UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2017 Matouš Kyncl

Univerzita Pardubice

Fakulta elektrotechniky a informatiky

Testování antivirových programů s využitím Kali a frameworků Veil a Metasploit

Matouš Kyncl

Bakalářská práce

2017

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: **2016/2017**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Matouš Kyncl**
Osobní číslo:	**I14126**
Studijní program:	**B2646 Informační technologie**
Studijní obor:	**Informační technologie**
Název tématu:	**Testování antivirových programů s využitím Kali a frameworků Veil a metasploit**
Zadávající katedra:	**Katedra informačních technologií**

Zásady pro vypracování:

Cílem práce je analyzovat a prakticky otestovat využití softwarových nástrojů Kali, framework Veil a metasploit.
Teoretické části provede rešerši zaměřenou na hodnocení antivirových free programů a analyzuje dnešní moderní typy virových nákaz. Dále představí softwarové nástroje Kali linux, framework Metasploit a framework Veil s důrazem na využitelnost těchto nástrojů, používaných k útokům na operační systémy.
V praktické části autor realizuje testovací síť s vybranými OS, ne kterých bude testovat různé antiviry proti zmíněným softwarovým nástrojům (metasploit, veil, kali). Na základě zjištěných výsledků provede komparativní analýzu chování a efektivity jednotlivých nástrojů a toho, jak jsou různé antivirové programy schopny odolávat zmíněným nástrojům.

Rozsah grafických prací:

Rozsah pracovní zprávy: **40 stran**

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

PRITCHETT, Willie L. a David DE SMET. Kali Linux cookbook. Birmingham: Packt Publishing, 2013. ISBN 978-1-78328-960-8.

ALLEN, Lee, Shakeel ALI a Tedi HERIYANTO. Kali Linux: assuring security by penetration testing. Birmingham: Packt Publishing, 2014. Community experience distilled. ISBN 978-1-84951-949-6.

Vedoucí bakalářské práce: **Mgr. Josef Horálek, Ph.D.**
Katedra informačních technologií

Datum zadání bakalářské práce: **31. října 2016**

Termín odevzdání bakalářské práce: **12. května 2017**

Univerzita Pardubice Česká republika

L.S.

Ing. Zdeněk Němec, Ph.D.
děkan

Mgr. Josef Horálek, Ph.D.
vedoucí katedry

V Pardubicích dne 31. března 2017

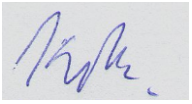
Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 27. 04. 2017



Matouš Kyncl

PODĚKOVÁNÍ

Na tomto místě bych chtěl poděkovat panu doktoru Josefu Janu Horálkovi za cenné rady při zpracování bakalářské práce, které mi poskytoval velice ochotně a rychle. Také bych chtěl poděkovat celé své rodině za pomoc s korekturou bakalářské práce.

ANOTACE

Tato práce se zabývá analýzou a testováním free antivirových programů s využitím systému Kali Linux a frameworků Metasploit a Veil. Cílem je ověřit možnost proniknutí do hostitelského PC a sledovat, zda antivirový program odhalí tento útok.

KLÍČOVÁ SLOVA

Kali, Linux, Metasploit, Veil, exploit, payload

TITLE

Testing antivir programs using Kali Linux and frameworks Metasploit and Veil

ANNOTATION

This work is focused on testing free antivir programs using Kali Linux and frameworks Metasploit and Veil. The aim is to penetrate into the host PC and monitor, if an antivir program will be respond to the unsolicited visit.

KEYWORDS

Kali, Linux, Metasploit, Veil, exploit, payload

OBSAH

SEZNAM ILUSTRACÍ A TABULEK

SEZNAM ZKRATEK A ZNAČEK

API	Application Programming Interface
CD	Compact Disc
CPU	Central processing unit
DLL	Dynamic-link library
DOS	Disk Operating Systém
DVD	Digital Versatile Disc
FHS	Filesystem Hierarchy Standard
GB	Gigabyte
GHz	Gigahertz
GPG	GNU Privacy Guard
GUI	Graphical User Interface
HDD	Hard Disk Drive
MB	Megabyte
MHz	Megahertz
MSF	Metasploit Framework
NOP	No-operation
OS	Operační systém
RAM	Random-access memory
RAT	Remote Administration Tool
UAC	User Access Control
URL	Uniform Resource Locator
USB	Universal Serial Bus
VNC	Virtual Network Computing

ÚVOD

Tato práce má za cíl otestovat schopnost zabezpečení jednotlivých antivirových programů, které jsou šířeny pod licencí free. K otestování jednotlivých antivirových programů bude použit OS Kali Linux. Ten má již v základní instalaci implementován framework Metasploit a pro potřeby testování bude doinstalovaný framework Veil.

V práci budou představena jednotlivá softwarová řešení, jež byla využita k otestování antivirových programů. Nejprve samotný OS Kali Linux a poté oba použité frameworky Metasploit a Veil. Následně v praktické části práce budou představeny jednotlivé platformy, jednotlivé způsoby, resp. jednotlivé payloady, se kterými se exploitace zkoušela a dále také výběr testovaných antivirových programů. Samotná testování budou doprovázena ukázkami v podobě screenshotů z testování. Na závěr budou představeny výsledky, jak si který antivirový program poradil s bezpečnostními riziky, a ty budou za využití metod popisné statistiky vyhodnoceny.

Obecně lze považovat exploitaci za velmi vhodný nástroj pro kontrolu zabezpečení jednotlivých OS, nebo také aplikací, například uvnitř nějaké firmy. Díky tomu se dá snáze předcházet bezpečnostním problémům a aplikovat vhodné bezpečnostní prvky, kterými mohou být právě také vhodně zvolené antivirové programy.

Přibližný postup testování vypadá asi takto. Pomocí frameworku Veil je vygenerován tzv. payload – neboli data, odkazující na část škodlivého softwaru, jenž vykonává nebezpečnou činnost. Tento payload je nasazen na systém, na který se útočí a pomocí frameworku Metasploit dojde k exploitaci tohoto systému.

1 AKTUÁLNÍ SITUACE

Problematikou testování antivirových programů s využitím frameworků Metasploit a Veil se doposavad (leden 2017) zabýval článek [1] z periodika News of Altai State University. Je zde testován pouze jediný komerční antivirový program, a to konkrétně Eset Smart Security 4, se systémem Windows 7. S použitím frameworku Veil (používaný pro vytváření takových virů, resp. payloadů, aby byly co nejhůře odhalitelné antivirovými programy) byl pro penetrační testování vytvořen payload python/meterpreter/rev_tcp. Tento byl nasazen do cílového systému a následně byl navázán kontakt s využitím frameworku Metasploit. Dle autorů, Eset Smart Security na vytvořený vir a přítomnost útočníka nijak nezareagoval, a proto doporučují používat nejen antivirový program, ale také například utilitu Antimeter2, což je program, který zjišťuje, zda nebyla v systému vytvořena instance programu Meterpreter (bude vysvětleno v kapitole 4.3.1) a pokud je zjištěno že ano, tak je okamžitě ukončena. Poté co se podařilo proniknout do systému, byla další překážkou UAC (User Access Control) kontrola, kterou je dobré mít z hlediska bezpečnosti neustále zapnutou, i když jde také obejít, jak je v článku ukázáno. Po prolomení UAC si může útočník v systému dělat víceméně cokoliv. Proto autoři v článku [1] vyslovují myšlenku, že v dnešní době obrovského rozmachu a vývoje informačních technologií se stává stále obtížnějším uchránit kterýkoli počítačový systém od různých druhů síťových útoků, a to i když je instalován komerční antivirový program.

Otázkou tedy je, jak si povedou a budou schopni odhalit hrozbu freeware verze antivirových programů po útoku s oběma frameworky?

2 VÝBĚR TESTOVANÝCH ANTIVIRŮ

2.1 Antivirové programy

2.1.1 Co je to antivirový program

Antivirový program (antivirus) je sada několika nástrojů, které hledají, odhalují a následně mažou viry, resp. nebezpečný malware [2]. V dnešní době jsou moderní antiviry vybaveny ochranou webových prohlížečů, ochranou před: ransomware, skenováním klávesnice, adware, spyware, rootkit, červy, trojskými koňmi a dalšími ochranami. Nejsou tedy podle názvu pouze proti-virovou ochranou, ale obecně ochranou před malware. [3]

Mohou existovat tyto druhy antivirových programů [5]:

- **On-demand skenery** – spouštějí se přes prostředí OS DOS v případě, kdy systém není schopen nastartovat běžným způsobem.
- **Jednoúčelové antiviry** – zaměřeny na detekci jednoho konkrétního viru.
- **Antivirové systémy** – komplexní systémy, které by měly ochránit daný systém před nežádoucími a nebezpečnými jevy.

V této práci budou využity a popisovány právě některé antivirové systémy.

2.1.2 Licence

Pokud vezmeme v potaz komplexní řešení antivirových systémů, tak poté takovéto systémy mohou existovat ve 3 (nebo 4) licenčních provedeních:

- **Freeware** – celý softwarový balíček (bez přidaných funkcionalit, které jsou dostupné v nejlepší verzi daného výrobce) je zcela zdarma, uživatel si ho stáhne a může používat neomezeně, jak dlouho chce, většinou i s pravidelnými aktualizacemi.
- **Trial** – licence zdarma, která je obvykle omezená na nějaký časový úsek (povětšinou 30 dnů) a nabízí základní funkce, obdobně jako freeware, ale přidává ještě některé další – ne však všechny funkcionality dostupné v nejlepší verzi. Uživatel si stáhne pouze testovací, zkušební balíček, který po daném časovém intervalu nebude moci dále využívat.
- **Shareware** – víceméně stejný princip jako trial verze, ovšem s tím rozdílem, že uživateli jsou poskytnuty k vyzkoušení veškeré funkcionality dostupné v nejlepší verzi antiviru. A po daném časovém intervalu si uživatel může pouze dokoupit licenci (sériové, registrační číslo), zaregistrovat svůj produkt a nadále ho používat již v placené verzi.

- **Placená verze** – uživatel si od společnosti kupuje licenci na používání daného produktu. U většiny společností platí, že licence se platí na rok, čili uživatel musí každoročně zaplatit poplatek, aby mohl nadále produkt využívat.

Obrázek (Obrázek 1) ilustruje freeware, trial a shareware licence poskytované společností Avast. První sloupec představuje licenci freeware, druhý licenci trial a poslední sloupec shareware verzi.

Obrázek 1 – Nabídka licencí Avast (zdroj [4])

2.1.3 Jak antivir funguje

Prvním důležitým předpokladem k zajištění ochrany je mít antivir neustále zapnutý, aby mohl ochránit systém. Druhým důležitým předpokladem je udržovat antivirový software neustále aktuální, resp. jeho virovou (malware) databázi. Tato databáze může být aktualizována dvěma způsoby: automaticky a manuálně. Pokud je aktualizována automaticky, aktualizace jsou uživateli dodávány periodicky (např. jednou za den), nebo okamžitě jakmile se objeví nový záznam v databázi. První způsob periodických aktualizací je v dnešní době běžnější, nicméně pokud je vybrána tato varianta, výrobci antivirových programů informují samotného uživatele, např. prostřednictvím emailu, o nejnovějších hrozbách. O automatické aktualizace se stará samotný antivirový program, který vše řeší za uživatele. Automatické aktualizace nemusí být vždy úplně ideálním řešením, protože i antivirový program je obyčejný software, který může obsahovat chyby a těchto chyb mohou využít útočníci. Zvláště pak velmi oblíbeným obdobím útoků je čas, kdy probíhá aktualizace a přenáší se nové informace do virové databáze, protože tyto informace mohou být zneužity a tím se může stát antivirus nebezpečným pro uživatele. Možná i proto se uživatel může automatickým aktualizacím bránit a řešit si vše po svém, manuálně. [5]

Každý antivirový program vyhledává, skenuje soubory nacházející se v systému a o toto vyhledávání se stará tzv. scanner, který je součástí antiviru. Vyhledat malware je možné těmito způsoby [5]:

- **Na vyžádání (on-demand)** – scanner začne pracovat až po požadavku uživatele. Většinou k tomuto požadavku uživatel přistoupí v případě, kdy chce ověřit podezřelý soubor, protože by mohlo hrozit nebezpečí. Antivir má na výběr několik možností (viz níže), jak bude toto nebezpečí analyzovat.
- **Po přístupu (on-access) –** scanner pracuje neustále, automaticky. Testuje každý nový soubor přicházející do systému. Díky neustálému provozu scanneru dochází ke zvýšení vstupně-výstupních operací a tím i zvýšení režijních nákladů pro běh systému. Některé antiviry proto nabízejí optimalizační řešení, kdy lze vybrat, aby soubory byly kontrolovány pouze při zápisu, resp. čtení.

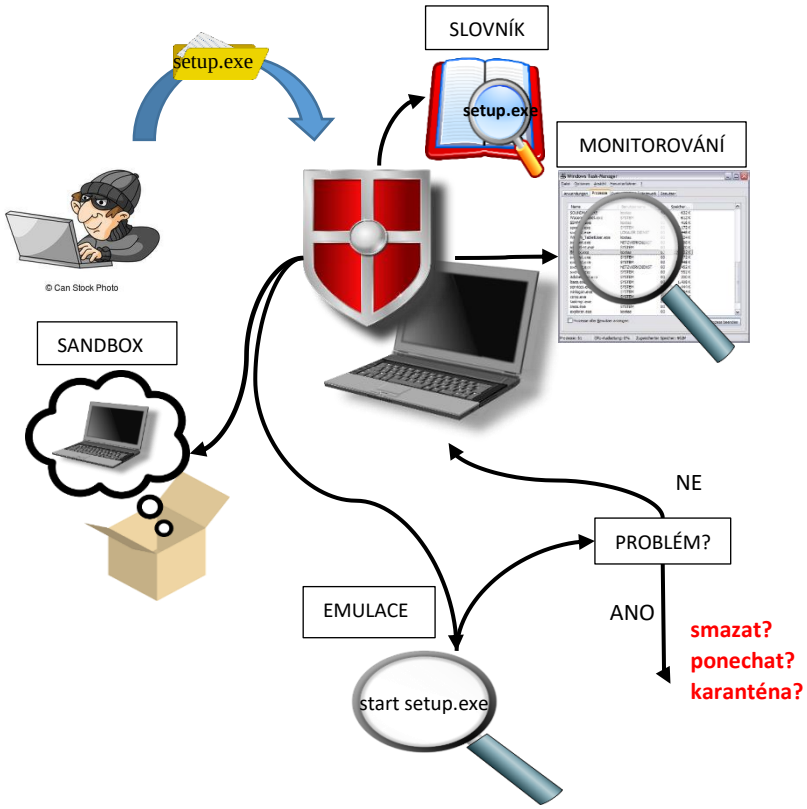
Způsoby, jak antivir odhaluje malware, jsou následující [6]:

- **Slovníkový přístup** – antivirus prohledává soubory a porovnává je se svojí malware databází. Po nalezení shody kódu programu s malware nacházejícím se v databázi, může dojít k těmto akcím: antivir smaže soubor (program), umístí soubor do

karantény, nebo se pokusí odstranit z programu část škodlivého kódu. Efektivní využití slovníkového přístupu je závislé především na aktualizované databázi malware.

- **Monitorování podezřelého chování** – antivirus nehledá malware, ale kontroluje chování spuštěných programů. V případě, kdy se program pokusí zapsat část kódu do spustitelného souboru, je toto chování označeno za podezřelé. Uživatel je proto vyzván antivirovým programem, aby určil, co se má s podezřelým programem dále stát. Výhodou této metody může být identifikování dosud neznámého malware, na druhou stranu, v dnešní době, kdy spousta programů zasahuje do spustitelných souborů, se stává metodou, která spíše obtěžuje uživatele a ten automaticky v dialogu odsouhlasí, že se nejedná o nic nebezpečného. Proto moderní antivirové systémy tuto metodu vyřazují ze svých řešení.
- **Emulování spustitelných souborů** – emulací dosáhne antivir toho, že malware je spuštěn před spuštěním samotného programu. V případě, že se kód programu sám od sebe mění anebo se chová jako malware, dochází antivir k názoru, že se jedná o nebezpečný software.
- **Odhalení v sandboxu** – sandbox je virtuální prostředí (dochází v něm ke spuštění emulovaného OS), které slouží k oddělení od reálného prostředí (OS). Při podezření na hrozící nebezpečí může uživatel spustit program v sandboxu a následně jsou zaznamenány změny emulovaného systému, které by mohly signalizovat přítomnost malware. Tato metoda je vysoce výkonově náročná, a proto k ní dochází jen na vyžádání uživatele.

Možnosti odhalení malware ilustruje obrázek (Obrázek 2).



Obrázek 2 – Způsoby odhalení malware (zdroj – vlastní)

2.2 Cloudová řešení

V dnešní době se stává velkým trendem provozovat antivirový systém v cloudu. I díky tomu, že se mnoho běžného softwaru přesouvá do cloudu. Čím méně se instalují desktopové aplikace, tím je menší potřeba mít desktopový antivirus. Výhodou používání antivirových systémů v cloudu může být možnost spojení více antivirových jader dohromady a tím lepší odhalení malware. Dále pak ochrana jak firemních sítí, tak domácích sítí bez vytížení výkonu vlastních počítačů. Ovšem neznámý malware nemusí odhalit ani větší množství antivirů. Spojení antivirů do cloudu se může označovat jako CloudAV. [7]

Prvním předpokladem pro správné fungování cloudových antivirů je připojení k internetu. V případě, kdy je uživatel offline, se informace a výsledky ukládají do cache paměti, která se nachází na straně uživatele, takže např. některé zkontrolované soubory nemusejí být kontrolovány znova. V případě, že je uživatel připojen k internetu, může být jeho systém kontrolován cloudovým antivirem. Nabízí se otázka, jakým způsobem provádí takovýto antivir kontrolu? Odpověď je jednoduchá. Antivir přijímá od uživatele informace o souborech

(nepřijímá celé soubory, ty by zahltily linku spojující uživatele s internetem), konkrétně jejich metadata.

Každý koncový bod (uživatel) má svého agenta, který shromažďuje informace a analyzuje soubory na systému. Antivirus v cloudu si tyto informace od agenta neustálé sbírá a pokud agent zaregistruje soubor, který se jeví jako škodlivý, antivir v cloudu na tuto skutečnost zareaguje a změní pohled vnímání na tento soubor. Díky této analogii, kdy rozhodování o tom, zda se jedná o malware či nikoliv je předáno na stranu serveru, není útočníkům přímo přístupný rozhodovací mechanismus samotného antiviru a činí pro ně větší úsilí vytvořit nový malware.

Řešení cloudových antivirů zahrnují stejné techniky detekce malware jako tradiční antiviry. Stejně jako desktopové antiviry, zahrnuje cloudové řešení detekci malware pomocí behaviorálních a heuristických analýz. Díky behaviorálním analýzám lze na koncových zařízeních odhalit: malware modifikující soubory, malware vytvářející autorun.inf na přenosných discích nebo sdílející autorun.inf po síti, malware zasílající několik emailů v krátkém časovém intervalu, malware generující nové spustitelné programy, malware modifikující auto-run registrační klíče. Pokud dosáhnou tyto malware aktivity určitého množství, agent na koncovém bodě může takový nebezpečný program zablokovat a oznámit tuto skutečnost do cloudu. Díky tomu mohou z takovéto skutečnosti těžit ostatní uživatelé, kteří jsou připojeni do stejného cloudu.

Antivirus v cloudu je pro dnešní dobu slibné řešení, a to i z důvodu toho, že většina tradičních výrobců antivirů tento způsob již podporuje ve snaze držet krok s všemožným malware, který se objevuje ve světě. [8]

2.3 AVAST

2.3.1 Charakteristika antiviru

Oblíbený antivirus od českých tvůrců, zástupce cloudových řešení, který zajišťuje ve spojení komerčních a nekomerčních řešení ochranu pro téměř 400 miliónů uživatelů. Antivirus Avast v neplacené verzi nabízí ochranu proti virům, malware, spyware, rootkitům a dalším problémům ohrožující například domácí síť. Analyzovat a zachytit hrozbu dokáže v reálném čase. Od roku 2016 je nejen v neplacené verzi k dispozici správce hesel – Hesla Avast. Umožňuje nahradit funkci automatických vyplňování formulářů ve webových prohlížečích mnohem bezpečnějším způsobem, především díky propracovanějšímu šifrování jednotlivých hesel. Zároveň lze do tohoto správce hesla uložit a na všechny oblíbené stránky se dostat jedním jediným heslem tzv. Hlavním heslem. V neplacené verzi lze také získat SafeZone, což je

bezpečný webový prohlížeč, který neobsahuje nežádoucí rozšíření a neobtěžuje manipulací výsledků vyhledávání hackery, za účelem zisku peněz. [9]

2.3.2 Systémové požadavky + funkce placených verzí

Minimální systémové požadavky pro antivir Avast jsou:

- OS: Windows 10, 8.1, 8, 7, Vista nebo XP SP
- CPU: neuvedeno
- RAM: 256 MB+
- HDD: 1,5 GB volného místa

Výrobce podporuje i OS Mac OS X a mobilní zařízení se systémem Android.

Placené verze oproti verzi free nabízejí např.: vlastní firewall, inspektor Wi-Fi sítí, několikanásobný přepis dat pro snížení rizika jejich obnovení – tzv. Skartovač dat, bezpečné DNS a další. [10]

2.4 AVG

2.4.1 Charakteristika antiviru

Další antivirový program vytvořený českými tvůrci (od minulého roku součást společnosti Avast, kterou bylo AVG koupeno). Dle tvůrců ochrana tímto antivirem nezklame. Nabízí blokování virů, spyware a jiného malware. Dále pak blokování nebezpečných odkazů, stahovaných souborů a emailových příloh. AVG Antivirus Free se také snaží kontrolovat výkon počítače a minimalizovat jeho vytížení. Posledním důležitým prvkem, který AVG v neplacené verzi nabízí, jsou aktualizace zabezpečení v reálném čase. Stejně jako Avast je AVG dostupné pro několik platforem. [11]

2.4.2 Systémové požadavky + funkce placených verzí

Minimální systémové požadavky pro antivir AVG jsou:

- OS: Windows 10, 8.1, 8, 7, Vista nebo XP
- CPU: Intel Pentium 1,5 GHz nebo rychlejší
- RAM: 512 MB (Win XP), 1024 MB (novější Windows)
- HDD: 1,2 GB volného místa

AVG nabízí i podporu zařízení s OS Mac OS X a mobilních zařízení s OS Android.

Nejlepší placená verze AVG Ultimate nabízí oproti verzi free např.: šifrování soukromých dat, ochranu při placení, vlastní firewall a spousty dalších funkcí. [12]

2.5 AVIRA

2.5.1 Charakteristika antiviru

Antivirus Avira, je antivirový program vyvíjený německou společností Avira. V neplacené verzi nabízí „zabezpečení budoucí generace“, které umožňuje identifikovat, zablokovat, nebo odstranit ransomware (malware, který šifruje soubory, nebo blokuje přístupy, za jejichž odblokování/navrácení jsou poté vymáhány peníze), který by se mohl zmocnit důvěrných dat. Podobně jako AVG dokáže optimalizovat PC pro co nejlepší výkon. Nabízí pravidelné aktualizace pro zajištění co nejlepší bezpečnosti. Mezi dodatečnými funkcemi Avira nabízí osobní ochranu, která je zajištěna použitím Avira Phantom VPN. Ta spočívá ve skrývání IP adresy a šifrování komunikace, ochraně před nežádoucími inzeráty a také matení webových stránek s geografickým omezením, aby nemohl byt používaný systém sledován. [13]

2.5.2 Systémové požadavky + funkce placených verzí

Minimální systémové požadavky pro antivir Avira jsou:

- OS: Windows 10, 8.1, 8, 7
- CPU: Intel Pentium 4 1 GHz nebo rychlejší
- RAM: 1024 MB
- HDD: 800 MB volného místa

Antivir Avira je také k dostání na OS Mac OS X.

