PROJEKT INŻYNIERSKI

Badanie bezpieczeństwa systemów uwierzytelniania w domenie *Active Directory*

Bartłomiej Adamski
275244
Paweł Zapiór
294034

Kierunek: informatyka
Specjalność: Inżynieria Analizy Danych, Sieci Komputerowe: Bezpieczeństwo i Zarządzanie

PROWADZĄCY PRACĘ:
dr inż. Adrian Kapczyński
WYDZIAŁ MATEMATYKI STOSOWANEJ

GLIWICE 2022

Tytuł pracy: Badanie bezpieczeństwa systemów uwierzytelniania w domenie *Active Directory*

Streszczenie:

Tematem pracy jest badanie bezpieczeństwa systemów uwierzytelniania w domenie *Active Directory*. Celem jest zidentyfikowanie potencjalnych zagrożeń dla bezpieczeństwa uwierzytelniania do systemów teleinformatycznych, w których urządzenia są podłączone do domeny *Active Directory*. W celu realizacji pracy wdrożono przykładową infrastrukturę oraz przeprowadzono jej testy bezpieczeństwa.

W wyniku analizy zidentyfikowano następujące zagrożenia dla bezpieczeństwa uwierzytelniania w domenie *Active Directory*: ataki typu *brute force*, stosowanie słabych haseł, ataki na systemy rozwiązywania nazw oraz ataki wykorzystujące nieprawidłową konfigurację protokołu Kerberos. Analiza skuteczności obecnych mechanizmów ochrony wskazała, że istniejące rozwiązania są skuteczne w zapobieganiu większości zidentyfikowanych zagrożeń, jednak wymagają od administratora odpowiedniej konfiguracji oraz dostosowania systemu do specyfiki danej organizacji.

W pracy zaproponowano działania naprawcze, które po wdrożeniu zwiększyły poziom bezpieczeństwa wdrożonej infrastruktury. Zalecenia te mogą być jednak rozważane jako bardziej uniwersalne zestawienie dobrych praktyk projektowania systemu opartego na usłudze *Active Directory*.

Słowa kluczowe:

cyberbezpieczeństwo, uwierzytelnianie, testy penetracyjne, kerberos, *Active Directory*, Microsoft, Windows, eskalacja uprawnień.

Thesis title:

Investigation of the security of authentication systems in the *Active Directory* domain.

Abstract:

The subject of the study investigates the security of authentication systems in the *Active Directory* domain. The aim is to identify potential threats to the security of authentication to teleinformatics systems, in which devices are connected to the *Active Directory* domain. To realize the work, an exemplary infrastructure was implemented and its security tests were carried out.

As a result of the analysis, the following threats to the security of authentication in the *Active Directory* domain were identified: *brute force* attacks, the use of weak passwords, attacks on Domain Name System and attacks using incorrect configuration of the Kerberos protocol. The analysis of the effectiveness of current protection mechanisms indicated that existing solutions are effective in preventing most of the identified threats but appropriate configuration and adaptation of the system to the specific needs of the organization by the administrator is required.

The study also proposes corrective actions which increased the level of security of the implemented infrastructure. However the recommendations can be considered as a more universal set of good practices for designing a system based on the *Active Directory* service.

Keywords:

cybersecurity, authentication, penetration testing, kerberos, *Active Directory*, Microsoft, Windows, privilege escalation.

Spis treści

Wstęp

Cyberbezpieczeństwo wyzwaniem XXI wieku

Problem ochrony cyberprzestrzeni stał się jedynym z najgorętszych tematów ostatnich lat. Wynika to przede wszystkim z dwóch zdarzeń, przy czym stanowią one dla siebie logiczny ciąg przyczynowo-skutkowy. Wszystko zaczyna się od wynalezienia komputera, Internetu i postępującej informatyzacji, które determinowały istotne przemiany społeczne. W rezultacie powstała koncepcja „społeczeństwa informacyjnego". Rozwinął ją Daniell Bell, stawiając tezę, że dla nowego społeczeństwa strategicznymi zasobami stały się wiedza i informacja, zastępując w tej roli pracę i kapitał [1].

Postępująca cyfryzacja oraz łatwość przetwarzania ogromnych ilości danych są niespotykanym dotąd w historii świata zjawiskiem i stawiają przed ludzkością nowe, niespotykane dotąd w historii wyzwania. Faktem niezaprzeczalnym jest nadanie olbrzymiego znaczenia „informacji". Jeśli kluczowym zasobem stały się dane i są one przetwarzane na skalę masową za pośrednictwem Internetu to naturalną konsekwencją są próby ataków na systemy informatyczne celem przejęcia bądź pozbawienia prawowitego właściciela dostępu do cennych danych. Wprowadźmy tutaj pojęcia charakterystyczne dla branży cyberbezpieczeństwa. Haker (ang. *hacker*) to osoba, której celem będzie uzyskanie nieautoryzowanego dostępu do systemu teleinformatycznego. Nie każdy haker będzie jednak przestępcą. Wyróżnia się:

- Hakerów w czarnym kapeluszu, którzy naruszają bezpieczeństwo systemów teleinformatycznych z pobudek, które moralnie można ocenić jako złe. Wyznacznikiem zazwyczaj jest motyw działania — haker w czarnym kapeluszu działa dla własnych korzyści.

