS - Tridion R5 přichází se standardními funkcemi Web CMS. Jedná se o podnikový systém, který není nejlevnější. Pro jeho získání je třeba pořídit licenci. Z tohoto důvodu nenalezneme mnoho výukových materiálů, jako je tomu v případě open source CMS WordPress, Joomla, Drupal. Aktuální verze je značena jako SDL Web 8. Systém je určený pro rozsáhlé organizace, kdy například každá země potřebuje vlastní prostředí s vlastním obsahem. Obvyklou konstrukci pak tvoří 3 prostředí pro jeden projekt (vývoj, test, produkce). [18]

Historie

Společnost SDL Tridion původně pouze „Tridion“ byla založena v roce 1999. Sídlí v nizozemském Amsterdamu a má pobočky ve Velké Británii, Francii, Německu, Belgii, Španělsku a Švédsku. V roce 2007 získala společnost SDL ocenění jako přední poskytovatel v oblasti Global Information Management. Nejstarší verzí produktu je R3. Následovala řada nových, včetně nejnovější verze R5.3, která byla spuštěna na konci roku 2007. Nejčastěji používaným předchůdcem je R5.2 SP1. Nicméně mnozí zákazníci stále používají i starší verze. Hlavní předností SDL Tridion je vícejazyčná podpora a systém dědičnosti známý jako BluePrinting. [19]

Některé CMS nástroje jsou zdarma, za některé je třeba platit měsíční poplatek anebo pořídit licenci pro určitý počet uživatelů. Mnoho CMS poskytuje bezplatné základní komponenty. Často si ovšem musíte připlatit za kvalitnější šablonu nebo za její nadstandardní verzi. Než se rozhodnete pro volbu finálního CMS, je vhodné vyzkoušet více systémů a zjistit, který bude pro konkrétní projekt nejvhodnější. Zde neexistuje univerzální odpověď. Volba CMS proto bude vždy záviset na konkrétních požadavcích společnosti.

2 KYBERNETICKÁ BEZPEČNOST

V úvodní kapitole vysvětlím obecně problematiku kybernetické bezpečnosti. Kybernetická bezpečnost je podmnožinou Informační bezpečnosti. Tedy nejprve definuji informační bezpečnost a pak se budu věnovat specifikům bezpečnosti kybernetické.

2.1 Informační bezpečnost

Jedná se o ochranu informačních a komunikačních technologií a všeho, co s nimi souvisí. Zjednodušeně řečeno, cílem informační bezpečnosti je ochrana informací v jakékoliv podobě. Cílem kybernetické bezpečnosti je především ochrana informací v digitální podobě.

Informační bezpečnost pak můžeme definovat jako: *"Schopnost sítě nebo informačního systému jako celku odolat s určitou úrovní spolehlivosti náhodným událostem nebo nezákonným nebo svévolným zásahům, které ohrožují dostupnost, pravost, integritu a důvěrnost uchovávaných či přenášených údajů a souvisejících služeb poskytovaných nebo přístupných prostřednictvím těchto sítí a systémů."* [20]

Je to soubor opatření mající za úkol návrh, schválení, implementaci softwarových, hardwarových, technických a personálních opatření, které mají vést k minimalizaci potenciálních ztrát nebo vzniku škod způsobených narušením bezpečnosti. Cílem informační bezpečnosti je ochrana informací a majetku před krádeží, korupcí, nebo přírodní katastrofou. [21]

2.2 Kybernetická bezpečnost

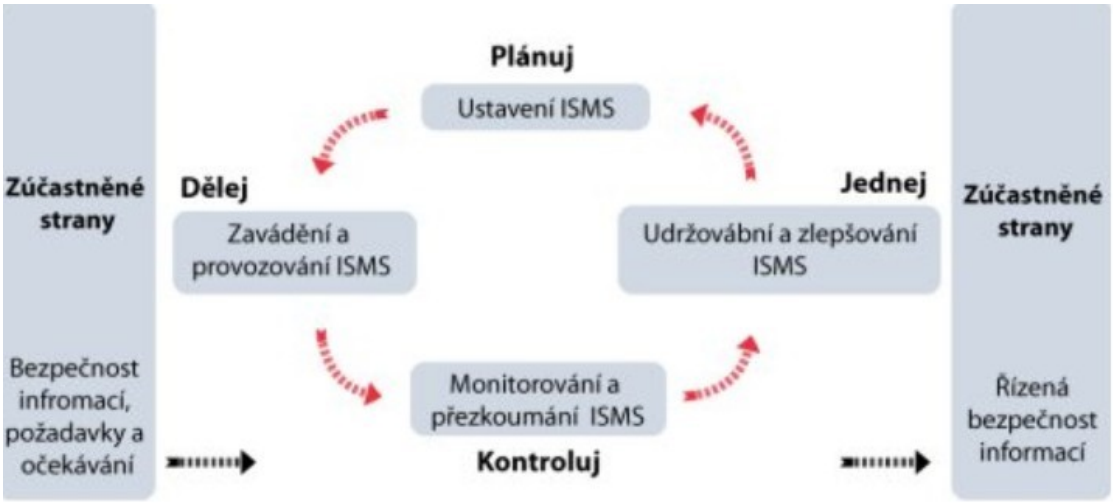
Kybernetická bezpečnost je podmnožina informační bezpečnosti. Zajištění kybernetické bezpečnosti v rámci informačního systému vyžaduje jisté úsilí a koordinaci v těchto aspek-tech:

- bezpečnost aplikací
- bezpečnost informací
- zabezpečení sítě
- vzdělávání koncových uživatelů

Standardy kybernetické bezpečnosti

Vhodnou pomůckou jsou standardy kybernetické bezpečnosti, jež byly zavedeny poměrně nedávno. Důvodem zavedení je vzrůstající množství citlivých údajů uchovávaných v počítačových systémech. Ty jsou téměř vždy propojeny v rámci sítě Internet. Důležitým aspektem kybernetické bezpečnosti je ochrana před krádeží identity. Instituce a firmy potřebují chránit citlivé osobní údaje, know-how a další soukromá data. Proto v roce 1995 vznikl standard ISO / IEC 27002, dnes přejmenovaný na ISO 27001. Národní institut pro standardy a technologie Spojených států Amerických (NIST) dále zavedl několik publikací řešících informační bezpečnost. Nejdůležitější je Příručka Informační bezpečnosti 800-12 a Obecně uznávané zásady a postupy pro zabezpečení informačních technologií 800-14.

Zavedení kybernetické bezpečnosti by mělo být v souladu s celkovou bezpečnostní politikou organizace. Jako klíčové se jeví zakotvení silného postavení nástrojů bezpečnosti v organizaci. Základním aktem je ustanovení bezpečnostního manažera do první linie řízení organice, který přímo podléhá generálnímu řediteli. Jeho cílem je stanovení bezpečnostní politiky organizace, ideálně v podobě systému řízení bezpečnosti ISMS[5] 27000 a jeho následné zavedení v organizaci.



Obrázek 8. *Model plánuj, dělej, kontroluj, jednej*, převzato z [22]

Norma ISO 27001 zavádí model Plánuj-Dělej-Kontroluj-Jednej jako součást přístupu systému řízení k vývoji, implementaci a zdokonalování efektivnosti systému řízení bezpečnosti informací v organizaci. [23]

[5] Systém řízení bezpečnosti informací, z angličtiny Information Security Management System

Hrozby kybernetické bezpečnosti

Držet krok s novými technologiemi, bezpečnostními pravidly a hrozbami z kyber světa je velice obtížný úkol. Přesto je velice důležité snažit se těmto hrozbám předcházet za každou cenu. Nejběžnější hrozby mohou mít mnoho podob:

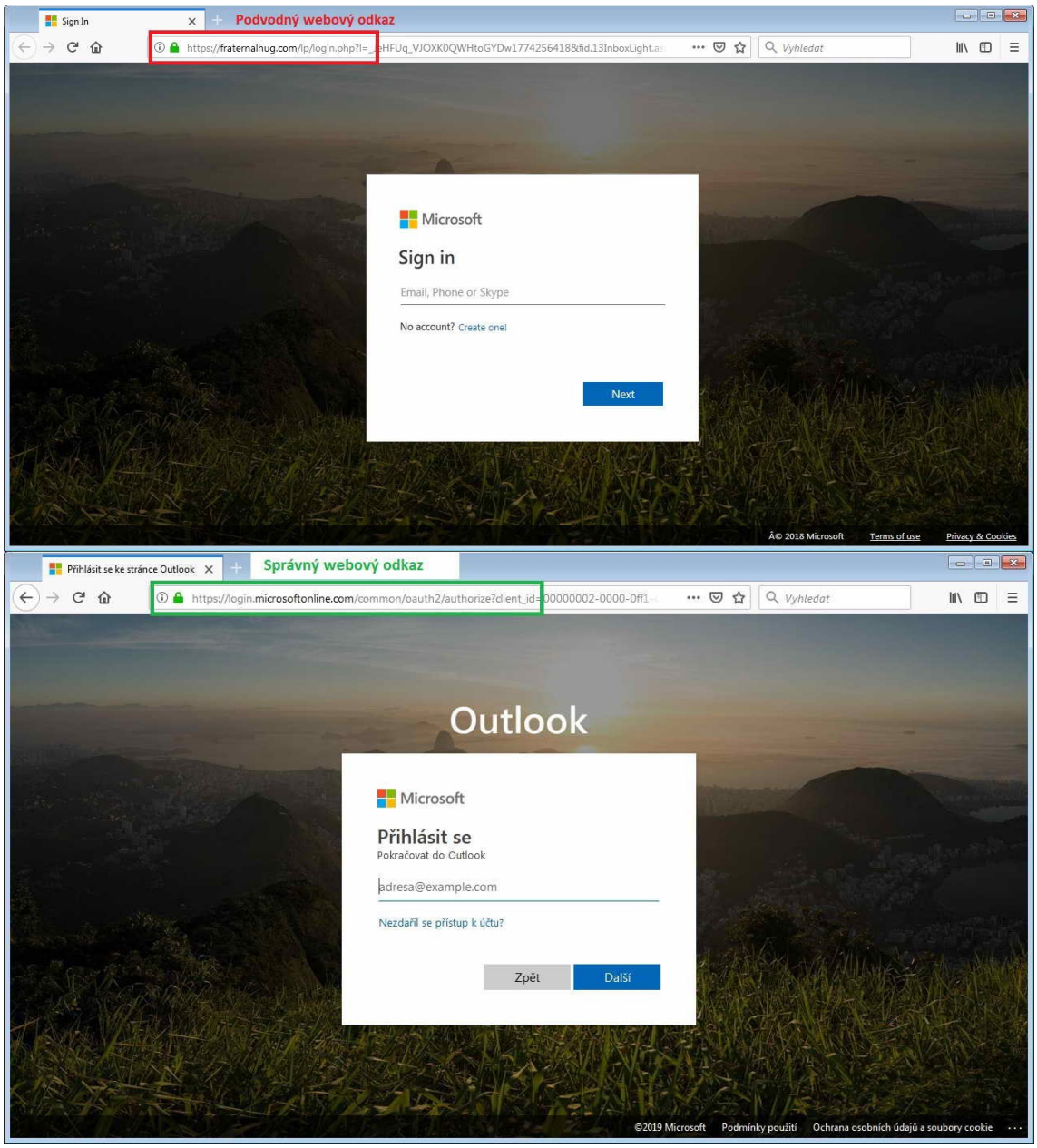
- Malware[6] je typ obtěžujícího nebo škodlivého softwaru, který má za úkol zajistit útočníkovi tajný přístup do zařízení. Mezi typy malwaru patří spyware, adware, phishing, viry, trojské koně, červy, ransomware a změny nastavení prohlížeče.
- Ransomware omezuje uživatelům přístup k jejich počítačovému systému nebo souborům. Za obnovení přístupu požaduje program zaplacení výkupného. Nejnebezpečnější útoky tohoto typu má na svědomí ransomware WannaCry, Petya, Cerber, CryptoLocker a Locky.
- Sociální inženýrství je útok, který spoléhá na lidskou neznalost. Útočník se snaží přimět uživatele k porušení bezpečnostních postupů, aby získal citlivé informace nejčastěji uživatelské jméno a heslo. Typický útok je cílen na jednotlivce emailem nebo telefonicky.
- Trojský kůň je typ viru, který předstírá, že uživateli přináší přidanou hodnotu v podobě užitečné funkce nebo bezplatného programu. Ve skutečnosti však způsobuje škody nebo krádeže dat. Na trojského koně můžete narazit nejčastěji v emailech a na stránkách, kde je sdílen software (často nelegální).
- Phishing je snaha počítačových podvodníků získat vaše citlivé osobní informace, jako jsou hesla, údaje o platebních kartách, rodná čísla nebo čísla bankovních účtů. Šíří se podvodnými emailovými zprávami nebo přesměrováním na falešné webové stránky.

Na následujícím obrázku číslo 9 můžete vidět nedávný[7] phishingový útok na přihlašovací stránku emailového serveru Outlook UTB. Jak je z obrázku patrné, jedná se o velice zdařilou napodobeninu přihlašovacího formuláře. Podvodná stránka dokonce disponuje SSL certifikátem, což pouze zvyšuje důvěru útoku. Proto je vždy třeba hlídat nejen vizuální podobu stránky, ale také adresní řádek. Při jakémkoliv podezření na podvrh

[6] Z angličtiny malicious software

[7] K datu 16.5.2019

web opustit a přijít na něj znovu oficiální cestou. Přehlédnutí právě těchto drobných rozdílů je to, na co strůjci phishingového útoku spoléhají.



Obrázek 9. *Phishingový útok Outlook UTB*, převzato z UTB

Společnost Microsoft se již delší dobu snaží o implementace passwordless přístupu uživatelů, tedy úplně bez hesel. Jako prostředek k dosažení tohoto cíle byl vybrán autentizační framework FIDO 2 společně s doplňkem YubiKey viz 2.11. Rozšíření

passwordless[8] přístupu by do budoucna mohlo zcela zamezit phishingovým útokům, a to z prostého důvodu: nebude již existovat přihlašovací stránka ve smyslu uživatelské jméno a heslo, kterou by útočník podvrhnul. [24]

Dodržováním zásad kybernetické bezpečnosti můžeme do jisté míry účinně předcházet kybernetickým útokům, krádežím citlivých dat nebo uživatelských identit. Ochrana koncového uživatele by měla být prioritou. Této problematice se podrobně věnuji v následující kapitole autentizace.

2.3 Autentizace

Autentizace je proces zjištění totožnosti uživatele. Primárním cílem autentizace je zjistit, zda je subjekt opravdu tím, za koho se vydává. Dalším úkolem autentizace je zjistit informace o vstupujícím subjektu (jméno, email, telefon, adresa, atd.).