Placená verze Avira Pro nabízí oproti verzi free: kontrolu zařízení (USB flash disk atd.), ochranu při nakupování, správě bankovních účtů, mobilní a emailovou podporu od tvůrců antiviru Avira a dále pak neobsahuje žádné reklamy. [14]

2.6 WINDOWS DEFENDER

2.6.1 Charakteristika antiviru

Windows Defender je antivirový program, zástupce cloudových antivirů, od společnosti Microsoft. Bezplatné řešení antiviru nabízí ochranu před potenciálními trhlinami. Chrání aplikace a data se zvyšujícím se přístupem ke cloudovým službám. Defender nabízí ochranu v reálném čase před viry, malware a spyware. Nabízí pravidelné aktualizace zdarma. Dle Microsoftu nabízí nejen zabezpečení osobních zařízení, ale i zabezpečení firemních zařízení. [15]

2.6.2 Systémové požadavky + funkce placených verzí

Systémové požadavky pro antivir Windows Defender jsou určeny požadavky na daný systém od Microsoftu. Neboli, jakou systémovou konfiguraci vyžadují Windows 10, 8.1, 8, 7, nebo XP.

Microsoft nevytváří žádnou vylepšenou (placenou) verzi oproti Windows Defender, pouze nabízí uživateli při zakoupení licence Windows antivir jako bonus zadarmo (v novějších systémech jako vestavěnou součást).

2.7 BITDEFENDER

2.7.1 Charakteristika antiviru

Antivirový program od rumunských tvůrců. Bitdefender obsahuje anti-phishingovou kontrolu, což znamená, že blokuje stránky, které se tváří důvěryhodně a pouze se snaží z uživatelů vymámit peníze, nebo získat hesla. Automaticky skenuje a odstraňuje viry, ale toto může dělat i na vyžádání. V případě, že se objeví hrozba, je zablokována/zničena na pozadí, aniž by byl uživatel vyrušen. Také obsahuje tzv. anti-fraud kontrolu, která upozorní uživatele při vstupu na stránky typu: kasina, erotické stránky, bankovní stránky, kde je vysoké riziko podvedení uživatele. [16]

2.7.2 Systémové požadavky + funkce placených verzí

Minimální systémové požadavky pro antivir Bitdefender jsou:

- OS: Windows 10, 8.1, 8, 7
- CPU: Intel Dual Core 1,6 GHz
- RAM: 1,5 GB
- HDD: 2 GB volného místa

Taktéž Bitdefender je k dispozici pro zařízení se systémem Mac OS X a mobilní zařízení se systémem Android.

Placené verze oproti verzi free nabízí: ochranu dat, aktivní ochranu před hrozbami, která je založena na detekci nepřirozeného chování aplikací, vlastní firewall a jiné další. [17]

2.8 PANDA

2.8.1 Charakteristika antiviru

Antivirus Panda vytvořený ve Španělsku nabízí řešení antivirového programu, který nezatíží výkon PC, na kterém je nainstalován, protože všechna práce probíhá v cloudu. Poskytuje

maximální zabezpečení proti nejnovějším virům bez toho, aniž by se museli provádět každodenní aktualizace a ukládat spousta souborů do PC. Instalace a následný provoz samotného antiviru je velice jednoduchý, protože je vše pro uživatele již připraveno, takže po instalaci již nemusí nic nastavovat. [18]

2.8.2 Systémové požadavky + funkce placených verzí

Minimální systémové požadavky pro antivir Bitdefender jsou:

- OS: Windows 10, 8.1, 8, 7, Windows Server 2003, 2008, 2012
- CPU: 400MHz
- RAM: 96 MB (klient), 128 MB (server)
- HDD: 280 MB (32bit), 610 MB (64bit)
- Rozlišení obrazovky: 800x600 (256 barev)

Antivir Panda je dostupný i na OS Mac OS X, OS Android a OS iOS pro mobilní zařízení, nebo také pro různé linuxové distribuce.

Placené verze Panda oproti verzi free nabízí: šifrování dat, ochranu při používání internetového bankovnictví, ochranu Wi-Fi, nebo zálohu souborů s rychlým a jednoduchým obnovením. [19]

2.9 HODNOCENÍ ANTIVIRŮ

V následující kapitole budou představeny výsledky (z roku 2016 a počátku roku 2017, čili hodnocení vychází spíše z roku 2016) z několika serverů.

Na prvním serveru pcmag.com byly antiviry testovány na Windows 10 a hodnoceny podle těchto kritérií: skenování malware na vyžádání, automatické skenování malware, ochrana na webu, blokování nežádoucích URL, ochrana před phishing, detekce na základě chování uživatele a skenování hrozeb. Z výsledků v souhrnu nejlépe vyšel antivir Avast, následován AVG, třetí skončil antivir Panda, čtvrtou příčku obsadil Bitdefender, pátou Avira a poslední příčku obsadil Microsoft Defender, který díky nízkému hodnocení nebyl zobrazen v podrobné tabulce výsledků. [20] (Tabulka 1)

Tabulka 1 – Srovnání pcmag.com (zdroj [20])

	Avast	AVG	Bitdefender	Avira	Panda
Detekce malware na požádání	ANO	ANO	ANO	ANO	ANO
Detekce malware automaticky	ANO	ANO	ANO	ANO	ANO
Website rating (zda je web bezpečný, či nikoliv)	ANO	ANO	NE	ANO	NE
Blokování nebezpečných URL	ANO	ANO	ANO	ANO	ANO
Ochrana proti phishing	ANO	ANO	ANO	ANO	ANO
Zachycení neobvyklého chování	ANO	ANO	ANO	NE	NE
Bonus: skenování slabých míst systému	ANO	NE	NE	NE	NE

Na serveru tomsguide.com byly testy prováděny opět na OS Windows 10 a hodnocení probíhalo postupně dle těchto kritérií: zatížení PC, jednoduchost použití antivirového programu, detekce malware (označován jako widespread) a zero-day útoku (útočník vytváří malware co nejdříve po tom, kdy je známa slabina systému). Pro výsledky, jak si který antivir poradí s hrozbami, byly použity výsledky nezávislých laboratoří AV-TEST v Německu a AV-Comparatives v Austrálii z počátku roku 2016 (pro Windows 7 – 32bit) a z konce roku 2015 (pro Windows 8.1 – 64bit), protože od té doby nevyšly testy pro zde vyjmenované antiviry. Pro účel této práce bude důležitý především přehled, jak si který antivir poradil s detekcí běžně známého malware. [21] (Tabulka 2, Tabulka 3)

Tabulka 2 – Srovnání antivirů ze serveru tomsguide.com, Windows 7 32bit – leden, únor 2016 (zdroj [22])

	Avast	AVG	Avira	Bitdefender	Microsoft	Panda
ZERO-DAY malware (leden)	98,2%	100%	99,1%	100%	91,8%	100%
ZERO-DAY malware (únor)	98,1%	98,1%	99,0%	100%	86,4%	99,0%
Běžně známý malware (leden)	99,7%	99,8%	99,9%	100%	99,7%	100%
Běžně známý malware (únor)	99,3%	99,9%	99,9%	100%	99,6%	100%
Nesprávně identifikované hrozby	3	1	0	0	3	5

Tabulka 3 – Srovnání antivirů ze serveru tomsguide.com, Windows 8.1 64bit – listopad, prosinec 2015 (zdroj [22])

	Avast	AVG	Avira	Bitdefender	Microsoft	Panda
ZERO-DAY malware (leden)	100%	98,8%	100%	100%	97,5%	98,8%
ZERO-DAY malware (únor)	98,3%	100%	100%	100%	90%	100%
Běžně známý malware (leden)	99,8%	99,9%	100%	99,9%	99,6%	99,9%
Běžně známý malware (únor)	99,8%	99,7%	99,9%	100%	99,6%	99,9%
Nesprávně identifikované hrozby	3	1	0	1	1	7

Dle serveru techradar.com, který opět využil služeb laboratoří AV-TEST, byly jednotlivé antiviry hodnoceny znovu z hlediska výkonnosti a spolehlivosti 24 hodin. Mimo jiné, jestli antivirus zbytečně neobtěžuje uživatele např. vyskakovacími okny s nabídkou prémiové (placené) verze programu. A v neposlední řadě testování na ochranu před hrozbami, které se shoduje s výsledky ze serveru tomsguide.com. V souhrnných výsledcích pak nejlépe dopadl antivir Avira, následovaný postupně antiviry AVG, Panda, Avast, Bitdefender a poslední příčku obsadil Microsoft Defender. [23]

Tato práce se bude snažit ukázat, jak si povede 6 výše uvedených antivirových programů v aktuálních verzích. Zároveň bude možné porovnat a zjistit, jak se naměřené výsledky shodují s výsledky ukázaných v tabulkách výše.

3 KALI LINUX

3.1 Charakteristika OS

Kali Linux je jednou z mnoha distribucí OS Linux, volně šiřitelného OS typu UNIX, a jeho první verze byla vydána v březnu roku 2013. Tento systém je využíván především k penetračnímu testování a kontrole zabezpečení. Předchůdcem této distribuce byla jiná distribuce Linuxu, a to distribuce BackTrack. Distribuce Kali Linux se stala jakýmsi vylepšením, přestavbou BackTrack a převzala pouze funkční nástroje. (BackTrack obsahoval mnohem více nástrojů než Kali, ale některé byly i nefunkční, a proto byly odstraněny.) Distribuce Kali Linux je založena na jiné distribuci a tou je Debian. Stejně jako BackTrack si klade za cíl, že je a bude vždy zcela zdarma a také, že veškerý kód, který je součástí Kali Linux, je a bude vždy dostupný pro kohokoli. Zároveň dodržuje standard FHS, umožňující uživatelům Linuxu snadno najít binární podpůrné soubory, knihovny atd. Dále podporuje tolik bezdrátových zařízení, kolik uživatel chce a potřebuje. Samotné jádro (kernel) je upraveno speciálně pro tuto distribuci a může být upraveno kýmkoli podle vlastního vkusu a dle vlastních potřeb. Neustále vycházejí bezpečnostní záplaty a bezpečnost je zajištěna také tím, že tým spravující a tvořící tuto distribuci je pouze malá skupina lidí, která jediná může schvalovat a přidávat do oficiálních repositářů nové balíčky programů (nástrojů). Každý nový balíček je též ošetřen zašifrovaným elektronickým podpisem (GPG) toho, kdo vytvořil a sdílel tento balíček. Kali Linux je dostupný v několika jazycích a je vyvíjen i s ohledem na ARMEL a ARMHF systémy[1]. [24]

3.2 VYBRANÉ NÁSTROJE KALI LINUX – STRUČNÝ POPIS

Kali Linux v sobě zahrnuje mnoho nástrojů (okolo 600), kterými lze dosáhnout potřebných kontrol, penetračních testování. Pro penetrační testování slouží nástroje, které jsou rozděleny do těchto kategorií:

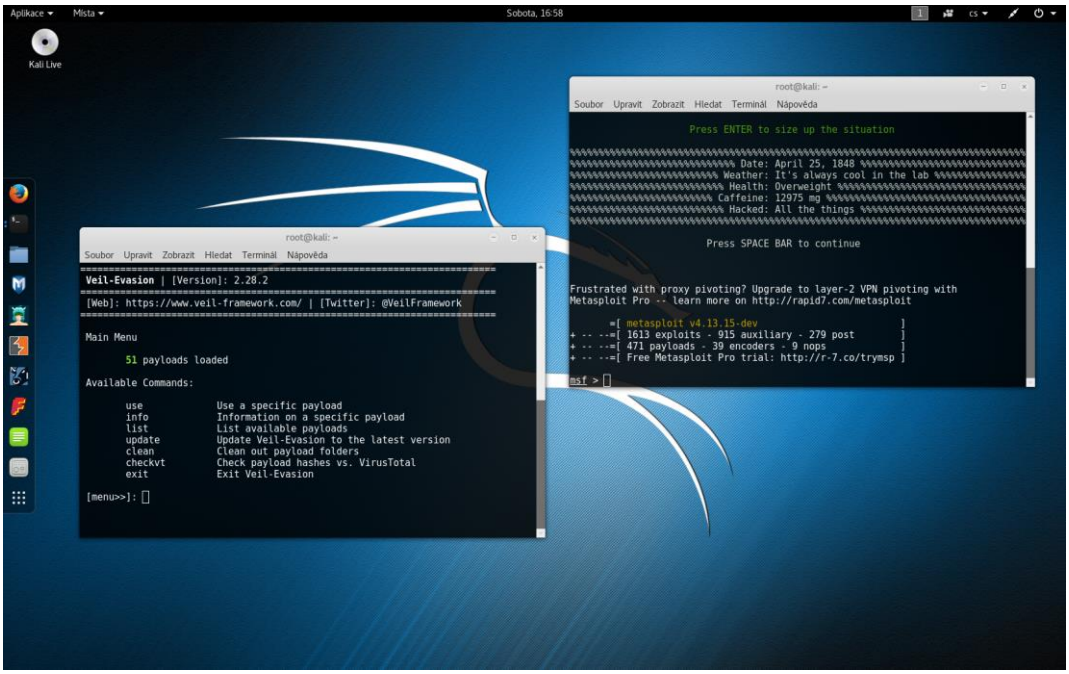
1. **sběr informací,**
2. **odhalování zranitelnosti,**
3. **webové aplikace,**
4. **odhalování hesel,**
5. **exploatační nástroje,**
6. **sniffing a spoofing,**

[1] **ARMEL a ARMHF systémy** – systémy používající architekturu procesorů ARM, jenž je používaná i díky své nízké spotřebě např. na mobilních zařízeních

7. **nástroje pro zachování přístupu,**
8. **nástroje pro dokumentaci,**
9. **systémové nástroje.**

Pro účely testování antivirových programů byly využity frameworky Metasploit a Veil jako zástupci exploitačních nástrojů [25].

Ukázka z prostředí Kali Linux s otevřenými příkazovými řádky framevorků Metasploit a Veil (Obrázek 3).



Obrázek 3 – Kali Linux (zdroj – vlastní)

3.2.1 Sběr informací

Zenmap/Nmap – jedněmi z nejpoužívanějších nástrojů z oblasti sběru informací jsou Zenmap a Nmap. Zenmap je grafické prostředí sloužící k snadnějšímu využití Nmap. Nmap je velmi populární a spolehlivý balíček síťových nástrojů. V tomto balíčku jsou zahrnuty nástroje jako: *nping* – síťový nástroj pro generování packetů pro široké spektrum protokolů (UDP, TCP, ICMP, ARP), *ndiff* – nástroj pro seřazení výsledků nasbíraných všemi nástroji Nmap balíčku (všechny nasbírané informace lze vyexportovat do několika formátů - xml, textový soubor, výstup do databáze nebo výstup zpracovatelný nástrojem grep, který dokáže vybrat data na základě regulárních výrazů), *ncat* – spojování a přesměrovávání portů, *nmap* – nástroj pro mapování sítě [26]. Pomocí Nmap lze zjistit několika způsoby, zda je zařízení vypnuté, nebo zapnuté. Existuje několik způsobů, jak sledovat dané zařízení (TCP SYN/ACK host discovery scan, UDP host discovery scan, ACK scanner) a jak najít jeho služby (TCP connect scan, UDP

connect scan, ACK scan). Dokáže také rozpoznat operační systém a verze běžících služeb (aplikací) na daném systému. Ve skriptovacím jazyce Lua je možné vytvořit vlastní skript, čímž lze Nmap neomezeně rozšiřovat. Mimo jiné obsahuje pokročilé techniky k oklamání firewall, IDS, anebo obsahuje sadu nástrojů pro benchmarking sítě [27]. Mnohem širší využití lze nalézt v příručce od samotného tvůrce nástrojů Nmap [28]. Nmap je hojně využíván právě v rámci frameworku Metasploit (je v něm zakomponován) pro zmonitorování sítě před exploitací.

3.2.2 Odhalování zranitelnosti

Mezi nástroje, které umí odhalovat zranitelnost, patří mimo jiné také Nmap popsaný v kapitole 3.2.1.

3.2.3 Webové aplikace

Burp Suite – velice užitečná sada nástrojů (Target, Proxy, Spider, Scanner, Intruder, Repeater, Sequencer, Decoder) pro penetrační testování webových aplikací, skenování, analyzování a exploitace webových aplikací. [29]

3.2.4 Odhalování hesel

Hashcat – více vláknový nástroj pro odhalení hesel. Využívá cca 80 algoritmů k odhalení hesel a algoritmy jsou počítány za pomoci CPU. [30]

Rainbowcrack – nástroj pro dešifrování hesel. Klasický útok „brute force“ porovnává heslo s hesly v tabulkách, které jsou při útoku překládány a šifrovány a tyto šifry se poté porovnávají s požadovaným heslem. Výhodou tohoto nástroje je urychlení díky tomu, že hesla jsou již předem šifrována a tento proces neprobíhá v průběhu provádění dešifrování požadovaného hesla. [30]

John – další z nástrojů pro dešifrování hesla. Výhodou oproti ostatním je to, že dokáže rozšifrovat až 40 druhů šifrovacích algoritmů (MD5, DES, LM, NT atd.). [30]

3.2.5 Exploitační nástroje

Mezi exploitační nástroje patří zejména Metasploit, ale také Veil. Jsou jim věnovány kapitoly 4 a 5.

3.2.6 Sniffing a spoofing

Wireshark – velice oblíbený nástroj. Jedná se o protokolový analyzér a paketový sniffer (zachytávač paketů). Dokáže pracovat s více jak 1000 protokoly. Uživatelské prostředí je přehledné, takže vyhledávání informací v jednotlivých paketech je velice usnadněno, a to i díky možnosti podrobného filtrování jednotlivých informací. [31]

Ettercap – sada nástrojů pro útok man-in-the-middle. Lze např. přesměrovat komunikaci, odhalit heslo pro FTP, HTTP, POP, SSH protokoly, nebo vytvořit falešný SSL certifikát, díky němuž je znemožněn přístup oběti přes HTTPS. [32]

3.2.7 Ostatní nástroje

Intersect – nástroj používaný pro post-exploitační úkony, kterými jsou sběr a kopírování hesel, SSH klíčů, sběr informací o síti a také identifikace antivirového programu nebo firewall aplikací. [33]

Casefile – nástroj pro zaznamenávání, vytváření jakéhosi diagramu myšlenek pro lepší přehlednost a grafickou představu při testování. [34]

Systémové nástroje – kategorie obsahující nástroje používané v průběhu penetračního testování: Metasploit, Apache služby, MySQL služby, SSH služby a další služby. [35]

Existují ještě další kategorie a nástroje, které se nepoužívají k penetračnímu testování. Mezi ně patří [35]:

- **Bezdrátové útoky** – útoky na sítě Bluetooth, RFID/NFC a na bezdrátové zařízení.
- **Reverzní inženýrství** – k odstranění chyb v programu nebo k získání kódu z programu.
- **Zátěžové testování** – pro otestování sítě, webu, VOIP prostředí v zátěži.
- **Hardware hacking** – nástroje pro práci s Andriod nebo Arduino aplikacemi.
- **Forenzní nástroje** – k vyšetřování, získávání image disků, poškozených dat, nebo také analýze image pevného disku.

Kali Linux obsahuje v jednotlivých kategoriích mnohem více nástrojů, než zde byly vyjmenovány. Ostatně jedná se o systém, který dokáže velice účinně otestovat zranitelnost jednoho zařízení, nebo i celé sítě. Množství informací k jednotlivým nástrojům lze nalézt na stránkách oficiální dokumentace Kali Linux: http://docs.kali.org/, nebo http://tools.kali.org/.

4 FRAMEWORK METASPLOIT

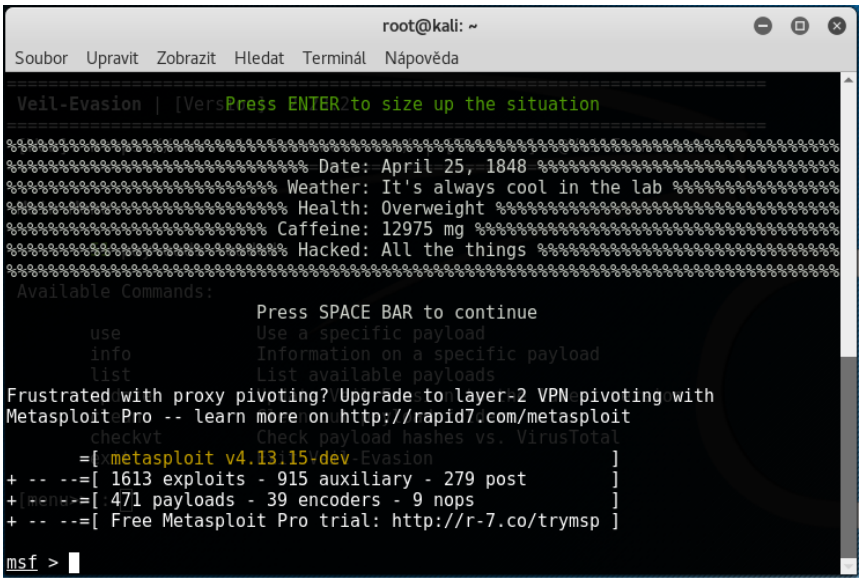
4.1 Charakteristika

Framework Metasploit byl vytvořen v programovacím jazyce Ruby. Zjednodušuje práci penetračním testerům při rozšiřování nebo vytváření nových pluginů a nástrojů. Je přizpůsoben, jak pro práci v UNIX systémech, tak pro práci v systémech Windows ve spolupráci s podpůrným prostředím Cygwin, jenž napodobuje chování UNIX systémů. Zároveň framework nabízí řadu doplňků v mnoha jazykových lokalizacích. Framework je možné ovládat 3 způsoby: příkazová řádka, webové rozhraní anebo GUI. [36]

Stavebně je framework rozdělen na 3 části: knihovny, rozhraní a moduly. Pro účely testování antivirových programů jsou důležité především rozhraní a moduly. Mezi moduly zahrnujeme **exploity** – kódy, které využívají zranitelnosti cizího systému; **payloads** – malware, je buď součástí exploitu, nebo oddělený, jako nezávislá jednotka vykonávající zadané příkazy na cílovém systému; **auxiliaries** – sada nástrojů určená k odposlouchávání, skenování např. otisků prstů; **encoders** – moduly určené k znemožnění odhalení payloads antivirovými programy; **NOP** (No Operation nebo No Operation Performed) – jazyk symbolických instrukcí přidaných do payloads k zajištění jejich konzistence. [36]

4.2 Ovládání frameworku

Prostředí příkazové řádky Metasploitu – MSF console vypadá následovně. (viz Obrázek 4)



Obrázek 4 – Metasploit – příkazový řádek (zdroj – vlastní)

K zobrazení jednotlivých modulů se používá příkaz *show* následovaný požadovaným modulem. Přehled možných *show* příkazů je popsán v následujícím seznamu (zdroj [37]):

- **show auxiliary** – zobrazení všech pomocných modulů,
- **show exploits** – zobrazení všech dostupných exploitů,
- **show payloads** – při použití tohoto příkazu dojde k zobrazení všech payloadů dostupných pro všechny platformy – pokud bude tento příkaz použit uvnitř daného exploitu, zobrazí se pouze kompatibilní payloady k tomuto exploitu,
- **show encoders** – zobrazení všech encoders (určeno k šifrování, skrývání payloads),
- **show nop** – zobrazení všech dostupných NOP generátorů,
- **show options** – zobrazení nastavení pro určitý modul (auxiliary, exploits, payloads),
- **show targets** – zobrazení takových OS, které jsou zranitelné – pomáhá vyselektovat jenom zranitelné systémy,
- **show advanced** – zobrazení více nastavení pro ještě lepší využití provádění exploitace.

Ostatní příkazy pro ovládání frameworku jsou obsaženy v této tabulce (Tabulka 4).

