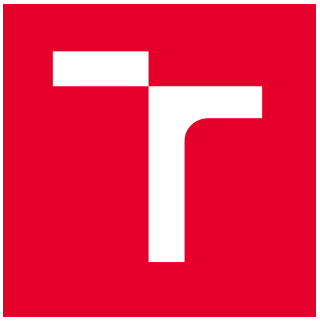


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

APLIKACE NA STAVBU A ZOBRAZOVÁNÍ VEKTORŮ ÚTOKŮ POMOCÍ METOD SOCIÁLNÍHO INŽENÝRSTVÍ

APPLICATION FOR BUILDING AND DISPLAYING ATTACK VECTORS USING SOCIAL ENGINEERING

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Jan Kašpar

VEDOUCÍ PRÁCE

SUPERVISOR

JUDr. Mgr. Jakub Harašta, Ph.D.

BRNO 2023

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní program **Informační bezpečnost**

Ústav telekomunikací

Student: Bc. Jan Kašpar ***ID:*** 211796

Ročník: 2 ***Akademický rok:*** 2022/23

NÁZEV TÉMATU:

Aplikace na stavbu a zobrazování vektorů útoků pomocí metod sociálního inženýrství

POKYNY PRO VYPRACOVÁNÍ:

V rámci práce student vytvoří aplikaci na stavbu a zobrazování vektorů útoků pomocí metod sociálního inženýrství. Cílem je vytvoření aplikace, která umožní namodelování vektorů útoků, jejich přiřazení členům týmu, a vložení dalších vhodných detailů. Aplikace musí umožnit práci se související smluvní dokumentací a generování dílčích a celkových zpráv o průběhu útoků.

DOPORUČENÁ LITERATURA:

podle pokynů vedoucího práce

Termín zadání: 6.2.2023 ***Termín odevzdání:*** 19.5.2023

Vedoucí práce: JUDr. Mgr. Jakub Harašta, Ph.D.
Konzultant: Daniel Hejda

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

ABSTRAKT

Tato diplomová práce se zaměřuje na nejvíce používané techniky sociálního inženýrství v dnešní době, legislativu spojenou se sociálním inženýrstvím a vývojem aplikace pro Red Teaming. V rámci práce jsou popsány používané techniky sociálního inženýrství, paragrafy se kterými je možné se v při používání těchto technik nejčastěji setkat a popsána funkčnost aplikace, která se zaměřuje na techniku Vishing a pomáhá k neporušení zákonů, které byly identifikovány jako nejvíce rizikové.

KLÍČOVÁ SLOVA

Sociální inženýrství, Legislativa, Red Teaming, Python, Vishing

ABSTRACT

This thesis focuses on the most used social engineering techniques today, legislation related to social engineering and the development of a Red Teaming application. The thesis describes the social engineering techniques that are used, the paragraphs that are most commonly encountered in the use of these techniques and describes the functionality of the application that focuses on the Vishing technique and helps not to break the laws that have been identified as the most risky.

KEYWORDS

Social Engineering, Legislative, Red Teaming, Python, Vishing

KAŠPAR, Jan. *Aplikace na stavbu a zobrazování vektorů útoků pomocí metod sociálního inženýrství*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2030, 81 s. Diplomová práce. Vedoucí práce: JUDr. Mgr. Jakub Harašta, Ph.D

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Bc. Jan Kašpar
VUT ID autora:	211796
Typ práce:	Diplomová práce
Akademický rok:	2029/30
Téma závěrečné práce:	Aplikace na stavbu a zobrazování vektorů útoků pomocí metod sociálního inženýrství

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

Brno ....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu JUDr. Mgr. Jakubu Haraštovi, Ph.D. za odborné vedení, konzultace, trpělivost a cenné rady k práci. Dále bych rád poděkoval panu Danielu Hejdovi ze společnosti Cyber Rangers, za odborné rady a konzultace při vytváření praktické části práce.

Obsah

Seznam obrázků

Úvod

V této době je kladen velký důraz na zabezpečení počítačové infrastruktury. Tato diplomová práce se zaměřuje na nejslabší článek v tomto zabezpečení a to na lidský faktor za pomocí využití metod sociálního inženýrství. Využitím těchto metod může dojít k získání citlivých informací prostřednictvím internetu nebo fyzické komunikace s obětí.

V teoretické části práce je vysvětleno co je sociální inženýrství. Kapitola 1.2 je věnována autentizaci se zaměřením na autentizaci znalostí, pro vysvětlení toho, čeho se útočníci snaží zmocnit. V kapitole 1.3 jsou popsány formy sociálního inženýrství, kde jsou popsány hrozby spojené s používáním sociálních sítí a komplikace, které mohou nastat při kompromitaci účtu na sociální síti nebo e-mailu. Kapitola 2 se věnuje popisu technik využívající sociální inženýrství s uvedenými příklady jak může být technika použita a jsou zde popsány typy škodlivých kódů, mezi kterými je dobré rozlišovat. V kapitole 3 je popsán rozdíl mezi penetračním testováním a Red Teamingem, uvedeny typy týmů a jejich funkce. Zároveň je zde popsáno zabezpečení objektu s překážkami, které je nutné překonat pro vniknutí do objektu a uvedena metodika Cyber Kill Chain pro vysvětlení aktivit, kterých chtějí testeři dosáhnout. Poslední kapitola teoretické části se věnuje legislativě, ve které jsou popsány zákony, a uvedeny příklady porušení těchto zákonů, se kterými se lze setkat při sociálním inženýrství. Zároveň jsou identifikovány nejvíce rizikové paragrafy při používání techniky Vishing a doporučen postup, jak se mají zaměstnanci chovat při aplikování této techniky.

Praktická část této práce se věnuje vývoji aplikace pro Red Team testery. V kapitole 5.1 jsou popsány frameworky používané při vývoji aplikace. Kapitola 5.2 popisuje vytvořené funkce aplikace a jejich použití. Zároveň je v podkapitole 5.2.3 uvedeno jakým způsobem bude aplikace pomáhat dodržování právních předpisů identifikovaných v kapitole 4, které byly určeny jako rizikové pro Vishing a představeny bezpečností prvky aplikace.

Cílem této práce je vytvoření aplikace pro Red Teaming, kde dojde k propojení technické a právnické složky. Tato aplikace pomůže v zajištění, že testování bude probíhat v souladu s právními předpisy a po technické stránce bude možné dohledat jak bylo testování provedeno společně s příslušnou dokumentací. Tato aplikace bude sloužit k testování zaměstnanců ve firmách za účelem zjištění nedostatků v zabezpečení.

1 Sociální inženýrství a jeho formy

V této části bude popsáno, co je a čím se vyznačuje sociální inženýrství, princip autentizace se zaměřením na autentizaci pomocí znalosti a formy sociálního inženýrství v dnešní době.

1.1 Sociální inženýrství

Sociální inženýrství neboli sociotechnika je metoda manipulace s lidmi, díky které lze získat důvěrné informace nebo informace, které mohou být i jinak prospěšné pro sociotechnika (útočníka). Pro úspěšný útok pomocí sociálního inženýrství je velmi důležitý předpoklad, že oběť je držena v nevědomosti. Tedy když je použita nějaká metoda sociálního inženýrství, tak oběť neví, že se stala cílem útoku a informace ve většině případů dobrovolně vydá.

V současné době jsou nejvíce rozšířeny bezkontaktní metody sociálního inženýrství. To znamená, že útočník není v přímém kontaktu s obětí, ale využívá ke komunikace jiné prostředky. Nejvíce rozšířeným prostředkem je internet, kde převládá metoda phishing. Dále lze využít kontaktních metod sociálního inženýrství, kde je oběť s útočníkem v přímém kontaktu. U těchto metod je využíván mobilní telefon, a to ve formě SMS zpráv nebo hovoru. Nebo v dnešní době velmi rozšířené a používané sociální sítě, kde se útočník může vydávat za jinou osobu nebo se pokusit přisvojit profil osoby, kterou oběť zná a důvěřuje jí. Na stejném principu je možné také využít e-mail. Při využití kontaktních metod jsou velmi důležité schopnosti sociotechnika, především jeho přesvědčovací schopnosti, díky kterým si u oběti vybuduje důvěru.

Sociotechnici využívají těchto metod, protože je mnohem jednodušší přelstít člověka k vydání informací, jako je heslo nebo jiné údaje, které mohou být pro útok prospěšné, než aby se pokoušeli překonat zabezpečení systému. Zabezpečení systému může totiž být velmi sofistikované a v některých případech i neprolomitelné. Při úspěšném použití některé z metod sociálního inženýrství oběť informace sama vydá a není tak nutné útočit přímo na software zabezpečení a pokoušet se ho prolomit pomocí hrubé síly nebo slovníkových útoků. Informace, které oběť poskytne, nemusejí vždy souviset s hesly nebo osobními údaji, ale mohou to být informace o čase výměny služeb na vrátnici, principu zabezpečení objektu apod.

Sociální inženýrství je nejvíce využíváno k získání osobních údajů jako jsou hesla do přihlašovacího sytému firmy, čísla bankovních účtů apod. Může ale také být využito k získání informací, které nemusí s osobními údaji vůbec souviset a jejich získání může utvořit celistvý obraz, díky kterému může být později veden úspěšný útok přímo na infrastrukturu. Zabezpečení počítačových systémů (pokud jsou správně

nastaveny) je v dnešní době na velmi dobré úrovni. Při použití vhodných asymetrických a symetrických šifer je matematicky dokázáno, že není možné toto zabezpečení prolomit v polynomiálním čase. Z toho vyplývá, že nejslabším článkem v zabezpečení systému je a vždycky bude člověk neboli lidský faktor. Ať je zabezpečení jakkoliv silné, pokud zaměstnanec má své heslo napsané na papírku pod klávesnicí, tak se stává zbytečným.[1, 2, 3, 4]

1.2 Autentizace

Samotná autentizace je klíčovým prvkem při ochraně informačních systémů před neoprávněným přístupem. Způsoby autentizace mohou být klasifikovány podle použitých prvků, jako jsou znalosti, předměty, biometrika, průkazy nebo činnosti. Mezi nejčastěji používané metody autentizace patří autentizace znalostí, která využívá sdíleného tajemství mezi uživatelem a systémem. Tento typ autentizace je však také nejčastěji cílem útoků pomocí sociálního inženýrství.

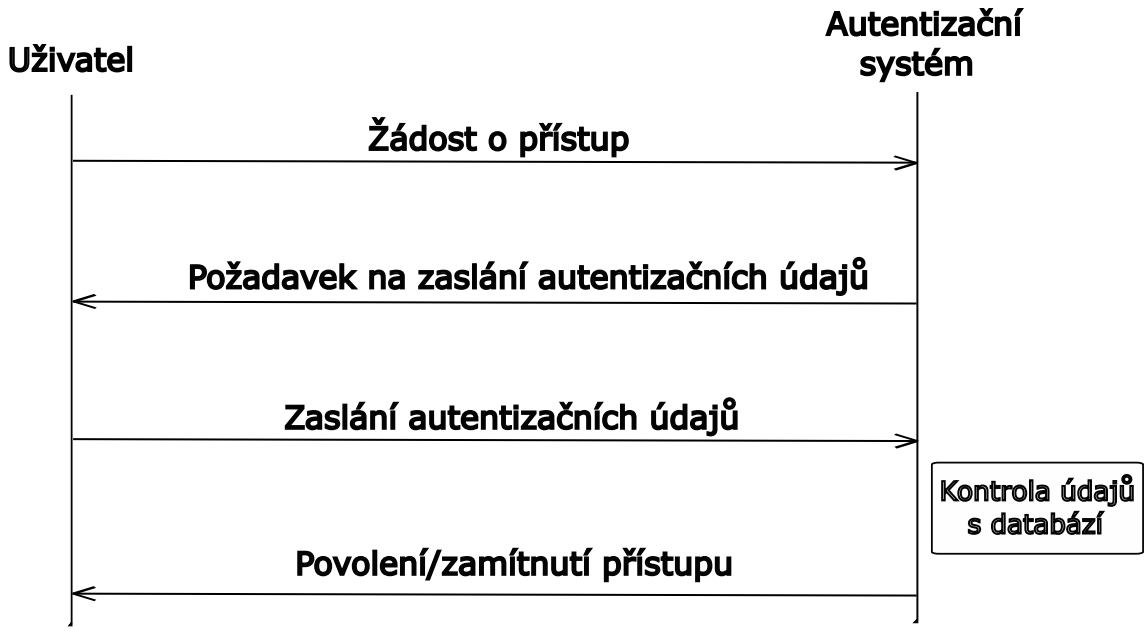
Útočníci mohou použít phishingové e-maily nebo jiné metody, aby získali autentizační údaje od uživatelů a následně získali neoprávněný přístup do systému. Vzhledem k tomu, že autentizace znalostí je nejrozšířenější autentizační metodou, je důležité brát v úvahu různé možnosti útoků a zvážit, zda by neměly být použity i jiné metody autentizace, jako je například autentizace biometrikou, která je více bezpečnější.

Kromě toho může být autentizace významným prvkem v sociálním inženýrství, zejména pokud uživatelé používají stejné nebo podobné autentizační údaje pro více systémů (přihlašovací jméno a heslo). V tomto případě může útočník získat údaje od jednoho systému a použít je pro útoky na další systémy, což zvyšuje celkové riziko bezpečnosti. Je tedy důležité, aby uživatelé používali silná hesla a pravidelně je měnili, aby minimalizovali riziko úspěšného útoku.

V kontextu sociálního inženýrství je třeba klást důraz na vzdělávání a povědomí o nebezpečí, které vyplývá z phishingu a dalších metod útoků používaných útočníky. Dobrá autentizace může být klíčem k ochraně citlivých informací a dat, ale musí být použita ve správném kontextu a s vědomím rizik a možností útoků.[5, 6]

1.3 Formy sociálního inženýrství v současnosti

Sociální inženýrství se dělí na 2 podoby. S první formou se lze setkat na internetu, tedy bez fyzického kontaktu, kdy útočník sbírá informace o zájmové osobě online. Útočník může zahájit komunikaci s obětí pomocí dnes velmi používaných sociálních sítí nebo prostřednictvím e-mailu.



Obr. 1.1: Autentizace znalostí

Výhodou sociálních sítí pro útočníka je, že se lidé cítí v online prostředí bezpečně, a tak je jednodušší získat potřebné informace. Spoustu informací lze zjistit přímo z profilu oběti na sociální síti. S druhou formou sociálního inženýrství, tedy kontaktní formou, je těžší se setkat, protože pro její provedení a dosažení cíleného je zapotřebí zkušený sociotechnik s dobrou znalostí lidského chování a neverbální komunikace.

1.3.1 Sociální sítě

Sociální sítě jsou v dnešní době nejrozšířenější a nejpopulárnější službou pro zprostředkování komunikace mezi lidmi. Jedná se o nejsnazší a nejrychlejší způsob, jak komunikovat s přáteli na dálku, sdílet fotky a videa, polohu a vyměňovat informace. Všechny tyto informace mají pro útočníka velkou cenu a dají se zneužít v jeho prospěch. Ze sdílených fotografií lze navíc pomocí metadat vyčíst další užitečné informace jako místo a čas pořízení. Většina sociálních sítí umožňuje uživatelům nastavit, kdo bude mít přístup k informacím na jejich profilu a tím zamezit zneužití informací pro nežádoucí osoby. Je důležité zmínit, že tyto sociální sítě také zálohují komunikaci, takže pokud útočník získá přístup k účtu, může získat zprávy od počátku založení účtu. Z důvodu rozšířenosti sociálních sítí jsou také velmi často cílem útoku [1, 7].

Pokud útočník získá přístup k profilu, nic mu nebrání zahájit komunikaci se spřátelenými kontakty a do zprávy vložit podvodnou stránku, odkaz na malware nebo požádat o peníze na specifický účet. Proto je dobré se vždy zamyslet, jaký odkaz bude rozkliknut, i když přijde od osoby, kterou známe a které věříme. Informace,

které jsou získány přístupem na profil oběti, lze dále využít například k vydírání (lechtivá fotografie) a to majitele profilu nebo osoby, která ji poslala.

1.3.2 E-mail

E-mail je často cílem útoků sociálních inženýrů. Pokud útočník získá autentizační údaje k e-mailu, může tak jednoduše získat i přístup k sociálním sítím (vyzkoušení stejného hesla, nahlášení zapomenutého hesla apod.) Pokud se mu tedy podaří získat přístup, může poté zahájit komunikaci se všemi kontakty oběti a rozesílat podvodné e-maily s odkazy a škodlivými přílohami.

Takový e-mail může obsahovat odkaz na stránku, kde se stáhne škodlivý software, který umožní útočníkovi přístup do počítače. Nebo podvodnou stránku, kde oběť zadá citlivé údaje, které tak útočník získá. Rozesílání takových e-mailů není vždy pravidlem, útočník může navázat komunikaci s jinou osobou, od které chce zjistit informace za použití identity oběti. Takové zprávy spoléhají na lidskou důvěřivost a zvědavost. Při obdržení pošty od někoho známého je větší šance, že oběť klikne na odkaz, než když by stejný e-mail přišel od někoho cizího[1, 7].

V první kapitole bylo vysvětleno co je a čím se vyznačuje sociální inženýrství a z jakých důvodů je výhodnější jeho použití před klasickými metodami na prolomení zabezpečení. Byl popsán princip autentizace se zaměřením na metodu autentizace pomocí znalostí. A byly uvedeny možné typy útoků na tento princip autentizace a formy sociálního inženýrství používané v této době.

2 Techniky sociálního inženýrství

Tato kapitola se věnuje technikám sociálního inženýrství neboli praktikám, které jsou využívány pro zjištění citlivých informací. Aplikace, která je vytvořena v rámci praktické části této práce,se zaměřuje na techniku Vishing. Ostatní techniky jsou uvedeny z důvodu podobnosti výše zmíněným praktikám a jejich možnosti využití pro zisk informací nebo snazšího získání důvěry u oběti.

Hlavní myšlenka sociálního inženýrství je přinutit osobu, která zná heslo, k jeho vydání než se obtěžovat s používáním hrubé síly na prolomení hesla. Jako výhodu lze navíc brát, že při dobře vedeném útoku si oběť vůbec nemusí uvědomit, že vyzradila citlivé informace útočníkovi. Tento fakt lze pokládat za nejnebezpečnější rys problematiky sociálního inženýrství. Pokud je někomu ukradena/ztracena kreditní karta, nejběžnější reakce bude danou kartu zablokovat, aby nemohlo dojít k jejímu zneužití. Ale když je použito sociální inženýrství pro zisk informací, oběť se nemusí dozvědět o tom, že byla „okradena“ o informace. Sociotechnika je tedy velmi často nejjednodušší a nejlevnější způsob, jak ohrozit bezpečnost počítačové infrastruktury.