Metody autentizace jsou založeny na prokázání identity subjektu. Máme tři základní způsoby autentizace:

- znalost – uživatel zná heslo, pin, kontrolní otázku
- vlastnictví – uživatel disponuje USB tokenem, identifikační kartou, SW klíč, platební karta
- vlastnost – uživatel se identifikuje pomocí biometrické informace, tedy otisk prstu, sítnice, DNA, rozpoznání obličeje

Nejčastěji autentizace uživatele vypadá následovně. Uživatel chce přistoupit k nějaké službě. Ta jej vyzve, aby se identifikoval. Možností identifikace je mnoho, tady uvádím některé z nich:

Jméno a heslo

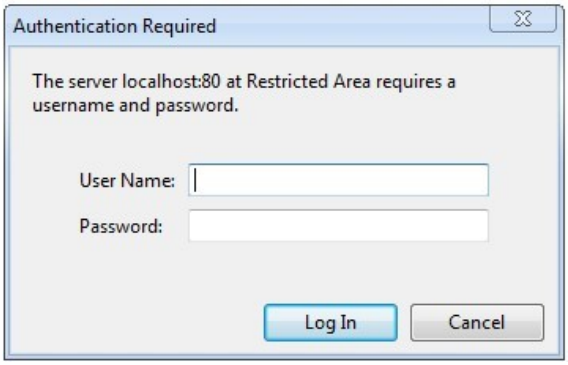
Jedná se o nejsnazší a velice rozšířené řešení. Web disponuje formulářem pro přihlášení a registraci. Uživatel se registruje, nastaví osobní údaje včetně hesla a ta jsou uložena na server do databáze uživatelských účtů. Odeslání přihlašovacích údajů je provedeno pomocí

[8] Z angličtiny bez hesla, passwordless

HTTP[9] metodou POST. Dnes především pomocí HTTPS. Jedná se o původní protokol rozšířený o šifrovací vrstvu, a proto je přenos jeho pomocí považován za bezpečný. Více o SSL[10] v 6.1. Výhodou je, že proces správy uživatelských účtu je plně přizpůsobitelný potřebám konkrétní služby a zcela v režii majitele služby. Zodpovídá také za bezpečnost citlivých údajů uložených v databázi. Nevýhoda je především pro uživatele: pro každou novou registraci si musí pamatovat nové přístupové údaje. Uživatelé pak často používají stejná hesla, jako mají u jiných služeb, což vede k ztrátě bezpečnosti hesla viz. 3.3

Jednoduché ověření přístupu

Basic access authentication je název pro jednoduchou autentizaci přístupu na webovou stránku. Web server požaduje po klientovi, v našem případě webový prohlížeč, zaslání jména a hesla. Je to velice triviální omezení přístupu na web pomocí HTTP hlaviček. Největší nevýhodou je absence zabezpečení. Heslo i jméno je přenášeno v plain textu[11] zakódované metodou Base64. Proto je třeba tuto metodu používat výhradně v kombinaci s HTTPS. [25]



Obrázek 10. *Basic access authentication*, zdroj: vlastní

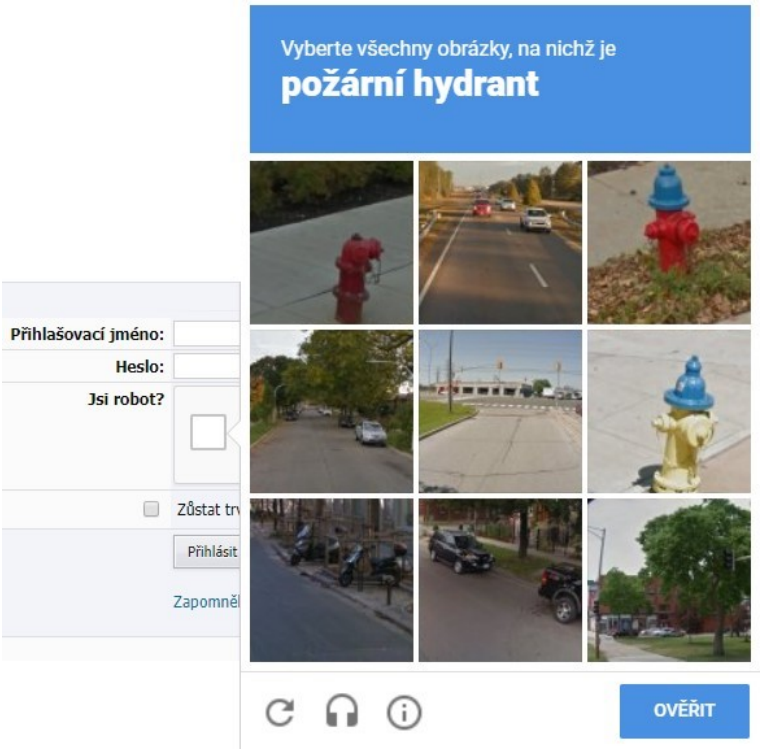
[9] Z angličtiny The Hypertext Transfer Protocol, protokol umožňující komunikaci mezi klientem a serverem v prostředí WWW

[10] Z angličtiny Secure Socket Layer, je protokol, který mezi transportní vrstvu TCP/IP a aplikační vrstvu HTTPS vloží relační vrstvu poskytující šifrování

[11] Z angličtiny prostý text, není žádným způsobem formátovaný ani šifrovaný (kódování ASCII, ANSI, UTF-8)

Captcha[12]

Plně automatický veřejný Turingův test k odlišení počítačů a lidí. Hlavním úkolem je zjistit, zda uživatel není robot. Test spočívá v předložení operace, kterou robot není schopen vyřešit. Existují různé druhy těchto operací. Může se jednat o identifikaci obrázků, zvuku nebo nejrozšířenější přepis textového řetězce.



Obrázek 11. *Obrázková captcha*, zdroj: www.google.com

Fyzické zařízení

K identifikaci uživatele se použije nějaké fyzické zařízení jako USB token, bankovní karta, softwarový klíč. Tato zařízení často požadují k vlastnímu správnému fungování heslo nebo PIN, což zvyšuje jejich bezpečnost.

Biometrická identifikace

Biometrie je věda zabývající se identifikací uživatele na základě jeho fyzických vlastností. Spadají sem metody jako otisk prstu, dlaně, sken sítnice, obličeje, hlasová identifikace.

[12] Z angličtiny Completely automated public Turing test to tell computers and humans apart

Nevýhodou této metody je nutnost speciálních skenovacích zařízení, která tuto identifikaci umožní.

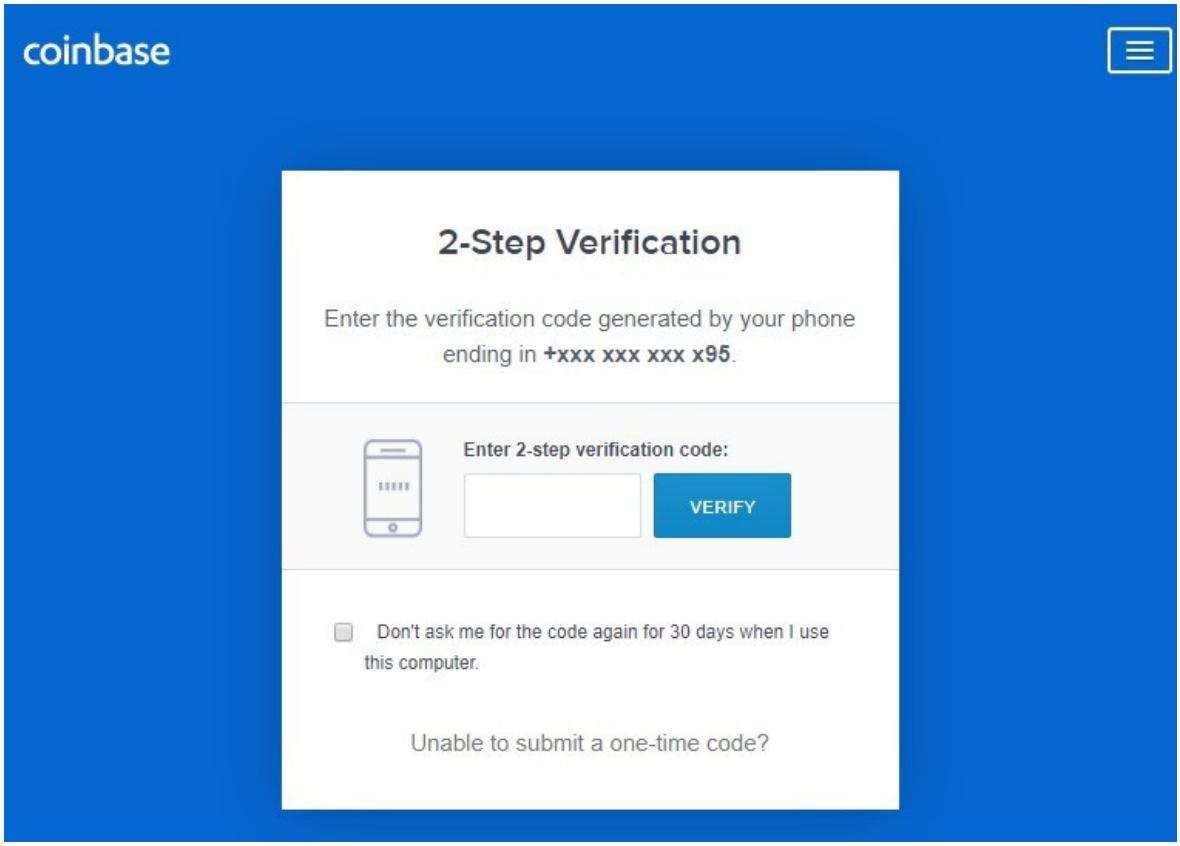
Token

Jedná se o fyzické zařízení, které uživatel použije pro přístup k zabezpečenému systému. Nejčastěji se jedná o kartu nebo čip. Výhodou je, že token disponuje silným jménem a heslem, uživatel je samozřejmě nezná a dle implementace tokenu se tyto přihlašovací údaje mohou v čase měnit. Jednoznačnou nevýhodou je pak ztráta tokenu.

Multi-Factor

Ověření na základě více faktorů obvykle vyžaduje dva nebo více nezávislých způsobů ověření identity uživatele. Nejčastěji jde o kombinaci předchozích metod ověření identity. Typickým příkladem je použití bankovní karty. Pro úspěšný výběr peněz potřebujete fyzicky kartu vlastnit a znát její PIN. Dnes typicky používané u služeb jako internetové bankovnictví (heslo + sms zpráva nebo lokálně instalovaný certifikát vydaný a ověřený certifikační autoritou tedy bankou). Tento způsob autentizace najdeme v podstatě u všech služeb manipulujících s financemi. Také Google, Facebook a další internetové služby, které de facto zastupují lidskou osobu na internetu, používají čím dál sofistikovanější možnosti autentizace uživatele. [26], [27]

Na obrázku číslo 11 je ukázka 2-faktorové autentizace. Uživatel je po zadání jména a hesla vyzván k vyplnění kódu, který byl odeslán na jeho telefon jako sms.



Obrázek 12. *2Faktorové ověření*, zdroj: www.coinbase.com

2.4 Autorizace

Autorizace je proces zjištění způsobilosti. Význam slova autorizovat je povolit nebo schválit. Jde o případ, kdy uživatel, program nebo zařízení chce přistupovat k určitým zdrojům (server, soubor, tiskárna, služba). Aby se tak mohlo stát, musí být daná entita autorizovaná tedy oprávněná, musí mít přístupová práva k potřebné akci. Předpokladem autorizace je předchozí úspěšná autentizace.

Typickým příkladem autorizačního standardu je oAuth 2.0 viz 2.9.

Autentizace vs. Autorizace

Termíny autentizace a autorizace bývají často mylně pochopeny. Přestože v některých případech mohou být realizovány společně, jedná se o dvě odlišné funkce. Autentizace je proces ověření identity uživatele. Aautorizace je proces ověření platnosti přístupu uživatele k nějakému prostředku. Proces, kterým je přístup k těmto prostředkům omezen na určitý počet uživatelů, se nazývá řízení přístupu. Autentizace vždy předchází procesu autorizace.

2.5 SSO

Single sign on je efektivní řešení autentizace uživatelů. Uživatel se přihlásí zadáním uživatelského jména a hesla pouze jednou a je tak přihlášen do všech aplikací, ke kterým má oprávnění a kterým se implementuje daný SSO model. Nemusí se tedy přihlašovat znovu při přechodu mezi aplikacemi. Typickým příkladem web SSO může být Google. Jakmile se přihlásíte do svého google účtu, jste automaticky přihlášeni do všech jeho služeb jako Gmail, Youtube, Disk, Photos, Calendar.

Největší výhodou je právě jednotné přihlášení do všech spolupracujících aplikací. Uživatel se nemusí přihlašovat do každé zvlášť, stačí mu pouze prvotní přihlášení. Vzhledem k tomu, že každá větší společnost poskytuje svým zaměstnancům více aplikací, může implementace SSO snížit náklady na provoz jednotlivých aplikací, jelikož ty nemusejí řešit problémy spojené s autentizací do jednotlivých systémů. Mezi největší nevýhody patří fakt, že při ztrátě nebo zcizení identity ověřeného uživatele se útočník dostane do všech aplikací a služeb, pro které je SSO implementováno.

