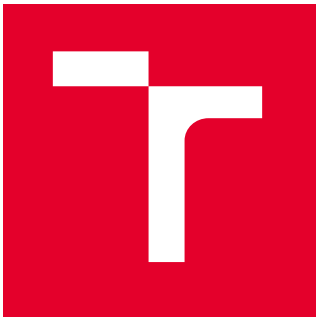


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

LABORATORNÍ ÚLOHA PRO DEMONSTRACI SOCIÁLNÍHO INŽENÝRSTVÍ

THE LABORATORY EXERCISE FOR DEMONSTRATING SOCIAL ENGINEERING

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **Marie Kuželová**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Tomáš Lieskovan**
SUPERVISOR

BRNO 2024

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní program **Informační bezpečnost**

Ústav telekomunikací

Studentka: Marie Kuželová ***ID:*** 221672

Ročník: 3 ***Akademický rok:*** 2023/24

NÁZEV TÉMATU:

Laboratorní úloha pro demonstraci sociálního inženýrství

POKYNY PRO VYPRACOVÁNÍ:

Tato bakalářská práce má za cíl rozšířit povědomí o praktikách sociálního inženýrství a jeho metodách pro získávání uživatelských dat. Teoretická část práce se bude věnovat detailnímu popisu klíčových pojmů, včetně malware, wormů a dalších, a praktik, jako jsou útoky typu keyloggery, CEO fraud a DNS a další moderní útoky. V praktické části práce se práce zaměří na vytvoření laboratorní úlohy pro demonstraci sociálního inženýrství v prostředí platformy BUTCA.

DOPORUČENÁ LITERATURA:

[1] SALAHDINE, Fatima; KAABOUCH, Naima. Social engineering attacks: A survey. Future internet, 2019, 11.4: 89.

[2] KROMBHOLZ, Katharina, et al. Advanced social engineering attacks. Journal of Information Security and applications, 2015, 22: 113-122.

Termín zadání: 5.2.2024 ***Termín odevzdání:*** 28.5.2024

Vedoucí práce: Ing. Tomáš Lieskovan

doc. Ing. Jan Hajný, Ph.D.
předseda rady studijního programu

ABSTRAKT

Cílem bakalářské práce je zvýšení povědomí o praktikách sociálního inženýrství a jeho metodách pro získávání uživatelských dat. V teoretické části je čtenář seznámen s aktuálními hrozbami, dále je vymezen pojem samotného sociálního inženýrství jeho metod a popis útoku. Tyto techniky mnohokrát vedou k rozšíření škodlivého softwaru, proto se práce zaměřuje i na jeho definici a rozdělení na jednotlivé typy. V praktické části se práce zaměřuje na tvorbu scénáře pro demonstraci sociálního inženýrství v prostředí BUTCA. V jednotlivých podkapitolách je popsáno osm úloh, které se zaměřují na spoofing, pretexting, krádež identity, phishing, podvodné jednání a baiting. Obsahují popis a průběh jejich tvorby. Nedílnou součástí je celkové testování této laboratorní úlohy studenty základních a středních škol, proto se práce v závěru zabývá jeho výstupem.

KLÍČOVÁ SLOVA

hrozby, klasifikace útoku, malware, model útoku, metody sociální inženýrství, sociální inženýrství, techniky sociální inženýrství

ABSTRACT

The aim of the bachelor's thesis is to raise awareness of social engineering practices and its methods for obtaining user data. In the theoretical part, the reader is introduced to current threats, the concept of social engineering itself, its methods and the description of the attack are also defined. These techniques often lead to the spread of malicious software, which is why the work also focuses on its definition and division into individual types. In the practical part, the work focuses on the creation of a scenario for the demonstration of social engineering in the BUTCA environment. The individual subsections describe eight tasks that focus on spoofing, pretexting, identity theft, phishing, fraudulent behavior, and baiting. They contain a description and the process of their creation. An integral part is the overall testing of this laboratory task by primary and secondary school students, therefore the thesis deals with its output at the end.

KEYWORDS

threats, attack classification, malware, attack model, social engineering methods, social engineering, social engineering techniques

Vysázeno pomocí balíčku `thesis` verze 4.09; `https://latex.fekt.vut.cz/`

KUŽELOVÁ, Marie. *Laboratorní úloha pro demonstraci sociálního inženýrství*. Bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2024. Vedoucí práce: Ing. Tomáš Lieskovan

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Marie Kuželová
VUT ID autora:	221672
Typ práce:	Bakalářská práce
Akademický rok:	2023/24
Téma závěrečné práce:	Laboratorní úloha pro demonstraci sociálního inženýrství

Prohlašuji, že svou závěrečnou práci jsem vypracovala samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

Brno ....

podpis autorky*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Ráda bych vyjádřila upřímné poděkování svému vedoucímu práce, panu Ing. Tomáši Lieskovanovi, za jeho neocenitelnou podporu, odborné vedení, trpělivost a vstřícný přístup, včetně jeho pomoci při vytváření virtuálního prostředí na platformě BUTCA, kterým přispěl k úspěšnému zpracování mé bakalářské práce. Jeho podnětné návrhy a ochota vždy pomoci byly pro mě nezbytné. Díky patří také mé rodině a známým, kteří mě na této náročné cestě podporovali a dodávali síly k zdárnému dokončení díla.

Obsah

Seznam obrázků

Seznam výpisů

Úvod

Tato práce je zaměřena na problematiku sociálního inženýrství, která je zařazena pod netechnické útoky, neboť k jejich provedení je převážně využívána manipulace k zisku citlivých informací nebo provedení určité akce. Heslo, číslo kreditní karty a jiné údaje, které jsou spjaty s obětí, jsou příklady citlivých informací, které mohou být získány pomocí sociálního inženýrství. O to nebezpečnější tyto hrozby jsou, neboť i se sebelepším zabezpečením mohou být data odcizena. Práce dále popisuje techniky, které sociální inženýři využívají k vlastnímu obohacení jako pharming, pretexting, baiting, CEO fraud a další. Jednou z významných osobností v této oblasti je Kevin Mitnick, jehož model útočného cyklu realizován za pomoci sociálního inženýrství bude v této práci podrobněji popsán. Práce také uvádí aktuální hrozby šířící se po síti internet a definuje pojem malware, který může být úzce spjat s útoky tohoto typu.

Práce si stanovuje za cíl edukaci široké veřejnosti v této oblasti, což je jedním z prvků obrany před metodami sociálního inženýrství. Prevence spočívá v představení a demonstraci metod využívaných sociálním inženýrstvím a také poukázáním na aktuální hrozby.

Praktická část práce bude zaměřena na simulaci praktik sociálního inženýrství. Nejprve bude čtenář seznámen se základním konceptem soutěže typu Capture The Flag a také samotným scénářem. Laboratorní úloha obsahuje osm úloh zaměřených na spoofing, pretexting, krádež identity, phishing, podvodné jednání a baiting. Na úvod každé podkapitoly bude stručné seznámení se s významem a zaměřením dané úlohy. K realizaci bude využito dvou virtuálních strojů. Práci s nimi a celkový popis vývoje jednotlivých aplikací využívaných v rámci jednotlivých úkolů bude součástí podkapitol.

Konečnou fází a také nedílnou součástí této práce bude testování. Za tímto účelem bude vytvořen jednak vědomostní kvíz, jednak dotazník zpětné vazby. Testování bude realizováno ve dvou fázích. Věcné připomínky shromážděné v první fázi budou použity ke zdokonalení scénáře. Druhá fáze bude zaměřená převážně na sledování průchodu scénářem. V průběhu tohoto procesu byly vytvářeny poznámky, které spolu s dotazníkem zpětné vazby přinesly nový pohled na danou problematiku. V rámci vyhodnocení budou tyto poznatky podrobně analyzovány.

Virtuální stroje jsou na vyžádání dostupné u vedoucího této bakalářské práce.

Cíle práce

Tato bakalářská práce má za cíl rozšířit povědomí o praktikách sociálního inženýrství a metodách využívaných k získání uživatelských dat. Často se setkáváme s tvrzením, že za úspěšným útokem stojí lidská důvěřivost. Ve většině případů je bohužel toto tvrzení pravdivé, a i se sebelepším zabezpečením systému je útočník schopen při užití správné manipulace proniknout do systému bez vynaložení přehnaného úsilí.

Obrana proti těmto útokům spočívá v edukaci uživatelů, a proto je tato práce koncipovaná tímto způsobem. Cílí na prevenci před těmito praktikami, provádí uživatele různými typy útoků a poukazuje na jejich dopad.

1 Úvod do problematiky

Každý den se dostáváme do kontaktu s internetovou sítí. Komunikujeme zde s lidmi pomocí elektronické komunikace, řešíme věci po finanční stránce, žádosti obsahující naše rodné číslo a další citlivé údaje, tvoříme a sdílíme obsah. Všechny tyto aktivity spolu nesou data, které mohou být úzce spjaté s námi, případně jejich zneužití nebo ztráta by měla fatální dopad.

S narůstajícím pohybem po sítích různého typu se stává tento pohyb rutinní, upadáme do stereotypu a hrozby, které nám dříve připadaly jako nepravděpodobné, mohou zasáhnout i nás. Kolikrát otevřeme odkaz, který nám do zprávy zaslala známá osoba bez ověření, kam vlastně odkazuje. Nedochází nám, že se může jednat o spoofing nebo že už dávno tato osoba tento účet nepoužívá.

Každý den si lidé vymění nesčetné množství zpráv, je tedy vhodné se chvíli zastavit a zamyslet se, jaké informace o sobě sdílíme, komu je poskytujeme a zda jsou doopravdy tak nezbytné, jak daná osoba tvrdí.

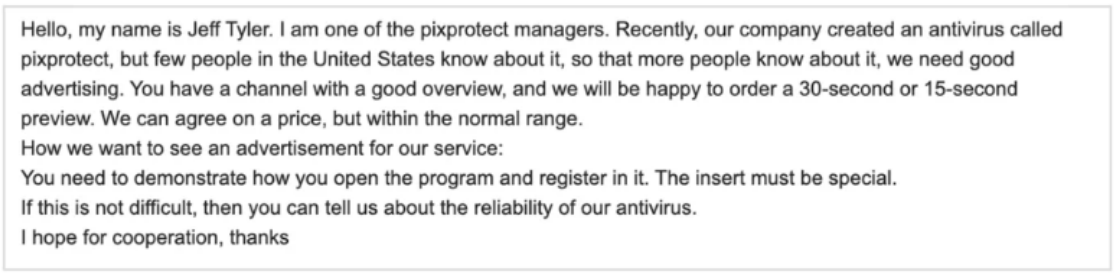
1.1 Aktuální Hrozby

Následující část se zabývá různými hrozbami kolujícím po síti internet. Od hrozeb sofistikovaných až po hrozby plynoucí z nedostatečné právní korekce či nepozornosti uživatele. Mezi jedny z nejčastějších hrozeb patří krádeže identity a zneužití přihlašovacích údajů.

1.1.1 Redline Stealer

Malware typu trojský kůň, jehož cílem je sběr dat. Využívá se ke krádeži hesel, přihlašovacích údajů, souborů cookies či bankovních údajů z prohlížeče. Je nabízen jako služba. [1] Malware as service, zkráceně MaaS, popisuje nezákonný pronájem software či hardware. Tuto službu je možné si zakoupit na Dark Webu spolu s návodem k použití. Kupující této služby pak nemusí být odborníkem v dané sféře, aby jednoduše zahájil kybernetický útok. [2]

Phishingová kampaň zasáhla i YouTube. Tvůrci na platformě YouTube využívají své dosahy k propagaci společností v rámci placené spolupráce. K tomu využívají e-mailovou komunikaci. V tomto případě šlo o phishingovou kampaň, kde tvůrcům nabízeli finanční odměnu za propagaci jejich produktu viz 1.1, většinou šlo o programy na úpravu fotografií, VPN, antivirový program nebo on-line hru. Potom co obě strany souhlasily, aktéři této kampaně zaslali odkaz na daný produkt, případně i smlouvy ve formátu PDF. [3]


Hello, my name is Jeff Tyler. I am one of the pixprotect managers. Recently, our company created an antivirus called pixprotect, but few people in the United States know about it, so that more people know about it, we need good advertising. You have a channel with a good overview, and we will be happy to order a 30-second or 15-second preview. We can agree on a price, but within the normal range.
How we want to see an advertisement for our service:
You need to demonstrate how you open the program and register in it. The insert must be special.
If this is not difficult, then you can tell us about the reliability of our antivirus.
I hope for cooperation, thanks

Obr. 1.1: Jeden ze zaslaných e-mailů [1]

Aby celá kampaň působila co nejvíce důvěřivě, vydávali se za legitimní stránky, které byly vytvářeny pomocí šablon, jako Cisco VPN nebo Steam, což je platforma pro distribuci her. V některých případech došlo dokonce ke kopírování obsahu sociálních sítí skutečné společnosti. [3]

Po spuštění falešného softwaru se data přenáší na útočníkův server. Samotné bezpečnostní mechanismy po spuštění škodlivý soubor nedetekovali. V rámci tohoto útoku nebyl využíván jen RedLine, ale i další, většinou volně dostupné malware na platformě GitHub, jako Vidar nebo Nexus stealer, obecně nazývané jako Cookie Thef malware. [3]

Získané YouTube kanály byly následně přetvořeny na on-line kryptoměnové směnárny, případně šířili tento malware dále vložením odkazu na škodlivé soubory. [3]

1.1.2 Bazarový vishing

Obětí se v tomto případě stává vystavitel inzerátu. Možný kupující vystavitele kontaktuje s nabídkou, komunikace probíhá bez menších pochyb až do doby platby. Platbu zboží chce zájemce provést přes platební bránu, která na první pohled vypadá důvěryhodně, avšak se nejedná o pravou platební bránu a údaje zde vyplněné končí v rukou útočníka. [4] V některých případech vytvoří falešné stránky známého doručovacího dopravce a přesvědčí oběť k vyplnění bankovních údajů, které jsou nutné pro doručení platby. Oběť ujišťují, že poštovné si uhradí sami a vše bude dopředu uhrazeno, o zbytek se postará kurýr. [5] V poslední řadě pachatel vytvoří v mobilní aplikaci kopii platební karty oběti. Díky všem sděleným informacím, jako je i PIN kód, může pachatel začít s výběrem finančních prostředků oběti, případně si může na dotyčnou osobu vzít i úvěr. [4]

Jak uvádí policie na svých webových stránkách: „Pin kód ke kartě je v tomto případě přirovnatelný ke klíči od domu, který také nikomu jen tak nedáte. [4]“

Průzkum společnosti ESET spolu s Policií České republiky mapuje rozsah podvodů spojené s bazary. V průzkumu šlo nejčastěji o osoby ve věkovém rozmezí

30–40 let z toho třetina všech dotázaných se osobně s podvodem setkala (31 %). Finanční škoda do 5 000 korun se týkala 5 % dotázaných, do výše 1 000 šlo o 9,36 % a pětina 20,64 % uvedla, že podvod ve správný okamžik rozeznala, tedy se vyvarovala finanční ztrátě. [6]

1.1.3 BaiRBIE.me, barbieselfie.ai

Velitelství Informačních a Kybernetických sil a Slovenský Národný bezpečnostný úrad vydaly varování před těmito podvodnými aplikacemi představující zásah do digitálního soukromí a bezpečnosti dat na svých profilech. Jedná se o neautorizovanou webovou aplikaci, využívající umělou inteligenci, která má za účel proměnit fotografii, za pomocí filtru, na nejslavnější panenku na světě. Smluvní podmínky této aplikace uvádí, že daná společnost může s fotografií a daty uživatelů nakládat, jakkoliv podle svého uvážení. Uživatelé se tedy nejen vzdávají kontroly nad svými daty, ale také možnosti kompenzace. V případě, že by došlo k právnímu sporu, budou zde platit zákony Izraele, tím pádem není v souladu s Evropským nařízením GDPR, a není jasné, jaké informace shromažďuje, a jak s nimi nakládá. Dále také nezaručují, že po stažení upraveného obrázku si uživatel nezanese do svého zařízení soubor infikovaný malwarem. U aplikace barbieselfie.ai po odsouhlasení zásad ochrany osobních údajů uživatel předává přístup k historii plateb, fotoaparátu, technickým údajům o zařízení nebo geolokaci. [7, 8]

Po vytvoření fotografie je nutné vyplnit e-mail na který slíbená fotografie nedorazí. Tento údaj je nutné vyplnit, neboť důvodem je propojení fotografie s e-mailovou adresou. Poskytovatel může následně databázi se sesbíranými daty prodat nejvyšší nabídce. [7]

1.1.4 TikTok

Ke dni 8. března vydal Národní úřad pro kybernetickou a informační bezpečnost varování před instalací a používáním aplikace TikTok, představující bezpečnostní hrozbu. Společnost ByteDance, provozující sociální platformu TikTok, patří mezi subjekty čínské národní legislativy. Ta ukládá povinnost nahlašovat zjištěné bezpečnostní zranitelnosti a přísně zakazuje zveřejňování těchto zranitelností jednotlivcům či zahraničním organizacím. Mimo to aplikace shromažďuje masivní množství informací o uživatelích jako lokaci, která je pravidelně kontrolována, dále přistupuje ke kontaktům, kalendáři, který má možnost také upravovat, zjišťuje informace o dalších spuštěných či nainstalovaných aplikacích, využívá nativní prohlížeč, kde sleduje stisknuté klávesy a další akce uživatele, sériové číslo zařízení, konfigurace Wi-Fi a další. Tyto informace mohou být následně využity v rámci kybernetického útoku

na vybranou osobu například pomoci spear-phishingu, vydírání či jiným metodám k prosazení zájmů kyberzločince.

Pozměněná verze Douyin pro čínský trh dokáže bez vědomí uživatele stáhnout a nainstalovat škodlivý kód do zařízení využívající tuto aplikaci. Nelze tedy vyloučit, že by podobnou funkcionalitu mohla v budoucnu používat i aplikace TikTok, kterou by mohla využívat pro zasazení škodlivého kódu do zařízení.

Je třeba dbát na to, že i další aplikace od společnosti ByteDance mohou představovat hrozbu v oblasti kybernetické bezpečnosti. Proto je vhodné kontrolovat, k čemu aplikace přistupuje, jaké informace shromažďuje a jak s nimi nakládá. [9]

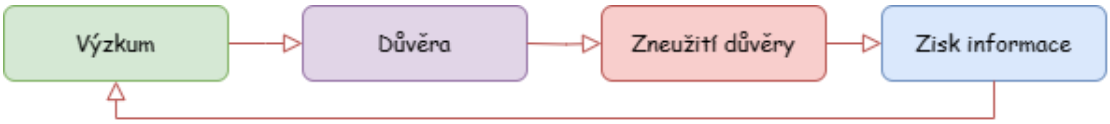
2 Sociální inženýrství

Jde o schopnost působit na jedince způsobem, který vede k získání citlivých informací, ať už se jedná o hesla, adresy či bankovní údaje, pomocí lidské zranitelnosti, tedy bez nutnosti přímého proniknutí do systému. Cílí na lidskou důvěru, slabost či snaží se v lidech vyvolat strach. I když tato strategie kybernetického útoku je méně pokročilá z technického pohledu, jde o velmi sofistikovanou metodu a může způsobit vážné škody. Mezi zločinci jde o velmi populární taktiku zaměřenou na přirozené tendence k důvěře, neboť je jednodušší přesvědčit oběť k akci než složitě obcházet bezpečnostní software. [10]

Záměrem sociálního inženýrství je získávání informací. Zároveň zde platí, že každá i ta téměř nepodstatná informace je velmi cenná pro zločince. Díky všem těmto částem si může sestavit kompletní přehled o oběti a prostředí, ve kterém se pohybuje [11]. Následně pokud zjistí, na co oběť reaguje, pak může s obětí snadno manipulovat [10].