Tabulka 4 – Příkazy pro ovládání frameworku Metasploit (zdroj [37])

PŘÍKAZ	POPIS
check	Ověření, zda je cíl zranitelný, aniž by proběhla exploitace.
connect *ip port*	Pracuje stejně jako Netcat nebo Telnet nástroje.
exploit	Spuštění vybraného exploitu.
run	Spuštění vybraného auxiliary.
jobs	Zobrazení běžících modulů na pozadí a možnost je ukončit.
route *add subnet netmask sessionid*	Přidání cesty skrze napadané zařízení k napadení dalšího zařízení patřící do sítě ovládaného zařízení.
info *module*	Zobrazení detailních informací týkajících se exploitu, auxiliary atd.
set *param value*	Nastavení parametru v aktuálním modulu.
setg *param value*	Nastavení parametru na globální, který může být použit ostatními moduly.
unset *param value*	Reset parametru. Příkazem unset all jsou všechny parametry vyresetovány.
unsetg *param*	Globální dopad příkazu výše.
sessions	Schopnost zobrazit cílové spojení. S přepínačem -l pro zobrazení, s -i pro interakci a s -k pro ukončení.
search *string*	Vyhledávání napříč všemi moduly.
use module	Výběr zvoleného modulu pro penetrační testování.

4.3 Příklady využití

4.3.1 Meterpreter

Meterpreter je nástroj (dynamicky rozšiřitelný payload) pro práci s napadeným zařízením. Nabízí jakousi vlastní příkazovou řádku (shell), která má vlastní sadu příkazů pro práci se zařízením.

Klasická příkazová řádka na jakékoli platformě je na této platformě závislá a pokud je spuštěna, tak se automaticky spouští nový proces v OS. Meterpreter to řeší jiným způsobem. Donutí existující proces, aby do svého virtuálního adresního prostoru zavedl DLL knihovnu, sloužící jako server (tzv. DLL injection) a pro ni vytvořil nové vlákno. Zde poté přijímá požadavky od klienta, kterým je konsole útočníka (MSF console). Požadavky jsou zpracovány ve vlákně, v němž běží meterpreter, čili funguje na principu klient-server. [38]

Funkcionality meterpreteru lze rozšiřovat za běhu. Meterpreter má své vlastní API (Application Programming Interface) tzn., že si kdokoliv může vytvořit svůj vlastní modul a ten nahrát do ovládaného zařízení, nebo i vytvořit vlastní skript a ten spustit. Např. na OS Windows lze také spustit libovolný kód, který je distribuován jako DLL knihovna. [38]

4.3.2 SNMP scanner

Jeden z modulů frameworku Metasploit, vytvářející Simple Network Management Protocol (SNMP) tj. síťový protokol, umožňující pravidelný sběr dat. A díky němu je tak možné odposlouchávat zařízení na síti, služby běžící na síti, síťové adresy, verze, verze oprav atd. [39]

4.3.3 VNC scanner

Tento modul slouží ke skenování Virtual Network Computing (VNC) serverů v určitém adresním rozsahu. VNC servery slouží ke vzdálené správě počítače fungující na principu klient-server. VNC scanner tyto servery ověřuje, a to z hlediska autentizace, zda je server dostupný, nebo nikoliv. [40]

5 FRAMEWORK VEIL

5.1 Stručná charakteristika

Framework Veil je sada bezpečnostních nástrojů obsahujících různé metody útoků, jež slouží k obelstění cílových systémů. Zahrnuje v sobě tyto nástroje [41]:

- **Veil-Evasion –** nástroj, sloužící ke generování payloads, schopných obejít antivirový program za pomocí různých útočných technik a za pomocí různých programovacích jazyků.
- **Veil-Catapult –** nástroj spolupracující s Veil-Evasion, pomocí něhož se může vygenerovaný payload dopravit na cílový systém.
- **Veil- Ordnance –** nástroj umožňující rychle vygenerovat validní shellcode (krátký kód psaný v assembleru), který slouží jako nosné medium pro payload.
- **Veil-PowerTools** – sada nástrojů obsahující projekty ve skriptovacím jazyce PowerShell (skriptovací jazyk a shell od firmy Microsoft), určené a tvořené pro útočné operace.
- **Veil-Pillage –** framework sloužící k post-exploitačním potřebám.

Specifičtější využití některých těchto nástrojů bude představeno v kapitole 7 týkající se praktického testování.

5.2 Instalace jednotlivých nástrojů

Jednotlivé nástroje frameworku Veil nejsou v Kali Linux ve výchozím nastavení nainstalovány, a proto je zapotřebí si tyto nástroje dostáhnout a doinstalovat.

Veil-Evasion – tento nástroj můžeme do systému Kali Linux nainstalovat jednoduchým příkazem *apt-get install veil-evasion*. Dobré je ještě před jeho instalací aktualizovat databázi balíčků z repositářů příkazem *apt-get update*.

Veil-Catapult – podobně jako Veil-Evasion se nainstaluje příkazem *apt-get install veil-catapult* a před jeho instalací je opět dobré aktualizovat databázi balíčků z repositářů.

Veil-Ordnance – je nástroj, který stačí stáhnout ze stránek GitHub. K nalezení pod tímto odkazem: https://github.com/Veil-Framework/Veil-Ordnance.

Veil-PowerTools – podobně jako u nástroje Veil-Ordnance lze celou sadu těchto nástrojů (PowerView, PowerPick, PowerUp, PowerBreach, PewPewPew) stáhnout ze stránek GitHub pod odkazem: https://github.com/PowerShellEmpire/PowerTools.

Veil-Pillage – tento nástroj je také možno najít na stránkách GitHub pod odkazem: https://github.com/Veil-Framework/Veil-Pillage.

6 MALWARE

6.1 Co je to malware

Malware je škodlivý kód, software. Není nijak vědecky určeno, jak se který malware nazývá. Lze však určit vlastnosti malware. Tyto vlastnosti jsou uvedeny v následujícím seznamu [42]:

- **Samo-replikovatelný** – takový malware, který sám vytváří kopie a rozšiřuje se dál např. po síti.
- **Populační růst** – určuje počet instancí vytvořených při sebe-replikaci.
- **Parazitní** – vyžaduje jiný spustitelný kód, na kterém parazituje.

6.2 Druhy malware

Malware je jakýkoli typ škodlivého software snažící se infikovat PC, tablet, mobil nebo jiné zařízení. Hackeři využívají malware pro různé účely: např. k získávání osobních údajů, krádeži peněz nebo firemních dat, případně k získání přístupu k napadenému zařízení. Malware může napadnout zařízení prostřednictvím internetu, většinou po stažení jakéhokoli souboru (fotografie, MP3, hra atd.), který ovládne počítač uživatele. V případě napadení systému, některým z typů malware, je dobré použít antivirový systém k jeho odhalení. [43]

6.2.1 Logic bomb

Škodlivý software skládající se ze dvou částí:

- **Payload** – akce, která se má provést. Důležité je, že má škodlivý účinek.
- **Trigger (spoušť)** – provádí akci (payload) např. na základě času, přihlášení uživatele. Také může být vytvořen tak, že je možné jej spustit na dálku.

Logic bomb může být zakomponován do nějakého spustitelného kódu, nebo může existovat jako samotný kód. [44]

Dopady logic bomb mohou způsobit např. smazání souborů v rámci jedné sítě konkrétní společnosti a tím i finanční ztrátu. [44]

6.2.2 Trojan horse

Trojan horse se tváří, jako by dělal neškodné činnosti, ale tím jen skrývá svojí škodlivou aktivitu. Krásným příkladem je přihlašovací formulář. Uživatel se dostane na přihlašovací formulář, který se nijak neliší od originálního formuláře. Zadá svoje přihlašovací údaje a vrátí se mu odezva s tím, že zadal špatné heslo. V domnění, že došlo k typografické chybě, zadá heslo ještě jednou, jenže tou dobou už trojan získal uživatelovi přihlašovací údaje. [44]

6.2.3 Backdoor

Mechanismus obcházející normální bezpečnostní kontrolu. Programátoři si někdy z jistých důvodů vytváří takováto „zadní vrátka“, např. z důvodu ušetření času při testování. Příkladem může být testování systému s přihlašováním, kde si programátor přímo v kódu zadá, že uživatel jistého hesla a jména má vždy povolen přístup do systému. (Programátor si nemusel vytvářet žádného uživatele pro účely testování a ušetřil čas.) [44]

Tento druh malware může být zakomponován buď do nějakého funkčního kódu, anebo může stejně jako logická bomba existovat samostatně.

Jedním z typů backdoor je RAT (Remote Administration Tool nebo Remote Access Trojan). Tyto programy jsou v PC určeny k monitorování a kontrole na dálku. Uživatelé si takový program záměrně nainstalují, když potřebují nějaký druh vzdálené podpory, anebo chtějí pracovat z domu. Nicméně pokud je v RAT ukryt škodlivý kód, dokáže bez vědomí uživatele otevřít zadní vrátka pro přístup. [44]

6.2.4 Virus

Virus je malware (škodlivý software), který se snaží sám sebe replikovat a navázat na jiný spustitelný kód. V případě, že dojde ke spuštění napadeného kódu, mluvíme o infikovaném kódu. A tento infikovaný kód může nakazit další a další kódy, proto mluvíme o sebe-replikaci.

První zmínky o počítačovém viru se objevují v 70. letech 20. stol. ovšem pouze v literatuře, a to v žánru science fiction (s virem se objevuje zmínka i o antiviru). První akademický výzkum virů proběhl v roce 1983, provedený Fredem Cohenem. [44]

Obyčejně je v dnešní době vir šířen buď v rámci jednoho počítače, nebo mezi několika počítači např. s použitím USB flash disků, CD/DVD médií, internetu atd. Vir se může v průběhu sebe-replikace nacházet v těchto stavech [44]:

- **Zárodečný** – originální forma viru, důraz hlavně na replikaci.
- **Oddaný** – virus, ve kterém se nachází chyba a není schopen replikace.
- **Spící** – přítomný v systému, ale doposud nic neinfikoval.

6.2.5 Worm

Sdílí několik charakteristik s viry. Jednou z nich je, že je samo-replikovatelný. Rozdílem je, že worm existuje sám o sobě, není závislý na jiném spustitelném kódu. Druhým rozdílem a důležitou charakteristikou worm je, že se šíří ze stroje na stroj po síti. [44]

6.2.6 Rabbit

Malware, který se množí exponenciální rychlostí. Existují dva typy tohoto malware. První typ způsobuje maximální čerpání systémových zdrojů, stejně jako místa na disku. Program tzv. fork bomb (volný překlad: „větvící se bomba“) spouští uměle vytvořené procesy v nekonečném cyklu. Mají tendenci po sobě zanechávat stopy. Druhým typem rabbit je malware podobný „červu“ (worm). Je samostatným kódem šířícím se po síti, který maže originální kopii po replikaci. Neboli, na síti je vždy pouze jedna kopie malware šířící se z PC na PC. [44]

6.2.7 Spyware

Spyware je software, který sbírá citlivá data a posílá je dál někomu dalšímu. Typicky jsou to uživatelská jména, hesla, emailové adresy (mohou být poté využity pro spam), bankovní účty, softwarové licence a další. Viry a „červi“ shromažďují stejná data, ovšem spyware se sám nereplikuje a do systému se dostane buď s instalací nějakého jiného softwaru, exploitací přes webový prohlížeč, nebo navštívením infikovaných webových stránek. [44]

6.2.8 Adware

Adware má mnoho společného se spyware, protože sbírá data o uživatelích a jejich zvycích. Je používán hlavně z marketingových důvodů a to např. formou vyskakovacích oken s reklamami na webových stránkách. [44]

6.2.9 Ostatní malware

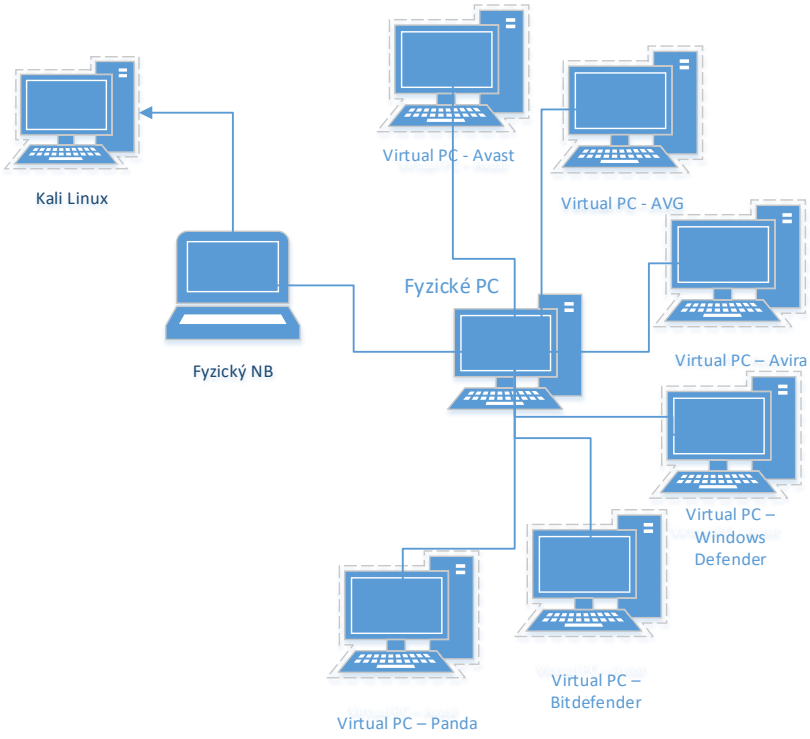
Malware by se mohl rozlišovat a rozdělovat do mnoha dalších kategorií. Mezi nevypsané typy malware může patřit *hybrid malware* – spojující více druhů malware do jednoho, *dropper* – zanechává za sebou, „ukapává“ jiný malware, *zombie* – počítač využívaný útočníkem např. ke spamování. [44]

Vzhledem k aktuální situaci na poli bezpečnosti v IT s velkou určitostí existují mnohem sofistikovanější typy malware, o nichž svět prozatím nemusel ani slyšet.

7 PRŮBĚH A ZKOUMÁNÍ ANTIVIROVÝCH PROGRAMŮ

7.1 Použité technologie

Ke zkoumání a testování antivirových programů byly využity 2 PC. Prvním z PC byl notebook značky HP konkrétně typ Probook 450, kde byl nasazen do virtuálního prostředí, VMware Workstation 11.0.0, Kali Linux (32bitová verze) s oběma frameworky Metasploit a Veil. Druhé PC mělo konfiguraci: procesor – AMD Phenom II X4 965, operační paměť – 4x2 GB DDR3, grafická karta – MSI GTX 660. Na tomto druhém PC bylo ve VMware Workstation 11.0.0 nainstalováno 6 systémů Windows 10 Pro x64, na kterých byly nainstalovány jednotlivé antivirové programy. Přenášení payloadů v komprimované formě zip na cílový systém probíhalo za pomoci flash disku ADATA 32 GB. Na obrázku (Obrázek 5) je zobrazena topologie sítě, která posloužila k testování.



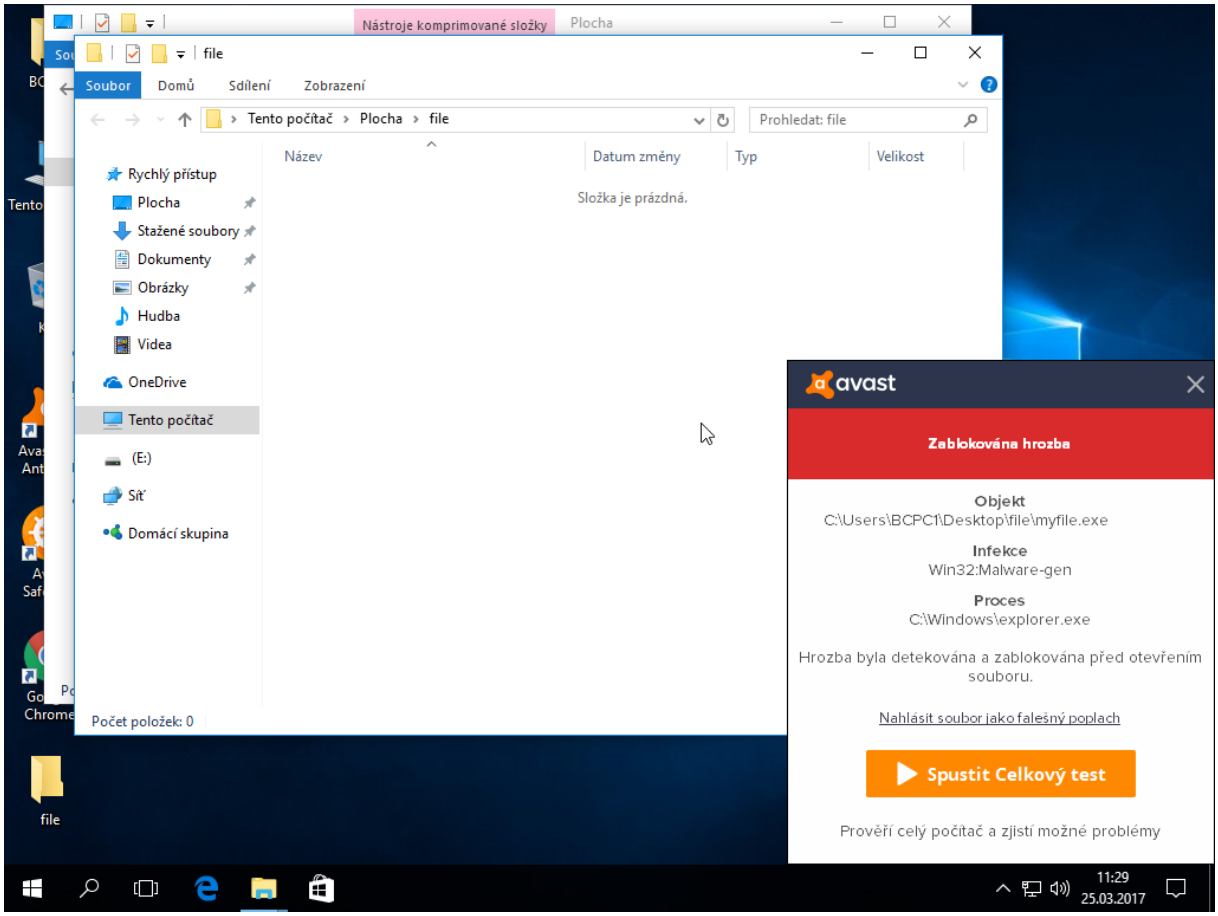
Obrázek 5 – Topologie sítě pro testování (zdroj – vlastní)

7.2 Scénář 1

První scénář testující jednotlivé antivirové programy, využívá model komunikace klient–server (pro komunikaci je na ovládnutném systému otevřen port na němž poté probíhá komunikace). Nejprve byl vygenerován payload ve frameworku Veil–Evasion, konkrétně v jazyce C. Po kompilaci zdrojového kódu vznikl spustitelný soubor myfile.exe, který byl postupně nasazován do jednotlivých systémů za pomoci USB flash disku. Ve frameworku metasploit byl spuštěn exploit multi/handler, vystupující jako server, který čekal, až myfile.exe otevře domluvený port a do cílového systému zavede meterpreter. V případě, že myfile.exe nebyl detekován antivirem, bylo spojení navázáno a server (meterpreter) mohl jakkoli ovládat cílové zařízení.

7.2.1 Avast

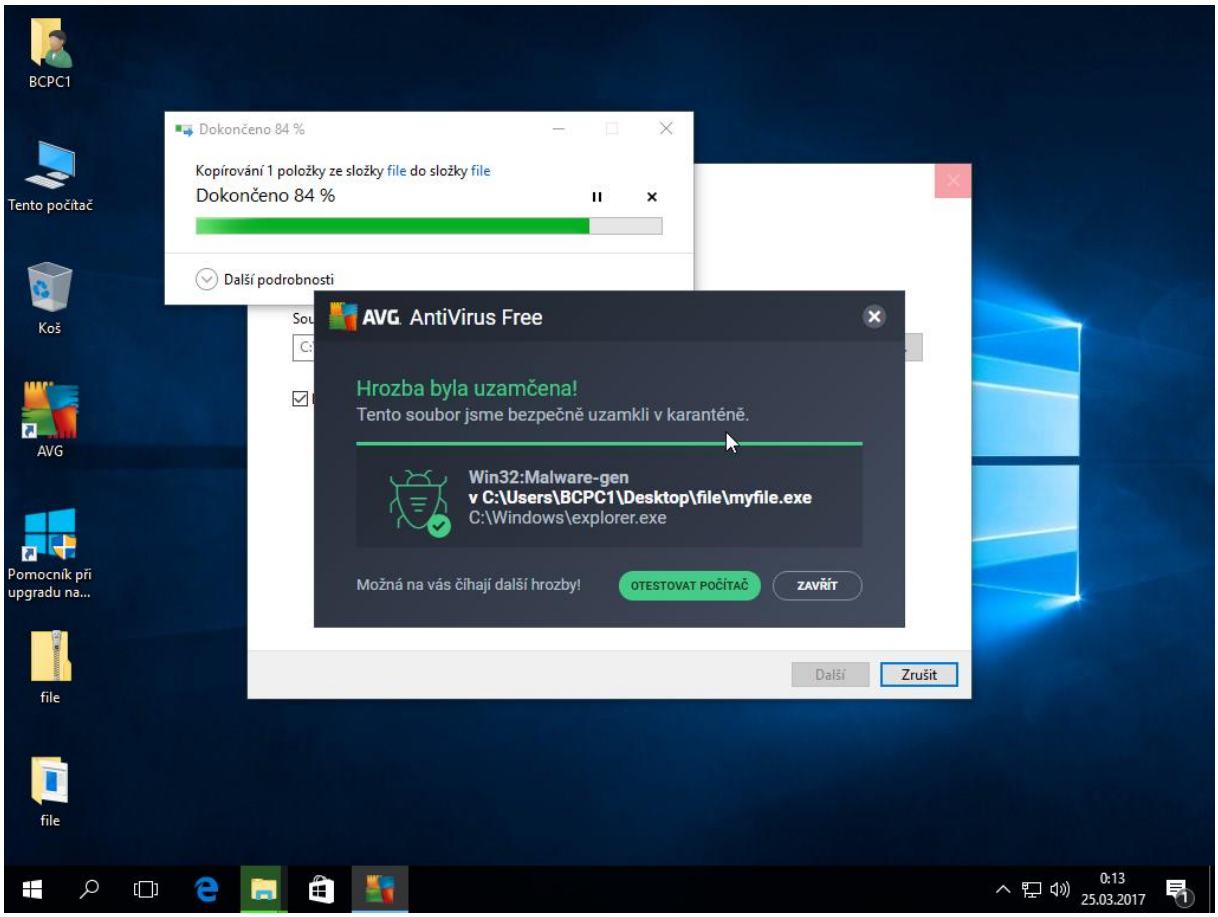
Antivirový program Avast bez problému detekoval, že myfile.exe je hrozbou pro systém. A to již při extrahování složky s tímto souborem. Avast soubor okamžitě zablokoval a informoval uživatele. (Obrázek 6)



Obrázek 6 – Detekce hrozby – scénář 1 – Avast (zdroj – vlastní)

7.2.2 AVG

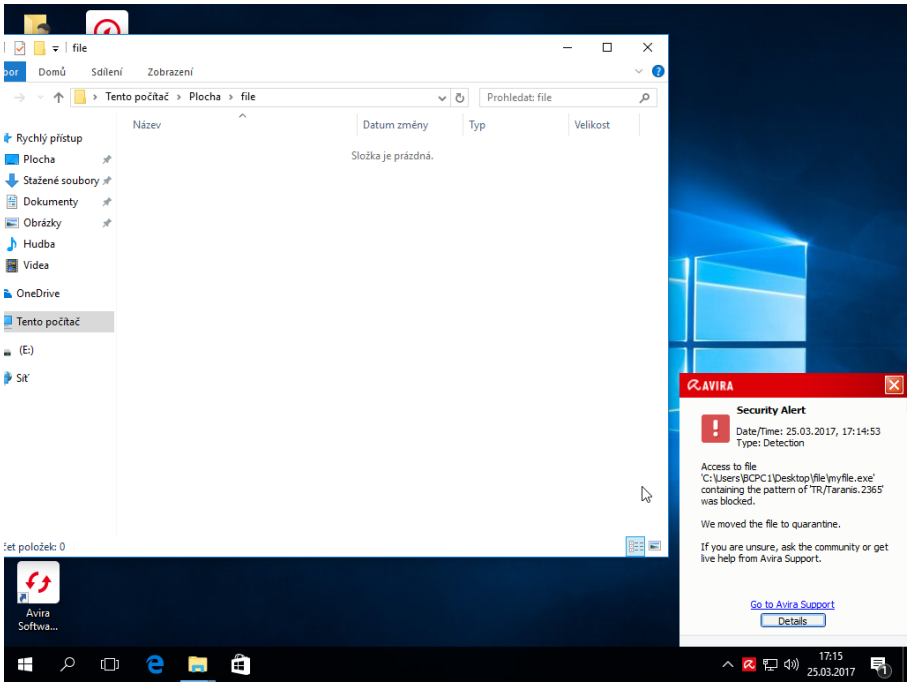
AVG podobně jako Avast bez problému identifikovalo hrozbu, a to stejným způsobem, takže už při extrahování složky s infikovaným souborem tento soubor zneškodnilo a opět oznámilo uživateli, co se se souborem stalo. (Obrázek 7)