2.6 Federační model

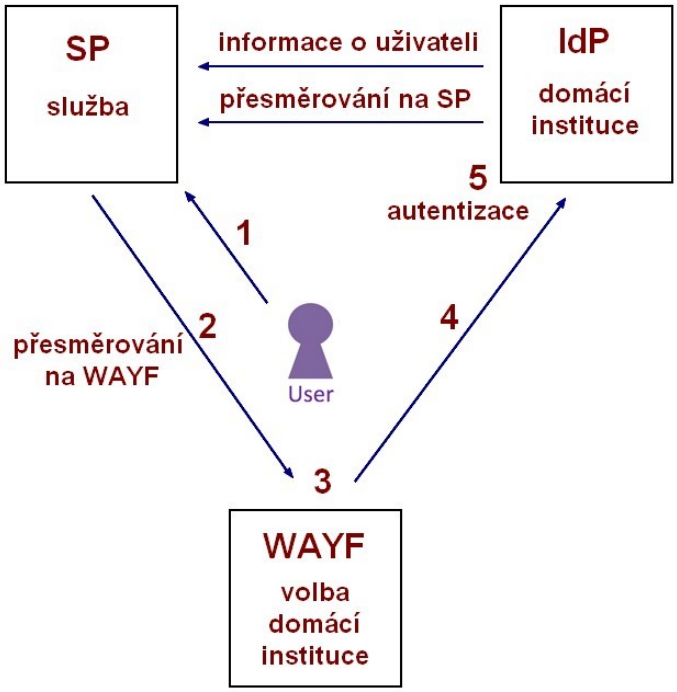
Federační model umožňuje i naprosto odděleným informačním systémům (jiná instituce, škola, společnost sídlící v jiném místě než domovská instituce), které jsou však propojeny s domovským informačním systémem uživatele v tzv. federaci, vyměňovat data o svých uživatelích. Systémy propojené ve federaci jsou schopné získat informaci o uživateli přímo z jeho domovské instituce. Tam se nacházejí aktuální data o uživateli a není tedy třeba udržovat totožná data uživatele na více místech. Eliminace redundantních dat v síti a současně odpadá nutnost synchronizace těchto dat při jejich změně. Federační model využívá SSO přístup takže usnadňuje práci i samotným uživatelům. Infrastruktura federace pak zajistí vzájemné předávání potřebných údajů mezi subjekty. [28]

Federační model obsahuje dvě základní komponenty:

- IdP (Identity Provider) – poskytovatel identity, typicky server, který autentizuje uživatele a poskytne o něm základní informace
- SP (Service Provider) – služba, obvykle webová aplikace, ke které chce uživatel získat přístup

Toto rozhraní (Identity Provider, Service Provider) implementují všechny instituce zapojené ve federaci. Jelikož jsou tyto komponenty dostupné napříč všemi systémy, může každá služba (SP) získat informaci od Identity Providera (IdP) o uživateli, který žádá o přístup. Architektura federačního modulu zajišťuje, že SP dotáže toho IdP, který pochází z domovského systému uživatele žádajícího o přístup k SP.

Tento princip zaštiťuje služba WAYF[13]. Na obrázku číslo 13 je zobrazen princip federačního modelu. Uživatel přistoupí k SP. SP kontaktuje službu WAYF, ta přeposílá požadavek na IdP uživatele, kde současně proběhne autentizace. Následně IdP posílá informace SP a uživatel je přesměrován na službu již autentizován. [28]



Obrázek 13. *Federační model flow*, zdroj: vlastní

Federace je dosti obsáhlé téma. Ještě zmíním, že existují dva hlavní přístupy:

- Federace bez hesla – implementována za pomocí standardů SAML2.0, OAuth2.0, OpenID, které zajišťují bezpečnost celé infrastruktury. Identita je ověřována pomocí tokenů, které standardy komunikují se serverem, a uživatel nezadává heslo.

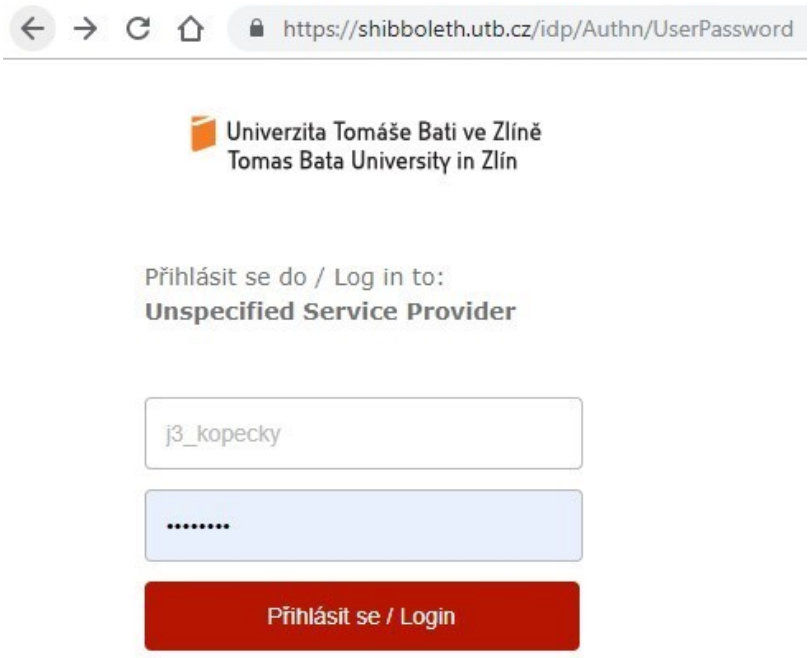
[13] Z Angličtiny Where Are You From, služba, která zjistí domovský IdP uživatele

- Enterprise federace (podniková) – služby vyžadují heslo, ale to je doplněno automaticky od IdP. Uživatel heslo fyzicky nezadává, ale pro každou aplikaci existují přihlašovací údaje a uživatel si je stále musí pamatovat a hesla pravidelně měnit tak, jak mu ukládají pravidla pro hesla.

Jako vhodné autentizační standardy nejen pro implementaci federačního modelu se během posledních let osvědčily SAML2 , OAuth 2.0 a OpenID. Jsou neustále zdokonalovány a jejich implementace nalezneme v knihovnách mnohých programovacích jazyků. [29]

2.7 Shibboleth

Jedna z nejčastějších implementací federačního modelu se nazývá Shibboleth. Celý projekt Shibboleth je zdarma, open source dostupný v rámci licence Apache Software License. Typický je pro akademické prostředí. Najdeme ho i v prostředí UTB viz obrázek 14 služba www.citace.cz ve federaci s portálem UTB.



Obrázek 14. *Federace Citace.cz spolu s UTB.cz*, zdroj: www.utb.cz

Technologie Shibboleth používá dvě komponenty:

- Identity Provider (IdP) – zpracovává autentizační žádosti od poskytovatelů služeb a zajišťuje samotnou autentizaci uživatelů typicky proti LDAP[14] nebo Kerberos[15]. Přihlašovací údaje se zadávají přímo IdP, nikam dál se neposílají. [30]
- Service Provider (SP) – vytváří a posílá autentizační žádosti na IdP. Zpracovává získané informace o uživatelích a předává je žádajícím aplikacím. [31]

Systémy implementující Shibboleth si mezi sebou dříve vyměňovaly soubory cookies. Nicméně ty jsou sdílené pouze v rámci stejné domény. To do jisté míry řešil multi-domain SSO. Později byl zaveden Framework SAML[16]. [32]

2.8 SAML 2.0

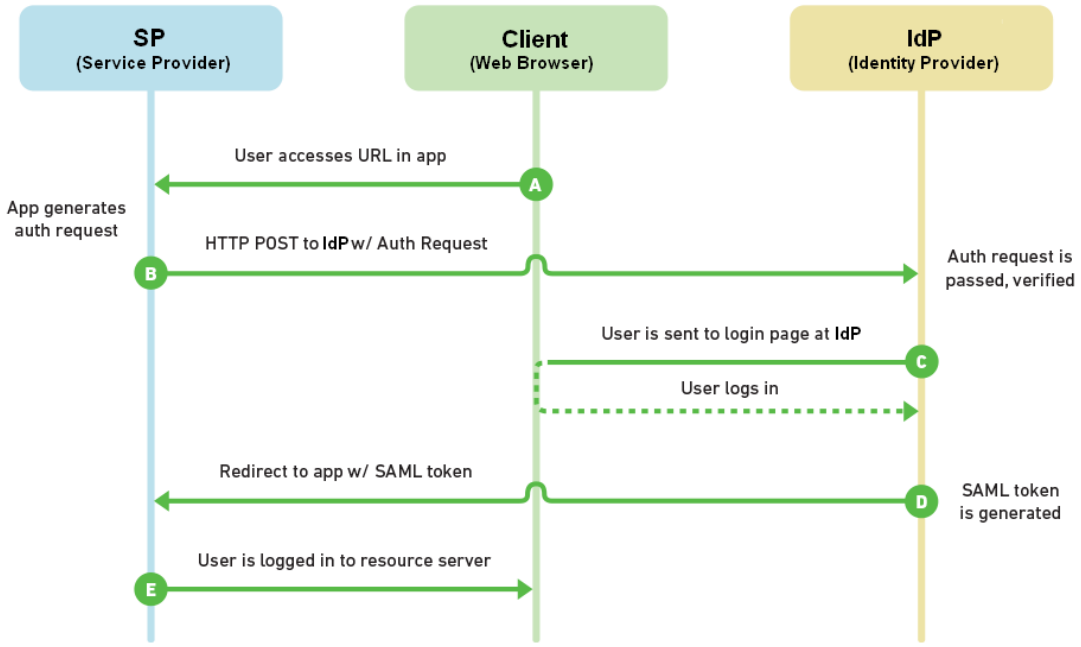
SAML je otevřený XML standard pro zajištění autentizace a autorizace mezi IdP a SP. Standard bezpečně sdílí identitu uživatele napříč různými doménami již od roku 2005. Nejčastěji najdeme jeho implementace ve federačním modelu.

- Service provider (SP) – služba, obvykle webová aplikace, ke které chce uživatel získat přístup
- Klient – uživatel komunikující skrze webový prohlížeč
- Identity Provider (IdP) - poskytovatel identity, typicky server, který autentizuje uživatele a poskytne o něm základní informace

14 Z angličtiny Lightweight Directory Access Protocol, způsob uložení dat na servru do stromové struktury. Vhodný pro práci s adresáři a informacemi o uživatelích.

15 Síťový autentizační protokol, ověřuje identitu aplikací ve vztahu klient/server, požívá kryptografii pro zabezpečenou komunikaci

16 Security Assertion Markup Language



Obrázek 15. *SAML 2.0. Flow*, převzato z: [33]

A) Uživatel otevře službu my.documents.com. Služba nemá na starost autentizaci uživatele.

B) SP vytvoří SAML authnrequest[17], podepíše ho a optimálně také zašifruje. Pak přesměruje uživatele na IdP. IdP obdrží authnrequest, dešifruje a ověří podpis.

C) Pokud je authnrequest validní, IdP zobrazí uživateli přihlašovací formulář, kde uživatel ověří svou identitu pomocí jména a hesla.

D) Jakmile se uživatel úspěšně přihlásí, IdP vytvoří SAML token nesoucí informace o uživatelovi a jeho identitě. IdP pošle SAML token SP (my.documents.com).

E) SP ověří SAML token, dešifruje, získá informace o uživateli (kdo to je a jakou úroveň přístupu je třeba nastavit). Poté SP přihlásí uživatele a poskytne patřičná oprávnění.

2.9 OAuth 2.0

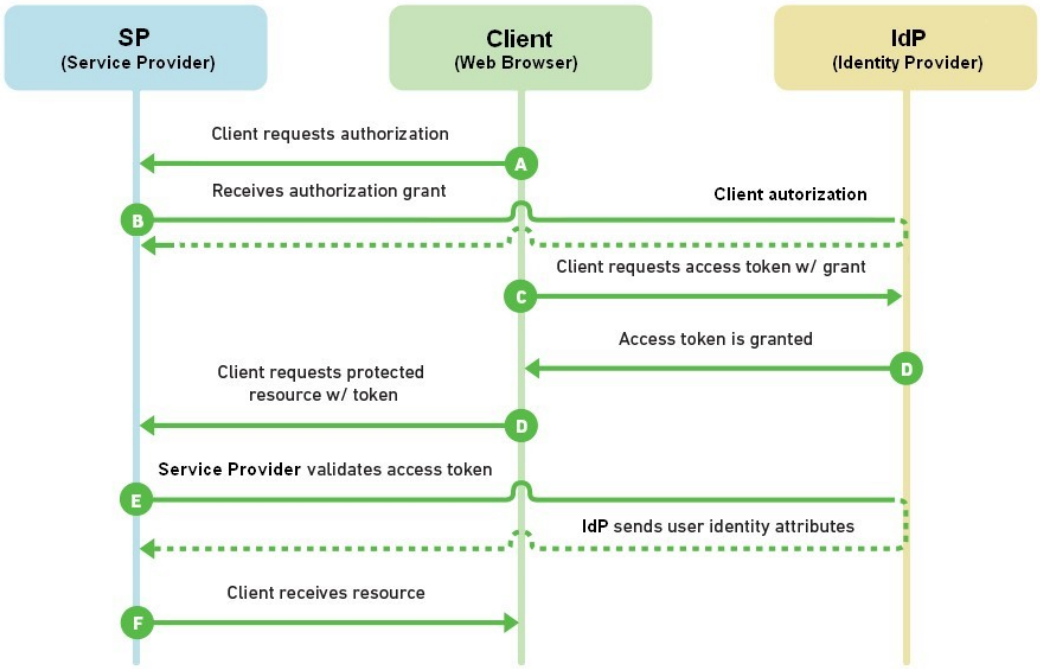
Na rozdíl od předchozího standardu je OAuth 2.0 docela nováček. Vznikl v roce 2012 a je to další otevřený standard pro autorizaci uživatele. Nutno podotknout, že pouze pro

[17] Spojení authentication a request – autentizační požadavek

autorizaci. Může být použit i pro autentizaci, avšak se svým doplňkem OpenID Connect. Standard používají poskytovatelé jako Google, Facebook, Twitter, Yahoo a další.

OAuth nedisponuje přihlašovacími údaji uživatele. Místo přihlašovacích údajů používá přístupové tokeny, jejichž pomocí ověřuje identitu uživatele.

- Service provider (SP) – služba, obvykle webová aplikace, ke které chce uživatel získat přístup
- Klient – uživatel komunikující skrze webový prohlížeč a mobilní zařízení, jelikož OAuth je vhodný pro všechna smart zařízení.
- Identity Provider (IdP) - poskytovatel identity, typicky server, který autentizuje uživatele a poskytne o něm základní informace



Obrázek 16. *OAuth 2.0. Flow*, převzato z: [33]

A) Uživatel otevře službu my.documents.com. Služba nemá na starost autentizaci uživatele.
B) Uživatel je přesměrován k IdP, kde se musí autorizovat. Obvykle pomocí formuláře jméno a heslo s dotazem od IdP, zda chce poskytnout přihlášení SP my.documents.com. Klient obdrží authorization grant a posílá ho dál k SP.
C) Klient žádá o access token IdP a posílá mu authorization grant.

D) Pokud je authorization grant platný, IdP poskytne access token. Klient získá access token a současně ho pošle SP.

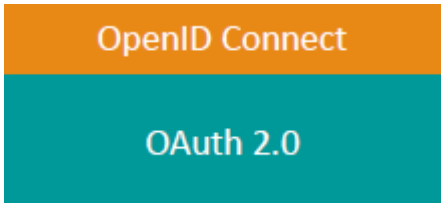
E) SP dostane od klienta access token a pro jistotu ještě znovu kontroluje jeho platnost přímo u IdP (ochranný mechanizmus proti podvrhnutí tokenu během komunikace). Pokud je access token platný, IdP posílá SP informace o uživateli.

F) SP posílá klientovi požadovaný obsah.