2.1 Vzorec útoku

I když se jednotlivé útoky prováděné pomocí SI liší, nesou v sobě jeden vzorec sestavený ze čtyř fází k vytvoření vztahu a důvěry: výzkum, důvěra, zneužití důvěry, výstup beze stop. Jde o nejznámější model útočného cyklu, jehož autorem je Kevin Mitnick. [12]



Obr. 2.1: Znázornění cyklus útoku za pomoci SI

1. **Výzkum** – první fáze, někdy také označovaná jako sběr informací, cílí na výběr oběti, následné rešerši, sběru informací o cíli a vyhodnocení útočné metody.
2. **Důvěra** – zaměřuje se na navázání vztahu založeného na důvěře mezi obětí a útočníkem za pomoci přímého kontaktu či e-mailové komunikace. K rozvoji tohoto vztahu využívají zkreslení identity, tváří se jako známá osoba oběti případně nějaká autorita. K zvýšení důvěry ve vztahu mohou také využít zasvěcené informace či projev nouze.
3. **Zneužití důvěry** – po navázání vztahu útočník manipuluje s obětí ve snaze získat konkrétní informaci, data nebo ji vyzve ke konkrétní akci.

4. **Získání informace** – díky předchozím fázím byl útočník schopen získat původně požadovanou informaci. Ukončuje komunikaci téměř beze stop a přesouvá se na další cíl. [10, 12, 13]

2.1.1 Praktický příklad

Jedním z praktických příkladů může být podvod přes inzerát. Podvodníci, většinou cizinci, se v tomto případě vydávali za kupující a manipulací se snažili získat údaje o platební kartě vystavitele inzerátu. Následně žádali o zaslání potvrzovací SMS. Rozšíření povědomí o těchto podvodech následně vedlo k zdokonalení této techniky. Více bylo popsáno v podkapitole 1.1.2.

Dalším, dá se říci trochu zastaralým příkladem, mohou být dopisy z Nigérie. Pachatelé se zaměřují na citovou stránku člověka, se kterým navazují vztah přes dopisy, seznamky či e-mailovou komunikaci, ve které prezentují svůj životní příběh spjatý s nepříjemnou životní situací, ve které se ocitli a prosí o finanční výpomoc nebo lákají na vidinu provize za pomoc s transakcí. Například v roce 2010 koloval následující dopis, který je možné v databázi hoax.cz dohledat pod názvem „volání o pomoc z miss Elna Carole Janvier“. Dopis je psán kostrbatou češtinou, kde mladá slečna Elena žijící v Pobřeží Slonoviny žádá o pomoc. Po tragickém úmrtí obou rodičů zůstala sama a nemá možnost, jak převést jmění jejího otce na sebe, neboť je jí teprve 17 let. Prosí o převod této částky na účet příjemce a slibuje možnost zisku z této spolupráce.

2.1.2 Klasifikace útoků

V návaznosti na cyklus útoku, lze, podle [13], definovat čtyři typy útoků.

- **Technické útoky** – patří mezi jedny z nejúčinnějších typů. Objevuje se zde snaha získat cenné informace předstíraje spolehlivý subjekt. V tomto případě nedochází k osobní interakci, využívají zvláště nástroje jako e-mail, webové stránky nebo stažení infikovaného souboru.
- **Ego útoky** – jsou založeny na přímém kontaktu, oběti je nabízena pomoc. Útočník prokazuje své schopnosti k řešení daného problému, následně oběť upadá do transparentní manipulace, kde snadno předá požadované informace.
- **Útoky za pomoci sympatií** – cílí na vytvoření důvěrného vztahu s obětí. Využívá podněty, k dosažení empatie, vytváří spontánní, plynulé konverzace, k čemuž využívá dokonalých konverzačních technik. V momentu, kdy se oběť cítí bezpečně, začne odhalovat své slabiny, čímž umožní útočníkovi jednoduše využít těchto zranitelností k získání potřebných informací.

- **Útoky postavené na šikaně** – zaměřují se na strach. Podobně jako u technického útoku, jde o jednu z účinných technik, která využívá zastrašování nebo nátlak k zisku informací.

Útoky SI mohou být jejich kombinací a k tomu využívat různé prostředky, ať už fyzické, jako shromažďování cenných dokumentů z popelnice oběti [14], sociální, zahrnující lidské interakce nebo technické, které jsou prováděny pomocí internetové sítě. [15]

2.2 Metody sociálního inženýrství

Určité koncepty SI vychází z taktiky pomocníků. Manipulují s obětí tak aby se zdálo, že je oběť nápomocná a tím dosáhli svého cíle [11]. Jiné cílí na strach či důvěru. Jak tyto metody pracují, je podrobněji popsáno v následující kapitole.

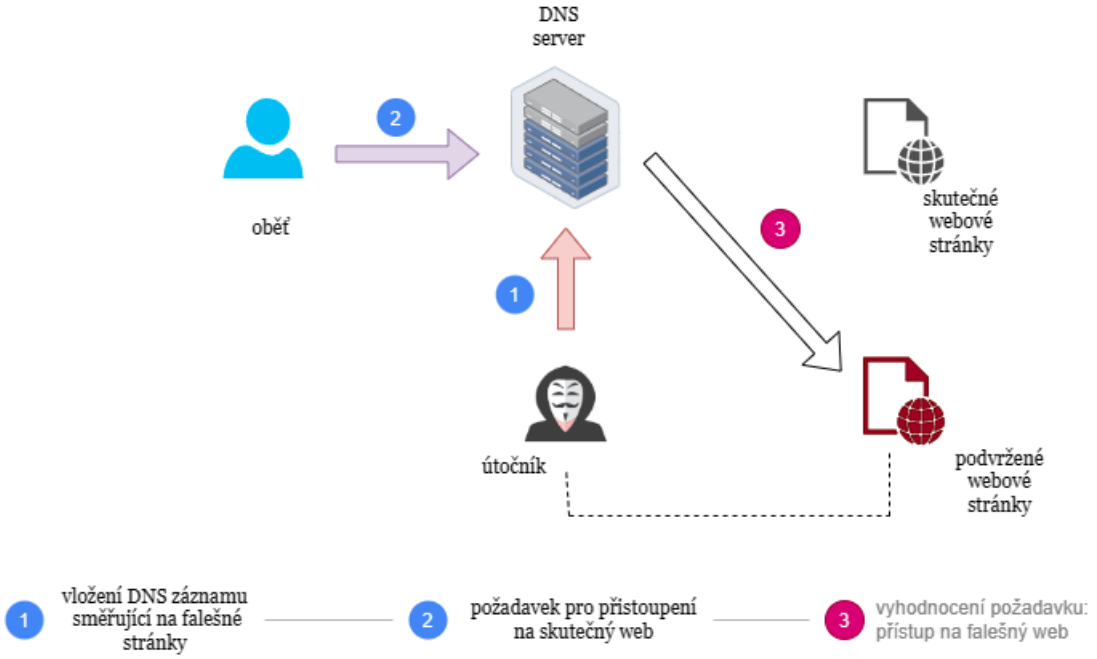
2.2.1 Phishing

Jde o jednu z nejrozšířenějších forem SI. Jedná se o technickou metodu shromažďování dat, při které se útočník vydává za známého jedince nebo legitimní organizaci, takzvaný spoofing, ve snaze získat soukromé informace, prostřednictvím telefonní či e-mailové komunikace ve které se odkazuje na falešné webové stránky. [16] Oběť může být přiměna k aktualizaci svých osobních údajů, avšak před tím musí vyplnit přihlašovací údaje, což provede na falešné stránce a v ten moment útočník získává její citlivé informace, ať už se jedná o heslo nebo číslo kreditní karty.

- **spear phishing** – jde o útoky cílené na konkrétního jedince, společnost či skupinu lidí. Na rozdíl od samotného phishingu necílí na větší skupinu náhodných lidí, ale na jedince, kteří mají něco společného. Například pracující ve stejné společnosti. Aby se zpráva tvářila důvěryhodně, přizpůsobí ji charakteristikám oběti. [10, 17] Tedy snaží se o to, aby zpráva působila stejně jako by ji odesílala doopravdy oběť, proto je nutná studie chování oběti, jak osoby oslovuje, jaký jazyk používá, jak se vyjadřuje.
- **whaling** – spear phishing, také známý jako CEO fraud, zaměřen na vysoce postavené osoby v určité společnosti. [18]
- **vishing** – název se skládá ze slov *voice* a *phishing*, jde tedy o útoky prostřednictvím protokolu VoIP, Voice over Internet Protocol [19]. Řadí se tedy mezi telefonický phishing [20]. Příkladem může být e-mail obsahující falešné bankovní telefonní číslo. V těle zprávy je po uživateli požadováno telefonické ověření identity. Následně dochází k opakovanému selhání přihlášení, aby oběť mohla několikrát zadat své údaje, případně vyzkoušela jiné typy hesel, které používá [21].

Může se také jednat o výhru. K převzetí výhry je nutno zaslat adresu, finanční prostředky a další osobní informace. Jedním ze stále celkem běžných příkladů vishingu je podvod, který se používá od roku 2009. Zde se útočník staví do role zaměstnance firmy Microsoft, případně Amazon nebo jedné z další známé společnosti. Telefonicky kontaktuje oběť a informuje ji o podezřelé aktivitě, připisujíce toto chování viru v jejím počítači. Následné zotavení počítače probíhá stažením „antivirového programu“, který je ve skutečnosti malware. [10]

- **SMS phishing** – forma telefonického phishingu využívající krátké textové zprávy SMS. [17]
- **pharming** – *DNS phishing*, je typ útoku přesměrovávající provoz webových stránek na stránku podvrženou. Doménové jméno se mapuje na IP adresu falešné webové stránky. [17] Pro ucelení této představy je přiložen obrázek 2.2 reprezentující útok pomocí pharming.



Obr. 2.2: Znázornění pharming útoku

2.2.2 Pretexting

„Akt vytvoření vymyšleného scénáře, který má přesvědčit cílovou oběť, aby uvolnila informace nebo provedla nějakou akci [22].“

Pretextingové techniky si zakládají na tvorbě přesvědčivých scénářů ke krádeži osobních údajů [23]. Snaží se vymyslet příběh, co nebude vzbuzovat žádné pochybnosti, k tomu je třeba znát zázemí, ve kterém se oběť nachází [10]. Například, pokud

cílí na velkou společnost, využije skutečnosti, že jednotlivé pobočky spolu nekomunikují přímo. Následně jednu z poboček osloví. Představí se jako generální ředitel dané společnosti, zná všechna důležitá fakta, odkazuje se na osoby, které na pobočce znají, může mít s sebou falešný průkaz, dokazující že přichází z dané firmy. Tím vytvořil geniální příběh pro získání finanční zprávy, kterou potřebuje nutně pro svou pobočku získat. Zaměstnanec ochotně tyto zprávy poskytne a stane se obětí pretextingu. [24]

2.2.3 Dumpster diving

Vzestupující technika, převážně v Americe, kde je tímto způsobem získáváno přes 88 % informací [21]. Jedná se o techniku získávání cenných papírů z firemního odpadu, jednoduše řečeno prozkoumáváním popelnic. Nemusí jít pouze o papíry, ale například o přenosná úložiště dat nebo staré vyřazené zařízení, kde nebyla data zformátována.

V dnešní době se zavádí striktní pravidla jako zamykaní košů nebo skartování dokumentů. Skartace může být efektivní, pokud využívá správných nástrojů [21]. Tabulka 2.1 uvádí, jaký typ skartace zvolit v závislosti na zvoleném dokumentu. U přenosných úložišť se mezi nejefektivnější metody řadí mechanické poškození, což způsobí nevratnou obnovu dat, neboť v důsledku špatného zformátování je zde šance obnovy.

Typ	**Velikost skartace**	**Záměr**
pásová skartace	3/8"	obecné dokumenty
příčná skartace	3/8"x 1 1/2"− 3 3/8"	obecné dokumenty
pásová skartace	1/4"− 1/8"	citlivé dokumenty
pásová skartace	1/16"	důvěrné dokumenty
příčná skartace	1/8"x 1-1/8"	důvěrné dokumenty
příčná skartace	1/16"x 5/8"	tajné dokumenty
příčná skartace	1/32"x 1/2"	přísně tajné dokumenty
příčná skartace	1/26"x 1/5"	nejvyšší stupeň zabezpečení

Tab. 2.1: Znázornění skartace v závislosti na cennosti dokumentu

U pásové skartace jde o typ drtiče papíru, který papír řeže na úzké proužky o šířce dle dané velikosti skartace v tabulce. Výsledkem příčné skartace jsou konfety vytvořené pomocí příčné a následně podélné skartace papíru.

2.2.4 Baiting

Do češtiny by se dala tato metoda přeložit jako útok s návnadou. Může jít o CD, flash disky či jiné fyzické předměty, které se nachází na veřejném prostranství či společném prostoru organizace nebo odkaz na stažení za normálních okolností placeného obsahu zdarma. [10] Věci zdánlivě tvářící se, že jsou zdarma, avšak opak je pravdou. Toto bezplatné stažení licencovaného softwaru se sebou přenáší do počítače také malware, který může různými způsoby poškodit systém. Objevuje se zde snaha o nalákání na produkty zdarma.

Trojský kůň **CANDY**, což je zkratka pro Controller Area Network, byl vyvinut pro výzkumné účely na operačních systémech Android. Jeho hlavním cílem je proniknout do automobilových systémů, což může ohrozit bezpečnost řidičů. Po stažení se aplikace chová předpokládaným způsobem, avšak zároveň transparentně otvírá zpětný komunikační kanál, díky kterému má útočník přístup k zařízení. Škodlivý kód byl, pro snížení podezření o legitimitě aplikace, vložen do platné aplikace pocházející z Android Market. [25]

Cílem tohoto útoku byla multimediální stanice *Android Infotainment Radio využívající verzi operačního systému Android 4.4 KitKat*, ta umožňuje přehrávání hudby, komunikaci či navigaci. Rádio patří mezi standardní produkty s nízkou mírou zabezpečení. Nenachází se zde firewally, IDS systémy nebo jiné bezpečnostní prvky. [25]

Za aplikací, která měla za úkol vyhledávat nejbližší čerpací stanice v okolí, se skrývala škodlivá aplikace. Následně díky ní bylo možno odcizit textové soubory s GPS souřadnicemi, trajektorii vozidla, zvukové konverzace, informace o ovladačích, snímky z parkovacích kamer, které jsou uloženy v *mygallery*. CANDY, škodlivá aplikace, je jedním z příkladů návnady, neboť byla za pomocí technik sociálního inženýrství rozšířena mezi širokou veřejnost. [25]

Můžeme zde vidět viditelné prvky připomínající *baiting*. Aplikace, nacházející se na relevantním místě Android Market, lákající uživatele ke stažení. Bez vědomí uživatele získává dostupné informace, které může dál využívat.

2.2.5 Quid Pro Quo

Metoda podobající se na *baiting*, která výměnou za citlivé informace nabízí službu. Příkladem může být útočník vydávající se za IT konzultanta. Po nalezení osoby, co skutečně má tento problém, bude pro jeho „opravu“ nutně potřebovat některé citlivé informace, jako hesla, číslo karty a další. [10]

2.2.6 Tailgating

Jedna z forem SI, která se zaměřuje na sblížení se s obětí po osobní stránce v reálném světě. Snaha o vniknutí do přísně střežených budov za pomoci morálních pravidel. Příkladem může být osoba nesoucí hromadu těžkých krabic a prosící Vás o podržení dveří. Kdo by takového člověka nepustil? Bez prokázání totožnosti je schopen se dostat do interiéru budovy během vteřiny, kde pak může využít další metody, jako již několikrát zmiňovaný *baiting*, tedy zanechání infikovaného fyzického zařízení na tomto místě.

3 Malware

Pojem malware pochází z anglického výrazu „malicious software – škodlivý software“ [26]. Software snažící se narušit standardní činnost systému, proniknout do soukromých počítačových sítí, sběr osobních a důvěrných informací, zisk přístupu k zařízení. Je schopný vykonávat více činností najednou, jako sběr informací a šíření se mezi další zařízení. [26, 27] Po vniknutí do zařízení se ukryjí, deaktivují bezpečnostní prvky a kontaktují řídící servery pro další instrukce [28]. Jde o souborový termín označující obtěžující software, pod nějž spadá spyware, adware, červ, trojský kůň, ransomware a další [29].

3.1 Spyware

Jak již z názvu vychází, jedná se o malware sloužící k monitorování nebo také špehování uživatele. Přistupuje k důvěrným informacím (bankovní, osobní či přihlašovací údaje) a sleduje chování uživatele, které může poskytovat třetím stranám. Jeho instalace probíhá skrytě nebo může být součástí bezpečného softwaru, pro monitorování vlastních aktivit za účelem cílené reklamy definované ve smluvních podmínkách. [26]

3.1.1 Druhy Spywaru

Keylogger – v jeho přítomnosti na zařízení nastává to, že se bez vědomí uživatele zaznamenávají stisknuté klávesy. Útočník je poté schopen jednoduše zaznamenat heslo a další citlivé údaje, může zaznamenávat navštívené weby, pořizovat snímky obrazovky nebo číst zaslané zprávy. Může se jednat i o hardwarové zařízení určené ke špionáži. K tomu je nutný fyzický přístup k zařízení, kdy jej kyberzločinec vloží mezi CPU a kabel od klávesnice. [30]

Infostealer – pro svou činnost může využívat keylogger, jedná se však o škodlivý software snažící se získat takové citlivé informace, které následně může využít ke krádeži identity. Narozdíl od keyloggeru nezaznamenává všechny kroky uživatele, které na zařízení prováděl. [31]

Browser hijacker – jde o malware vložený do prohlížeče. Útočníci pozmění nastavení prohlížeče tak, aby byla oběť odkazována na škodlivé stránky, případně může sloužit k nastavení cílené reklamy nebo zobrazení historie prohlížení útočníkovi. [32]

Banking trojan – zaměřuje se na získání bankovních údajů oběti. Jedním z příkladů je Zanubis, zdánlivě legitimní aplikace pro zařízení s operačním systémem Android,

který od uživatelů žádá oprávnění k přístupu, díky níž získá následně kontrolu nad samotným zařízením. Následně se Zanubis vyvinul do formy, kde se vydává za oficiální aplikaci perské vládní organizace SUNAT (Národní úřad pro celní a daňovou správu). [33]

3.2 Adware

Nejméně nebezpečný typ malware spojen s reklamou, ať už v rámci vyskakovacích oken či reklam přímo v prohlížeči, jehož instalace probíhá bez svolení uživatele. Hlavním cílem je získat výnos z reklam generující se po kliknutí na okno reklamy. Co však spolu s nevyžádanou reklamou do zařízení dokáže zanést je škodlivý software, například již zmíněný keylogger, či měnit nastavení. Adware může být zakomponován v rozšíření prohlížeče, bezplatném softwaru. [26]

3.2.1 Agent Smith

Jeden z nejznámějších adware, pojmenován po postavě z filmu Matrix, zaměřující se na systémy Android. V roce 2016 byl poprvé objeven a pracoval na základním principu adware, neustálé zobrazovaní reklam. V roce 2018 se tento produkt vyvinul do podoby, která umožňuje vkládání reklam do legitimních aplikací a zabraňuje jejich aktualizaci. Šířil se na platformě třetích stran 9Apps, infikoval přes 25 miliónů zařízení, převážně v Indii, kde je tato platforma oblíbená. Tvůrci tohoto adware se snažili o vniknutí na distribuční službu Google Play, avšak tyto škodlivé aplikace byly včas ohlášeny a smazány. [34]