Sociální inženýrství je technika útoku, která využívá lidské chyby a nedostatky, aby se dostala k důvěrným informacím nebo získala přístup k chráněným systémům. Útočník využívá přesvědčivých technik, jako jsou vydávání se za jinou osobu, vytváření falešných důvodů, emotivní manipulace a sociálního tlaku, aby donutil oběť udělat něco, co by normálně neudělala. V případě útoku pomocí sociálního inženýrství může být obtížné zjistit, kdo je skutečným útočníkem, protože často se využívají anonymní metody, jako jsou anonymní e-mailové účty nebo falešné identity na sociálních sítích. To ztěžuje identifikaci útočníka a ztěžuje práci orgánům činným v trestním řízení.

Mezi časté metody útoků pomocí sociálního inženýrství patří vydávání se za jinou osobu, například za pracovníka banky, IT technika, nebo údržbáře, a žádání o přístup k důvěrným informacím, jako jsou hesla nebo bankovní údaje. Další metody zahrnují phishingové útoky prostřednictvím e-mailů nebo textových zpráv, které se snaží přesvědčit oběť o odeslání citlivých informací. [8]

2.1 Pretexting

Pretexting, také známý jako Blagging, je metoda, při které útočník používá předem vytvořený fiktivní scénář, aby přesvědčil oběť k poskytnutí citlivých informací nebo provedení činnosti, která bude pro útočníka užitečná.

Aby tato metoda byla úspěšná, musí útočník připravit a shromáždit informace, které bude používat ve scénáři. Tyto informace mohou zahrnovat jméno, adresu,

datum narození, telefonní číslo a veřejně zveřejněné příspěvky, které oběť sdílela na sociálních sítích a dalších platformách.

Pokud útočník z těchto příspěvků zjistí, že oběť měla špatnou zkušenost s bankou, může se vydávat za zaměstnance této banky a zavolat oběti, aby ji upozornil na možné ohrožení jejího účtu. Pokud útočník disponuje výše uvedenými informacemi, bude pro oběť snazší mu uvěřit a poskytnout další informace, jako například číslo účtu nebo heslo. Útočník může také použít techniku "spoofing"telefonního čísla, aby vzbudil v oběti větší důvěru.

V každém případě je pro úspěšné provedení této metody klíčové dobře připravit a koordinovat scénář a sbírat informace o oběti předem.[9]

2.2 Caller ID Spoofing

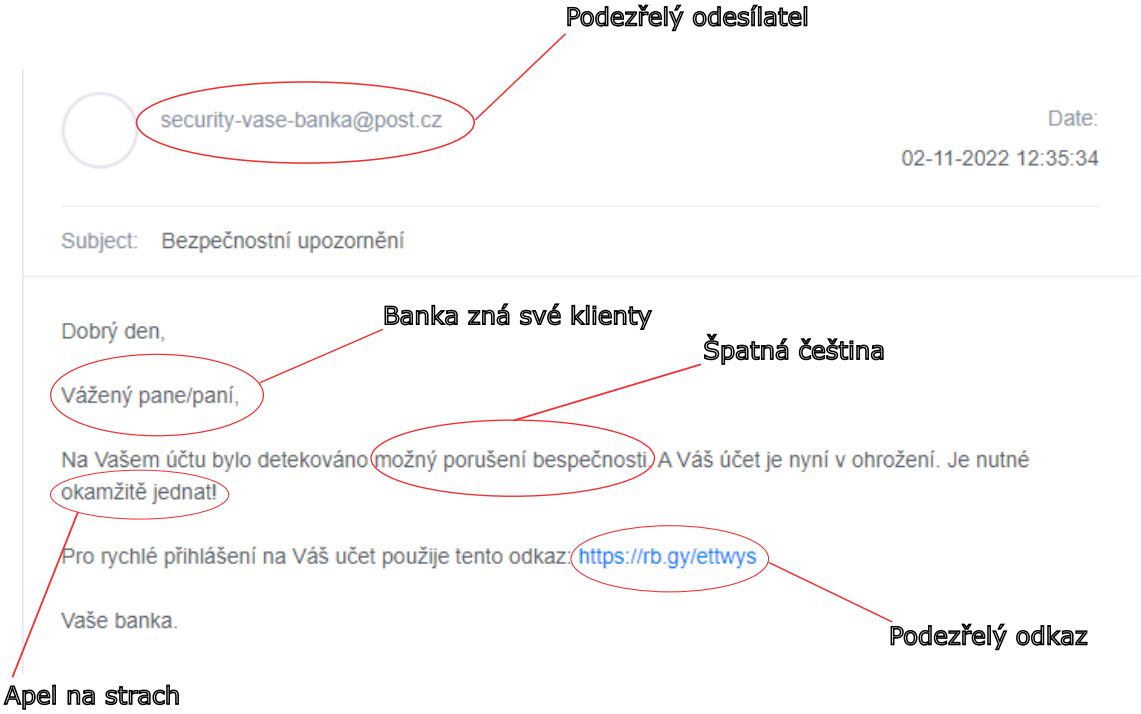
Tato metoda se často využívá v kombinaci s pretextingem a nazývá se spoofing. Spoofing spočívá v nahrazení čísla volajícího za číslo důvěryhodné osoby, například banky nebo policie. Útočník zavolá oběti a představí se jako pracovník banky, přičemž hovor jde přímo z banky oběti nebo se tak alespoň pro oběť jeví. Tím je oběť přesvědčena, že opravdu hovoří s pracovníkem banky a nemusí mít podezření. Výhodou spoofingu je, že pokud oběť zavolá zpět, skutečně se dovolá do své banky. Nicméně, spoofování čísel není jednoduché a často se využívají externí firmy, které mají potřebnou techniku a schopnosti, ale za úplatu.[10]

2.3 Phishing

Phishing je velmi využívaná metoda pro zjištění hesel pomocí internetových stránek. Útočník vytvoří internetovou stránku, která se tváří jako legitimní stránka banky. Často je tato stránka provázána s oficiální stránkou pro vytvoření větší důvěry. Pokud tedy oběť klikne na nějaký odkaz na stránce, je přesměrována na legitimní stránku banky. Při vložení údajů do podvodné stránky jsou všechny informace poslány útočníkovi a zároveň může dojít k přihlášení do internetového bankovnictví pomocí vložených údajů, které útočník přesměruje na legitimní stránku. Oběť se na takovéto stránky nejčastěji dostane kliknutím na odkaz v e-mailu, který vypadá jako zpráva od banky s vymyšleným scénářem jako je možné napadení účtu s nutností změnit heslo.

Phishing je nebezpečný v tom, že uživatelé klikají na odkazy bez rozmyslu. Útočníci spoléhají na strach oběti, při oznámení o napadení účtu nebude člověk reagovat tak, jak by reagoval v normální situaci a na odkaz bez rozmyslu klikne. Prevencí

proti phishingu je kontrolovat, kam daný odkaz doopravdy vede a zamyslet se nad tím, zda se opravdu jedná o zprávu z banky.



Obr. 2.1: Příklad podvodného e-mailu

2.4 Spear phishing

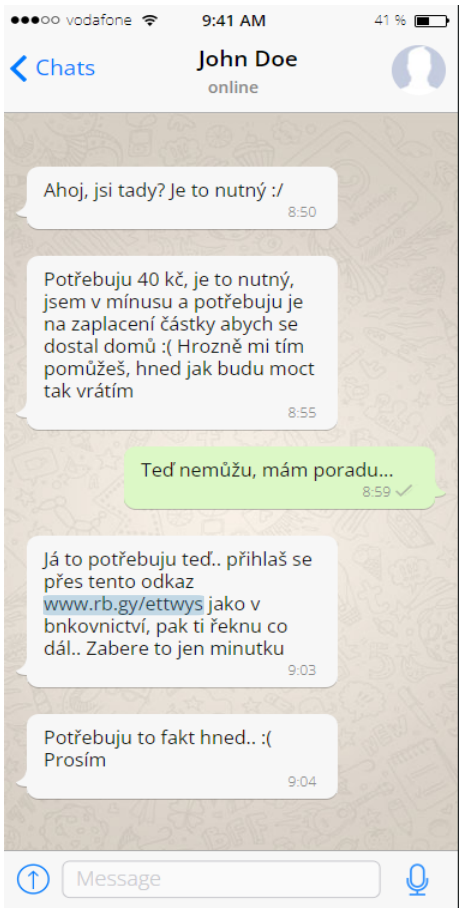
Je velmi podobná metoda získávání informací jako phishing s tím rozdílem, že phishing není cílen na určenou osobu, ale je rozesílán masivně. Oproti tomu spear phishing je mířen přímo na konkrétní osobu. Z toho vyplývá, že je nutný propracovanější scénář, a proto je tato metoda více nebezpečná než obyčejný phishing.

2.5 Smishing

Je forma phishingu provedená pomocí SMS zpráv. Pro oběť je snazší stát se obětí smishingu, protože lidé jsou více náchylní věřit SMS zprávám než zprávám, které přijdou přes e-mail. Při phishingu je také potřeba znát e-mail oběti, který může mít jakoukoliv délku a znění.

Oproti tomu smishing může být rozeslán masivně pomocí náhodně generovaných čísel. V ČR mají operátoři předvolbu a o to je jednodušší se trefit na nějaké číslo. Lze také použít mířený smishing na zájmovou osobu, ale zde je nutné již znát její telefonní číslo nebo se zmocnit účtu osoby, kterou má zájmová osoba již v kontaktech

a vydávat se za ní. Tyto mířené zprávy často obsahují prosbu o zaslání malé částky pro osobu v nouzi nebo přeposlání kódů, které přijdou oběti na zařízení. Zprávy obsahují odkaz na podvodnou stránku jako při phishingu.



Obr. 2.2: Mířený smishing

2.6 Vishing

Stejně jako smishing využívá mobilní telefon, ale zde již útočník přijde do kontaktu s obětí přes mobilní hovor. Při vishingu je vytvořen věrohodný scénář pomocí informací, které byly o oběti získány. Poté je uskutečněn hovor, kde se útočník může představit jako nový pracovník firmy, který zapomněl své heslo a nutně se potřebuje vzdáleně přihlásit do systému. IT pracovník firmy se bude snažit kolegovi pomoci a heslo mu může vyzradit. Nebo se role mohou obrátit a útočník zavolá pracovníkovi firmy a předstírá, že je z technické podpory a potřebuje přístup na zařízení oběti. Kroky podle kterých byl vytvořen takový scénář popisuje kapitola 5.2.11 [14]

2.7 Pharming

Název této metody je odvozen z anglických slov phishing a farming. Jedná se o podvod podobný phishingu, kde je oběť přesměrována na falešné stránky, kde jsou odcizeny citlivé údaje. Pharmingové útoky mohou být provedeny 2 způsoby. Při prvním útočník odešle oběti e-mail obsahující virus nebo trojského koně. Škodlivý kód poté pozmění soubory na hostitelském počítači takovým způsobem, že pokud oběť zadá adresu například své banky, bude přesměrována na falešné stránky. Tento způsob pharmingu je označován jako „pharming založený na malwaru“. Druhý způsob je založen na útoku zvaném DNS poisoning. DNS - Domain Name Systém, neboli systém doménových jmen je používán k překladu názvů domén na IP adresu. Pro uživatele je jednodušší si zapamatovat název domény jako www.vutbr.cz než její IP adresu 147.229.2.90. Tímto útokem lze infikovat DNS server takovým způsobem, že přepíše svůj záznam o IP adrese a tím odkazoval oběti na podvodné stránky.[11]

2.8 Quid Pro Quo

Quid Pro Quo neboli v doslovném překladu „něco za něco“ je technika využívaná prostřednictvím internetu, hovoru nebo přímého kontaktu. Útočník může aktivně obvolávat zaměstnance firmy a nabízet nějakou odměnu, nejčastěji peněžní, za provedení nějaké akce. Například sdělení přihlašovacích údajů, nainstalování backdooru nebo poskytnutí vzdáleného přístupu. Tato technika spoléhá na lidskou důvěřivost a chamtivost, možnost rychlého výdělku může oběť rychle zlákat a své údaje dobrovolně sdělí. Bohužel pro ni, slíbené peníze nakonec nedostane a v případě, že se přijde na to, že firma utrpěla škodu právě kvůli tomu, že oběť dobrovolně konala, tak jak jí útočník řekl s vidinou peněz ještě může být trestně stíhaná.[12]

2.9 Watering hole

Tento typ útoku není úplně běžný a je vysoce cílený na jednu osobu. Jméno této techniky bylo odvozeno od predátorů čekajících u napajedla pro svou kořist. Útočník čeká na specifickou událost, která bude obětí provedena, aby mohl zaútočit. Jak již bylo řečeno, tato technika je vysoce cílená, a proto je také nutná příprava.

Jako příklad lze uvést web donáškové služby, ze které si oběť objednává jídlo. Tuto informaci může zjistit pomocí trashingu. Poté se může pokusit najít zranitelnost na webu dodavatele a potom, co si oběť objedná jídlo, pomocí zranitelnosti dostat škodlivý software do zařízení oběti.[14]

2.10 Baiting

Je metoda útoku, která spoléhá na lidskou zvědavost. Útočník umístí paměťové zařízení (USB, CD) na místě, kde ho někdo najde. Zařízení má na sobě často lákavé označení jako „prémie březen“, oběť samozřejmě chce zjistit, jakou dostane prémii a jako dostanou její kolegové, a tak vloží zařízení do počítače. Tím však dojde k aktivaci skriptu, který na počítači nainstaluje malware nebo keylogger. Oběť se tak nedozví, kolik dostane peněz, ale ještě kompromituje zařízení. Známým případem baitingu je USB disk s programem, který zpozdil jaderný program Iránu o 2 roky.[15]

2.11 Trashing

Metoda Trashing se využívá pro získávání citlivých informací, jako jsou hesla, čísla kreditních karet, adresy, telefonní čísla a další údaje, které jsou důležité pro útočníka. Útočník hledá tyto informace ve vyhozených dokumentech, které lidé vyhazují do běžného odpadu. Tyto dokumenty mohou být například účetní doklady, výpis z bankovního účtu, obálky s osobními údaji a další podobné dokumenty.

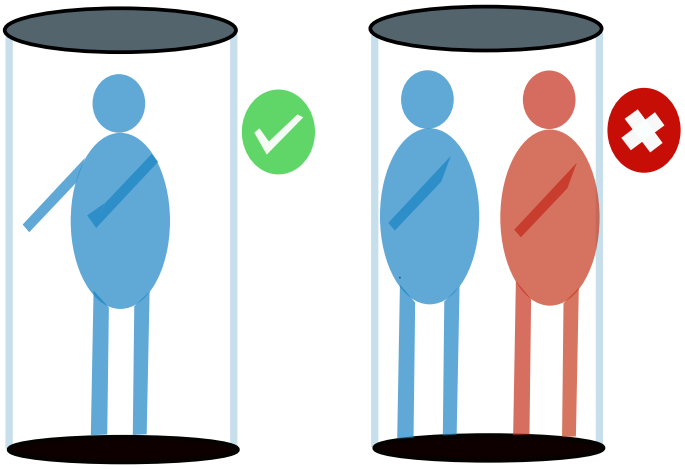
2.12 Tailgating a piggybacking

Tyto metody využívají fyzického kontaktu s obětí, nikoliv však pro získání citlivých informací, ale pro vpuštění do zabezpečené oblasti. Tyto techniky se používají hlavně ve velkých firmách, kde není možné znát všechny zaměstnance.

Tailgating spočívá v následování oběti do zabezpečené zóny v jejím závěsu. Nejčastěji pomocí karty oběť otevře dveře a útočník vstoupí hned za ní, protože on sám kartu nevlastní.

Piggybacking funguje na podobném principu, útočník předstírá, že kartu nechal doma, a tak poprosí zaměstnance venku, zda by mu nemohl otevřít svojí kartou. Tailgating je tedy nevědomé vpuštění osoby do objektu a piggybacking vědomé vpuštění osoby do objektu.[1]

Proti tomuto typu útoku se lze bránit politikou firmy. Firma Google po svých zaměstnancích vyžaduje, aby do zabezpečené oblasti vstupovali jednotlivě. Tedy pokud je nutné k otevření dveří přiložit kartu, tak zaměstnanec kartu přiloží, otevře dveře, vstoupí, a zavře je, i když hned za ním stojí někdo jiný. Druhou metodou je dvojí ověření. Před vstupem do místnosti je box nebo malá místnost pouze pro jednu osobu. Pokud se v boxu nachází více osob, systém nedovolí dveře otevřít ani po přiložení platné karty. Pokud je všechno v pořádku, osoba vstoupí do zabezpečené místnosti a další osoba může vstoupit do boxu [13].



Obr. 2.3: Box proti tailgatingu

2.13 Shoulder surfing

Metoda Shoulder surfing spočívá v pozorování osoby při zadávání citlivých informací, jako jsou hesla nebo čísla kreditních karet, aniž by si osoba byla vědoma pozorování. Tento útok se obvykle provádí na místech s velkým množstvím lidí, jako jsou nádraží, letiště nebo veřejné prostory s počítači, kde útočník může snadno získat a zapamatovat si klíčové informace.[8, 14]

2.14 Typy škodlivých kódů

Útočníci mohou používat různé typy škodlivých kódů pro získání informací o oběti, získání kontroly nad počítačem nebo přinucení oběti vykonat nějakou činnost. Tyto druhy kódů lze souhrnně označit jako škodlivé kódy. Pro pochopení názvosloví jednotlivých druhů škodlivých kódů byly tyto kódy popsány níže.

1. Virus – jedná se o kód se schopností replikace, bývá přiložen jako část přílohy souboru, kde po jeho spuštění dojde k replikaci kódu bez uživatelského vědomí. Jsou využívány k vytváření škodlivých akcí na uživatelském zařízení jako je vytěžování disku počítače, infikování dat nebo vytvoření „zombie“ PC. Zombie počítače mohou být poté aktivovány na dálku a mohou přes ně být vedeny útoky nebo je počítač pouze využit jako výpočetní síla pro DDos útok.
2. Trojský kůň – je částí legitimního softwaru bez schopnosti replikace. Je využíván jako nástroj pro připojení k infikovanému počítači, ke změně nebo destrukci dat nebo k narušení výkonosti zařízení.
3. Spyware – špionážní program ke zjištění dění na zařízení, může být použit

k legitimní činnosti jako je kontrola zaměstnanců v práci, ale útočníky je často využit pro krádež osobních dat. Jako příklad spyware lze uvést keylogger.

