VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2019 Bc. Ivo Procházka

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

ANALÝZA RANSOMWARU GLOBEIMPOSTER
ANALYSIS OF THE GLOBEIMPOSTER RANSOMWARE

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR
Bc. Ivo Procházka

VEDOUCÍ PRÁCE
SUPERVISOR
Ing. Zdeněk Martinásek, Ph.D.

BRNO 2019

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**
Ústav telekomunikací

Student: Bc. Ivo Procházka ***ID:*** 125293
Ročník: 2 ***Akademický rok:*** 2018/19

NÁZEV TÉMATU:

Analýza ransomwaru GlobeImposter

POKYNY PRO VYPRACOVÁNÍ:

Cílem diplomové práce je analýza ransomwaru GlobeImposter a jeho jednotlivých typů. V teoretické části se seznamte s metodami statické i dynamické analýzy binárního spustitelného kódu (př. reverzní inženýrství, sandboxing a dekompilace). Po domluvě s vedoucím detailně analyzujte vybraný ransomware GlobeImposter (veškeré analýzy budou probíhat na izolovaném virtuálním pracovišti). Prostudujte techniky šifrování uživatelských dat a komunikaci s řídícím centrem. Analyzujte použitý algoritmus a práci s šifrovacím klíčem, navrhněte dešifrovací program pro zvolený typ ransomware GlobeImposter.

DOPORUČENÁ LITERATURA:

[1] HAMPTON, Nikolai; BAIG, Zubair; ZEADALLY, Sherali. Ransomware behavioural analysis on windows platforms. Journal of information security and applications, 2018, 40: 44-51.

[2] CRACIUN, Vlad Constantin; MOGAGE, Andrei; SIMION, Emil. Trends in design of ransomware viruses. IACR Cryptology ePrint Archive, 2018, 2018: 598.

Termín zadání: 1.2.2019 ***Termín odevzdání:*** 16.5.2019

Vedoucí práce: Ing. Zdeněk Martinásek, Ph.D.
Konzultant:

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

ABSTRAKT

Cílem této diplomové práce je analýza vzorku ransomwaru GlobeImposter získaného z napadeného zařízení. Teoretická část práce se zabývá rozdělením škodlivého kódu a typů ransomwaru podle funkce a popisem práce se šifrovacími algoritmy a klíči. Dále jsou představeny postupy statické a dynamické analýzy škodlivého kódu a požadavky na testovací prostředí. V praktické části je popsán zdroj vzorku škodlivého kódu a navržené prostředí (virtuální a na fyzickém hardwaru) a provedena statická a dynamická analýza získaného vzorku ransomwaru GlobeImposter. V závěru práce jsou zhodnoceny dosažené výsledky a navržen další postup k realizaci dekompilátoru pro analyzovaný vzorek.

KLÍČOVÁ SLOVA

Malware, ransomware, počítačové viry, kyberbezpečnost, GlobeImposter

ABSTRACT

The aim of this diploma thesis is to analyze an instance of the GlobeImposter ransomware extracted from an affected device. The first part outlines various types of malware and ransomware and includes a description of encryption mechanisms and key distribution systems. It also discusses possible approaches of static and dynamic analysis of malware samples and requirements for test environments. The practical part describes the source of the malware sample, the physical and virtual test environment and the results of the static and dynamic analysis of the GlobeImposter ransomware. The final part discusses the results and the possibility of implementing a decryptor for the analyzed GlobeImposter ransomware.

KEYWORDS

Malware, ransomware, computer viruses, cybersecurity, GlobeImposter

PROCHÁZKA, Ivo. *Analýza ransomwaru GlobeImposter*. Brno, 2019, 52 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Zdeněk Martinásek, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Analýza ransomwaru GlobeImposter“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové 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

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Zdeňku Martináskovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno ....

podpis autora

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Bezpečnost v informačních technologiích je v poslední době ve společnosti velmi diskutované téma. Ve světě se odehrává velké množství útoků v kyberprostoru, od útoků na velké firmy či státy (například s cílem krádeže či pozměnění dat) po kampaně, které cílí pouze na koncové uživatele (například formou nevyžádané pošty).

V posledních letech došlo k výraznému růstu popularity tzv. ransomwaru, tedy softwaru, který zašifruje veškerá data v napadeném zařízení a jejich dešifrování podmiňuje zaplacením výkupného. Z posledních let je možné zmínit dva nejznámější útoky. První se stal v roce 2017, kdy byla napadena společnosti A.P. Moller–Maersk. Po útoku ransomwaru NotPetya byla společnost nucena přeinstalovat 45 000 počítačů a 4 000 serverů. Celkové škody způsobené tímto útokem se odhadují na 200 až 300 milionů amerických dolarů. Ke druhému velkému útoku došlo v březnu 2019, kdy společnost Norsk Hydro byla napadena ransomwarem LockerGoga. V současné době se odhaduje, že škody napáchané tímto útokem přesáhnou 40 milionů amerických dolarů [1] [2] [3].

Ve světle těchto událostí je jasné, že takovéto útoky je třeba analyzovat a předcházet jim. Ransomware může ve firemním prostředí napáchat nejen velké finanční škody, ale také výrazně poškodit reputaci a důvěryhodnost napadené společnosti. Pro koncové uživatele navíc tento druh škodlivého kódu představuje další hrozbu ztráty cenných dat ve svých zařízeních.

Cílem této práce je provést analýzu zařízení, které bylo napadeno ransomwarem, a to konkrétně variantou GlobeImposter. V práci je provedena analýza stavu napadeného zařízení, lokalizace vzorku škodlivého kódu a jeho samotná analýza.

1 Škodlivý kód

1.1 Malware

Termín malware vznikl v angličtině spojením slov „malicious“ (zlovolný, zlomyslný) a „software“ [5]. Do češtiny bývá přeložen jako škodlivý software či škodlivý kód. Pod pojmem malware rozumíme obecně jakýkoli počítačový program, jehož cílem je mazat, blokovat, měnit či kopírovat uživatelská data, nežádoucím způsobem ovlivňovat chod zařízení, ve kterém běží, nebo sítě, k níž je dané zařízení připojeno. Tato kapitola stručně popisuje nejčastější typy malwaru. Je třeba zmínit, že uvedené definice typů škodlivého kódu nejsou pevně dané a že jeden vzorek malwaru může vykonávat (a obvykle také vykonává) více než jednu funkci, a proto je možné jej zařadit do více skupin [6].

1.1.1 Viry a počítačové červy

Tento typ škodlivého kódu se replikuje v počítači či v počítačové síti bez vědomí uživatele či správce. Následné kopie tohoto kódu musí disponovat schopností se dále replikovat, aby je bylo možné zařadit do této kategorie. Nejčastěji se tento druh malwaru šíří prostřednictvím příloh elektronické pošty, hypertextových odkazů na webu či FTP, pomocí komunikačních nástrojů (např. ICQ, IRC) nebo P2P sítí. K proniknutí do cílového zařízení využívá metod sociálního inženýrství (např. e-mailové zprávy, které nabádají uživatele k otevření infikované přílohy), šíří se přes nezabezpečená síťová úložiště nebo dokáže zneužívat chyby v operačních systémech či aplikačním softwaru [6].

1.1.2 Trojské koně

Jedná se o škodlivý kód, který se není schopen dále replikovat, na rozdíl od předchozí kategorie. Trojské koně je možné dále klasifikovat podle typu aktivity, kterou vykonávají v napadeném zařízení.

- Backdoor – jde o typ kódu, jenž umožní jeho tvůrci vzdáleně ovládat napadené zařízení bez vědomí uživatele. Šíří se podobnými způsoby jako viry.
- Rootkit – tento druh škodlivého kódu je navržen tak, aby získal přístup k počítači či prostoru, jenž obvykle není přístupný. Většinou také maskuje svoji přítomnost, aby jej nebylo možné detekovat bezpečnostním softwarem.
- Exploit – tento typ kódu slouží k využití jedné či více zranitelností v operačním systému či aplikačním softwaru. Obvykle se využívá k proniknutí do zařízení oběti, do kterého se následně nainstaluje další škodlivý kód [6].

1.1.3 Suspicious packers

Tento typ kódu neslouží přímo k poškození cílového počítače, ale k zabalení či skrytí škodlivého kódu tak, aby bylo složité jej rozeznat pomocí konvenčních metod (např. antivirových programů). Hlavním cílem těchto nástrojů je zamezit analýze samotného kódu nebo ji ztížit [6].

1.1.4 Škodlivé nástroje

Tato třída škodlivého kódu na rozdíl od předchozích druhů nepředstavuje nebezpečí pro zařízení, v němž běží. Tyto nástroje se používají k automatickému vykonání předem definovaného útoku. Tento typ malwaru může sloužit k cílené infekci viry či červy nebo k provedení útoku Denial of Service na cílovou stanici [6].

1.2 Ransomware

Termín ransomware vznikl v angličtině spojením slov „ransom“ (výkupné) a „software“. Jedná se o malware, jehož cílem je znemožnit uživateli přístup k zařízení či souborům v něm uloženým. Zároveň nabízí možnost získat zpět data či přístup k zařízení v případě, že uživatel zaplatí „výkupné“. Tento typ škodlivého kódu tedy slouží ke generování zisku pro útočníka.

1.2.1 Varianty ransomwaru

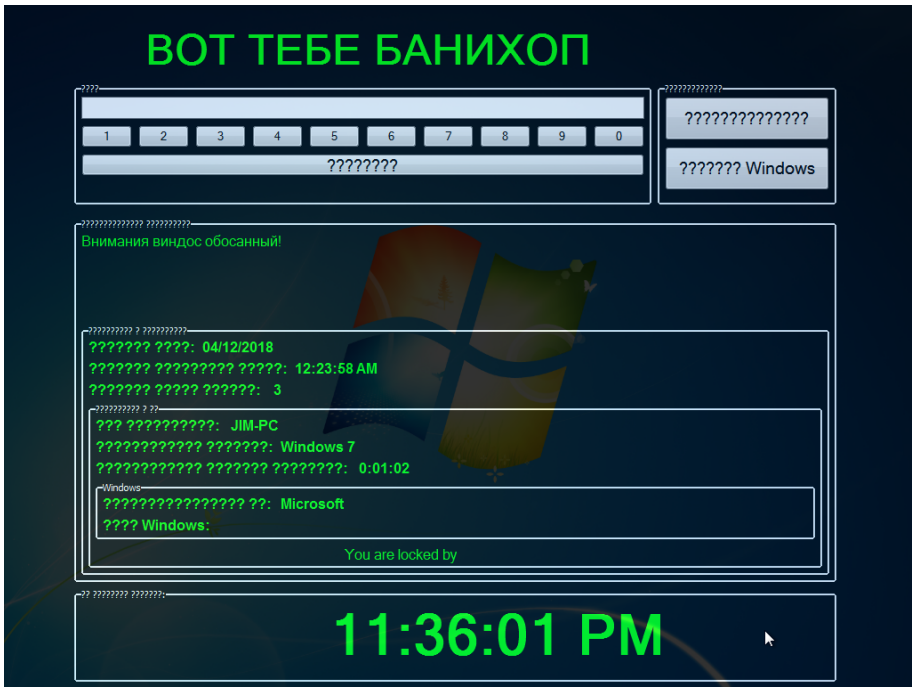
Locker ransomware

Tato varianta je určena pouze k zamezení přístupu uživatele k napadenému zařízení. Obvykle dojde ke znepřístupnění uživatelského rozhraní, přičemž ransomware přímo vyzývá uživatele k zaplacení výkupného za opětovné zpřístupnění funkčnosti. Oběť má obvykle omezené možnosti práce se zařízením, například může používat pouze určitou část uživatelského rozhraní (jako je klávesnice) [4].

Na rozdíl od následujícího typu většina variant locker ransomwaru nijak nezasahuje do dat uložených v zařízení. To znamená, že v případě napadení je možné infekci odstranit navrácením zařízení do předchozího stavu. Locker ransomware je tedy méně efektivní v získávání výkupného od uživatelů, protože zkušenější uživatelé mohou data získat zpět pomocí nástrojů určených přímo k odstranění dané varianty ransomwaru. Možností je také připojení úložiště napadeného zařízení k jinému zařízení, ve kterém si uživatel může netknutá data stáhnout [4].

Vzhledem k tomu, že tento druh ransomwaru je možné většinou odstranit bez větších škod na zařízení, využívají jeho tvůrci různé techniky sociálního inženýrství,

aby vyvolali u uživatelů pocit, že je nutné ihned zaplatit výkupné, aby nedošlo ke ztrátě dat. Velice populární je zneužívání jmen policejních či státních institucí a maskování výkupného jako pokuty za ilegální aktivitu. Typickou ukázkou takového škodlivého kódu je například varianta Win32/Ransom, která je pro laiky v ČR známá jako „Virus policie ČR“. Ukázku locker ransomwaru znázorňuje obrázek 1.1 [4].



Obr. 1.1: Locker ransomware

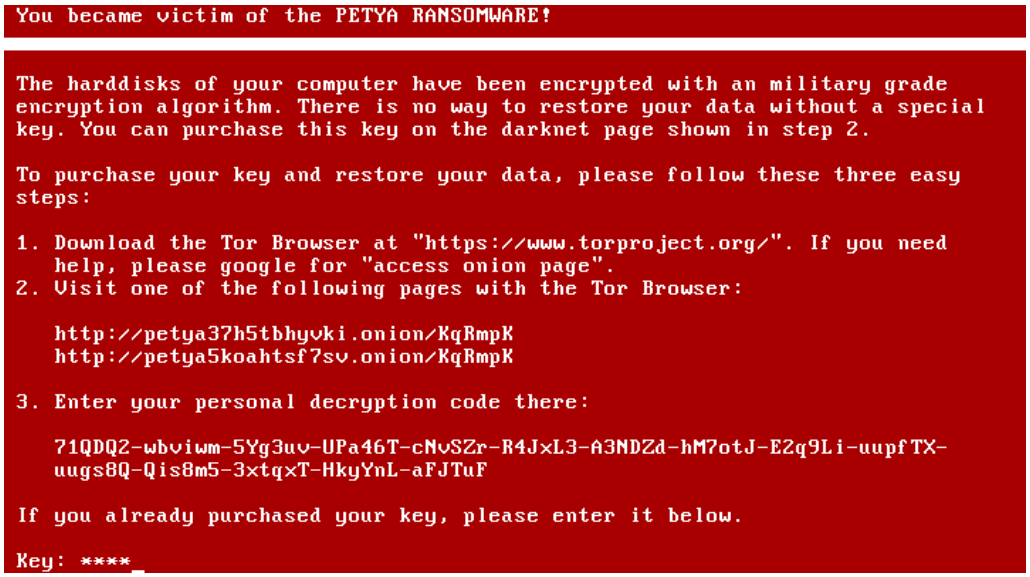
Crypto ransomware

Crypto ransomware je navržen tak, aby v napadeném zařízení vyhledal a následně zašifroval uživatelská data. Zašifrováním je uživateli znemožněn přístup k datům, dokud nezíská příslušné šifrovací klíče. Tato varianta obvykle nechává zařízení bez poškození operačního systému, aby bylo možné provést platbu ihned po zašifrování.

Uživatelé si velice často neuvědomují cenu dat uložených v zařízení (např. rodinných fotografií, souborů k pracovním projektům) a v případě úspěšného útoku se může zaplacení výkupného jevit jako jediný způsob, jak data získat zpět. Tvůrci crypto ransomwaru také spoléhají na to, že uživatelé neprovádí či podceňují pravidelné zálohování uživatelských dat. Tento fakt je podpořen například průzkumem společnosti Backblaze, který zjistil, že 24 % uživatelů neprovádí pravidelné zálohy vůbec a 67 % uživatelů zálohuje svá data méně často než jednou měsíčně [4] [7].

Crypto ransomware se typicky spouští na pozadí a v napadeném zařízení hledá uživatelská data (fotky, dokumenty či jiná data, která mohou mít pro uživatele cenu).