3.3 Červ

Programy spadající pod tuto kategorii nepotřebují ke svému šíření hostitele, tedy nějaký přídavný program, v kterém by se schovaly. Nejprve se snaží nalézt zranitelnosti, které by mohl využít ke vstupu do zařízení. Vnikne-li do zařízení, ukryje se a následně shromažďuje informace. V poslední fázi vytvoří svou kopii, kterou šíří pomocí síťové komunikace mezi další zařízení. [35]

3.3.1 Christma Exec

Tento červ se šířil pomocí e-mailu. V případě, že si příjemci tělo zprávy uloží jako spustitelný soubor, získají krásné vánoční přání. Ti, co tyto instrukce vykonali, nedostali jako dárek jen pohlednici s vánoční tematikou, ale i červa, který skenoval

jména dalších korespondentů, kterým se následně přeposlal. Dopadem tohoto útoku byla nedostupnost sítí IBM po dobu, dokud nebyl červ odstraněn ze sítě. [36]

3.4 Ransomware

Tento typ malware zásadně ovlivňuje oběť způsobem šifrování a uzamčení přístupu ke všem složkám a souborům. Pokud by chtěl uživatel data zpět, musí zaplatit výkupné v těžko vyhledatelné měně jako je například virtuální měna bitcoin. [13] Z tohoto důvodu je také označován jako vyděračský software. Do počítače vniká pomocí jiného typu malwere, jako červ, ten se může nacházet v příloze e-mailu nebo na webových stránkách. Červ po vniknutí nejprve přípravný půdu pro ransomware, následně probíhá jeho stažení. [26]

Dle omezení provozu systému se ransomware dělí na dva typy. V prvním případě se jedná o takový druh ransomware, který zcela omezí přístup k zařízení, většinou blokuje samotné spuštění operačního systému. Typem druhým je pak znepřístupnění dat, avšak systém zůstává v provozu. Důležitá data pro uživatele jako fotky, tabulky či videa jsou zašifrována. Uživatel je o této situaci informován, zobrazením zprávy, a aby došlo ke zpětné dešifraci, je nutné zaslat peněžní obnos, většinou do předem stanovené časové lhůty. [26]

3.5 Denial of Service

Snaží se o vyřazení služby z obvyklé činnosti, zamítnutí přístupu k aplikaci nebo jinému prostředku, případně zahltí zařízení a tím sníží jeho výkon. [26]

- **DoS** – v tomto případě je útočníkem jedno zařízení. Existuje zde relativně snadná obrana, tedy blokace jednoho zařízení. [26]
- **DDoS** – *„Distributed Denial of Service“*, zdrojem útoku je více různě geologicky umístěných zařízení. Ty se pak stávají součástí botnet sítě, která je ovládána z jednoho bodu. Útoky tohoto typu jdou potencionálně ničivější a těžko odhalitelné. [26]
- **DRDoS** – *„Distributed Reflected Denial of Service“*, jde o jedny z nejsilnějších útoků tohoto typu. Útočník využije IP adresu oběti pro zaslání požadavku z mnoha různých zdrojů. Požadavek se zdá být legitimní, neboť působí jako dotaz od serveru. Odpovědi na tyto požadavky se hromadí na serveru a ten se pak stává nedostupným. [26]

4 Realizace praktické části

Capture the Flag (CTF) je soutěž obsahující skryté výzvy, jenž uživatel musí odhalit. Tyto výzvy jsou označovány jako „flagy“, těmi mohou být například soubory nebo textové řetězce. Samotný scénář uživatele navádí k jejich nalezení. Když uživatel správně identifikuje flag, postupuje k další úloze.

Praktická část práce se zaměřuje na tvorbu scénáře v prostředí platformy BUTCA, který se věnuje tématu sociálního inženýrství. Jelikož se jedná o rozsáhlé téma zasahující každého je jeho složitost přizpůsobena široké veřejnosti. Obsahuje jak jednodušší úlohy poukazující na možný dopad útoku, tak úkoly více komplikované, kde uživatel musí zadaný „flag“ složitěji nalézt.

4.1 Popis scénáře

Prevence je klíčovým prvkem v boji proti útokům tohoto druhu. Z tohoto důvodu má scénář za cíl provést uživatele situacemi a taktikami, které jsou jim typické, a ukázat, jak snadno mohou pod nátlakem nebo nepozorností poskytnout citlivé informace, které by za normálních okolností nikdy nikomu neprozradili. Díky této preventivní formě se uživatel naučí rozpoznávat základní varovné signály pokusu o získání těchto informací. V tomto scénáři uživatel ztvárňuje nadějnou studentku Evu z VUT, která řeší běžné studentské problémy a obtíže.

Scénář obsahuje osm úloh zaměřených na spoofing, pretexting, krádež identity, phishing, podvodné jednání a baiting.

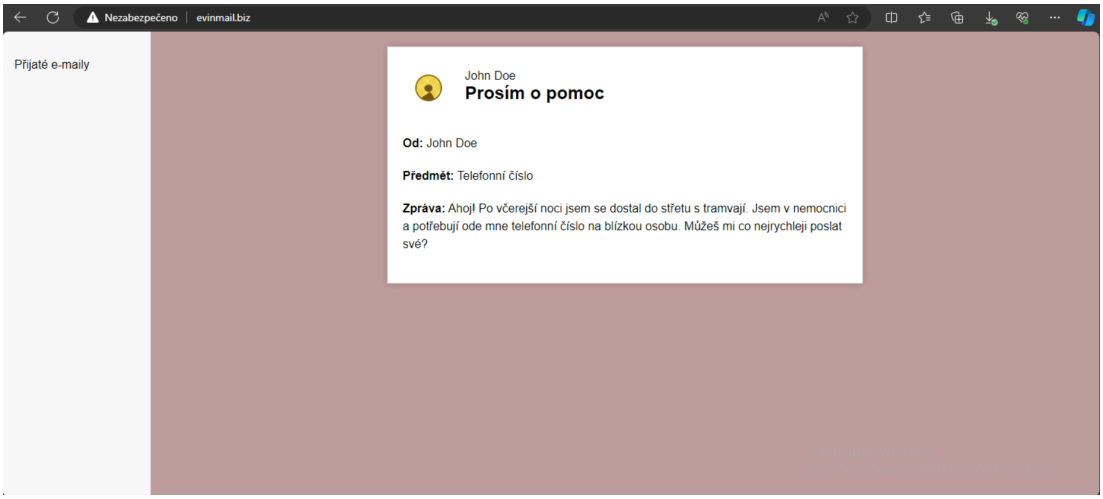
Pro demonstraci je zde využito dvou virtuálních strojů. Stroj s operačním systémem Windows 10 představuje potencionální oběť naopak systém Kali Linux prezentuje útočníka.

4.2 Útok první – E-mail spoofing

První úkol se zaměřuje na podvodné e-maily, kde útočník využívá taktiku změny zobrazovaného jména. Pravděpodobně jste již obdrželi podobný e-mail, kde útočník falešně předstírá, že má váš účet, a strachuje vás, že má přístup k vašim přihlašovacím údajům. Útočník se snaží vytvořit dojem naléhavé situace, aby vás přiměl k akci. Podobně je tomu i v tomto úkolu, který poukazuje na incidenty, kde po odeslání telefonního čísla přišli uživatelé o své finanční prostředky, případně toto číslo využili k spoofingu aby vylákali citlivé údaje od známých majitele čísla.

Při tvorbě tohoto úkolu bylo nejprve nutné vytvořit iluzi této situace. To by bylo možné například pomocí interaktivního obrázku, avšak jsem dospěla k závěru, že by bylo vhodnější zhotovit vlastní webovou stránku, která bude provozována na serveru

Apache. Ta využívá stylů předdefinovaných v souboru *style.css*. Díky definovaným třídám vytváří šablonu 4.1, která obsahuje postranní lištu a samotné tělo e-mailu, jenž se skládá z hlavičky obsahující profilový obrázek a základní informace o e-mailu jakožto jméno odesílatele a popis. Další část obsahuje shrnutí a samotnou zprávu.



Obr. 4.1: Webová stránka e-mailové schránky.

4.2.1 Apache server

Na straně serveru nejprve bylo nutné vytvořené soubory vložit na správné místo, tedy */var/www/*, které standardně slouží k umístnění webových souborů. Aby uživatel byl schopen přistupovat na stránku přímo zadáním adresy *evinmail.biz* bylo nutné vytvořit konfiguraci virtuálních hostů pro tento webový server. Prvním krokem tedy byla tvorba podadresáře, čímž byl oddělen obsah jednotlivých webů. Následně samotné konfigurační soubory nalezneme ve složce */etc/apache2/sites-enabled/*, kde každý soubor obsahuje konfiguraci pro konkrétní stránky, je tedy potřeba vytvořit také konfigurační soubor pro doménu *evinmail.biz*. Tento soubor obsahuje 4.1 definici naslouchajícího portu, pro všechny síťové rozhraní, typicky pro HTTP port 80. Pomocí direktivy *ServerName* určujeme název serveru pro daného virtuálního hosta. Na základě tohoto jména se následně vybírá vhodný virtuální host, to probíhá tím způsobem, že pokud přijde požadavek s IP adresou a portem, která je stejná pro více hostů, začne porovnávat právě direktivy *ServerName* a *ServerAlias*, který určuje alternativní názvy. Dále se uvádí kořenový adresář, tedy dříve vytvořená složka se soubory. V poslední řadě povoluje přepis direktiv v daném adresáři a určení umístnění protokolů chyb.

Stejným způsobem byly vytvořeny konfigurace pro stránky používané v dalších úkolech.

Výpis 4.1: Konfigurační soubor virtuálního hosta.

```
1  <VirtualHost *:80>
2    ServerName evinmail.biz
3    ServerAlias *.evinmail.biz
4    DocumentRoot /var/www/evinmail.biz
5
6  <Directory /var/www/evinmail.biz>
7    AllowOverride All
8  </Directory>
9
10   ErrorLog ${APACHE_LOG_DIR}/error.log
11   CustomLog ${APACHE_LOG_DIR}/access.log combined
12
13 </VirtualHost>
```

4.2.2 Virtuální stroj oběti – Windows 10

Jelikož požaduji, aby doména *evinmail.biz* také směřovala na tuto stránku, je nutné provést úpravu hostů také u stroje s operačním systémem Windows. Soubor *hosts*, překládající IP adresy na jména, se v tomto systému nachází v adresáři s cestou *C:\Windows\System32\drivers\etc*. Při překladu prohlížeč kontroluje záznamy uvedeny v tomto souboru, proto je potřeba na konec souboru vložit nový záznam ve tvaru IP adresa serveru – doménové jméno.

Stejným způsobem byly vytvořeny záznamy pro stránky používané v dalších úkolech.

4.3 Útok druhý – Krádež identity

Tato úloha předvádí situaci, kdy byl odcizen účet spolužačce Evině, se kterou si velmi často dopisuje. Častým nešvarem na sociálních sítích bylo poslední dobou odcizení identity a odeslání zprávy všem přátelům daného účtu. Tento zjevný pokus o zisk přihlašovacích údajů spočíval v zaslání šokující zprávy typu „Proč tato žena stále zveřejňuje toto video, které o vás prozrazuje vše?“ nebo „Jsi to ty ve videu?“ odkazující zdánlivě na platformu TikTok. Kromě podivného formátu odkazu alarmující zde také bylo, že se jednalo o HTTP spojení. V druhém případě tento odkaz povýšili na spojení HTTPS, avšak doména *videow.privad0.com* na věrohodnosti nepřidávala. Přesto věřím, že kdyby mi tyto zprávy zaslal někdo, s kým komunikuji denně nikoli osoby, se kterými jsem nikdy nekomunikovala, a celkově by to do dané komunikace zapadalo, byla by tu určitá šance na otevření daného odkazu. V případě, že uživatel

odeslaný odkaz otevřel, zobrazila se mu část videa a pro pokračování ve sledování bylo nutné se přihlásit. Tím by útočník získal přihlašovací údaje do požadované aplikace.

4.3.1 Tvorba úlohy

Pro tvorbu této úlohy bylo potřeba vytvořit prostředí připomínající chatovací aplikaci, kde si spolužačky dopisují. K tomuto jsem využila šablony dostupné na platformě GitHub [1]. Jedná se o repliku aplikace Messenger vytvořené pomocí jazyků HTML a CSS. Soubor *index.html* obsahuje plně stylovou stránku pro chat, dále bylo tedy potřeba upravit textaci, aby seděla scénáři. Aby komunikace působila důvěryhodně, přidala jsem zde zprávy, které by si typicky mezi sebou kolegové vyměnili. Jelikož se jedná o vysokoškolské studenty, tak jejich hlavním tématem je složení zápočtu z určitého předmětu. Aby zpráva odeslaná po odcizení identity vypadala věrohodněji je zde přidána i komunikace o důvěrném tématu, na které zpráva navazuje.

Jako další bylo potřeba vytvořit podvrženou stránku aplikace Instagram, ke které se útočník snaží získat přihlašovací údaje. Na platformě GitHub se nachází plně funkční klon této aplikace [2]. Po stažení tohoto projektu jsem se snažila aplikaci trošku vylepšit. Jelikož se zde snažím o upozornění na možné odcizení přihlašovacích údajů, a tedy samotného účtu, napadlo mě tyto údaje přímo zobrazit na tomto webu. Možností, jak toto učinit bylo více. Původně jsem zamýšlela toto provést pomocí metody *HTTP Query String*, která přenáší parametry a jejich hodnoty přímo v URL adrese, například takto 4.2.

Výpis 4.2: HTTP Query String

```
http://localhost/profile.html?email=ff%7C%40e.sk&password=ff
```

Následovala by syntaktická analýza (*parsování*) parametrů z dané URL pomocí jazyka JavaScript. Tato metoda je relativně jednoduchá na provedení, avšak nese se sebou určitý problém. Je potřeba se totiž okamžitě odkázat na stránku, na které mají být tyto údaje zobrazeny. Abych však předešla přílišné zjevnosti, využívám možností úložiště (*storage*), konkrétně *session storage*. Úložiště umožňuje ukládat data, ve formátu klíč hodnota, v prohlížeči, na straně klienta, a následně k nim i přistoupit. Narozdíl od *local storage*, který data uchovává déle, jsou v případě *session storage* data uložena jen po dobu relace. Nejprve bylo potřeba na stránce *login.html* upravit přihlašovací formulář, tak aby bylo možné se na něj odkazovat ve funkci.

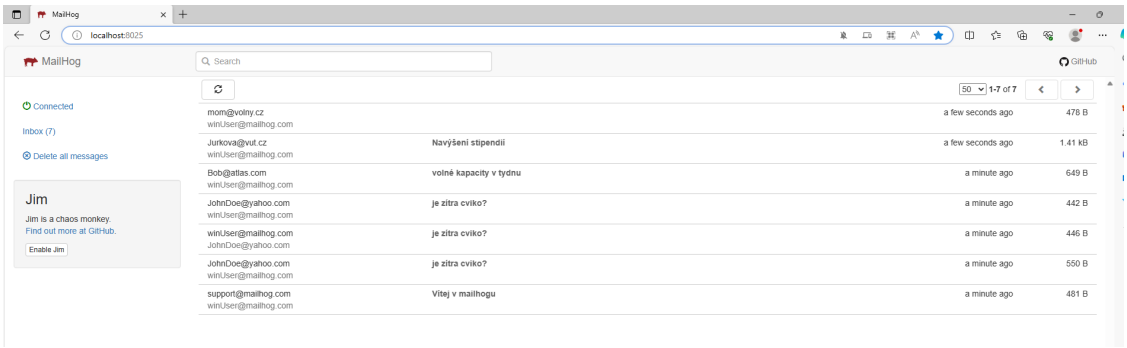
[1] `https://github.com/ravisankarchinnam/tailwindcss-messenger-clone`

[2] `https://github.com/DesignToWebsite/instagram_clone`

To znamenalo přidání jednoznačného identifikátoru a tlačítka pro odeslání formuláře. Samotný skript následně čeká na akci *submit* aby mohl začít vykonávat zbytek kódu, tedy uložit zaslaná data do *session storage* pomocí metody *setItem('klíč', hodnota)*. Následný zisk těchto dat probíhá v rámci stránky *profile.html* pomocí metody *getItem('klíč')*.

4.4 Útok třetí – SPAM

Po spuštění e-mailového klienta MailHog, je schránka prázdná neobsahující žádný e-mail. Pro zvýšení autentičnosti je vhodné zde nějakou komunikaci zaslat. K tomu je využíván skript napsán v jazyce Python, jehož úkolem je zaslání e-mailu na naslouchající port 1025. Generátor je složen z funkce *mail_generator*, využívající parametr *type* k odlišení role odesílatele a příjemce. Zasílání zpráv je zde realizováno pomocí knihovny *smtplib*. Program mezi zasláním jednotlivých e-mailů vkládá náhodnou prodlevu, což má na uživatele působit více realisticky. Vytvořenou e-mailovou komunikaci lze vidět na obrázku 4.2. Jedna z těchto zpráv má uživatele nalákat za vidinou zisku na podvržené stránky VUT. Uživateli je nabízeno z důvodu inflace, zpětné navýšení stipendií o 5 %. Jedinou podmínkou je kontrola údajů v dokumentu na zaslané adrese. Po přesměrování se uživatel nachází na zdánlivě podobné stránce VUT, tato však obsahuje momentálně neexistující loga a další indicie napovídající, že se viditelně jedná o phishingový útok. V případě stažení souboru s makry a jeho následné spuštění, způsobí po uzavření souboru aktivaci jednoduchého spambota, pomocí příkazu v programu PowerShell, který odešle nevyžádanou řetězovou e-mailovou zprávu, tedy spam.



Obr. 4.2: Zaslaná komunikace pomocí Python skriptu.

4.4.1 Způsob zaslání e-mailové komunikace

Výpis 4.3: První část skriptu pro zaslání zpráv

```
1  from random import randrange
2  from email.mime.multipart import MIMEMultipart
3  import smtplib
4  from email.mime.text import MIMEText
5  import getpass
6  from time import sleep
7
8  #nastaveni serveru + prijemce je temer vzdy stejny
9  server = smtplib.SMTP('localhost:1025')
10 sendTo =   getpass.getuser() +"@mailhog.com"
11
12 #generovani mailu
13 #type = response || recieve
14 def mail_generator(type, sendFrom, subject, body):
15     mailtext = MIMEMultipart()
16     if type == "userSend":
17         mailtext['From'] = sendTo
18         mailtext['To'] = sendFrom
19     else:
20         mailtext['From'] = sendFrom
21         mailtext['To'] = sendTo
22
23
24     mailtext['Subject'] = subject
25
26     bodyContent = body
27     mailtext.attach(MIMEText(bodyContent, "html"))
28
29     mailToString = mailtext.as_string()
30     server.sendmail(sendFrom, sendTo, mailToString)
```

Jedná se o skript sloužící k zaslání zpráv do e-mailového klienta MailHog. Cílem je vytvořit pocit skutečné komunikace napříč poštovním klientem. Z tohoto důvodu bylo vytvořeno pět osob, a to John, který je známý z vysoké školy, Bob, který je nadřízený ve firmě, kde jako hráč pracujete, Mom, rodinný příslušník, a paní Jurková z finančního oddělení VUT. Dále zde vystupuje Alice, která je však pouze zmíněna v jedné e-mailové komunikaci.