Obecně lze říci, že se OAuth flow od SAML příliš neliší. Hlavní rozdíl mezi oběma standardy je ten, že SAML zajistí autentizaci a autorizaci zatímco OAuth pouze autorizaci. SAML používá XML konstrukci tokenů, zatímco OAuth tokeny jsou obvykle ve JSON formátu. I proto SAML není vhodný pro komunikaci s mobilními zařízeními. [34]

2.10 OpenID Connect

OpenID Connect přidává další vrstvu standardu Oauth 2.0. Dalo by se říci, že se jedná o jeho rozšíření. OpenID Connect obohacuje Oauth o ID token a userInfo endpoint[18] pro získání informací o uživateli od IdP. Vše je součástí implementace standardu. [35]

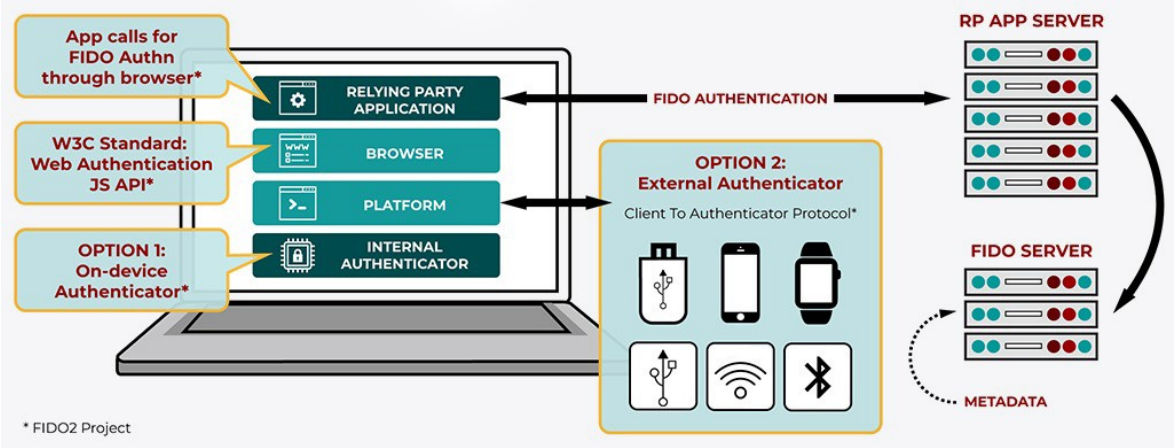


Obrázek 17. *OpenID Connect rozšíření OAuth 2.0,* zdroj: vlastní

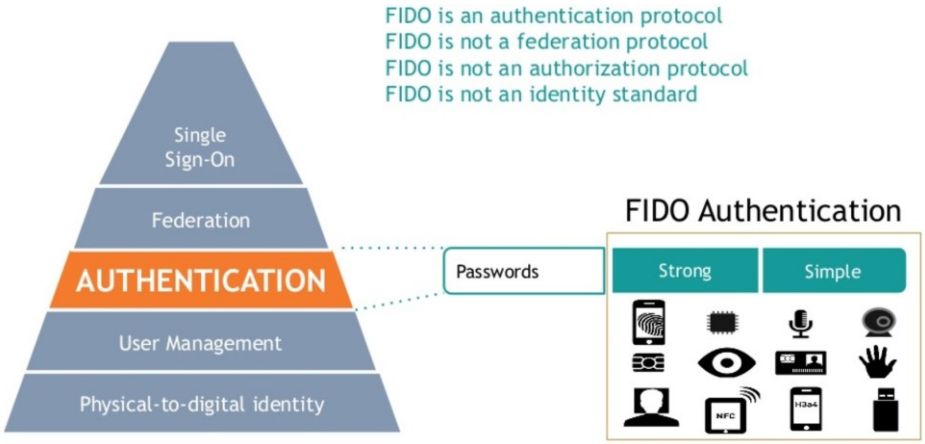
2.11 FIDO 2

FIDO je anglická zkratka pro Fast Identity Online a FIDO 2 je nový standard, který spíše doplňuje identity standardy OpenID Connect a SAML. FIDO si klade za cíl usnadnit implementaci nejrůznějších způsobů autentizace. Řeší pouze autentizaci, neposkytuje žádná data o uživateli, komunikuje pouze detaily o autentizaci (úspěch, neúspěch, počet opakování, atd.) klienta s Identity providerem (IdP). [36]

[18] Z angličtiny koncový bod, v našem případě na serveru IdP, ze kterého jsou poskytnuty informace o uživateli



Obrázek 18. *FIDO 2 flow,* převzato z: [36]



Obrázek 19. *FIDO 2 autentizace,* převzato z: [37]

3 IDENTITY A ACCESS MANAGEMENT

Identity management je spojen s autentizací uživatelů, zatímco access management řeší jejich autorizaci. Problémem je, že většina uživatelů si neuvědomuje rozdíl mezi autentizací a autorizací, což může vést k chybám, kterých hacker dokáže využít.

3.1 Identita

Stejně jako v reálném i v digitálním světě máme identitu. V tom digitálním naší identitu představuje sada záznamů a atributů v nějaké databázi daného systému. Současným trendem je shromáždit co nejvíce informací o uživateli a připravit mu tak personalizované uživatelské prostředí, profil, nabídku produktů, reklamu, atd.

Na základě těchto atributů se odlišujeme od ostatních uživatelů systému. Rozlišujeme dva typy atributů:

- statické – nemění se, typicky jde o jméno, pohlaví, původ
- dynamické – proměnné v čase, věk, pracovní pozice, adresa, rodinný stav

Digitální identita vzniká při registraci uživatele v systému. Počet a typ atributů se bude u každé registrace lišit v závislosti na tom, jaká data chceme o uživatelích při registraci sbírat. Registrace do emailové služby se bude lišit od registrace na sociální síť a ta zase bude úplně jiná, než registrace pro datovou schránku. Identity management se zabývá především těmi atributy, které definují jedince.

Na základě získaných atributů nebo dat o uživateli můžeme spustit proces autorizace. V žádném případě by jejich existence neměla automaticky znamenat autorizaci samotnou. Je třeba dávat zásadní pečlivě hlídat právě ty atributy, na jejichž základě je uživateli přiděleno nějaké oprávnění. Například pokud uživatel při registrace nevyplní profesi redaktor, nebude mu pravděpodobně přiděleno oprávnění vytvářet a editovat příspěvky. [38]

3.2 Přístup

Přístup uživateli buď udělíme, nebo jej zamítneme. Jedná se o rozhodnutí ano/ne. Neexistuje žádná mezihodnota. U každého uživatele je třeba stanovit, zda konkrétní přístup k datům, systému, tiskárně povolit nebo zamítnout.

Autentizace je proces, kdy uživatel ověří svou identitu. Jsou různé způsoby, jak toho dosáhnout. Může se jednat o základní ověření jménem a heslem až po složitou, několika faktorovou autentizaci pomocí biometrických údajů. Některé metody jsem popsal v kapitole 2.3.

Jakmile je uživatel autentizován a vpuštěn do systému, nabízí se otázka, zda je na čase přidělit mu oprávnění ke službám. Odpověď zní v žádném případě. Autentizace nerovná se Autorizace. Jsou to dvě odlišné ale navzájem se doplňující techniky.

Výsledkem autentizace je kolekce jistých atributů, které jsme získali od uživatele. Budeme je tedy nazývat identita uživatele. Naproti tomu autorizace je vyhodnocení identity uživatele dle nastavených pravidel a výsledkem je buď povoleno, nebo zamítnuto.

K nastavení pravidel, dle kterých vyhodnocujeme povolení přístupu uživateli k nějakému prostředku, stanovuje autorizační politika. Ta může být implementována na lokální nebo centrální úrovni.

Uživatel během registrace vyplní svou profesi. Jeho identita pak bude obsahovat atribut účetní. Jiný uživatel zadá při registraci inženýr. Na základě pravidel autorizační politiky obdrží každý z nich jiná přístupová opatření. [39], [40]

Na závěr rekapitulace:

- Identity management spravuje sadu atributů vztaženou k uživateli
- Access management vyhodnocuje dané atributy dle pravidel autorizační politiky a přiděluje nebo zamítá přístupová oprávnění

3.3 Heslo

Heslo je pro uživatele informačních a komunikačních technologií prvotní a základní ochrannou hradbou proti případným útočníkům, a proto je třeba heslu věnovat velkou pozornost.

3.3.1 Sdílení

Nikdy své heslo nikomu nevyzrazujte ani příteli ani kolegovi v prací, nikomu. Pokud za uživatele musí udělat nějakou činnost někdo jiný, k tomu existuje zastupitelnost a je třeba

tyto funkce zástupu a delegaci rolí používat. Jakmile je heslo vyzrazeno, můžete ho pouze změnit.

3.3.2 Jedno heslo

Typická vlastnost bohužel znám z vlastní zkušenosti. Obvykle používám relativně slabé heslo pro služby, na kterých mi nezáleží. Typicky se jedná o služby, kde nefiguruje moje identita (jméno, adresa, kontaktní údaje) např. registrace kvůli vstupu na jinak důvěrný obsah, jednorázový nákup v neznámém eshopu, atd.

3.3.3 Opětovné užití hesla

Pokud bylo heslo jednou odhaleno a útočník zná kombinaci uživatelského jména a hesla, tak už logicky znovupoužití tohoto hesla v kombinaci s uživatelským jménem nebude nikdy bezpečné. Na internetu je možné stáhnout seznamy prolomených hesel. Útočníci tyto seznamy používají na tzv. slovníkový typ útoku. Jedná se de facto o útok hrubou silou (brute force) útok ale s použitím ohromného množství slov. Mezi nimi již uniklá hesla, nejčastější hesla, všechna jména v kombinaci s čísly atd. Pokud heslo používáte ve více službách a na jedné z nich dojde k jeho prolomení, je téměř nutné změnit heslo i ve službě, která ještě nebyla napadena. Tedy optimálně by uživatel měl pro každou službu používat unikátní heslo. Z vlastní zkušenosti však vím, že to není možné. [41]

3.3.4 Silná hesla

Dle nejnovějších průzkumů délka hesla naprosto zásadně ovlivňuje jeho bezpečnost. Odborníci se tedy shodují čím dál častěji na používání nejrůznějších frázích. Čím delší tím lepší. Prolomit heslo s délkou 12-14 znaků je výpočetně nesrovnatelně náročnější než heslo se 4 znaky. Délka hesla by měla být minimálně 8 znaků (doporučení 12 – 14 znaků) a mělo by se jednat o kombinace číslic, malých a velkých písmen a speciálního znaku (!, #, $, & apod.). Při tvorbě hesla je vhodné nahradit některá písmena jejich číselnými ekvivalenty. Je to vhodný kompromis, jak si heslo stále ještě pamatovat a současně zvýšit jeho odolnost. Frázi „rád chodím do lesa“ bychom mohli přepsat následovně: Radch0d!md0l3sa

3.3.5 Změna hesla

Hesla by se měla pravidelně obměňovat. Někteří odborníci doporučují obměnu optimálně po 90 dnech. Názory na časový interval výměny hesla se různí. Heslo obvykle nesmí být nápadně podobné tomu původnímu.

3.3.6 Vícefaktorová autentizace

V praxi se setkáme nejčastěji s použitím kombinace faktoru znalosti a vlastnictví. Jako příklad můžeme uvést transakci platební kartou, kde uživatel musí něco mít, tj. debetní karta, a něco znát, tj. PIN kód.

Vhodnou alternativou je mobilní aplikace, která vám vygeneruje šestimístný ověřovací PIN kód s platností nejvýše 20 sekund. Dobrým příkladem je Google Authenticator, který lze nastavit k účtům mnoha poskytovatelů. Nejrozšířenějším druhým faktorem je vlastnictví. S tímto případem se můžeme setkat u bank, kde se pro přístup k internetovému účtu musí uživatel přihlásit prostřednictvím jména nebo čísla účtu a hesla. Následně je mu na mobilní telefon zaslaná SMS s jednorázově vygenerovaným heslem, které obsahuje obvykle 6–8 číslic. [42]

3.3.7 Správce hesel

Softwarový nástroj, který dokáže generovat silná hesla za uživatele. Ta si následně uloží šifrované do centrálního uložiště dnes nejčastěji na cloud (aby měl uživatel dostupná hesla z různých zařízení). K přístupu do správce hesel slouží hlavní heslo, které opravdu nechcete ztratit! Existují bezplatné aplikace stejně jako placené. Osobně zatím nemám s takovým nástrojem větší zkušenost a narazil jsem na tuto funkci při psaní práce. Jediný správce hesel který znám je součástí webového prohlížeče Google Chrome. Pro odhalení hesel uložených v prohlížeči Google Chrome je nutné znát heslo do počítače, na kterém je uživatel ve svém Google Chrome účtu přihlášen. Bezpečnost tohoto řešení tedy nechám na čtenáři. [43]

V následující tabulce můžeme vidět seznam nejhorších mezinárodních hesel dle společnosti SplashData. Takový seznam, samozřejmě mnohem obsáhlejší, bude použit při slovníkovém útoku pro prolomení hesla. Pokud je v tabulce heslo, které používáte, útočník zcela jistě váš účet odemkne.

Tabulka 1: *Seznam nejslabších hesel 2016,* zdroj: www.splashdata.com

1. 12456	9. princess	17. flower
2. password	10. 1234	18. passw0rd
3. 12345	11. login	19. dragon
4. 12345678	12. welcome	20. sunchine
5. football	13. solo	21. master
6. qwerty	14. abc123	22. hottie
7. 1234567890	15. admin	23. loveme
8. 1234567	16. 121212	24. zaq1zaq1

II. PRAKTICKÁ ČÁST

4 AKTUÁLNÍ STAV

V této kapitole se věnuji dvěma redakčním systémům. Popisuji současné řešení. Jakým způsobem autentizují a autorizují uživatele. Poté uvádím největší nedostatky těchto přístupů.

4.1 Volba redakčních systémů

Z rozsáhlé nabídky redakčních systému jsem se rozhodl pro dva zástupce. První volba je open source CMS WordPress. Volba WordPressu vychází z mých znalostí a zkušeností s tímto systémem a pak také s počtem přes 75 milionů aktivních instalací [44] jde o nejpoužívanější open source redakční systém vůbec.

Jako druhý systém si vybral řešení od SDL Tridion. Jedná se zástupce z oblasti placených redakčních systémů. Oproti systému WordPress je robustní a hodí se spíše pro rozsáhle projekty a velké korporace.

4.2 CMS Wordpress

CMS Wordpress je redakční systém pro tvorbu a správu webových stránek. Systém je napsán v jazyce PHP a používá databázi MySQL. Pro samotné používání je nutné stáhnout současnou verzi WordPressu[19] a nainstalovat na webhosting. Pro účely práce používám čistou instalaci redakčního systému WordPress na webhostingu Wedos.

Vysvětlím, jakým způsobem redakční systém řeší autentizaci a autorizaci uživatelů, a poté uvedu z mého pohledu největší bezpečnostní rizika.