- Hakerów w szarym kapeluszu, którzy naruszają bezpieczeństwo systemów teleinformatycznych, bez zgody osób odpowiedzialnych za te systemy, jednak ich motywacje moralne nie są tak łatwe do jednoznacznej oceny. Mogą być to aktywiści, którzy atakują systemy informatyczne państw łamiących prawa człowieka lub też nie wykorzystują odnalezionych błędów w bezpieczeństwie do własnych korzyści, zamiast tego zgłaszają je odpowiednim osobom i instytucjom, dzięki czemu dbają o wspólne dobro. Co należy podkreślić, nie działają oni jednak w zgodzie z prawem.

- Hakerów w białym kapeluszu, którzy działają zgodnie z prawem i na zlecenie właściciela systemu informatycznego, który chce przekonać się, czy jego zasoby są bezpieczne. Hakerzy w białym kapeluszu pracują jako **pentesterzy** lub członkowie ***Red Teamów*** (z ang. czerwonych zespołów), czyli zespołów, których celem jest symulowanie nielegalnych ataków hakerskich.

Wśród specjalistów od cyberbezpieczeństwa wyróżniamy także ich przeciwieństwo hakerów, czyli tzw. *blue team'y* — ten termin oznacza obrońców, których zadaniem jest wykonywanie prac związanych z zabezpieczeniem systemów teleinformatycznych przed atakami. W ten sposób dziedzinę cyberbezpieczeństwa można przedstawić jako ciągłą rywalizację hakerów i obrońców, jednocześnie w przypadku hakerów w białych (a czasem też szarych) kapeluszach prowadzi ona do zwiększenia ogólnego poziomu bezpieczeństwa w Internecie.

Motywacja do podjęcia tematu

My również chcemy przyczynić się do zwiększenia ogólnego poziomu bezpieczeństwa cyberprzestrzeni. Z pośród wielu aspektów cyberpezpieczeństwa zdecydowaliśmy, że chcemy przyjrzeć się bliżej mechanizmom uwierzytelniania w domenie *Active Directory*. Warto w tym miejscu zaznaczyć, że problem bezpieczeństwa może wynikać z wielu przyczyn. Czasem będzie to przestarzałe oprogramowanie, jednak nie to będzie dla nas najistotniejsze, gdyż naprawa takich błędów wymaga najczęściej jedynie zmiany wersji oprogramowania. Znacznie bardziej interesujące z naszej perspektywy są konfiguracje — konfigurując usługę *Active Directory* można bowiem popełnić bardzo wiele błędów, które umożliwią atakującym naruszenie bezpieczeństwa infrastruktury.

Cele projektu

Celem niniejszej pracy jest analiza bezpieczeństwa mechanizmów uwierzytelniania urządzeń podłączonych do domeny *Active Directory*. To specyficzna sytuacja, ponieważ uwierzytelnianie jest realizowane poprzez protokół Kerberos, a baza użytkowników jest wspólna dla wielu urządzeń. Wiążą się z tym pewne zagrożenia, które zostaną przedstawione w kolejnych rozdziałach. Aby uporządkować plan projektu, poniżej przedstawiono kamienie milowe pracy:

- Przedstawiona zostanie teoria związana z uwierzytelnianiem użytkowników w systemach informatycznych. Omówione zostanie działanie protokołu Kerberos,

bazy LDAP i usługi *Active Directory*. Zostaną zdefiniowane problemy związane z ich bezpieczeństwem, charakterystyczne dla systemów teleinformatycznych opartych o przynależność urządzeń do domeny.

- Zdefiniowane zostanie ryzyko, z którym wiąże się proces uwierzytelniania w domenie *Active Directory*. W tym celu przedstawiona zostanie zasada działania ataków, które mogą stanowić zagrożenie dla mechanizmu uwierzytelniania w domenie Active Directrory.

- Wdrożony zostanie system informatyczny oparty o domenę *Active Directory* w środowisku zwirtualizowanym.

- Przeprowadzone zostaną testy penetracyjne wdrożonej infrastruktury. Dzięki temu ukazane będą skutki, które z punktu widzenia bezpieczeństwa determinuje wadliwe konfiguracja domeny *Active Directory*.