Následně je znepřístupní pomocí různých kryptografických algoritmů. Během útoku i po něm je tedy obvykle možné počítač dále používat, protože systémová data zůstávají nedotčena. Malware se skrývá, dokud nejsou zašifrována všechna data. Po dokončení šifrování je uživatel informován, že všechna data jsou znepřístupněna a k jejich získání je nutné zaplatit výkupné (ukázku lze vidět na obrázku 1.2) [4].



Obr. 1.2: Crypto ransomware – varianta Petya

1.2.2 Šifrovací algoritmy

Moderní ransomware obvykle používá standardizované šifrovací algoritmy, většinou kombinaci symetrických kryptosystémů pro šifrování uživatelských dat a asymetrických šifrovacích systémů pro práci s klíči (viz dále) [8].

Symetrické šifrovací kryptosystémy používají stejný klíč k šifrování i dešifrování dat. Dělí se do dvou skupin:

- Proudové šifry
- Blokové šifry [14]

V případě proudových šifer jsou data šifrována po jednotlivých bitech pomocí předem vygenerované pseudonáhodné posloupnosti, která se následně použije jako šifrovací klíč. Výhodou proudových šifer je vysoká rychlost a hardwarová nenáročnost. Nevýhodou je menší odolnost vůči kryptoanalýze. V ransomwaru se z blokových šifer obvykle používají algoritmy Salsa20, RC4 [8] [14]. Blokové šifry data šifrují po blocích s pevně danou bitovou délkou. Každému bloku vstupních bitů o definované délce je jednoznačně přiřazen výstupní blok dat. Výhodou této metody je rychlost šifrování a vysoká odolnost vůči kryptoanalýze. Ransomware nejčastěji

využívá k šifrování Advanced Encryption Standard (AES) s délkou klíče 128, 256 nebo 512 bitů. Jednou z výhod použití tohoto standardů je i skutečnost, že moderní procesory obsahují instrukční sadu pro akceleraci operací AES. Kromě šifry AES je možné se setkat také s šiframi Blowfish nebo RC2 [8] [14] [13].

Je třeba také zmínit, že ransomware ve většině případů (v rámci [8] bylo zjištěno, že jde o více než 80 % útoků) používá pro šifrování dat standardizované šifrovací knihovny, jako je Microsoft Crypto API (součást MS Windows od verze Windows 95) nebo OpenSSL. Pouze několik vzorků zvolilo nějakou variantu proprietárních šifrovacích implementací (například Salsa nebo Chacha). Tento přístup se používá pro svou jednoduchost a také proto, že implementace přímo v operačních systémech využívají hardwarovou akceleraci zmíněnou výše.

Tab. 1.1: Varianty ransomwaru a použitých šifrovacích algoritmů, převzato z [8]

Název ransomwaru	**Typ šifrování**	**Klíč**
OpenToYou	RC4	Předdefinovaný klíč
Annabelle	AES	Předdefinovaný klíč
Nemucod	Cyklické XOR	Předdefinovaný klíč
Amnesia	AES128, CBC	Čas (a funkce Rand() v programovacím jazyce C)
Globe V3	AES256, ECB	Čas (a funkce Rand() v programovacím jazyce C)
Nemesis	AES256/512, ECB,CBC	Čas (a funkce Rand() v programovacím jazyce C)
Xorist	TEA/XOR	Čas (a funkce Rand() v programovacím jazyce C)
Xmas	CUSTOM	Čas (a funkce Rand() v programovacím jazyce C)
LeChiffre	BlowFish	Předdefinovaný klíč a informace o uživateli
Petya	Salsa20	secp192k1

1.2.3 Práce se šifrovacími klíči a komunikace s Command & Control servery

Jak bylo zmíněno výše, moderní ransomware pro svoji funkci používá kombinaci symetrických kryptosystémů, které slouží k zašifrování uživatelských dat, a asymetrických kryptosystémů pro zabezpečení klíčů použitých symetrickými kryptosystémy. Ransomware s klíči k symetrickým kryptosystémům obvykle pracuje jedním z těchto způsobů:

- Použije předem definovaný šifrovací klíč přímo v programu (v současnosti se používá pouze výjimečně).
- Škodlivý kód obsahuje vestavěný veřejný klíč, s jehož pomocí se následně zašifruje lokálně vygenerovaný klíč k blokové šifře, kterou jsou zašifrována uživatelská data.
- Použije šifrovací klíče stažené z Command & Control (C&C) serveru. Tyto servery slouží útočníkům k řízení útoků ransomwaru a vzdálenému ukládání klíčů.

- Vygeneruje šifrovací klíče lokálně v zařízení a následně je nahraje na C&C server.
- Použije protokol ECDH k výměně klíčů přes síť.

Většina ransomwaru v současnosti šifruje klíče kryptosystémem RSA či protokolem ECDH. Tento přístup umožňuje bezpečnou výměnu klíčů a nedovoluje oběti (či virovému analytikovi) získat potřebné klíče k dešifrování dat. Zároveň útočníkovi výrazně zjednodušuje správu klíčů a umožňuje sestavení seznamu či databáze napadených uživatelů (ve formě jedinečných uživatelských ID) a odpovídajících RSA klíčů použitých k zašifrování dat. Tvůrci ransomwaru tedy stačí vyhledat klíče k dešifrování podle uživatelského ID (obvykle přímo od uživatele) a následně je poslat napadenému uživateli [8].

Z asymetrických kryptosystémů se nejčastěji používá kryptosystém RSA. Při práci s kryptosystémem RSA tvůrci používají následující řetězení šifer:

1. Vestavěným RSA klíčem se zašifruje globální klíč k blokové šifře, pomocí které jsou zašifrována uživatelská data.
2. Vestavěný RSA klíč slouží k zašifrování náhodně generovaných klíčů, kterými jsou zašifrovány jednotlivé soubory v napadeném zařízení.
3. Ransomware obsahuje vestavěný veřejný klíč A, pomocí kterého se zašifruje náhodně generovaná dvojice veřejného a soukromého RSA klíče B. Veřejným klíčem B se následně zabezpečí náhodně vygenerované klíče pro zašifrování jednotlivých souborů v zařízení za použití symetrických šifer [8].

1.3 Analýza škodlivého kódu

Analýza škodlivého kódu slouží k získání informací potřebných k náležité reakci na průnik do sítě či zařízení. Cílem je určit příčinu průniku a identifikovat všechny napadené soubory a zařízení. Při analýze je nutné zjistit o použitém škodlivém kódu co nejvíce informací a také určit, jak jej spolehlivě rozeznat v síti a případně zamezit dalším škodám. Po identifikaci škodlivých souborů v zařízení je vhodné vytvořit pro napadené prostředí definice škodlivého kódu, aby bylo možné zjistit rozsah napadení. Virové definice se dělí na:

- Hostitelské definice (či indikátory) – slouží k detekci škodlivého kódu v napadených zařízeních. Na rozdíl od antivirových definic se tyto indikátory soustředí na identifikaci následků útoku (změny v registrech, soubory vytvořené škodlivým kódem) a ne na charakteristiky škodlivého kódu samotného.
- Síťové definice – používají se k detekci škodlivého kódu na síťové úrovni. Tyto definice je možné vytvořit bez analýzy samotného vzorku škodlivého kódu, ale v případě, že škodlivý kód nejdříve analyzujeme, bývají definice kvalitnější a účinnější.

Ve většině případů se analyzuje škodlivý kód ve formě binárních souborů. Tyto soubory není možné jednoduše „přečíst“ a k pochopení funkce kódu je potřeba použít vhodnou sadu nástrojů. Při analýze škodlivého kódu se většinou uplatňuje sada tzv. best practices“ (obecných osvědčených postupů) a různé techniky se kombinují za účelem získání co největšího množství informací o zkoumaném vzorku. V praxi se provádí *statická* a *dynamická* analýza, jež jsou popsány dále [9].

1.3.1 Statická analýza

Statická analýza je obvykle prvním krokem při studiu škodlivého kódu. Jejím cílem je pochopit strukturu a pokud možno celý kód programu bez spuštění studovaného vzorku. Při provádění statické analýzy se obvykle uplatňují následující postupy:

Skenování antivirovými programy

Prvním krokem při zkoumání nechtěného kódu je analýza vzorku větším množstvím antivirových programů. Ty ke své práci používají databáze vlastních virových definic a provádí behaviorální analýzu daného vzorku (tzv. heuristiku). Nevýhodou tohoto přístupu je, že tvůrci mohou škodlivý kód jednoduše upravit a tím změnit i jeho takzvaný otisk. V takovém případě je možné, že daný vzorek nebude odpovídat virovým definicím, a nebude tedy antivirovým softwarem detekován. Heuristická analýza může pomoci k odhalení i nového a dosud neznámého škodlivého kódu, nemusí však vždy fungovat. Výrobců antivirového softwaru je celá řada a každý výrobce si vytváří vlastní definice. Proto je vhodné při skenování použít více druhů antivirového softwaru. K tomu je možné použít například web `https://www.virustotal.com/`, který analyzovaný soubor prověří pomocí většiny běžně dostupných antivirových programů [9].

Identifikace škodlivého kódu podle haše

Při analýze škodlivého kódu je obvykle vhodné vypočítat haš vzorku pomocí softwaru. Hašovací program podle použitého algoritmu (nejčastěji MD5 či různé varianty SHA) přiřadí danému vzorku unikátní řetězec – tzv. haš. Haš funguje jako identifikátor (nebo „otisk prstu“) analyzovaného vzorku. Při analýze je možné:

- sdílet haš s komunitou analytiků a pomoci tak s identifikací škodlivého kódu,
- vyhledat haš ve veřejných zdrojích a zjistit, zda kód nebyl už dříve analyzován,
- použít haš jako jednoznačné označení pro analyzovaný vzorek [9].

Detekce zabaleného či zašifrovaného kódu

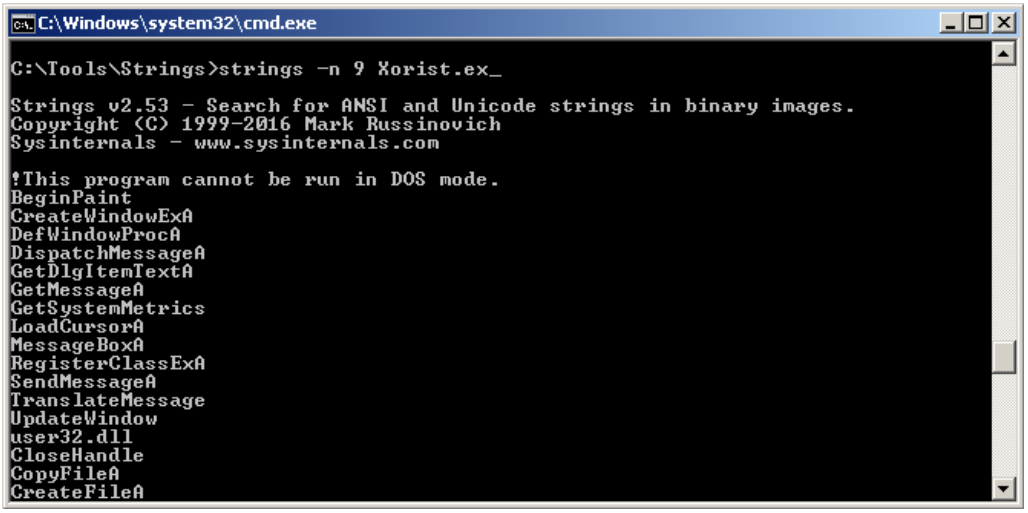
Tvůrci škodlivého kódu používají různé techniky k maskování celého škodlivého kódu či jeho částí. Maskované kódy se nejčastěji dělí do těchto dvou skupin:

- „Obfuscated“ (volně přeloženo jako skrytý či zašifrovaný) program
- Zabalený program

V obou případech je cílem ztížit či znemožnit statickou analýzu škodlivého kódu. Po spuštění zamaskovaného programu dojde nejdříve k rozbalení kódu a následně k jeho spuštění v paměti zařízení. Zabalené programy obvykle obsahují pouze malé množství řetězců a samotný program (či jeho části) jsou skryty.

Extrakce řetězců z analyzovaného vzorku

Při analýze je vhodné ze spustitelného souboru vyextrahovat textové řetězce. S takto získanými informacemi je možné programu lépe porozumět, případně z něj získat cenné informace ve formě URL adres či funkcí volaných při běhu programu. V případě špatně navrženého ransomwaru lze také z kódu přímo extrahovat šifrovací klíče. Pro extrakci je možné použít například program Strings, jehož ukázka je zobrazena na obrázku 1.3 [9].



Obr. 1.3: Ukázka programu Strings

1.3.2 Dynamická analýza

Hlavním rozdílem mezi statickou a dynamickou analýzou je skutečnost, že v případě dynamické analýzy škodlivý kód spouštíme a pozorujeme, jakým způsobem ovlivňuje chod daného zařízení. Pro potřeby dynamické analýzy je důležité mít vytvořené

vhodné prostředí. Požadavky na tvorbu takového prostředí jsou popsány v kapitole 1.4 [10].

Při dynamické analýze se obvykle nepoužívají programy navržené k analýze škodlivého kódu. Typicky se používají nástroje, které zaznamenávají veškerou probíhající činnost v běžícím zařízení či síti. Analytik musí tyto nástroje nastavit tak, aby běžný provoz zařízení byl odfiltrován [10].

Monitorování běžících procesů

Cílem monitorování běžících procesů je zjistit co nejvíce informací o spuštěném škodlivém kódu. Nejčastěji se k tomuto účelu využívá program Sysinternals Process Monitor. Jedná se o pokročilý monitorovací program systémů Microsoft Windows. Nástroj v reálném čase sleduje souborový systém, registry, procesy a vlákna běžící v zařízení. Process Monitor funguje na principu injekce ovladače do jádra operačního systému. Pomocí něj převádí data do uživatelského prostředí, kde je s nimi možné volně pracovat prostřednictvím běžného rozhraní. Nástroj disponuje robustním filtrovacím modulem, což je pro efektivní analýzu škodlivého kódu nedocenitelný nástroj [10].

Monitorování sítě

Monitorování síťové komunikace je nedílnou součástí analýzy škodlivého kódu. Účelem je zjistit, zda malware komunikuje s vnějším světem, a pokud ano, tuto komunikaci zdokumentovat. V současné době se k tomuto účelu nejčastěji využívá multiplatformní nástroj Wireshark, který slouží k zachytávání a analýze veškerého síťového provozu v analyzovaném zařízení. V případě ransomwaru síťová analýza umožňuje jednoduše lokalizovat Command & Control servery či pokusy o infekci okolních zařízení. Nevýhodou tohoto nástroje je, že neumožňuje identifikaci programu, který vygeneroval analyzovaný síťový provoz [10].

Extrakce procesu z paměti

Jak již bylo zmíněno, malware obvykle používá různé maskovací techniky, což ztěžuje celou analýzu. Aby bylo možné kód analyzovat, je vhodné jej nejdříve vyextrahovat ze spustitelného souboru. K tomuto účelu je možné použít různé metody (například programy určené pro konkrétní typy packerů). Jednou z nejpoužívanějších je extrakce běžícího programu přímo z paměti operačního systému. Tato technika vyžaduje izolované prostředí, v němž spustíme škodlivý kód a pomocí specializovaného nástroje vyexportujeme spuštěný program včetně připojených knihoven. Je třeba zmínit, že takto vyexportovaný soubor obvykle není dále spustitelný, protože celou operací dochází k poškození hlavičky souboru. Soubor je ale možné dále zkoumat

například metodami statické analýzy (zejména extrakcí řetězců a funkcí ze spustitelného souboru) [16].

Dekompilace a debugging

Posledním krokem v analýze škodlivého kódu je obvykle jeho dekompilace a debugging. Dekompilace je proces, ve kterém je jako vstup použit spustitelný soubor a výstupem je soubor obsahující zdrojový kód programu. Cílem této operace je rekonstrukce zdrojového kódu programu a pochopení jeho funkce. Dekompilátory obvykle nejsou schopny plně rekonstruovat vstupní soubory, umožňují ale pochopit logiku kódu a funkce volané programem [17].