4. Scareware – je podvodná taktika s cílem vystrašit oběť, může být šířen pomocí e-mailu nebo pomocí vyskakovacích reklam při navštívení některých stránek. Při otevření odkazu je oběť přesměrována na stránky, kde mohou být další typy škodlivých kódů. Nejčastější obsah scarewaru je oznámení o infikování počítače virem a nutnosti rychle jednat.
5. Adware – existují dva typy adwaru, první je neškodný v podobě aplikace, která nabízí levné hry zákazníkům a profituje z jejich prodeje. Druhým typem je speciální typ spywaru, který nabízí konkrétní reklamy. Například při použití webového prohlížeče pro vyhledání dárku pro 5leté děti se poté na sociálních sítích začnou objevovat reklamy související s vyhledávanou tématikou.
6. Ransomware – škodlivý kód, který po spuštění zašifruje všechny soubory na zařízení, oběť tak ztratí přístup k veškerým datům. Tento kód není používán k získání informací o oběti, ale pomocí technik sociálního inženýrství je spuštěn na zařízení oběti. Ta je poté vyzvána k zaplacení „výkupného“ pro odemčení veškerých dat. Pro platbu jsou v současné době používány kryptoměny, pro jejich pseudoanonymitu, nejčastěji bitcoin. Po odeslání požadované částky útočníci data odemknou, v případě, že by po zaplacení data nebyla dešifrována, uživatelé zasaženi stejným ransomwarem by věděli, že nemá cenu výkupné platit, což je špatné pro „byznys“. Nejznámějším ransomwarem je WannaCry.
7. Worm – jedná se o speciální druh viru se schopností replikace a šíření na další zařízení přes počítačovou síť. Stejně jako virus dokáže vytvořit zadní vrátka na zařízení, zpomalit jeho výkonnost nebo může omezit rychlost přenosu po síti.[16, 17, 18]

V této části byly popsány techniky sociálního inženýrství hojně využívané v dnešní době společně s uvedením některých možných scénářů útoků a možností obrany, podle kterých lze vidět, že sociální inženýrství není jen prostřednictvím internetu, ale dá se aplikovat i pomocí fyzického kontaktu. Tyto obecně popsané metody lze aplikovat na netušící oběti, avšak hlavně při fyzickém kontaktu je nutná improvizace útočníka. Tyto metody je dobré znát pro jejich odhalení. Byly popsány typy škodlivých kódů využívaných pro získání kontroly nad zařízením nebo pro manipulaci s obětí, mezi kterými je dobré rozlišovat. Praktická část se zaměřuje ne metodu Vishing.

Obr. 2.4: WannaCry Ransomware [19]

Obr. 2.5: Scareware na webové stránce

3 Red Teaming

Cílem této práce je vytvoření aplikace pro Red Teaming. Z toho důvodu je nutné vědět o co se jedná, jaké jsou rozdíly oproti penetračnímu testování, jaké subjekty figurují při testování zabezpečení a jaké jsou aktivity, kterých je třeba během testování dosáhnout.

Jako červený tým lze označit skupinu lidí, kteří se snaží zlepšit jak fyzické tak kybernetické zabezpečení firmy prostřednictvím útoků na danou instituci, stejným způsobem jako by byl veden útok hackerskou skupinou.

3.1 Penetrační testování vs Red Teaming

Při testování bezpečnosti sítě jsou často zaměňovány pojmy penetrační testování a Red Teaming. I když tyto metody používají podobné principy, každá se liší svým přístupem.

Penetrační testování je metoda testování zabezpečení, která umožňuje organizacím zjistit úroveň zabezpečení sítí, platforem, hardwaru, aplikací a dalších systémů v předem definovaném rozsahu s cílem identifikovat zranitelnosti. Pro tento účel se používají penetrační testeři, kteří postupují podle jasně definované metodiky, například OWASP, PTES nebo OSSTMM. Tyto testy nejsou skryté a správci ICT příslušné organizace o nich vědí, případně se na nich mohou podílet.

Na druhé straně Red Teaming je aktivita, která se snaží co nejvíce napodobit chování útočníků, kteří by se mohli pokusit získat přístup do sítě. Tato metoda testování zabezpečení se zaměřuje nejen na kybernetické, ale také na fyzické zabezpečení a využívá také sociálního inženýrství. Cílem Red Teamingu není pouze identifikovat zranitelnosti, ale také ověřit, jak organizace reaguje na hrozby a jak úspěšně může bránit svá aktiva. Testování je často prováděno skrytě a testovací tým má k dispozici dokumenty, které dokazují, že provádí testování zadané firmou a jsou oprávněni v objektu pobývat. Tyto dokumenty jsou v případě odhalení předloženy ochrance objektu, ale někdy i tak může dojít k nechtěným incidentům.

Red Teaming se také používá pro trénink interních týmů v reakci na dané události a hodnotí se správnost zvoleného postupu v souladu s incident response planem[1]. Při této metodě se také často používá MITRE ATTACK a MITRE DEFENSE pro identifikaci hrozeb a návodů na zajištění bezpečnosti. MITRE ATTACK popisuje útoky a taktiky používané útočníky, zatímco MITRE DEFENSE poskytuje návody na zajištění bezpečnosti pro obranu proti těmto útokům. Tyto frameworky mohou být použity pro identifikaci hrozeb a vytvoření lepšího incident response plánu. Při

[1]Incident response plan (IRP) je dokument, který popisuje postupy, kroky a odpovědnosti při reakci na kybernetický útok nebo jinou bezpečnostní událost.

testování zabezpečení se často používají společně s jinými metodami jako jsou penetrační testování a Red Teaming.

Je důležité si uvědomit, že penetrační testování a Red Teaming jsou dvě různé metody testování zabezpečení, které mohou být použity v různých situacích a mají odlišné cíle [20].

3.2 Typy týmů

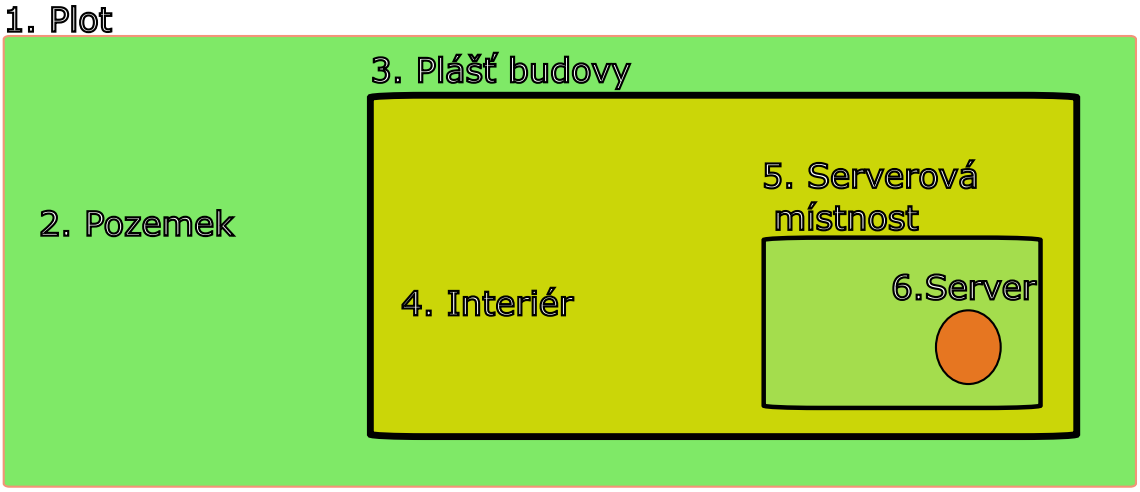
Testování zabezpečení pomocí Red Teamingu si lze představit jako hru na kočku a myš, ve které jsou 3 týmy - červený, modrý a bílý.

1. Červený tým - Tým útočníků, kteří mají za úkol napadnout organizaci a získat určený cíl. Často se jedná o informace, přístupové údaje nebo data, která mohou být použita pro škodlivé účely. Červený tým se snaží využít všech dostupných technik a postupů, aby byl útok úspěšný.
2. Modrý tým - Obranný tým, jehož úkolem je chránit organizaci před útoky. Tento tým je složen z odborníků na zabezpečení sítě a IT infrastruktury, kteří mají za úkol identifikovat a odvrátit útoky. Modrý tým také sleduje bezpečnostní události a incidenty a zajišťuje reakci na ně. Tyto týmy jsou často označovány jako Security Operations (SOC) či Cyber Defense Centrum (CDC).
3. Bílý tým - Tým organizátorů testování, kteří určují pravidla a cíle útoku. Bílý tým zajišťuje koordinaci a komunikaci mezi červeným a modrým týmem. Tento tým také zajišťuje transparentnost a dokumentaci všech kroků v rámci testování a poskytuje zprávu s výsledky.

3.3 Zabezpečení objektu

Red Team testeři provádějí i testování fyzického zabezpečení objektu s cílem nepovoleného vstoupení do objektu nebo areálu firmy pro zisk vlajky. Získání fyzického přístupu do objektu není vždy jednoduché (pokud neexistuje někdo, kdo nám dovnitř pomůže). Je totiž nutné překonat množství zabezpečení. Jednotlivé vrstvy zabezpečení popisuje obrázek níže kde je jako příklad uveden server, ke kterému by testeři chtěli získat fyzický přístup. Pro získání přístupu k serveru je nutné většinou překonat 6 linií překážek.

1. Překonání plotu
2. Průchod přes pozemek
3. Průnik pláštěm budovy



Obr. 3.1: Zabezpečení objektu

4. Průchod interiérem
5. Vstup do místnosti se serverem
6. Napojení na server

Při překonávání linií 1–5 může být útočník odhalen pomocí detektorů (pohybu, tepla, tříštění skla, otřesů), které automaticky spustí poplach, nebo kamer připojených na vrátnici obsluhovaných pověřenou osobou. U větších firem se lze setkat s liniemi 1-6 a u menších s 3-6, když firma sídlí například v centru bez pozemku.

Překonání všech linií nepozorovaně může být časově náročné a existuje velké riziko odhalení útočníka. Proto je mnohem jednodušší využít techniky sociálního inženýrství a do objektu být vpuštěn ostrahou objektu (falešná zabezpečovací firma), pokusit se o tailgaiting a vstoupit do objektu neautorizovaně, získat přístup do firemní sítě pomocí zadních vrátek nebo pomocí phishingové kampaně.[2][5]

3.4 Cyber Kill Chain

Stejně jako existují metodiky pro penetrační testování - OWASP, OSSTMM, atd., tak existují i různé frameworky pro Red Teaming. V této části bude popsán framework Cyber Kill Chain (CKC), který se oproti ostatním frameworkům vyznačuje svou přímočarostí. Součástí CKC je 7 aktivit, které je nutné vykonat Red Teamem pro splnění úspěšného útoku. Tyto aktivity se naopak Blue Team snaží narušit.

[2]Citlivé informace se mohou nacházet na serveru, který je od sítě odpojen a je k němu nutné získat fyzický přístup.

Těmto aktivitám předchází vymezení cíle, kterého má být dosaženo a jaké techniky mohou být použity. Po těchto aktivitách následuje konečná zpráva, kde je popsáno jak byl proveden útok, i v případě, že nebyl úspěšný a navrženo opatření.

Aktivity obsažené v CKC jsou následující:

1. Reconnaissance (Průzkum) - Sběr informací pro provedení útoku. O této aktivitě více pojednává kapitola 3.4.1
2. Weaponization (Vyzbrojení) - Vytvoření vektoru útoku na míru, může se jednat o virus, malware, červa, apod.
3. Delivery (Doručení) - Spuštění útoku, především se jedná o phishingové e-maily.
4. Exploitation (Zneužití) - Spuštění kódu na zařízení, který byl doručen v předchozím kroku. Může se jednat buď o špatné zabezpečení sítě nebo o lidskou chybu.
5. Installation (Instalace) - Instalace malwaru nebo backdooru pro zajištění přístupu.
6. Command and Control (Velení a řízení) - Navázání komunikace směrem ven ze sítě pro využití nainstalovaného malwaru, tak aby mohl být ovládán Red Teamem. Pro toto řízení se používají především protokoly DNS a HTTP pro jejich nenápadnost. V tuto chvíli útočník může postupovat dále sítí, rozšiřovat backdoor na další zařízení či eskalovat svoje práva pro dosažení cíle.
7. Actions on Objectives (Akce týkající se cílů) - Podnikání kroků k získání vlajky, tedy dosažení cíle, který byl stanoven v zadání.

3.4.1 Reconnaissance - Průzkum

Stejně jako při plánování vyloupení banky je velice nutná příprava a stejně je tomu tak i při útoku realizovaného Red Teamem. Jsou využívány veřejně dostupné i neveřejné zdroje pro zjištění co nejvíce informací o cíli. K získání těchto informací jsou používány sociální sítě jako Facebook, Instagram, Twitter, LinkedIn, techniky Open-source intelligence (OSINT) a další nástroje.

Součástí průzkumu je i zjištění informací o zaměstnancích. Tyto informace jsou buď poskytnuty White Teamem nebo je nutné je zjistit jiným způsobem. Zároveň jsou zjišťovány služby a technologie používané ve firmě. [20, 21]

Sběr konkrétních informací

Při útoku na organizaci není útok prováděn na organizaci samotnou, ale pouze na individuální pracovníky.
Zdrojem informací pro útok podle OSINT jsou: [22]

1. Veřejně dostupné databáze
   (a) Soudní záznamy
   (b) Politické příspěvky
   (c) Profesionální licence a registrace
2. Sociální sítě
   (a) Metadata
   (b) Charakter osoby
   (c) Frekvence přidávání příspěvků
   (d) Navštívená místa
   (e) Přítomnost na sociálních sítích
3. Používání internetu
   (a) E-mail
   (b) Přezdívky
   (c) Registrované domény
   (d) IP adresa
4. Bydliště
5. Mobilní stopa
   (a) Telefonní číslo
   (b) Typ zařízení
   (c) Nainstalované aplikace
   (d) Administrátorská oprávnění
6. Informace za zaplacení

V této kapitole bylo vysvětleno co je Red Teaming a jeho rozdíly oproti penetračnímu testování. Byly popsány funkce červeného, modrého a bílého týmu. Byly vysvětleny aktivity obsažené ve frameworku Cyber Kill Chain s důrazem na nejdůležitější část zjišťování informací a bylo popsáno zabezpečení objektu. V praktické je nejvíce používanou aktivitou z CKC Delivery.

4 Legislativa

Jak již bylo zmíněno v předchozí kapitole, Red Teaming je činností, která vyžaduje velkou opatrnost a dodržování platných zákonů a nařízení. Tato kapitola se zaměřuje na právní rámec, který se týká sociálního inženýrství a souvisejících aktivit. V České republice neexistuje žádná zvláštní legislativa pro sociální inženýrství, ale to neznamená, že by se tato činnost nemusela řídit určitými normami a principy.
V této kapitole jsou přiblíženy platné zákony a nařízení, které mohou být aplikovány na Red Teaming a sociální inženýrství obecně. Zároveň se tato kapitola zaměřuje na konkrétní paragrafy, které je nutné dodržovat, aby nedošlo k porušení zákona při technice Vishing.

4.1 Občanský zákoník

Zákon č. 89/2012 Sb. občanský zákoník
§ 2586 Smlouva o dílo
Pro provádění činností v oblasti Red Teamingu je nutné mít vytvořenou smlouvu o dílo, která definuje základní parametry projektu a zajišťuje soulad s platnými zákony a obecnými ustanoveními. Tato smlouva je základním právním nástrojem, kterým se zhotovitel zavazuje k vytvoření určitého díla a zadavatel k zaplacení za provedenou práci.

Smlouva o dílo slouží v Red Teamingu k definování cílů projektu a stanovení konkrétních výsledků, kterých má být dosaženo. V smlouvě jsou specifikovány požadované testy, rozsah práce a termíny pro doručení výsledků. Je důležité stanovit v smlouvě podmínky a termíny platby za provedenou práci, aby zhotovitel byl motivován k dokončení práce včas a v souladu se specifikacemi. Smlouva o dílo také chrání obě strany před možnými sporovými situacemi.

Výsledkem smlouvy o dílo v Red Teamingu je jasně definovaný projekt s konkrétními cíli a výsledky.

§ 86
Nikdo nesmí zasáhnout do soukromí jiného, nemá-li k tomu zákonný důvod. Zejména nelze bez svolení člověka narušit jeho soukromé prostory, sledovat jeho soukromý život nebo pořizovat o tom zvukový nebo obrazový záznam, využívat takové či jiné záznamy pořízené o soukromém životě člověka třetí osobou, nebo takové záznamy o jeho soukromém životě šířit. Ve stejném rozsahu jsou chráněny i soukromé písemnosti osobní povahy.[23]

S tímto paragrafem se lze setkat nejčastěji při kontaktní formě sociálního inženýrství, kdy se útočník snaží získat informace pomocí rozhovoru s obětí. Jako příklad lze uvést scénář, kdy se útočník uchází o pozici vrátného a tak se jako zájemce baví s pracovníkem ostrahy, kde se pomocí rozhovoru snaží získat potřebné informace (pauza na kafe, střídání směn apod.). Při takovém rozhovoru ale může být složité si zapamatovat všechny informace, obvzlášť pokud je pracovník velmi sdílný, a proto je možné rozhovor nahrát, čímž se útočník dopouští neoprávněného zásahu do soukromí, pokud rozhovor nahrává bez vědomí a souhlasu dotčené osoby. Pokud je ale účelem nahrávky získat informace za účelem ochrany majetku, lze tak učinit, pokud jsou splněny některé podmínky. Dotčená osoba musí být předem informována o účelu nahrávky a musí s nahráváním souhlasit. Nahrávka je po dokončení následně zpracována a poté zničena. Pro minimalizaci rizika zásahu do soukromí dotčené osoby, lze vytvořit smlouvu o dílo se zaměstnavatelem a útočníkem, který svolí nahrávání za účelem ochrany majetku.

4.2 Trestní Zákoník

K trestné činnosti **Zákona č. 40/2009 Sb. Zákon trestní zákoník,** je vztaženo mnoho zákonů, legislativ, nařízení a ustanovení. V této podkapitole budou vypsány trestné činy a obecná ustanovení a k nim uvedeny příklady, se kterými je možné se setkat při sociálním inženýrství.

4.2.1 Obecná ustanovení

§ 29 Nutná obrana
(1) Čin jinak trestný, kterým někdo odvrací přímo hrozící nebo trvající útok na zájem chráněný trestním zákonem, není trestným činem.
(2) Nejde o nutnou obranu, byla-li obrana zcela zjevně nepřiměřená způsobu útoku.

Tento paragraf se nevztahuje přímo na útočníky, ale na osoby, které se starají o infrastrukturu pod útokem. Pokud je například veden útok na firemní server a není možné ho zastavit jiným způsobem (např. ukončením spojení nebo uvedením IP na blacklist), může být na útočníka použita reakce, tzv. **Hack Back**. Protože se bude jednat o nutnou obranu, nebude to trestný čin. Nicméně nutná obrana musí být úměrná nebo mírně větší než přicházející útok, jinak se nejedná o nutnou obranu. Pro lepší pochopení nepřiměřené obrany je uveden příklad z jiného prostředí. Když je osoba napadena pěstmi, použití automatické zbraně by bylo považováno za nepřiměřenou obranu.