- Dla każdej znalezionej w ramach testów penetracyjnych luki w bezpieczeństwie przedstawiona zostanie metoda naprawy. Stworzona w rezultacie zostanie w ten sposób infrastruktura oparta o *Active Directory* i charakteryzująca się wyższym poziomem bezpieczeństwa.

Układ i zakres pracy

Niniejsza praca składa się ze wstępu, 4 rozdziałów zawierających przedstawienie rozwiązania problemu badawczego wynikającego z tematu pracy oraz podsumowania.

- **Wstęp** zawiera wprowadzenie do problematyki pracy oraz uzasadnienie wyboru tematu. We wstępie zdefiniowane zostały też cele pracy, jej zakres i układ.

- **Rozdział 1: Analiza teoretyczna mechanizmów uwierzytelniania w domenie *Active Directory*** stanowi część teoretyczną niniejszej pracy. Przedstawione zostały ogólne zagadnienia związane z uwierzytelnianiem, a następnie omówiono najważniejsze dla usługi *Active Directory* protokoły **Kerberos** i **LDAP**.

- **Rozdział 2: Metodyka i zastosowane technologie** stanowi wprowadzenie do części praktycznej w postaci przedstawienia zastosowanych narzędzi i technologii oraz uzupełnienie dokumentacji projektu o przedstawienie aspektów związanych z organizacją pracy.

- **Rozdział 3. Wdrożenie infrastruktury opartej na usłudze *Active Directory*** stanowi dokumentacje etapu wdrożeniowego.
- **Rozdział 4. Testy bezpieczeństwa wdrożonej infrastruktury** stanowi dokumentację z testów penetracyjnych przeprowadzonych na wdrożonym środowisku. Każda technika ataku została przedstawiona, a następnie wykonana w ramach sekcji „Scenariusz ataku".
- **Podsumowanie** w podsumowaniu można znaleźć listę wykrytych podatności oraz listę zalecanych działań naprawczych. Na podstawie listy zalecanych działań naprawczych przeprowadzone zostały modyfikacje w konfiguracji wdrożonego środowiska. Sformułowano wnioski z całości pracy.

1. Analiza teoretyczna mechanizmów uwierzytelniania w domenie *Active Directory*

Część teoretyczna zostanie rozpoczęta od zdefiniowania przedmiotu rozważań. Dla porządku przedstawić należy więc formalną definicje słowa „uwierzytelnianie". Zanim jednak porozmawiamy o znaczeniu tego słowa dla Informatyki warto przytoczyć ogólną definicję językową.

1.1. Definicja

Definicja 1.1. (Uwierzytelnianie - Słownik Języka Polskiego [2]) *„uwierzytelnić — uwierzytelniać*

1. *«uczynić coś wiarygodnym»*
2. *«stwierdzić autentyczność dokumentu lub podpisu, zgodność z prawem jakiejś czynności prawnej»*
3. *«zaopatrzyć kogoś w dokumenty stwierdzające powierzenie mu funkcji dyplomatycznej»".*

Należy tutaj skupić się na pierwszym znaczeniu podawanym przez SJP — „uczynić coś wiarygodnym". Co zatem w systemie informatycznym należy uczynić wiarygodnym? Oczywiście tożsamość. Z punktu widzenia Informatyki celem uwierzytelniania standardowo jest przedstawienie dowodu na to, że osoba uwierzytelniająca się jest tą osobą, za którą się podaje. Bardziej techniczną definicję znajdziemy na wielu stronach Internetowych. Warto przyjrzeć się jednej z nich:

Definicja 1.2. (Uwierzytelnianie - Microsoft [3]) *„Gdy logujesz się do swoich kont online —– proces, który nazywamy „uwierzytelnianiem", udowodniasz w usłudze, że jesteś tym, kim jesteś".*

tłumaczy na swojej oficjalnej stronie firma Microsoft. Co warto podkreślić, nie jest to równoznaczne z przyznaniem jakiegokolwiek dostępu użytkownikowi w usłudze do które się uwierzytelnił. Przyznanie odpowiednich dostępów jest osobnym procesem

nazywanym **autoryzacją**. Są to zagadnienia powiązane, ponieważ autoryzacja jest realizowana zazwyczaj na bazie rezultatu procesu uwierzytelniania. Aby udzielić prawidłowych dostępów system teleinformatyczny może potrzebować wiarygodnej informacji o tożsamości użytkownika. Na tej podstawie przydziela odpowiedni dostęp bądź odmawia dostępu. Oznacza to, że mimo skutecznego uwierzytelnienia się do systemu informatycznego użytkownik może nie uzyskać dostępu do żądnego zasobu, ponieważ nie uda się przejść pomyślnie procesu autoryzacji.