Debugging, nebo také ladění programu, slouží obvykle k odstranění chyb v programu. V kontextu škodlivého kódu se používá k analýze běhu programu. Obvykle se škodlivý kód spouští pomocí specializovaného nástroje, který umožňuje uživatelem řízený průběh programu a pozorování jeho vstupů a výstupů [18] [10].

1.4 Prostředí pro analýzu škodlivého kódu

1.4.1 Tvorba testovacího prostředí

Protože škodlivý kód je navržen k tomu, aby nežádoucím způsobem měnil či přebíral kontrolu nad napadeným zařízením, je žádoucí mít pro účely analýzy vytvořené kompletně oddělené prostředí, které lze kdykoli uvést do původního stavu. U testovacích prostředí je zároveň vhodné oddělit i síť od produkčních systémů, aby se předešlo případnému šíření nákazy (například s využitím útoků nultého dne). K vytvoření takového prostředí je možné použít:

- virtualizované prostředí
- nebo prostředí běžící na fyzickém hardwaru [10].

Hlavní výhodou virtualizovaného prostředí je cena a možnost používat „snímky“ virtuálních strojů. Tyto snímky umožňují uložení stavu analyzovaného zařízení a přepínání mezi uloženými stavy podle potřeby. Nevýhodou je, že některé druhy škodlivého kódu se snaží virtuální prostředí detekovat a v případě úspěšné detekce se nespustí či nevykonají útok. Při použití fyzického hardwaru je nevýhodou vyšší cena a nižší flexibilita (například návrat do stavu před útokem je časově náročný). Fyzické prostředí na druhou stranu umožňuje sledovat chování škodlivého kódu v reálných podmínkách [11] [10].

1.4.2 Sandboxing

V oboru počítačové bezpečností je sandbox obecné označení pro bezpečnostní mechanismus určený k oddělení běžícího kódu v zařízení. Obvykle slouží k identifikaci, testování či zmírnění bezpečnostních chyb v programech. Často se také používá ke spouštění neotestovaných či nedůvěryhodných programů bez ohrožení zařízení. V těchto nástrojích jsou často obsaženy další nástroje určené k analýze různých aspektů chování spuštěného programu, například formou virtualizovaného hardwaru, služeb či sítě. Bezpečnostní analytici se na sandboxové prostředí spoléhají při analýze, protože umožňuje věrně simulovat napadené zařízení bez rizika infekce [15].

Sandboxy mají i určité nevýhody:

- Moderní škodlivý kód se při běhu snaží detekovat virtualizované prostředí. Při úspěšné detekci se určité části kódu spustí pouze částečně nebo vůbec.
- Různá varianty škodlivého kódu mohou spoléhat na přítomnost souborů, klíčů v registru nebo přítomnost konkrétního softwaru v napadeném zařízení. V prostředí sandboxu tyto proměnné nemusí být dostupné, což obvykle vyústí v nefunkčnost škodlivého kódu.
- Prostředí sandboxu nemusí být vhodné pro spuštění daného kódu například z důvodu nekompatibility s operačním systémem [10].

Na trhu je k dispozici velké množství různých komerčních i open source řešení pro sandboxing. Namátkou lze zmínit například Joe Sandbox, HybridAnalysis. V této diplomové práci byl zvolen Cuckoo Sandbox, a to zejména proto, že se jedná o zdarma dostupné, dobře zdokumentované open source řešení.

Cuckoo Sandbox

Cuckoo Sandbox je nástroj pro automatizovanou analýzu škodlivého kódu. Analyzovaná data o aktivitě dokáže přenášet z izolovaného operačního systému. U analyzovaného zařízení dokáže:

- vytvořit výpis systémových volání prováděných škodlivým kódem,
- trasovat soubory vytvořené škodlivým kódem,
- provádět výpis paměti škodlivého kódu,
- zaznamenávat síťový provoz napadeného stroje,
- snímat obrazovku během provádění kódu,
- vytvořit plný výpis paměti napadeného zařízení [12].

2 Testovací prostředí a výchozí situace

2.1 Návrh testovacího prostředí

Pro potřeby analýzy škodlivého kódu byla vytvořena dvě testovací prostředí. Jak bylo zmíněno v teoretické části, testovací prostředí je možné provozovat přímo na fyzickém hardwaru nebo ve virtualizovaném prostředí. Práce s virtualizovaným prostředím je jednodušší a rychlejší, ale některé vzorky malwaru se v něm mohou chovat jinak, a proto bylo použito i prostředí přímo na fyzickém hardwaru.

2.1.1 Virtuální testovací prostředí

Analýzu škodlivého kódu je vhodné vždy provádět na odděleném prostředí, aby nedošlo k šíření infekce, či v případě ransomwaru zašifrování hostitelského zařízení. Pro potřeby analýzy tedy bylo sestaveno testovací prostředí, jehož schéma je zobrazeno na obrázku 2.1. Testovací prostředí bylo vytvořeno na operačním systému Ubuntu 18.04. Pro virtualizaci byl zvolen nástroj Oracle VirtualBox ve verzi 5.2. Ve virtualizačním nástroji byly vytvořeny dvě oddělené virtuální sítě. Podsíť *vboxnet0* slouží k dynamické analýze škodlivého kódu a také umožňuje automatizované testování malwaru pomocí nástroje Cuckoo Sandbox (viz dále). V této podsíti běží stanice, na kterých přímo probíhala analýza škodlivého kódu. Stanice byly nakonfigurovány následovně:

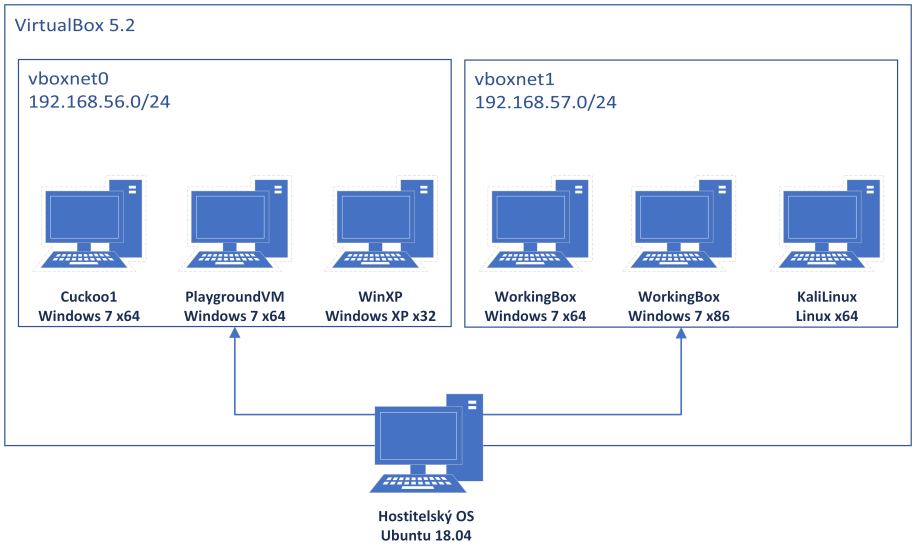
- operační systém Windows 7 nebo Windows XP
- nastaveno jako obecný virtuální stroj v prostředí VirtualBox, nejsou nainstalovány dodatečné nástroje pro hostitelský OS
- bez nainstalovaných aktualizací či antivirového softwaru
- bezpečnostní funkce (firewall, řízení uživatelských účtů) jsou vypnuté
- nainstalován základní uživatelský software

Síť *vboxnet1* slouží ke statické analýze a dekompilaci škodlivého kódu. V této síti se nachází další virtualizované stanice, která běží na operačním systému Windows 7 nebo Kali Linux, včetně všech dostupných aktualizací. Konfigurace je následující:

- nastaveno jako stanice pro daný operační systém v prostředí VirtualBox, nainstalované dodatečné nástroje pro hostitelský OS
- nainstalovány poslední aktualizace, antivirový software je vypnut
- bezpečnostní funkce (firewall a řízení uživatelských účtů) jsou zapnuté
- nainstalován základní uživatelský software a software potřebný pro analýzu škodlivého kódu

Virtualizované prostředí bylo zvoleno zejména proto, že umožňuje jednoduše vytvářet snímky virtuálních strojů. S jejich pomocí se lze podle potřeby přesouvat mezi

větším množstvím stavů sytému. Zároveň je takto vytvořené prostředí výborně škálovatelné a je možné mít spuštěných více virtuálních strojů jak pro automatickou, tak pro uživatelskou analýzu. Lze i vytvářet další virtuální stroje s různými verzemi operačních systémů.



Obr. 2.1: Schéma virtuálního pracoviště

2.1.2 Testovací prostředí na fyzickém hardwaru

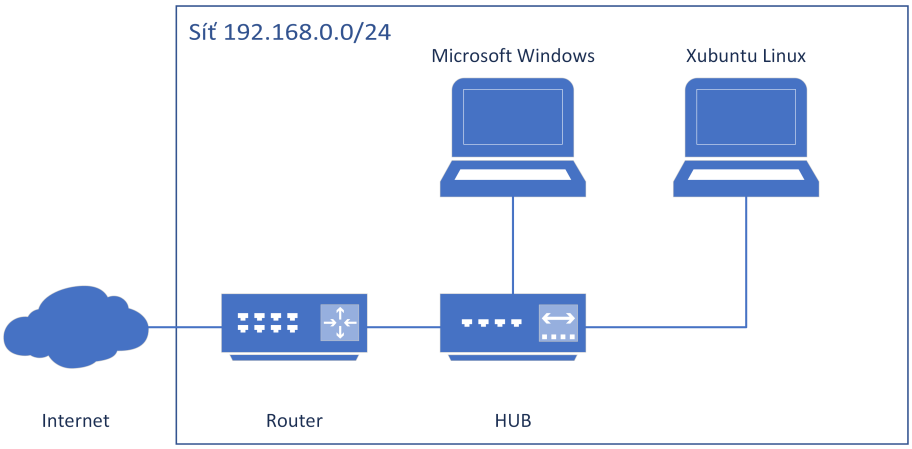
Fyzické prostředí bylo vytvořeno s využitím dvou notebooků, routeru a hubu. Hlavním účelem tohoto prostředí bylo otestovat, zda se chování ransomwaru na fyzickém hardwaru mění (zejména z pohledu síťové komunikace, což bylo zjišťováno pomocí nástroje Wireshark). Z tohoto důvodu byl také zvolen starší síťový hub, který na rozdíl od modernějších zařízení zrcadlí veškerý síťový provoz na ostatní porty. Schéma sítě je zobrazeno na obrázku 2.2.

Vlastnosti stanice s OS Windows:

- operační systém Windows 7 SP1 (32-bitová varianta) nebo Windows XP SP3,
- bez nainstalovaných aktualizací či antivirového softwaru,
- bezpečnostní funkce (firewall a řízení uživatelských účtů) byly vypnuty.

Vlastnosti Linuxové stanice:

- operační systém Lubuntu Linux (zvolen z důvodu nižších hardwarových nároků),
- nainstalovány poslední aktualizace operačního systému,
- nainstalován nástroj Wireshark.



Obr. 2.2: Schéma fyzického pracoviště

2.2 Výchozí situace

Vstupem pro diplomovou práci byl obraz disku, který byl zašifrován ransomwarem. Vlastnosti disku byly následující:

- Jméno souboru: WD5000_Ransomware
- Velikost souboru: 500 107 862 016 bajtů
- Hash: eeacaa3ea1fade944f75843555c2c155c73273116e0e0b20426620ea919bc4ee
- Operační systém zařízení: Windows 10 Enterprise ve verzi 1709

Vzhledem k tomu, že obraz byl vytvořen pomocí linuxového nástroje *DD*, nebylo jej možné přímo procházet a dále s ním pracovat. Pro účely analýzy jej bylo možné připojit jako disk do operačního systému nebo převést do podoby virtuálního stroje. Byla zvolena druhá možnost, zejména z důvodu vyšší bezpečnosti (virtuální stroj běží v izolovaném prostředí a síti) a také proto, že v tomto případě je možné vytvářet snímky virtuálního počítače.

2.2.1 Analýza stavu počítače

Po konverzi byl virtuální disk připojen k virtuálnímu počítači a následně byl proveden pokus o start virtuálního stroje. Virtuální stroj po konverzi nenastartoval a pouze nabízel možnosti opravy operačního systému. V rámci pokusů o zprovoznění byly provedeny následující úkony:

- Hardwarová konfigurace virtuálního stroje byla různými způsoby upravována s cílem dosáhnout kompatibilního nastavení.
- Na virtuálním stroji bylo resetováno heslo k uživatelskému účtu „Admin“ a spuštěn nástroj pro kontrolu a opravu disku.
- Bylo provedeno několik pokusů o opravu operačního systému. K tomu byly použity nástroje *sfc* a *DISM* s negativním výsledkem.

- Virtuální disk byl připojen k funkční stanici běžící na operačním systému Windows 7 a následně byla provedena kontrola dvěma antivirovými programy – ClamAV a Eset Online Scanner. Pomocí těchto nástrojů byly lokalizovány vzorky ransomwaru ve složkách:
  C:\Users\Admin\Music\PH new\svchost.exe
  C:\Windows\SysWOW64\config\systemprofile\AppData\Local\svchost.exe
- Po kontrole operačního systému bylo zjištěno, že některé důležité soubory operačního systému byly ransomwarem zašifrovány, což obnovení funkčnosti napadeného počítače znemožňuje.

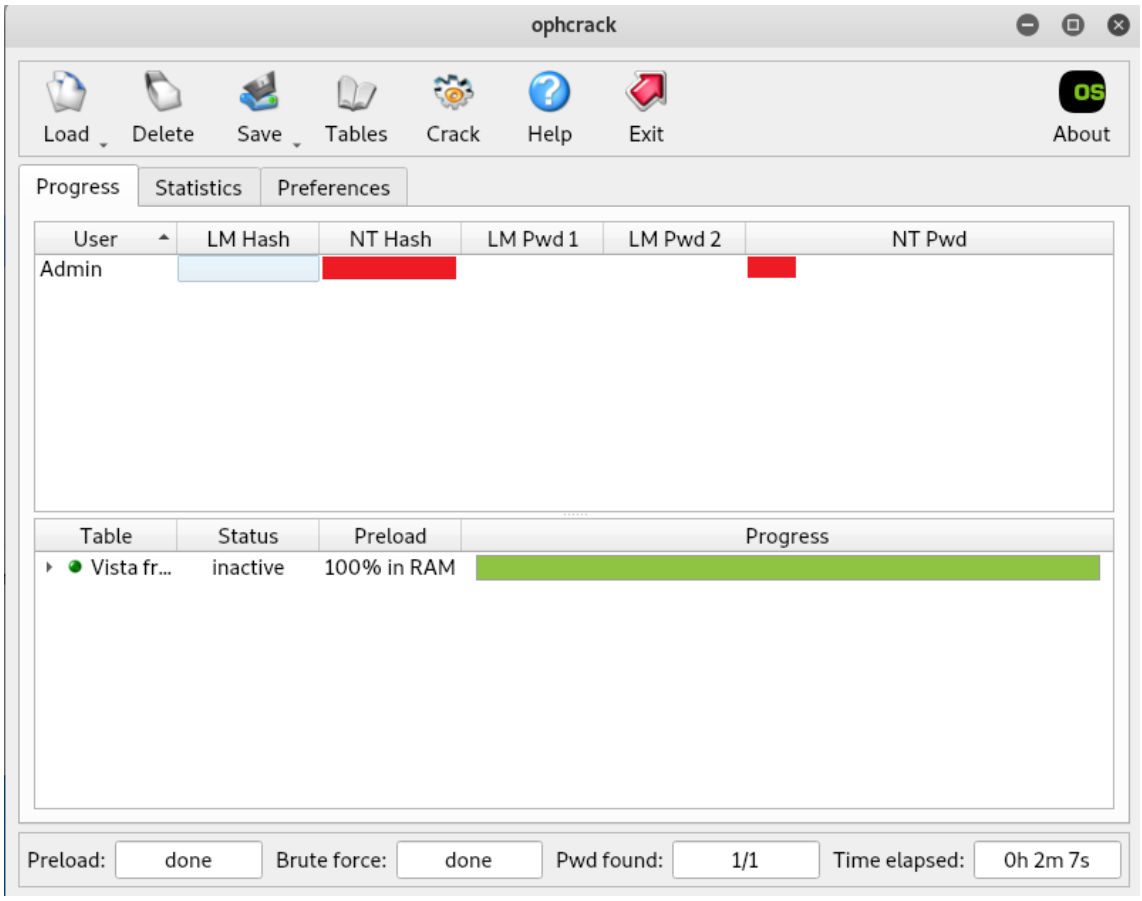
2.3 Analýza dat na napadeném zařízení

Podle časových razítek v operačním systému byl virus spuštěn 19. 4. 2018 ve 20:29:17. Tato informace vychází z prvního souboru, který byl vytvořen ransomwarem a obsahuje údaj „personal ID“. Poslední soubor byl ransomwarem vytvořen 19. 4. 2018 v 23:15:13. Z těchto časových razítek lze usoudit, že ransomware v napadeném počítači běžel 2 hodiny a 14 minut.