§ 30 Svolení poškozeného

(1) Trestný čin nespáchá, kdo jedná na základě svolení osoby, jejíž zájmy, o nichž tato osoba může bez omezení oprávněně rozhodovat, jsou činem dotčeny.

(2) Svolení podle odstavce 1 musí být dáno předem nebo současně s jednáním osoby páchající čin jinak trestný, dobrovolně, určitě, vážně a srozumitelně; je-li takové svolení dáno až po spáchání činu, je pachatel beztrestný, mohl-li důvodně předpokládat, že osoba uvedená v odstavci 1 by tento souhlas jinak udělila vzhledem k okolnostem případu a svým poměrům.

Tento paragraf je uveden z důvodu Red Teamingu popsaného v kapitole 3, kdy je nutné mít vytvořenou smlouvu se zadavatelem. Nebo při soutěžích, kde firmy svolí k útoku na jejich služby za cílem zvýšení bezpečnosti. Pokud je takové svolení uděleno, útočníci nepáchají trestnou činnost, pokud nepřekročí rozsah toho, co jim bylo povoleno. Bez tohoto paragrafu by nebylo možné provádět penetrační testování sítě bez možných právních následků.

§ 120 Uvedení někoho v omyl a využití něčího omylu prostřednictvím technického zařízení

Uvést někoho v omyl či využít něčího omylu lze i provedením zásahu do dat uložených v počítačovém systému nebo na nosiči informací, zásahu do programového nebo technického vybavení počítačového systému nebo provedením jiné operace v počítačovém systému, zásahu do elektronického nebo jiného technického zařízení, včetně zásahu do předmětů sloužících k ovládání takového zařízení, anebo využitím takové operace či takového zásahu provedeného jiným.

S tímto paragrafem se lze setkat například při technice Vishing 2.6, kdy se útočník připojí pomocí vzdálené plochy na počítač po přesvědčení zaměstnance, že se jedná o správce sítě, který na něho potřebuje nutně přístup nebo při kontaktní formě, kdy je mu umožněn pod falešnou záminkou přístup na počítač. Pokud by poté bylo nějak manipulováno s daty, jedná se o trestný čin.

4.2.2 Trestné činy

§ 182 Porušení tajemství dopravovaných zpráv

(1) Kdo úmyslně poruší tajemství

b) datové, textové, hlasové, zvukové či obrazové zprávy posílané prostřednictvím sítě elektronických komunikací a přiřaditelné k identifikovanému účastníku

nebo uživateli, který zprávu přijímá, nebo

c) neveřejného přenosu dat do počítačového systému, z něj nebo v jeho rámci, včetně elektromagnetického vyzařování z počítačového systému, přenášejícího taková data, bude potrestán odnětím svobody až na dvě léta nebo zákazem činnosti.
(2) Stejně bude potrestán, kdo v úmyslu způsobit jinému škodu nebo opatřit sobě nebo jinému neoprávněný prospěch

a) prozradí tajemství, o němž se dozvěděl z písemnosti, telegramu, telefonního hovoru nebo přenosu prostřednictvím sítě elektronických komunikací, který nebyl určen jemu, nebo

b) takového tajemství využije.

Samotné čtení zpráv, které nebyly určeny pro nás jako příjemce, je trestné, pokud k tomu nebyl udělen souhlas. Pomocí phishingu může být zjištěno heslo, které může být využito pro přístup na e-mail. Čtením zpráv v této e-mailové schránce je porušení tohoto paragrafu. Stejně tak lze tento paragraf uplatnit i na sociální sítě nebo SMS. Typickým způsobem, jakým se tento zákon porušuje, je sniffing, tedy zachytávání komunikace v síti, což umožňuje získat citlivé informace nejen o provozu, ale i o obsahu. [24].

§ 183 Porušení tajemství listin a jiných dokumentů uchovávaných v soukromí

(1) Kdo neoprávněně poruší tajemství listiny nebo jiné písemnosti, fotografie, filmu nebo jiného záznamu, dat uložených v počítačovém systému nebo na nosiči informací anebo jiného dokumentu uchovávaného v soukromí jiného tím, že je zveřejní, zpřístupní třetí osobě nebo je jiným způsobem použije, bude potrestán odnětím svobody až na jeden rok, zákazem činnosti nebo propadnutím věci.
(2) Odnětím svobody až na dvě léta, zákazem činnosti nebo propadnutím věci bude pachatel potrestán, spáchá-li čin uvedený v odstavci 1 v úmyslu získat pro sebe nebo pro jiného majetkový nebo jiný prospěch, způsobit jinému škodu nebo jinou vážnou újmu, anebo ohrozit jeho společenskou vážnost.

Tento paragraf se při použití technik sociálního inženýrství příliš často neuplatňuje, neboť pro splnění jeho podmínek je nutné zveřejnit získaný obsah zpráv. Nicméně, je uveden z důvodu možnosti využití technik sociálního inženýrství pro získání hesla nebo přístupu k informacím. Například hackeři mohou zveřejnit obsah zpráv politika, aby odhalili jeho lži před veřejností.

§ 209 Podvod

(1) Kdo sebe nebo jiného obohatí tím, že uvede někoho v omyl, využije něčího omylu

nebo zamlčí podstatné skutečnosti, a způsobí tak na cizím majetku škodu nikoli nepatrnou, bude potrestán odnětím svobody až na dvě léta, zákazem činnosti nebo propadnutím věci.

Jako příklad lze uvést scénář využívající pretexting, také známý jako "blagging"2.1. Pachatel se vydává za ředitele společnosti nebo jinou důvěryhodnou osobu s odpovídajícími pravomocemi, aby navázal kontakt. Pod touto záminkou kontaktují zaměstnance firmy s tím, že jim byl zadán úkol od jejich nadřízeného, aby je přiměli k předání citlivých informací nebo k účasti na jiných útočných aktivitách.

Druhým příkladem mohou být podvodné e-shopy, které nabízejí brigády, kde je náplní práce zakládání účtů. Pachatelé poté přeposílají výdělky z nelegálních aktivit na tyto účty, aby byli hůře vystopovatelní. Tento druh útoku je znám jako "money muling"[25]

§ 230 Neoprávněný přístup k počítačovému systému a neoprávněný zásah do počítačového systému nebo nosiče informací

(1) Kdo překoná bezpečnostní opatření, a tím neoprávněně získá přístup k počítačovému systému nebo k jeho části, bude potrestán odnětím svobody až na dvě léta, zákazem činnosti nebo propadnutím věci.

(2) Kdo zasáhne do počítačového systému nebo nosiče informací tím, že

a) neoprávněně užije data uložená v počítačovém systému nebo na nosiči informací,

b) data uložená v počítačovém systému nebo na nosiči informací neoprávněně vymaže nebo jinak zničí, poškodí, změní, potlačí, sníží jejich kvalitu nebo je učiní neupotřebitelnými,

c) padělá nebo pozmění data uložená v počítačovém systému nebo na nosiči informací tak, aby byla považována za pravá nebo podle nich bylo jednáno tak, jako by to byla data pravá, bez ohledu na to, zda jsou tato data přímo čitelná a srozumitelná, nebo

d) neoprávněně vloží nebo přenese data do počítačového systému nebo na nosič informací nebo učiní jiný zásah do programového nebo technického vybavení počítačového systému nebo jiného technického zařízení pro zpracování dat, bude potrestán odnětím svobody až na tři léta, zákazem činnosti nebo propadnutím věci.

Tento paragraf se týká neoprávněného přístupu k počítačovému systému a zásahu do počítačového systému nebo nosiče informací, což zahrnuje mnoho aktivit, které jsou obvykle označovány jako hacking. Mezi takové aktivity patří například narušení, změna nebo smazání dat, přenos dat a podobně. Často se prověřuje, zda bylo překonáno zabezpečení systému a zda byly získány informace o oběti. Pod tuto ka-

tegorii spadají také kybernetické útoky, jako jsou DDoS útoky nebo zašifrování dat na zařízení pomocí ransomware[25] [26].

§ 231 Opatření a přechovávání přístupového zařízení a hesla k počítačovému systému a jiných takových dat

(1) Kdo vyrobí, uvede do oběhu, doveze, vyveze, proveze, nabízí, zprostředkuje, prodá nebo jinak zpřístupní, sobě nebo jinému opatří nebo přechovává

a) zařízení nebo jeho součást, postup, nástroj nebo jakýkoli jiný prostředek, včetně počítačového programu, vytvořený nebo přizpůsobený k neoprávněnému přístupu do sítě elektronických komunikací, k počítačovému systému nebo k jeho části nebo k neoprávněnému zásahu do počítačového systému nebo nosiče informací, nebo

b) počítačové heslo, přístupový kód, data, postup nebo jakýkoli jiný podobný prostředek, pomocí něhož lze získat přístup k počítačovému systému nebo jeho části, v úmyslu, aby jej bylo užito ke spáchání trestného činu porušení tajemství dopravovaných zpráv podle § 182 odst. 1 písm. b) nebo c) nebo trestného činu neoprávněného přístupu k počítačovému systému a neoprávněného zásahu do počítačového systému nebo nosiče informací podle § 230 odst. 1 nebo 2, bude potrestán odnětím svobody až na dvě léta, propadnutím věci nebo zákazem činnosti.

Tento paragraf se zabývá trestnými činy souvisejícími s opatřením a přechovávání přístupových zařízení a hesel k počítačovému systému. Mezi typické příklady patří neoprávněné získání přístupových údajů pomocí technik sociálního inženýrství popsaných v kapitole 2, jejich přechovávání a použití pro neoprávněný přístup k počítačovému systému nebo k datům v něm uloženým. Legální opatření hesla samo o sobě není trestným činem, dokud není zjištěno, že bude použito pro neoprávněný přístup k počítačovému systému [26].

§ 232 Neoprávněný zásah do počítačového systému nebo nosiče informací z nedbalosti

(1) Kdo z hrubé nedbalosti porušením povinnosti vyplývající ze zaměstnání, povolání, postavení nebo funkce nebo uložené podle zákona nebo smluvně převzaté

a) data uložená v počítačovém systému nebo na nosiči informací zničí, poškodí, pozmění nebo učiní neupotřebitelnými, nebo

b) učiní zásah do technického nebo programového vybavení počítačového systému nebo jiného technického zařízení pro zpracování dat,a tím způsobí na cizím majetku značnou škodu, bude potrestán odnětím svobody až na šest měsíců, zákazem činnosti nebo propadnutím věci.

Paragraf § 232 se vztahuje na případy neoprávněného zásahu do počítačového sys-

tému nebo nosiče informací z nedbalosti. Tento paragraf je významný zejména v oblasti Red Teamingu, kdy jsou testovány zabezpečovací mechanismy počítačových systémů. Pro splnění trestného činu musí být porušena povinnost vyplývající ze zaměstnání, povolání nebo smlouvy a musí dojít k zavinění z nedbalosti, tedy zapomenutí nebo přímé ignorování povinností, což vede ke škodě. Hlavně se jedná o případy, kdy ICT správci, administrátoři nebo právě Red Team testeři neplní své povinnosti a tím způsobují škodu [27].

Jak již bylo řečeno v závěru kapitoly 2 aplikace, která je popsána v praktické části této práce se zaměřuje na techniku Vishing. Pro tuto techniku byly vybrány jako nejvíce rizikové paragrafy **§ 209** a **§ 230** a jako potencionálně rizikové paragrafy **§ 182** a **§ 231**. Zaměstnanci, kteří budou pracovat s touto aplikací by proto měli být řádně proškoleni v oblasti těchto právních paragrafů a měly by být stanoveny interní postupy firmy, aby se minimalizovalo riziko porušení těchto paragrafů. Vishingové scénáře se mohou značně lišit i když jsou založené na stejném principu. Proto by zaměstnanci, kteří budou vytvářet tyto scénáře měli být poučeni, aby se v případě nejasností nebo pochyb obrátili na svého nadřízeného nebo právníka firmy pro konzultaci dalšího postupu.

V této kapitole byly popsány paragrafy týkající se sociálního inženýrství, které je možné při jeho praktikování porušit. Definovány nejvíce rizikové paragrafy týkající se techniky Vishing a doporučen postup jak by se měli zaměstnanci chovat. V praktické části je popsáno vytvoření aplikace, která pomáhá k zajištění toho, že nedojde k porušení paragrafů definovaných pro Vishing při testování zaměstnanců za pomocí Red Teamingu.

5 Implementace a zabezpečení aplikace

Praktická část této práce se věnuje vývoji aplikace pro vytváření a zobrazování vektorů útoků za pomocí metod sociálního inženýrství. Tato aplikace bude sloužit pro potřeby Red Teamingu a měla by usnadnit testování zaměstnanců. Zároveň je vytvořeno programové řešení, které by mělo pomoci se zajištěním, že testování bude probíhat v souladu s právem, v rozsahu a za použití technik, které budou stanoveny zadavatelem v smlouvě o dílo.

5.1 Využité frameworky

Tato kapitola se věnuje popisu využitých frameworků a knihoven použitých při vytváření aplikace a uvedení některých výhod a nevýhod při použití daných frameworků.

5.1.1 Python

Programovací jazyk, ve kterém je aplikace napsaná je Python. Jedná se o vysokoúrovňový, objektově orientovaný programovací jazyk s dynamickou sémantikou. Jeho syntaxe je jednoduše naučitelná, čímž z něho činí velice atraktivní jazyk pro vývoj aplikací, vytváření skriptů nebo spojovací jazyk pro propojení existujících komponent. Jazyk Python je volně dostupný a obsahuje debugger, který je napsán v jazyce Python, určený pro odladění kódu a rychlé kontroly proměnných pro nalezení chyb [28].

5.1.2 Pycharm

Jedná se o integrované vývojové prostředí (IDE) pro programovací jazyk Python, které poskytuje velkou škálu základních nástrojů pro programátory píšící v Pythonu. Pycharm poskytuje pohodlné prostředí pro produktivní vývoj aplikací psaných v Pythonu. Programovací prostředí bylo vytvořeno českou společností JetBrains a je podporováno na operačních systémech Windows, Linux a macOS. Od roku 2013 lze získat volně dostupnou (open-source) verzi [29].

5.1.3 SQLite

Jedná se o knihovnu napsanou v jazyce C, která je součástí vývojového prostředí Pycharm. Tato knihovna implementuje rychlý, samostatný a vysoce spolehlivý databázový stroj SQL, který je nejpoužívanější databázový stroj na světě. V současné

době se používá více než 1 bilión databázi SQL. Zdrojový kód SQLite je volně dostupný komukoliv pro libovolné účely [30].

5.1.4 Tkinter

Je knihovna, která je součástí jazyku Python, určená pro vytváření grafického rozhraní (GUI), stejně jako je uživatel zvyklý v prostředí Windows. Tato knihovna je jednoduchá k používání a určená především pro začátečníky v oblasti grafického programování.

5.1.5 CustomTkinter

CustomTkinter je rozšiřující knihovna pro pro vytváření grafického rozhraní založená na knihovně Tkinter. Poskytuje jednotlivým prvkům v grafickém rozhraní moderní vzhled a je možné používat prvky od obou knihoven současně. Knihovna CustomTkinter není součástí vývojového prostředí Pycharm a je nutné ji pro používání nainstalovat příkazem:

```
pip3 install customtkinter
```

Knihovna je pod stálým vývojem a neobsahuje všechny grafické prvky jako Tkinter [31].

5.1.6 Nuitka

Pro převedení souborů vytvořených v Pycharmu na spustitelný soubor bez nutnosti instalace je použit optimalizační kompilátor pro Python Nuitka. Nuitka převede kód napsaný v Pythonu do nativního kódu pro operační systém, na kterém je spuštěna. Výsledný program je poté možné spustit i na zařízení, které nemá nainstalovaný interpret Pythonu, protože se spouští přímo jako aplikace. Aplikace převedená tímto způsobem je mnohem rychleji, protože není nutné spouštět interpret Pythonu pro každou instrukci [32].

5.1.7 Bitly

Bitly je online služba, která slouží ke zkrácení URL adres. Tato služba také poskytuje statistiky o kliknutí na zkrácené odkazy, což umožňuje sledovat kolik uživatelů na odkaz kliklo. Tato služba je běžné používána na sociálních sítích, e-mailové komunikaci a SMS komunikaci. Zároveň služba disponuje vlastním API, které bylo využito v této práci. Pro používání Bitly API je nutné mít účet na Bitly a API token pro ověření [33].

5.1.8 Textbelt

Pro odesílání SMS je využita služba API Textbelt. Tato služba umožňuje odeslání jedné SMS denně zdarma nebo více SMS po vytvoření účtu a nahrání kreditů. Díky využití služby API Textbelt v této práci je umožněno rychlé a snadné odesílání SMS zpráv. Bohužel bez účtu, který obsahuje kredity není možné využít API Textbelt pro odeslání SMS zprávy s krátkým odkazem, který byl vytvořen pomocí služby Bitly. Pro odeslání SMS zprávy zdarma, která by obsahovala odkaz vytvořený pomocí Bitly, byly testovány různé stránky a jejich API. Bohužel žádná taková služba nebyla nalezena, nebo se vyskytly větší překážky, jako nutnost registrace telefonního čísla, na které má být SMS odeslána [34].

5.2 Vývoj aplikace

Aplikace vytvářená v této práci umožňuje základní funkce jako je vytvoření uživatele, vytvoření scénáře pro útok, zadávání úkolů skupinám nebo jednotlivci, výpis zpráv o provedeném testování, možnost práce se smluvní tématikou a další funkce, které podporují funkčnost a užitečnost aplikace. V následujících podkapitolách jsou popsána jednotlivá okna, která aplikace obsahuje a jejich funkčnost. Popis podle, kterého byl vytvořen testovací scénář a návod jak převést soubory vytvořené v prostředí Pycharm na spustitelné.