Skript 4.3 využívá mnoha různých knihoven. Například MIME slouží k formáto-

vání e-mailu pomocí jazyka HTML nebo *smtplib* pro zaslání zprávy.

Samotný server je nadefinován jakožto globální proměnná, neboť všechny zprávy budou zasílány právě zde.

Funkce *mail_generator* slouží k zaslání e-mailu hráči. Pro autentičnost není jméno v e-mailu definováno staticky, místo toho je zde využito funkce *getuser*, která vrací první neprázdný řetězec obsahující přihlašovací jméno uživatele. Následná doména byla staticky zvolena.

V prvotním návrhu byla tato funkcionalita rozdělena do dvou funkcí. Jedna sloužila k zaslání e-mailu hráči a druhá reprezentovala odpověď hráče. Šlo tedy pouze o záměnu polí *From* a *To*. Aby se předešlo zbytečnému opakování kódu, byl přidán parametr *type*, označující odesílatele zprávy. Následně proběhne vyhodnocení jednoduché podmínky. Nakonec po naplnění všech částí je zpráva odeslána na předdefinovaný server.

Výpis 4.4: Odeslání zprávy pomocí definice proměnných

```
1 #greetings
2 subject = "Vítej v mailhogu"
3 body = "<h3>Vítej nový uživateli,</h3> děkujeme, že
4 využíváte naše služby!"
5 sendFrom = "support@mailhog.com"
6 mail_generator(sendFrom,subject,body)
```

Ukázka první zaslané zprávy do e-mailového klienta pomocí metody první 4.4. Zde jsou vytvářeny jednotlivé proměnné pro tělo zprávy, předmět zprávy a odesílatele.

Výpis 4.5: Odeslání zprávy pomocí přímého definování parametrů

```
1  mail_generator(sendFrom="Bob@atlas.com",
2  subject="volné kapacity v týdnu",
3  body="Dobrý den, <br><br> bude mít v následujícím
4  týdnu nějaké volné kapacity? Potřebovali bychom zaplnit
5  směny ve čtvrtek a pátek, hodně lidí je marod.
6  <br> Předem díky! Hezký den Bob")
7
8  sleep(randrange(20))
9
10 mail_generator(sendFrom="Jurkova@vut.cz", subject="Navýšení s
```

Metoda zaslání e-mailu pomocí přímého vložení řetězce do parametru v rámci volání funkce. Pro přidání na věrohodnosti je, jak lze vidět na řádku 8 u 4.5, vložena funkce sloužící k pozastavení provádění skriptu na náhodně daný počet sekund.

O náhodnost se zde stará funkce *randrange*, která generuje náhodné číslo ze zadaného rozsahu.

4.4.2 Tvorba podvržené stránky

Z počátku je nutné danou stránku napodobit, jedním ze způsobů je využití aplikací zaměřených na klonování stránek, mezi něž patří také open-source phishing toolkit GoPhish. Na levé straně se nachází nabídka jednotlivých šablon, které lze pro phishing využít. V tomto případě se jedná o webovou stránku, landning page. Pomocí import site, jak lze vidět na obrázku 4.3, je nutné vložit odkaz na vybrané stránky, které je potřeba naklonovat, tedy `https://www.vut.cz/studenti/stipendia`. Po dokončení však výsledek není ideální, neboť zde chybí ikony a je zřejmé, že stránka byla naklonována viz obrázek 4.4.

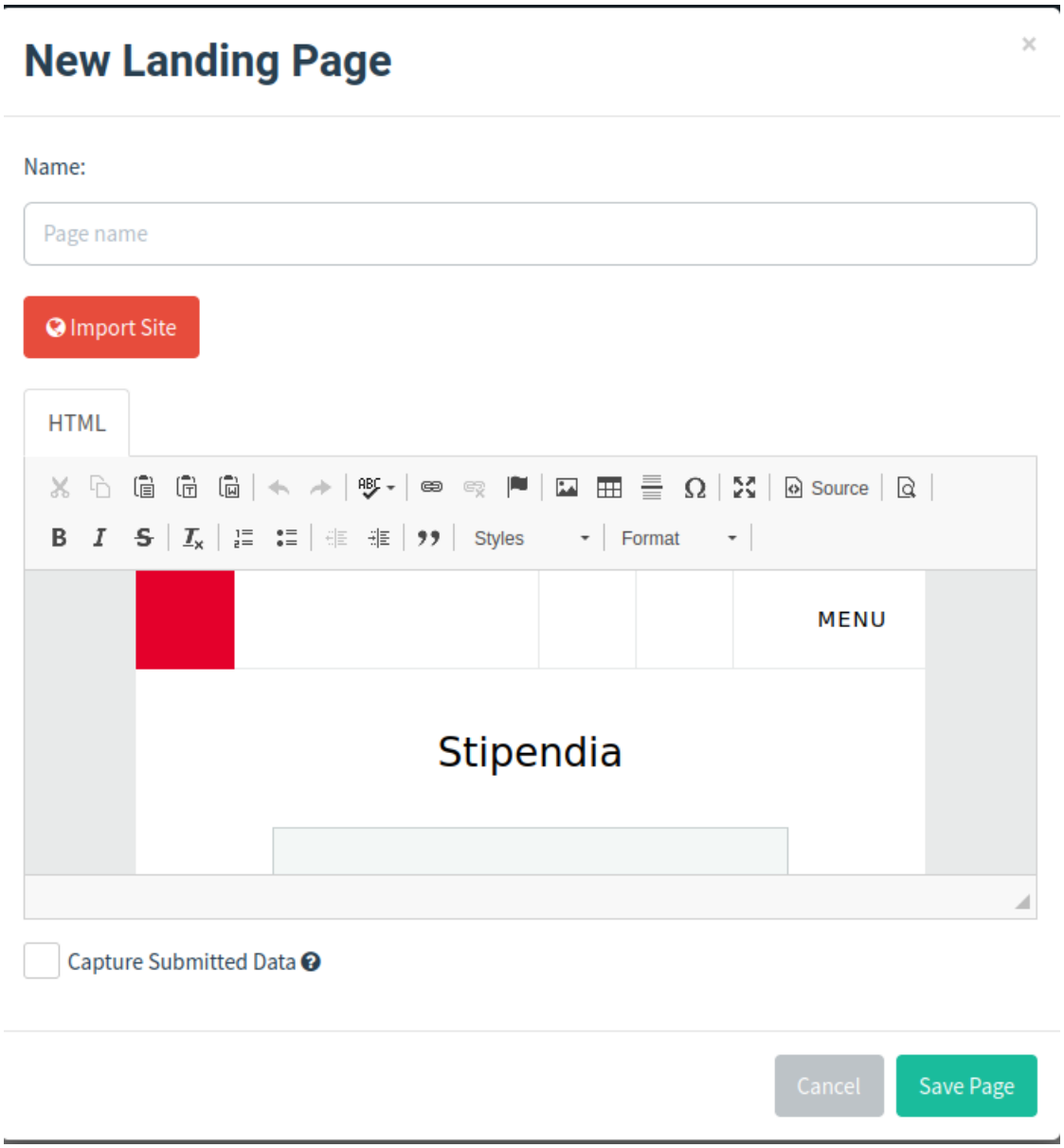
Můžeme si povšimnout, že v menu chybí jak logo, tak také i ikony sociálních médií v patičce webu.

Použitím on-line grafického softwaru Photopea byly vytvořeny ikony pro podvržený web. Inspirací pro jejich design byla již existující loga, která byla mírně upravena, aby se odlišila od původních.

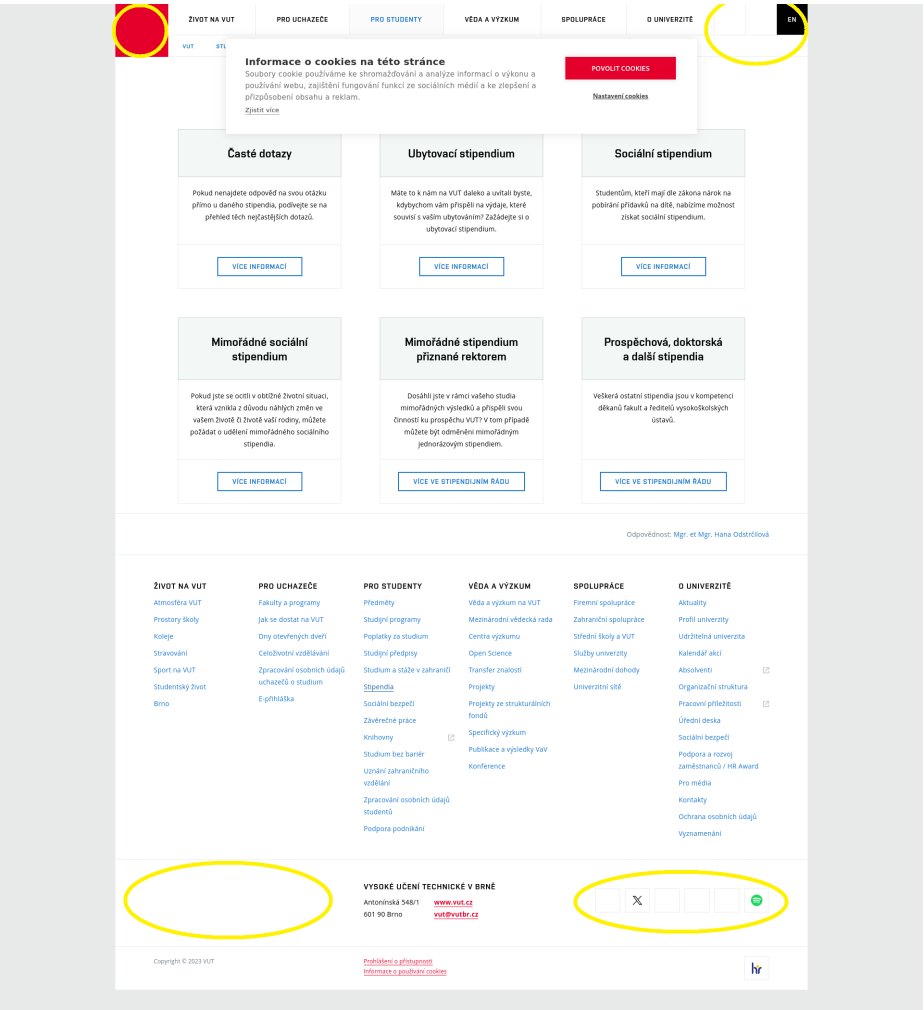
Další částí tvorby tohoto webu, na kterou je nutné se zaměřit je správné vložení obrázku. Samotný nástroj GoPhish umožňuje vkládání obrázků, avšak ani po vložení ikony s požadovanými rozměry se menu nezobrazuje dle očekávaných představ. Zmizí položky menu, které by se museli zpětně přidat ručně. Tento editor stránek je možná uživatelsky přívětivý, ale výsledky nikoliv viz obrázek 4.5. Lepším způsobem je úprava zdrojového kódu stránky, což GoPhish také umožňuje. Pro redukci potřeby přepínání mezi kódem a náhledem je nejlepším způsobem využít možnosti webového prohlížeče samotného. Pomoci nástroje pro vývojáře lze zobrazit a upravovat kód stránky. Tento postup pracuje na podobném principu jako vývojové prostředí, ale výsledky se zobrazí pouze u klienta, pro ostatní změny nejsou viditelné. Kód je možné dle představ upravovat a následně změněné prvky převést do zdrojového kódu podvržené stránky.

Obrázky, které se na webové stránce zobrazují, bývají uloženy v knihovně na serveru, na kterou se následně v kódu odkazuje (přes cestu k obrázku). Další z možností je převod obrázku na URL adresu pomocí base64, tímto způsobem je možné vložit obrázky přímo do kódu. Jednou z výhod je snížení počtu dotazů, což vede k rychlému načtení stránky. Obrázek lze zakódovat pomocí volně dostupného nástroje *Atatus*.

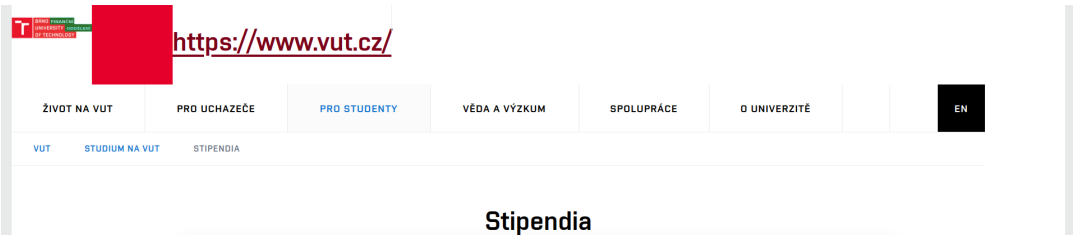
Aby web působil co nejdůvěryhodněji, je vhodné změnit také odkazy na samotnou stránku. Pokud bychom tak neučinili a uživatel by přešel na stránku *stipendia* v menu či drobečkové navigaci byl by odkázán na reálné stránky VUT. Stejně tak je nutné vytvořit stránku pro odesílatelku e-mailu, která je na stránce zmíněná. Je



Obr. 4.3: Ukázka prostředí pro tvorbu klonu stránky



Obr. 4.4: Nedostatečně upravená stránka



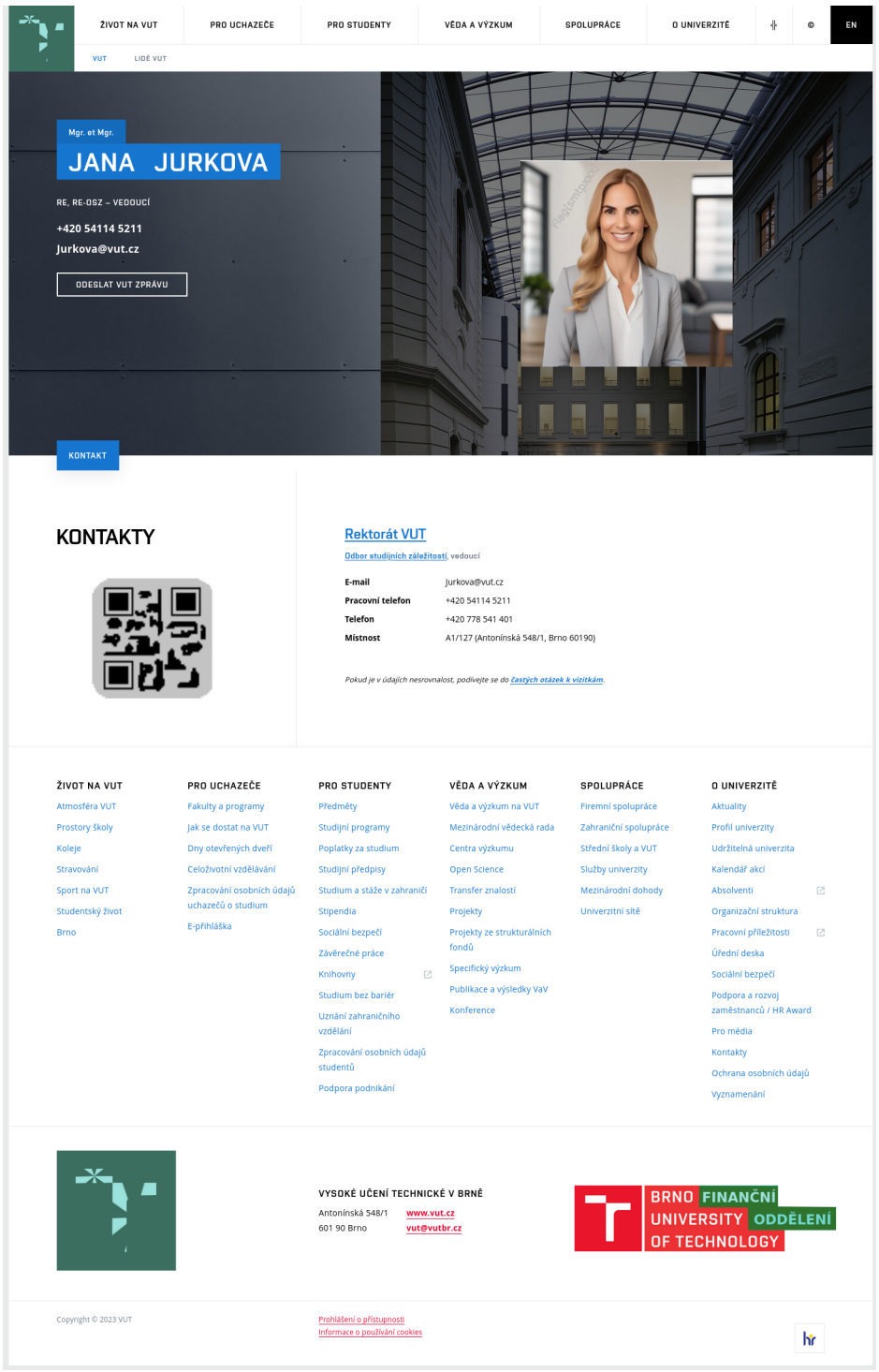
Obr. 4.5: Vložení ikony pomocí editoru

třeba naklonovat vizitku na stránce lidé a upravit zde jméno a e-mailovou adresu, jak lze vidět na obrázku 4.6. Dále je vhodné odstranit cookies lištu, která se po potvrzení znovu vyvolá. To by mohlo vést k podezření, že se jedná o falešný web nebo přinejmenším nefunkční. Nyní web vypadá dostatečně důvěryhodně.

4.4.3 Infikovaný soubor

Phishingový e-mail se snaží oběť nalákat na vidinu zisku z poměrně nenákladné akce. V rámci stipendijního řádu bylo usneseno zpětné navýšení částky u sociálního stipendia. Uživatel má pouze zkontrolovat vyplněný formulář a případné nedostatky odstranit. Následně mu bude zaslána navýšená částka. Formulář se na webu tváří jako obyčejný excel soubor, avšak pokud si všímavý uživatel před stažením zkontroluje odkaz, zjistí, že se jedná o soubor s makry. V případě verze pro LibreOffice se jedná o .ods, což je pro tento software typická přípona. Nyní nastává otázka, proč se na webu vyskytují dvě verze souboru. Jedním z důvodů je škálovatelnost útoku. Ne všichni uživatelé mají licenci Microsoft Office, najdou se i tací co z tohoto důvodu využívají volně dostupný software. Dalším důvodem je využití nástroje pro analýzu souborů.

Oba soubory se chovají téměř totožně. Jediným rozdílem je čas spuštění skriptu. Pro Microsoft Office se skript spustí hned po otevření souboru, pro LibreOffice se spustí po uzavření souboru. PowerShell skript, jehož volání je skryto v makru sešitu, má za úkol rozeslat zprávu všem kontaktům. Pro tuto demonstraci útoku byla zvolena metoda odeslání mailu pomocí příkazové řádky viz výpis 4.6. Nejprve je vytvořena instance objektu, který umožňuje manipulaci za pomocí příkazové řádky. Následně je skript spuštěn s těmito parametry: *-noExit*, zabraňující ukončení po vykonání příkazu, *-c* slouží ke spuštění příkazu. Dále je využíváno pole obsahující emaily, které byly použity v předchozí komunikaci, aktuální uživatelské jméno a následně pomocí cyklu je rozeslána zpráva na definovaný SMTP server a port.



Obr. 4.6: Ukázka upravené vizitky na podvržených stránkách.