1.2. Problematyka uwierzytelniania użytkowników w systemach informatycznych

W poprzednim podrozdziale wyjaśniono zagadnienie uwierzytelniania. Teraz warto zastanowić się jak w praktyce można je realizować. Istnieje podział metod uwierzytelniania, który wyróżnia 3 podstawowe możliwości. Można uwierzytelnić się:

1. czymś, co wiemy — to najprostsza metoda i opiera się na zapamiętaniu przez użytkownika pewnej informacji, którą będzie musiał przedstawić podczas procesu uwierzytelniania.

2. czymś, co mamy — do tej grupy zaliczamy wszelkiego rodzaju fizyczne klucze i generatory tokenów.

3. czymś, czym jesteśmy — chodzi przede wszystkim o biometrię, czyli cechy naszego ciała. Niektórzy sugerują, że jest to to samo co punkt drugi — nie można w języku polskim powiedzieć, że „jestem kciukiem", zamiast tego mówi się, że „mam kciuk". To jednak akademicka dyskusja, którą pominiemy. Warta wyróżnienia jest natomiast informacja, że człowiek posiada pewne unikalne cechy, które pozwalają na jednoznaczną identyfikację i potwierdzenie jego tożsamości. Jest to między innymi odcisk palca, barwa i ton głosu, tęczówka oka i wiele innych.

Nie trudno dostrzec, że każda z powyższych metod ma wady z punktu widzenia bezpieczeństwa. Na początek „coś co wiemy" czyli hasła dostępowe i kody PIN. Idea uwierzytelniania się tą metodą była błędna niemal od samego początku. Opierała się bowiem na idealistycznym założeniu o zdolności człowieka do zapamiętywania silnych haseł. Szybko okazało się, że pamięć to najgorsze miejsce do przechowywania tego typu danych. Taka informacja musi być bowiem prosta, aby mogła zostać zapamiętana. Na początku nie stanowiło to wielkiego problemu, bo jedno silne hasło mogło zostać zapamiętane, ale wraz ze wzrostem mocy obliczeniowej komputerów

hasła proste do zapamiętania stały się równie proste do złamania. Aby informacja uwierzytelniająca mogła być bardziej skomplikowana, można ją zapisać. Trudno jednak znaleźć bezpieczne miejsce. Na pewno takim miejscem nie jest kartka przyklejona do monitora (co niestety zdarza się w przedsiębiorstwach). Trafnym pomysłem będzie za to użycie menadżerów haseł, które są obecnie polecane przez ekspertów. Jest to dużo lepsza metoda, jednak przejęcie przez atakującego danych z menadżera haseł zawsze będzie skutkowało kompromitacją danych dostępowych do wielu serwisów na raz. Nie trzeba też bujnej wyobraźni, aby zauważyć, że klucz fizyczny można zgubić, choć *YubiKey* przed atakami przeprowadzanymi zdalnie takimi jak phishing stanowi znakomitą ochronę. Biometria jest natomiast wyjątkowo kosztowna w implementacji. Systemy czytające cechy naszego ciała są często bardzo drogie. Ponadto nie powinna być używana samodzielnie, ponieważ uszkodzenie danej części ciała np. przecięcia palca skutkuje utratą dostępu do systemu. Jak zatem bezpiecznie się uwierzytelniać? Na dzień dzisiejszy nie znamy bezpieczniejszej metody uwierzytelniania niż połączenie przynajmniej dwóch z trzech powyższych możliwości. Taki mechanizm nazywamy **2FA** (ang. *Two Factor Authentication*). Dzięki 2FA znacząco zmniejszone zostaje prawdopodobieństwo przejęcia konta.