Analýzou dat v napadeném zařízení bylo zjištěno, že ransomware byl spuštěn pod uživatelským účtem „Admin“. Složka C:\Users\Admin\Music\PH new\ obsahuje vzorek ransomwaru a byla vytvořena 19. 4. 2018 v 20:26:07 (několik minut před spuštěním ransomwaru). V této složce se nachází nástroj *Process Hacker* a část tohoto nástroje nebyla ransomwarem zašifrována. To znamená, že nástroj byl v době útoku spuštěn.

Výše uvedená zjištění byla konzultována s uživatelem napadeného zařízení, který podle svých slov k účtu „Admin“ neměl přístup. Uživatel ke své práci používal aplikaci Vzdálená plocha (zařízení bylo stále připojeno k internetu), a proto se jako nejpravděpodobnější jeví útok přes protokol RDP a slabé heslo uživatele „Admin“. Tento typ útoku spoléhá na nezabezpečený port 3389 (výchozí port pro aplikaci vzdálené plochy). Tato domněnka byla následně potvrzena extrahováním haše hesla uživatele „Admin“ z napadeného systému a útokem pomocí *rainbow tables* s využitím nástroje Ophcrack. Uživatelský účet měl nastavený pouze PIN v délce čtyř numerických znaků (viz obrázek 2.3).

Útoky na vzdálenou plochu s použitím ransomwaru byly v roce 2017 registrovány i v České republice [1].



Obr. 2.3: Prolomené heslo pomocí nástroje Ophcrack

[1] https://www.root.cz/clanky/analyza-napadeni-ransomware-staci-otevreny-port-rdp-a-slabe-heslo/

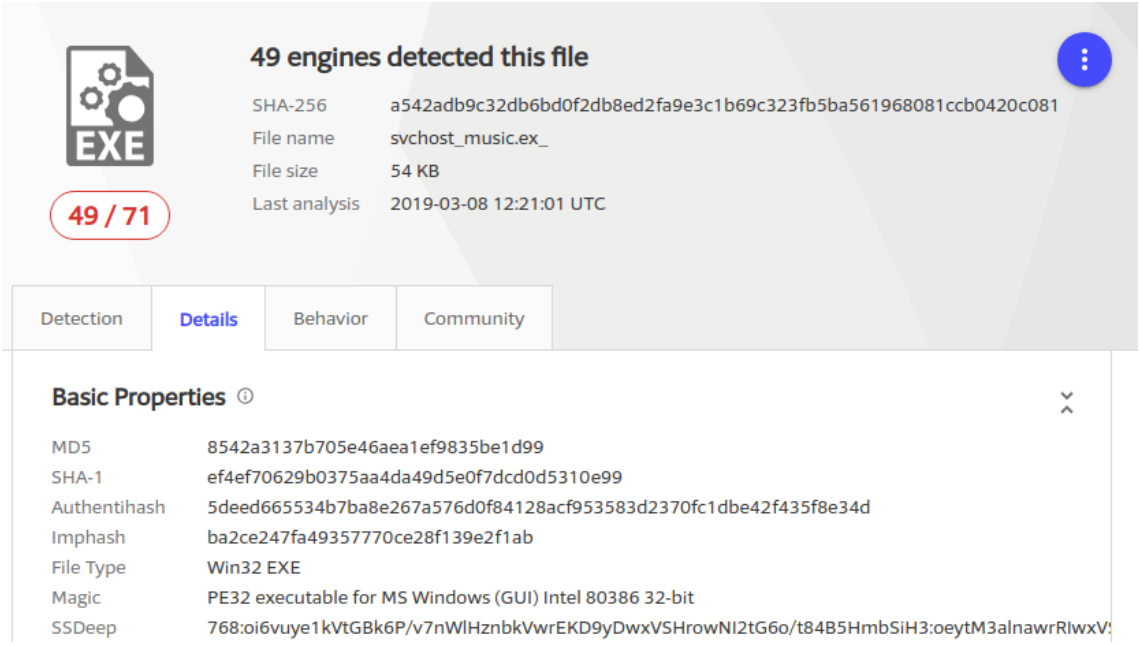
3 Vlastní analýza škodlivého kódu

V této kapitole jsou popsány postupy použité při analýze vzorku a ukázky z analýzy.

3.1 Statická analýza vzorku

Kontrola antivirovým softwarem

Antivirová kontrola byla provedena pomocí webového nástroje VirusTotal. Vzorek byl poprvé analyzován 3. 3. 2019. Nástroj v něm potvrdil přítomnost varianty ransomwaru GlobeImposter. Další informace o souboru včetně hašů jsou zobrazeny na obrázku 3.1.



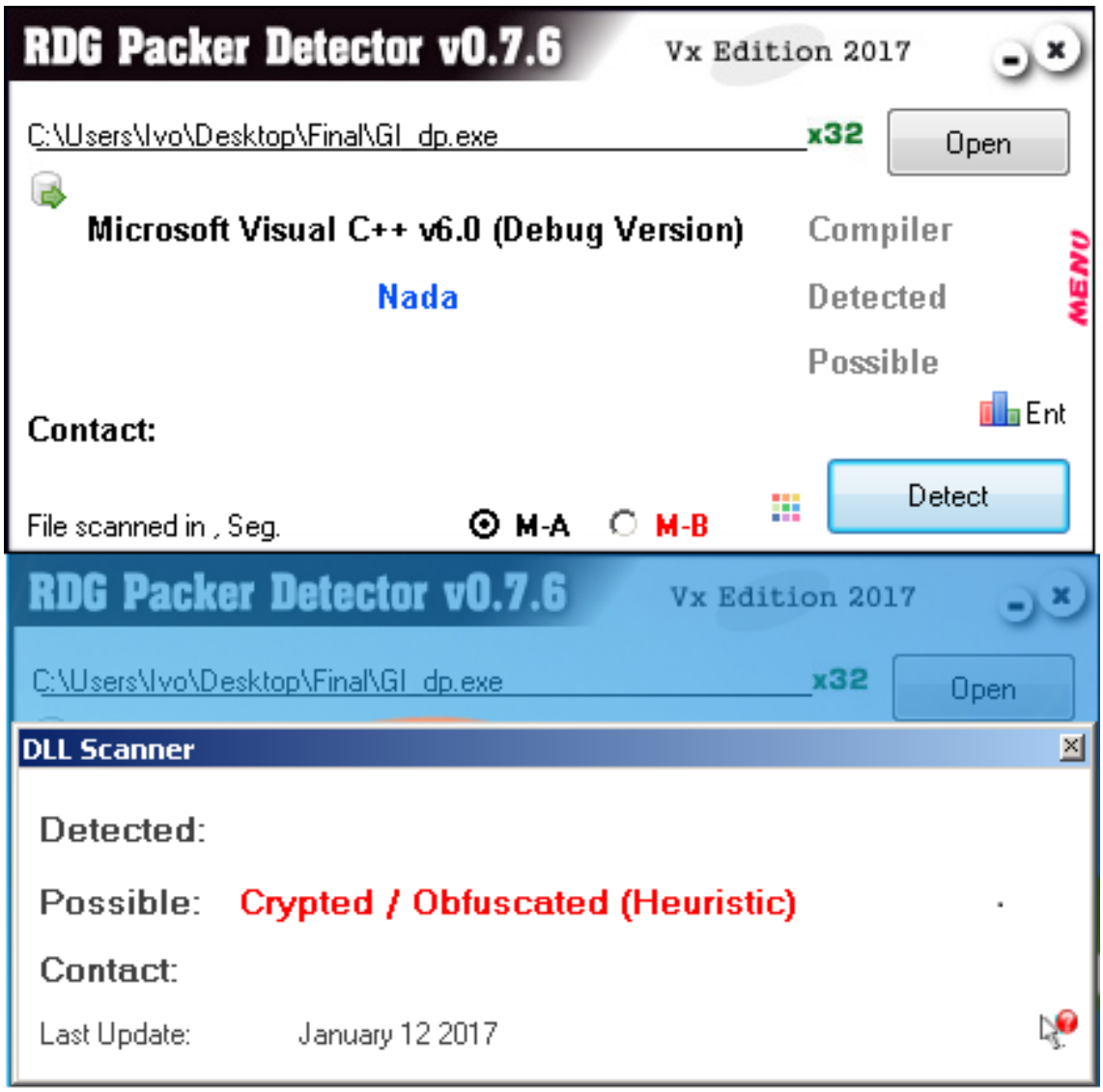
Obr. 3.1: Zpráva z webového nástroje VirusTotal

Test na použití kompresních či šifrovacích programů

Tato kontrola zjišťuje, zda program nepoužívá kompresní software, což by značně ztížilo další statickou analýzu. Na obrázku 3.2 je vidět, že vzorek (či jeho část) je s největší pravděpodobností zašifrován či jinak zamaskován. Z výpisu programu je patrné, že program byl nejspíše sestaven v prostředí Microsoft Visual C++ 6.0.

Extrakce řetězců ze vzorku

Protože program používá kompresní software, lze očekávat, že výpis řetězců ze vzorku bude omezený.



Obr. 3.2: Test programem PeID

Z výpisu je možné zjistit, že vzorek ransomwaru používá či volá funkce spojené s asymetrickým kryptosystémem RSA (řetězce rsa_genkey a rsa_encrypt). Řetězce také obsahují klíčová slova SHA-224 a SHA-256, což znamená, že vzorek implementuje nebo volá tyto funkce v operačním systému. Ve výpisu se také nachází řetězec *Software\Microsoft\Windows\Current Version\RunOnce*. To značí, že ransomware nějakým způsobem čte tento klíč v registru operačního systému či do něj zapisuje. Ve výpisu lze také najít obsáhlý soupis funkcí, které ransomware používá pro svou činnost. Většina funkcí v programu je standardní, za zmínku ale stojí funkce *CryptAcquireContextW*, *CryptReleaseContext* a *CryptGenRandom*. Přítomnost těchto funkcí znamená, že ransomware volá funkce implementované v Crypto API rozhraní OS Windows.

Knihovny použité ransomwarem

Pro tento druh analýzy byl zvolen program Dependency Walker. Analýzou bylo zjištěno, že program používá následující DLL soubory:

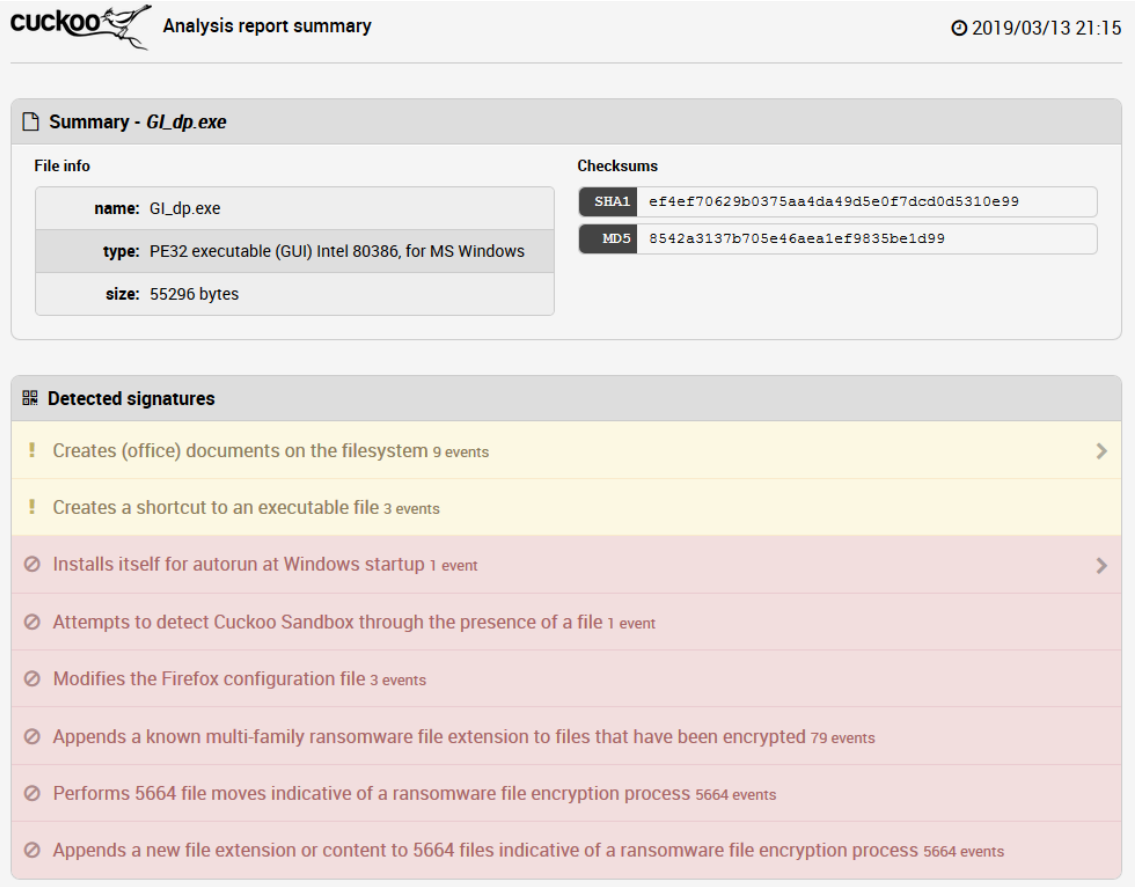
- KERNEL32.DLL – Jedná se o nejčastěji používanou knihovnu, jejíž využití značí práci s pamětí, soubory či hardwarem.
- ADVAPI32.DLL – Tato knihovna slouží k obsluze různých komponent (např. registrů) nebo CryptoAPI v operačním systému Windows.
- SHELL32.DLL – Tato knihovna slouží ke spouštění dalších souborů či obsluze příkazového řádku.
- SHLWAPI32.DLL – Tato knihovna obsluhuje funkce operačního systému (UNC, URL, přístup k registrům, nastavení barev).
- NTDLL.DLL – Jedná se o rozhraní pro jádro operačního systému. Použití této knihovny naznačuje, že autor malwaru v kódu používá funkci, která není programům standardně dostupná.

3.2 Dynamická analýza vzorku

Automatická analýza pomocí Cuckoo Sandboxu

Prvním krokem dynamické analýzy je spuštění automatické analýzy v prostředí Cuckoo Sandbox. Ve výsledné zprávě je možné nalézt stručné shrnutí analyzovaného kódu. Nejdůležitější části zprávy jsou znázorněny na obrázku 3.3 a 3.4. Ze zprávy je možné zjistit, že ransomware v PC zůstává a spouští se při každém startu PC. Zároveň se pokouší o detekci sandboxovaného prostředí a mění nastavení systému (viz dále). Následně šifruje soubory na disku a k souboru přidá příponu známou pro ransomware. Ze síťové činnosti je vidět, že ransomware nekomunikuje směrem

do internetu, a tudíž ani s Command and Control servery. Tento fakt byl dodatečně ověřen i v nevirtualizovaném prostředí a bylo potvrzeno, že použitý vzorek negeneruje žádný síťový provoz (viz obrázek B.4).