Výpis 4.6: Odeslání zprávy s využitím jazyka VBScript

```
1  Sub Main
2  Dim oShell
3  Set oShell = CreateObject("WScript.Shell")
4  oShell.Run "powershell.exe -noExit -c ""$sendTo =
5  'JohnDoe@yahoo.com', 'mom@volny.cz','recipient@test.biz',
6  'admin@vut.cz', 'Bob@atlas.com','Alice@seznam.cz';
7  $username = $Env:UserName ;
8  For ($i = 0; $i -lt $sendTo.length; $i++) {
9  Send-MailMessage -To $sendTo[$i]
10 -From $username'@mailhog.com'
11 -Subject 'Novy iphon uplne zadarmo, to vazne!??'
12 -body 'Pomoct mi vyhrat novy iPhone, staci kdyz mi na
13 nasledujici strance vyplnis dotaznik a uvedes
14 me jako pritele, kterz te do souteze pozval!
15 Do slosovani pak budes zarazen i ty!! Prosim $sendTo[$i] :)'
16 -SmtpServer 'localhost' -Port 1025;}", 1, True
17 End Sub
```

4.4.4 Apache server

Na straně útočníka, Kali Linux, běží softwarový webový server Apache. Pro jeho spuštění byla nejprve nutná instalace serveru a následné přidání HTML souborů do složky */var/www/html*. Odkaz v podvodném emailu odkazuje na URL adresu *http://10.10.10.11/stipendium.html*, jde tedy o útočníkovu adresu. Díky tomu je možné klientům, virtuálním strojům, poskytnout požadovaný webový obsah.

4.4.5 Analýza souboru

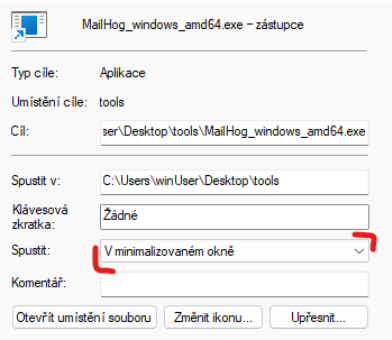
Uživatel má zde dvě možnosti. Může projít touto úlohou a na vlastní kůži okusit jaké to je být obětí phishingu nebo se může snažit útoku předejít. Úloha obsahuje několik indicií. Snadná vidina zisku, odkazy vedoucí na webové stránky, které neodpovídají tomu, co bylo původně uvedeno ve zprávě a odlišné formáty souborů. Dalším prvkem, který může uživatel využít je nástroj *olevba*, ten slouží k analýze souborů MS Office. Dokáže detekovat makra a přímo zobrazit kód nacházející se v nich. Dále dovede překládat jednotlivé obusfukace jak lze vidět na obrázku 4.7. Uživatel zjistí, co dané makro provádí, aniž by soubor otevřel.

Obr. 4.7: Nástroj olevba s infikovaným souborem na vstupu.

4.4.6 Virtuální stroj oběti – Windows 10

Pro simulaci mailového klienta využívaného v popsaných úlohách byla použita aplikace Mailhog, která je převážně využívána v rámci testování e-mailů vývojáři. V našem případě je vhodným kandidátem z toho důvodu, že se jedná o izolované prostředí, ve kterém můžeme útoky krásně simulovat. Tento jednoduchý SMTP server po spuštění naslouchá na portu 1025 na, který je možné zaslat požadovanou komunikaci. Dále obsahuje webové rozhraní na adrese *localhost:8025*. Pro systémy Windows je možné stáhnout aplikaci z dostupných stránek [3].

Bylo potřeba zde vyřešit situaci spuštění aplikace Mailhog a zmíněného Python skriptu. Existuje více řešení, jak tohoto dosáhnout, a to například vložení zástupce těchto aplikací do složky po spuštění. Prvotně se to zdálo jako ideální způsob řešení tohoto problému, avšak ne tak docela tomu bylo, neboť Mailhog po spuštění zobrazuje konzolové okno, kde je možné vidět akce probíhající na tomto serveru. Stejně tak samotný Python skript po spuštění toto okno zobrazuje, avšak jen po dobu jeho průběhu. V případě, kdy by uživatel jedno z oken buď i nedopatřením zavřel, vedlo by jeho konání k ukončení relace s SMTP serverem nebo nedokončení akcí skriptu. Tento problém by šel vyřešit jednoduše tím, že zástupci definujeme, aby se spustil v minimalizovaném okně, jak lze vidět na 4.8, to však vede pouze k tomu, že sice uživatel na první pohled toto okno nevidí, avšak stále jej může ukončit, neboť se nachází na spodní liště stroje. Otázkou nyní nastává, jak provést



Obr. 4.8: Zástupce programu Mailhog.exe.

tyto úkony na pozadí. Jednou z doporučovaných možností je instalace aplikace jako služby. K tomu je však většinou potřeba software třetích stran jakožto nástroj pro správu služeb NSSM nebo skriptovací jazyk AutoHotkey. Oba jsou sice bezplatné, ale hledala jsem způsob, jakým tohoto docílit jen pomocí nástrojů, které poskytuje samotný Windows.

Řešením tohoto problému je využití síly PowerShell, který umožňuje spuštění procesu na pozadí.

[3] `https://github.com/mailhog/MailHog/releases/tag/v1.0.1`

Výpis 4.7: Spuštění programu Mailhog spolu s Python skriptem

```
$exeFiles = @(
    "MailHog_windows_amd64.exe",
    "se_ctf.exe"
)

foreach ($exeFile in $exeFiles) {
    $exePath = Join-Path -Path (Join-Path -Path
    $env:USERPROFILE -ChildPath "\Documents\tools")
    -ChildPath $exeFile
    $process = Start-Process -FilePath $exePath
    -WindowStyle Hidden -PassThru
    Start-Sleep -Seconds 1
}
```

Jelikož se snažím spustit dvě aplikace, je výhodnější toto provést pomocí volání stejného příkazu ve smyčce, než vytvářet duplicitně stejnou část. Abychom zde neuváděla pevně danou cestu, myšleno ve smyslu „C:\Users\...“, využívám proměnné *$env:USERPROFILE*, která obsahuje cestu k profilovému adresáři aktuálně přihlášeného uživatele. K té následně přidám cestu k podadresáři, ve kterém se soubory nachází, spolu s jejich názvem. PowerShell spustí jednotlivé spustitelné soubory z pole *exeFiles* v samostatných procesech. Parametr *-WindowsStyle Hidden* zajistí běh programu na pozadí, parametr *-PassThru* vrací samotný objekt procesu.

Vytvořený PowerShell skript však nemůžeme pouze vložit do složky po spuštění, neboť se nejedná o spustitelný soubor. Je tedy nutné nějak zajistit jeho spuštění automaticky. V tuto chvíli přichází na řadu dávkový soubor, který vykonává sérii definovaných příkazů a jedná se tedy o spustitelný soubor, který obsahuje pouze jeden řádek kódu podobající se zápisu CronJob.

Výpis 4.8: Dávkový soubor start.cmd

```
PowerShell
%USERPROFILE%\Documents\tools\startSetUp\autorun.ps1
>> "%TEMP%\StartupLog.txt" 2>&1
```

Příkaz spustí skript *autorun.ps1* a přesměruje výstup do dočasného logovacího souboru *StartupLog.txt*. Aby vše výše zmíněné bylo proveditelné, je nejprve potřeba zjistit informace o politikách spuštění pomocí příkazu *Get-ExecutionPolicy*, Pokud je odpovědí *Restricted* je třeba tuto politiku změnit následovně *Set-ExecutionPolicy Unrestricted*. Je nutné zde však upozornit, že tato změna politiky umožnuje spouštění skriptů nehledě na jeho původ. Tím se může zvýšit riziko hrozeb, proto toto nedoporučuji v běžném životě.

4.4.7 Reakce na zpětnou vazbu

Jedním z komentářů v rámci zpětné vazby bylo upozornění na „cmd okno“ objevující se hned po spuštění virtuálního stroje a přebývá zde několik minut. Řeč je o oknu dávkového souboru, který spouští aplikaci Mailhog. Bylo by vhodné, kdyby vše, co soubor provádí, probíhalo skrytě na pozadí. Toho lze docílit vytvořením nového souboru, který by skrytě spouštěl dávkový soubor, což je již zbytečně komplikované, proto jsem opustila od dávkového souboru a nahradila jsem jej za VBScript, který za pomoci WSH, tedy Windows Script Host, umožňuje spouštět aplikaci na pozadí. Bylo nutno převést příkaz do podoby VBScriptu. Ten vypadá následovně 4.9

Výpis 4.9: Ukázka spuštění za pomoci VBScript

```
1 Set WshShell = CreateObject("WScript.Shell")
2 WshShell.Run "powershell.exe -nologo -command ""& { &
3 '%USERPROFILE%\Documents\tools\startSetUp\autorun.ps1'
4 }""", 0, False
5 Set WshShell = Nothing
```

Nejprve je vytvořena instance umožňující interakci s operačním systémem. Za pomoci metody *WshSell.Run* je spuštěn PowerShell příkaz, kde přepínač *-nologo* zaručuje, že nebude zobrazeno jeho logo při startu. Následuje samotný příkaz, nulový parametr, jenž určuje styl okna, kde 0 představuje skryté okno a logická hodnota, která udává, zda má skript čekat na dokončení programu.

Pokud dojde k selhání, jedním z možných řešení je přímé spuštění aplikací, které se nachází ve složce \Documents\tools na virtuálním stroji, nebo ruční spuštění PowerShell skriptu *autorun.ps1* ve složce \Documents\tools\startSetUp.

4.4.8 Zástupce aplikace Mailhog

Aby uživatel nemusel složitě v rámci hry zadávat adresu webové rozhraní této aplikace, tedy *localhost:8025*, je na ploše vytvořen zástupce obsahující tuto adresu, kterému byla vytvořena speciální ikona pro snazší orientaci hráče, jak můžeme vidět na 4.9.

Obr. 4.9: Zástupce spolu s vytvořenou ikonou.

4.5 Útok čtvrtý – podvody na bazaru

Tento segment si dává za cíl seznámení uživatele s volně dostupnými prostředky, díky kterým je možné se vyhnout podvodům v této oblasti. V rámci zkoumání této problematiky jsem narazila na webové stránky zaměřující se právě na podvody na bazaru. Nachází se zde nejenom databáze podvodných účtů, ale také zajímavé články a rady. Jelikož se tato tématika týká každého z nás, přišlo mi vhodné, na tento web poukázat [4]. Ve spojení s tímto scénářem bylo nejvhodnější využít databáze obsahující čísla účtů, poněvadž se jedná o citlivý údaj, bylo zapotřebí vytvořit podobnou stránku se stejným účelem.

4.5.1 Tvorba stránky

Nejprve bylo potřeba stránku naklonovat pomocí klonovacího nástroje GoPhish. Jde o databázi obsahující informace o nahlášených podvodech, ta se skládá z několika prvků obsahující číslo účtu, jméno případně telefon. Abych si práci zautomatizovala a nemusela jsem vše složitě vypisovat staticky, vytvořila jsem skript v jazyce JavaScript, který tyto prvky náhodně vytváří.

Jméno je generováno náhodně skriptem, který obsahuje dvě pole. Jedno je naplněno jmény naopak druhé obsahuje možná příjmení. Prvky jsou zde poskládány tak, aby byly vždy elementy ženského rodu na sudém indexu, což zjednodušuje následné párování jména s příjmením. Stačí totiž provést kontrolu zbytku po celočíselném dělení, jak lze vidět na 4.10.

Výpis 4.10: Výběr indexu

```
1 index = Math.floor(Math.random() * 3);
2 name = Math.floor(Math.random() * 7); // 7 prvků v poli
3 if (name % 2 == 0) {
4     surname = Math.floor(Math.random() * 7 / 2) * 2;
5 } else {
6     surname = Math.floor(Math.random() * 7 / 2) * 2 + 1;
7 }
```

Nejprve je tedy získán index jména, od kterého se následně odvíjí přiřazení indexu pro příjmení.

Při tvorbě čísla účtu je využíváno možností textového řetězce, kde je v cyklu přikládána náhodně vygenerovaná hodnota v rozsahu 0–9.

[4] https://podvodnabazaru.cz/database/list

Výpis 4.11: Přirazení prvků

```
1
2 if (type == 1) {
3     items[i].innerHTML = text + "/" + kod[index] + " " +
4     jmena[name] + " " + prijmeni[surname];
5 } else {
6     items[i].innerHTML = text + "/" + kod[index];
7 }
```

Po vyhodnocení indexu jsou přiřazeny konkrétní prvky z polí, viz 4.11.

Vyhledávací pole je řešeno formou formuláře, který po potvrzení volá funkci *search*. Ta obstarává nalezení vloženého řetězce či jeho podřetězce. Funkcionalita této funkce bude dále popsána.

Hodnota vyhledávacího pole, která je získána pomocí unikátního identifikátoru, je testována na bílé znaky, které pokud je obsahuje, odstraní. Hledaným řetězcem je nyní číslo účtu uvedené ve scénáři. Aby bylo možné najít nejen celý řetězec, ale i jeho části, byla vytvořena funkce *findSubstrings*, která vytvoří všechny kombinace posloupnosti znaků plynoucí z rozdělení této proměnné obsahující číslo účtu. Konkrétně jde o vnořené cykly, které řetězec rozdělí na menší části a ty vloží do pole. Funkce *compareStrings* následně porovnává hodnotu vyhledávacího pole s daným účtem. V případě splnění podmínky dojde k otevření *name.html*, obsahující flag, v novém okně. Pokud podmínce nevyhovuje, dochází ke kontrole podřetězce voláním funkce *findSubstrings*, je-li nalezen, zobrazí se upozornění s odkazem na zmíněnou stránku. V opačném případě se zobrazuje chybová hláška.

4.6 Útok pátý – Falešný profil

Tento úkol navazuje na předchozí a zabývá se problematikou podvodů na internetových bazarech.

Stejně tak by se lidé měli zaměřit na validaci profilu. Často se setkávám s nabídkami již od pohledu falešných profilů, které používají veřejně dostupné fotky. Proto se v navazující části tohoto úkolu zaměřuji na obrázkový vyhledávač *Tineye*, který umožňuje jejich vyhledávaní, a poskytuje informace o jejich umístnění na internetu.

4.6.1 Scénář

Součástí těchto úkolů, čtvrtého a pátého, je také PDF obsahující doporučení, kterými by se měl uživatel řídit při nákupech na internetu, které byly vytvářeny za pomocí článků dostupných na tomto webu. Tento dokument obsahuje vodítka k jednotlivým klíčům, konkrétně odkazy na uvedené webové stránky.

4.7 Útok šestý – podvržené stránky

V této části se ve scénáři posouváme k tvorbě podvržených stránek legitimní zásilkové služby spolu s falešnou platební bránou. Tímto úkolem se snažím upozornit na nekalé praktiky uživatelů na internetových bazarech, kteří se v oběti snaží vzbudit falešný pocit důvěry, například tím, že sami byli obětí podvodu. V blízké minulosti bylo právě toto trendem k zisku citlivých údajů.

4.7.1 Tvorba podvržené stránky

Za tímto účelem jsem vybrala stránky České pošty, konkrétně formulář [5], který slouží k odeslání zásilky, což do daného scénáře perfektně zapadá. Jedná se totiž o konverzaci mezi hráčem, tedy Evou, a zájemkyní Annou, která má zájem o nabízené zboží. Tento formulář již vyplnila a zaplatila cenu jak za zboží, tak za dopravu. Nyní pouze stačí si vybrané peníze na zadaném odkaze vybrat. Opak je však pravdou, a pokud uživatel poskytne pravé údaje, tedy kartu Evy ze scénáře, dojde ke stržení částky ze zadané kreditní karty, tedy fiktivního konta. Z tohoto důvodu mi přišlo ideální využít existujícího formuláře České pošty, neboť potencionální oběť může nabýt falešné důvěry, jelikož tuto službu reálně Česká pošta nabízí.

Po uložení stránky ve formátu HTML bylo potřeba vyplnit formulář, aby odpovídal scénáři. Vše má být totiž nachystáno a hráč si má jen vyzvednout své peníze. Jednotlivé prvky formuláře proto obsahují předvyplněné hodnoty *value*. Hlavním prvkem tohoto úkolu je následující stránka *gate.html*, ta obsahuje formulář představující platební bránu, kde má uživatel zadat číslo své karty, pin a další důležité údaje. Na první pohled má být hráči zřejmé, že nejde o oficiální platební bránu.

4.7.2 Zpracování dat

Po přesměrování na stránku *gate.html* je potřeba zpracovat a validovat zadaná data, proto byla vytvořena funkce v jazyce JavaScript, která toto obstarává.

Funkce *isCardValid* ověřuje, zda byla zadaná platná kreditní karta. Validace probíhá pomocí Luhnova algoritmu ve třech krocích. Přijatý řetězec obsahující číslo karty nejprve otestuji, zda neobsahuje nežádoucí znaky, jako mezery nebo lomítko. Pokud ano, je tento znak odstraněn a přistupuje se k prvnímu kroku. Zde je úkolem rozdělit řetězec dle pořadí na sudá a lichá čísla. Luhnův algoritmus počítá prvky zprava, proto je nutné přijatý řetězec reverzně přeskládat. V případě sudých čísel je počáteční pozice na indexu 1 a provádí se postupně posun o dvě pozice. Liché naopak začínají na indexu 0. Pokud jde o číslo, je vloženo pomocí *push* do příslušného pole.

[5] `https://www.postaonline.cz/odvozy/odvozbaliku/parametrybaliku`

Nyní existují dvě pole. Jedno obsahuje sudé prvky, naopak druhé prvky liché, to znamená, že je možné se přesunout ke druhému kroku. Je potřeba provést součet prvků v těchto polích. V případě lichých čísel zde nenastává problém a stačí jednoduše pomocí cyklu pole projít a jednotlivé prvky sečíst. Pokud se jedná o pole sudých čísel, je nejprve potřeba získat dvojnásobek prvků v tomto poli, avšak v tomto místě narážíme na problém, kdy, pokud je součet těchto prvků dvojciferné číslo, je potřeba provést ještě sčítání číslic, které vede k zisku jednociferného čísla. Jak tohoto dosáhnout v jazyce JavaScript? Rozhodla jsem se využít síly textového řetězce, který můžeme jednoduše rozdělit pomocí funkce *split*. Výstupem této funkce je pole obsahující jednotlivé cifry jako elementy na odpovídajících indexech, což usnadní jejich součet.

Posledním krokem algoritmu je kontrola finálního součtu, kde je potřeba sečíst sumu lichých a sudých čísel. V případě, že výsledek končí nulou, jedná se o validní kartu.

Validace data probíhá pomocí funkce *isDateValid*. Karta obsahuje datum ve formátu MM/YYYY nebo také MM/YY. Uživatel tedy může zadat tyto vstupy, je proto nutné funkci na toto připravit. Jelikož se jedná o řetězec, je možné jej rozdělit pomocí funkce *split*. Následně stačí jen provést kontrolu roku, přesněji jeho délky. Voláním metody *getFullYear* objektu třídy *Date* je možné získat celý rok, tedy ve formátu čtyřciferném. Pro zisk formátu dvojciferného je potřeba převést rok na řetězec a pomocí metody *substring* odstranit přebytečné cifry.

Jelikož tento skript také provádí kontrolu, zda uživatel zadal informace o své kartě, která je součástí scénáře, bude po uplynutí doby platnosti karty, tedy po roce 2030, potřeba upravit vyhodnocení podmínky obsahující *dateOnCard == '07/2030' || dateOnCard == '07/30'* na nový datum, neboť po roce 2030 bude systém považovat tuto kartu za neplatnou a z tohoto důvodu neprovede odesílání informací PHP skriptu, neboť ten je kontaktován pouze v případě validních údajů.