5.2.1 Přihlášení

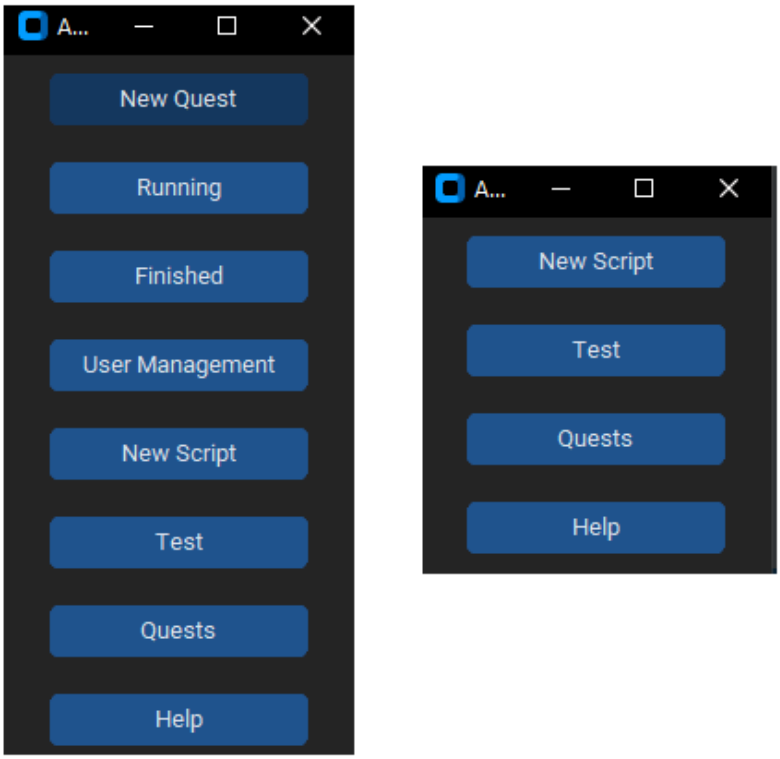
Po spuštění souboru **main.exe** je uživateli zobrazeno přihlašovací okno kde je nutné vyplnit přihlašovací údaje. Pro účely testování jsou vytvořeni dva uživatelé. **Admin**, který je členem týmu Management a člen obecného týmu **John**[1]. Přihlášení do aplikace lze provést kliknutím na tlačítko **Click To Log In!** nebo stisknutím klávesy **Enter**. V případě vložení nesprávných přihlašovacích údajů je uživatel upozorněn chybovou hláškou.

5.2.2 Hlavní okno aplikace

Po úspěšném přihlášení je zobrazeno hlavní okno aplikace, které slouží k navigaci mezi jednotlivými funkcemi. Podoba hlavního okna záleží na týmu, do kterého je uživatel přiřazen. Uživatelé v týmu Management mají více možností práce s aplikací než běžní členové. Podoba oken je zobrazena obrázku 5.1

Po stisknutí jednotlivých tlačítek jsou zobrazena další okna, ve kterých může uživatel provádět požadovanou činnost.

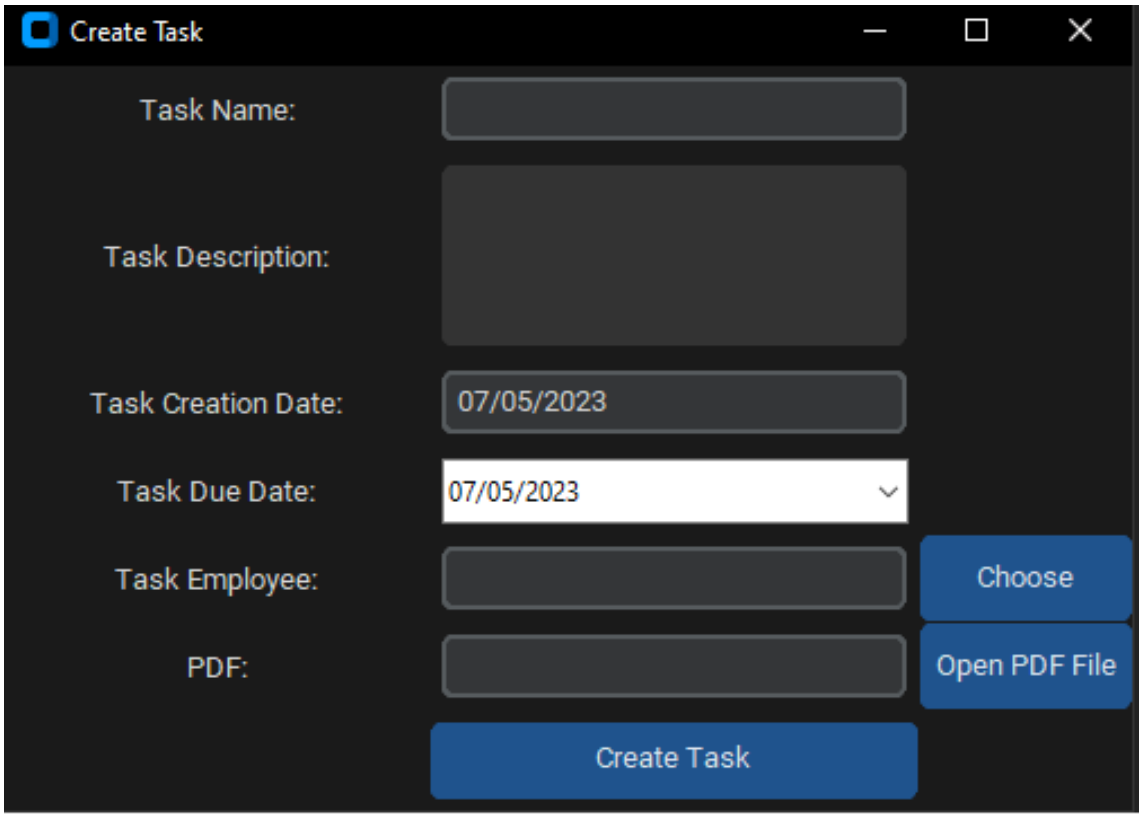
[1]Údaje pro přihlášení jsou součástí návodu, který lze nalézt v příloze této práce



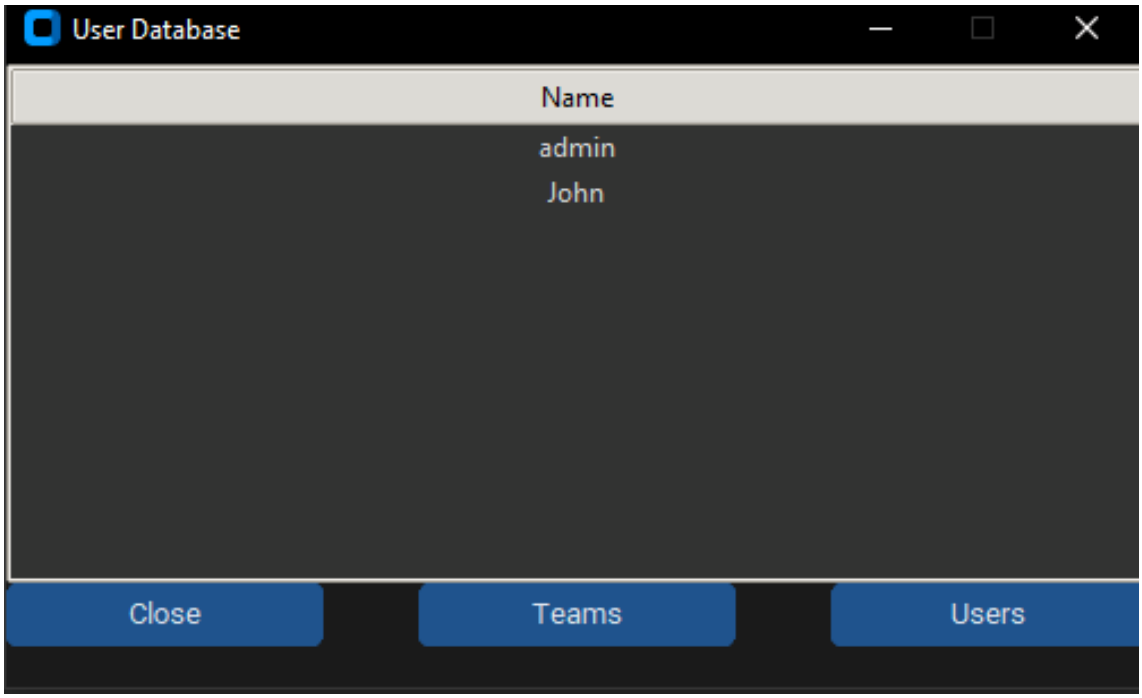
Obr. 5.1: Vlevo pro Management vpravo pro běžné uživatele

5.2.3 Vytvoření nového úkolu

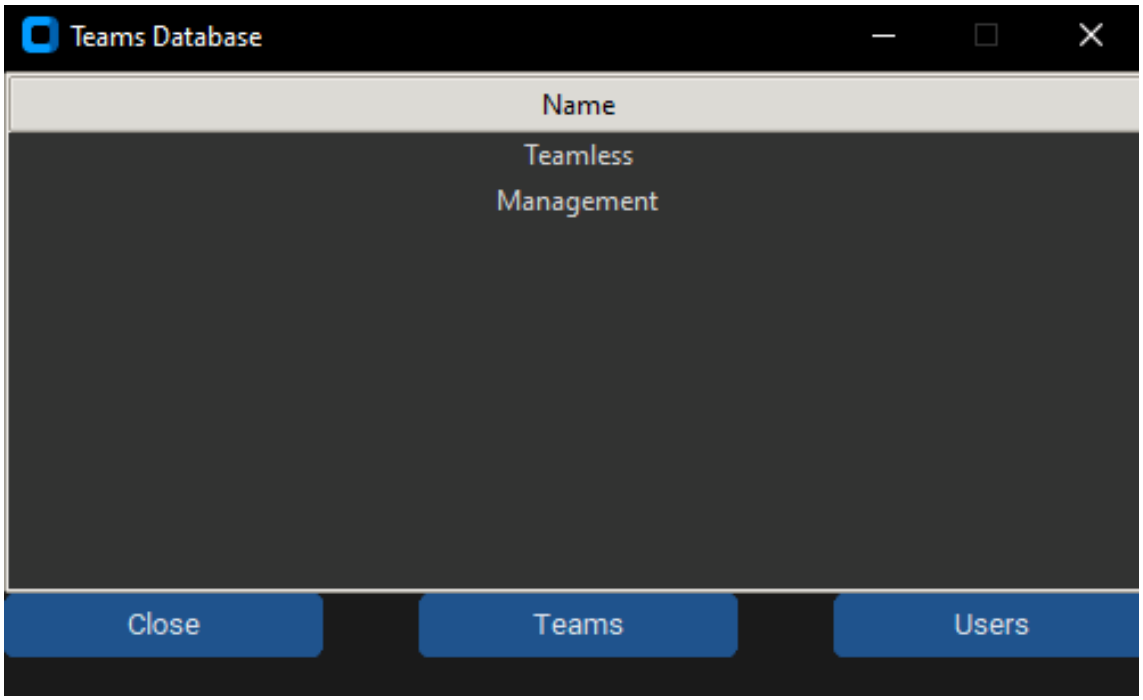
Po stisknutí tlačítka **New Quest** je zobrazeno okno 5.2, které slouží k vytvoření nového úkolu. Uživatel zadá jméno úkolu, popis úkolu, datum do kterého má být úkol dokončen a přiřadí k úkolu uživatele nebo celý tým, zároveň je možné vložit PDF dokument. Při vložení PDF souboru se předpokládá, že se jedná o smlouvu o dílo a uživatel je vyzván, aby zaškrtl paragrafy, které mohou být problematické a popsal, jak by se měl uživatel, který bude na úkolu pracovat chovat, aby nedošlo k jejich porušení. Smlouvu může uživatel kdykoliv zobrazit po zobrazení podrobností o úkolu. Úkol může být přiřazen uživateli zadáním jeho jména do kolonky **Employee** nebo kliknutím na tlačítko **Choose** kde dojde k zobrazení okna 5.3 s databází uživatelů. Dvojklikem na jméno uživatele dojde k přiřazení jména uživatele k úkolu. Nebo lze kliknout na tlačítko **Teams**, tím dojde k zobrazení všech týmů a dvojklikem na jméno týmu k přiřazení úkolu celému týmu. Podobu zobrazení všech týmu popisuje obrázek 5.4 Stiskem **Create Task** dojde k zápisu úkolu do databáze.



Obr. 5.2: Okno pro vytvoření nového úkolu



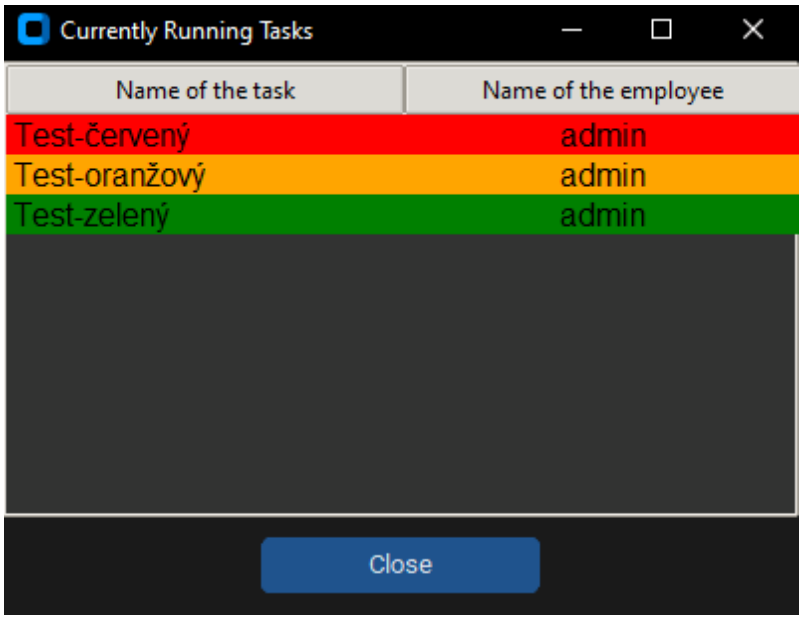
Obr. 5.3: Databáze uživatelů po kliknutí na tlačítko **Choose**



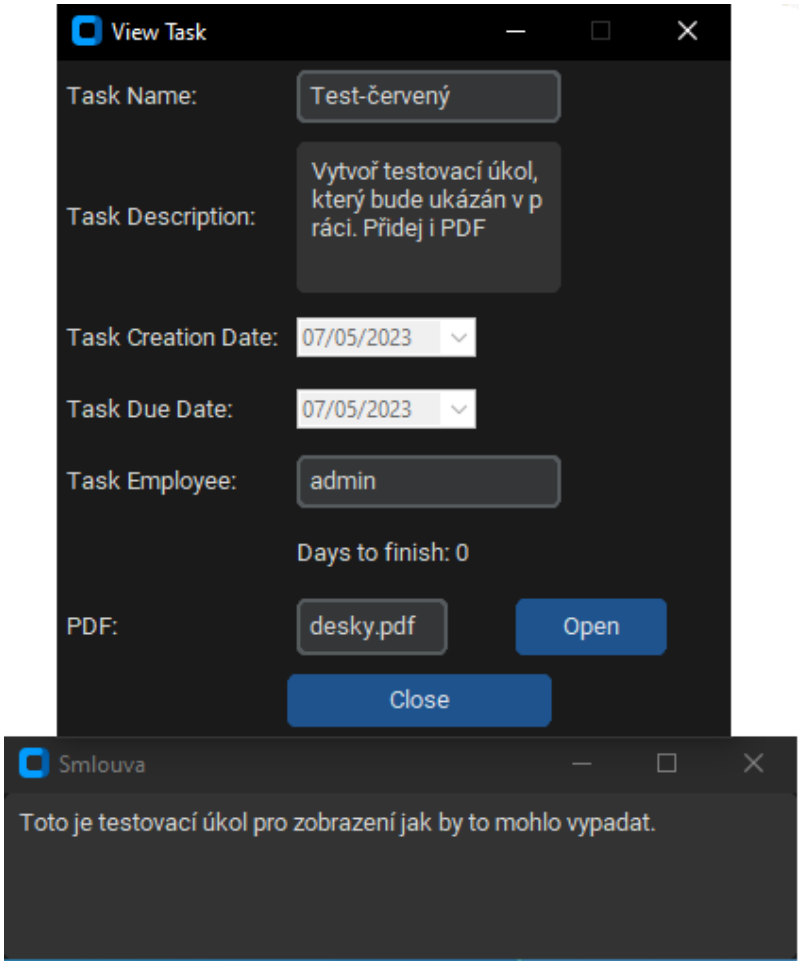
Obr. 5.4: Databáze týmu po kliknutí na tlačítko **Teams**

5.2.4 Probíhající úkoly

Stisknutím tlačítka **Running** dojde k zobrazení probíhajících úkolů a členů, kteří mají úkol na starost. Podobu tohoto okna zachycuje obrázek 5.5 Úkoly jsou seřazeny sestupně podle data dokončení úkolu a odlišeny barvou, která určuje kolik času zbývá do dokončení úkolu. Pokud zbývá méně než čtyři dny do dokončení úkolu bude zobrazen červenou barvou. Pokud zbývá více než čtyři, ale méně než sedm dní úkol bude zobrazen oranžovou barvou a pokud zbývá do dokončení více než týden úkol bude zobrazen zelenou barvou. Tabulka s probíhajícími úkoly je interaktivní a po dvojkliku na jméno úkolu se vytvoří nové okno s údaji, které byly zadány při vytváření úkolu. Toto okno popisuje obrázek 5.6 V případě, že byla vložena smlouva zobrazí se i okno obsahující informace, na co si má dát uživatel pozor. Při dvojkliku na jméno uživatele se zobrazí okno s úkoly, které má uživatel v dané chvíli přiřazené. O tomto oknu více pojednává kapitola 5.2.9.