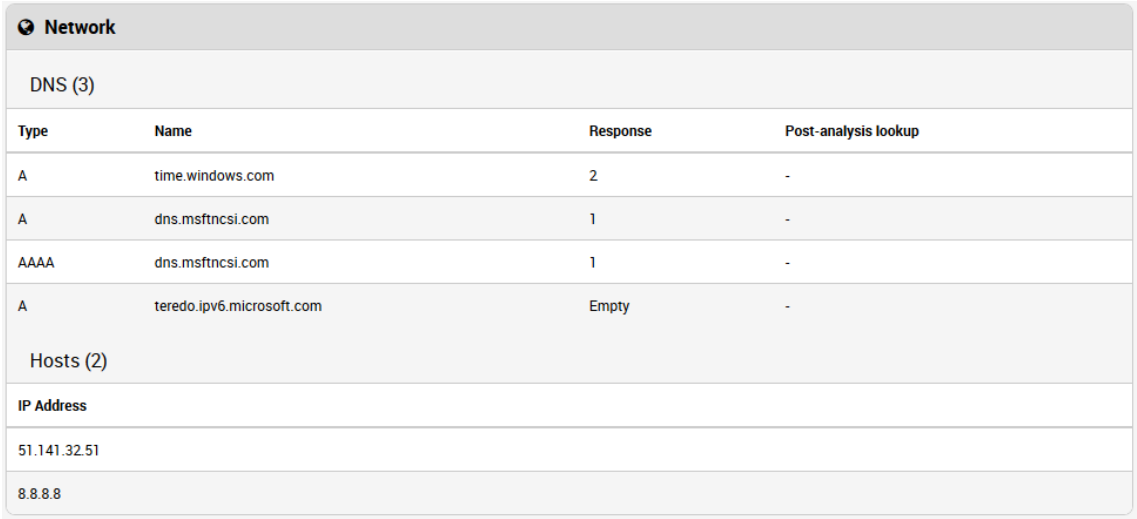


Obr. 3.3: Report z programu Cuckoo Sandbox – shrnutí

Stav operačního systému po spuštění ransomwaru

V této kapitole jsou zkoumány následky spuštění ransomwaru. Po jeho spuštění bylo pozorovatelné zvýšené využití procesoru, samotný program ale neměl žádné uživatelské rozhraní a běžel na pozadí.

Po zašifrování dat byl v každé složce s daty vytvořen soubor obsahující instrukce, jak získat zpět data. Analyzovaný vzorek požadoval po uživateli zaslání jednoho zašifrovaného souboru na uvedenou e-mailovou adresu. Analýza zašifrovaných souborů ukázala, že na konci všech zašifrovaných souborů je vložen řetězec „personal ID“. Tento identifikátor je identický pro všechny zašifrované soubory, což znamená, že všechna data na disku jsou zašifrována pomocí stejného klíče.



Network

DNS (3)

Type	Name	Response	Post-analysis lookup
A	time.windows.com	2	-
A	dns.msftncsi.com	1	-
AAAA	dns.msftncsi.com	1	-
A	teredo.ipv6.microsoft.com	Empty	-

Hosts (2)

IP Address
51.141.32.51
8.8.8.8

Obr. 3.4: Report z programu Cuckoo Sandbox – síťová činnost

Po spuštění ransomwaru se kód vloží do registru operačního systému, konkrétně do klíče *HKEY_CURRENT_USER\Software\Microsoft\Windows\Current Version\RunOnce* (byl zmíněn výše v části věnované statické analýze). V důsledku této operace se škodlivý kód spustí při dalším startu počítače. Po spuštění je zároveň do systémové proměnné *%TEMP%* vložen skript, který smaže stínové kopie systému Windows (jedná se o technologii systému Windows pro automatické zálohování souborů[1]). Dále skript z PC odstraní uložené relace pro klienta terminálového serveru a vzdálené plochy a vyčistí protokol událostí operačního systému. Snímky obrazovky dokumentující toto chování lze najít v příloze B.

[1] `https://docs.microsoft.com/en-us/windows/desktop/vss/volume-shadow-copy-service-overview`

Obr. 3.5: Informační HTML soubor

Extrakce programu z paměti

Jak bylo zjištěno v sekci 3.1, celý program (nebo jeho část) je zašifrován či dynamicky generován. Proto je nutné virus spustit a vyextrahovat běžící proces z paměti. K tomuto účelu byl použit program Process Dump[2]. Ve vyextrahovaném souboru lze najít větší množství proměnných. V této části jsou rozebrány nejdůležitější části analyzovaného vzorku.

Na začátku souboru se nachází následující řetězec:

```
BEA2E4C4 F6F77D2E 42882138 AB35A477 30880F9B B594F99C 4187ADDC
327739E7 9F85A28D ACEFE7B0 B7DD26EF 0043032C A5DA87BD B89AB7D2
07434B21 1C1CCCC5 E1F42522 CE89B1D5 2FB3C50B 88CA15DA 00699C1C
D74D103C 3E876C5A 844054E2 C1FEE98B A278CF05 DA9B3B61 1E8BCCD2
064514D9 5A698C76 2FB0216C 9C3DBA7E E611490B 48BF18FC F991A70E
AEBD86B4 84F3F894 5E0982DD 9E0B4EEC 5E03852C EC0F4564 F7AAAB32
08139784 5C56CC25 35ADE6B7 8303F5CA F8A2709D 2102D971 9BB6070E
8020A0D0 EB76341F AFA070B6 A9DA0D53 FA646390 149D7C36 3491E73E
DE96F2EB 836D716D 4DC61900 DB4CB9A9 8818EEF4 C81EBD76 C3C7CF6C
CD5DAC91
```

[2] http://www.split-code.com/processdump.html

Tento řetězec se skládá z celkem 512 hexadecimálních znaků. Převodem tohoto klíče do binární soustavy vznikne 2048bitová sekvence. Vzhledem k tomu, že vzorek nekomunikuje s Command & Control serverem po síti (viz výše) a tento řetězec je konstantní nezávisle na spuštěném PC, se jedná o veřejný klíč kryptosystému RSA. Tento klíč slouží k zabezpečení řetězce „personal ID“, který obsahuje klíč k symetrickému kryptosystému, s jehož pomocí jsou zašifrována uživatelská data.

Následujících 41 řetězců slouží k identifikaci složek a přípon souborů, které jsou vyloučeny ze šifrování. Výpis vyloučených složek je uveden v příloze A. Seznam obsahuje zejména složky operačního systému Windows, antivirových programů a aplikačního softwaru. Jedinou příponou, která je explicitně vyloučena ze šifrování, je přípona *.LIN+*.

Následuje řetězec obsahující název informačního HTML souboru vygenerovaného programem. V programu se nachází několik řetězců, které neslouží k žádnému konkrétnímu účelu. Jsou to tyto:

- C:\Users\VahtangTelecom\Desktop\SW T\20180403 (New key)\patch.tmp
- .$er,.4db,.4dd,.4d,.4mp,.a

V běžícím programu je také obsažen řetězec „personal ID“. To znamená, že klíč k šifrování je vygenerován hned po spuštění programu, zašifrován a uložen v paměti programu, odkud se následně zapisuje do šifrovaných souborů.

Průběh útoku

Pro potřeby analýzy běhu programu byl použit nástroj Process Monitor[3]. Vzorek byl společně s monitorovacím programem spuštěn ve virtuálním prostředí běžícím na operačním systému Windows 7 (64bitová varianta). Po dokončení šifrování byl protokol vyexportován z programu a analyzován na odděleném stroji.

Inicializace programu

Prvním krokem po spuštění ransomwaru je načtení systémových knihoven. Průběh této operace je zachycen na obrázku 3.6. V této fázi se program chová standardně a načítá moduly operačního systému. V kontextu ransomwaru stojí za zmínku pouze modul *cryptbase.dll*, což je poskytovatel základních kryptografických funkcí OS Windows.

Po načtení základních knihoven se program začne chovat nestandardně. Na obrázku 3.7 je zdokumentováno, že dalším krokem je zkopírování viru do složky proměnné *%TEMP%* definované v operačním systému. Po úspěšném zkopírování je proveden zápis do registru operačního systému Windows do klíče *HKEY_CURRENT_USER\Software\Microsoft\Windows\Current Version\RunOnce*. Zapsaná hodnota

[3] `https://docs.microsoft.com/en-us/sysinternals/downloads/procmon`

Time of Day	Process Name	PID	Operation	Path	Result
21:24:07,7481917	Gl_dp.exe	2380	Load Image	C:\Users\Jim\Desktop\Gl_dp.exe	SUCCESS
21:24:07,7482438	Gl_dp.exe	2380	Load Image	C:\Windows\System32\ntdll.dll	SUCCESS
21:24:07,7483152	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\ntdll.dll	SUCCESS
21:24:07,7490129	Gl_dp.exe	2380	Load Image	C:\Windows\System32\wow64.dll	SUCCESS
21:24:07,7494176	Gl_dp.exe	2380	Load Image	C:\Windows\System32\wow64win.dll	SUCCESS
21:24:07,7497837	Gl_dp.exe	2380	Load Image	C:\Windows\System32\wow64cpu.dll	SUCCESS
21:24:07,7502433	Gl_dp.exe	2380	Load Image	C:\Windows\System32\kernel32.dll	SUCCESS
21:24:07,7503342	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\kernel32.dll	SUCCESS
21:24:07,7503977	Gl_dp.exe	2380	Load Image	C:\Windows\System32\kernel32.dll	SUCCESS
21:24:07,7504525	Gl_dp.exe	2380	Load Image	C:\Windows\System32\user32.dll	SUCCESS
21:24:07,7509422	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\kernel32.dll	SUCCESS
21:24:07,7510771	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\KernelBase.dll	SUCCESS
21:24:07,7515908	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\advapi32.dll	SUCCESS
21:24:07,7516847	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\msvcrt.dll	SUCCESS
21:24:07,7523020	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\sechost.dll	SUCCESS
21:24:07,7524592	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\rpcrt4.dll	SUCCESS
21:24:07,7526218	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\sspicli.dll	SUCCESS
21:24:07,7527218	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\cryptbase.dll	SUCCESS
21:24:07,7529320	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\shell32.dll	SUCCESS
21:24:07,7531223	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\shlwapi.dll	SUCCESS
21:24:07,7532298	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\gdi32.dll	SUCCESS
21:24:07,7533301	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\user32.dll	SUCCESS
21:24:07,7534966	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\lpk.dll	SUCCESS
21:24:07,7536044	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\usp10.dll	SUCCESS
21:24:07,7554563	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\imm32.dll	SUCCESS
21:24:07,7556021	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\msctf.dll	SUCCESS

Obr. 3.6: Načtené systémové knihovny po spuštění ransomwaru

je pojmenována *BrowserUpdateCheck* a obsahuje cestu k souboru vytvořenému v předchozím kroku. Cílem této operace je zajistit, že při příštím startu operačního systému bude ransomware znovu spuštěn.

Time of Day	Process Name	PID	Operation	Path	Result
21:24:07,8813675	Gl_dp.exe	2380	CreateFile	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8816039	Gl_dp.exe	2380	CloseFile	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8817882	Gl_dp.exe	2380	CreateFile	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8818066	Gl_dp.exe	2380	QueryAttributeInfor...	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8818162	Gl_dp.exe	2380	QueryBasicInformati...	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8818240	Gl_dp.exe	2380	QueryAttributeInfor...	C:\Users\Jim\Desktop\Gl_dp.exe	SUCCESS
21:24:07,8818337	Gl_dp.exe	2380	SetEndOfFileInform...	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8819141	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Wow6432Node\Policies\Microsoft\Windows\System	REPARSE
21:24:07,8819249	Gl_dp.exe	2380	RegOpenKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\System	SUCCESS
21:24:07,8819351	Gl_dp.exe	2380	RegSetInfoKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\System	SUCCESS
21:24:07,8819415	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\System\CopyFileChunkSize	NAME NOT FOUND
21:24:07,8819571	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\System\CopyFileOverlapp...	NAME NOT FOUND
21:24:07,8819647	Gl_dp.exe	2380	RegCloseKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\System	SUCCESS
21:24:07,8819984	Gl_dp.exe	2380	ReadFile	C:\Users\Jim\Desktop\Gl_dp.exe	SUCCESS
21:24:07,8820402	Gl_dp.exe	2380	WriteFile	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8821390	Gl_dp.exe	2380	SetBasicInformation...	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8822043	Gl_dp.exe	2380	CloseFile	C:\Users\Jim\Desktop\Gl_dp.exe	SUCCESS
21:24:07,8822146	Gl_dp.exe	2380	CloseFile	C:\Users\Jim\AppData\Local\Gl_dp.exe	SUCCESS
21:24:07,8823109	Gl_dp.exe	2380	RegOpenKey	HKCU	SUCCESS
21:24:07,8823299	Gl_dp.exe	2380	RegQueryKey	HKCU	SUCCESS
21:24:07,8823399	Gl_dp.exe	2380	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce	SUCCESS
21:24:07,8823540	Gl_dp.exe	2380	RegSetInfoKey	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce	SUCCESS
21:24:07,8823615	Gl_dp.exe	2380	RegQueryValue	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce\BrowserUpd...	NAME NOT FOUND
21:24:07,8823760	Gl_dp.exe	2380	RegQueryKey	HKCU	SUCCESS
21:24:07,8823868	Gl_dp.exe	2380	RegCreateKey	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce	SUCCESS
21:24:07,8823953	Gl_dp.exe	2380	RegSetInfoKey	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce	SUCCESS
21:24:07,8824085	Gl_dp.exe	2380	RegQueryKey	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce	SUCCESS
21:24:07,8824254	Gl_dp.exe	2380	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce\BrowserUpd...	SUCCESS
21:24:07,8824480	Gl_dp.exe	2380	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce	SUCCESS

Obr. 3.7: Načtené systémové knihovny po spuštění ransomwaru

Prvním krokem programu je vygenerování souboru s pomocnými daty v systémové proměnné *%ALL USERS%\Public*. Jméno tohoto souboru odpovídá haši

(SHA-256) klíče uvedeného v kapitole 3.2, který byl lokalizován ve spustitelném souboru a slouží jako jedinečný identifikátor daného ransomwaru. Dále ransomware načte knihovny CryptoAPI *cryptsp.dll* a *rsaenh.dll*. Malware používá kryptografického poskytovatele *Microsoft Strong Cryptography Provider* ve variantě *PROV_RSA_FULL* (viz obrázek 3.8). Jedná se o obecně použitelný kryptografický modul[4]. Po načtení knihoven jsou do souboru zapsána pomocná data ransomwaru. Sled těchto událostí je znázorněn na obrázku 3.8.