4.7.3 Zaslání e-mailu

Po validaci údajů se přechází k zaslání e-mailu do schránky uživatele. Toto však již nespadá do kompetence JavaScriptu, proto je nutné zaslat informace o validaci na backend, kde probíhá samostatné zaslání e-mailu, o které se stará PHP skript.

Funkce *sendDataToPHP* obstarává komunikaci pomocí API, která zasílá data na server dotazovací metodou POST. Její výhodou je, že přenášená data jsou součástí HTTP dotazu a tím pádem nejsou viditelná.

Pro odesílání e-mailů na straně PHP bylo potřeba využít knihovnu PHPMailer, která má integrovanou podporu SMTP. Tato knihovna se nachází ve stejném adresáři jako samotný PHP skript. Po zahrnutí této knihovny následuje vytvoření

instance, která je následně využívána ke konfiguraci e-mailu. V tomto případě je potřeba zaslat e-mail do aplikace Mailhog, tedy na adresu localhost s portem 1025, to znamená, že je potřeba zjistit IP adresu daného klienta. V tomto případě, kde jsou využívány virtuální stroje, je možné využít statickou IP adresu. Pokud by bylo nutné dynamicky přiřadit veřejnou IP adresu klienta připojujícího se k serveru, je možné k jejímu zjištění využít asociativní pole *$_SERVER['REMOTE_ADDR']*.

V konečné fázi dochází k plnění samotného předmětu a těla e-mailu. Textace se odvíjí od splnění podmínky. Součástí scénáře je platební karta hráče, která byla náhodně vygenerována tak, aby splňovala dané náležitosti. Pokud uživatel zadá tuto karu a spolu s ní všechny informace s ní svázanou, je do schránky zaslána zpráva informující o stržení určité finanční částky z této karty. Zadá-li pouze číslo této karty a zbylé údaje smyšlené stále zaslaná zpráva nebude obsahovat klíč, neboť tato úloha klade důraz na prověření stránky. Správným postupem má být otestování platební brány vygenerováním falešné karty.

4.8 Útok sedmý – baiting

Tato úloha cílí na seznámení se s metodou sociálního inženýrství zvanou baiting. Ta se snaží uživatele nalákat ke stažení software, který se zdánlivě tváří jako bezplatný, ale ve skutečnosti obsahuje spyware špehující uživatele. Dá se říci, že jde o výměnný obchod soukromí na úkor penězům.

Pro vytvoření této úlohy bylo nejprve nutné vytvořit spyware, který bude sbírat informace o uživateli. Nejvhodnějším kandidátem pro tento účel je keylogger, který po spuštění skrytě na pozadí sleduje stisknuté klávesy a ty následně odesílá na server.

4.8.1 Server

Skript na straně serveru má za úkol navázat spojení na zadané adrese a portu. Následně naslouchá a přijímá příchozí komunikaci.

K zdokonalení této úlohy jsem zaslaná data zobrazila na hlavní stránce serveru, tedy po zadání jeho IP adresy do prohlížeče se zobrazí stránka se stisknutými klávesami. Pro docílení tohoto bylo potřeba nejprve zaslaná data uložit do souboru, což provádí samotný python skript. Samotné zobrazení na stránce má na starost PHP skript, který zadaný soubor otevře a jeho obsah vypíše.

V okamžiku, kdy uživatel aplikaci obsahující tento spyware na straně klienta, uzavře je spolu s ním uzavřeno také spojení klienta a serveru, což je značný problém, jelikož server musí neustále naslouchat a je tedy potřeba zajistit, aby se neustále spojení kontrolovalo, k tomu byl využit softwarový démon Cron, který spouští příkazy v určitý časový okamžik. Za tímto účelem byl vytvořen Bash skript

(server_start.sh), který má za úkol spustit server v případě, že došlo k jeho ukončení. To provádí následujícím způsobem. Nejprve probíhá vyhledání v rámci procesů na základě vzoru, kterým je jméno python skriptu. V případě, že tento proces neběží, je spuštěn.

Tento skript je vložen jako záznam do konfiguračního souboru Cron, který je kontrolován každou minutu.

4.8.2 Klient

Jednoduchý keylogger vytvořený za pomocí pythonu může obsahovat jen pár řádků kódu. V takovém případě však data vypisuje buď přímo do konzole, nebo přidáním pár řádků kódu můžeme tyto informace zapisovat do textového souboru. V případě, že chceme stisknuté klávesy odesílat na server, je potřeba zajistit komunikaci mezi těmito zařízeními. Způsoby, kterými toho docílit bude více, jako například pomocí API. V mém případě jsem zvolila možností, které nabízí Python, a to konkrétně síťové komunikace za pomocí soket. Pro začátek je nutné definovat adresu spolu s portem, na kterém server naslouchá, kromě ní dále definuji typ, konkrétně *AF_INET*, který udává, že se jedná o komunikaci prostřednictvím IPv4. Spojení je následně navázáno pomocí *socket.connect*, kde je ovšem nutné, aby byl server v tomto bodě již dostupný.

Pro úplnou identifikaci zařízení je na začátku serveru zaslána skupina dodatečných informací jako je datum spolu s časem, kdy byl keylogger spuštěn nebo systémové informace.

Důležitou částí samotného odposlechu stisknutých kláves je nastavení Listeneru, kterému definuji funkci *on_press*, která kromě zaznamenávání stisknutých kláves, také počítá čas, který uběhl od poslední stisknuté klávesy. To slouží k sloučení písmen do celků, tedy na výstupu vznikají celá slova nikoliv pouze jednotlivá písmena. Využívám pro to globální proměnné, která se vynuluje vždy po stisku nové klávesy. Dále zde kontroluji čas, po který klávesa stlačena nebyla. V případě, že překročil 3 sekundy, znamená to, že jde o nové slovo, a tedy pokud pole stisknutých kláves není nulové, tedy uživatel již něco v předchozí době stiskl, zasílám obsah tohoto pole na server. V opačném případě se klávesa přidá do pole. Podstatné je také zmínit speciální klávesy, které jsou ve tvaru *Key.tab*, ty se na výstupu chovají odlišným způsobem od písmen, neboť neobsahují uvozující znaky, které jsou před zasláním z pole odstraněny, což způsobovalo značný problém při jejich seskupování. Řešení zde však bylo poměrně jednoduché, které spočívá v tvorbě funkce *replaceKey*, která ve smyčce tyto znaky přidává, což vede k čitelnému výstupu.

V poslední řadě nesmí také chybět ošetření nestisknuté klávesy, neboť v případě, kdy uživatel již další klávesu nestiskne, by nastala situace, že poslední skupina stla-

čených kláves by nebyla na server nikdy zaslána, proto je zde kontrola globální proměnné času, která sleduje, kdy byla naposledy stlačena klávesa. V případě, že tento čas překročí dobu 6 sekund a pole obsahující stisknuté znaky není nulové, jsou tyto znaky zaslány na server.

V rámci scénáře jsem jako návnadu pro uživatele použila Adobe Photoshop, který je poskytován zcela zdarma. Nejprve tedy bylo nutné vytvořit stránky, na kterých se bude tento software nacházet. K tomu byl použit responzivní CSS framework Water, který celému webu zlepšuje estetiku. Pomocí hypertextového odkazu doplněného o atribut download je možné stáhnout instalační soubor software, tvářícího se jako Photoshop.

Kompilace tohoto skriptu na spustitelný soubor byla provedena pomocí nástroje *PyInstaller*, viz 4.12. Příznak *-windowed*, zajišťuje běh aplikace na pozadí, díky příznaku *–onefile* je výstupem pouze jeden spustitelný soubor, dále *–icon* udává ikonu, kterou bude soubor obsahovat a *–name* jméno samotného souboru.

Výpis 4.12: Kompilace skriptu s funkcionalitou keylogger

```
1 python -m PyInstaller --onefile --icon=photoshop.ico
2 --name=Adobe_Photoshop_25.4_windows --windowed
3 clientPlusKeylog.py
```

Po dokončení kompilace je možné vytvořit instalátor tohoto programu. Pro jeho tvorbu jsem použila bezplatný nástroj *Inno Setup Compiler*, ten umožňuje mimo jiné nastavení jazyka instalátoru nebo přidání licenčních podmínek, které zde byly taktéž přidány pro zajištění vyšší míry autentičnosti instalátoru.

4.9 Útok osmý – Adware

Finálním úkolem tohoto scénáře je nalezení klíče neboli flagu, který se skrývá v aplikaci obsahující adware, to je jeden z méně nebezpečných typů malware, který obsahuje vyskakovací okna s reklamou. Jedná se tedy o spíše nepříjemný software, který však může být doplněn i o více nebezpečný typ malware. V této úloze se proto zaměřuji na skrytí adware do obrázku, postup tvorby bude nyní popsán.

4.9.1 Tvorba aplikace obsahující adware

Hlavní ideou při tvorbě tohoto úkolu bylo vytvořit software, který uživatelovi značným způsobem změní vizuální stránku jeho zařízení, tedy uživatelské rozhraní. Stejně tak pro tuto tvorbu byl využit vysokoúrovňový jazyk Python.

Jelikož chci uživateli výrazným způsobem změnit uživatelské rozhraní, proto pracuji s plochou uživatele. V první fázi se snažím přejmenovat všechny soubory a složky

nacházející se právě zde. Na začátku je tedy nutné načíst cestu k tomuto souboru pro aktuálního uživatele, k tomu napomáhá modul *os*, který dokáže komunikovat s nativním operačním systémem, kde právě běží tento skript. Díky práci s proměnným prostředím je možné za pomocí *os.environ* získat cestu k požadované složce aktuálního uživatele pro systémy Windows. Nyní je možné získat všechny dostupné složky a soubory v zadané cestě. Tato funkce získá všechny dostupné soubory, tedy i ty skryté, jako je i *desktop.ini*, ten slouží k ukládání informací a v případě nekorektních úprav může dojít k jeho poškození. Z tohoto důvodu je třeba tento soubor vynechat.

Po načtení všech souborů do listu přichází čas na vytvoření nových jmen pro tyto soubory. Iterací procházím pole, kde pro jednotlivé prvky vytvářím nová jména, o náhodné délce, ze směsice znaků čínské abecedy, čímž je upoutána uživatelova pozornost. Toto jméno je následně uloženo do struktury obsahující další informace o souboru, jako jeho velikost, čas vytvoření nebo původní název. Pomocí funkce *rename_dir_chWall* je následně provedena změna aktuálního jména na nově vygenerované, a to pomocí *os.rename*.

V další fázi na řadu přichází vyskakovací okna. Nejprve bylo nutné každému modálnímu oknu přiřadit název a příslušný text. Textace byly inspirovány známými a pro tento typ malware typickými hláškami. Aby mohly být všechny okna spuštěny najednou, bylo nutné využít vlákna. Ty jsou v rámci cyklu přidány do pole. Po ukončení cyklu přichází na řadu spuštění jednotlivých vláken nacházejících se v tomto poli. Pokud došlo k dokončení všech vláken, přesouváme se do další fáze.

V této konečné fázi dochází ke změně pozadí na obrázek obsahující zadaný klíč, tedy flag. S využitím knihovny *PIL*, je možno vytvářet 2D objekty, které lze použít k tvorbě libovolných obrázků. Předně je potřeba vytvořit takzvaný *Image* o daném rozlišení. Následuje tvorba objektu, který zajišťuje generování různých tvarů jako mnohoúhelník, obdélník, čtverec či kruh. Těmto tvarům je možné nastavit barvu, ale hlavně pozici na plátně. Posledním z objektů zajišťující rendering textu je *ImageFont*, díky němuž je možné do obrázku vložit text obsahující flag. Po dokončení tvorby obrázku je uložen na pracovní ploše uživatele a nastaven jakožto pozadí plochy následovně 4.13.

Výpis 4.13: Změna pozadí pomocí jazyka Python

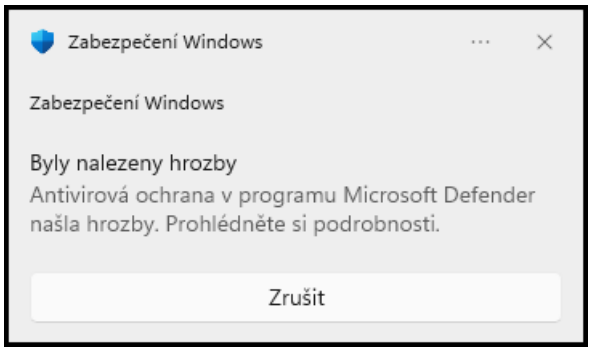
```
ctypes.windll.user32.SystemParametersInfoW(20, 0, path , 0)
```

Jde o funkci systému Windows, jenž povoluje manipulaci se systémovými prvky. Prvním parametrem funkce je *uiAction* udávající, o jakou operaci se bude jednat. Konstanta 20 zde představuje příznak *SPI_SETDESKWALLPAPER*, který slouží ke změně tapety. *UiParam* se používá k zisku dodatečných informací, v aktuálním

případě je nastaven na 0, tedy není použit. *PvParam* definuje cestu k nové tapetě, jenž má být nastavena a posledním parametrem je *fWinIni*, který se zaměřuje na aktualizace, v případě 0 není použit.

4.9.2 Neblahé vlivy v rámci vývoje

V rámci vývoje bylo nutno finální skript převést na spustitelný soubor, jenž zajistí změnu názvu souborů na Ploše. V případě spuštění přímo z příkazové řádky problém nenastával. Pokud byl však spuštěn soubor převedený pomocí PyInstaller na spustitelný, začal se Microsoft Defender bránit. V tomto případě nešlo ani daný skript zkompilovat pomocí příznaků *–windowed* nebo *–noConsole*. Bylo tedy nutné provést kompilaci jen pomocí *–console*. To znamená, že se uživateli zobrazí konzolové okno a aplikace tím pádem neběží sama na pozadí. Bohužel se ani v tomto případě nepodařilo přejmenovat jednotlivé soubory. Chyba *„Operace nebyla úspěšně dokončena, protože obsahuje virus nebo potencionálně nežádoucí software“* nebyla momentálně vyřešena, avšak nijakým způsobem nedopadá na provedení úkolu.

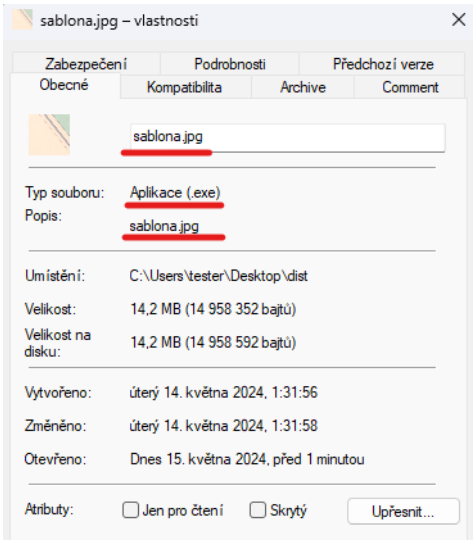


Obr. 4.10: Microsoft Defender při kompilaci.

4.9.3 Skrytí aplikace do obrázku

Aby tento software nebyl tak očividný, rozhodla jsem se ho přidat k obrázku. Vytvořila jsem webovou galerii obsahující několik obyčejných fotografií a jeden speciální obrázek, který se tváří jako obyčejná skica ve formátu PNG, opak je však pravdou. K tomuto obrázku jsem přidala také aplikaci popsanou výše, tedy adware. K tomu jsem využila nástroj pro archivaci souborů *WinRAR*, který dokáže nejen spojit více souborů do archivu, ale také dovoluje vytvořit SFX, tedy samorozbalovací archiv. Stačí tedy vybrat soubory, které chceme k archivu přidat, konkrétně spustitelný soubor obsahující adware, obrázek a ikonu, kterou pro archiv použijeme. Následně je potřeba změnit možnosti SFX archivu, kde definujeme, jaké soubory mají být po extrahování spuštěny, tedy *exeimage.exe* a také *sablona.jpg*, to zajistí jak spuštění

obrázku v programu fotografie, tak také samotného adware. Rozbalené soubory se snažíme skrýt do dočasné složky, aby uživatel neviděl, že se jedná o více souborů, zatím se stále zdá, že jde pouze o obrázek, pokud však tento soubor prozkoumáme, zjistíme, že jde o aplikaci, jak lze vidět na 4.11.



Obr. 4.11: Ukázka spustitelného souboru.

Skrýváním informací se zabývá také vědní disciplína steganografie, program, který tuto technologii využívá, se nazývá *steghide*, který dokáže ukrýt data i do obrázků. Tímto způsobem je možné skrýt jak text, tak i spustitelný soubor po zadání přístupového hesla. Následně je možné skrytá data extrahovat opětovným zadáním hesla. Tato metoda se zdá být efektivnější, program je ovládán jednoduše pomocí konzole, avšak neplní zcela účel, který požaduji. Je sice možné zde spustitelný soubor ukrýt, avšak jej automaticky nepustí. Bylo by potřeba mít například vzdálený přístup k tomuto zařízení a také provést extrakci souboru.

4.9.4 Úprava virtuálního stroje Windows 11

Zabezpečení virtuálního stroje nedovolovalo stažení tohoto souboru z příslušné webové stránky, bylo tedy zapotřebí pozměnit jeho ochranu, což znamenalo permanentní deaktivování aplikace Microsoft Defender, které spočívalo ve vytvoření nové DWORD hodnoty, pomocí editoru registrů, a to jak pro Windows Security Center s cestou *\ Microsoft\ Center\ Feature*, který řídí ochranu, tak i *\ Policies\ Microsoft\ Windows Defender*. Nastavení těchto hodnot na 1 vede k deaktivování antispyware ochrany (DisableAntiSpyware) spolu s automatickou kontrolou disku (DisableAvCheck). Poté následovalo nastavení skupinových politik, kde bylo potřeba potvrdit vypnutí všech

typů ochrany od samotného Microsoft Defenderu, spolu s jeho částmi jako Smart Screen, až po Real-Time Protection.

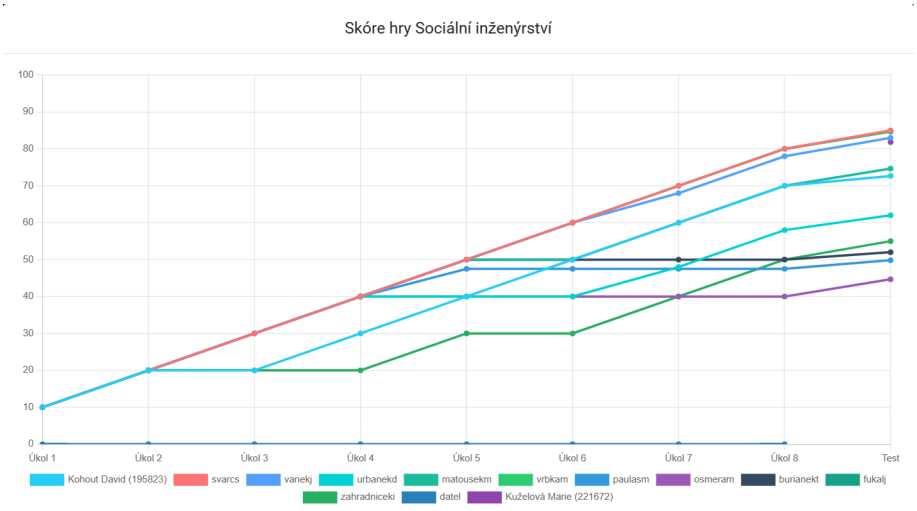
4.10 Testování scénáře

Nezbytnou součástí vývoje scénáře bylo i jeho testování studenty. Těm bylo na úvod předvedeno prostředí a koncept hry.