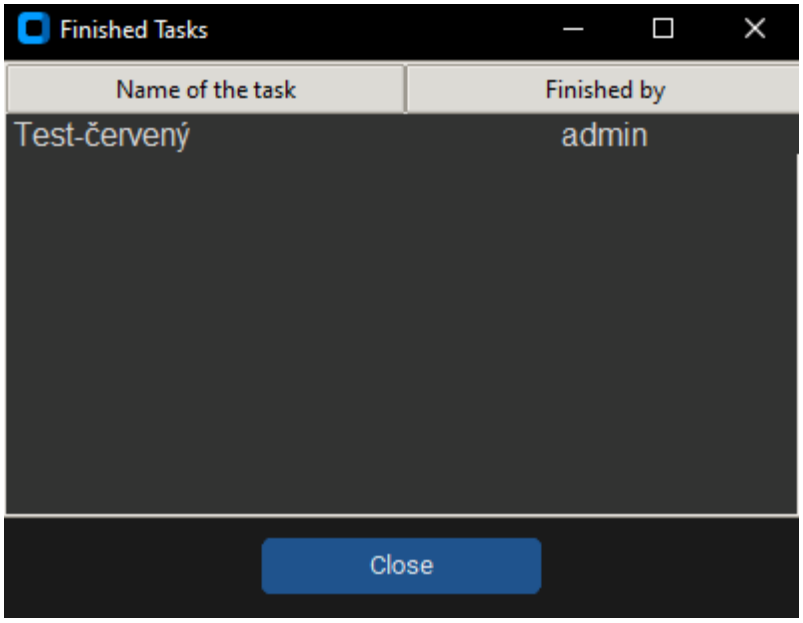
Obr. 5.5: Zobrazení právě probíhajících úkolů



Obr. 5.6: Zobrazení informací po kliknutí na úkol

5.2.5 Dokončené úkoly

Stisknutím tlačítka **Finished** jsou zobrazeny úkoly, které již byly dokončeny a uživatel, který byl k úkolu přiřazen. Toto okno je zobrazeno na obrázku 5.7 Dvojklikem na jméno úkolu dojde k jeho zobrazení a pokud byl předmětem úkolu útok, tak i informace o průběhu útoku, které byly uživatelem zadány po jeho dokončení. Zobrazení informací o úkolu a informacích zadaných uživatelem po dokončení úkolu lze vidět na obrázku 5.8

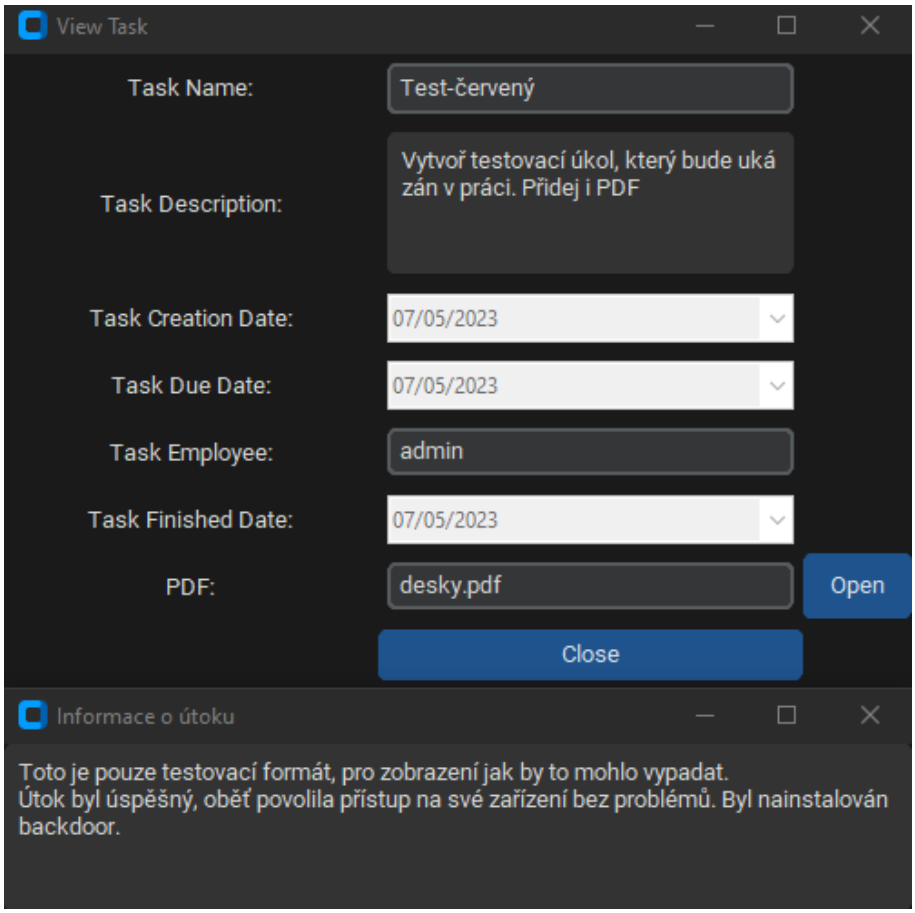


Obr. 5.7: Zobrazení dokončených úkolů

5.2.6 Správa uživatelů

Po kliknutí na tlačítko **User Management** se zobrazí okno určené pro správu uživatelů a týmů v aplikaci. Toto okno obsahuje dvě tabulky, z nichž jedna je interaktivní a druhá slouží k zobrazení výsledků. Dvojklikem na název týmu je tabulka přepsána a jsou zde vypsáni všichni uživatelé, které do daného týmu patří. Kliknutím na **Default** dojde k zobrazení všech uživatelů v databázi. Zároveň je možné vyhledávat uživatele podle jména. Možná podoba okna Management je na obrázku 5.9

V Tomto okně jsou implementovány funkcionality pro přidání a odebrání uživatelů, přidání uživatele do týmu, a vytvoření a smazání týmu. Pro přidání uživatele je nutné dodržet základní parametry jako je délka hesla a jména. Pro ostatní funkcionality není nutné dodržovat parametry, aplikace kontroluje zda uživatel nebo tým, který má být smazán nebo přidán již existuje a případně o tom uživatele informuje.



Obr. 5.8: Zobrazení informací o úkolu a útoku

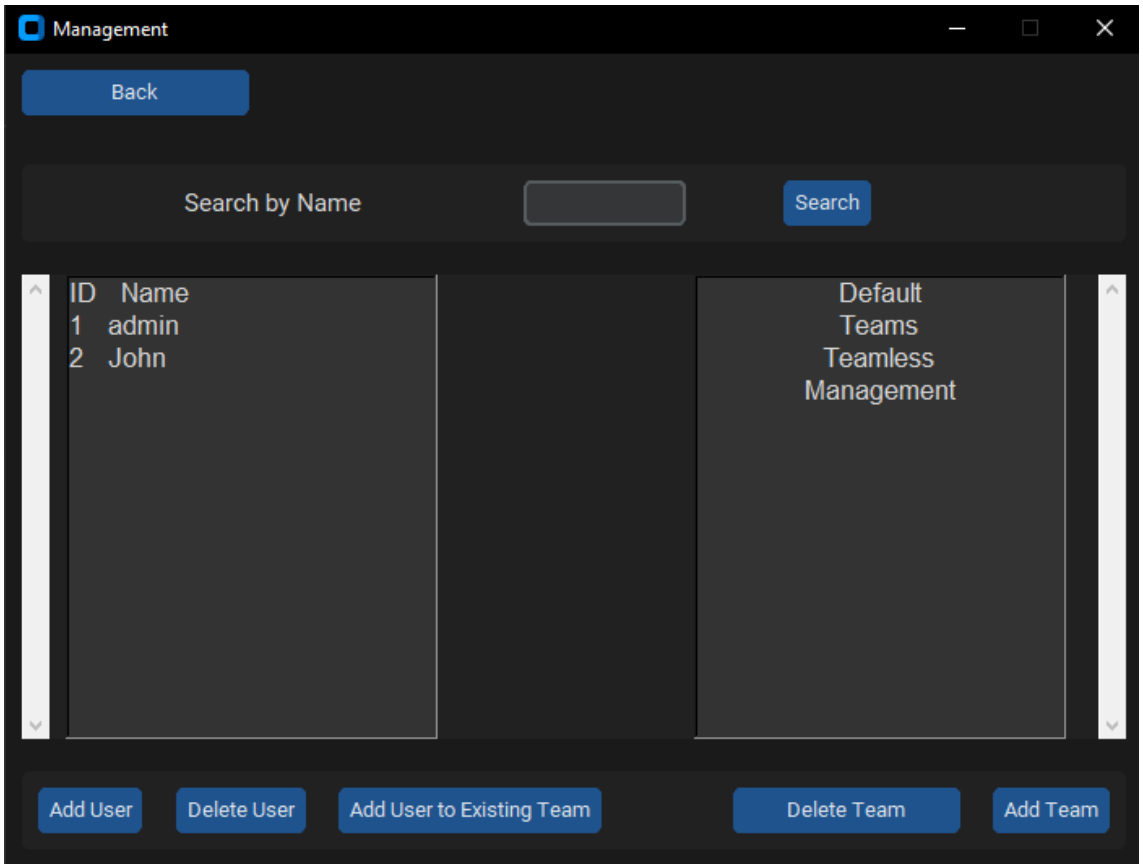
Zároveň je implementováno opatření proti smazání týmu Management a zajištění, že v týmu Management vždy bude alespoň jeden uživatel.

5.2.7 Vytvoření nového scénáře

Toto okno společně s oknem **Test** tvoří jádro této aplikace. Slouží k vytvoření nových scénářů pro útok a pro přiřazení obrázků k jednotlivým útokům. Pro vytvoření nového úkolu je nutné zadat jeho jméno, avšak toto jméno nesmí být **default** toto jméno je rezervováno pro účely vkládání obrázků, které jsou sdíleny všemi scénáři.

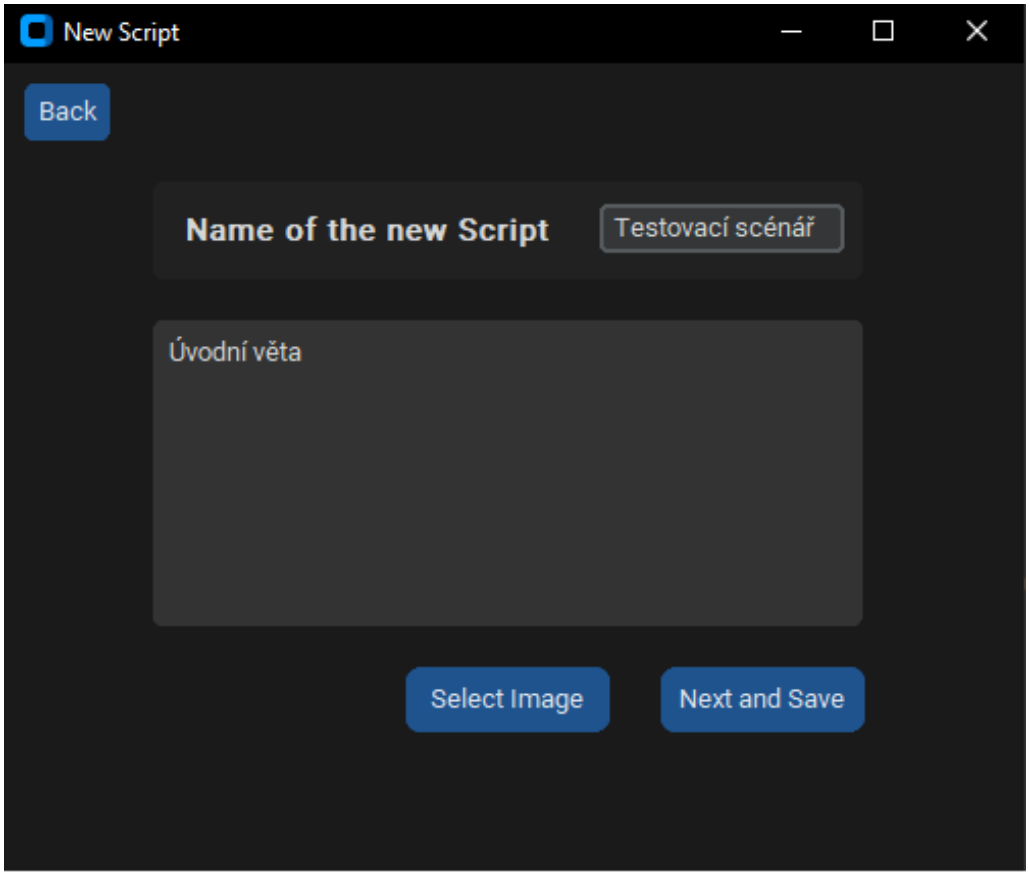
Při zadání jména **default** a stisknutí **Select Image** je vybraný obrázek vložen do základních obrázků, při pokusu vytvořit scénář s tímto jménem je uživatel upozorněn, že to není možné. Při zadání jiného jména lze již vytvářet scénář a vkládat obrázky, které budou použity specificky při tomto scénáři.

Vytváření samotného scénáře je založeno na stromové struktuře, kde kořenem stromu je úvodní věta. Po zadání úvodní věty a zmáčknutí tlačítka **Next and Save** je uživatel vyzván, aby zadal odpověď v případě negativní odpovědi na úvodní větu. Následně je uživatel po stisknutí tlačítka vyzván, aby zadal odpověď v případě

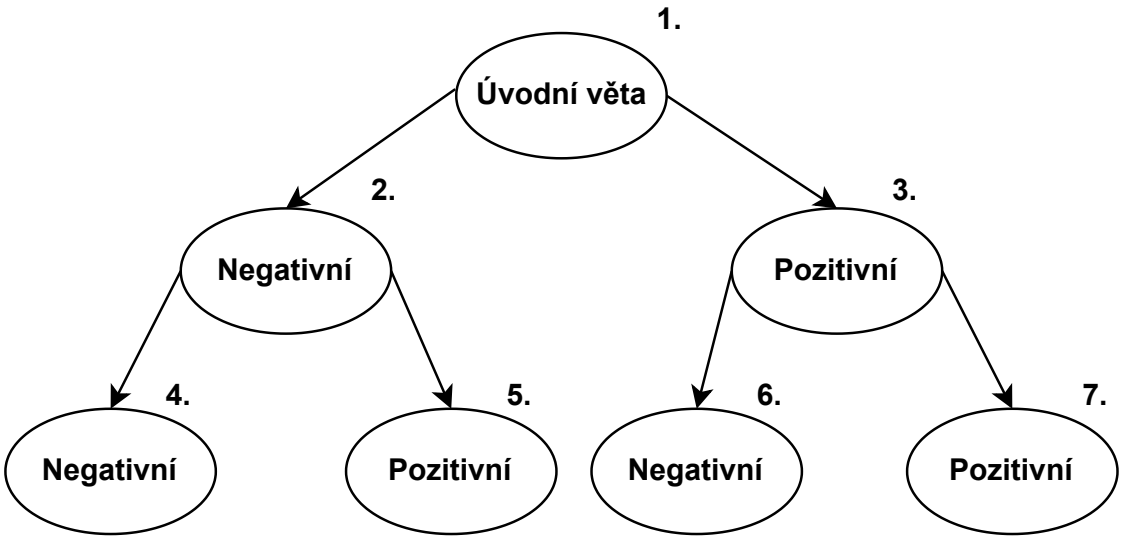


Obr. 5.9: Zobrazení informací o uživatelích a týmech

pozitivní odpovědi na úvodní větu. Poté je vyzván k zadání negativní odpovědi na negativní větu, která byla zadána při odpovědi na úvodní větu. Takovýmto způsobem se pokračuje až do doby než je scénář vytvořen. Systém vytváření scénáře lépe popisuje obrázek 5.11 kde je čísly označeno, v jakém pořadí jsou odpovědi zadávány.



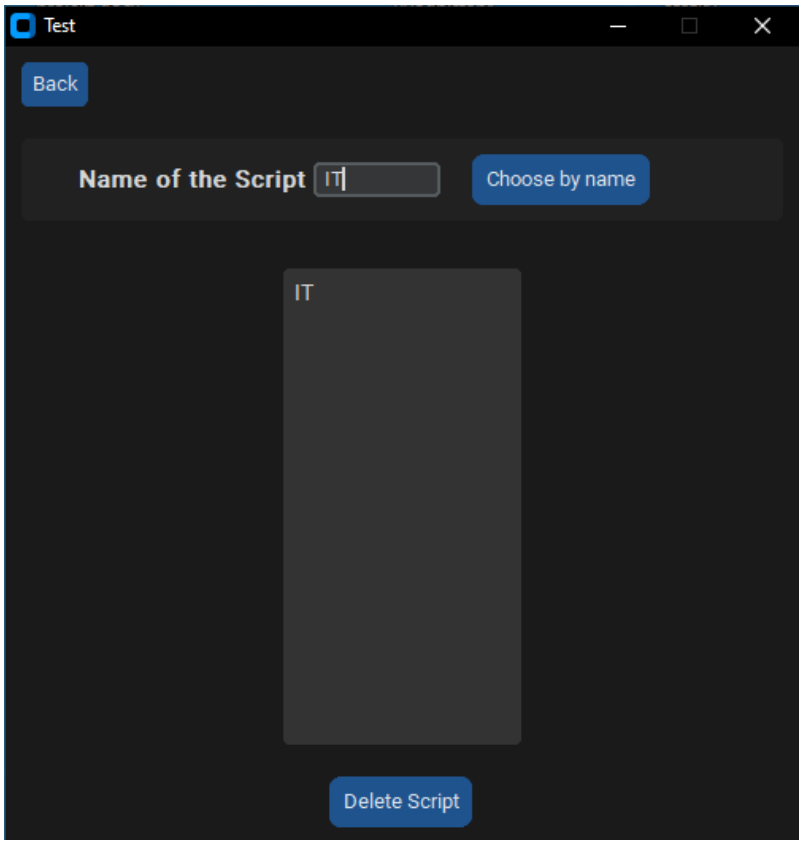
Obr. 5.10: Vytváření nového scénáře



Obr. 5.11: Systém vytváření scénáře

5.2.8 Testování scénáře

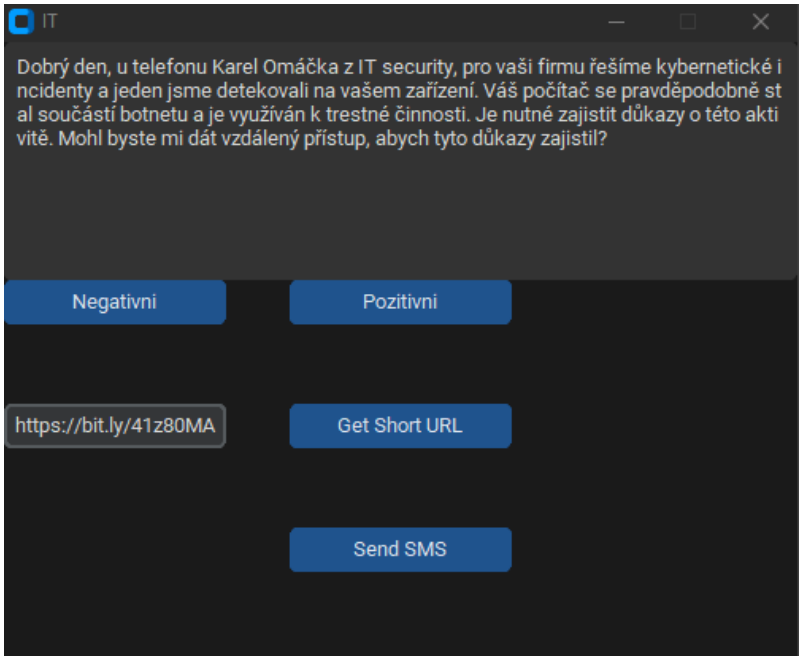
Po vytvoření scénáře, jak je popsáno v kapitole 5.2.7 a stisknutí tlačítka **Test** se zobrazí okno s tabulkou, která obsahuje existující scénáře. Toto okno je zobrazeno na obrázku 5.12 Vybrání scénáře je provedeno dvojklikem na jeho jméno v tabulce nebo zadáním příslušného jména a stisknutí tlačítka **Choose By Name**. Pro smazání scénáře je třeba zadat jeho jméno a stisknout tlačítko **Delete Script**.