Time of Day	Process Name	PID	Operation	Path	Result	Detail
21:24:07,8825416	Gl_dp.exe	2380	CreateFile	C:\Users\Public\15AA54916B492125CDE4BF363E94DF0B805EBFF2C71A...	SUCCESS	Desired Access: Generi
21:24:07,8828069	Gl_dp.exe	2380	CreateFile	C:\Users\Jim\Desktop\CRYPTSP.dll	NAME NOT FOUND	Desired Access: Read A
21:24:07,8829605	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\cryptsp.dll	SUCCESS	Desired Access: Read A
21:24:07,8831309	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\cryptsp.dll	SUCCESS	Desired Access: Read D
21:24:07,8833348	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\cryptsp.dll	SUCCESS	Image Base: 0x74bc000
21:24:07,8834658	Gl_dp.exe	2380	RegOpenKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provi...	SUCCESS	Desired Access: Read
21:24:07,8835483	Gl_dp.exe	2380	RegOpenKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provi...	SUCCESS	Desired Access: Read
21:24:07,8837503	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read A
21:24:07,8839111	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read D
21:24:07,8841912	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read A
21:24:07,8843490	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read D
21:24:07,8846191	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read A
21:24:07,8847720	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read D
21:24:07,8850677	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read A
21:24:07,8852210	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read D
21:24:07,8856001	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read A
21:24:07,8857636	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Generi
21:24:07,8884768	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read A
21:24:07,8886273	Gl_dp.exe	2380	CreateFile	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Desired Access: Read D
21:24:07,8888113	Gl_dp.exe	2380	Load Image	C:\Windows\SysWOW64\rsaenh.dll	SUCCESS	Image Base: 0x74b1000
21:24:07,8889251	Gl_dp.exe	2380	RegOpenKey	HKLM\System\CurrentControlSet\Control\Lsa\FipsAlgorithmPolicy	REPARSE	Desired Access: Query
21:24:07,8889381	Gl_dp.exe	2380	RegOpenKey	HKLM\System\CurrentControlSet\Control\Lsa\FipsAlgorithmPolicy	SUCCESS	Desired Access: Query
21:24:07,8889667	Gl_dp.exe	2380	RegOpenKey	HKLM\System\CurrentControlSet\Control\Lsa	REPARSE	Desired Access: Query
21:24:07,8889754	Gl_dp.exe	2380	RegOpenKey	HKLM\System\CurrentControlSet\Control\Lsa	SUCCESS	Desired Access: Query
21:24:07,8890176	Gl_dp.exe	2380	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Policies\Microsoft\Cryptography\Config...	REPARSE	Desired Access: Query
21:24:07,8890263	Gl_dp.exe	2380	RegOpenKey	HKLM\System\CurrentControlSet\Policies\Microsoft\Cryptography\Configur...	NAME NOT FOUND	Desired Access: Query
21:24:07,8890579	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Policies\Microsoft\Cryptography	SUCCESS	Desired Access: Read
21:24:07,8891148	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Microsoft\Cryptography	SUCCESS	Desired Access: Read
21:24:07,8891889	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Wow6432Node\Microsoft\Cryptography\Offload	NAME NOT FOUND	Desired Access: Read
21:24:08,0158698	Gl_dp.exe	2380	WriteFile	C:\Users\Public\15AA54916B492125CDE4BF363E94DF0B805EBFF2C71A...	SUCCESS	Offset: 0, Length: 258, P
21:24:08,0160713	Gl_dp.exe	2380	RegOpenKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provi...	SUCCESS	Desired Access: Read
21:24:08,0161321	Gl_dp.exe	2380	RegOpenKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provi...	SUCCESS	Desired Access: Read
21:24:08,0162191	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Microsoft\Cryptography	SUCCESS	Desired Access: Read
21:24:08,0162980	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Wow6432Node\Microsoft\Cryptography\Offload	NAME NOT FOUND	Desired Access: Read
21:24:08,0168473	Gl_dp.exe	2380	WriteFile	C:\Users\Public\15AA54916B492125CDE4BF363E94DF0B805EBFF2C71A...	SUCCESS	Offset: 258, Length: 768

Obr. 3.8: Načtení knihoven CryptoAPI a vytvoření pomocných souborů

Šifrování uživatelských dat

Po inicializaci programu dojde k šifrování dat. V prvním kroku malware provede rekurzivní dotaz na obsah připojených diskových jednotek a následně zašifruje všechny jednotlivé soubory. Proces šifrování je identický pro každou složku, která není v seznamu výjimek, a probíhá v těchto krocích (viz 3.9):

1. Načtení informací o souboru
2. Volání funkcí CryptoAPI
3. Načtení a zašifrování souboru
4. Vložení řetězce „personal ID“ na konec souboru
5. Uzavření souboru a přidání koncovky .LIN+
6. Vložení/kontrola přítomnosti informačního HTML souboru ve složce

[4] `https://docs.microsoft.com/cs-cz/windows/desktop/SecCrypto/prov-rsa-full`

21:24:10,0240221	Gl_dp.exe	2380	CreateFile	C:\Users\Jim\Documents\CV_Jim.pdf
21:24:10,0242212	Gl_dp.exe	2380	QueryStandardInfor...	C:\Users\Jim\Documents\CV_Jim.pdf
21:24:10,0242383	Gl_dp.exe	2380	QueryStandardInfor...	C:\Users\Jim\Documents\CV_Jim.pdf
21:24:10,0242673	Gl_dp.exe	2380	RegQueryKey	HKLM
21:24:10,0242865	Gl_dp.exe	2380	RegOpenKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001
21:24:10,0243091	Gl_dp.exe	2380	RegSetInfoKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001
21:24:10,0243190	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001\Name
21:24:10,0243296	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001\Name
21:24:10,0243383	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001\Name
21:24:10,0243449	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001\Name
21:24:10,0243555	Gl_dp.exe	2380	RegCloseKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider Types\Type 001
21:24:10,0243657	Gl_dp.exe	2380	RegQueryKey	HKLM
21:24:10,0243766	Gl_dp.exe	2380	RegOpenKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider
21:24:10,0243886	Gl_dp.exe	2380	RegSetInfoKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider
21:24:10,0243958	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\T
21:24:10,0244031	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\I
21:24:10,0244097	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\I
21:24:10,0244166	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\I
21:24:10,0244241	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider\I
21:24:10,0244533	Gl_dp.exe	2380	RegQueryKey	HKLM
21:24:10,0244618	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Microsoft\Cryptography
21:24:10,0244729	Gl_dp.exe	2380	RegSetInfoKey	HKLM\SOFTWARE\Microsoft\Cryptography
21:24:10,0244801	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Microsoft\Cryptography\MachineGuid
21:24:10,0244889	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Microsoft\Cryptography\MachineGuid
21:24:10,0244967	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Microsoft\Cryptography\MachineGuid
21:24:10,0245036	Gl_dp.exe	2380	RegQueryValue	HKLM\SOFTWARE\Microsoft\Cryptography\MachineGuid
21:24:10,0245169	Gl_dp.exe	2380	RegCloseKey	HKLM\SOFTWARE\Microsoft\Cryptography
21:24:10,0245271	Gl_dp.exe	2380	RegQueryKey	HKLM
21:24:10,0245371	Gl_dp.exe	2380	RegOpenKey	HKLM\Software\Wow6432Node\Microsoft\Cryptography\Offload
21:24:10,0245660	Gl_dp.exe	2380	RegCloseKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Cryptography\Defaults\Provider\Microsoft Strong Cryptographic Provider

Obr. 3.9: Otevření souboru a identifikace poskytovatele kryptografických služeb

21:24:10,0247768	Gl_dp.exe	2380	WriteFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 25 320, Length: 896, I/C
21:24:10,0247978	Gl_dp.exe	2380	ReadFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 24 576, Length: 744, I/C
21:24:10,0249041	Gl_dp.exe	2380	WriteFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 24 576, Length: 4 096, I
21:24:10,0252676	Gl_dp.exe	2380	ReadFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 0, Length: 25 320
21:24:10,0252838	Gl_dp.exe	2380	ReadFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 0, Length: 24 576, I/O F
21:24:10,0327581	Gl_dp.exe	2380	WriteFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 0, Length: 25 320, I/O F
21:24:10,0329839	Gl_dp.exe	2380	WriteFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 0, Length: 28 672, I/O F
21:24:10,0331022	Gl_dp.exe	2380	QueryStandardInfor...	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	AllocationSize: 28 672, EndOfl
21:24:10,0331197	Gl_dp.exe	2380	WriteFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 26 216, Length: 48, I/O
21:24:10,0331308	Gl_dp.exe	2380	WriteFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Offset: 24 576, Length: 4 096, I
21:24:10,0333197	Gl_dp.exe	2380	CloseFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	
21:24:10,0336837	Gl_dp.exe	2380	CreateFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Desired Access: Read Attribut
21:24:10,0337048	Gl_dp.exe	2380	QueryAttributeTagFile	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	Attributes: A, ReparseTag: 0x(
21:24:10,0337201	Gl_dp.exe	2380	QueryBasicInformati...	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	CreationTime: 10. 11. 2018 17
21:24:10,0337750	Gl_dp.exe	2380	CreateFile	C:\Users\Jim\Documents	SUCCESS	Desired Access: Write Data/Ac
21:24:10,0338081	Gl_dp.exe	2380	SetRenameInformati...	C:\Users\Jim\Documents\CV_Jim.pdf	SUCCESS	ReplaceIfExists: True, FileNan
21:24:10,0339869	Gl_dp.exe	2380	CloseFile	C:\Users\Jim\Documents	SUCCESS	
21:24:10,0340333	Gl_dp.exe	2380	CloseFile	C:\Users\Jim\Documents\CV_Jim.pdf.LIN+	SUCCESS	
21:24:10,0342884	Gl_dp.exe	2380	CreateFile	C:\Users\Jim\Documents\how_to_back_files....	SUCCESS	Desired Access: Read Attribut
21:24:10,0343043	Gl_dp.exe	2380	QueryBasicInformati...	C:\Users\Jim\Documents\how_to_back_files....	SUCCESS	CreationTime: 28. 4. 2019 21:2
21:24:10,0343122	Gl_dp.exe	2380	CloseFile	C:\Users\Jim\Documents\how_to_back_files....	SUCCESS	

Obr. 3.10: Zápis šifrovaného souboru

Ukončení programu

Po dokončení šifrování souborů na disku je v proměnné *%TEMP%* vytvořen dávkový soubor. Formát názvu tohoto souboru je *tmp****.bat*, se čtyřmi náhodnými alfanumerickými znaky místo hvězdiček. Obsah tohoto souboru znázorňuje obrázek B.2. Tento soubor smaže stínové kopie operačního systému a známé systémy v aplikaci Vzdálená plocha a vyčistí protokol událostí Windows. Následně je ransomware ukončen.

Šifrování dat a generování klíčů

K rekonstrukci dalších funkcí programu bylo nutné provést dekompilaci a debugging kódu. K tomuto účelu byl použit nástroj Interactive Disassembler (IDA)[5] ve verzi Freeware. Pro debugging byl zvolen nástroj x64dbg[6].

Nejprve program vygeneruje 1024 bitů dlouhý řetězec. Tento řetězec se vytváří funkcí *rsa_keygen* v programu. Ukázka náhodně vygenerovaného řetězce je na na obrázku 3.11. K vygenerování tohoto řetězce je použita funkce *CryptoGenRandom* (viz 3.12), pomocí kterých program získává z operačního systému kryptograficky náhodná čísla. Funkce *CryptAcquireContextW* a *CryptReleaseContext* slouží k obsluze Crypto API.

Pro šifrování uživatelských dat program používá algoritmus AES. Ani v tomto případě program nevyužívá funkce operačního systému, ale sám přímo implementuje algoritmus AES v kódu programu. Zajímavostí je, že program obsahuje funkce jak pro šifrování, tak pro dešifrování (na obrázku 3.15). Dešifrovací větev slouží k extrahování informačního HTML souboru a šifrovací větev umožňuje útok na uživatelská data.

Generování klíče k šifře AES je prováděno v programu. Ke generování se používá řetězec vytvořený pomocí funkce *rsa_genkey*, který je 8192× zhašován pomocí tří hašovacích funkcí (viz obrázek 3.14).

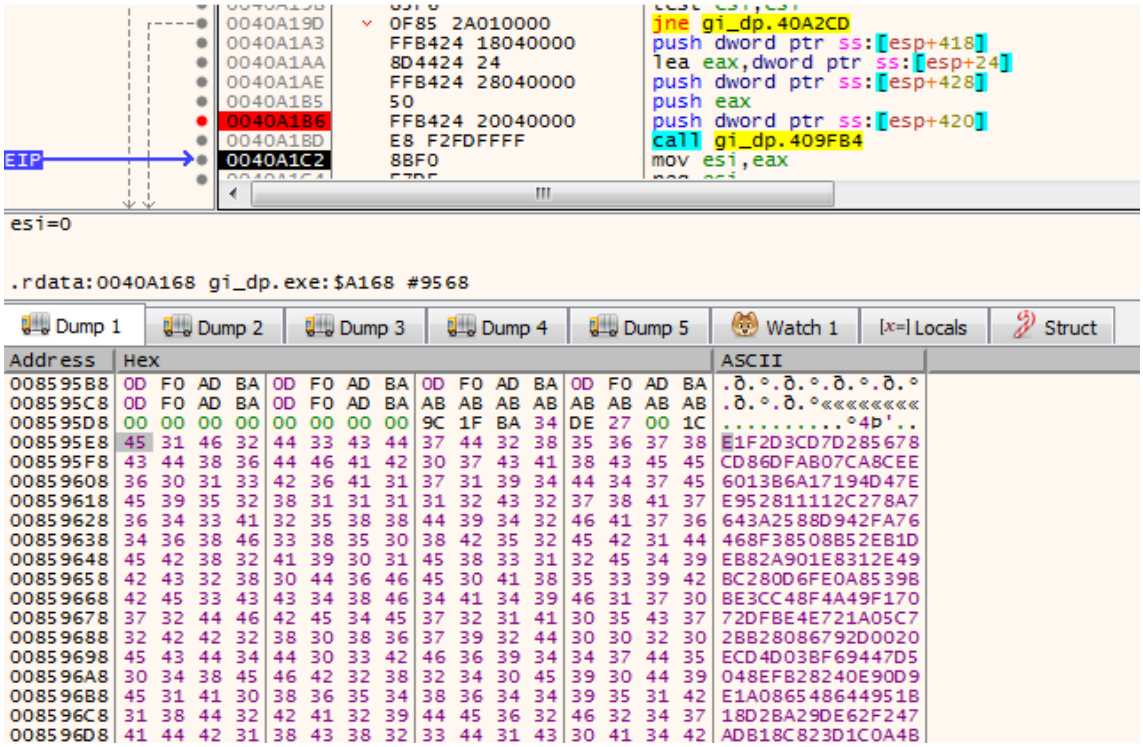
Práce se šifrovacími klíči

Proces generování klíčů je zobrazen na obrázku 3.16 a probíhá takto:

1. Nejprve se pomocí funkce CryptoAPI operačního systému vygeneruje náhodný řetězec.
2. Tento řetězec je transformován pomocí funkce na obrázku 3.14 do podoby klíče k symetrické šifře. Tímto klíčem se zašifrují uživatelská data.

[5] `https://www.hex-rays.com/products/ida/`

[6] `https://x64dbg.com/`



Obr. 3.11: Ukázka vygenerovaného řetězce

3. Řetězec uvedený v prvním kroku je za použití kryptosystému RSA transformován do podoby řetězce „personal ID“, který je zobrazen uživateli a zapsán do zašifrovaných souborů.