Obrázek 7 – Detekce hrozby – scénář 1 – AVG (zdroj – vlastní)

7.2.3 Avira

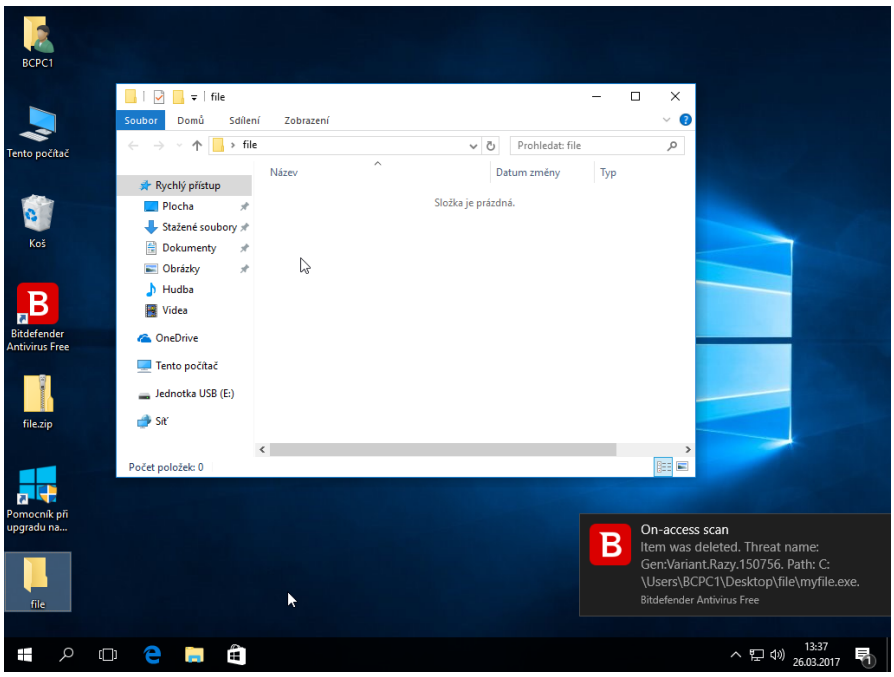
Stejných výsledků jako AVG a Avast dosáhla i Avira. (Obrázek 8)



Obrázek 8 – Detekce hrozby – scénář 1 – Avira (zdroj – vlastní)

7.2.4 Bitdefender

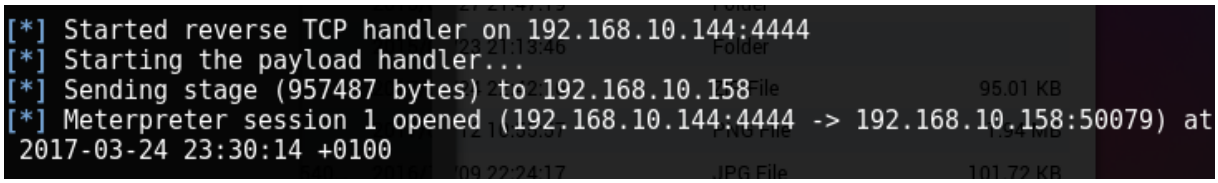
I Bitdefender identifikoval myfile.exe jako hrozbu, tuto hrozbu odstranil a informoval o tom uživatele. (Obrázek 9)



Obrázek 9 – Detekce hrozby – scénář 1 – Bitdefender (zdroj – vlastní)

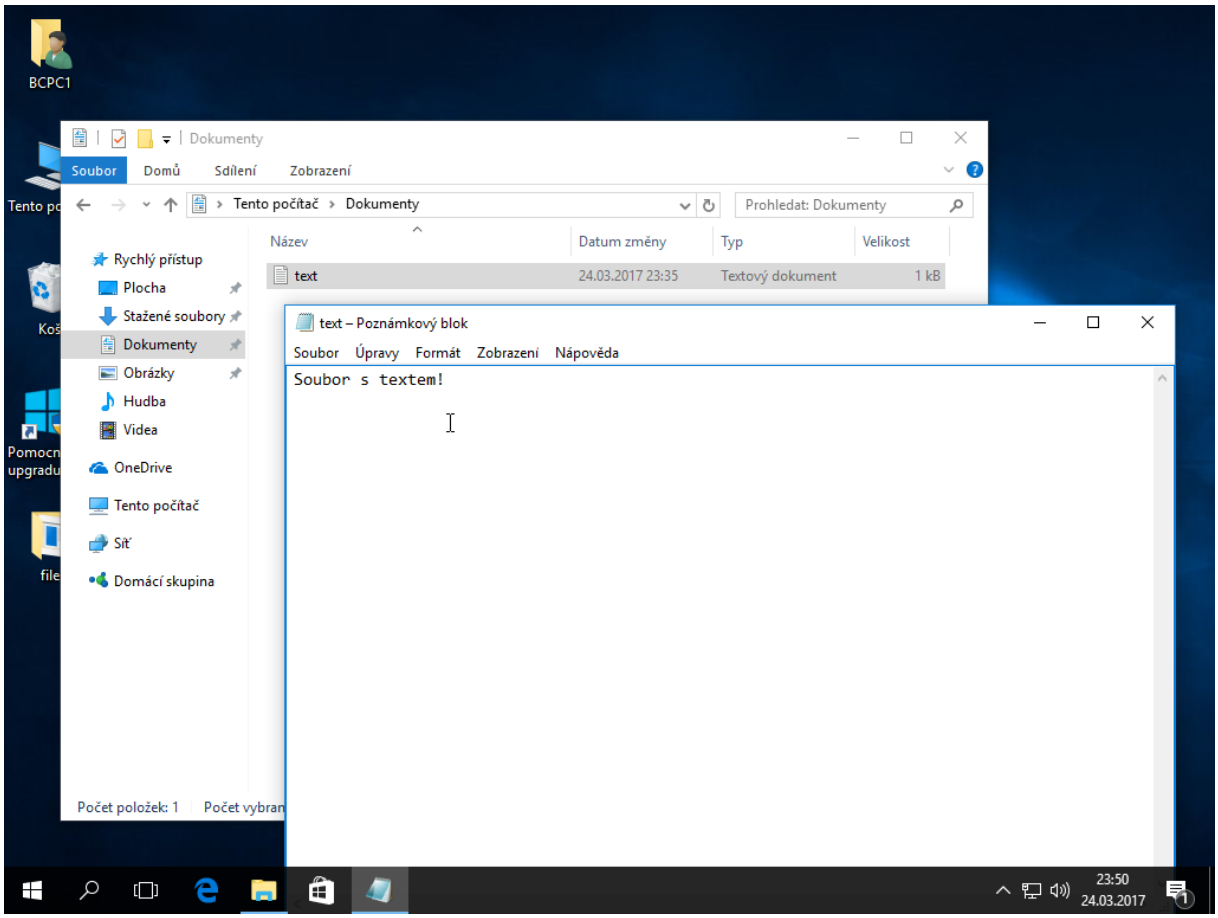
7.2.5 **Windows Defender**

Antivir Defender nepovažoval myfile.exe za jakoukoli hrozbu. I po manuální kontrole detekoval, že je vše v pořádku. Soubor myfile.exe proto mohl být spuštěn a mohlo být navázáno spojení. (Obrázek 10)



Obrázek 10 – Připojení meterpreter – scénář 1 – Windows Defender (zdroj – vlastní)

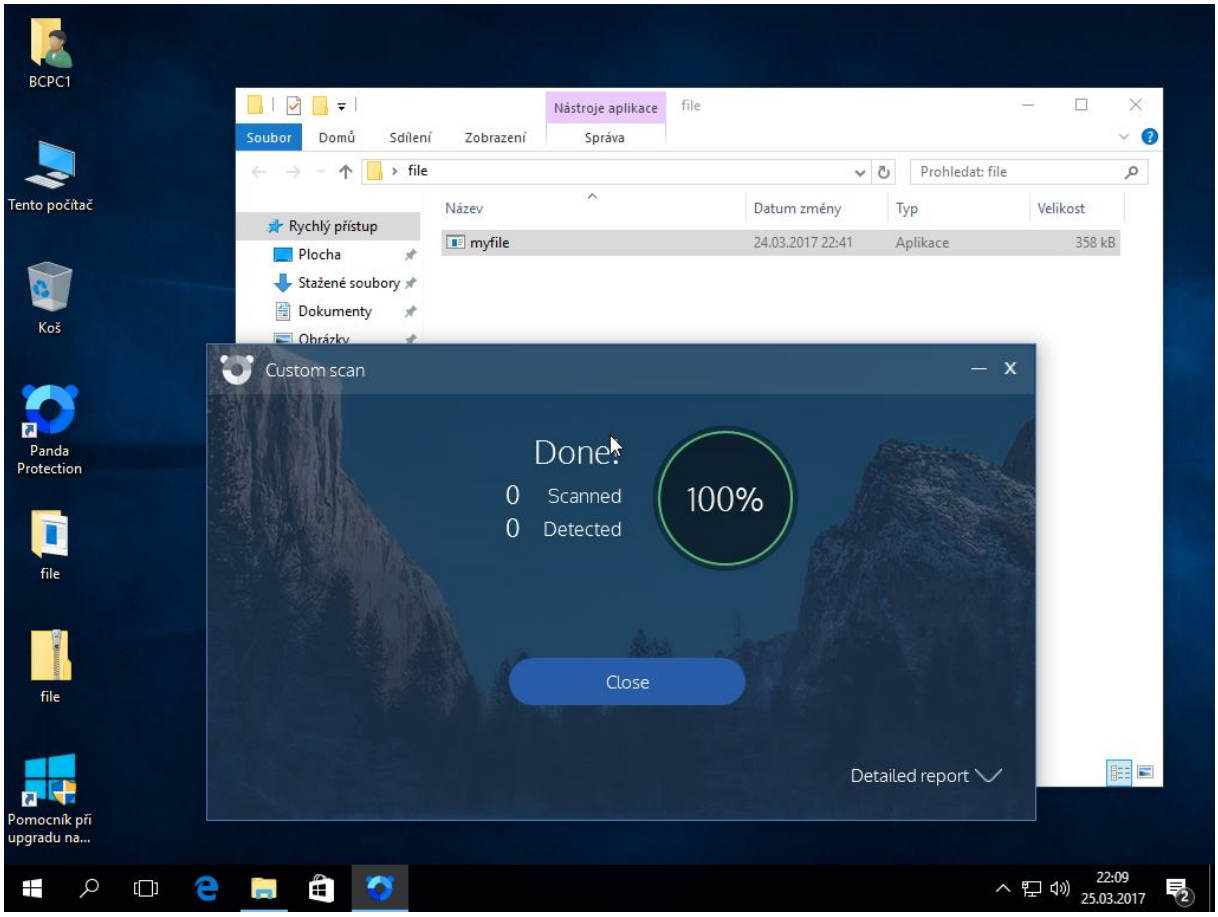
Pro ukázku byl na systém s Windows Defender nahrán textový soubor text.txt do Dukumentů (Obrázek 11), s textem uvnitř: „Soubor s textem!“. Realizováno příkazem *upload* v msfconsole.



Obrázek 11 – Soubor na cílovém systému – scénář 1 – Windows Defender (zdroj – vlastní)

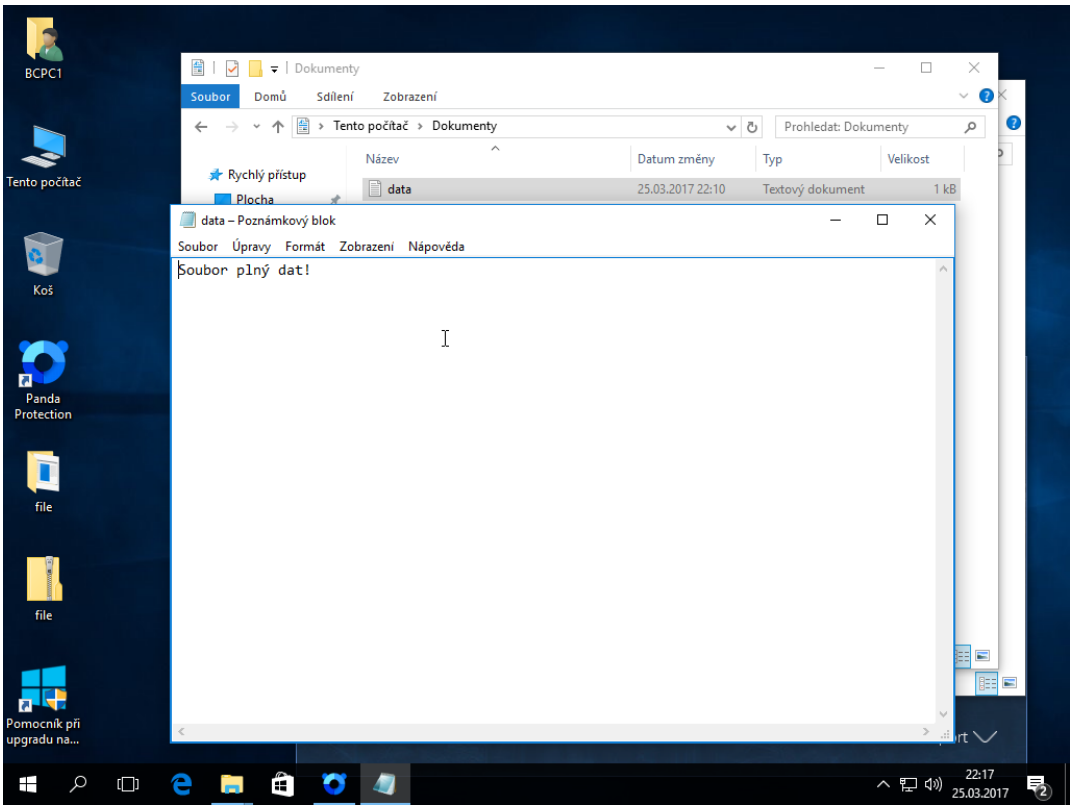
7.2.6 Panda

Antivir Panda vyšel z testu obdobně jako Windows Defender, protože myfile.exe opět považoval za neškodný soubor. (Obrázek 12)

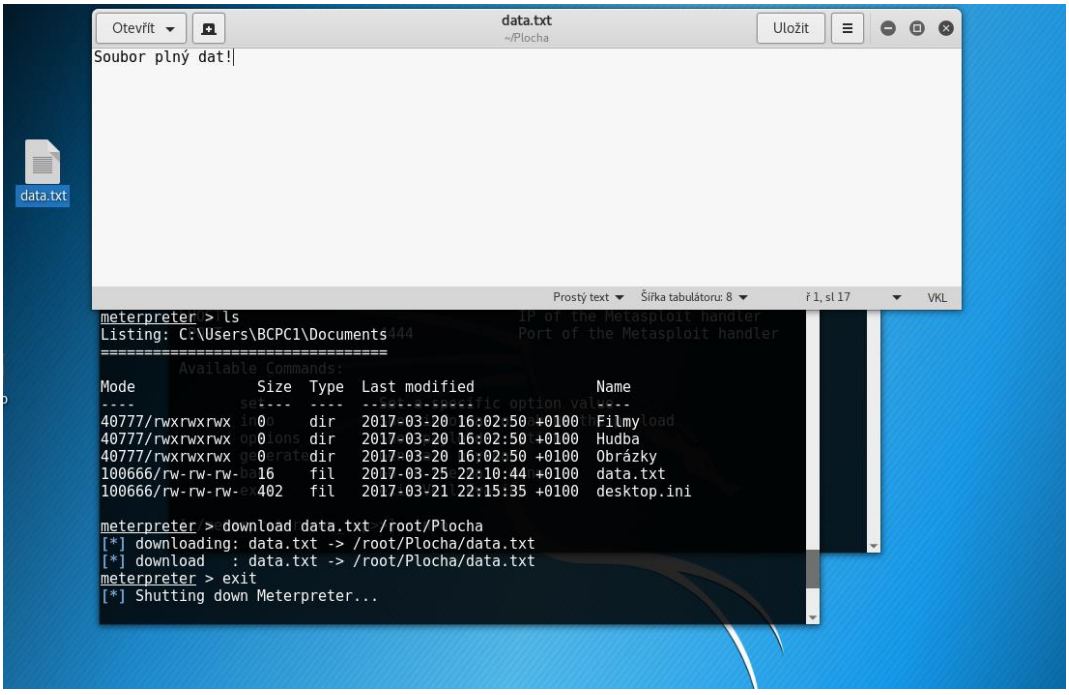


Obrázek 12 – Hrozba neidentifikována – scénář 1 – Panda (zdroj – vlastní)

Opět bylo jednoduše navázáno spojení s Windows 10 s antivirem Panda a pro ukázku byl stažen vytvořený soubor (na Windows) data.txt (Obrázek 13) na server, resp. Kali Linux (Obrázek 14).



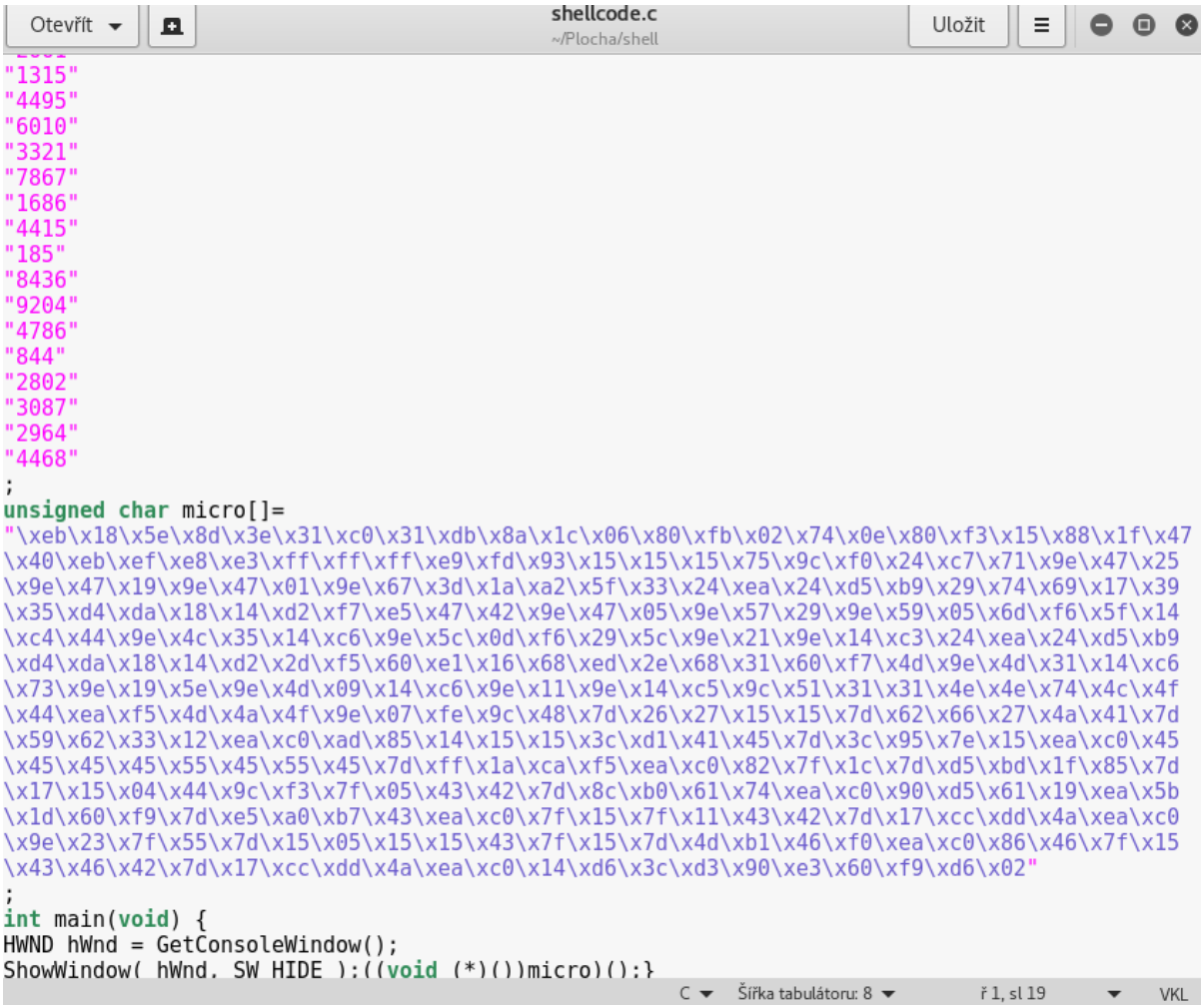
Obrázek 13 – Vytvořený soubor – scénář 1 – Panda (zdroj – vlastní)



Obrázek 14 – Stažený soubor – scénář 1 – Panda (zdroj – vlastní)

7.3 Scénář 2

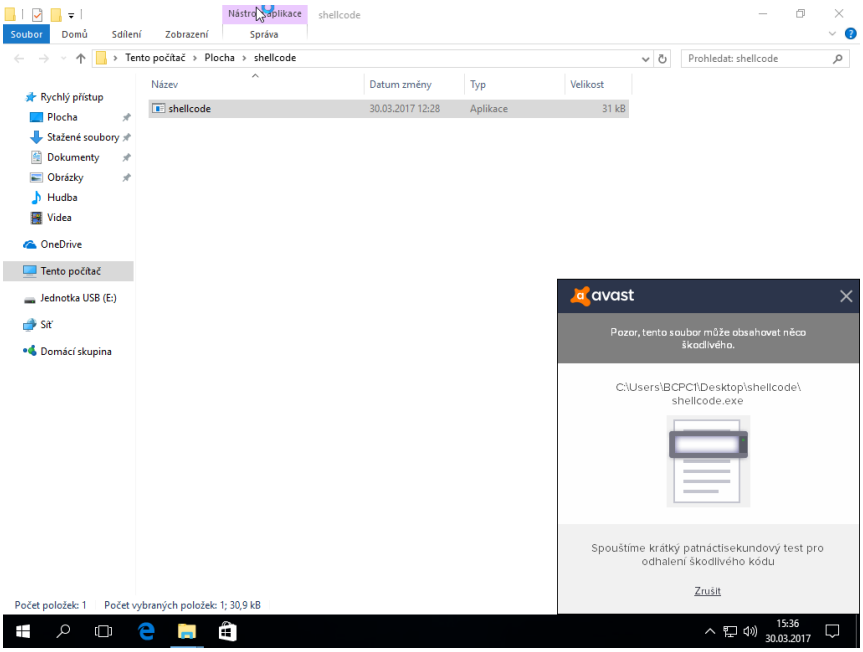
Ve druhém scénáři byl použit Veil–Ordnance k vygenerování shellcode (Obrázek 15). Vygenerovaný shellcode byl přidán do kódu napsaného v jazyce C a poté zkompilován do spustitelného souboru pro platformy Windows. Opět se jedná o spojení klient–server (řízené pomocí frameworku metasploit, resp. příkazovým řádkem meterpreter), soubor byl pojmenován jako shellcode.exe (na cílové systémy byl opět přenášen v komprimované formě zip).