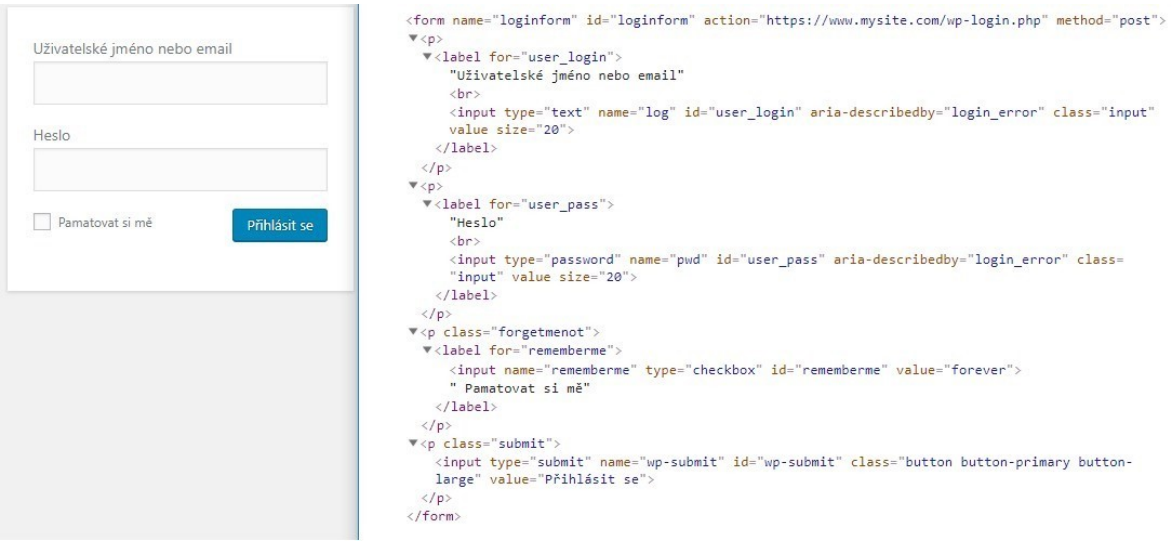
4.2.1 Autentizace

WordPress nabízí základní přihlašovací formulář s ověřením pomocí jména a hesla.

Uživatelský účet vytvoří administrátor systému v admin rozhraní. Heslo zvolí administrátor. WordPress disponuje nástrojem pro generování náhodného hesla, které je považováno za bezpečné. Vzhledem k tomu, že ve WordPressu není možnost definovat

[19] WordPress 5.2.1 https://wordpress.org/download/

žádná pravidla pro tvorbu a aktualizaci hesla, může uživatel změnit heslo a tím výrazně snížit bezpečnost uživatelského účtu.



Obrázek 20. *Standardní WordPress login form*, zdroj: vlastní

WordPress ukládá veškeré údaje o uživatelských účtech do vlastní databáze. Veškerá data o uživateli jsou šifrována pomocí algoritmu MD5. I kdyby byla databáze napadena nebo odcizena, nelze získat původní plain text.

4.2.2 Autorizace

WordPress umožňuje nastavení (konfiguraci) následujících uživatelských rolí:

- administrátor
- šéfredaktor
- redaktor
- spolupracovník
- návštěvník

Uživatelskou roli zvolí administrátor při založení účtu. Systém nedisponuje žádným autorizačním nástrojem, který by automaticky přiděloval uživatelskou roli na základě zvolené logiky tedy existence specifického parametru v profilu uživatele.

Obrázek 21. *WordPress správa uživatelů,* zdroj: vlastní

4.2.3 Rizika

Za největší bezpečnostní slabiny systému WordPress považuji:

Autentizace

Autentizace pouze pomocí jména a hesla. Chybí dnes často používaná dvou faktorová autentizace za pomoci sms nebo google authenticator aplikace v mobilním telefonu. Také nelze využít přihlášení pomocí existujícího účtu jiného IdP.

Zachycení hesla

Největším rizikem je odposlechnutí hesla v HTTP paketu, kde je heslo předáno společně s uživatelským jménem v plain textu. K úspěšnému odposlechnutí stačí být ve stejné síti jako přihlašující se uživatel a pomocí např. SW Wireshark monitorovat veškerou síťovou komunikaci. Pomocí filtru na HTTP pakety pak snadno najdeme ty, které obsahují přihlašovací sekvenci. Proto je zcela zásadní provozovat CMS WordPress na hostingu, který disponuje platným SSL certifikátem, a komunikace klient – webserver probíhá v rámci zabezpečeného HTTPS. Pokud nemůžeme zaručit používání HTTPS, nabízí se řešení přenechat registraci a přihlašování jinému IdP.

Slabá hesla

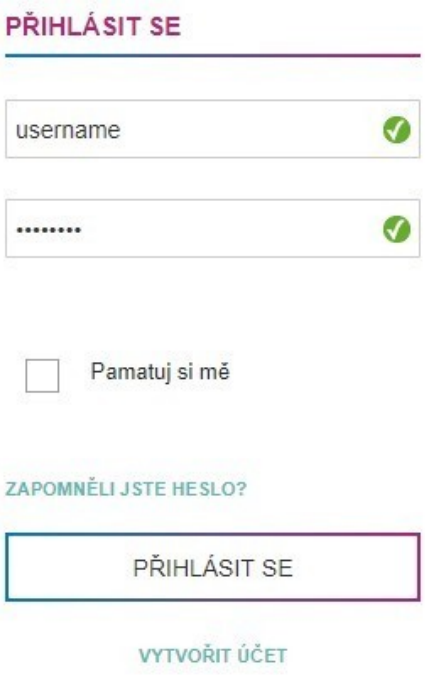
Absence nástroje pro kontrolu a stanovení pravidel pro tvorbu uživatelských hesel.

4.3 SDL Tridion

SDL Tridion je robustní redakční systém dostupný pouze v placené verzi. Uvádím jej zde z toho důvodu, že jsem s ním téměř v každodenním kontaktu.

4.3.1 Autentizace

Přihlašování a registraci do systému zajišťuje přihlašovací formulář, který je vytvořen na míru dle požadavku zákazníka. Přihlášení je implementováno pomocí Oauth 2.0 autentizačního protokolu. Identity Provider a vlastník všech uživatelských účtů je Akamai[20] Identity Cloud. Tato služba je nakoupena od externího dodavatele a pomocí nástroje Akamai Console máme plnou kontrolu nad všemi uživatelskými účty viz 4.3.2

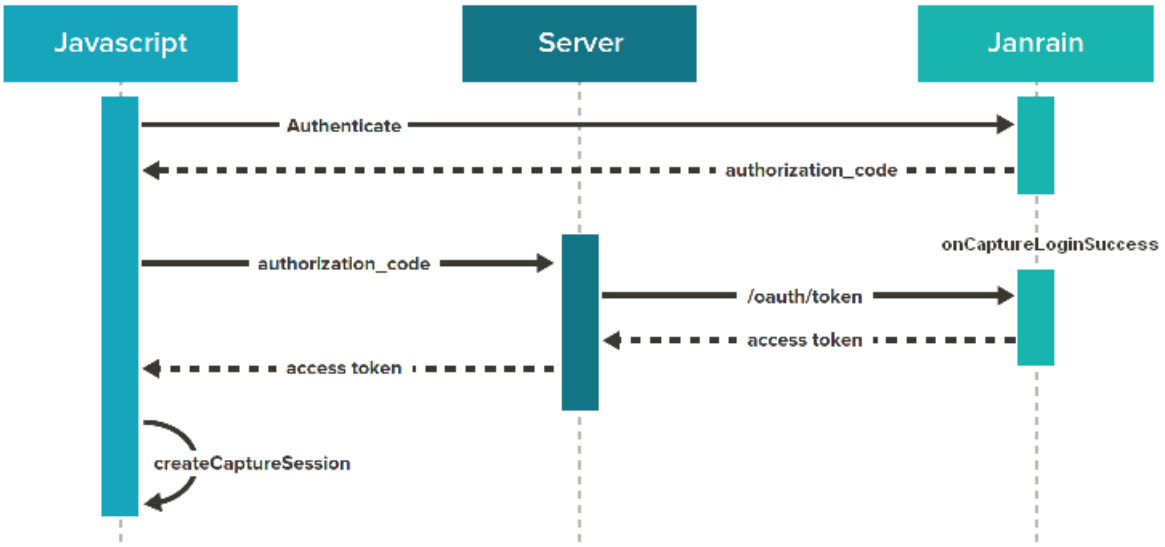


Obrázek 22. *SDL Tridion login form*, zdroj: vlastní

[20] Společnost Akamai dříve Janrain - proto se v textu bude objevovat také označení Janrain

```
<h4>Přihlásit se</h4>
<div class="capture_backgroundColor">
    <div class="capture_signin">

        {* #userInformationForm *}
        {* username *}
        {* currentPassword *}
        <p class="rememberUnameP">
            <input type="checkbox" value="checked" name="uname" id="rememberUname"><label for="rememberUname">Pamatuj si mě</label>
        </p>
        <div class="capture_form_item mot-de-passe">
            <a href="/forgotpassword.xhtml?screenToRender=forgotPassword">Zapomněli jste heslo?</a>
        </div>
        <div class="capture_rightText">{* signInButton *}</div>
        <div class="capture_form_item">
            <a name="capture_signIn_signInCreateButton" href="/registration.xhtml?screenToRender=shortRegistration" id="capture_signIn
        </div>
        {* /userInformationForm *}
    </div>
</div>
```

Obrázek 23. *Zdrojový HTML kód SDL Tridion login form*, zdroj: vlastní



Obrázek 24. *Janrain OAuth flow*, zdroj: vlastní

Důležitou událostí je onCaptureLoginSuccess. Zde již janrain login widget dokončil veškerou OAuth autentizaci a poskytuje access token. Na tomto místě je možné vytvořit AJAX volání na server, které předá access token společně s dalšími informacemi jako checkbox zapamatuj si mě a jiné.

Server následně musí ověřit access token, zda je platný. Ověření probíhá zasláním požadavku na Janrain API.

```
curl -H "Authorization: OAuth {access token}" https:// dev.janraincapture.com/entity?
attributes=["email"]
```

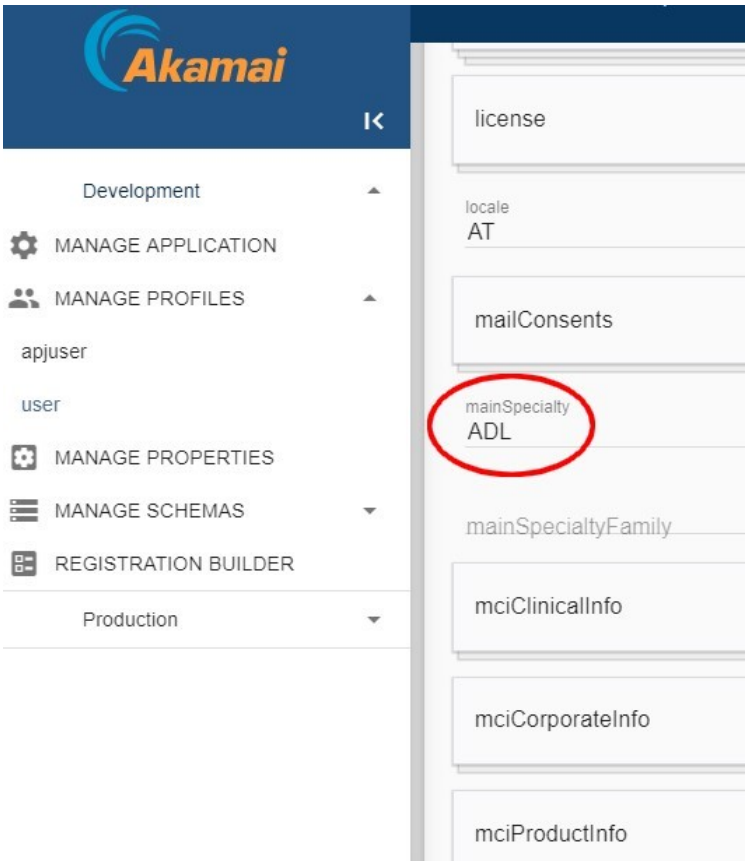
Obrázek 25. *Validace access tokenu*, zdroj: vlastní

Pokud je odpověď serveru 200 OK a vrácený JSON objekt má atribut stat: ok, pak je token validní. V opačném případě je token neplatný a může pocházet z nedůvěryhodného zdroje, tedy mohl být během komunikace podvrhnut. Janrain API vrátí vždy 200 OK, ale v případě, že token není pravý, bude v atributu stat: error. Tato kontrola je zcela zásadní.

Jakmile je access token ověřen, server vytvoří platnou uživatelskou relaci. Výchozí délka platnosti uživatelské relace je nastavena na 60 minut. Pokud je zaškrtnut checkbox zapamatuj si mě, je platnost prodloužena až na 90 dnů.

4.3.2 Autorizace

O autorizaci se stará Akamai IAM. Na základě atributů získaných při registraci uživatele přiděluje dle autorizační politiky různé úrovně oprávnění pro jednotlivé uživatele.



Obrázek 26. *Akamai console editace uživatele,* zdroj: vlastní

4.3.3 Požadavky

Implementovat tzv. social login, tedy aby přihlašovací formulář podporoval registraci a přihlášení pomocí třetí strany jako je Facebook, Google.

5 NALÝZA MOŽNOSTÍ ZABEZPEČNÉHO PŘÍSTUPU

Hlavním cílem práce je najít nejvhodnější způsob, jak zajistit co možná nejbezpečnější přihlašovací mechanismus. Budu zde vycházet ze znalostí nabytých v teoretické části práce. Nejprve se budu zabývat systémem WordPress a poté přijde na řadu SDL Tridion.

5.1 CMS WordPress

Vycházím z kapitoly 4.2.3 a představím postupně několik možných řešení, jak odstranit uvedená rizika.

Varianta 1

Zabezpečit web pomocí HTTPS. Instalace certifikátu SSL na webhosting. Otestovat funkci stránek na protokolu HTTPS a opravit případné chyby. Zkontrolovat stránky tak, aby se veškeré stránky a skripty načítaly z HTTPS protokolu. Nakonec nastavit přesměrování z HTTP na HTTPS.

Implementace Okta Identity Provider. Okta umožňuje vytvářet, editovat a spravovat uživatele a jejich data a spojit je s jednou nebo více aplikacemi, v našem případě jedna instance CMS WordPress. Toto řešení získá na hodnotě, jakmile založíme další web a budeme chtít, aby se všichni uživatelé webu A mohli přihlásit i do webu B. Pokud na webu B implementujeme také Okta IdP a nastavíme stejný Client ID a Client secret, vytvoříme vlastně federaci mezi těmito weby.