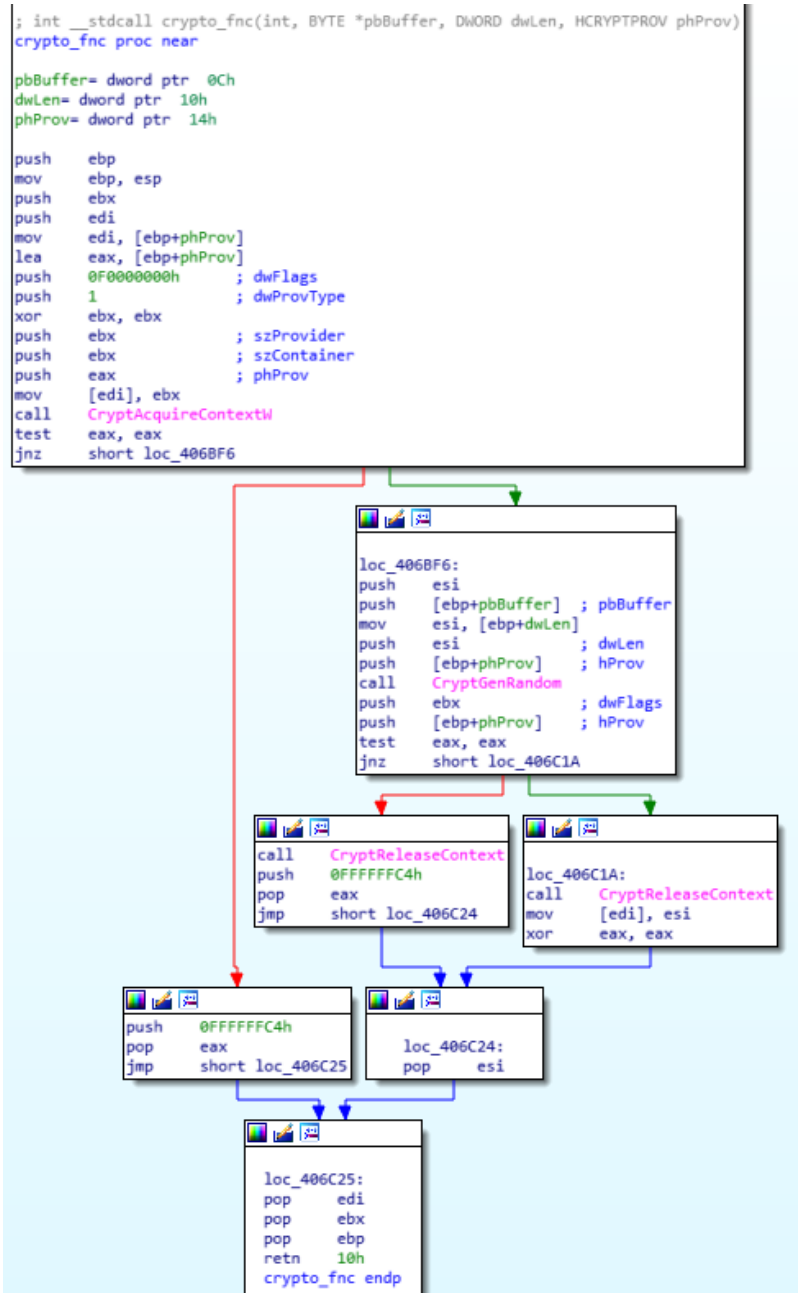
Lokálně vygenerovaný řetězec zároveň zůstává v napadeném zařízení v konfiguračním souboru. V případě, že je škodlivý kód znovu spuštěn (například po restartu zařízení), se nová data zašifrují stejným klíčem a řetězec „personal ID“ zůstává stejný. Jedinečný identifikátor zůstane beze změny i při přenosu konfiguračního souboru do jiného zařízení. Lokálně vygenerovaný řetězec, respektive konfigurační soubor, je tedy jedinou slabinou analyzovaného vzorku a jeho znalost umožňuje zpětně vypočítat použité šifrovací klíče.

Možnosti dešifrování dat

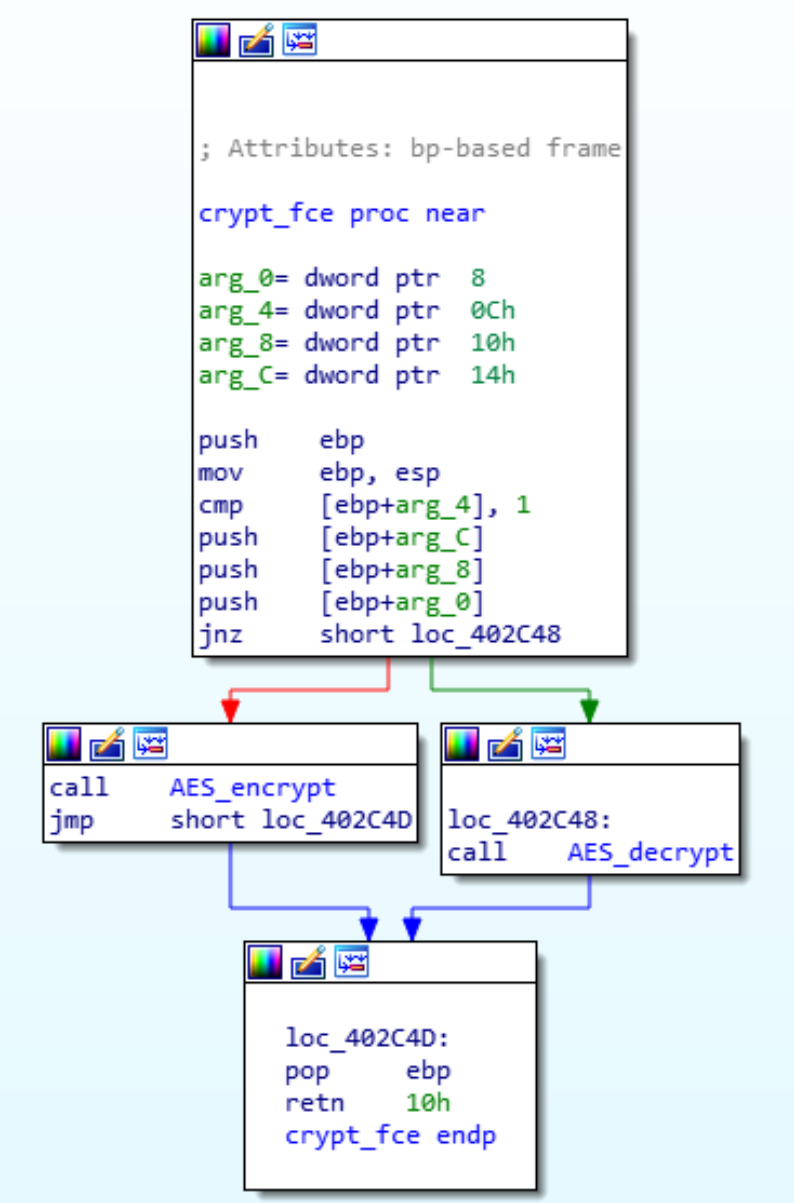
Na analyzovaný vzorek v současné době neexistuje dostupný dešifrovací nástroj. V rámci analýzy byly vyzkoušeny dekryptory od společnosti Emsisoft[7] a TrendMicro [8]. Žádný z testovaných dekryptorů na zašifrovaná data nefungoval.

[7] https://www.emsisoft.com/decrypter/

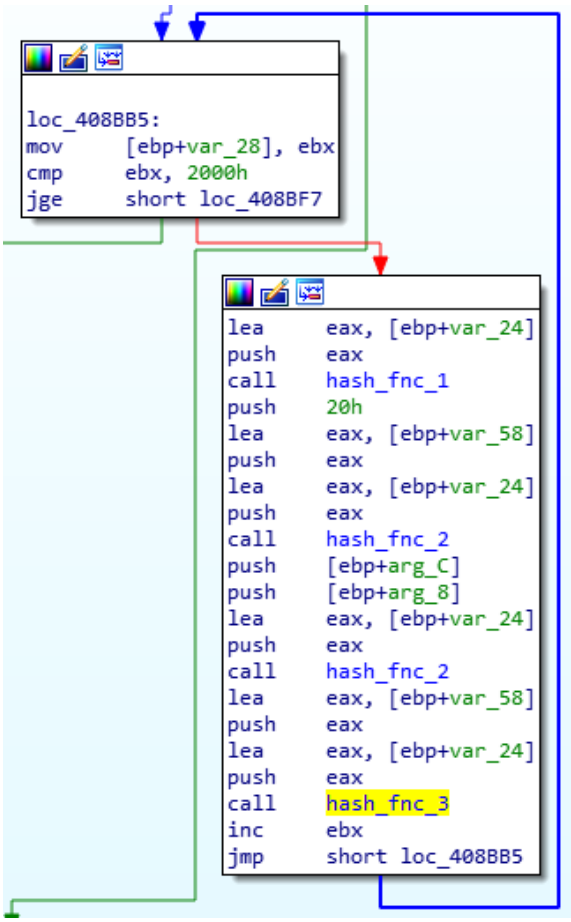
[8] https://success.trendmicro.com/solution/1114221-downloading-and-using-the-trend-micro-ransomware-file-decryptor



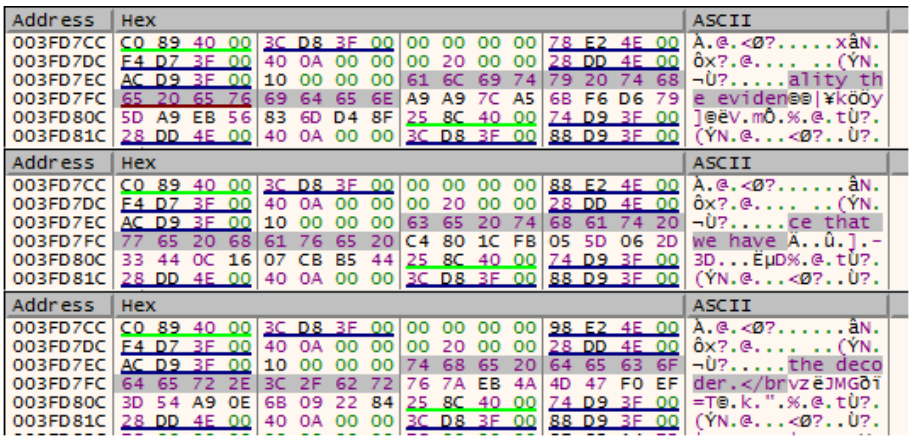
Obr. 3.12: Funkce pro generování náhodných čísel



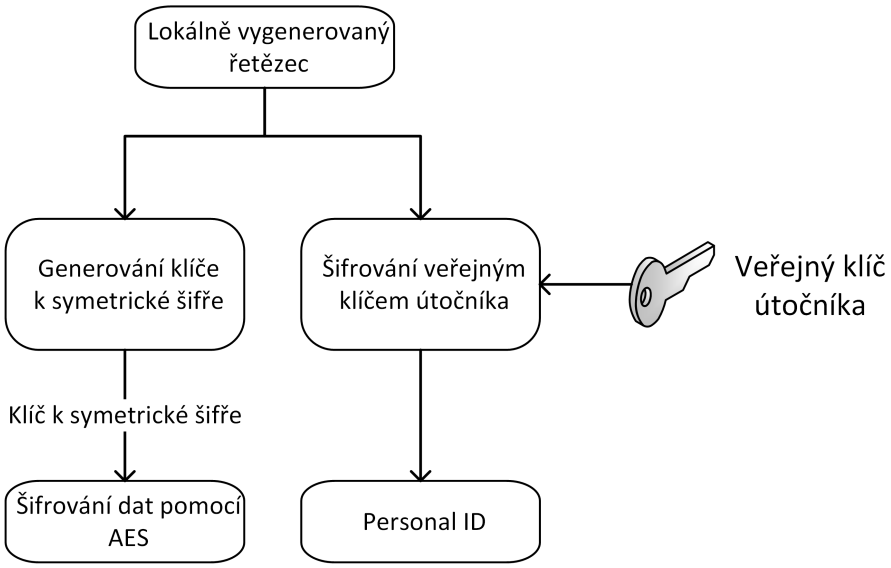
Obr. 3.13: Funkce pro šifrování a dešifrování dat



Obr. 3.14: Generování klíče k symetrické šifře AES



Obr. 3.15: Dynamické generování informačního HTML souboru

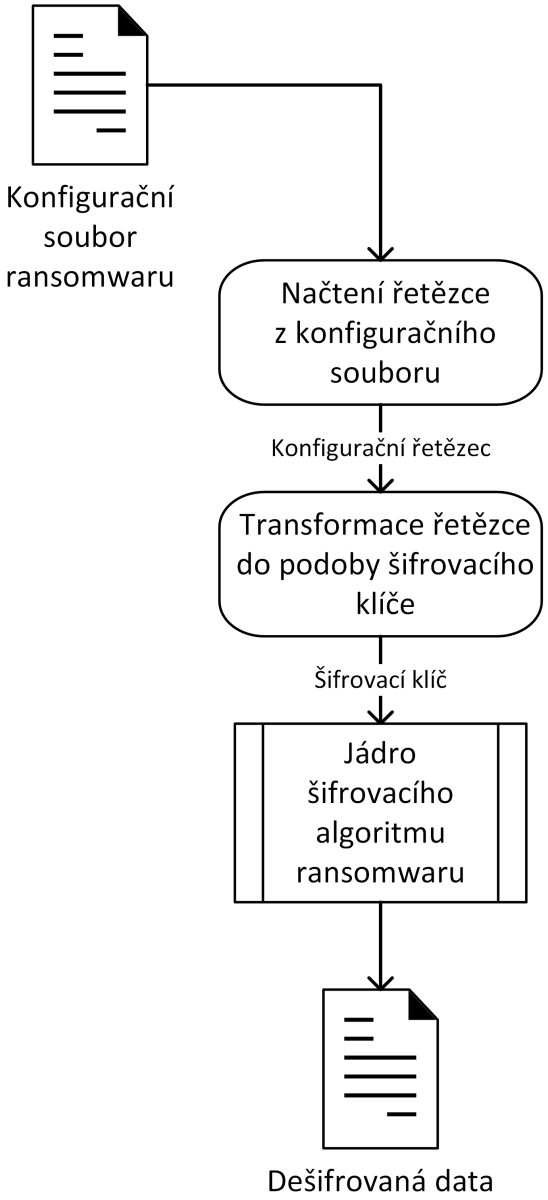


Obr. 3.16: Schéma práce s šifrovacími klíči

V rámci práce byl také realizován pokus o kontaktování tvůrců ransomwaru prostřednictvím e-mailu (na adrasu uvedenou v informační zprávě). Uvedené e-maily jsou s největší pravděpodobností stále funkční (e-mail nebyl vrácen jako nedoručitelný), ale odpověď od útočníků nedorazila.

Možnosti dešifrování jednotlivých souborů jsou u analyzovaného vzorku omezené. Vzhledem k tomu, že jako šifrovací algoritmy jsou použity šifry RSA a AES, není možné data dešifrovat v reálném čase. Jedinou slabinou použitého kódu je skutečnost, že v případě, že se v napadeném systému nachází konfigurační soubor, údaje z něj se automaticky načtou.

Dekryptor by tedy bylo možné realizovat tak, že jako vstup by byl použit konfigurační soubor vzorku, ze kterého by se stejným způsobem jako v programu vygenerovaly potřebné šifrovací klíče. Ty by se následně použily k dešifrování uživatelských dat. K realizaci navrženého dekryptoru je třeba použít funkci odpovídající té, který byla použita v ransomwaru. Schéma navrženého dekryptoru je na obrázku 3.17.



Obr. 3.17: Schéma dekryptoru

4 Závěr

Z výsledků analýzy je možné zrekonstruovat běh programu. Program po spuštění vygeneruje lokálně řetězec, který je transformován do podoby klíče k symetrickému kryptosystému. Tento řetězec se zašifruje vestavěným klíčem asymetrického kryptosystému RSA do podoby řetězce „personal ID“, který je zobrazen uživateli. Vzhledem k zabezpečení kryptosystémem RSA nelze potřebný řetězec z „personal ID“ v reálném čase získat.

Jako symetrická šifra je v programu použit algoritmus AES, který je implementován přímo v programu a pro svoji funkci nevyužívá CryptoAPI operačního systému. Tato šifra je použita jak pro rozbalení informační zprávy pro uživatele, tak pro šifrování uživatelských dat. V programu je také použitý kryptografický poskytovatel *Microsoft Strong Cryptography Provider* a na základě výše provedené analýzy je použit pouze pro generování náhodných čísel, ale ne pro samotné šifrování dat. Tento kryptografický poskytovatel byl zvolen s největší pravděpodobností z důvodu kompatibility se staršími verzemi operačního systému Windows (tento poskytovatel je implementován v operačních systémech Windows XP a novějších). U analyzovaného vzorku nebyla zjištěna jakákoliv síťová aktivita. Malware se tedy sám dále nešíří a nevyužívá zranitelností aplikačního softwaru.

Byla také provedena analýza zařízení, ze kterého byl extrahován analyzovaný vzorek. Zařízení bylo s největší pravděpodobností napadeno prostřednictvím slabého hesla u administrátorského uživatelského účtu v kombinaci s otevřeným přihlašováním pomocí aplikace Vzdálená plocha. Potvrzení této teorie není možné, protože po útoku zůstalo zařízení nefunkční z důvodu poškozeného operačního systému.

V závěru práce je nastíněna možnost realizace dekryptoru s využitím konfiguračního souboru ransomwaru.