Obrázek 15 – Ukázka kódu s shellcode v jazyce C (zdroj – vlastní)

7.3.1 Avast

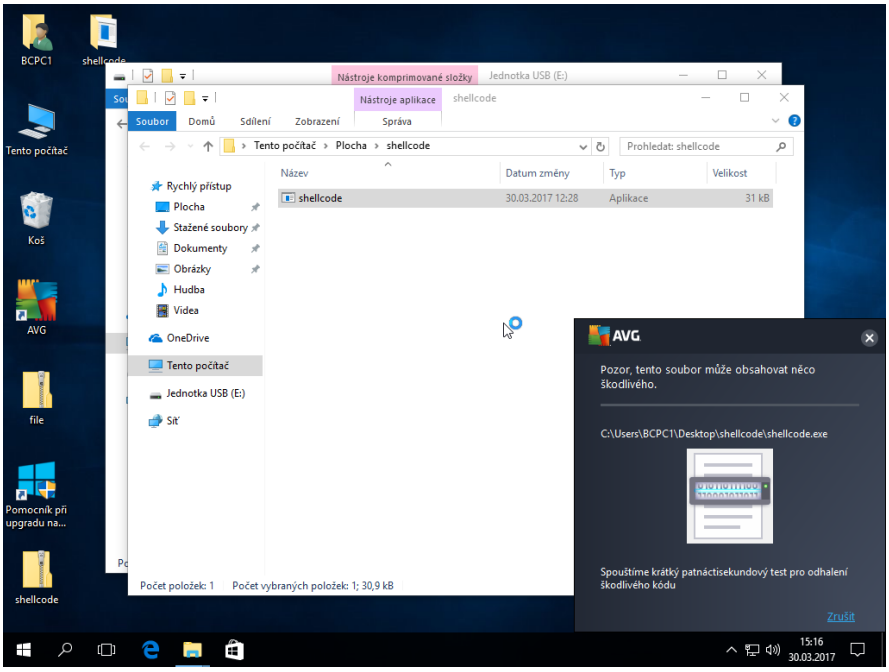
Antivir Avast neměl opět nejmenší problém s tím detekovat shellcode.exe jako hrozící nebezpečí, ale tentokrát umožnil uživateli se pokusit spustit shellcode.exe. (Obrázek 16)



Obrázek 16 – Detekce hrozby při spuštění – scénář 2 – Avast (zdroj – vlastní)

7.3.2 AVG

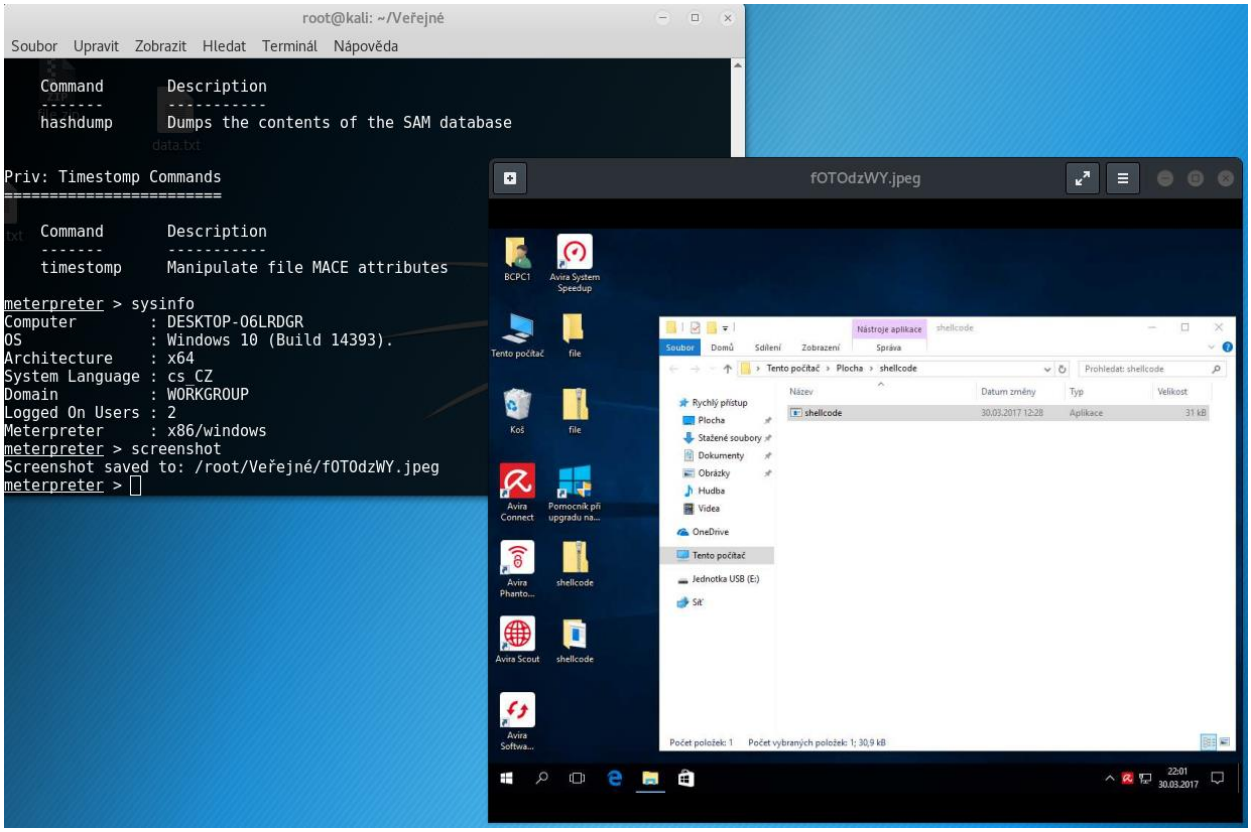
AVG se zachovalo úplně stejně jako Avast, zobrazilo i stejné dialogové okno. Projevilo se zde sloučení těchto antivirů do jedné společnosti. (Obrázek 17)



Obrázek 17 – Detekce hrozby při spuštění – scénář 2 – AVG (zdroj – vlastní)

7.3.3 Avira

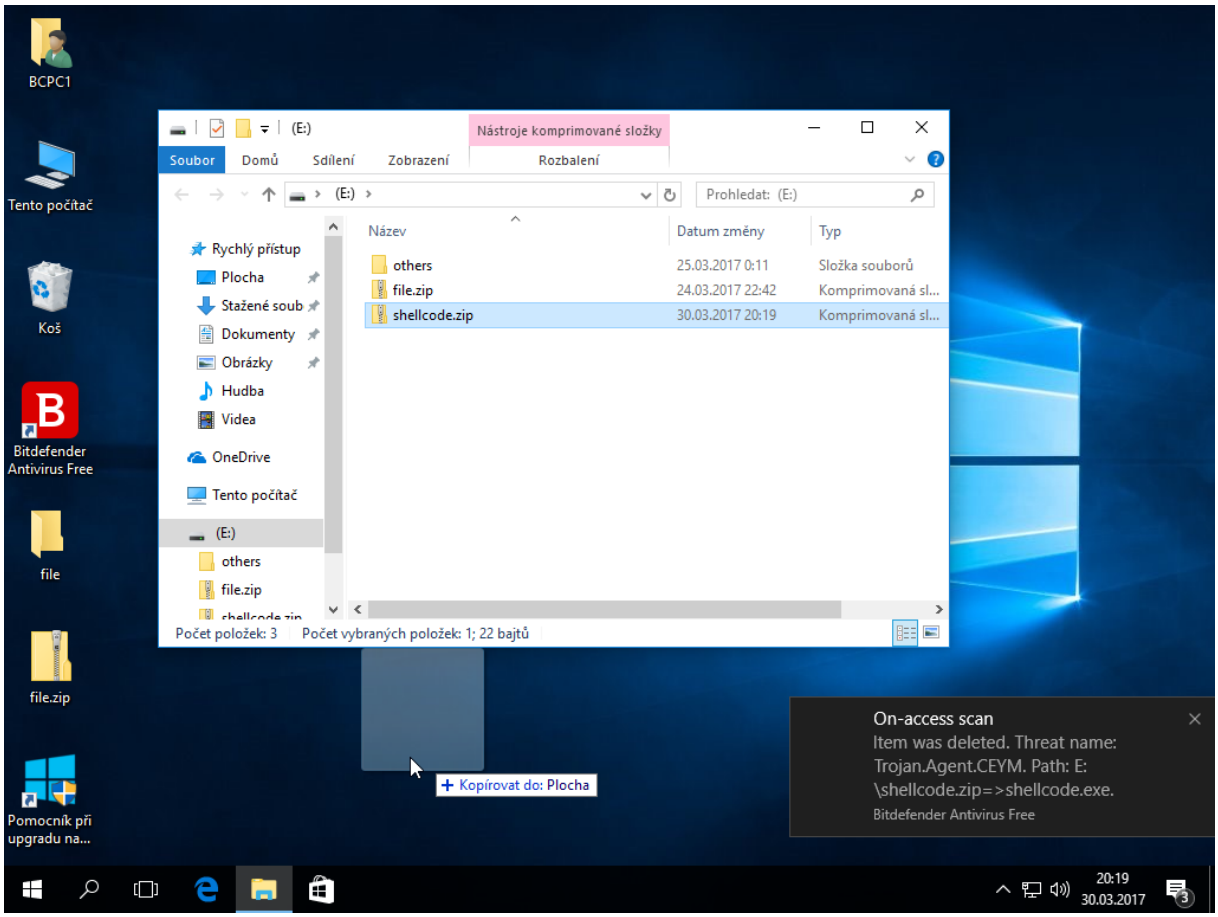
Antivir Avira v tomto testu neuspěl a dovolil uživateli bez problémů spustit shellcode.exe a navázat tak spojení s útočníkem. Pro ukázku, že došlo ke spojení, bylo využito funkcí meterpreteru a to konkrétně: zobrazení informací o ovládnutém zařízení + vytvoření screenshotu obrazovky ovládnutého zařízení. (Obrázek 18)



Obrázek 18 – Ovládnuté zařízení – scénář 2 – Avira (zdroj – vlastní)

7.3.4 Bitdefender

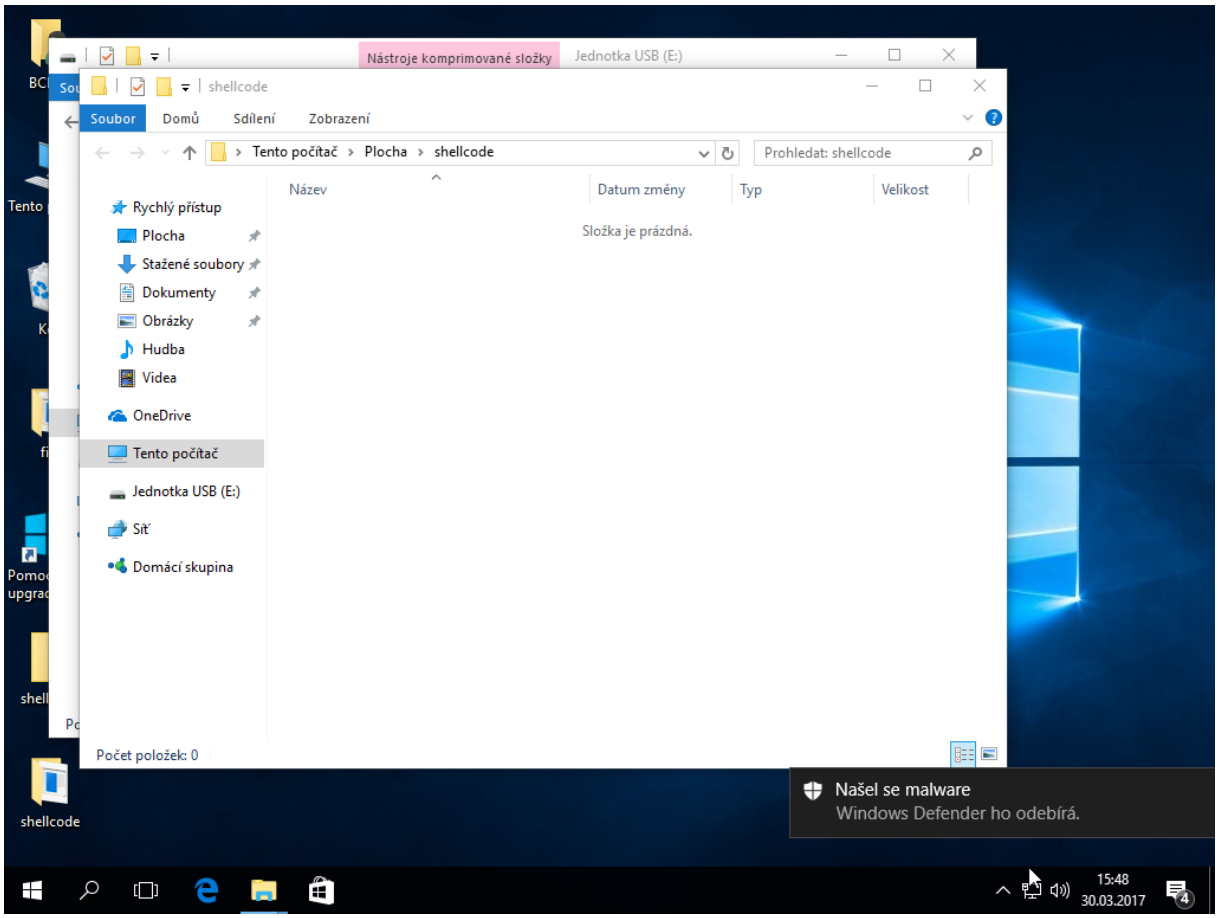
Antivir Bitdefender neměl s identifikováním hrozby žádný problém, stejně jako většina ostatních antivirů, s tím rozdílem, že Bitdefender dokázal hrozbu identifikovat již při přenášení zip souboru z flash disku do cílového PC. (Obrázek 19)



Obrázek 19 – Detekce hrozby při kopírování – scénář 2 – Bitdefender (zdroj – vlastní)

7.3.5 Windows Defender

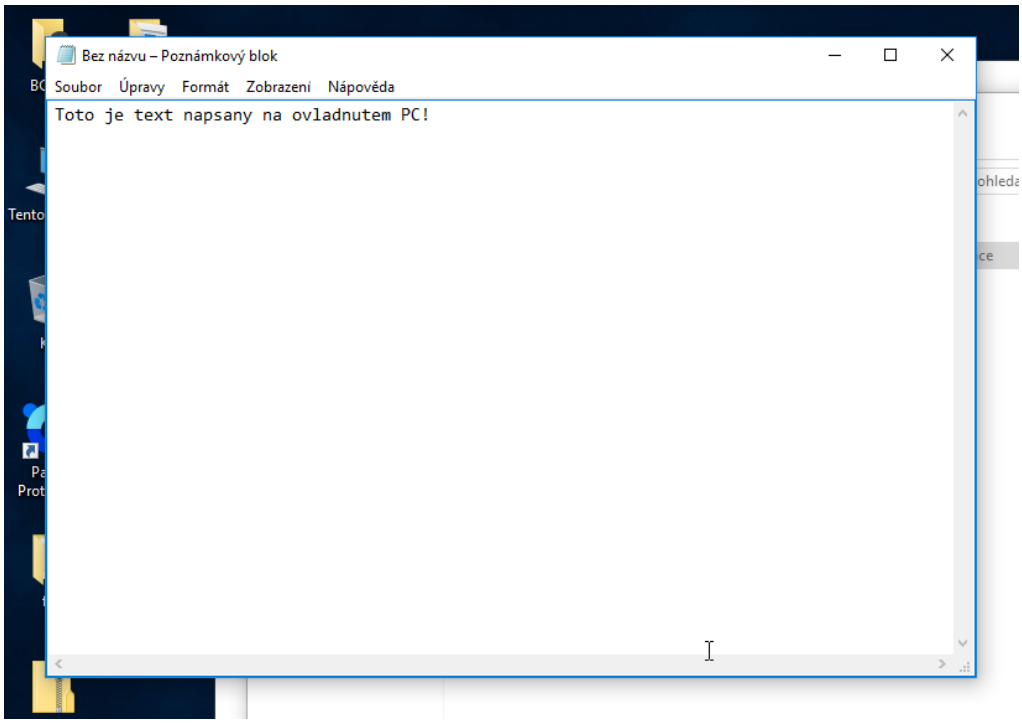
I Windows Defender si počínal velice dobře, protože hrozbu identifikoval okamžitě po extrahování zip souboru. (Obrázek 20)



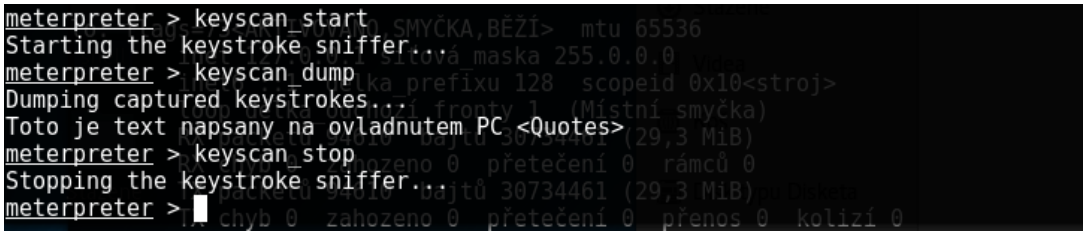
Obrázek 20 – Detekce hrozby po extrahování – scénář 2 – Windows Defender (zdroj – vlastní)

7.3.6 Panda

Antivir Panda neuspěl podobně jako antivir Avira. Nijak uživateli nebránil při spouštění souboru shellcode.exe a pro ukázku zmocnění se systému s antivirem Panda byla využita funkce meterpreteru, skenování klávesnice ovládnutého systému. (Obrázek 21, Obrázek 22)



Obrázek 21 – Text psaný na ovládnutém zařízení – scénář 2 – Panda (zdroj – vlastní)



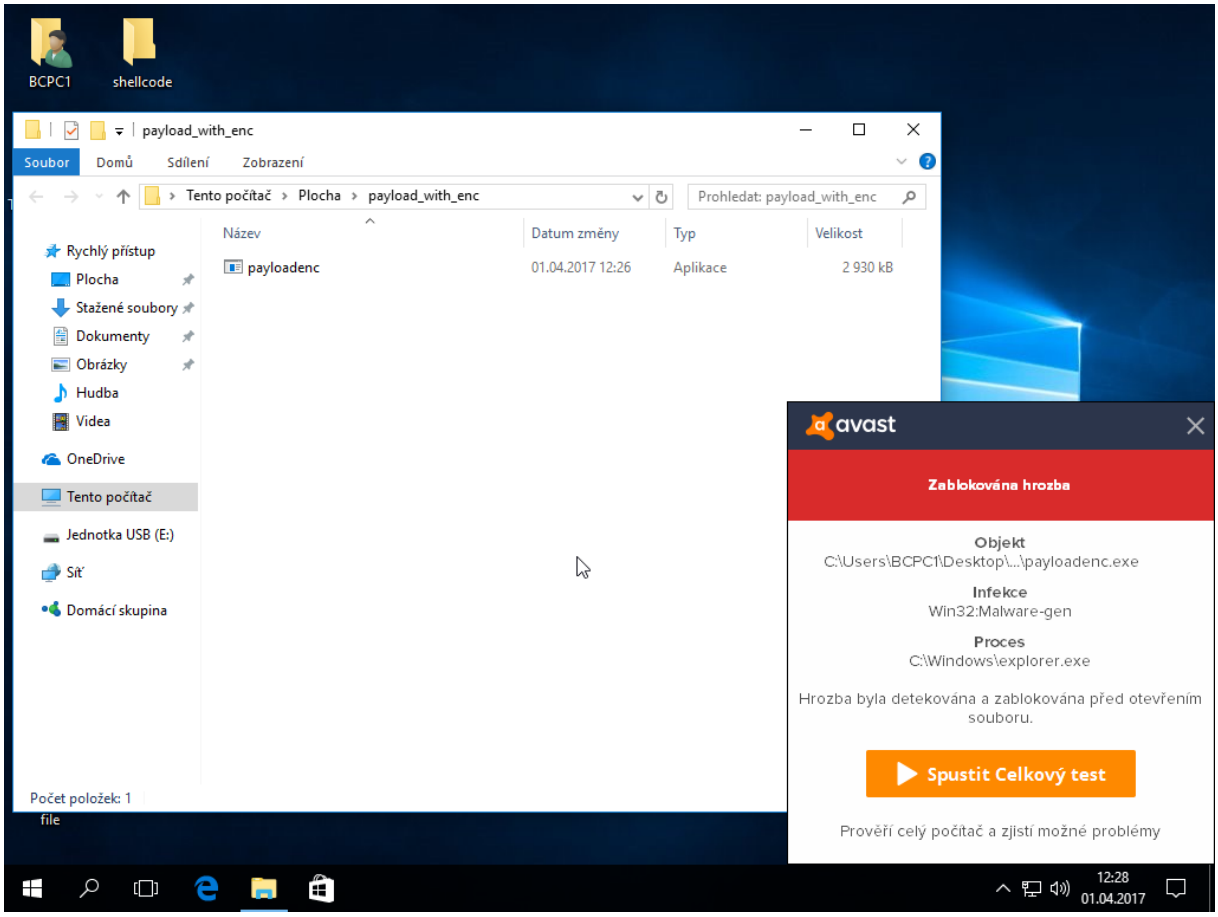
Obrázek 22 – Ovládnuté zařízení, keyscan – scénář 2 – Panda (zdroj – vlastní)

7.4 Scénář 3

Scénář 3 opět využil možností frameworku Veil–Evasion, který vygeneroval payload (napsaný v jazyce python) zašifrovaný algoritmem AES. Veil–Evasion z tohoto payloadu vytvořil také spustitelný soubor exe, který byl pojmenován payloadenc.exe. Přenos na jednotlivá zařízení probíhal v zazipované formě a spojení bylo řízeno opět frameworkem Metasploit.