Okta nabízí vlastní login formulář, který můžeme snadno upravit a vložit do systému. Pro správné fungování je třeba registrovat

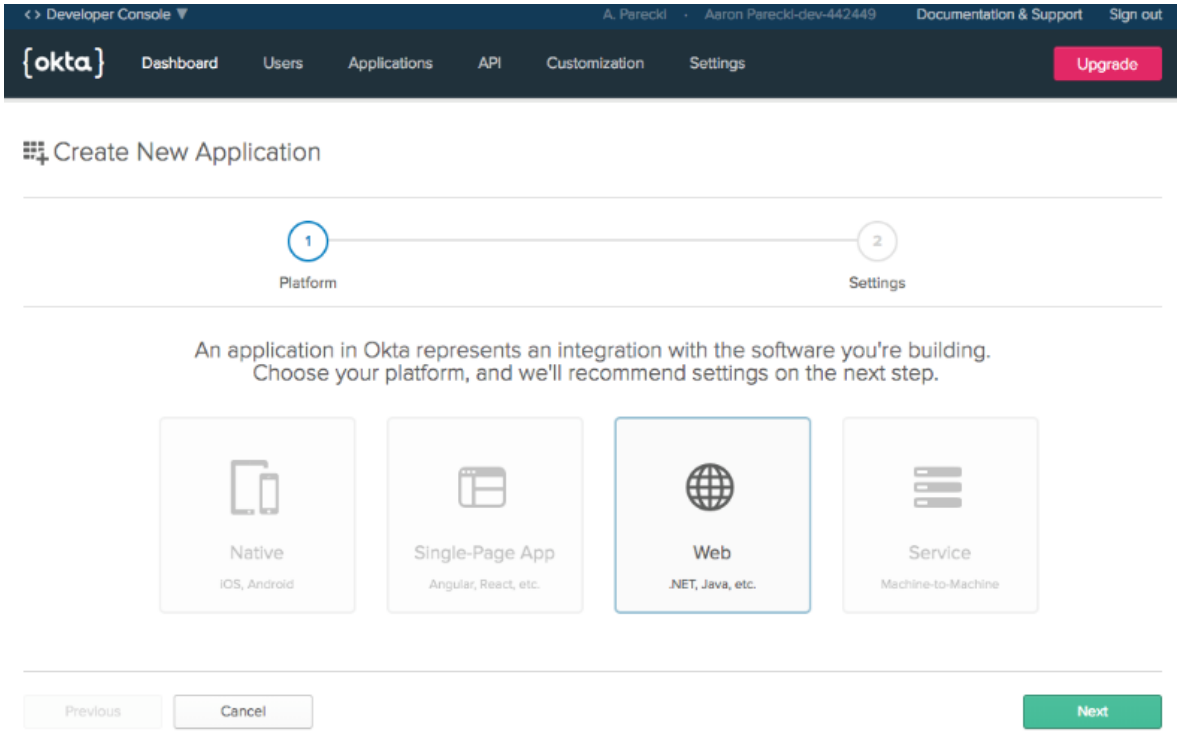
Okta developer účet:

https://developer.okta.com/signup/

Okta WordPress plugin:

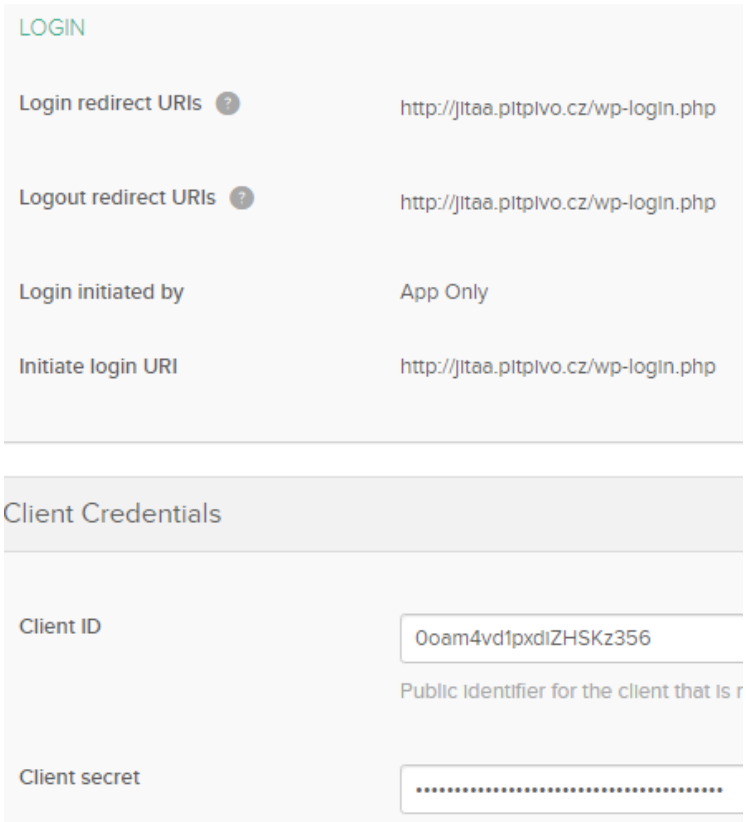
https://github.com/oktadeveloper/okta-wordpress-sign-in-widget

- plugin vložíme do instalace WordPressu do složky wp-plugins
- v Okta developer účtu založíme novou aplikaci typu Web



Obrázek 27. *Okta nastavení,* zdroj: vlastní

- dále je třeba vyplnit Base URI, redirect URI a login URI



Obrázek 28. *Okta* nastavení 2, zdroj: vlastní

- client ID a client secret zkopírujeme do konfiguračního souboru plugin env.php. Jako base URL slouží *https://dev-281598.okta.com*

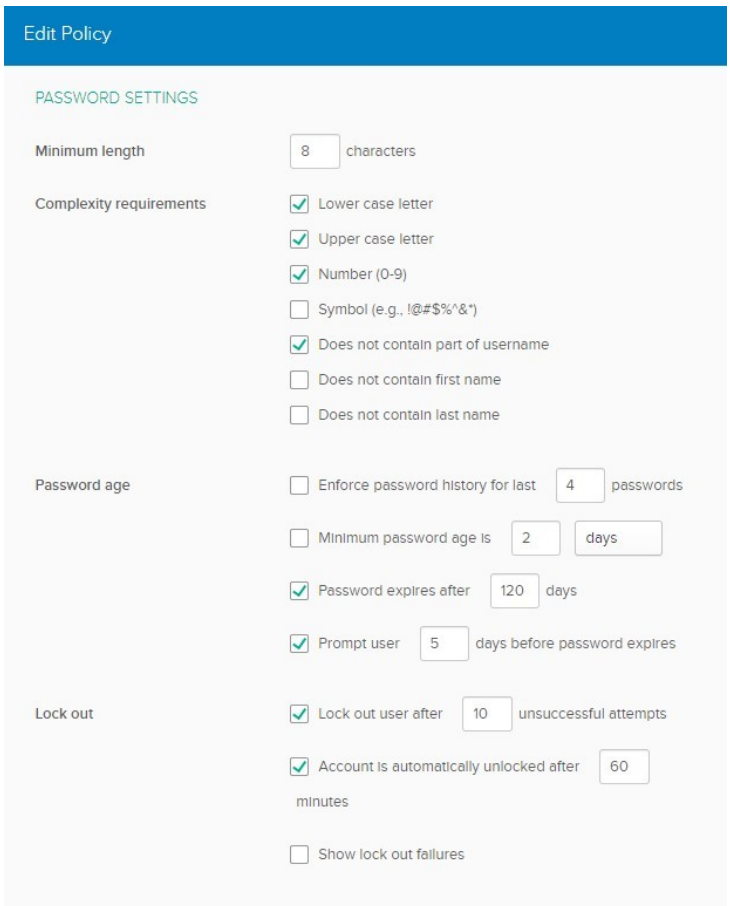
```php
<?php
define('OKTA_BASE_URL', 'https://dev-281598.okta.com');
define('OKTA_CLIENT_ID', '0oam4vd1pxdiZHSKz');
define('OKTA_CLIENT_SECRET', 'rCEvDdi_odmY0GfeNs5q-9atIYOQRZALe');

# If you're using API Access Management, define the auth server ID below.
# Otherwise leave it commented out.
# define('OKTA_AUTH_SERVER_ID', 'default');
?>
```

Obrázek 29. *Okta nastavení env.php,* zdroj: vlastní

- v CMS WordPress nyní aktivujeme plugin *Okta Sign-in Widget*

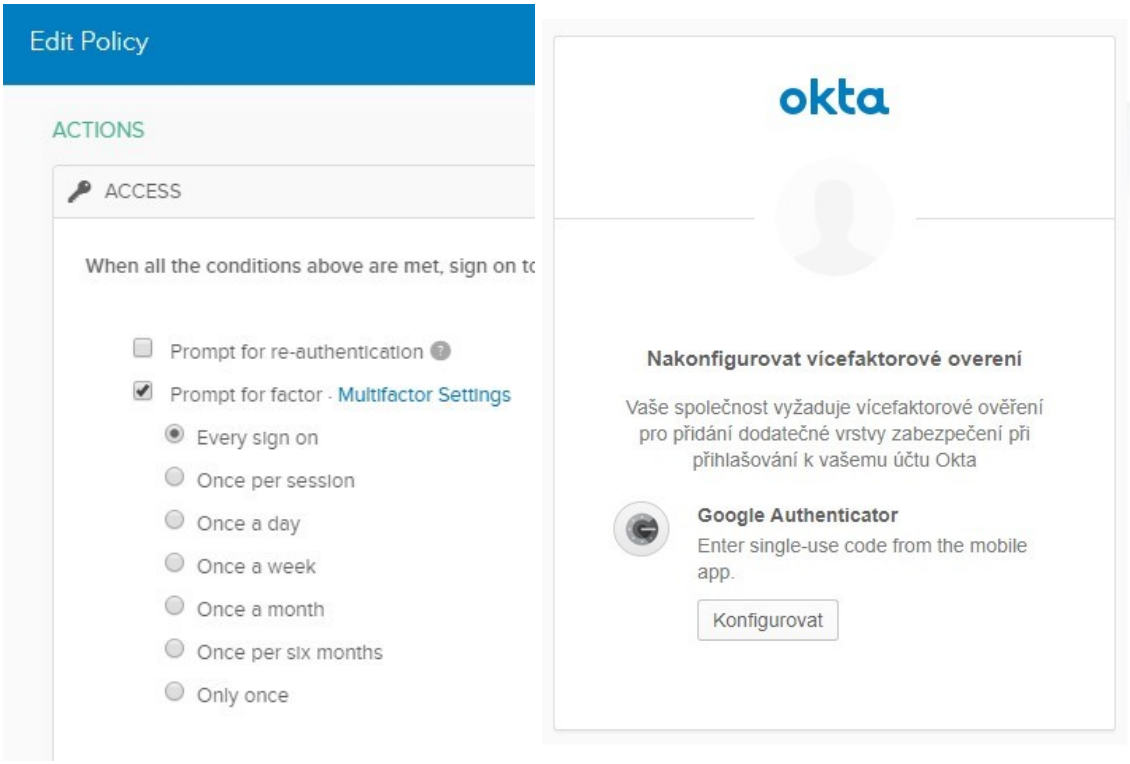
V tuto chvíli již najdeme Okta formulář na přihlašovací straně wp-login.php.

Bezpečnost hesel udržuje v databázi u sebe, odpadá nutnost držet hesla ve WordPress databázi. Nastavení bezpečnosti hesel můžeme pohodlně administrovat v sekci Security -> Authentication



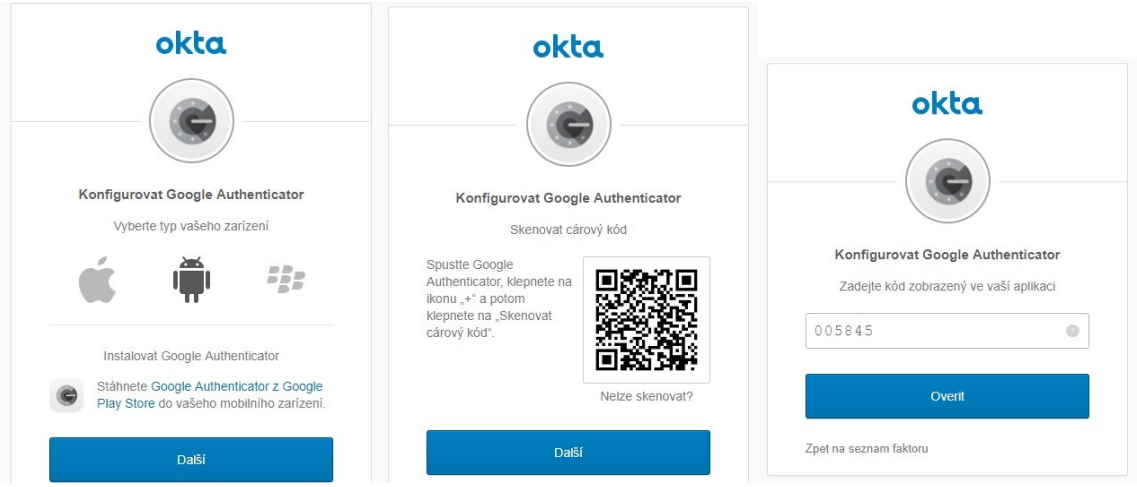
Obrázek 30. *Okta nastavení bezpečnostní politiky hesel,* zdroj: admin.okta.com

Dalším přínosem používání Okta tools je snadná konfigurace více faktorové autentizace. V sekci Security -> Multifactor -> Google Authenticator nastavíme službu jako aktivní. Následně přidáme pravidlo pro login. V sekci Aplications -> zvolíme naší aplikaci -> Sign On -> Sign On Policy a přidáme nové pravidlo .



Obrázek 31. *Okta nastavení MFA 1,* zdroj: admin.okta.com

Alternativa ke službě Okta je Auth0 *https://auth0.com/wordpress* nabízí velmi podobný rozsah služeb a nastavení. Vyzkoušené mám však řešení od Okta developers.