1.3. Historia protokołu Kerberos

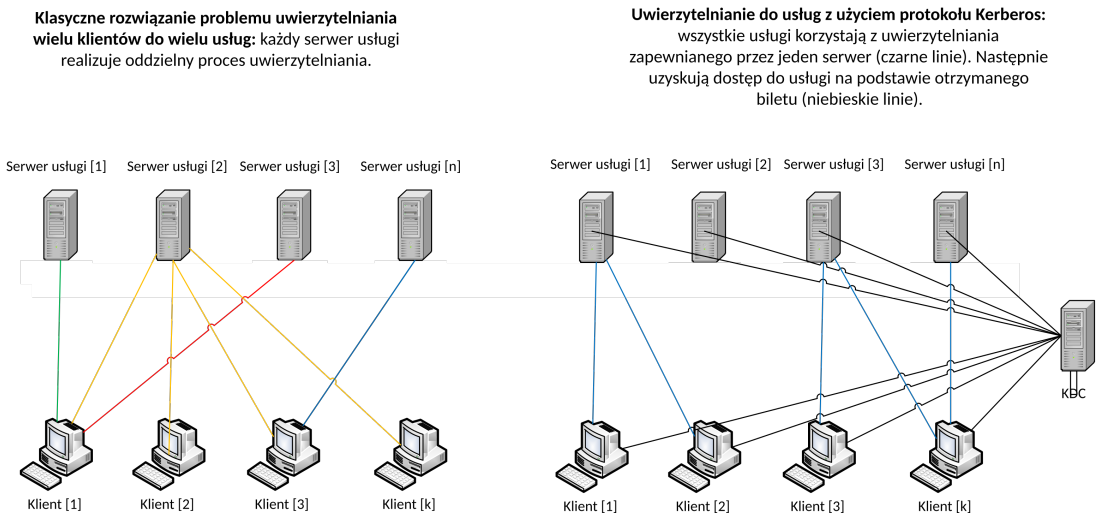
Zanim przedstawiona zostanie historia protokołu Kerberos, warto odwołać się do kilku istotnych wydarzeń w dziejach rozwoju komputerów. Lata 60. dla Informatyków brzmią jak prehistoria, ale właśnie wtedy powstała pierwsza sieć rozległa, ARPANET, która dała podwaliny dla dzisiejszego Internetu. ARPANET był projektem rozwijanym z inicjatywy Pentagonu i do takiego Internetu jaki ludzkość zna dziś jest mu daleko [5]. Jednak pierwsze sieci, niezależnie od tego czy są publicznie dostępne, determinują pojawienie się problemów związanych z ochroną danych i co za tym idzie, wymuszają rozwój bezpieczeństwa systemów uwierzytelniania. Na rozwiązania nie trzeba było długo czekać, ale to, które w niniejszej pracy jest najistotniejsze pojawia się w latach 80. minionego stulecia. To właśnie wtedy *Massachusetts Institute of Technology* w ramach projektu *Athena* projektuje nowy protokół uwierzytelniania, który ochrzczony został nazwą **Kerberos** na cześć trójgłowego psa znanego z mitologii greckiej. Początkowo był on wykorzystywany wyłącznie w Stanach Zjednoczonych i znalazł zastosowania w projektach militarnych. Z czasem został on jednak zaadoptowany do wielu innych rozwiązań komercyjnych i niekomercyjnych. Skąd tak wielka popularność protokołu Kerberos? Prawdopodobnie najważniejszą jego cechą jest to, że zaprojektowany został z godnie z logiką biznesową, która dawała odpo-

wiedni poziom bezpieczeństwa. Nie oznacza to, że zastosowanie protokołu Kerberos zawsze sprawi, że użytkownik nie musi przejmować się atakami - wręcz przeciwnie, co będzie tematem części praktycznej. Nie mniej jednak zasada działania protokołu przetrwała do dziś i jest wykorzystywana w wielu komercyjnych systemach teleinformatycznych oraz uznawana za bezpieczną. Zagrożenia wynikają z błędów w implementacji, konfiguracji, kryptografii lub czynnika ludzkiego, który jak powszechnie uznaje się w dziedzinie cyberbezpieczeństwa, jest zawsze najsłabszym ogniwem.

1.4. Techniczna analiza działania protokołu Kerberos

Protokół Kerberos zapewnia uwierzytelnienie sieciowe wielu klientów do wielu usług. Klasycznym rozwiązaniem takiego problemu jest indywidualnie uwierzytelnianie się każdego klienta w każdej usłudze z osobna. To rozwiązanie ma jednak poważne wady: w systemie informatycznym stworzonym na tej zasadzie niezbędne jest przeprowadzanie bardzo dużej ilości procesów uwierzytelniania. Trudne jest też wprowadzenie zmian w infrastrukturze, ponieważ kiedy dołączony zostanie kolejny serwer, wystąpi konieczność zapewnienia kolejnych mechanizmów uwierzytelniania dla każdego klienta z osobna. Te właśnie problemy rozwiązuje protokół Kerberos. Idea jego działania opiera się na dodaniu serwera, który będzie realizował uwierzytelnianie klientów w sposób scentralizowany. Serwer ten nazywany jest **KDC** (ang. *Key Distribution Center* - Centrum Dystrybucji Kluczy) i jego zadaniem jest wydawanie klientom kluczy, przy pomocy których będą mogli dostać się do usługi. KDC nie odpowiada za utrzymywanie kluczy otwartych sesji. Wydaje on jedynie bilet (ang. *Ticket*). Bilet ten zawiera informację o dostępach należących się danemu klientowi. Jest on zaszyfrowany i nie jest zrozumiały nawet dla klienta. Odczytać go może tylko inny KDC lub/i serwer docelowy. Jedynie poprawnie odkodowany będzie mógł zostać użyty. Jeśli odkodowanie się nie powiodło, żaden dostęp nie zostanie przyznany. Wyróżniamy dwa rodzaje biletów Kerberos:

- **Bilet przydzielania biletów** używany jest do weryfikacji tożsamości podczas żądania inicjalizującego. Bilet jest wydawany po pozytywnym przejściu procesu preuwierzytelniania (jeśli jest ono wymagane).
- **Bilet usługi** jest wydawany na podstawie biletu przydzielania biletów w rezultacie wystosowanego żądania o bilet dla konkretnej usługi.