4.10.1 Testování na SPŠ Třebíč

První testování proběhlo na SPŠ Třebíč. V této fázi byly shromážděny věcné připomínky k scénáři, které byly následně zapracovány. Jednalo se jak o zlepšení formulace scénáře, tak i přidání určitých funkcionalit, jako přidání upozornění při odeslání e-mailu nebo vylepšení vyhledávání na stránce.

I přes drobné nedostatky se studentům v první fázi vedlo velmi dobře, neboť jejich celkové hodnocení bylo nadprůměrné.



Obr. 4.12: Skóre hry testované respondenty na SPŠ Třebíč.

Díky těmto statistikám 4.12 bylo možné zmapovat, které úlohy dělali studentům největší potíže. Jak můžeme vidět, ke zlomu došlo hlavně od úkolu pátého. V rámci tohoto úkolu byl k dispozici dokument PDF, který obsahoval skryté nápovědy. Problém zde působilo to, že studenti na něj byli upozorněni až v rámci jedné z nápověd, což jim značně zkomplikovalo orientaci u tohoto úkolu. Proto byla v rámci reakce na zpětnou vazbu přidána informace o existenci tohoto souboru přímo do scénáře, kde jsou studenti přímo upozorněni, na co se mají zaměřit, tedy na odkazy. Dále byla změněna logika šestého úkolu, kde byla hráči přidána fiktivní platební karta.

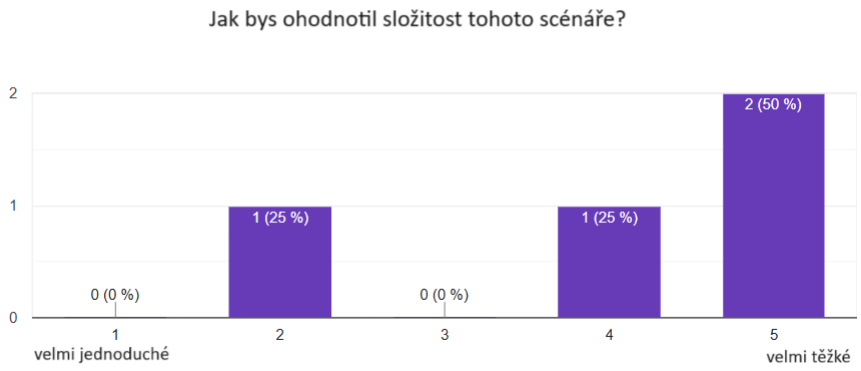
Původním záměrem bylo, aby uživatel vkládal neplatné údaje, tedy nevalidní karty, což nebylo úplně intuitivní.

4.10.2 Testování na gymnáziu

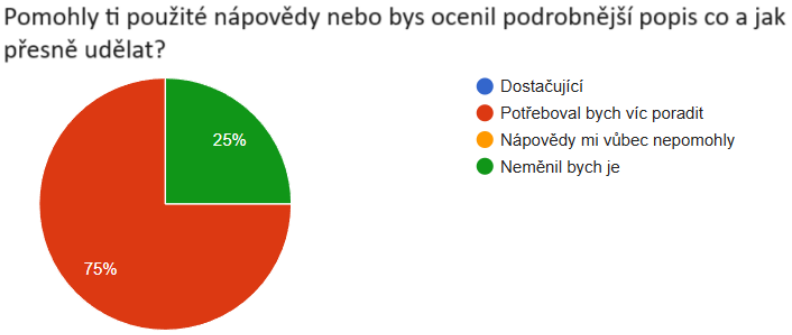
V rámci výuky výpočetní techniky mi bylo umožněno testování scénáře i na gymnáziu, které probíhalo ve dvou etapách. Celkem se zúčastnilo 21 respondentů (7 žáků sekundy, 13 žáků primy a jeden učitel). Kromě jednotlivců byly vytvořeny i dvojice. Jelikož poskytnutý čas byl o něco kratší než čas odhadovaný na průchod scénářem, dostala se až do finálového konce menší skupina jedinců 4.15. Tím pádem nebyl vyplněn samotný dotazník ohledně zpětné vazby, avšak bych řekla, že možnost pozorovat žáky, jak nad úkoly přemýšlí, bylo přínosnější. Průběžně jsem žáky instruovala ohledně postupu, zaměření a významu úlohy. Tedy jsem jim vysvětlovala, co si z dané úlohy mají odnést.

V reakci na tuto zkušenost bych ještě více upravila scénář. Vzhledem k tomu, že se jednalo o žáky základní školy, bylo by vhodné scénář doplnit o podrobnější popis toho, co v úkolu mají provádět. Stejně tak by bylo vhodné doplnit informace o tom, na co se daný úkol zaměřoval. Vhodným řešením by mohlo být doplnění scénáře o stručné a smysluplné shrnutí dané problematiky.

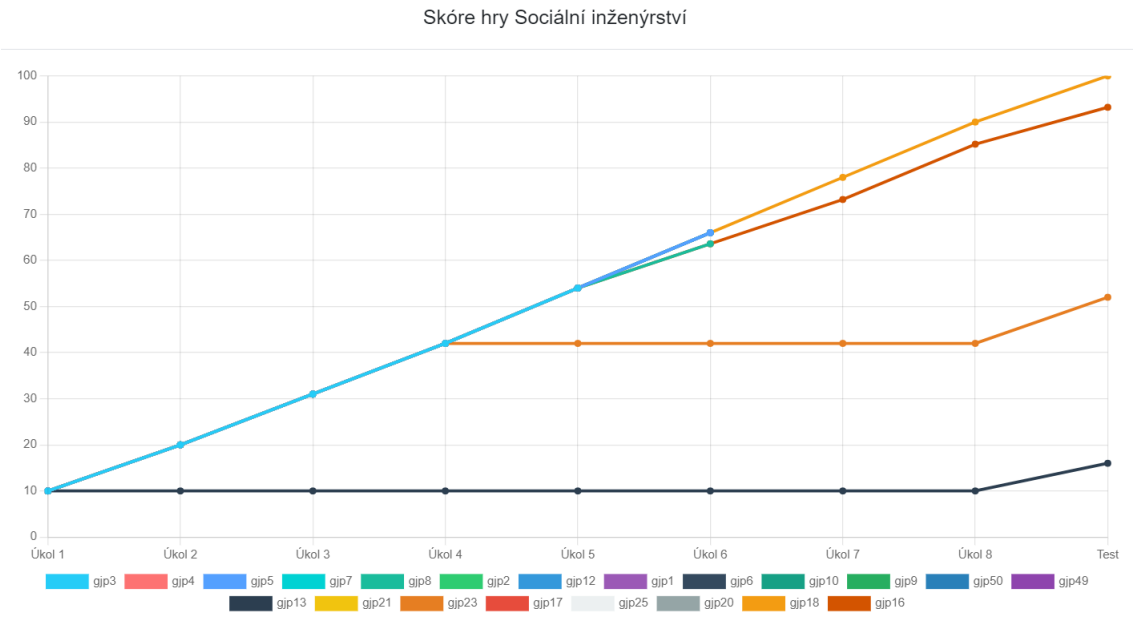
Pro získání zpětné vazby byl vytvořen dotazník. Všechny respondenty tento typ hry zaujal a připadal jim přínosný. Polovina z dotázaných uvedla úroveň složitosti scénáře jako velmi těžkou, jak lze vidět na grafu 4.13. Následující dotaz se zabýval nápovědami a celkovým popisem toho, co má respondent v dané úloze provádět. Většina odpovědí, jak je uvedeno 4.14, se shoduje v tom, že by bylo vhodné poskytnout více rad. Zvláště u formátů řetězce flag byly uživatelé, kteří se s tímto kontextem hry setkali poprvé, mírně zmateni. Dalším významným aspektem při dotazování byla délka hry z pohledu hráče. Všichni oslovení respondenti uvedli, že hra byla příliš dlouhá, a ocenili by kratší variantu.



Obr. 4.13: Graf zhodnocující složitost.



Obr. 4.14: Graf hodnocení nápověd.



Obr. 4.15: Skóre hry testované respondenty na gymnáziu.

Závěr

Cílem bakalářské práce bylo provést analýzu aktuálních hrozeb kolujících po síti internet a vytvořit demonstraci úspěšného útoku pomocí metod sociálního inženýrství.

Teoretická část je převážně zaměřena na představení sociálního inženýrství s důrazem na aktuální hrozby spadající do této oblasti. Jsou v ní uvedeny jednotlivé metody mající za cíl oklamat oběti a tím získat žádaná data nebo donutit osobu k provedení určité akce. Tyto způsoby se zakládají na využití kognitivních chyb úsudku. V práci byly popsány čtyři typy útoků vycházející z útočného cyklu a skutečné hrozby uplynulých let, které stále mohou představovat určité riziko. Je v ní definován pojem malware, rozdělení škodlivého software a příklady hrozeb s ním spojené.

V praktické části je práce zaměřena na metody spoofing, pretexting, krádež identity, phishing, podvodné jednání a baiting. V rámci platformy BUTCA bylo vytvořeno testovací prostředí zaměřující se na výše uvedené metody. Podkapitoly popisující tvorbu scénáře jsou rozděleny podle jednotlivých úkolů, které mají uživatelé splnit. Ty byly inspirovány známými hrozbami kolujících po síti internet. Na metodu baiting se zaměřuji dva úkoly v rámci nich byly vytvořeny dva typy malware a to spyware, konkrétně keylogger běžící skrytě na pozadí po instalaci zdánlivě legitimního programu, a adware, který byl inspirován případem Christma Exec uvedeným v teoretické části. Soubor jevící se jako obrázek je ve skutečnosti archiv obsahující spustitelný soubor nesoucí škodlivý software. Každým dnem přibývá počet podvedených uživatelů internetových bazarů, proto jsem se v práce zaměřila na podvody tohoto typu. Byly vytvořeny typické situace pro tyto podvody. Stejně tak se zabývá hrozbou phishing nebo zobrazení falešného jména odesílatele. Díky simulaci těchto situací by měl uživatel pochopit, kde problém nastává, a až se s ním setká v reálné situaci, bude obezřetný a bude také vědět, jak postupovat.

Součástí scénáře je vědomostní kvíz a dotazník zpětné vazby. Testování laboratorní úlohy probíhalo ve dvou fázích. Jednak na SPŠ Třebíč jednak na gymnáziu (GJP a SOŠ Slavičín).

Názory respondentů na funkčnost, užitečnost a složitost lze využít pro budoucí vylepšení bakalářské práce.

Jsem přesvědčena, že se podařilo dosáhnout cíle této práce, rozšířit povědomí respondentů o technikách a taktikách sociálního inženýrství, které jsou využívány k získávání citlivých informací.

Literatura

[1] *RedLine: self-spreading stealer targets gamers on YouTube*. Online. In: Kaspersky. 2022. Dostupné z: https://usa.kaspersky.com/about/press-releases/2022_redline-self-spreading-stealer-targets-gamers-on-youtube. [cit. 2023-11-23].

[2] TUDOR, Dora. Malware as a Service (MaaS). *What It Is and How It Can Threaten Your Business?* Online. In: Heimdalsecurity. 2022, 2022-03-24. Dostupné z: https://heimdalsecurity.com/blog/what-is-malware-as-a-service-maas/#:~:text=Malware-as-a-Service%20%28MaaS%29%20is%20the%20term%20used%20to%20describe,botnet%20service%20that%20allows%20them%20to%20disseminate%20malware. [cit. 2023-11-22].

[3] SHEN, Ashley. *Phishing campaign targets YouTube creators with cookie theft malware*. Online. Threat Analysis Group (TAG). 2021. Dostupné z: https://blog.google/threat-analysis-group/phishing-campaign-targets-youtube-creators-cookie-theft-malware/. [cit. 2023-11-23].

[4] HRDINA, Richard. *Podvody na internetových bazarech*. Online. In: Policie České republiky. 2022. Dostupné z: https://www.policie.cz/clanek/podvody-na-internetovych-bazarech.aspx. [cit. 2023-11-28].

[5] PPL CZ. *Podvodné stránky PPL: Nenechte se napálit!*. Online. In: Ppl. 2023. Dostupné z: https://www.ppl.cz/cs/w/podvodne-webove-stranky-ppl. [cit. 2023-11-28].

[6] ESET. *Průzkum ESET: Třetina lidí se stala cílem podvodu na internetových bazarech*. Online. In: Eset. 2022. Dostupné z: https://www.eset.com/cz/o-nas/pro-novinare/tiskove-zpravy/pruzkum-eset-tretina-lidi-se-stala-cilem-podvodu-na-internetovych-bazarech/. [cit. 2023-11-28].

[7] JONES, Jack. *This is why you shouldn't use the app to become Barbie*. Online. In: Gearrice. 2023. Dostupné z: https://www.gearrice.com/update/this-is-why-you-shouldnt-use-the-app-to-become-barbie/. [cit. 2023-11-24].

[8] ZYLM, Wojciech. *Is there a Barbie scam? The Ministry of Digital Affairs issues a warning*. Online. In: BBN Breaking. 2023, 2023-9-25. Dostupné z: https://bnn.network/politics/is-there-a-barbie-scam-the-ministry-of-digital-affairs-issues-\protect\penalty-\@Ma-warning/. [cit. 2023-11-24].

[9] NÁRODNÍ ÚŘAD PRO KYBERNETICKOU A INFORMAČNÍ BEZPEČNOST [NÚKIB]. *Aplikace TikTok představuje bezpečnostní hrozbu*. Online, PDF. In: Národní úřad pro kybernetickou a informační bezpečnost. Brno, 2023, 8. března 2023. Dostupné z: https://www.nukib.cz/download/uredni_deska/2023-03-08_Varovani-TikTok_final.pdf. [cit. 2023-11-28].

[10] CHETIOUI, Kaouthar; BAH, Birom; ALAMI, Abderrahim Ouali a BAHNASSE, Ayoub. Overview of Social Engineering Attacks on Social Networks. Online. *Procedia computer science*. 2022, roč. 198, s. 656-661. ISSN 1877-0509. Dostupné z: Elsevier B.V, https://doi.org/10.1016/j.procs.2021.12.302. [cit. 2023-11-02].

[11] WANG, Zuoguang; SUN, Limin a ZHU, Hongsong. Defining Social Engineering in Cybersecurity. Online. *IEEE Access*. 2020, roč. 8, s. 85094-85115. ISSN 2169-3536. Dostupné z: https://doi.org/10.1109/ACCESS.2020.2992807. [cit. 2023-11-19].

[12] MITNICK, Kevin a SIMON, William. *Umění klamu*. Gliwice: Helion, 2003. ISBN 83-7361-210-6.

[13] GALLEGOS-SEGOVIA, Pablo L.; BRAVO-TORRES, Jack F.; LARIOS-ROSILLO, Víctor M.; VINTIMILLA-TAPIA, Paúl E.; YUQUILIMA-ALBARAD, Iván F. et al. Social engineering as an attack vector for ransomware. In: *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. Pucon: IEEE, 2017, s. 1-6. ISBN 978-1-5386-3123-2. Dostupné z: https://doi.org/10.1109/CHILECON.2017.8229528.

[14] KALNIŅŠ, Rūdolfs; PURIŅŠ, Jānis a ALKSNIS, Gundars. Security Evaluation of Wireless Network Access Points. Online. *Applied Computer Systems*. 2017, roč. 21, č. 1, s. 38-45. ISSN 2255-8691. Dostupné z: https://doi.org/10.1515/acss-2017-0005. [cit. 2023-11-19].

[15] POKROVSKAIA, Nadezhda N. a SNISARENKO, Svetlana O. Social engineering and digital technologies for the security of the social capital' development. In: *2017 International Conference "Quality Management,Transport and Information Security, Information Technologies"(IT&QM&IS)*. St. Petersburg, Russia: IEEE, 2017, s. 16-18. ISBN 978-1-5386-0703-9. Dostupné z: https://doi.org/10.1109/ITMQIS.2017.8085750.

[16] BTOUSH, Mohammad; ALARABEYAT, Abdulsalam; RYATI, Omar; HASSAN, Muneer a AHMAD, Sulieman. INCREASING INFORMATION SECURITY INSIDE ORGANIZATIONS THROUGH AWARENESS LEARNING FOR EMPLOYEES. Online. *Journal of Theoretical and Applied Information Technology*. 2005, roč. [24], č. [2], s. 1-7. Dostupné z: http://www.jatit.org/volumes/Vol24No2/2Vol24No2.pdf. [cit. 2023-11-19].

[17] BANU, Dr. M. Nazreen a BANU, S. Munawara. A comprehensive study of phishing attacks. Online. *International Journal of Computer Science and Information Technologies*. 2013, roč. 4, č. 6, s. 783-786. ISSN 0975-9646. Dostupné z: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=2bf12ff75150903efee426f23035c94d599597ae. [cit. 2023-11-19].

[18] KROMBHOLZ, Katharina; HOBEL, Heidelinde; HUBER, Markus a WEIPPL, Edgar. Advanced social engineering attacks. Online. *Journal of Information Security and Applications*. 2015, roč. 22, s. 113-122. ISSN 2214-2126. Dostupné z: https://doi.org/10.1016/j.jisa.2014.09.005. [cit. 2023-11-19].

[19] HOFBAUER, Stefan; BECKERS, Kristian a QUIRCHMAYR, Gerald. Defense Methods against VoIP and Video Hacking Attacks in Enterprise Networks. In: *10th International Conference on e-Business (iNCEB2015)*. Bangkok, Thailand: 2015, s. 101-110. ISBN 978-974-456-769-7. Dostupné také z: http://www.inceb2015.sit.kmutt.ac.th/paper/P02Stefan.pdf.

[20] HASAN, Mosin; PRAJAPATI, Nilesh a VOHARA, Safvan. CASE STUDY ON SOCIAL ENGINEERING TECHNIQUES FOR PERSUASION. Online. *International journal on applications of graph theory in wireless ad hoc networks and sensor networks*. 2010, roč. 2, č. 2, s. 1-7. Dostupné z: https://doi.org/10.5121/jgraphoc.2010.2202. [cit. 2023-11-19].

[21] ALAZRI, Aisha Suliaman. The awareness of social engineering in information revolution: Techniques and challenges. In: *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*. London, UK: IEEE, 2015, s. 198-201. ISBN 978-1-9083-2052-0. Dostupné z: https://doi.org/10.1109/ICITST.2015.7412088.

[22] OOSTERLOO, Bernard. *Managing Social Engineering Risk: Making social engineering transparent*. Online, Master thesis. Utrecht: University of Twente, 2008. Dostupné z: http://essay.utwente.nl/59233/1/scriptie_B_Oosterloo.pdf. [cit. 2023-11-28].

[23] GHAFIR, Ibrahim; PRENOSIL, Vaclav; ALHEJAILAN, Ahmad a HAMMOUDEH, Mohammad. Social Engineering Attack Strategies and Defence Approaches. In: *2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*. Vienna, Austria: IEEE, 2016, s. 145-149. ISBN 978-1-5090-4052-0. Dostupné z: https://doi.org/10.1109/FiCloud.2016.28.

[24] VENKATESHA, Sushruth; REDDY, K. Rahul a CHANDAVARKAR, B. R. Social engineering attacks during the COVID-19 pandemic. Online. *SN computer science*. 2021, roč. 2, č. 78, s. 1-9. Dostupné z: https://doi.org/10.1007/s42979-020-00443-1. [cit. 2023-11-19].