Obrázek 32. *Okta nastavení MFA 2,* zdroj: admin.okta.com

Varianta 2

Úprava vestavěného WordPress přihlašovacího formuláře. Pro tuto úpravu je nutná detailní znalost fungování CMS WordPress. Redakční systém má velmi dobře zpracovanou dokumentaci, viz. *https://usersinsights.com/wordpress-user-login-hooks/*

Samotná úprava vestavěného přihlášení systému WordPress by vedla do útrob systému do komponenty pluggable.php. Zde existuje mnoho možností, jak obsluhovat nejenom přihlašování. Na obrázku vlastní filtr na metodu authenticate. V těle metody my_auth je prostor pro změnu autentizace. Filtry v CMS WordPress přepisují chování původních metod. [45]

```php
add_filter( 'authenticate', 'my_auth', 10, 3 );

function my_auth( $user, $username, $password )
{
    // My functionality
    return $user;
}
```

Obrázek 33. *WordPress metoda authenticate*, zdroj: vlastní

Varianta 3

Implementace vlastního přihlašovacího formuláře úplně od začátku. Nedoporučuji, budeme vymýšlet postup od začátku, přestože existuje mnoho použitelných implementací. Vlastní vývoj může najít uplatnění ve speciálních případech, kdy budeme chtít použít jiný hashovací algoritmus pro ukládání hesel v databázi nebo uvnitř firemní zabezpečené sítě

budeme z nějakého důvodu potřebovat vynechat šifrování hesel úplně na úkor rychlosti a jednoduchosti implementace. Jednoduché ověření přístupu Autentizace2.3. Na úkor bezpečnosti tak získáme velice rychlý přihlašovací mechanizmus, který ovšem není zabezpečený.

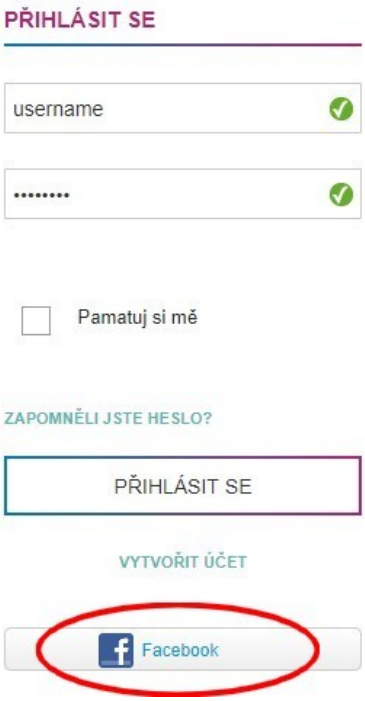
V každém případě, ať už vývoj přihlašovacího formuláře ve formě vlastního pluginu nebo úpravě toho defaultního, je pro takto malý projekt (1 instalace) finančně i časově nerentabilní. V tomto případě zvolíme variantu číslo 1 a implementujeme Okta plugin, který za nás vyřeší autentizaci, bezpečnostní politiku hesel a přidá Facebook a Google jako IdP.

5.2 SDL Tridion

Navazuji na kapitolu 4.3.3 a budu se věnovat plánovanému rozšíření registračního a přihlašovacího formuláře.

Varianta 1

Rozšíření současného řešení na míru o možnost registrace a přihlášení pomocí Facebooku. Požadovaný výsledný stav.



Obrázek 34. *SDL Tridion login form,* zdroj: vlastní

Úprava html formuláře, rozšíření o Facebook login tlačítko.

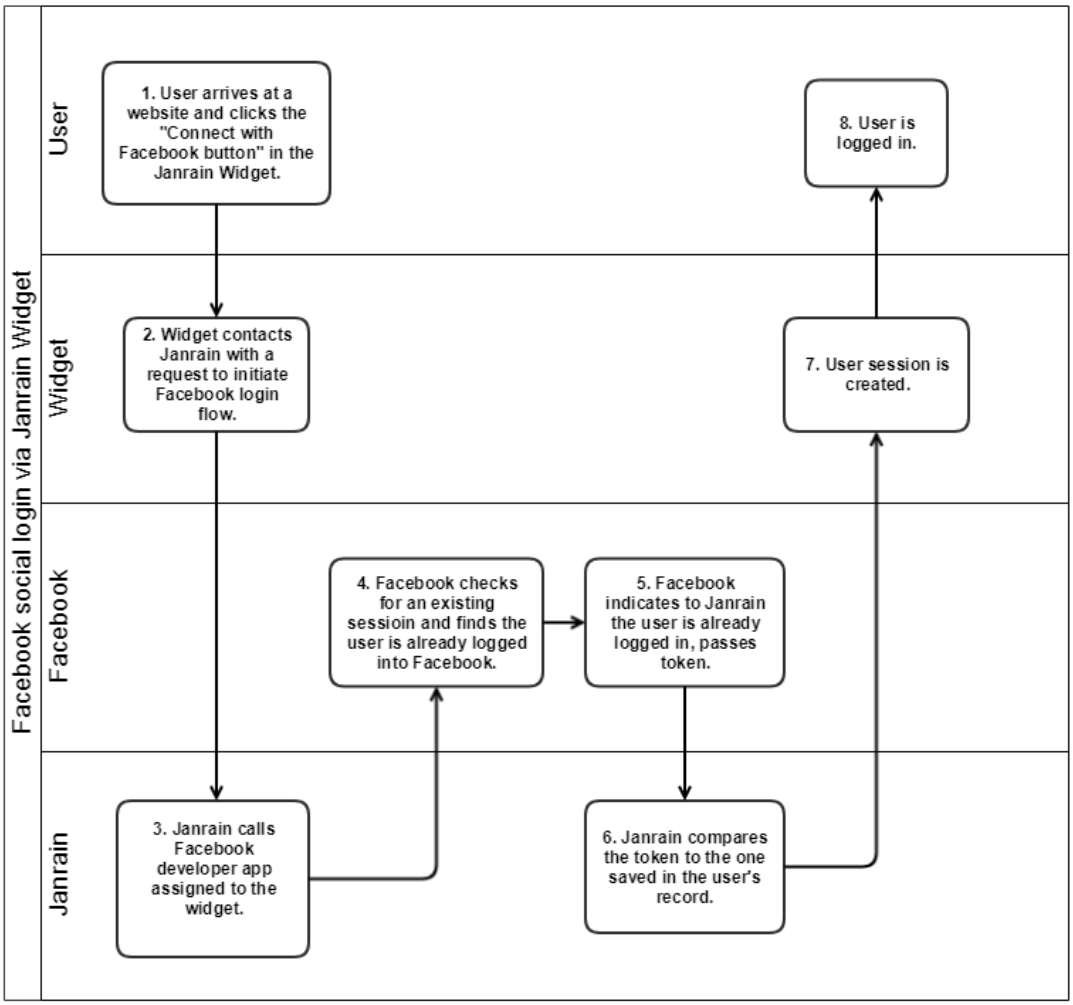
```
<!-- Facebook button code START -->
<a class="social-login-button"
href="javascript:void(0);"
onclick="janrain.engage.signin.triggerFlow('facebook');">
    <li id="janrain-facebook"
        role="button">
        <div class="janrain-provider-container"
            tabindex="1"
            href="javascript:void(0);">
            <span class="janrain-provider-i con-24 janrain-provider-icon-facebook"></span>
            <span class="janrain-provider-text-color-facebook">Se connecter avec <strong>Facebook</strong></span>
        </div>
    </li>
</a>
<!-- Facebook button code END -->
```

Obrázek 35. *HTML prezentace Facebook tlačítka,* zdroj: vlastní

Chování tlačítka bude vykonávat pomocný soubor facebookHelper.js umístěný na webu, tedy v systému Tridion.

```
function janrainCaptureWidgetOnLoad() {
    {
        janrain.settings.providers = ['facebook'];
    }
    function setWindowForRegistration(result) {
        var jqLightRegistration = $('[data-light-registration]');
        if (jqLightRegistration.length == 0) {
            //var redirectUrl = "/thankyouregistration.xhtml";
            var redirectUrl = "";
            if (!!urlParam('lvUrl')) {
                redirectUrl = window.location.search.split('&lvUrl=').pop();
            }

            }
            // Facebook redirect after register.
            if (result.authProvider == "facebook") {
                redirectUrl = top.window.location.href;
            }
            janrain.capture.ui.modal.close();
            if (window.console && window.console.log) console.log(result);
            var localvars = JSON.parse(localStorage.janrainCaptureProfileData || '{}');
            var paramMap = {
                "uuid": localvars.uuid,
                "email": localvars.email,
                "name": localvars.displayName,
                "captureToken": localStorage.janrainCaptureToken,
                "redirectto": redirectUrl
            };
            createUserSession("/jumpcontroller", paramMap);
        }
    janrain.events.onCaptureForgotPasswordCodeSuccess.addHandler(function (result) {
        if (result.accessToken != "" && typeof result.accessToken != "undefined") {
            //janrain.capture.ui.createCaptureSession(result.accessToken);
        }
    });
    janrainAccessToken = "";
    facebookLogin = "false";
    janrain.events.onCaptureSaveSuccess.addHandler(function (result) {
        if (result.form == "dataMissingForm" && result.statusMessage == "signedIn") {
            if (result.accessToken != "" && typeof result.accessToken != "undefined") {
                janrain.capture.ui.createCaptureSession(result.accessToken);
            }
            janrain.capture.ui.modal.close();
        }
        if (result.action == "socialSignin" && result.authProvider == "facebook") {
            janrainAccessToken = result.accessToken;
            facebookLogin = "true";
        }
    });
}
```

Obrázek 36. *komponenta facebookHelper.js,* zdroj: vlastní



Obrázek 37. *Facebook social login flow,* zdroj: vlastní

Varianta 2

Implementace hotového řešení třetí strany. Řešení, které může zprvu působit jednodušeji, ale ve skutečnosti to tak není. Přihlašovací formulář, jak je zobrazen na obrázku číslo 22, je implementován na mnoha místech. Při implementaci nového řešení bychom museli všechny výskyty současného formuláře podrobit analýze, zda nové řešení bude v konkrétní publikaci taktéž funkční.

Vzhledem k tomu, že celá problematika registrace a přihlášení je od začátku programována na míru interně, bude i implementace Facebook loginu provedena interně dle představené varianty 1.

6 BEZPEČNOST CMS

Kapitola obecně o bezpečnosti CMS. Jaká pravidla se vyplatí dodržovat při provozování jakéhokoliv CMS.

6.1 SSL certifikát

Zabezpečit komunikaci mezi webovým serverem a webovým prohlížečem na straně klienta by měl každý majitel e-shopu a dalších stránek, kterým návštěvníci sdělují citlivá data. V případě HTTPS nelze komunikaci odposlechnout, resp. nelze zjistit přenášený obsah.

SSL certifikát lze pořídit podobně jako digitální podpis u certifikační autority. Prodejců na českém trhu je mnoho a ceny se odvíjí od důvěryhodnosti autority. [46]

6.2 Aktualizace

Zlaté pravidlo zní mít redakční systém neustále aktuální. Tedy jakmile vyjde nová verze systému, je třeba okamžitě nainstalovat. Dobrým zvykem je před každou aktualizací redakční systém pro jistotu zálohovat. Nikdy nevíte, co se může stát a záloha pomůže s obnovou. Co se zálohování týče, to je třeba také dělat pravidelně a ne pouze před aktualizací systému. Obecně lze říci, že frekvence zálohování je na každém provozovateli. Pokud na webu vznikne během týdne mnoho nového obsahu, bude vhodné nastavit týdenní zálohování. Pokud se na webu neobjeví nový obsah celý měsíc, postačí interval zálohy i delší.

6.3 Kontrola pluginů

Stejně jako je nutné aktualizovat redakční systém samotný, je třeba aktualizovat i používané pluginy. Čas od času také zkontrolovat jejich funkčnost, a jakmile plugin přestaneme používat nebo jej nahradíme jiným, je třeba starý deaktivovat a posléze ze systému odstranit.

6.4 Silná hesla

Odkážu se na kapitolu 3.3.4 Silná hesla. Je třeba nastavit silné heslo pro administrátorský přístup do systému, databáze ale i na webhosting. Pokud možno využít protokol SFTP[21]. Stejně tak je třeba vyžadovat používání silných hesel i po uživatelích systému. Vhodným řešením může být plugin, pomocí kterého nastavíme základní politiku hesel.

6.5 Webhosting

Vedle rizik spojených s redakčními systémy je třeba minimalizovat i rizika spojená s hostingem. Poskytovatel hostingu může totiž zásadně ohrozit vaše stránky. Zeptejte se svého poskytovatele, jakou ochranu vašemu webu zajišťuje a jak je připraven na útoky a nečekané události.

6.6 Užitečné pluginy

Existuje velké množství pluginů, které se zabezpečením redakčního systému pomohou. Je třeba vědět, co budeme zabezpečovat a na co se zaměřit.

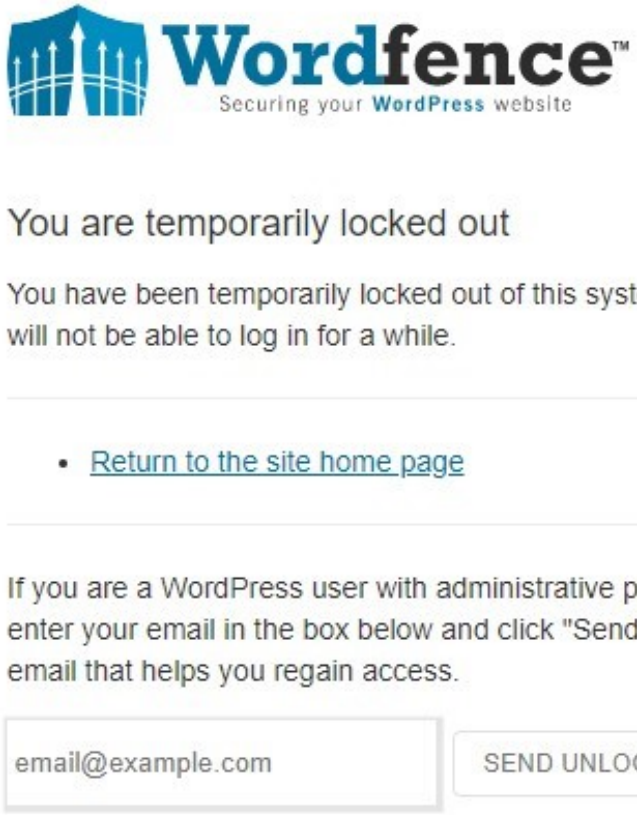
WordFence

Nejpopulárnější řešení pro zabezpečení WordPress stránek. WordFence má v sobě přímo integrovaný základní WAF[22] a nabízí i skenování webu v reálném čase. Automaticky blokuje pokusy o napadení webu, uhodnutí hesla hrubou silou, či infikování škodlivými soubory. Jde o komplexní řešení, které je vhodným doplňkem každého webu.

Plugin umožňuje zabezpečit web před útokem hrubou silou. Po několika neúspěšných pokusech o přihlášení WordFence zablokuje IP adresu, ze které se uživatel snaží přihlásit a odešle informační email administrátorovi.