Literatura

[1] World's biggest shipper: cyberattack cost up to $300 million. *Phys.org* [online]. USA: Science X, 2018, August 16, 2017 [cit. 2018-12-09]. Dostupné z: https://phys.org/news/2017-08-moller-maersk-cyberattack-million.html

[2] CIMPANU, Catalin. Maersk Reinstalled 45,000 PCs and 4,000 Servers to Recover From NotPetya Attack. In: *BleepingComputer.com* [online]. USA: Bleeping Computer, 2018, 10 Jan 2018 [cit. 2018-12-09]. Dostupné z: https://www.bleepingcomputer.com/news/security/maersk-reinstalled-45-000-pcs-and-4-000-servers-to-recover-from-notpetya-attack/

[3] SOLSVIK, Terje, FOUCHE, Gwladys a Jane MERRIMAN, ed. Norsk Hydro's initial loss from cyber attack may exceed $40 million. *Reuters* [online]. UK: Thomson Reuters, 2019 [cit. 2019-05-14]. Dostupné z: https://www.reuters.com/article/us-norway-cyber/norsk-hydros-initial-loss-from-cyber-attack-may-exceed-40-million-idUSKCN1R71X9

[4] SAVAGE, Kevin, Peter COOGAN a Hon LAU. The evolution of ransomware. In: *Symantec Corporation* [online]. Mountain View, CA 94043 USA: Symantec Corporation, 2015 [cit. 2018-12-01]. Dostupné z: http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf

[5] Bab.la - Internetový slovník. *Bab.la - Internetový slovník* [online]. Hamburk: bab.la, 2018 [cit. 2018-12-01]. Dostupné z: https://cs.bab.la/slovnik/cesky-anglicky/%C5%A1kodliv%C3%BD-software

[6] Malicious programs. *Kaspersky Lab Encyclopedia* [online]. 2018: Kaspersky, 2018 [cit. 2018-12-01]. Dostupné z: https://encyclopedia.kaspersky.com/knowledge/malicious-programs/

[7] KLEIN, Andy. Computer Backup Awareness in 2018: Getting Better and Getting Worse. *Backblaze* [online]. USA: Backblaze, 2018, June 21st, 2018 [cit. 2018-12-01]. Dostupné z: https://www.backblaze.com/blog/computer-backup-awareness-in-2018/

[8] CRACIUN, Vlad Constantin, Andrei MOGAGE a Emil SIMION. Trends in design of ransomware viruses. In: *Cryptology ePrint Archive: Report 2018/598* [online]. N/A: Cryptology ePrint Archive, 2018, 2018 [cit. 2018-12-01]. Dostupné z: https://eprint.iacr.org/2018/598

[9] SIKORSKI, Michael a Andrew HONIG. *Practical malware analysis: the hands-on guide to dissecting malicious software*. San Francisco: No Starch Press, c2012. ISBN 978-1-59327-290-6.

[10] KENDALL, Kris. Practical Malware Analysis. In: *Blackhat.com* [online]. USA, 2007 [cit. 2018-12-08]. Dostupné z: https://www.blackhat.com/presentations/bh-dc-07/Kendall_McMillan/Paper/bh-dc-07-Kendall_McMillan-WP.pdf

[11] FENTON, Caleb. Anti-VM tricks. *Sentinel One* [online]. USA: Sentinel One, 2016 [cit. 2018-12-08]. Dostupné z: https://www.sentinelone.com/blog/anti-vm-tricks/

[12] What is Cuckoo?. *Cuckoo Sandbox Book* [online]. Web: Cuckoo Foundation, 2018 [cit. 2018-12-08]. Dostupné z: https://cuckoo.sh/docs/introduction/what.html

[13] AES instruction set. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2019-04-21]. Dostupné z: https://en.wikipedia.org/wiki/AES_instruction_set

[14] BURDA, Karel. *Bezpečnost informačních systémů*. Brno: Vysoké učení technické v Brně, 2013. ISBN 978-80-214-4890-2.

[15] Sandbox (computer security). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2019-04-25]. Dostupné z: https://en.wikipedia.org/wiki/Sandbox_(computer_security)

[16] AQUILINA, James M., Eoghan CASEY a Cameron H. MALIN. *Malware forensics: investigating and analyzing malicious code*. Burlington, MA: Syngress Pub., c2008. ISBN 978-1-59749-268-3.

[17] Decompiler. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2019-05-14]. Dostupné z: https://en.wikipedia.org/wiki/Decompiler

[18] Ladění (programování). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2019-05-14]. Dostupné z: https://cs.wikipedia.org/wiki/Lad%C4%9Bn%C3%AD_(programov%C3%A1n%C3%AD)

Seznam symbolů, veličin a zkratek

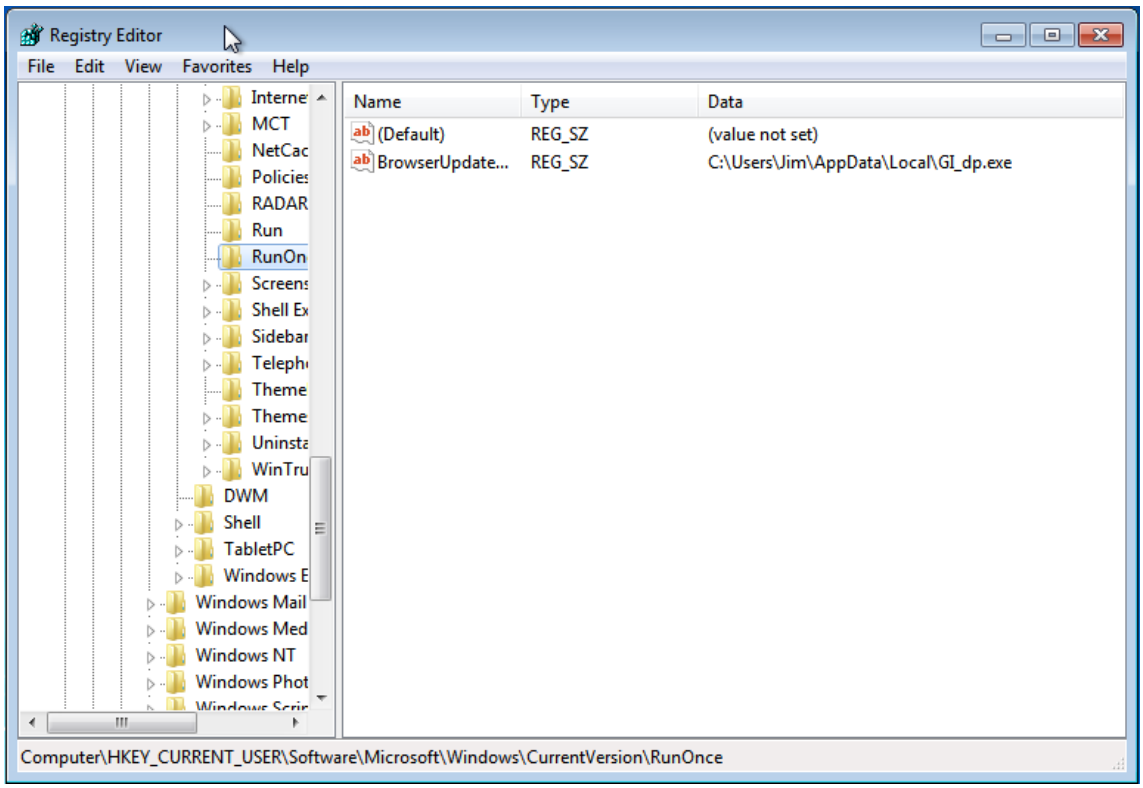
AES	Advanced Encryption Standard
API	Application Programming Interface
C&C	Command & Control
DoS	Denial of Service
ECDH	Elliptic-curve Diffie–Hellman
FTP	File Transfer Protocol
ICQ	I Seek You – komunikační program
IRC	Internet Relay Chat
MD5	Message-Digest
P2P	Peer to Peer
RC4	Rivest Cipher 4
RSA	Rivest, Shamir, Adleman
SHA	Secure Hash Algorithm
SSL	Secure Sockets Layer

Seznam příloh

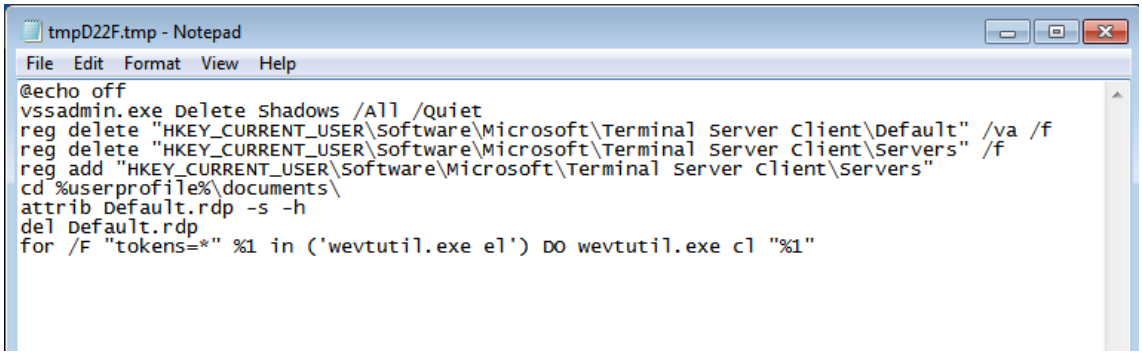
A Seznam vyloučených složek

Wz Jc
Windows
Microsoft
Microsoft Help
Windows App Certification Kit
Windows Defender
COMODO
Windows NT
Windows Kits
Windows Mail
Windows Media Player
Windows Multimedia Platform
Windows Phone Kits
Windows Phone Silverlight Kits
Windows Photo Viewer
Windows Portable Devices
Windows Sidebar
WindowsPowerShell
NVIDIA Corporation
Microsoft.NET
Internet Explorer
Kaspersky Lab
McAfee
Avira
spytech software
sysconfig
Avast
Dr.Web
Symantec
Symantec_Client_Security
system volume information
Microsoft Shared
Common Files
Outlook Express
Movie Maker
Chrome
Mozilla Firefox
Opera
YandexBrowser
ntldr
ProgramData

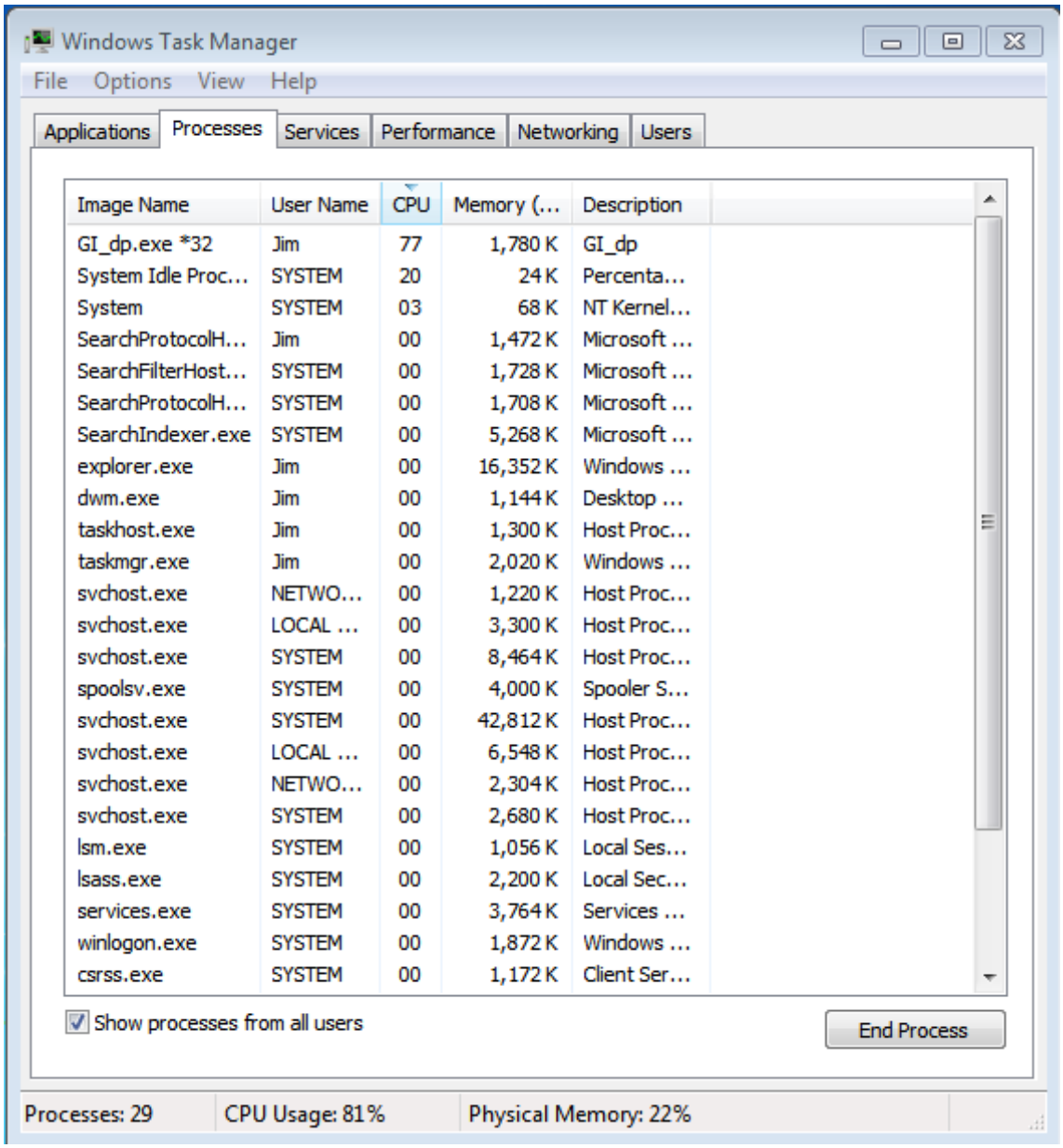
B Snímky obrazovky po útoku ransomwarem



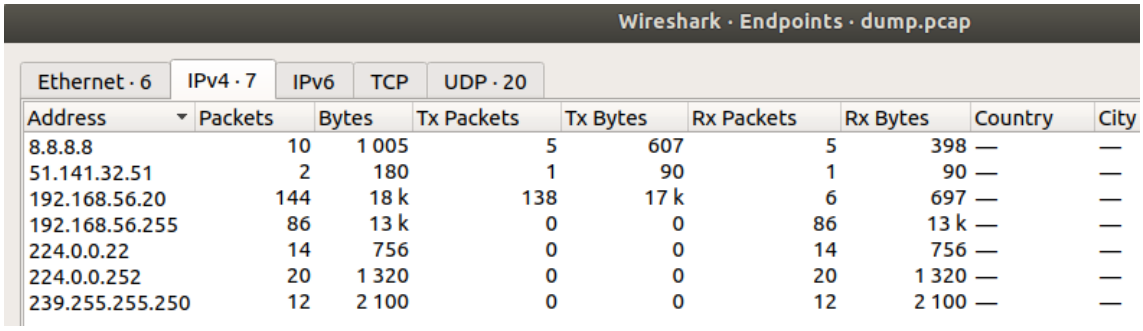
Obr. B.1: Hodnota zapsaná do registru operačního systému



Obr. B.2: Dávkový soubor ke smazání stínových kopií a protokolu událostí



Obr. B.3: Vytížení procesoru během šifrování



Obr. B.4: Analýza síťového provozu – výpis IP adres z programu Wireshark

C Obsah přiloženého DVD

- /
  - DP ..... Složka s elektronickou verzí práce
    - DP-xproch67.pdf ..... Elektronická verze práce
  - Log ..... Složka s log soubory z programu ProcessMonitor
    - PhysicalHW ..... Složka s logy z fyzického stroje
      - ProcessMonitor_Log_CSV.zip ..... Log ve formátu CSV
      - ProcessMonitor_Log_PML.zip ..... Log ve formátu PML
    - VM ..... Složka s logy z virtuálního stroje
      - ProcessMonitor_Log_CSV.zip ..... Log ve formátu CSV
      - ProcessMonitor_Log_PML.zip ..... Log ve formátu PML
  - Vzorek ..... Složka s vzorkem ransomwaru GlobeImposter
    - GI_vzorek.zip ..... Analyzovaný vzorek ransomwaru
  - INFO.txt ..... Informační soubor k DVD