Rysunek 1: Schemat przedstawia ideę działania protokołu Kerberos. W klasycznym rozwiązaniu problemu uwierzytelniania n klientów do k usług (gdzie $n, k \in \mathbb{N}$) procesy są realizowane niezależnie od siebie między każdym klientem, a serwerem usługi. Kiedy wdrożony zostanie protokół Kerberos, proces uwierzytelniania zostaje scentralizowany i jest realizowany przez KDC. Serwer usługi realizuje jedynie walidację biletu usługi i autoryzację dostępu. Źródło: opracowanie własne.

1.4.1. Proces uzyskania biletu Kerberos

Opis powstał na podstawie analizy implementacji protokołu Kerberos autorstwa firmy Microsoft.

Pierwszym krokiem do uzyskania biletu Kerberos jest wysłanie żądania ***Authentication Service Request*** do KDC. Na tym etapie użytkownik nie podaje danych uwierzytelniających. Zgodnie z domyślną konfiguracją usługi *Active Directory*, takie zapytanie nie zostanie zaakceptowane. Serwer odpowie błędem i będzie wymagał od użytkownika **preuwierzytelniania**.

Definicja 1.3. (Preuwierzytelnianie) *to wstępna weryfikacja tożsamości przez serwer KDC. Mechanizm ten chroni przed atakami siłowymi typu password guessing [12].*

Weryfikacja tożsamości w ramach preuwierzytelniania polega na tym, że klient wystosuje drugie żądanie Authentication Service Request. W tym żądaniu zawarte zostają informacje, które umożliwiają weryfikacje tożsamości. Są to:

- adres IP,
- lokalny czas.

Żądanie to jest szyfrowane wynikiem funkcji skrótu obliczonym na podstawie hasła użytkownika wnioskującego o uwierzytelnienie. Serwer KDC podejmuje próbę odszyfrowania otrzymanego żądania. W tym celu z perspektywy serwera niezbędna jest znajomość informacji szyfrującej, a więc skrótu hasła uwierzytelnianego użytkownika. KDC ma dostęp do tej informacji. Stosuje więc jako klucz deszyfrujący rzeczony wynik funkcji skrótu. Warto tutaj uzupełnić, że stosowana jest kryptografia symetryczna, co oznacza, że ten sam klucz pozwala na szyfrowanie i deszyfrowanie informacji. Jeśli więc udało się odszyfrować żądanie, to oznacza, że hasło jest prawidłowe. Nie jest to jednak koniec procesu. Należy bowiem zadać istotne pytanie o bezpieczeństwo. Co jeśli atakujący przechwycił żądanie i użyje go w imieniu prawowitego użytkownika? Jest to atak o nazwie *Token Replay*. Rozwiązaniem tego problemu jest weryfikacja adresu IP i czasu lokalnego, czyli informacji zawartych w żądaniu. Serwer porównuje je z tym adresem IP, z którego wysłany został pakiet danych. Jeśli adresy IP nie są takie same, serwer nie zwróci biletu.

Jeżeli wszystkie wyżej przedstawione warunki zostały spełnione (udało się odszyfrować żądanie wynikiem funkcji skrótu hasła, zgodny był adres IP oraz czas wysłania żądania) to serwer udziela użytkownikowi bilet Kerberos przesyłając odpowiedź ***Authentication Service Response***. Otrzymanie kończy etap preuwierzytelniania. Na podstawie powyższej analizy można wysnuć wniosek, że ogromne znaczenie dla bezpieczeństwa ma wymóg stosowania preuwierzytelniania. W części praktycznej niniejszej pracy przedstawione zostaną wybrane zagrożenia wynikające z błędnej konfiguracji kont użytkownika w postaci braku wymogu preuwierzytelniania.

1.4.2. Bilet przydzielania biletów

Bilet przekazany w odpowiedzi *Authentication Service Response* zawiera informacje o tym, do jakich usług powinien mieć dostęp użytkownik. Kolejny raz należy więc zadać pytanie o bezpieczeństwo - czy użytkownik jest w stanie zmodyfikować otrzymany bilet i wynikające z niego uprawnienia dostępu? Odpowiedź jest negatywna: użytkownik nie może tego zrobić, ponieważ otrzymany bilet jest zaszyfrowany. Nie jest on jednak zaszyfrowany wynikiem funkcji skrótu swojego własnego hasła tak jak żądanie, które wysłał do serwera. Gdyby tak było, to otrzymany bilet mógłby dowolnie modyfikować. Założenie polega więc na tym, aby zaszyfrować wydany bilet kluczem do którego użytkownik nie będzie miał dostępu, a który odczytać będzie mógł każdy kontroler domeny. W przypadku *Active Directory* w implementacji Microsoft takim kluczem jest funkcja skrótu wyliczona na podstawie hasła użytkownika **„krbtgt”**. Konto to jest obecne na każdym kontrolerze domeny, dzięki czemu każdy kontroler domeny może odczytać bilet Kerberos.