[21] Z angličtiny Secure File Transfer protocol, komunikace je šifrovaná, nehrozí czizení jména a hesla v paketu

[22] Z angličtiny Web Application Firewall



Obrázek 38. *WordPress plugin Wordfence blokace přístupu*, zdroj: vlastní

Obrázek 39. *WordPress plugin Wordfence emailové upozornění*, zdroj: vlastní

ZÁVĚR

Hlavním cílem práce bylo vyhodnotit bezpečnost přihlašování uživatelů do informačního systému a navrhnout možná zlepšení. Teoretická část měla za úkol seznámit čtenáře s informačním systémem a jeho funkcemi a typickým použitím. Dále zavedla čtenáře mezi redakční systémy, problém autentizace a autorizace a v závěru teorie jsem se věnoval Identity a Access Managementu a bezpečnosti hesel.

Teoretická část měla za úkol představit a definovat principy a techniky, které jsou využívány v praktické části. Čtvrtá kapitola analyzuje současný stav dvou redakčních systémů. Zaměřuje se pouze na problematiku autentizace a autorizace uživatele. Výstupem je sada slabin a možných rizik, na které jsou v páté kapitole nabídnuta možná řešení. Poslední šestá kapitola se věnuje bezpečnému provozu redakčního systému a jsou zde uvedena obecná doporučení.

Tento materiál je podložen jak znalostmi z teoretické části práce tak i mými zkušenostmi z oblasti redakčních systémů a webových technologií. Klíčovým zdrojem informací pro tuto práci byl internet. Avšak podklady jsou čerpány jak z mnohých elektronických článků, tak i z knih.

Během psaní práce jsem se seznámil s mnoha důležitými informacemi a postupy, které se týkají bezpečnosti na internetu, ať už se jednalo o zabezpečení přístupu, ověření uživatele, správa a generovaní bezpečných hesel, autentizační a autorizační standardy. Několikrát jsem během svého výzkumu narazil na techniky prolamování hesel a získávání přístupu do webových služeb. To by jistě bylo vhodné téma pro další práci. V průběhu psaní práce jsem zdokonalil přihlašovací formulář na dvou webech, které jsem vytvořil, a zdokonalil zásady tvorby a změny hesla.

Předpokládám, že tato práce bude pro čtenáře přínosem, protože mně již samotné psaní práce velmi pomohlo, kdy jsem teoretické předpoklady mohl prakticky ověřit, doplnit a následně vybrat optimální řešení.

Závěrem je třeba dodat, že se jedná o velice důležitou a neustále se vyvíjející problematiku. Je otázkou času, kdy informace a doporučení v této práci zastarají, ztratí svou účinnost a budou překonána. Proto je třeba vývoj na poli webových bezpečnostních technologií neustále sledovat.

SEZNAM POUŽITÉ LITERATURY

[1] *Příruční slovník naučný*. Praha: Academia, 1964.

[2] WIENER, Norbert. *Kybernetika a společnost*. Praha: Academia, 1963. ISBN 21-030-63.

[3] SODOMKA, Petr. *Informační systémy v podnikové praxi*. Vyd. 1. Brno: Computer Press, 2006. ISBN 80-251-1200-4.

[4] CMS. *The Tech Term Computer Dictionary* [online]. 2013 [cit. 2019-02-13]. Dostupné z: http://www.techterms.com/definition/cms/

[5] FIŠER, Jakub. Proč používat redakční systém. *Programujte* [online]. 2005 [cit. 2019-01-13]. Dostupné z: http://programujte.com/clanek/2005110801-proc-pouzivat-redakcni-system/

[6] Usage of content management systems for websites. In: *W3Techs* [online]. 2019 [cit. 2019-02-10]. Dostupné z: https://w3techs.com/technologies/overview/content_management/all

[7] Content Management System. In: *MuxSoft Technology* [online]. 2014 [cit. 2019-01-13]. Dostupné z: http://muxsofttech.com/cms

[8] WordPress Versions. *WordPress.org* [online]. 2019 [cit. 2019-05-07]. Dostupné z: https://codex.wordpress.org/Current_events

[9] History. *WordPress.org* [online]. 2015 [cit. 2019-03-13]. Dostupné z: https://codex.wordpress.org/History

[10] Joomla CMS. In: *Joomla.org* [online]. 2019 [cit. 2019-02-13]. Dostupné z: https://www.joomla.org/

[11] Co je Joomla?!. *JoomlaPortal.cz* [online]. 2018 [cit. 2019-03-13]. Dostupné z: http://www.joomlaportal.cz/index.php/clanky-a-novinky/zaciname-s-cms-joomla/493-bart#Technicke_pozadavky

[12] Drupal logo blue. In: *Drupal.org* [online]. 2018 [cit. 2019-02-24]. Dostupné z: https://drupal.org/drupal-media-kit

[13] History. *Drupal.org* [online]. 2018 [cit. 2019-02-17]. Dostupné z: https://drupal.org/about/history

[14] SharePoint logo. In: *SharePoint* [online]. 2019 [cit. 2019-02-24]. Dostupné z: https://products.office.com/cs-cz/sharepoint

[15] Microsoft Sharepoint. *Weblinx* [online]. 2019 [cit. 2019-04-14]. Dostupné z: https://www.weblinxinc.com/blog/is-microsoft-sharepoint-a-good-cms/

[16] Microsoft SharePoint Through the Years. *Fallsdigital* [online]. 2018 [cit. 2019-03-14].

[17] SDL logo. In: *SDL.com* [online]. 2019 [cit. 2019-03-04]. Dostupné z: https://www.sdl.com/Content/themes/Responsive/public/images/logo-corporate.svg

[18] What is SDL Tridion. *Quora* [online]. 2016 [cit. 2019-03-04]. Dostupné z: https://www.quora.com/What-is-CMS-What-is-SDL-Tridion-Where-can-I-find-nice-CMS-tutorials-for-a-beginner

[19] SDL Tridion R5. *Cmswire.com* [online]. 2008 [cit. 2019-04-24].

[20] JAŠEK, Roman. *Informační a datová bezpečnost*. Zlín: Univerzita Tomáše Bati ve Zlíně, 2006. ISBN 80-731-8456-7.

[21] Kybernetická bezpečnost: Co s tím?. *Businessinfo.cz* [online]. 2016 [cit. 2019-03-27]. Dostupné z: https://www.businessinfo.cz/cs/clanky/kyberneticka-bezpecnost-co-s-tim-84467.html

[22] Model PDCA. In: *Risk Analysis Consultants* [online]. 2018 [cit. 2019-03-27]. Dostupné z: http://www.iso27000.cz/rac/homepage.nsf/CZ/27001

[23] Kybernetická bezpečnost. *Cybersecurity.cz* [online]. 2017 [cit. 2019-03-27].

[24] Password-less sign-in. *Www.microsoft.com* [online]. 2018 [cit. 2019-04-28].

[25] HTTP Basic auth. *Www.kutac.cz* [online]. 2017 [cit. 2019-04-21].

[26] The Best Ways of Authentication. *JetRuby.com* [online]. 2018 [cit. 2019-03-29].

[27] Implementing Security Access Control (SAC). *AgileData.com* [online]. 2019 [cit. 2019-03-29].

[28] Federace identit aneb spolčení totožností. *Http://webserver.ics.muni.cz* [online]. 2011 [cit. 2019-04-21].

[29] What is federation? And how is it different from SSO?. *Blog.empowerid.com* [online]. 2012 [cit. 2019-04-21].

[30] Identity Provider. *Www.shibboleth.net* [online]. 2018 [cit. 2019-04-22].

[31] Service Provider. *Www.shibboleth.net* [online]. 2018 [cit. 2019-04-22].

[32] Shibboleth. *Www.internet2.edu* [online]. 2017 [cit. 2019-05-22].

[33] Choosing an SSO Strategy. *Www.mutuallyhuman.com* [online]. 2013 [cit. 2019-04-22].

[34] SAML vs. OAuth: Which One Should I Use?. *Dzone.com* [online]. 2013 [cit. 2019-03-22].

[35] ULTIMATE GUIDE TO SSO. *Pingidentity.com* [online]. 2019 [cit. 2019-03-22].

[36] FIDO IS NOT THE END OF PASSWORDS. *Pingidentity.com* [online]. 2014 [cit. 2019-03-22].

[37] Integrating FIDO Authentication & Federation Protocols. *Slideshare.net* [online]. 2018 [cit. 2019-03-22].

[38] Identity Management vs Access Management. *Www.vintegris.tech* [online]. 2018 [cit. 2019-04-21].

[39] The Difference Between Identity Management and Access Management. *Www.globalsign.com* [online]. 2016 [cit. 2019-04-10].

[40] What is Identity and Access Management and Why is it a Vital IT Security Layer?. *Www.beyondtrust.com* [online]. 2018 [cit. 2019-05-21].

[41] 8 Tips to Make Your Passwords as Strong as Possible. *Mentalfloss* [online]. 2017 [cit. 2019-03-23].

[42] Bezpečnostní politika hesel a vícefaktorová autentizace. *Systemonline* [online]. 2016 [cit. 2019-03-23].

[43] Heslo. *Internetembezpecne* [online]. 2016 [cit. 2019-03-23].

[44] 25 Astonishing WordPress Facts That Will Blow Your Mind. *Whoishostingthis* [online]. 2019 [cit. 2019-04-22].

[45] A visual guide to WordPress user login hooks. *Usersinsights* [online]. 2018 [cit. 2019-05-23].

[46] Jak snadno a rychle zabezpečit web pomocí SSL certifikátu?. *Blog.buchtic.net* [online]. 2015 [cit. 2019-03-23].

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

API	Rozhraní pro programování aplikací (zkratka pro Application Programming Interface)
CMS	Systém pro správu obsahu (Content management system)
CSS	Cascading Syle Sheet, kaskádové styly znamenají souhrn metod pro úpravu vzhledu HTML stránky
Framework	Framework je softwarová struktura, která slouží jako podpora při programování a vývoji a organizaci jiných softwarových projektů
GNU	GNU's Not Unix operační systém vyvíjen pod volnou licencí, který může ožívat různá jádra (Linux, Hurd, Solaris). GNU je také název společnosti, která OS vyvíjí. Jejich produktem je také GPL
GNU GPL	General Public License je licence pro svobodný software. Veškerá odvozená díla automaticky přebírají původní licenci
HTTP	Hyper Text Transfer Protocol je internetový protokol určený pro komunikaci s WWW servery
HTTPS	Hyper Text Transfer Protocol Secure je internetový protokol určený pro zabezpečenou komunikaci s WWW servery
ISMS	Systém řízení bezpečnosti informací (Information Security Management System)
IS	Informační systém
Javascript	Nejpoužívanější skriptovací jazyk na internetu. Jeho pomocí je tvořen dynamický obsah na internetových stránkách
Open Source	Otevřený zdrojový kód. Zdrojový kód s touto licencí je volně dostupný a kdokoliv ho může dále používat a vylepšovat

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

REJSTŘÍK

PŘÍLOHA P I: WORDPRESS UŽIVATELSKÉ ROLE

Permissions	Administrator	Editor	Author	Contributor	Subscriber
Read Site	✔	✔	✔	✔	✔
Edit Posts	✔	✔	✔	✔	✖
Delete Posts	✔	✔	✔	✔	✖
Publish Posts	✔	✔	✔	✖	✖
Delete Published Posts	✔	✔	✔	✖	✖
Edit Published Posts	✔	✔	✔	✖	✖
Upload Files	✔	✔	✔	✖	✖
Publish Pages	✔	✔	✖	✖	✖
Delete Pages	✔	✔	✖	✖	✖
Edit Other's Posts & Pages	✔	✔	✖	✖	✖
Delete Other's Posts & Pages	✔	✔	✖	✖	✖
Read Private Pages	✔	✔	✖	✖	✖
Edit Private Pages	✔	✔	✖	✖	✖
Delete Private Pages	✔	✔	✖	✖	✖
Manage Categories	✔	✔	✖	✖	✖
Moderate Comments	✔	✔	✖	✖	✖
Activate Plugins	✔	✖	✖	✖	✖
Create Users	✔	✖	✖	✖	✖
Delete Plugins	✔	✖	✖	✖	✖
Delete Themes	✔	✖	✖	✖	✖
Delete Users	✔	✖	✖	✖	✖
Edit Files	✔	✖	✖	✖	✖
Edit Plugins	✔	✖	✖	✖	✖
Edit Theme Options	✔	✖	✖	✖	✖
Edit Themes	✔	✖	✖	✖	✖
Edit Users	✔	✖	✖	✖	✖
Export Content	✔	✖	✖	✖	✖
Import Content	✔	✖	✖	✖	✖
Install Plugins	✔	✖	✖	✖	✖
Install Themes	✔	✖	✖	✖	✖
See All Users	✔	✖	✖	✖	✖
Manage Site Options	✔	✖	✖	✖	✖
Promote Users	✔	✖	✖	✖	✖
Remove Users	✔	✖	✖	✖	✖
Switch Themes	✔	✖	✖	✖	✖
Update Core	✔	✖	✖	✖	✖
Update Plugins	✔	✖	✖	✖	✖
Update Themes	✔	✖	✖	✖	✖
Edit Dashboard	✔	✖	✖	✖	✖

PŘÍLOHA P II: SDL TRIDION UŽIVATELSKÉ ROLE

Role Name	Description
ROLE_USER	Registered Users, Validation Pending
ROLE_OWNER	Non-Autologin Users
ROLE_MEMBER	Validated Healthcare Provider
ROLE_MRK_EMP	Merck Employee
ROLE_SSO_USER_VP_FP	SSO Valid FullProfile Authority
ROLE_SSO_USER_VP_NP	SSO Valid NoProfile Authority
ROLE_SSO_USER_PV_FP	SSO pending Full profile Authority
ROLE_SSO_USER_PV_NP	SSO Pending No Profile Authorities
ROLE_ANONYMOUS	Unauthenticated user (Validation status Rejected)
ROLE_HCP	Validated Healthcare Provider: Physician
ROLE_MDM_MEMBER	Merck Master Data Manager User login (for icods if validation status is 50) (for janrain validation status is mdmvalidated)
ROLE_REJECTED	Validation Status Rejected: Based on validation status for icods(status=30)
ROLE_USER_VAL	Validated Single Sign On, Profile Sharing Enabled
ROLE_USER_VAL_PNS	Validated Single Sign On, Profile Sharing Disabled
ROLE_ADMIN	Administrator
ROLE_BD	Third-Party Agency
ROLE_TEMP_JANRAIN	Implanon User, Terms of Use Acceptance Pending
ROLE_TEMP_USER	Impanon User, Password Change In-Progress
ROLE_PRE_REG_USER	Pre-registered User
ROLE_VEEVA_MDM	mdm veeva auto login
CAMPAIGN_USER	campaign auto login
ROLE_TRANSIENT	transient Users

PŘÍLOHA P III: WORDPRESS & OKTA PLUGIN

wordpress-5.1.1-cs_CZ.zip

okta-wordpress-sign-in-widget-master.zip