7.4.1 Avast

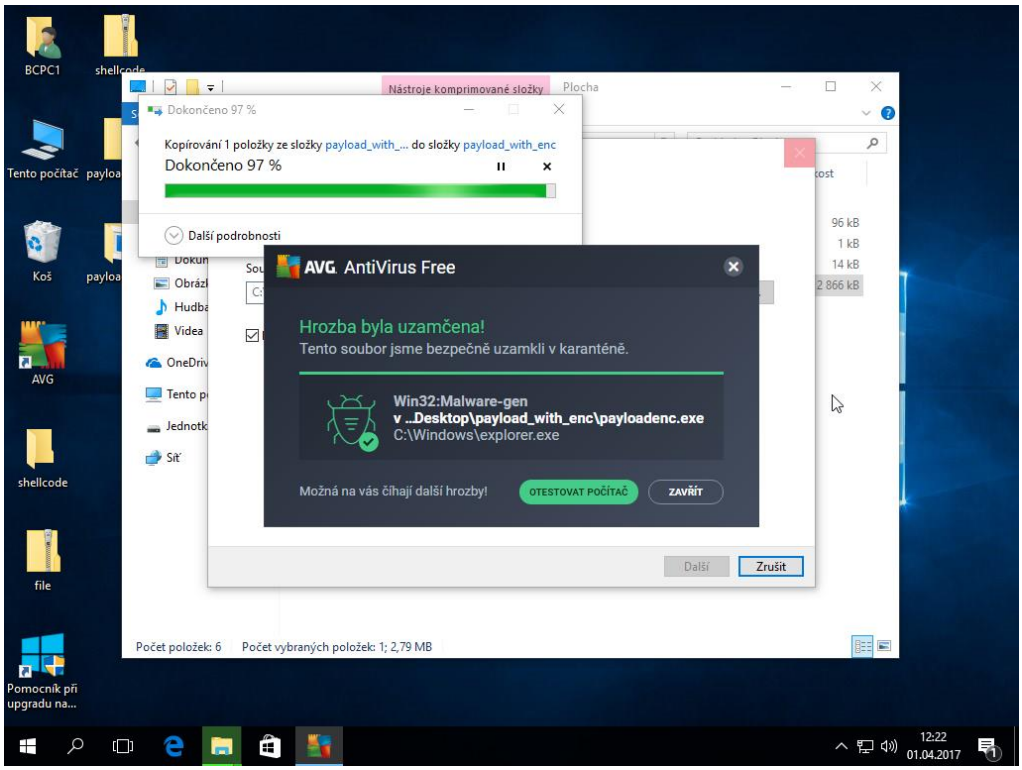
Antivir Avast zareagoval na soubor payloadenc.exe negativně a tento soubor odmítl uživateli používat jeho zablokováním. Vše okamžitě po extrahování zazipovaného adresáře. (Obrázek 23)



Obrázek 23 – Detekce hrozby – scénář 3 – Avast (zdroj – vlastní)

7.4.2 AVG

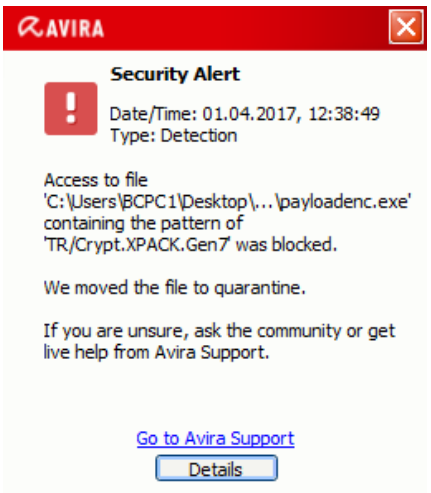
AVG podobně jako Avast neumožnil uživateli jakoukoli interakci se souborem payloadenc.exe a tento soubor umístil do karantény již při extrahování. (Obrázek 24)



Obrázek 24 – Detekce hrozby – scénář 3 – AVG (zdroj – vlastní)

7.4.3 Avira

Antivir Avira označil také soubor payloadenc.exe jako hrozbu a po extrahování jej přesunul do karantény. (Obrázek 25)



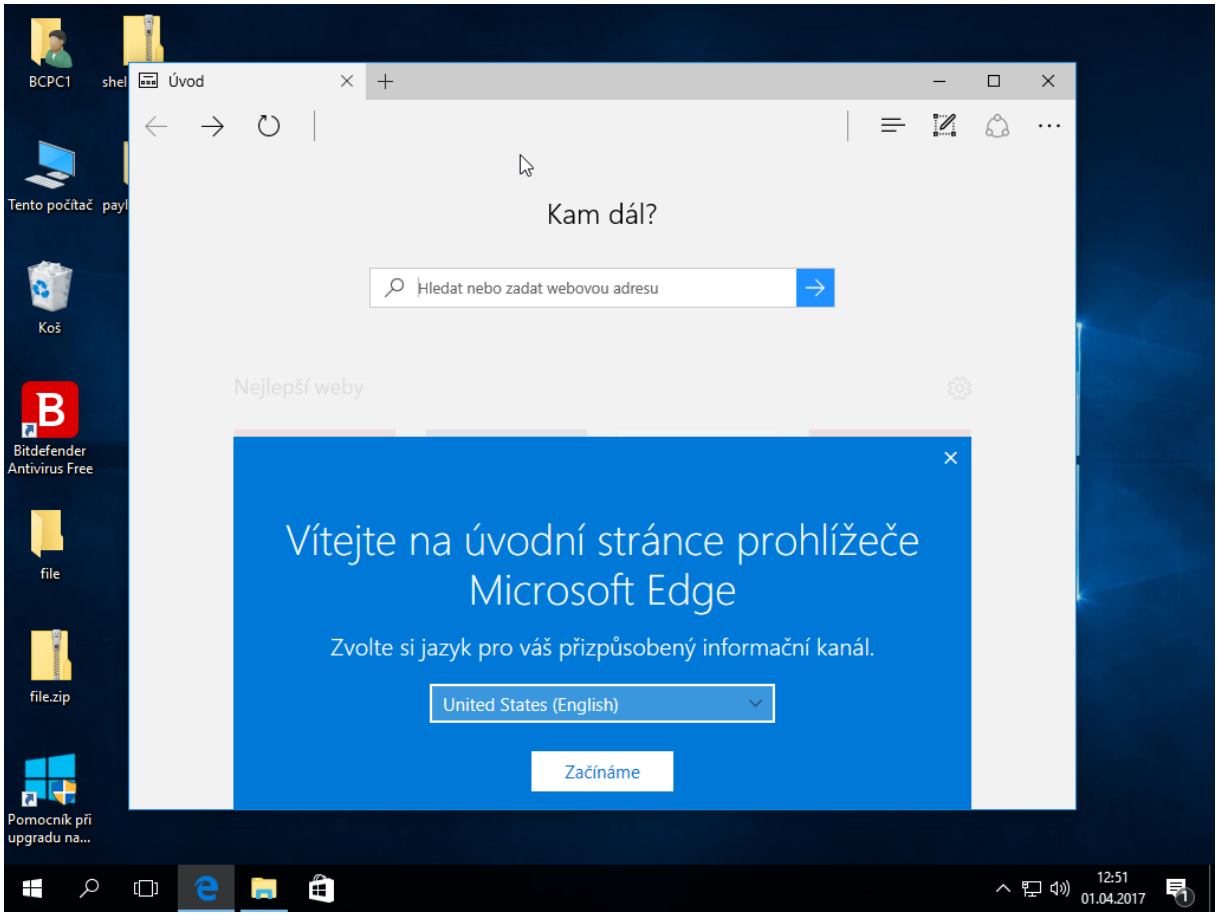
Obrázek 25 – Detekce hrozby – scénář 3 – Avira (zdroj – vlastní)

7.4.4 Bitdefender

Antivir Bitdefender v testu neobstál a soubor payloadenc.exe nechal bez problému uživatele spustit a tím mohlo být navázáno spojení mezi ovládnutým PC a meterpreterem. Pro ukázku byl na ovládnutém PC spuštěn webový prohlížeč Microsoft Edge, jež byl následně v meterpreteru zobrazen mezi běžícími procesy ovládnutého PC. (Obrázek 26, Obrázek 27)

```
4584  2700  MicrosoftEdgeCP.exe          x64   1       DESKTOP-O6LRDGR\BCPC1  C:\Windows\SystemApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdgeCP.exe
4756  896   CompatTelRunner.exe
5304  620   TrustedInstaller.exe
5340  708   smartscreen.exe              x64   1       DESKTOP-O6LRDGR\BCPC1  C:\Windows\System32\smartscreen.exe
5416  1820  OneDrive.exe                 x86   1       DESKTOP-O6LRDGR\BCPC1  C:\Users\BCPC1\AppData\Local\Microsoft\OneDrive\OneDrive.exe
5576  708   InstallAgent.exe             x64   1       DESKTOP-O6LRDGR\BCPC1  C:\Windows\System32\InstallAgent.exe
5680  904   WUDFHost.exe
5708  1312  audiodg.exe                  x64   0
5720  4756  conhost.exe
5860  708   ApplicationFrameHost.exe     x64   1       DESKTOP-O6LRDGR\BCPC1  C:\Windows\System32\ApplicationFrameHost.exe
6048  620   svchost.exe
6204  1820  payloadenc.exe               x86   1       DESKTOP-O6LRDGR\BCPC1  C:\Users\BCPC1\Desktop\payload_with_enc\payloadenc.exe
6220  6204  payloadenc.exe               x86   1       DESKTOP-O6LRDGR\BCPC1  C:\Users\BCPC1\Desktop\payload_with_enc\payloadenc.exe
6504  708   WmiPrvSE.exe
6644  708   InstallAgentUserBroker.exe   x64   1       DESKTOP-O6LRDGR\BCPC1  C:\Windows\System32\InstallAgentUserBroker.exe
6692  620   svchost.exe
6816  2700  MicrosoftEdgeCP.exe          x64   1       DESKTOP-O6LRDGR\BCPC1  C:\Windows\SystemApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdgeCP.exe
6856  708   WmiPrvSE.exe
6988  1436  downloader.exe
7016  708   MicrosoftEdge.exe            x64   1       DESKTOP-O6LRDGR\BCPC1  C:\Windows\SystemApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdge.exe
```

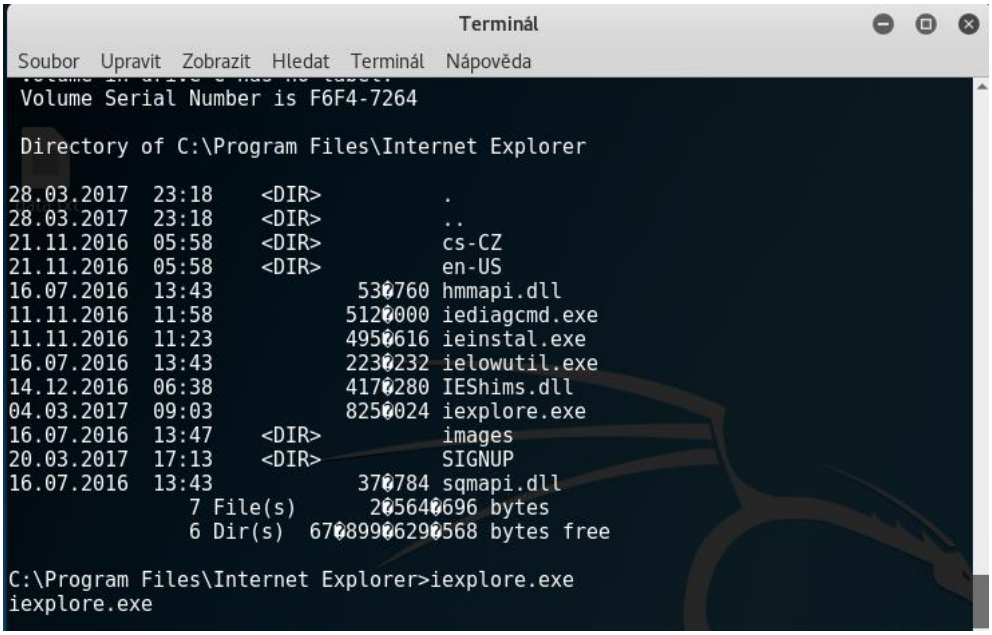
Obrázek 26 – Meterpreter, příkaz ps – scénář 3 – Bitdefender (zdroj – vlastní)



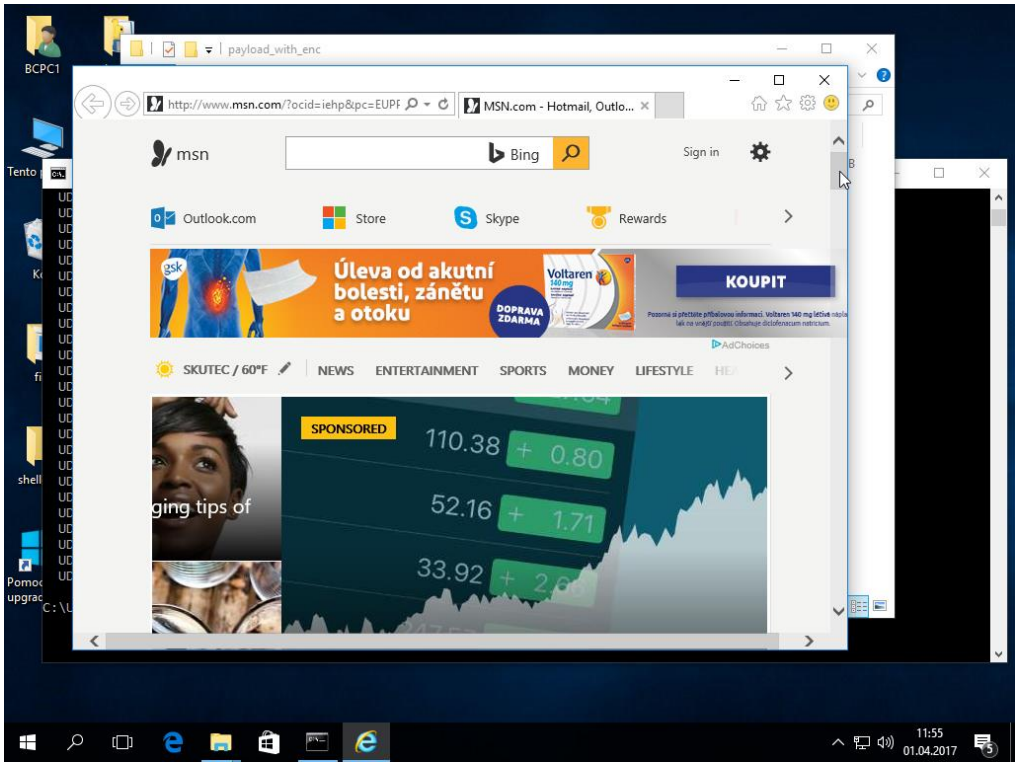
Obrázek 27 – Spuštěný Microsoft Edge na ovládnutém PC – scénář 3 – Bitdefender (zdroj – vlastní)

7.4.5 Windows Defender

Windows Defender považoval soubor payloadenc.exe taktéž za neškodný, a proto ho uživateli dovolil bez obav pustit. Pro ukázku byla využita jiná funkce meterpreteru, a to spuštění příkazové řádky na ovládnutém systému s následným spuštěním webového prohlížeče Internet Explorer z této řádky. (Obrázek 28, Obrázek 29)



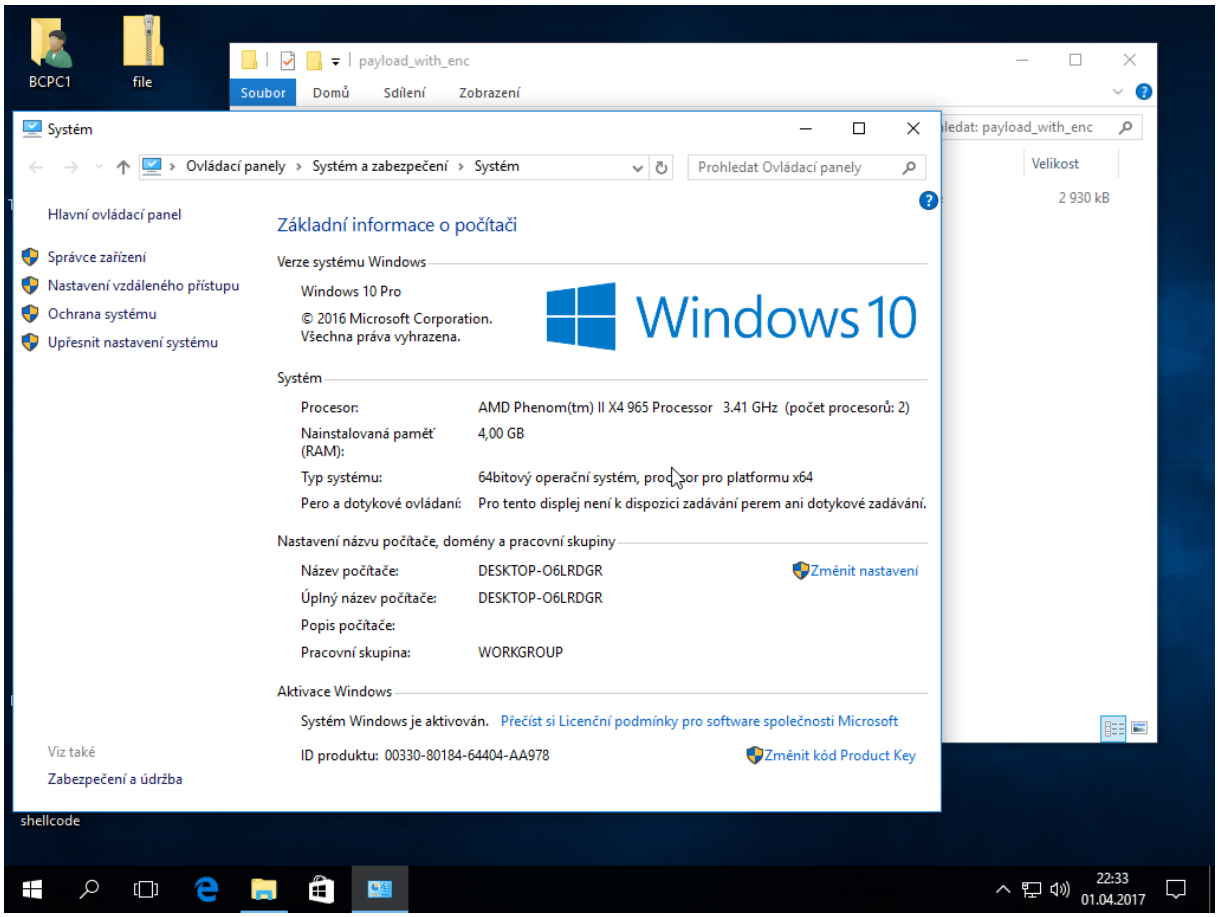
Obrázek 28 – Meterpreter, spuštění IE – scénář 3 – Windows Defender (zdroj – vlastní)



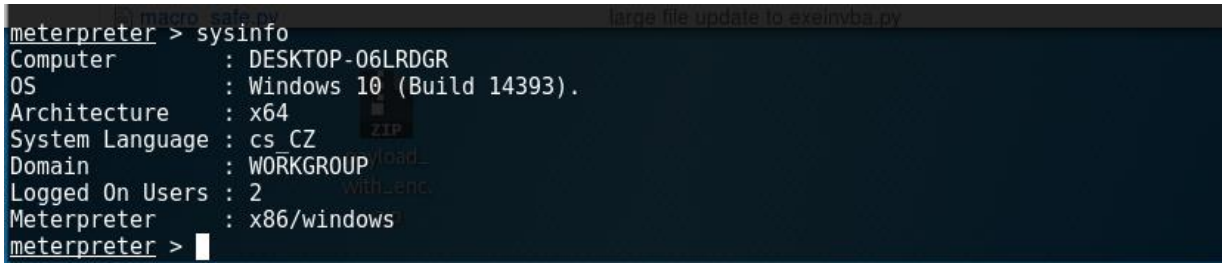
Obrázek 29 – Spuštěný IE na ovládnutém PC – scénář 3 – Windows Defender (zdroj – vlastní)

7.4.6 Panda

Antivir Panda podobně jako antiviry Windows Defender a Bitdefender v testu neobstál a taktéž uživateli povolil spustit soubor pyloadenc.exe. Pro ukázku zobrazeny z meterpreteru pouze základní systémové informace o ovládnutém PC. (Obrázek 30, Obrázek 31)



Obrázek 30 – Informace ovládnuté PC – scénář 3 – Panda (zdroj – vlastní)



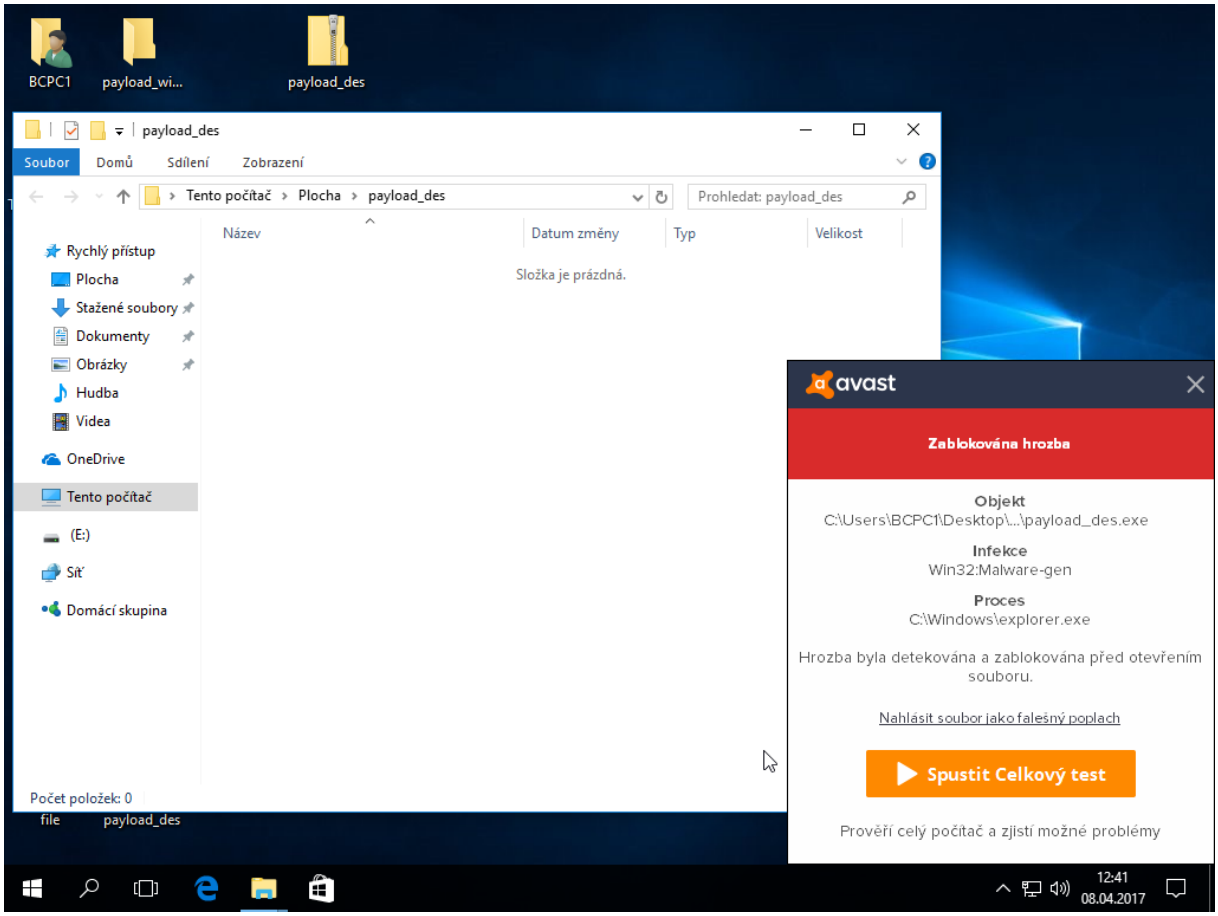
Obrázek 31 – Meterpreter, sysinfo – scénář 3 – Panda (zdroj – vlastní)

7.5 Scénář 4

Ve scénáři číslo 4 byt opět využit framework Veil–Evasion, konkrétně payload vytvořený v jazyce python s šifrováním DES proti odhalení. Zároveň byla využita funkce, kterou framework nabízí, a to migrace spuštěného procesu zkompilovaným souborem paylad_des.exe, do procesu rundll32.exe, resp. pod tímto procesem je spuštěna instance meterpreteru. Tato migrace se nastavuje pomocí parametru PrependMigrate na hodnotu true a používá se pro stabilitu spojení s ovládnutým systémem (často se stává, že obyčejný proces spuštěného souboru může být ukončen jiným procesem, kdežto rundll32 běží jako hostitelský proces). Komunikace opět probíhala za pomoci meterpreteru.