1.4.3. Bilet usługi

Użytkownik mając bilet Kerberos może wnioskować o udzielenie dostępu do usługi. Musi w tym celu wysłać do dowolnego kontrolera domeny zapytanie ***Ticket Granting Service Request***. W zapytaniu określona jest nazwa usługi (***Service Principal Name*** **(SPN)**), a do treści załączony zostaje bilet przydzielania biletów. Żądanie zaszyfrowane zostanie wynikiem funkcji skrótu hasła użytkownika, który o nie wnioskuje. W przypadku *Active Directory* po otrzymaniu żądania kontroler domeny dokonuje skanowania bazy danych celem ustalenia konta powiązanego z usługą wskazaną przez SPN. Odpowiedź, którą udzieli serwer będzie zaszyfrowana wynikiem funkcji skrótu powiązanej z SPN. Zawiera ona członkostwa grup. Nie są one jednak pobrane z bazy danych serwera — są zdefiniowane na bazie członkostw grup, które wynikają z biletu przydzielania biletów Kerberos.

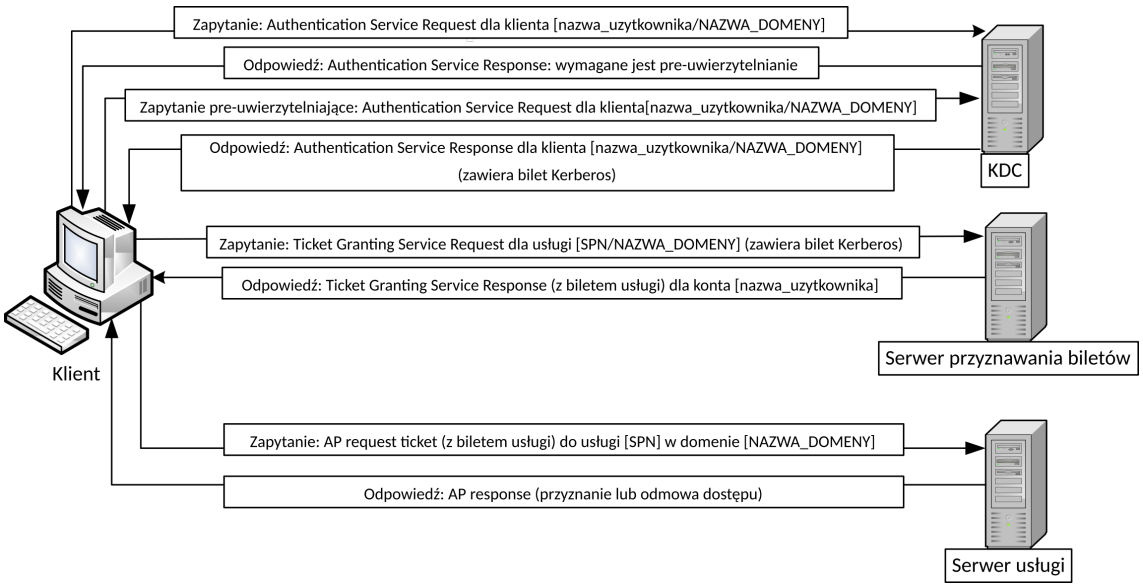
1.4.4. Uzyskanie dostępu do usługi

Na tym etapie klient posiada bilet usługi Kerberos. Ponownie nie jest w stanie go jednak odczytać, ponieważ jest zaszyfrowany wynikiem funkcji skrótu hasła konta powiązanego z usługą. Bilet odczytać może jednak serwer usługi. Klient przesyła więc bilet dla usługi o zdefiniowanej przez SPN nazwie załączając go w żądaniu ***Application Request Ticket***. Jeżeli serwer aplikacji poprawnie odszyfrował bilet usługi za pomocą skrótu hasła konta powiązanego z tą usługą, to oznacza, że bilet jest ważny, wszystkie poprzednie kroki były wykonane zgodnie z założeniem protokołu i serwer aplikacji może przystąpić do procesu autoryzacji. Kolejny raz należy bowiem zwrócić uwagę na różnice między procesem autoryzacji, a uwierzytelnieniem. Można powiedzieć, że na razie użytkownik przeszedł jedynie proces uwierzytelnienia — udowodnił serwerowi aplikacyjnemu swoją tożsamość, ale to nie znaczy, że należy mu przyznać dostęp. Serwer aplikacji przyznaje dostęp na podstawie informacji o członkostwie grup zawartym w załączonym bilecie usługi Kerberos [11].