[25] COSTANTINO, Gianpiero; LA MARRA, Antonio; MARTINELLI, Fabio a MATTEUCCI, Ilaria. CANDY: A Social Engineering Attack to Leak Information from Infotainment System. In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. Porto, Portugal: IEEE, 2018, s. 1-5. ISBN 978-1-5386-6355-4. ISSN 2577-2465. Dostupné z: https://doi.org/10.1109/VTCSpring.2018.8417879.

[26] KOLOUCH, Jan. *CyberCrime*. Praha: CZ.NIC, z.s.p.o, 2016. ISBN 978-80-88168-15-7.

[27] ELISAN, Christopher C. *Malware, Rootkits & Botnets A Beginner's Guide*. McGraw Hill, 2012. ISBN 978-0-07-179205-9.

[28] GANDOTRA, Ekta; BANSAL, Divya a SOFAT, Sanjeev. Malware Analysis and Classification: A Survey. Online. *Journal of Information Security*. 2014, roč. 5, č. 2, s. 56-64. ISSN 2153-1234, 2153-1242. Dostupné z: https://doi.org/10.4236/jis.2014.52006. [cit. 2023-12-03].

[29] SAEED, Imtithal; SELAMAT, Ali a ABDELRAHMAN, Ali. A Survey on Malwares and Malware Detection Systems. Online. *International Journal of Computer Applications (IJCA)*. 2013, roč. 67, č. 16, s. 25-31. Dostupné z: https://www.researchgate.net/profile/Imtithal-Saeed/publication/272238656_A_Survey_on_Malwares_and_Malware_Detection_Systems/links/566284c608ae192bbf8cf1a5/A-Survey-on-Malwares-and-Malware-Detection-Systems.pdf. [cit. 2023-12-03].

[30] ARJUN, Singh; PUSHPA, Choudhary; TYAGI a DHEERENDRA KUMAR, Tyagi. Keylogger Detection and Prevention. Online. *Journal of Physics: Conference Series*. 2021, roč. 2007, č. 1, s. 1-3. Dostupné z: https://doi.org/10.1088/1742-6596/2007/1/012005. [cit. 2023-12-03].

[31] ESET. *Infostealer: Co je infostealer?* Online. In: Eset. B. r. Dostupné z: https://www.eset.com/cz/infostealer/. [cit. 2023-12-03].

[32] KASPERSKY. *What Is Browser Hijacking?* Online. In: Kaspersky. B. r. Dostupné z: https://www.kaspersky.com/resource-center/threats/browser-hijacking. [cit. 2023-12-03].

[33] KASPERSKY. *Unmasking Zanubis: banking Trojan's sneaky evolution and cryptocurrency threats unveiled.* Online. In: Kaspersky. [2023]. Dostupné z: https://www.kaspersky.com/about/press-releases/2023_unmasking-zanubis-banking-trojans-sneaky-evolution-and-cryptocurrency-threats-unveiled. [cit. 2023-12-03].

[34] ŠURKALA, Milan. *Adware "Agent Smith"měnil aplikace, napadl 25 milionů telefonů.* Online. In: Svět mobilně. B. r. Dostupné z: https://www.svetmobilne.cz/adware-agent-smith-menil-aplikace-napadl-25-milionu-telefonu/7772#:~:text=Sv%C4%9Btem%20aplikac%C3%AD%20pro%20Android%20se%20za%C4%8Dal%20%C5%A1%C3%AD%C5%99it%20pokro%C4%8Dil%C3%BD,po%20cel%C3%A9m%20sv%C4%9Bt%C4%9B%20bylo%20napadeno%20adwarem%20Agent%20Smith. [cit. 2023-12-04].

[35] XIAOJUN, Tong; ZHANGQUAN, Zhao; HUIMIN, Shuai a ZHU, Wang. The research of worm distributed detection technology based on network security. In: *2010 IEEE International Conference on Wireless Communications, Networking and Information Security.* Beijing, China: IEEE, 2010, s. 416-417. ISBN 978-1-4244-5849-3. Dostupné z: https://doi.org/10.1109/WCINS.2010.5541811.

[36] BISHOP, M. Education in information security. Online. *IEEE Concurrency.* 2000, roč. 8, č. 4, s. 7-8. ISSN 1558-0849. Dostupné z: https://doi.org/10.1109/4434.895087. [cit. 2023-12-05].

Seznam symbolů a zkratek

AI Artificial Intelligence

API Application Programming Interface

CD Compact Disc

CMD Command Prompt

CSS Cascading Style Sheets

CTF Capture The Flag

DWORD Double Word

HTML Hypertext Markup Language

HTTP Hypertext Transfer Protocol

HTTPS Hypertext Transfer Protocol Secure

IP Internet Protocol

Maas Malware as a Service, Malware jako Služba

MIME Multipurpose Internet Mail Extensions

NSSM Non-Sucking Service Manager

PDF Portable Document Format

PHP Hypertext Preprocessor

PIL Python Imaging Library

SFX Self-extracting archive

SI Sociální Inženýrství

SMS Short Message Service

SMTP Simple Mail Transfer Protocol

VPN Virtual Private Network, Virtuální Privátní Síť

WSH Windows Script Host

A Scénář hry demonstrující sociální inženýrství

A.1 Prolog

Eva je nadějnou studentkou na VUT. Řeší typické studentské trable a obtíže. Naštěstí má kolem sebe mnoho dobrých lidí, díky kterým vše zvládá s nadhledem. Poslední dobou se cítí celkem unaveně, má toho hodně – práce, škola, doučování – a přehlíží tak návnady kyberzločinců (sociálních inženýrů), které ji občas do schránky zavítají. Pomoz Evě zvládnout tyto situace. Právě teď máš možnost si na vlastní kůži vyzkoušet jejich taktiky, abys naostro nenaletěl!

Celou úlohu provádějte v sandbox prostředí, které je dostupné níže.

A.2 Úkol 1

Evě se po delší době ozval její bratranec se kterým se dlouhou dobu neviděli, má na ni podivnou prosbu, žádá ji, aby mu napsala své telefonní číslo, které potřebuje urgentně! Je to vážně on?

Zprávu si můžeš zobrazit zadáním adresy: **evinmail.biz**

Flag je adresa odesílatele ve tvaru Flag(email@email.cz)

A.3 Úkol 2

Eva má na vysoké škole jednu velmi dobrou kamarádku Magdu. Píšou si celkem často a pomáhají si navzájem se studiem. Proto nijak nezpochybňuje to, co jí Magda zasílá a neváhá otevřít jakýkoliv odkaz. Myslíš, že je to správné?

Jejich konverzaci si můžeš prohlédnout na ploše v aplikaci -ChatMe-.

Na falešnou stránku je možné se přihlásit. Flag je ve formátu Flag(nickname) po přihlášení na falešnou stránku.

A.4 Úkol 3

Evu tíží další nemilá situace, finanční rezerva, kterou si vytvořila jí postačila pouze na jeden semestr. Nyní šetří, kde se dá. Využívá také možnosti čerpat stipendium, její známky a aktuální životní situace jí dovoluje požádat si o prospěchové i sociální stipendium.

Jako každý den otevřela svou emailovou poštu a velmi se zaradovala, proč pak? Má vůbec důvod k radosti?

Flag je dělen na dvě části, XXX značí chybějící znaky, které musíš najít a doplnit. Email najdeš na ploše jako „mailhog“. Od této chvíle budeš využívat pouze tento email v rámci celé hry.

A.5 Úkol 4

Blíží se bratrovy 10. narozeniny. Taková jubilejní oslava si žádá větší dar. Eva se rozhodla mu splnit jeho dlouholetý sen a chce mu koupit jeho vlastní herní konzoli. Její rozpočet však na to nestačí, a tak se rozhlíží po inzerátech. Jeden slibný našla, avšak konverzace s majitelkou v ní vzbudilo podezření.

Její konverzaci s Vlastou si můžeš prohlédnout na ploše v aplikaci -ChatMe-.

Eva má na ploše zajímavý dokument ohledně nákupů na bazarech, bazary.pdf, přečti si ho. Zaměř se hlavně na odkazy!

Flag je pravé jméno majitele účtu ve formátu Flag(Příjmení Jméno).

A.6 Úkol 5

“Dobře, i kdyby šlo o účet manžela, stejně se mi na tom něco nezdá.”

Eva se snaží zjistit, jak to celé je. Majitelka na profilu skoro nic nemá, kromě fotky. (Stále jde o konverzaci s Vlastou.)

Eva má na ploše zajímavý dokument ohledně nákupů na bazarech, bazary.pdf, přečti si ho. Zaměř se hlavně na odkazy!

Flag je první výskyt fotky Flag(měsíc den, rok)

A.7 Úkol 6

„Dobře, nakupovat na bazarech by se mi moc teda nevyplatilo. Ale když budu prodávat já, nic se mi přece nemůže stát“, říká si Eva.

Zde je karta Evy A.1, kterou možná budeš potřebovat:

Obr. A.1: Karta Evy.

Její konverzaci se zájemkyní Annou si můžeš prohlédnout na ploše v aplikaci -ChatMe-.

Flag je ve formátu Flag(flag).

A.8 Úkol 7

“Ahoj Evo. Jelikož Ti grafika a taky kreslení vždy šlo mohla bys mi, prosím, vytvořit pozvánku na mou oslavu? Brácha” Jistě, že Eva ráda vytvoří pozvánku na tento event. Napadlo ji vytvořit koláž ze všech fotek bráchy, které si u sebe nosí. Bohužel však nemá žádný software, pomocí kterého by to byla schopná udělat. Kamarád ji doporučil super stránky, jenže si není jistá, zda si adresu správně pamatuje. Bylo to nějak stahuj.wz.cz, myslím ...

Co se po instalaci a spuštění photoshopu stalo? Že by se něco dělo na pozadí?

Flag je ve formátu Flag(IP adresa serveru, kam se odesílá něco, co by nemělo :))

Pro zajímavost si můžeš nalezenou IP adresu zadat do prohlížeče. Co zajímavého se na stránce nachází?

A.9 Úkol 8

“Dobře, poslední, co ještě zkusím. Najdu si pěkný obrázek na netu a přidám zde jen nějaký text, prostě základ jako datum, čas a místo konání.”

Eva si to brouzdá po netu a zaujme ji tato galerie `http://skvela.galerie.cz/`.

Flag se zjeví na pozadí ve formátu Flag(flag).

A.10 Epilog

Tak to byl příběh Evy, která si vyzkoušela, a Vy spolu s ní, všemožné nástrahy číhající na internetu. Teď už ví, že ne vše, co je zadarmo je doopravdy zadarmo. Někdy jde o výměnný obchod vašeho soukromí na úkor penězům. Také zjistila, že ochrana identity je na internetu velmi cenná, a i když se někdo tváří jako bratranec, nemusí to být právě on! Naučila se jaké taktiky používají sociální inženýři k získání citlivých informací, nyní zná také pojmy jako spoofing, phishing, baiting a další. Setkala se s malware typu keylogger a adware. Teď jí už jen zbývá všechen tento škodlivý software zcela odstranit ze svého počítače. K tomu si už raději přizve kamaráda, co se v tomto odvětví více vyzná. :)

A.11 Vědomostní kvíz

1. S jakými metodami sociálního inženýrství jste se dnes potkali:
   - Tailgating
   - Phishing
   - Pretexting
   - Baiting
   - Vishing
2. Jaké druhy malware poslední dvě úlohy obsahovaly?
   - Spyware
   - Ransomware
   - Worm
   - Adware
3. Co jste použili pro zjištění původu fotografie na profilu Vlasty?
   - Google Lens
   - pimeyes.com
   - facecheck.id
   - tineye.com
4. Pamatuješ si jaký nadpis, vytvořený pomocí ASCII artu, obsahovala stránka na adrese 10.10.10.11, kde se v úloze 6. zasílala data?
   - Find your Keylogger
   - Just Keylog website
   - Keylogger
   - Ahoj světe!
5. Byla stránka od České pošty jejich klonem?
   - ano
   - ne

B Průchod scénářem – řešení

B.1 Úkol 1

Po zadání stránky `evinmail.biz` do internetového prohlížeče, zobrazí se tato HTML stránka B.1, která běží na webovém serveru Apache a je tedy dostupná pouze z virtualizovaného prostředí. To platí i pro další uvedené stránky.

Hlavním cílem je nalezení pravého jména odesílatele, tedy jeho e-mailové adresy, která je aktuálně skrytá. Toho docílíme najetím myší na část vedle ikony, kde se adresa zobrazí.

Obr. B.1: Ukázka stránky evinmail.biz.

B.2 Úkol 2

Zmíněná aplikace „ChatMe“ je dostupná na adrese `chatMe-basicChattingApp.cz`. Pro zjednodušení byl na ploše vytvořen zástupce s touto adresou, hráči tedy stačí tohoto zástupce otevřít. Tím se dostává do komunikace mezi Evou a Magdou. Přečte si komunikaci a po kliknutí na odkaz je přesměrován na „instagram“, tedy jeho falešnou napodobeninu „instgram“. Flag této úlohy je následně profil, pod kterým se na tuto platformu dostali. Flag lze konkrétně najít na /profile.html (nebo na homepage, kde je ukázáno, pod jakým účtem je uživatel přihlášen).

B.3 Úkol 3

Aplikace Mailhog a Python skript se spustí hned po nastartování virtuálního stroje (obstarává VBScript). V případě selhání (aplikace Mailhog se nenachází mezi spuštěnými procesy) je skript *autorun.ps1* možné spustit ručně ve složce \Documents\tools \startSetUp. Pokud by aplikace Mailhog jela, ale schránka by byla prázdná, je možné spustit pouze *se_ctf.exe* obstarávající zaslání pošty.

Na ploše je zástupce této aplikace, odkazující na adresu localhost:8025, uživateli stačí jen poklepat na zástupce, aby se dostal do schránky. Hráče v tomto úkolu zajímá phishing zpráva s návnadou "zvýšení stipendií". E-mail obsahuje odkaz, který odkazuje na falešné stránky VUT.

První část flagu se nachází na stránce odpovědné osoby, konkrétně v obrázku. Ten je možno zobrazit pravým tlačítkem myši – otevřít obrázek na nové kartě.

Dále by měl uživatel stáhnout dostupný infikovaný soubor. Po jeho spuštění v aplikaci LibreOffice se spolu s ním spustí i makro, které do schránky Mailhog zašle e-mail spolu s druhou částí flagu.

B.4 Úkol 4

V ChatMe si můžete zobrazit konverzaci s Vlastou Němou, na postranní liště nalevo. Jako první se Eva zeptala, zda se opravdu jmenuje Vlasta a proč je majitelem účtu někdo jiný. To by mělo hráče navést k zamyšlení. Dalším vodítkem je PDF na ploše, nazvané *bazary.pdf*. To obsahuje prezentaci o tom, jak se při nákupech na bazarech pohybovat. Hlavně však obsahuje odkazy na užitečné nástroje jako `http://podvodnabazaru.cz`, kde uživatel má vyhledat číslo účtu *2020201111/1204*. To je možno zadat přímo do vyhledávače, nebo jej naleznout přímo ve výpisu všech čísel účtů a další informaci, které tato databáze obsahuje. Na stránce potom stačí nalézt opravdové jméno majitele účtu, které je Mýval Milan.

B.5 Úkol 5

V ChatMe si můžete zobrazit konverzaci s Vlastou Němou. Jako další se Eva ptá na aktuálnost fotky, aby hráče navedla, na co se má v tomto úkolu zaměřit. Dalším vodítkem je PDF na ploše nazvané *bazary.pdf*. To obsahuje prezentaci o tom, jak se při nákupech na bazarech pohybovat. Hlavně však obsahuje odkazy na užitečné nástroje jako *tineye*, kde flag je datum prvního výskytu fotky, které je možné získat zvolením filtru *sort by Oldest*. Fotku lze z aplikace ChatMe stáhnout (pravé tlačítko – uložit obrázek jako).

B.6 Úkol 6

V aplikaci ChatMe je dostupná konverzace s Annou Skromnou, která má zájem o nabízené zboží. Po domluvě pošle odkaz, kde si má Eva peníze za zboží převzít. Jde o kopii stránky České pošty. Jelikož tato stránka reálně existuje, přišlo mi vhodné ji zde použít. Následně je zde potřeba jen doplnit nějaký telefon a e-mail, pak je uživatel přesměrován na stránku *gate.html*. Tady se nachází formulář, kde má uživatel zadat číslo karty a dokonce pin. Jsou zde tři stádia:

1. Uživatel zadal všechny údaje spojené s kartu Evy ze scénáře. Je zaslán e-mail do aplikace Mailhog o stržení peněžního obnosu z karty.
2. Uživatel zadal kartu Evy ze scénáře, to znamená, že buď udělal někde chybu nebo si zbytek informací vymyslel. Je zaslán e-mail do aplikace Mailhog s upozorněním, že by neměl zadávat svou kartu na nedůvěryhodné stránky.
3. Uživatel zadal validní kartu, vytvořenou pomocí volně dostupného generátoru. Toto je správné řešení, proto je zaslán e-mail do aplikace Mailhog spolu s klíčem.

Po uplynutí data na kartě je nutno předělat, tedy aktualizovat obrázek karty a také na stránce *gate.html* hodnotu proměnné dateOnCard == '7/2030' || dateOnCard == '07/30'. Po uplynutí roku 2030 bude karta považována za neplatnou.

B.7 Úkol 7

Na stránkách uvedených ve scénáři, tedy `stahuj.wz.cz`, je možné stažení rádoby Photoshopu. Po jeho instalaci a spuštění začne na pozadí běžet keylogger, který zaznamenává stisknuté klávesy a odesílá tyto data na server, který je pomocí PHP skriptu zobrazuje na homepage. Po zadání IP adresy do prohlížeče se stránka zobrazí. Avšak flag je ta samotná IP adresa. Tuto adresu je možné získat pomocí aplikace Wireshark, kde je možné síťový provoz filtrovat. Pokud využijeme filtr *frame contains "test"*, kde *test* nahradíme konkrétní sekvenci stlačených znaků, je možné jednoduše získat IP adresu serveru.

B.8 Úkol 8

Na stránce `http://skvela.galerie.cz/` se nachází galerie obrázku a fotek. Ten pravý obrázek je ve skutečnosti aplikace obsahující adware. Po stažení a zobrazení obrázku se aplikace spustí a vyskočí několik adware oken, které uživatel postupně ukončí. Následně jsou všechny zbylá okna minimalizovány. Nastane vygenerování obrázku, který obsahuje hledaný flag. Ten je uložen na ploše. V poslední části je tento obrázek nastaven jako pozadí plochy. Flag je tedy možné získat otevřením obrázku nebo přímo shlédnutím pozadí plochy.

Obr. B.2: Obrázek obsahující spustitelný soubor.

B.9 Vědomostní kvíz

1. S jakými metodami sociálního inženýrství jste se dnes potkali:
   - Tailgating
   - **Phishing**
   - **Pretexting**
   - **Baiting**
   - Vishing
2. Jaké druhy malware poslední dvě úlohy obsahovaly?
   - **Spyware**
   - Ransomware
   - Worm
   - **Adware**
3. Co jste použili pro zjištění původu fotografie na profilu Vlasty?
   - Google Lens
   - pimeyes.com
   - facecheck.id
   - **tineye.com**
4. Pamatuješ si jaký nadpis, vytvořený pomocí ASCII artu, obsahovala stránka na adrese 10.10.10.11, kde se v úloze 6. zasílala data?
   - **Find your Keylogger**
   - Just Keylog website
   - Keylogger
   - Ahoj světe!
5. Byla stránka od České pošty jejich klonem?
   - **ano**
   - ne