7.5.1 Avast

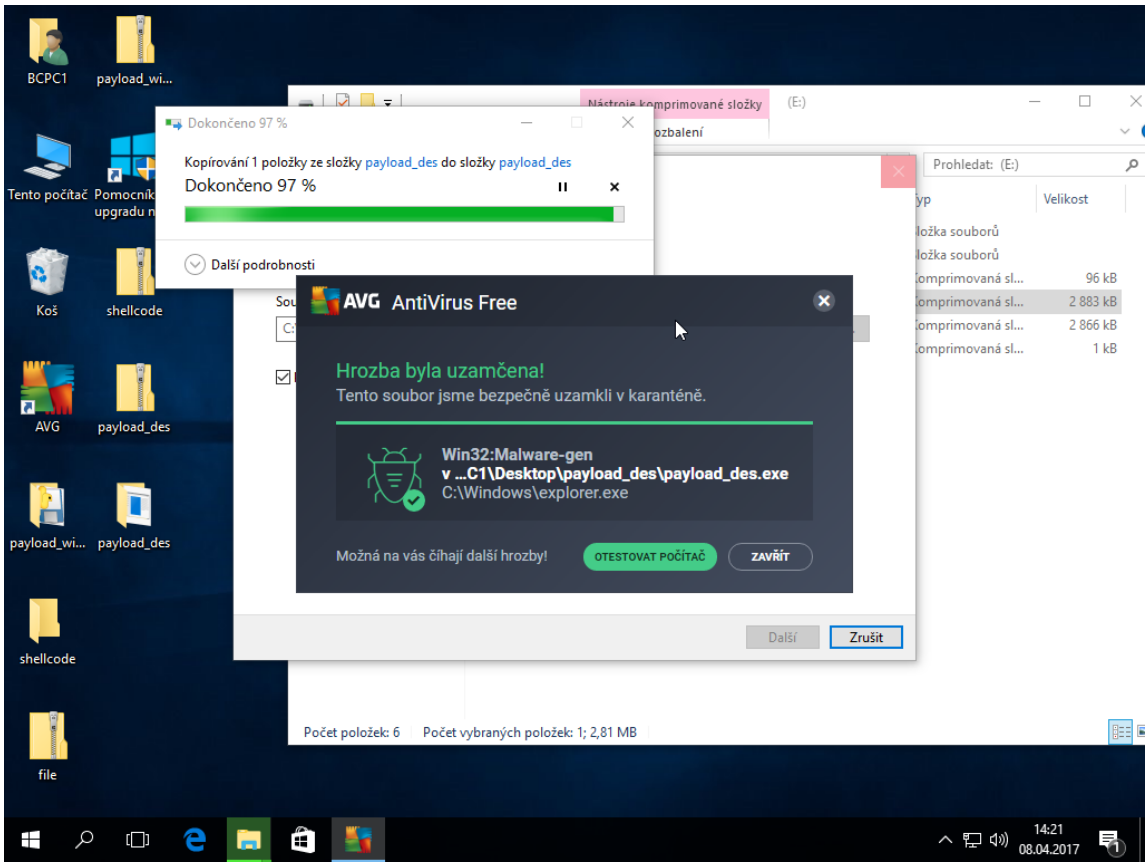
Avast jako první testovaný antivir odhalil soubor payload_des.exe okamžitě po extrahování zip adresáře, ve kterém se nacházel. Lze vidět na obrázku. (Obrázek 32)



Obrázek 32 – Detekce hrozby – scénář 4 – Avast (zdroj – vlastní)

7.5.2 AVG

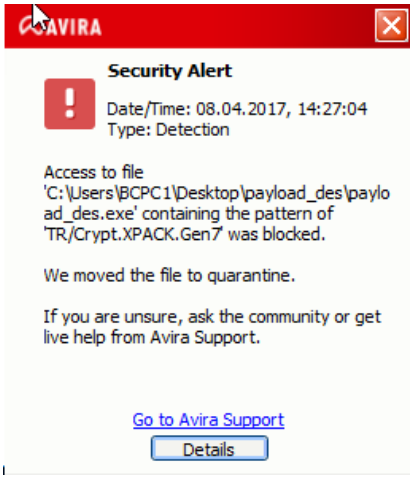
AVG si s tímto malware taktéž poradilo a detekovalo jej jako hrozbu hned po extrahování. (Obrázek 33)



Obrázek 33 – Detekce hrozby – scénář 4 – AVG (zdroj – vlastní)

7.5.3 Avira

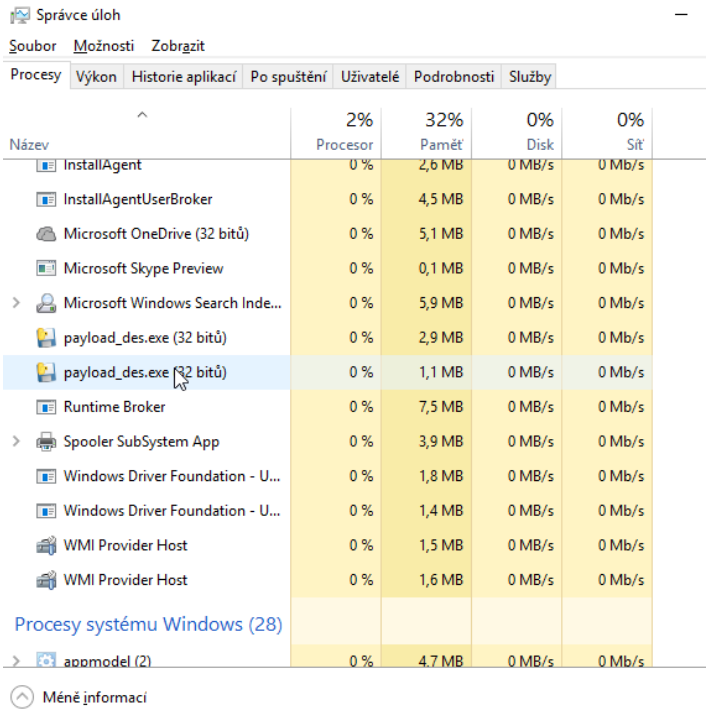
Obdobně jako Avast a AVG reagovala Avira a hrozbu bez problému detekovala. (Obrázek 34)



Obrázek 34 – Detekce hrozby – scénář 4 – Avira (zdroj – vlastní)

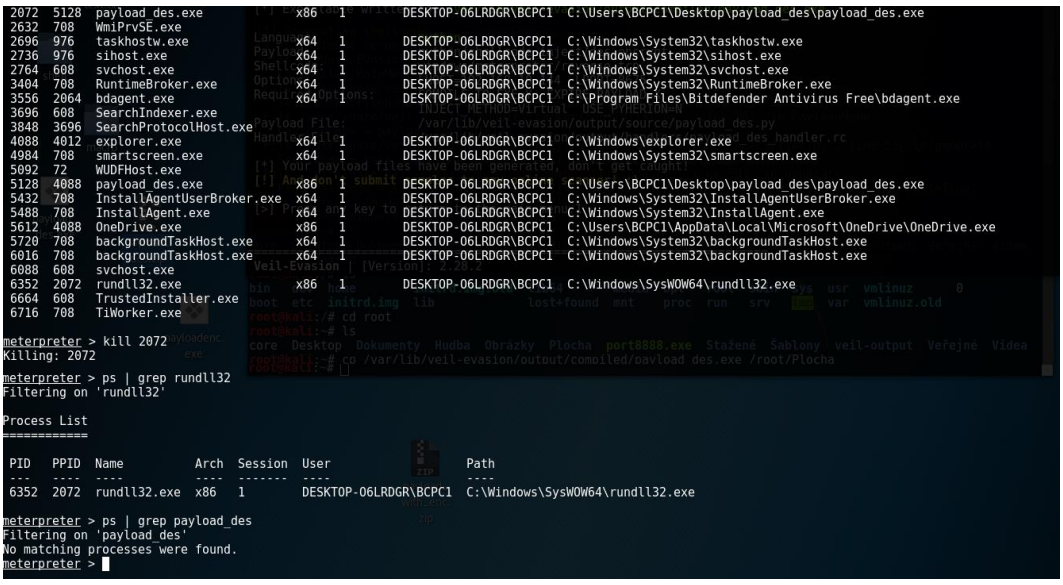
7.5.4 Bitdefender

Antivir Bitdefender neidentifikoval payload_des.exe za hrozbu a umožnil proto ovládnutí zařízení. (Obrázek 35)



Obrázek 35 – Spuštěné procesy ovládnutého zařízení – scénář 4 – Bitdefender (zdroj – vlastní)

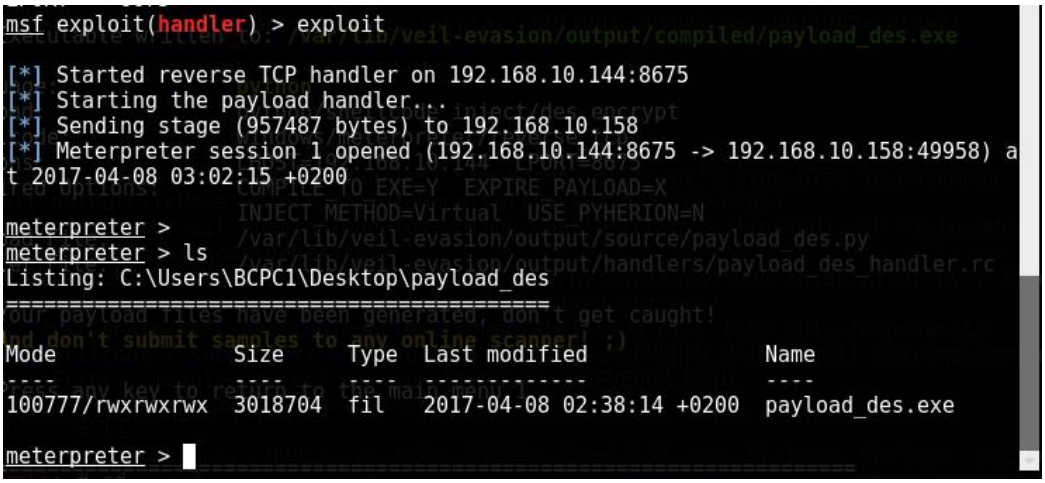
Na straně meterpreteru bylo vyzkoušeno ukončení procesu payload_des.exe, aniž by se přerušila komunikace. (Obrázek 36)



Obrázek 36 – Ukončení procesu (meterpreter) – scénář 4 – Bitdefender (zdroj – vlastní)

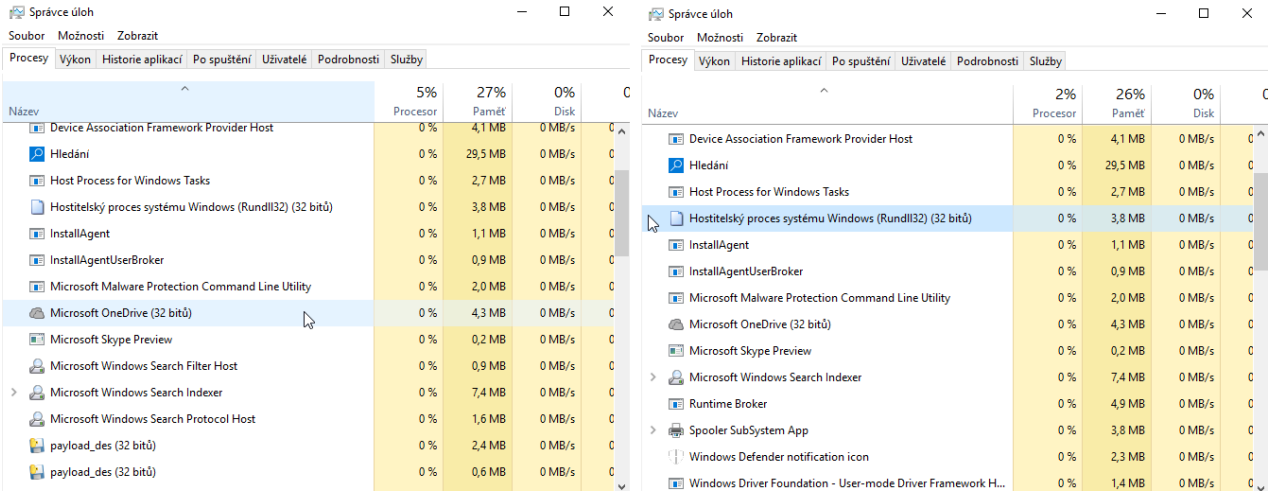
7.5.5 Windows Defender

Windows Defender hrozbu neidentifikoval, a tak mohlo dojít k navázání komunikace mezi ovládnutým PC a meterpreterem. (Obrázek 37)



Obrázek 37 – Ovládnuté zařízení (meterpreter) – scénář 4 – Windows Defender (zdroj – vlastní)

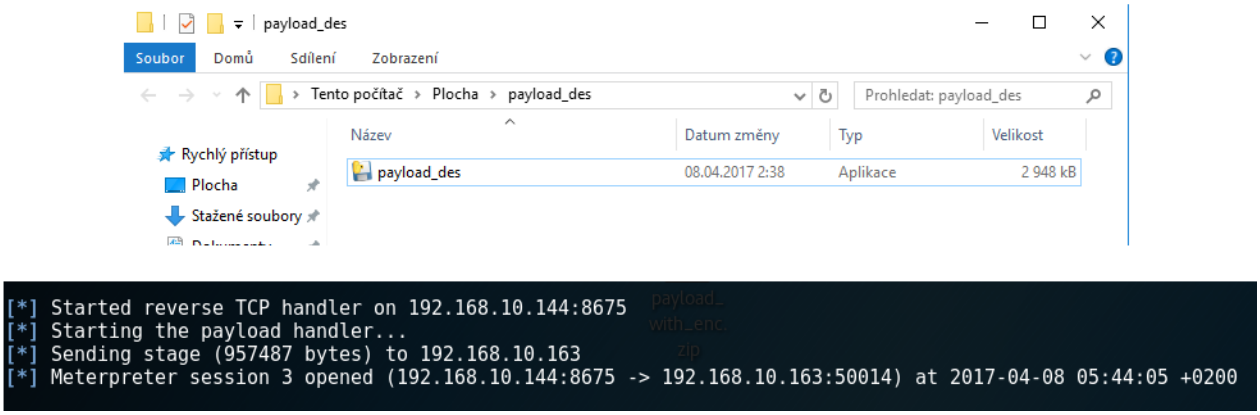
I po ukončení procesů payload_des.exe na ovládnutém zařízení (ručně) bylo spojení nepřerušeno, protože instance meterpreteru byla migrována do jiného procesu, který zůstal zachován, jak je vidět na obrázku (Obrázek 38).



Obrázek 38 – Procesy ovládnutého zařízení – scénář 4 – Windows Defender (zdroj – vlastní)

7.5.6 Panda

Poslední testovaný antivir Panda nedokázal odhalit možnou hrozbu, čímž umožnil ovládnout zařízení, na kterém byl nainstalován. (Obrázek 39)



Obrázek 39 – Ovládnutí zařízení – scénář 4 – Panda (zdroj – vlastní)

7.6 Scénář 5

V tomto scénáři byl vytvořen stejný payload jako ve scénáři 4, ale zašifrovaný algoritmem ARC. Doručený k uživateli ve formě zip, kde se ukrýval nikoli spustitelný soubor .exe, ale dávkový soubor .bat (pow_meter.bat).

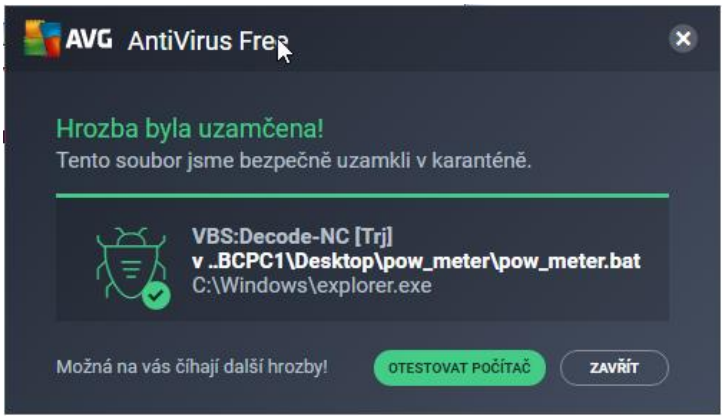
7.6.1 Avast

Avast opět neměl problém s detekcí souboru pow_meter.bat a označením, že se jedná o hrozbu. Tento soubor zablokoval. (Obrázek 40)

Obrázek 40 – Detekce hrozby – scénář 5 – Avast (zdroj – vlastní)

7.6.2 AVG

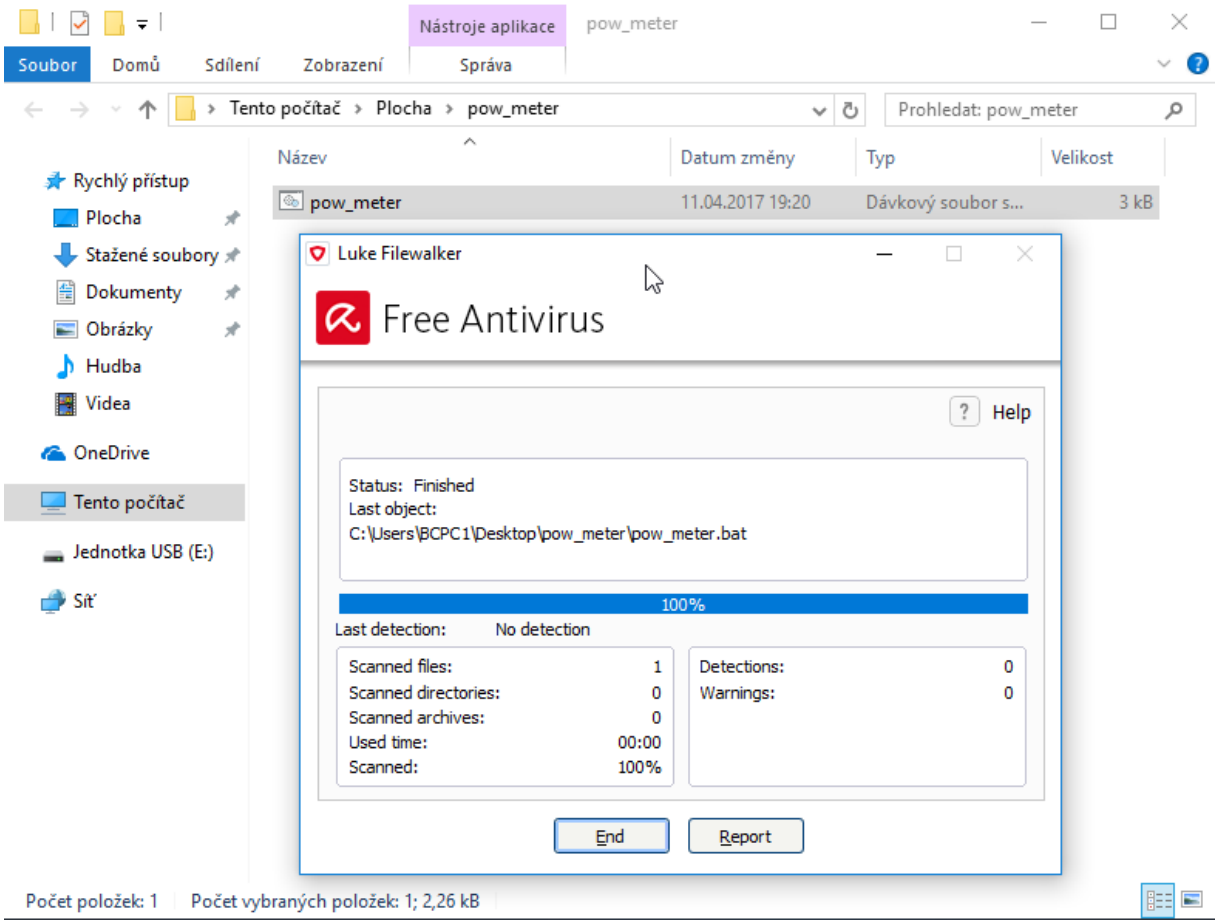
Podobně jako Avast i AVG detekovalo soubor pow_meter.bat jako hrozbu. (Obrázek 41)



Obrázek 41 – Detekce hrozby – scénář 5 – AVG (zdroj – vlastní)

7.6.3 Avira

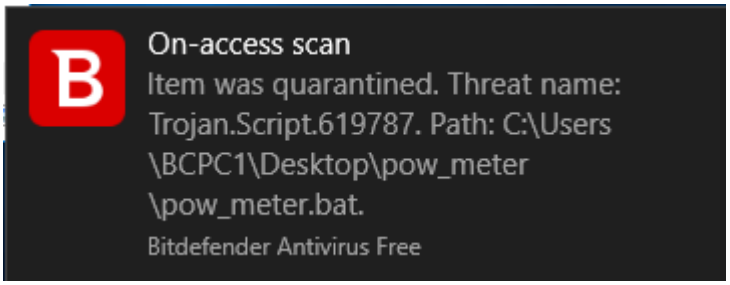
Avira byla prvním antivirem, který povolil uživateli pow_meter.bat spustit a umožnit tak ovládnout PC na němž byl antivir nainstalován. Ani po manuální kontrole nebyla identifikována hrozba. (Obrázek 42)



Obrázek 42 – Hrozba neidentifikována – scénář 5 – Avira (zdroj – vlastní)

7.6.4 Bitdefender

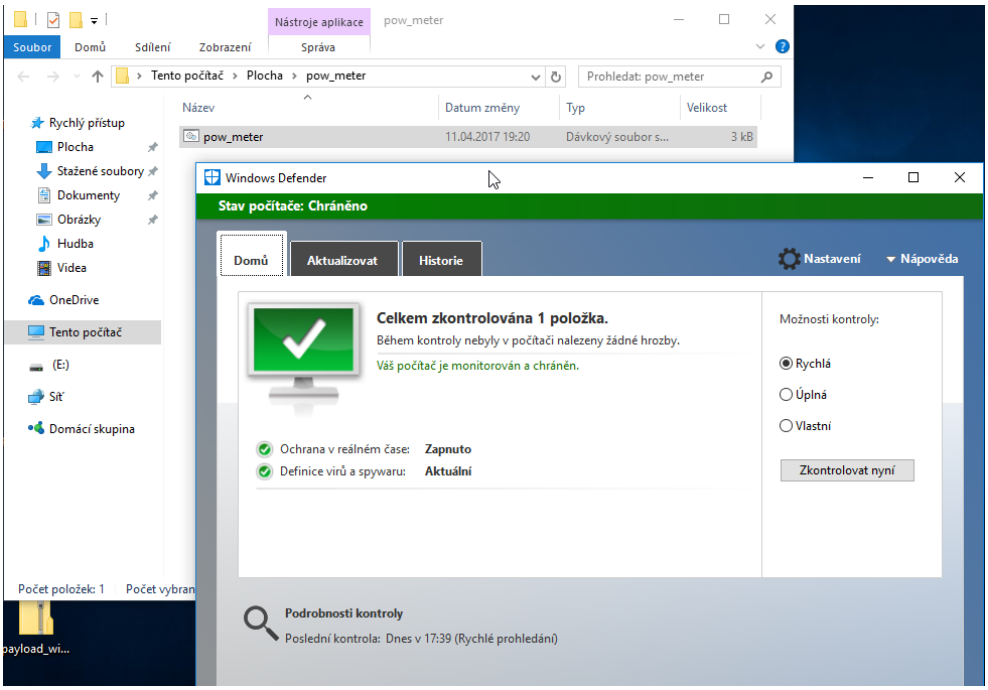
Bitdefender po extrahování označil soubor pow_meter.bat jako hrozbu a zablokoval jí. (Obrázek 43)



Obrázek 43 – Detekce hrozby – scénář 5 – Bitdefender (zdroj – vlastní)

7.6.5 Windows Defender

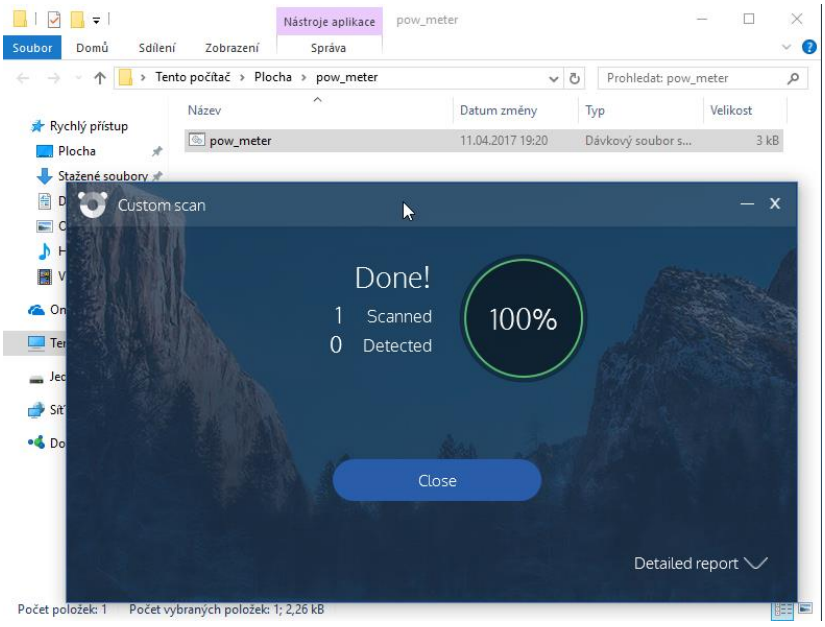
Windows Defender ani po manuální kontrole neoznačil pow_meter.bat jako hrozbu a umožnil uživateli spustit soubor. (Obrázek 44)



Obrázek 44 – Hrozba neidentifikována – scénář 5 – Windows Defender (zdroj – vlastní)

7.6.6 Panda

Antivir Panda podobně jako Windows Defender, Bitdefender, nebo i Avira soubor pow_meter.bat neoznačil jako hrozbu ani po manuální kontrole. (Obrázek 45)



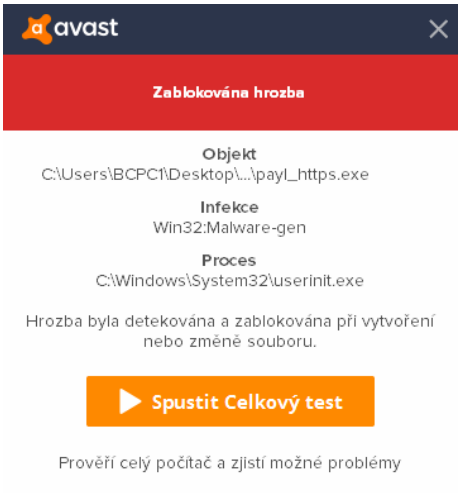
Obrázek 45 – Hrozba neidentifikována – scénář 5 – Panda (zdroj – vlastní)

7.7 Scénář 6

Ve scénáři byl vytvořen payload ve skriptovacím jazyce powershell (s užitím frameworku Veil–Evasion) a k uživateli byl doručen ve formě zip, kde se ukrýval spustitelný soubor payl_https.exe. V tomto scénáři byl k ovládnutí cílového zařízení využit protokol HTTPS, který na rozdíl od TCP (použito v předchozích scénářích) představuje zašifrovanou komunikaci, takže pro firewall se tváří mnohem bezpečněji, resp. důvěryhodněji.