Obr. 5.12: Okno pro výběr a smazání scénářů

Po výběru scénáře se zobrazí nové okno 5.13, které obsahuje úvodní větu pro daný scénář. V tomto okně jsou k dispozici tlačítka **Negative** a **Positive**, která slouží k načtení odpovědi v případě, že bude negativní nebo pozitivní. Dále je zde pole pro zadání URL adresy a tlačítko pro převedení této adresy na zkrácenou adresu. Například při zadání adresy **seznam.cz** bude výsledná adresa **https://bit.ly/41z80MA** změna na adresu v tomto formátu slouží k přesměrování oběti na podvodnou stránku, kterou si operátor předem připravil. Adresu uživatel oběti předá například pomocí SMS nebo ji může oběti nadiktovat. Po dokončení scénáře je uživatel vyzván k poskytnutí informací o proběhlém útoku, které jsou volitelné, ale doporučené k vyplnění. Uživatel může uvést, zda útok byl úspěšný a pokud ano, také popsat dosažený

výsledek. Tyto informace jsou následně přiřazeny k úkolu, na kterém uživatel pracoval. Je nutné zadat název úkolu, ke kterému mají tyto informace být přiřazeny.



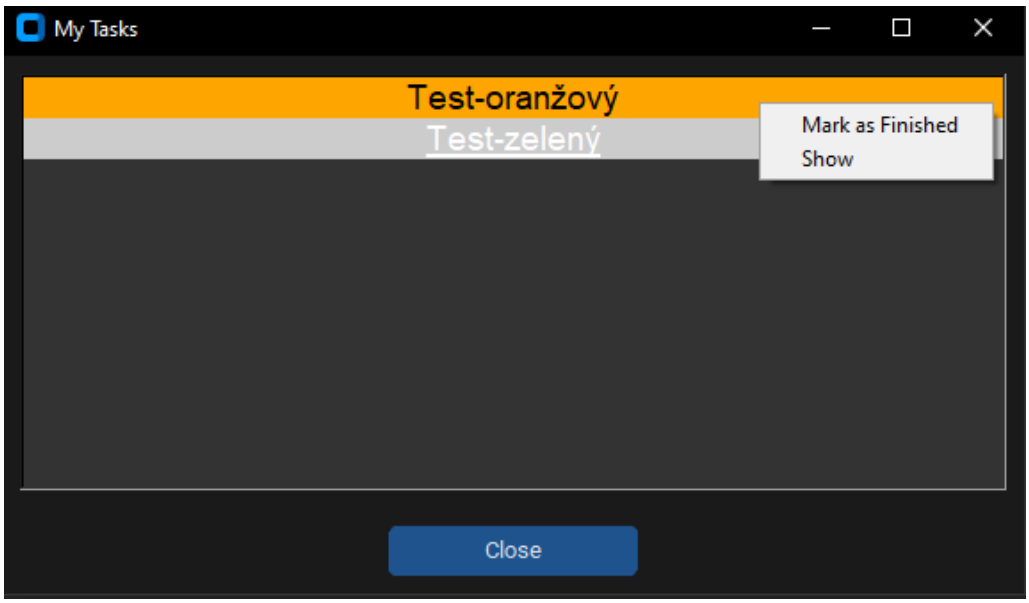
Obr. 5.13: Úvodní okno pro vybraný scénář

Posledním tlačítkem je **Send SMS**. Po jeho stisknutí se zobrazí okno, kde je nutné zadat telefonní číslo, na které bude SMS zaslána a text zprávy. Po stisknutí tlačítka **Send** bude zpráva odeslána a zobrazí se informační hláška, zda byla SMS úspěšně odeslána nebo nikoliv. Společně s oknem, ve kterém se nachází scénář je zobrazeno i okno s obrázky. Toto okno obsahuje základní obrázky pro všechny scénáře a obrázky specifické pro daný scénář. Dvojklikem na obrázek dojde k jeho zobrazení na obrazovce. To je využíváno při navigování oběti, na co má kliknout.Toto okno s obrázky může uživatel kdykoliv minimalizovat a opětovně rozbalit pomocí tlačítka v pravém horním rohu okna. Tím má uživatel větší kontrolu nad tím, co se mu na obrazovce zobrazuje a může si lépe upravit pracovní prostředí podle svých potřeb.

5.2.9 Úkoly uživatele

Kliknutím na tlačítko **Quests** jsou uživateli zobrazeny úkoly, které má v současné době přidělené a ještě nebyly označeny jako dokončené. Tabulka, ve které jsou úkoly zobrazeny je interaktivní a dvojklikem na jméno úkolu jsou zobrazeny jeho podrobnosti, stejně jako je popsáno na obrázku 5.6. Při kliknutí pravým tlačítkem myši na

jméno úkolu je zobrazena nabídka, ve které je možné zvolit **Show**, k zobrazení podrobností o úkolu, nebo **Mark as Finished**, tím dojde k přesunu úkolu do databáze dokončených úkolů.



Obr. 5.14: Aktivní úkoly

5.2.10 Pomoc

Posledním tlačítkem je **Help** toto tlačítko slouží k zobrazení návodu k ovládání aplikace. Stejně jak funguje toto tlačítko fungují i tlačítka při zobrazení podrobností o úkolu **Open**, kdy při kliknutí na ně dojde k zobrazení PDF souboru, který byl nahrán při vytváření úkolu.

5.2.11 Vytvoření testovacího scénáře

V aplikaci je dostupný testovací scénář s názvem **IT**. Tento scénář je zaměřen na simulaci útoku, kdy se útočník vydává za technickou podporu s cílem získání vzdáleného přístupu na zařízení oběti.

Jednotlivé kroky scénáře jsou:

- Operátor si o uživateli získá informace. Možný zisk informací popisuje kapitola 3.4.1
- Operátor volá uživateli a říká, že je z technické podpory společnosti XY, která pro společnost oběti řeší podporu při kybernetickém útoku
- Operátor informuje oběť, že z jeho počítače byla páchána trestná činnost a že je nutné okamžitě zajistit důkazy (apel na okamžitou reakci)

- Oběť se brání přístupu vzdáleného operátora
- Operátor informuje uživatele, že pokud tak neučiní, tak může být s celou věcí spojován a vyšetřován (vytvoření nátlaku)
- Oběť svolí k přístupu podvodníka do PC
- Operátor se ptá uživatele zda má na PC nainstalovaný TeamViewer nebo podobnou aplikaci pro správu zařízení
- Oběť odpovídá, že takovou aplikaci nemá / má
- Operátor buď požádá o otevření aplikace a nadiktování čísla pro TeamViewer nebo požádá uživatele o přístup na stránky AnyDesk
- Následně uživatel diktuje přístup (ID)
- Operátor se přihlásí a operuje dle svého postupu
  - Spustí nějaký sofistikovaný tool a informuje uživatele, že to bude nějakou dobu trvat
  - Začne vybírat přístupy a zajímavé soubory nebo rovnou spustí InfoStealer

Tyto kroky jsou všeobecné pro vysvětlení jak daný scénář funguje. Je totiž možné, že oběť nebude mít nainstalovaný software pro vzdálenou správu, a tak ho bude nutné nainstalovat. Vytváření scénáře pro vishing může být časově náročné a složité na uchopení, protože je nutné dopředu myslet na všechny situace, které mohou nastat. Cílem každého scénáře je přinutit oběť k vykonání určité akce, která je prospěšná pro útočníka.

Velmi sofistikovaný scénář, který se v nedávné době objevil v České republice je scénář s falešným bankéřem, kde se oběť stala součástí bankovního podvodu a je nutné řešit danou situaci zablokováním účtu a výběrem všech peněz z účtu společně se schválením před schválených úvěrů. Tento podvod velice pěkně vysvětluje **Jirka Burýšek** na platformě YouTube známý jako **Jirka vysvětluje věci** [35].

5.2.12 Převedení na spustitelný soubor

Tato podkapitola slouží k popisu jak převést soubory vytvořené v prostředí Pycharm na spustitelné soubory na operačním systému Windows bez nutnosti interpretu. Existuje několik způsobů, jak převést soubory vytvořené v PyCharmu na spustitelné soubory na Windows. Mezi nejčastější patří použití překladače **PyInstaller** nebo knihovny **cx Freeze**.

Při pokusu převést souboru za použití překladače **PyInstaller** nemohlo dojít dojít k převodu z důvodu využívání knihovny **CustomTkinter**, která obsahuje soubory, které tento překladač nedokáže převést. Proto byl zvolen překladač Nuitka, který převede všechny soubory do jazyka C a poté z nich vytvoří spustitelný soubor. Překladač Nuitka byl zvolen z důvodu rychlosti a snadnosti převodu. **Nuitka** není

součástí prostředí Pycharm a tak je nutné ji nainstalovat příkazem:

```
python -m pip install nuitka
```

Pro vytvoření spustitelných souborů na Windows je poté použít příkaz:

```
python -m nuitka --standalone --enable-plugin=tk-inter --follow-imports
main.py
```

Tím dojde k vytvoření složky **main.dist**, která obsahuje všechny potřebné soubory pro spuštění aplikace. Spuštěním souboru **main.exe** dojde ke spuštění aplikace. Jediné co je potřeba udělat, je vložení databází a obrázku, který je použit při vytváření logovacího okna do této složky. Nevýhodou je, že je možné spustit tento soubor pouze na Windows. Pro použití na operačním systému Linux, by bylo nutné převést soubory specificky pro tento operační systém a to za použití jiného překladače než Nuitka nebo spuštění Nuitky pro převod na systému Linux.

5.2.13 Bezpečnost Aplikace

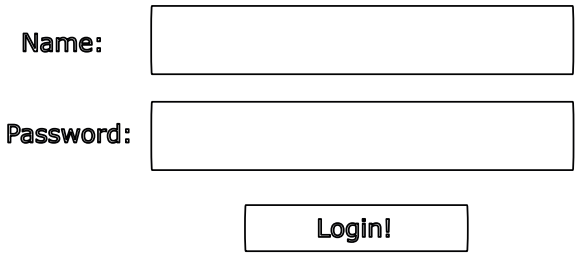
Tato podkapitola se zaměřuje na bezpečností opatření, která byla v aplikaci implementována k minimalizaci rizik spojených s útoky na bezpečnost. Podkapitola se zaměřuje na dva klíčové aspekty bezpečnosti aplikace, a to na SQL Injection a hashování hesel.

SQL Injection

SQL Injection je nejčastějším typem útoku na webové aplikace, konkrétně na přihlašovací formuláře na stránkách. Tento typ útoku se využívá k získání neoprávněného přístupu k databázi, kde poté může útočník číst, zapisovat nebo mazat data. Pro získání neoprávněného přístupu útočník vloží do SQL dotazu škodlivý kód a pokud je uživatelský vstup aplikace špatně implementován může dojít k nežádoucí reakci. V aplikaci, která byla vytvořena v praktické částí práce by mohlo být SQL Injection aplikováno při přihlašování uživatele, proto byl uživatelský vstup ošetřen proti tomuto typu útoku. Jak vypadá přihlašovací okno aplikace zobrazuje obrázek 5.15. Níže vložený příkaz je zranitelný na SQL Injection:

```
c.execute("SELECT * FROM members WHERE name='" + jméno + "' and
password ='" + heslo + "'")
```

Tento příkaz vybere všechny řádky z tabulky members, kde hodnota v sloupci name odpovídá proměnné jméno a hodnota proměnné password odpovídá hodnotě heslo. Tento příkaz je nebezpečný protože není nijak ošetřen uživatelský vstup a útočník, tak může vložit nežádoucí kód. Útočník by poté mohl vložit do kolonky Name například: **admin' OR '1'='1**. Příkaz pro čtení z databáze by poté vypadal následovně:



Obr. 5.15: Logovací okno

```
c.execute("SELECT * FROM members WHERE name='admin' OR '1'='1  and
password ='" + heslo + "'")
```

Vyhodnocování tohoto dotazu by probíhalo nejprve kontrolou zda existuje uživatel admin, který má heslo s danou hodnotou. Tento dotaz by vyl vyhodnocen jako **False**, protože útočník nezadal heslo, ale poté bude dotaz vyhodnocen jako **True**, protože existuje uživatel admin a 1=1 vždy. Takto by útočník mohl získat přístup do aplikace. Útočník nemusí znát uživatelské jméno a do kolonky name stačí vložit **' or 1=1**– a bude přihlášen do aplikace. Proti SQL Injection se lze bránit například použitím parametrizovaných příkazů jako:

```
c.execute("SELECT * FROM members WHERE name=? and password =?"
,(name,password))
```

Použitím parametrizovaných příkazů útočník nemůže vkládat škodlivý kód do příkazu. Další možností obrany je použití sanitizační funkce, tato funkce odfiltruje z uživatelského vstupu nebezpečné znaky jako uvozovky, středníky a apostrofy nebo odřádkování znaků v uživatelském vstupu. Při použití odřádkování jsou nebezpečné znaky nahrazeny lomítkem \, tím se zabrání vložení nebezpečných znaků do SQL dotazu [36] [37].

Hashování hesel

Hashování hesel je v aplikaci použito pro ukládání hesel v podobě znaků a čísel, tím je zajištěno, že hesla nejsou ukládána v otevřené formě. Kdyby byla hesla ukládána v otevřené formě, tak kdokoliv, kdo získá přístup k databázi, může zjistit hesla všech uživatelů. K tomu, aby byla hesla ukládána v bezpečné formě je proto použita hashovací funkce. Hashovací funkce je jednosměrnou funkcí, která pro stejný vstup vždy vygeneruje stejný hash, ale ze znalosti hashe nelze zpětně zjistit vstup. Pro zajištění větší bezpečnosti je přidávána kryptografická sůl což je náhodně vygenerovaná hodnota. Tím se zvýší bezpečnost proti útoku hrubou silou. V aplikaci je pro ukládání hesel použita hashovací funkce **Bcrypt** a kryptografická sůl [38].

V této kapitole byly popsány frameworky a knihovny, které byly použity při vývoji aplikace a popsány funkce jednotlivých oken aplikace k pochopení její funkčnosti. Byl představen koncept podle, kterého bude aplikace pomáhat zajišťovat dodržování platných právních předpisů, bylo popsáno jakým způsobem je možné převést soubory vytvořené v prostředí Pycharm na spustitelnou aplikaci a v neposlední řadě byla představena bezpečnost aplikace, která je velmi důležitá pro pro ochranu dat uživatelů. Bezpečnost se zaměřuje na SQL Injection a hashování hesel. Pro zabezpečení aplikace proti útokům je nutné používat parametrizované dotazy a sanitazovat uživatelský vstup. Tím dojde k zajištění toho, že aplikace nebude zranitelná na vstupy, které obsahují škodlivý kód. Hashováním hesel dojde k převedení hesel z otevřeného formátu na řetězec znaků a čísel a přidáním kryptografické soli k vytvoření silného hashe hesla, tím dojde k minimalizaci rizika úniku hesel. Důležité je také udržovat používat nejnovější knihovny a frameworky nebo je pravidelně aktualizovat.

Závěr

V teoretické části práce bylo vysvětleno, co je a čím se vyznačuje sociální inženýrství a popsány jeho nejvyužívanější techniky společně s uvedenými scénáři a možnostmi obrany proti těmto technikám. Byl vysvětlen princip autentizace a uvedeny možnosti, kterými je možné autentizaci provést. Důraz byl kladen na autentizaci znalostí, na kterou se sociální inženýři nejčastěji zaměřují.

Bylo popsáno, čím se vyznačuje Red Teaming a jeho rozdíly oproti penetračnímu testování. Byly popsány subjekty, které figurují při testování zabezpečení a aktivity, kterých je třeba dosáhnout. Byl popsán Cyber Kill Chain, který je při testování využíván, a jeho části se zaměřením na jeho prvotní část - Průzkum. Bylo popsáno zabezpečení objektu s liniemi, se kterými je možné se setkat při testování fyzické bezpečnosti a vysvětleno, proč je lepší využít metody sociálního inženýrství pro vnik do objektu.

V poslední kapitole teoretické části byly popsány zákony, které je možné při praktikování sociálního inženýrství porušit, uvedeny příklady jak je možné tyto zákony porušit a specifikovány nejvíce rizikové paragrafy týkající se techniky Vishing. Byla uvedena doporučení, které by měla firma zabývající se Red Teamingem aplikovat, aby minimalizovala rizika porušení těchto paragrafů. Mezi tyto doporučení patří školení zaměstnanců a stanovení vnitřních postupů firmy pro zaměstnance a možnost konzultace s odpovědnou osobou.

V praktické části diplomové práce, která se věnuje vývoji aplikace pro Vishing, byly popsány využívané frameworky, pomocí kterých byla aplikace vytvořena a popsána funkčnost jednotlivých oken této aplikace. Pro dosažení moderního vzhledu aplikace byla použita knihovna CustomTkinter společně s knihovnou Tkinter. Pro zkrácení URL adres byla použita online služba Bitly a její API a pro jednoduché odesílání SMS byla do aplikace integrována možnost využít Textbelt API. Tím bylo zajištěno, ža uživatele mohou používat tyto funkce v rámci aplikace. Bylo uvedeno, že aplikace bude pomáhat se zajištěním dodržování paragrafů možností zobrazit smlouvu o dílo v její původní podobě, společně s informacemi, které se smlouvy týkají a byly zadány při vytváření úkolu a je možné je zobrazit kdykoliv po zobrazení podrobností o úkolu. V posledních podkapitolách byly uvedeny kroky podle, kterých byl vytvořen testovací scénář a popsáno jak je možné převést soubory z prostředí Pycharm pomocí kompilátoru Nuitka na spustitelnou aplikaci na operačním systému Windows a jak je možné tento kompilátor nainstalovat do prostředí Pycharm. Poslední podkapitola se věnuje bezpečnosti aplikace se zaměřením na SQL Injection a hashování hesel, oba bezbečnostní prvky se zaměřují na přihlašovací okno aplikace. Ochrana proti SQL Injection byla implementována pomocí parametrizovaných dotazů a hashování hesel bylo realizováno knihovnou Bcrypt společně s použitím

kryptografické soli pro zvýšení bezpečnosti, tím bylo zajištěno, že hesla nebudou v databázi ukládána v otevřené formě, ale ve formě hashe.

Literatura

[1] *POLÁŠEK, Adam. Sociální inženýrství jako metoda vytěžování osob.* Zlín, 2018. Diplomová práce. Univerzita Tomáše Bati ve Zlíně. Vedoucí práce Ing. Dora Lapková.

[2] *What is Social Engineering?* [online]. WEEBROOT [cit. 2022-10-31]. Dostupné z: https://www.webroot.com/us/en/resources/tips-articles/what-is-social-engineering

[3] ŠIMEK, Richard. *Sociotechnika (sociální inženýrství)* [online]. Brno, 2003 [cit. 2022-10-31]. Dostupné z: https://www.fi.muni.cz/usr/jkucera/pv109/2003p/xsimek3sociotechnika.

[4] *What Is Social Engineering?* [online]. proofpoint [cit. 2022-10-31]. Dostupné z: https://www.proofpoint.com/us/threat-reference/social-engineering

[5] BURDA, Karel. *Elektronická kontrola vstupu.* VUT Brno.