7.7.1 Avast

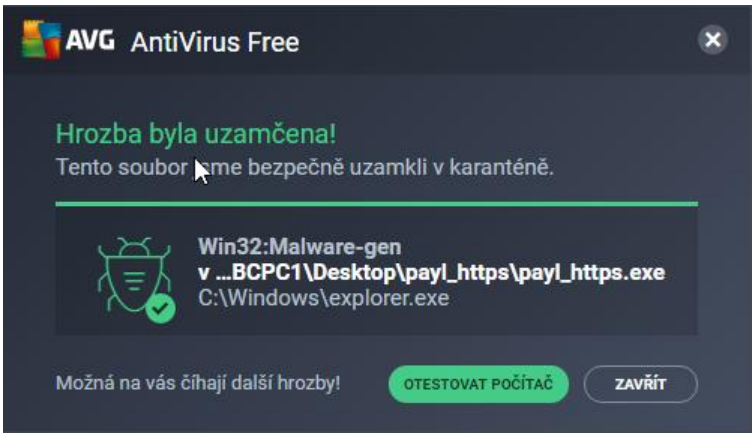
Ani protokol HTTPS nezměnil nic na tom, že antivir Avast detekoval soubor payl_https.exe jako hrozbu a neumožnil uživateli jej spustit. (Obrázek 46)



Obrázek 46 – Detekce hrozby – scénář 6 – Avast (zdroj – vlastní)

7.7.2 AVG

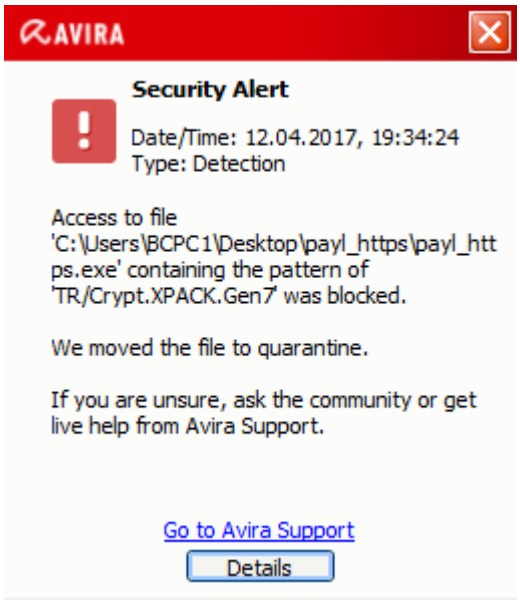
AVG opět jako Avast detekovalo hrozbu a neumožnilo uživateli pracovat se souborem payl_https.exe. (Obrázek 47)



Obrázek 47 – Detekce hrozby – scénář 6 – AVG (zdroj – vlastní)

7.7.3 Avira

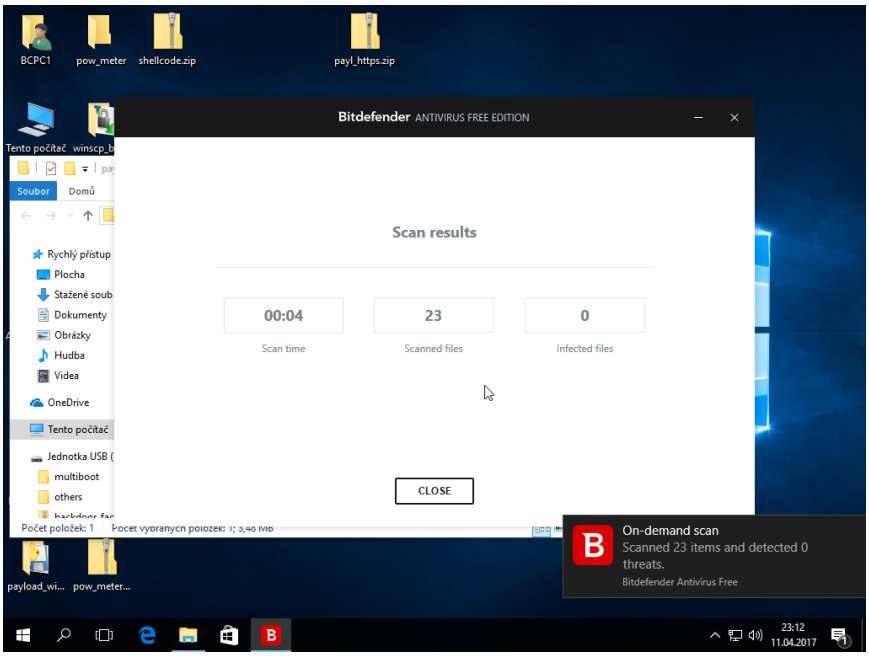
Antivir Avira nedovolil payl_https.exe spustit a soubor umístil do karantény, což oznámil uživateli. (Obrázek 48)



Obrázek 48 – Detekce hrozby – scénář 6 – Avira (zdroj – vlastní)

7.7.4 Bitdefender

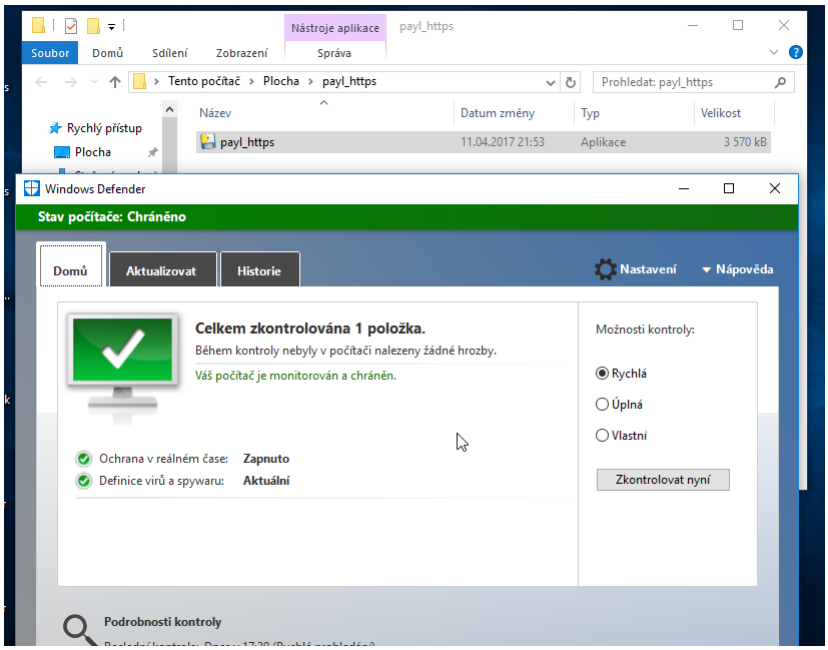
Antivir Bitdefender soubor payl_https.exe po manuální kontrole (do kontroly zahrnul více souborů) neidentifikoval tento soubor jako hrozbu. (Obrázek 49)



Obrázek 49 – Hrozba neidentifikována – scénář 6 – Bitdefender (zdroj – vlastní)

7.7.5 Windows Defender

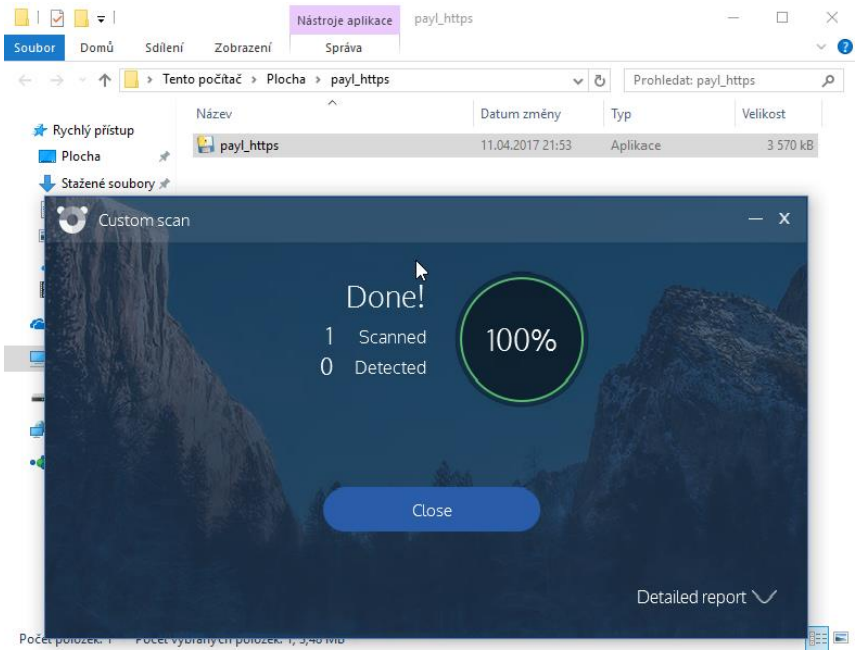
Windows Defender neidentifikoval payl_https.exe jako hrozbu, a to ani po manuální kontrole. (Obrázek 50)



Obrázek 50 – Hrozba neidentifikována – scénář 6 – Windows Defender (zdroj – vlastní)

7.7.6 Panda

Antivir Panda payl_https.exe po manuální kontrole neidentifikoval jako hrozbu, a proto tento soubor ponechal. (Obrázek 51)



Obrázek 51 – Hrozba neidentifikována – scénář 6 – Panda (zdroj – vlastní)

7.8 Souhrn a výsledky

V následující tabulce (Tabulka 5) budou shrnuty výsledky v jednotlivých scénářích, jak si vedl každý z vybraných antivirů. K otestování jednotlivých antivirů bylo využito 6 scénářů. Nicméně v běžné praxi jsou antivirové programy testovány mnohem důkladněji a takovýchto scénářů mohou být stovky, nebo i tisíce. Je nutné podotknout, že k otestování antivirů ve specializovaných laboratořích se používá také zero–day malware, kdežto ve scénářích zde popsaných se vyskytuje již známý malware, o kterém je záznam ve virové (malware) databázi.

Tabulka 5 – Výsledky testování (zdroj – vlastní)

	DETEKCE					
	Avast	**AVG**	**Avira**	**Bitdefender**	**Windows Defender**	**Panda**
Scénář 1 – payload klient–server	ANO	ANO	ANO	ANO	NE	NE
Scénář 2 – Veil–Ordnance vlastní shellcode	ANO	ANO	NE	ANO	ANO	NE
Scénář 3 – payload s AES šifrováním	ANO	ANO	ANO	NE	NE	NE
Scénář 4 – payload s DES šifrováním	ANO	ANO	ANO	NE	NE	NE
Scénář 5 – payload (.bat) s ARC šifrováním	ANO	ANO	NE	ANO	NE	NE
Scénář 6 – payload s využitím HTTPS	ANO	ANO	ANO	NE	NE	NE

Z výsledků v tabulce (Tabulka 5) vyplývá, že i freewarový antivirový program může uživatele velice dobře ochránit v případě stažení škodlivého malware např. z internetu. Antiviry Avast a AVG odhalily všechny hrozby z této práce okamžitě po extrahování souboru .zip, ve kterém se nacházely. Horších výsledků dosáhly antiviry Avira a Bitdefender, téměř v polovině případů oba antiviry hrozbu nedokázaly zachytit. V případě, že byla hrozba zachycena těmito antiviry, došlo k jejímu odhalení hned po extrakci souboru .zip. Vůbec nejhůře dopadly antiviry

Windows Defender a Panda, které nezachytily jedinou hrozbu (Windows Defender zachytil pouze jednu). Což pro koncového uživatele, zejména pak uživatele Windows, není potěšující zpráva. Zajímavým jevem bylo na systému Windows 10 s antivirem Windows Defender, že některé hrozby dokázal Windows Defender identifikovat až několik dní po nasazení do systému. Nicméně to může být pro uživatele Windows pouze malá útěcha, protože uživatel potřebuje být ochráněn před všemi nástrahami v reálném čase.

8 ZÁVĚR

Tato práce si klade za cíl otestovat freeware antiviry s využitím OS Kali Linux a frameworků metasploit a Veil. Užitím těchto nástrojů se ukázalo, že umožňují a nabízejí testerovi spoustu možností, jak otestovat zranitelnost různých systémů.

Výsledky ukázaly, že použití některých freeware antivirových programů může velmi dobře ochránit systém, na kterém je vybraný antivir nainstalován. Zvláště pak uživatelé z Čech mohou být spokojeni s antiviry vytvářenými v rodném kraji. Antiviry Avast a AVG v této práci, resp. v daných scénářích, obstály na výbornou. Antiviry Avira a Bitdefender některé hrozby zachytily, ale pro uživatele, očekávajícího téměř 100 % ochranu, nemusejí být správnou volbou. Špatnou zprávou pro uživatele Windows je, že Windows Defender, jenž je v nejnovějších verzích Windows ve výchozím nastavení již předinstalován, představuje velmi slabou ochranu. Z 6 nasazených malware dokázal odhalit pouze jeden, což není uspokojující zpráva. Vůbec nejhůře dopadl antivir Panda. Neodhalil ani jeden malware a stal se tak spíše pouze jakýmsi standardně běžícím programem než antivirem.

Je zřejmé, že jednotlivé scénáře neotestovaly antiviry dostatečně, bylo by zapotřebí mnohem více scénářů. Přesto i tak výsledky naměřené v této práci se víceméně shodují s výsledky nasbíranými v kapitole (kapitola 2.9) a tyto výsledky spíše potvrdily. Jediná větší odchylka se vyskytla ve výsledcích antiviru Panda, což mohlo být způsobeno právě nedostatkem jednotlivých scénářů, ve kterých byl tento antivir testován.

IT bezpečnost je v dnešním světě opravdu aktuální téma. Každý uživatel by měl být opatrný a chovat se nejen na internetu (jedná se i o chování na vlastním systému, např. při připojení přenosných zařízení) velice obezřetně, protože je plný různých nástrah, jež by mohly uživateli ublížit. Výběr antiviru je na samotném uživateli. Jak z této práce vyplývá, existují antivirové programy, které dokáží velmi dobře ochránit uživatele v případě jeho pochybení. Příkladem je stažení některého malware a jeho následné otevření. Je tedy dobré mít nějaký antivirový program nainstalován. Pro uživatele Windows je vhodné nespoléhat na vestavěný antivirový systém Microsoft Defender, ale zvolit jiný freewarový, nebo i placený antivirový program. Avast nebo AVG, které ve výsledcích testování z této práce, ale i ve výsledcích z nezávislých laboratoří, mají vysokou úspěšnost v odhalování bezpečnostních hrozeb, mohou být pro běžného uživatele ideální volbou. Je ale nutné podotknout, že sebelepší antivir neochrání uživatele stoprocentně. Proto by měl být uživatel neustále pozorný a přemýšlet nad svým chováním a jednáním.

9 LITERATURA

[1] PLESHKOV, A. S. a D. D. RUDER. Penetration Testing as a Security Analysis of Computer Systems. *News of Altai State University* [online]. 2015, 1(85), 174-181 [cit. 2017-02-06]. ISSN 1561-9451. Dostupné z: http://izvestia.asu.ru/media/files/issue/13/articles/ru/174-181.pdf

[2] What is Anti-Virus Software? WEBROOT. *Webroot* [online]. 2017 [cit. 2017-02-27]. Dostupné z: https://www.webroot.com/in/en/home/resources/tips/pc-security/security-what-is-anti-virus-software

[3] The Difference Between Antivirus and Anti-Malware (and Which to Use). GIZMODO MEDIA GROUP. *Lifehacker* [online]. 2017 [cit. 2017-02-27]. Dostupné z: http://lifehacker.com/the-difference-between-antivirus-and-anti-malware-and-1176942277

[4] Avast Free Antivirus. ECONOMIA, A.S. *Stahuj.cz* [online]. 2017 [cit. 2017-03-02]. Dostupné z: http://www.stahuj.centrum.cz/utility_a_ostatni/antiviry/kompletni/avast-free-antivirus-2/

[5] AYOCK, John. *Computer viruses and malware*. New York: Springer, 2006. ISBN 0387341889.

[6] SVOBODA, Petr. *Zabezpečení Windows Server 2008 pomocí systému Kaspersky*. Pardubice, 2013. Bakalářská práce. Univerzita Pardubice, Fakulta elektrotechniky a informatiky. Vedoucí práce Soňa Neradová.

[7] ZHANG, Zhenyu, Wujun ZHANG, Jianfeng WANG a Xiaofeng CHEN. *An Effective Cloud-Based Active Defense System against Malicious Codes*. s. 690. DOI: 10.1007/978-3-642-55032-4_71. Dostupné také z: http://link.springer.com/10.1007/978-3-642-55032-4_71

[8] ZELTSER, Lenny. *LENNY ZELSTER* [online]. c1995-2017 [cit. 2017-03-07]. Dostupné z: https://zeltser.com/what-is-cloud-anti-virus/

[9] AVAST SOFTWARE S.R.O. *Avast* [online]. c1988-2017 [cit. 2017-04-18]. Dostupné z: https://www.avast.com/cs-cz/index

[10] Avast Premier. AVAST SOFTWARE S.R.O. *Avast* [online]. c1988-2017 [cit. 2017-04-06]. Dostupné z: https://www.avast.com/cs-cz/premier

[11] AVG AntiVirus FREE. AVG TECHNOLOGIES CZ. *AVG* [online]. 2017 [cit. 2017-02-06]. Dostupné z: http://www.avg.com/cz-cs/free-antivirus-download

[12] AVG Internet Security – bez omezení. AVG TECHNOLOGIES CZ. *AVG* [online]. c2017 [cit. 2017-04-06]. Dostupné z: http://www.avg.com/cz-cs/internet-security

[13] Avira. AVIRA OPERATIONS GMBH & CO. KG. *Avira* [online]. 2017 [cit. 2017-02-07]. Dostupné z: https://www.avira.com/en/free-security-suite

[14] Free or premium protection? AVIRA OPERATIONS GMBH & CO. KG. *Avira* [online]. c2017 [cit. 2017-04-06]. Dostupné z: https://www.avira.com/#antivirus-overlay

[15] Microsoft Secure. MICROSOFT. *Microsoft* [online]. 2017 [cit. 2017-02-08]. Dostupné z: https://www.microsoft.com/en-us/security/default.aspx

[16] Bitdefender. BITDEFENDER. *Bitdefender* [online]. c1997-2017 [cit. 2017-02-08]. Dostupné z: https://www.bitdefender.com/solutions/free.html

[17] Take a Look Inside. BITDEFENDER. *Bitdefender* [online]. c1997-2017 [cit. 2017-04-06]. Dostupné z: https://www.bitdefender.com/solutions/total-security.html

[18] Panda. PANDA SECURITY. *Panda* [online]. 2017 [cit. 2017-02-10]. Dostupné z: http://www.pandasecurity.com/czechia/homeusers/solutions/free-antivirus/

[19] Panda Gold Protection. PANDA SECURITY. *Panda* [online]. 2017 [cit. 2017-02-10]. Dostupné z: http://www.pandasecurity.com/czechia/homeusers/solutions/gold-protection/?ref=menu

[20] RUBENKING, NEIL J. The Best Free Antivirus Protection of 2017. *PC Magazine* [online]. 2017 [cit. 2017-02-13]. ISSN 0888-8507. Dostupné z: http://www.pcmag.com/article2/0,2817,2388652,00.asp

[21] Best Free Overall PC Antivirus. PURCH. *Tom's guide* [online]. 2017 [cit. 2017-02-15]. Dostupné z: http://www.tomsguide.com/us/best-antivirus,review-2588-5.html
[22] Avira Free Antivirus Review. *Tom's guide* [online]. 2017 [cit. 2017-03-15]. Dostupné z: http://www.tomsguide.com/us/avira-free-antivirus,review-2207.html
[23] The best free antivirus 2017. PURCH. *Techradar* [online]. 2017 [cit. 2017-02-15]. Dostupné z: http://www.techradar.com/news/software/applications/best-free-antivirus-1321277
[24] Kali Linux Official Documentation. OFFENSIVE SECURITY. *Kali* [online]. c2017 [cit. 2017-03-12]. Dostupné z: http://docs.kali.org/introduction/what-is-kali-linux
[25] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 10. ISBN 9781849519489.
[26] Nmap. OFFENSIVE SECURITY. *KALI TOOLS* [online]. 2017 [cit. 2017-03-16]. Dostupné z: http://tools.kali.org/information-gathering/nmap
[27] ZITTA, Stanislav. *Penetrační testování*. Pardubice, 2013. Bakalářská práce. Univerzita Pardubice, Fakulta elektrotechniky a informatiky. Vedoucí práce Josef Horálek.
[28] LYON, Gordon Fyodor. *Nmap network scanning: official Nmap project guide to network discovery and security scanning*. Sunnyvale, CA: Insecure.Com, c2008, s. 468. ISBN 978- 0-9799587- 1-7.
[29] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 220-223. ISBN 9781849519489.
[30] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 290-300. ISBN 9781849519489.
[31] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 323-325. ISBN 9781849519489.
[32] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 318. ISBN 9781849519489.
[33] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 332-333. ISBN 9781849519489.
[34] CaseFile. PATERVA PTY LIMITED. *Paterva* [online]. 2017 [cit. 2017-03-16]. Dostupné z: https://www.paterva.com/web7/buy/maltego-clients/casefile.php
[35] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 11. ISBN 9781849519489.
[36] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 250-251. ISBN 9781849519489.
[37] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 250. ISBN 9781849519489.
[38] ZITTA, Stanislav. *Penetrační testování*. Univerzita Pardubice, 2013. Diplomová práce.
[39] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 256. ISBN 9781849519489.
[40] ALLEN, Lee, Tedi HERIYANTO a Shakeel ALI. *Kali linux: assuring security by penetration testing*. Second Edition. S.l.: Packt Publishing Limited, 2014, s. 258. ISBN 9781849519489.
[41] *Veil* [online]. c2017 [cit. 2017-04-14]. Dostupné z: https://www.veil-framework.com/
[42] AYCOCK, John Daniel. *Computer viruses and malware*. New York: Springer, 2006, s. 11-19. ISBN 978-0-387-30236-2.
[43] Malware. AVAST SOFTWARE S.R.O. *Avast* [online]. c1988-2017 [cit. 2017-04-13]. Dostupné z: https://www.avast.com/cs-cz/c-malware
[44] AYCOCK, John Daniel. *Computer viruses and malware*. New York: Springer, 2006, s. 11-19. ISBN 978-0-387-30236-2.

10 PŘÍLOHY

Příloha A – *CD se zpracovanou bakalářskou prací*

- soubor KynclM_TestovaniAntivirovych_JH_2017.pdf