[6] ČERMÁK, Miroslav. *Autentizace: řekni mi své heslo* [online]. 2009, 29.08.2011 [cit. 2022-10-31]. Dostupné z: https://www.cleverandsmart.cz/autentizace-neco-vi/

[7] GOODCHILD, Joan. *Social engineering techniques: 4 ways criminal outsiders get inside* [online]. 2010 [cit. 2022-10-31]. Dostupné z: https://www.csoonline.com/article/2125205/social-engineering-techniques–4-ways-criminal-outsiders-get-inside.html

[8] HEJDA, Daniel. *TECHNIKY SOCIÁLNÍHO INŽENÝRSTVÍ, NA ČEM JSOU ZALOŽENY A JAK JSOU VYUŽÍVÁNY PŘI INICIALIZAČNÍM VEKTORU.*

[9] *PRETEXTING LIKE A BOSS* [online]. TrustedSec: 2015 [cit. 2022-10-31]. Dostupné z: https://www.trustedsec.com/blog/pretexting-like-boss/

[10] *Caller ID Spoofing* [online]. 2022 [cit. 2022-10-31]. Dostupné z: https://www.fcc.gov/spoofing

[11] *What Is Pharming and How to Protect Yourself* [online]. [cit. 2022-10-31]. Dostupné z: https://usa.kaspersky.com/resource-center/definitions/pharming

[12] JANÍK, David. *Jak na kybernetickou bezpečnost: 1. díl Sociální Inženýrství* [online]. 10.09.2020 [cit. 2023-05-14]. Dostupné z: https://www.vas-hosting.cz/blog-jak-na-kybernetickou-bezpecnost-1-dil-socialni-inzenyrstvi#quipproquo

[13] Google Cloud Tech. *Google Data Center Security: 6 Layers Deep* [online]. 2021 [cit. 2023-05-15]. Dostupné z: https://www.youtube.com/watch?v=kd33UVZhnAA

[14] MIKULOVÁ, Petra a Barbora FUKAROVÁ. *Příběhy sociálního inženýrství [online].* [cit. 2022-10-31]. Dostupné z: https://security.muni.cz/socialni_inzenyrstvi

[15] WINER, Stuart. *'Dutch mole' planted Stuxnet virus in Iran nuclear site on behalf of CIA, Mossad* [online]. 2019 [cit. 2022-10-31]. Dostupné z: https://www.timesofisrael.com/dutch-mole-planted-infamous-stuxnet-virus-in-iran-nuclear-site-report/

[16] *Social Engineering and Malicious Code* [online]. [cit. 2022-10-31]. Dostupné z: https://studyrocket.co.uk/revision/gcse-computer-science-aqa/written-assessment/social-engineering-and-malicious-code

[17] *Scareware* [online]. [cit. 2022-10-31]. Dostupné z: https://www.fortinet.com/resources/cyberglossary/scareware

[18] *All about ransomware attacks* [online]. [cit. 2022-10-31]. Dostupné z: https://www.malwarebytes.com/ransomware

[19] https://images.techhive.com/images/article/2017/05/wannacry_ransom_screenshot-100722810-large.jpg

[20] ANTAL, Lukáš. *Red Teaming – Červená proti modré, aneb evoluce penetračních testů* [online]. 2019 [cit. 2022-11-20]. Dostupné z: https://hackinglab.cz/cs/blog/red-teaming-cervena-proti-modre-aneb-evoluce-penetracnich-testu/

[21] *WHAT IS THE CYBER KILL CHAIN? PROCESS & MODEL* [online]. 2022 [cit. 2022-11-20]. Dostupné z: https://www.crowdstrike.com/cybersecurity-101/cyber-kill-chain/

[22] ¨*Intelligence Gathering* [online]. 2014 [cit. 2022-11-20]. Dostupné z: http://www.pentest-standard.org/index.php/Intelligence_Gathering#Individual

[23] Zákon č. 89/2012 Sb. (Zákon občanský zákoník)

[24] GŘES, Petr. *Tajemství dopravovaných zpráv* [online]. Nový Jičín, 2011 [cit. 2023-05-15]. Dostupné z: https://www.policie.cz/clanek/tajemstvi-dopravovanych-zprav.aspx

[25] *Jednotlivé druhy kyberkriminality* [online]. [cit. 2022-11-19]. Dostupné z: https://www.policie.cz/clanek/jednotlive-druhy-kyberkriminality.aspx

[26] *Právní následky neoprávněného přístupu do účtu na sociální síti* [online]. c2023 [cit. 2023-01-14]. Dostupné z: https://muj-pravnik.cz/pravni-nasledky-neopravneneho-pristupu-do-uctu-na-socialni-siti/

[27] Zákon č. 40/2009 Sb. (Zákon trestní zákoník)

[28] *What is Python? Executive Summary* [online]. [cit. 2022-12-11]. Dostupné z: https://www.python.org/doc/essays/blurb/

[29] *What is PyCharm? Features, Advantages & Disadvantages* [online]. 2022 [cit. 2022-12-11]. Dostupné z: https://hackr.io/blog/what-is-pycharm

[30] *What Is SQLite?* [online]. 2022 [cit. 2022-12-11]. Dostupné z: https://www.sqlite.org/index.html

[31] SCHIMANSKY, Tom. CustomTkinter. *Official Documentation And Tutorial / CustomTkinter* [online]. c2023, 22.5.2022 [cit. 2023-05-07]. Dostupné z: https://customtkinter.tomschimansky.com/

[32] HAYEN, Kay. Nuitka the Python Compiler. *Nuitka the Python Compiler* [online]. c2023 [cit. 2023-05-07]. Dostupné z: https://nuitka.net/

[33] URL Shortener - Short URLs & Custom Free Link Shortener | Bitly. *Bitly* [online]. New York City, c2023 [cit. 2023-05-08]. Dostupné z: https://bitly.com/

[34] LLC, Alioth. Textbelt: *Send and receive SMS with a clean, simple API* [online]. [cit. 2023-05-08]. Dostupné z: https://textbelt.com/

[35] BURÝŠEK, Jiří. *Podvodníci, kteří se vydávají za Policii* [online]. 2022 [cit. 2023-05-15]. Dostupné z: https://www.youtube.com/watch?v=rB0x09uAyyE&t=1s

[36] SQL injection [online]. PortSwigger, c2023 [cit. 2023-05-14]. Dostupné z: https://portswigger.net/web-security/sql-injection

[37] *HOW TO PROTECT AGAINST SQL INJECTION ATTACKS.* Crowdstrike [online]. c2023, 10.09.2022 [cit. 2023-05-14]. Dostupné z: https://www.crowdstrike.com/cybersecurity-101/sql-injection/

[38] ŠTRÁFELDA, Jan. Hash [online]. [cit. 2023-05-14]. Dostupné z: https://www.strafelda.cz/hash

Seznam symbolů a zkratek

Šířka levého sloupce Seznamu symbolů a zkratek je určena šířkou parametru prostředí `acronym` (viz řádek 1 výpisu zdrojáku na str. 77)

KolikMista pouze ukázka vyhrazeného místa

SMS Služba krátkých textových zpráv – Short Message Service

ID Identifikace – Identification

IT Informační technologie – Information Technology

DNS Systém doménových jmen – Domain Name System

IP Internetový protokol – Internet Protocol

USB Univerzální sériová sběrnice – Universal Serial Bus

CD Kompaktní disk – Compact Disk

DDoS Distribuovaný DoS útok – Distributed Denial of Service

CKC Kybernetický řetězec útoků – Cyber Kill Chain

HTTP Protokol přenosu hypertextu – Hypertext Transfer Protocol

OSINT Otevřená zpravodajská služba – Open-source intelligence

IDE Integrované vývojové prostředí – Integrated Development Environment

GUI Grafické uživatelské rozhraní – Graphic User Interface

SQL Strukturovaný dotazovací jazyk – Structured Query Language

URL Jednotný lokátor zdroje – Uniform Resource Locator

API Rozhraní pro programování aplikací – Application Programming Interface

PDF Přenosný formát dokumentů – Portable Document Format

PC Osobní počítač – Personal computer

A Některé příkazy balíčku `thesis`

A.1 Příkazy pro sazbu veličin a jednotek

Tab. A.1: Přehled příkazů pro matematické prostředí

Příkaz	Příklad	Zdroj příkladu	Význam
`\textind{...}`	$\beta_{\max}$	`$\beta_\textind{max}$`	textový index
`\const{...}`	U_{in}	`$\const{U}_\textind{in}$`	konstantní veličina
`\var{...}`	u_{in}	`$\var{u}_\textind{in}$`	proměnná veličina
`\complex{...}`	$\underline{u}_{\mathrm{in}}$	`$\complex{u}_\textind{in}$`	komplexní veličina
`\vect{...}`	$\mathbf{y}$	`$\vect{y}$`	vektor
`\mat{...}`	$\mathbf{Z}$	`$\mat{Z}$`	matice
`\unit{...}`	kV	`$\unit{kV}$` či `\unit{kV}`	jednotka

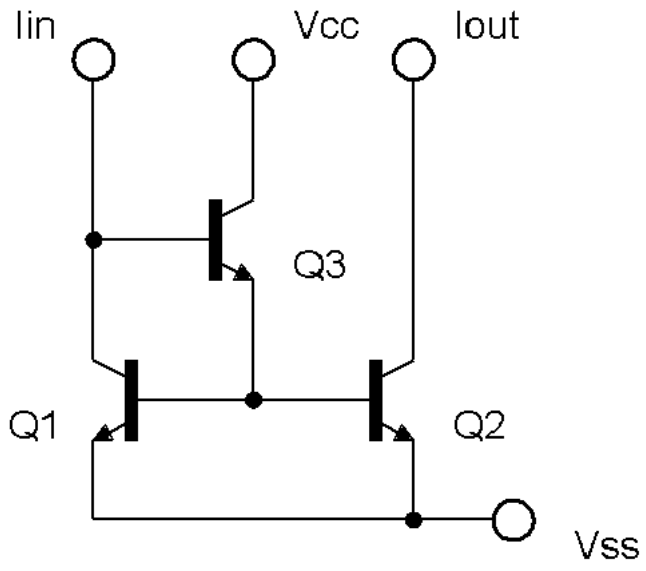
A.2 Příkazy pro sazbu symbolů

- `\E`, `\eul` – sazba Eulerova čísla: e,
- `\J`, `\jmag`, `\I`, `\imag` – sazba imaginární jednotky: j, i,
- `\dif` – sazba diferenciálu: d,
- `\sinc` – sazba funkce: sinc,
- `\mikro` – sazba symbolu mikro stojatým písmem[1]: µ,
- `\uppi` – sazba symbolu π (stojaté řecké pí, na rozdíl od `\pi`, což sází π).

Všechny symboly jsou určeny pro matematický mód, vyjma `\mikro`, jenž je použitelný rovněž v textovém módu.

[1] znak pochází z balíčku `textcomp`

B Druhá příloha



Obr. B.1: Zlepšené Wilsonovo proudové zrcadlo.

Pro sazbu vektorových obrázků přímo v LaTeXu je možné doporučit balíček `TikZ`. Příklady sazby je možné najít na TeXample. Pro vyzkoušení je možné použít programy QTikz nebo TikzEdt.

C Příklad sazby zdrojových kódů

C.1 Balíček `listings`

Pro vysázení zdrojových souborů je možné použít balíček `listings`. Balíček zavádí nové prostředí `lstlisting` pro sazbu zdrojových kódů, jako například:

```
\section{Balíček lstlistings}
Pro vysázení zdrojových souborů je možné použít
  balíček \href{https://www.ctan.org/pkg/listings}%
  {\texttt{listings}}.
Balíček zavádí nové prostředí \texttt{lstlisting} pro
  sazbu zdrojových kódů.
```

Podporuje množství programovacích jazyků. Kód k vysázení může být načítán přímo ze zdrojových souborů. Umožňuje vkládat čísla řádků nebo vypisovat jen vybrané úseky kódu. Např.:
Zkratky jsou sázeny v prostředí `acronym`:

```
6 \begin{acronym}[KolikMista]
```

Šířka textu volitelného parametru `KolikMista` udává šířku prvního sloupce se zkratkami. Proto by měla být zadávána nejdelší zkratka nebo symbol. Příklad definice zkratky **symfvz!** je na výpisu C.1.

Výpis C.1: Ukázka sazby zkratek

```
21   \acro{ID}            % název
22     {Identifikace -- Identification}          % popis
```

Ukončení seznamu je provedeno ukončením prostředí:

```
26     \acro{DNS}
27         {Systém doménových jmen -- Domain Name System}
28     \acro{IP}
29         {Internetový protokol -- Internet Protocol}
30     \acro{USB}
31         {Univerzální sériová sběrnice -- Universal Serial Bus}
32     \acro{CD}
33         {Kompaktní disk -- Compact Disk}
34     \acro{DDoS}
35         {Distribuovaný DoS útok -- Distributed Denial of Service}
36     \acro{CKC}
37         {Kybernetický řetězec útoků -- Cyber Kill Chain}
```

```
38      \acro{HTTP}
39          {Protokol přenosu hypertextu -- Hypertext Transfer Protocol}
40      \acro{OSINT}
41          {Otevřená zpravodajská služba -- Open-source intelligence}
42      \acro{IDE}
43          {Integrované vývojové prostředí -- Integrated Development Envir
44      \acro{GUI}
45          {Grafické uživatelské rozhraní -- Graphic User Interface}
46      \acro{SQL}
47          {Strukturovaný dotazovací jazyk -- Structured Query Language}
48      \acro{URL}
49          {Jednotný lokátor zdroje -- Uniform Resource Locator}
50      \acro{API}
51          {Rozhraní pro programování aplikací -- Application Programming
52      \acro{PDF}
53          {Přenosný formát dokumentů -- Portable Document Format}
54      \acro{PC}
55          {Osobní počítač -- Personal computer}
56
57
58
59
60
61
62
63
64  \end{acronym}
```

Poznámka k výpisům s použitím volby jazyka `czech` nebo `slovak`:
Pokud Váš zdrojový kód obsahuje znak spojovníku `-`, pak překlad může skončit chybou. Ta je způsobená tím, že znak `-` je v českém nebo slovenském nastavení balíčku `babel` tzv. aktivním znakem. Přepněte znak `-` na neaktivní příkazem `\shorthandoff{-}` těsně před výpisem a hned za ním jej vraťte na aktivní příkazem `\shorthandon{-}`. Podobně jako to je ukázáno ve zdrojovém kódu šablony.

Na výpisu C.2 naleznete příklad kódu pro Matlab, na výpisu C.3 zase pro jazyk C.

Výpis C.2: Příklad Schur-Cohnova testu stability v prostředí Matlab.

```
%% Priklad testovani stability filtru

% koeficienty polynomu ve jmenovateli
a = [ 5, 11.2, 5.44, -0.384, -2.3552, -1.2288];
disp( 'Polynom:'); disp(poly2str( a, 'z'))

disp('Kontrola pomoci korenu polynomu:');
zx = roots( a);
if( all( abs( zx) < 1))
    disp('System je stabilni')
else
    disp('System je nestabilni nebo na mezi stability');
end

disp(' '); disp('Kontrola pomoci Schur-Cohn:');
ma = zeros( length(a)-1,length(a));
ma(1,:) = a/a(1);
for( k = 1:length(a)-2)
    aa = ma(k,1:end-k+1);
    bb = fliplr( aa);
    ma(k+1,1:end-k+1) = (aa-aa(end)*bb)/(1-aa(end)^2);
end

if( all( abs( diag( ma.'))))
    disp('System je stabilni')
else
    disp('System je nestabilni nebo na mezi stability');
end
```

Výpis C.3: Příklad implementace první kanonické formy v jazyce C.

```
// první kanonická forma
short fxdf2t( short coef[][5], short sample)
{
  static int v1[SECTIONS] = {0,0},v2[SECTIONS] = {0,0};
  int x, y, accu;
  short k;

  x = sample;
  for( k = 0; k < SECTIONS; k++){
    accu = v1[k] >> 1;
    y = _sadd( accu, _smpy( coef[k][0], x));
    y = _sshl(y, 1) >> 16;

    accu = v2[k] >> 1;
    accu = _sadd( accu, _smpy( coef[k][1], x));
    accu = _sadd( accu, _smpy( coef[k][2], y));
    v1[k] = _sshl( accu, 1);

    accu = _smpy( coef[k][3], x);
    accu = _sadd( accu, _smpy( coef[k][4], y));
    v2[k] = _sshl( accu, 1);

    x = y;
  }
  return( y);
}
```

D Obsah elektronické přílohy

Elektronická příloha je často nedílnou součástí semestrální nebo závěrečné práce. Vkládá se do informačního systému VUT v Brně ve vhodném formátu (ZIP, PDF . . .).

Nezapomeňte uvést, co čtenář v této příloze najde. Je vhodné okomentovat obsah každého adresáře, specifikovat, který soubor obsahuje důležitá nastavení, který soubor je určen ke spuštění, uvést nastavení kompilátoru atd. Také je dobře napsat, v jaké verzi software byl kód testován (např. Matlab 2018b). Pokud bylo cílem práce vytvořit hardwarové zařízení, musí elektronická příloha obsahovat veškeré podklady pro výrobu (např. soubory s návrhem DPS v Eagle).

Pokud je souborů hodně a jsou organizovány ve více složkách, je možné pro výpis adresářové struktury použít balíček `dirtree`.

```
/ ..... kořenový adresář přiloženého archivu
├── logo ..... loga školy a fakulty
│   ├── BUT_abbreviation_color_PANTONE_EN.pdf
│   ├── BUT_color_PANTONE_EN.pdf
│   ├── FEEC_abbreviation_color_PANTONE_EN.pdf
│   ├── FEKT_zkratka_barevne_PANTONE_CZ.pdf
│   ├── UTKO_color_PANTONE_CZ.pdf
│   ├── UTKO_color_PANTONE_EN.pdf
│   ├── VUT_barevne_PANTONE_CZ.pdf
│   ├── VUT_symbol_barevne_PANTONE_CZ.pdf
│   └── VUT_zkratka_barevne_PANTONE_CZ.pdf
├── obrazky ..... ostatní obrázky
│   ├── soucastky.png
│   ├── spoje.png
│   ├── ZlepseneWilsonovoZrcadloNPN.png
│   └── ZlepseneWilsonovoZrcadloPNP.png
├── pdf ..... pdf stránky generované informačním systémem
│   ├── student-desky.pdf
│   ├── student-titulka.pdf
│   └── student-zadani.pdf
├── text ..... zdrojové textové soubory
│   ├── literatura.tex
│   ├── prilohy.tex
│   ├── reseni.tex
│   ├── uvod.tex
│   ├── vysledky.tex
│   ├── zaver.tex
│   └── zkratky.tex
├── sablona-obhaj.tex ..... hlavní soubor pro sazbu prezentace k obhajobě
├── sablona-prace.tex ..... hlavní soubor pro sazbu kvalifikační práce
└── thesis.sty ..... balíček pro sazbu kvalifikačních prací
```