1.5. *Active Directory* - czym jest i co ma wspólnego z protokołem Kerberos?

Active Directory to usługa katalogowa będąca implementację protokołu ***Lightweight Directory Access Protocol*** (LDAP).

Definicja 1.4. (Lightweight Directory Access Protocol) *to protokół pozwalający na wymianę i aktualizację informacji poprzez połączenie bazą danych w oparciu o protokół Transmission Control Protocol/Internet Protocol (TCP/IP) [21].*



Rysunek 2: Schemat przedstawia proces uwierzytelnienia użytkownika do usługi w ramach protokołu Kerberos. Źródło: opracowanie własne.

W przypadku usługi *Active Directory* baza danych przechowuje dane o organizacji i wszelkich obiektach, które do niej należą. Obiekty w *Active Directory* reprezentują:

- jednostki organizacyjne, czyli kontenery, które pozwalają na przechowywanie innych obiektów,
- użytkowników,
- komputery,
- drukarki,
- grupy,
- udziały sieciowe,
- kontakty.

Active Directory pozwala więc na scentralizowane zarządzanie infrastrukturą IT. Dzięki zastosowaniu protokołu Kerberos kontrolery domeny realizują również proces uwierzytelniania użytkowników do usług działających w ramach domeny. Komunikacja w usłudze *Active Directory* opiera się o hierarchiczny system rozwiązywania nazw. Każdy obiekt posiada pełną nazwę (ang. *DistinguishedName*), która opisuje jego położenie w strukturze hierarchicznej. Poniższa tabela przedstawia najważniejsze atrybuty:

Atrybuty obiektów w usłudze *Active Directory*:

Atrybut	**Komponent (ang)**	**Komponent (pl)**
CN	commonName	powszechna nazwa
OU	organizationalUnitName	nazwa jednostki organizacyjnej
O	organizationName	nazwa organizacji
STREET	streetAddress	ulica
L	localityName	nazwa miejscowości
ST	stateOrProvinceName	stan lub prowincja
C	countryName	kraj
UID	userid	ID użytkownika

Źródło: [22]

Przykład poprawnej, pełnej nazwy DN:

```
1 CN=Jan Kowalski,OU=IT,DC=Company,DC=SEC
```

1.6. Podsumowanie

Usługa *Active Directory* pozwala zespołowi administratorów systemu teleinformatycznego na kompleksowe zarządzanie zasobami firmy. Zastosowanie protokołu **Kerberos** pozwala na scentralizowane uwierzytelnianie natomiast protokół **LDAP** pozwala na integrację z bazą danych. Z tej przyczyny bezpieczeństwo mechanizmów reglamentacji dostępu w środowisku opartym o *Active Directory* jest kluczowe, a nieautoryzowany dostęp może skutkować przejęciem przez atakującego wszystkich zasobów organizacji.

2. Metodyka i zastosowane technologie

Rozdział stanowi wprowadzenie do części praktycznej. Omówiona została w nim zastosowana metodyka podejmowanych działań. Stanowi on uzupełnienie dokumentacji projektu o przedstawienie aspektów związanych z organizacją pracy. Przedstawione zostały w nim też wykorzystane narzędzia i technologie.

2.1. Metodyka

Zastosowana metodyka pracy opiera się o zasady **iteracyjnych metod wytwarzania**. Projekt podzielony był na nieokreśloną z początku liczbę iteracji — zamkniętych cykli, których rezultatem jest fragment finalnego efektu pracy. Każdy cykl kończy się zdefiniowaniem problemów i określeniem nowych potrzeb. Z tego powodu akceptowane było wprowadzanie zmian we wcześniej przygotowanych zasobach jeśli wymagała tego sytuacja.

2.2. Metodyka wdrażania infrastruktury

Zmiany wprowadzane wstecznie często dotyczyły infrastruktury. Celem pracy nie było bowiem pokazanie dobrze działającego środowiska opartego o usługę *Active Directory*. Wręcz przeciwnie — celem było pokazanie błędów konfiguracyjnych, które generują pewne zagrożenia. Wprowadzane iteracyjne zmiany we wdrożonej infrastrukturze polegały więc często na dodaniu funkcjonalności i błędów konfiguracyjnych, dzięki którym możliwe było zaprezentowanie kolejnych ataków. Każdy celowo spreparowany błąd w konfiguracji był opatrzony scenariuszem, czyli krótką historią zawierającą uzasadnienie, dlaczego administrator zarządzający usługą *Active Directory* mógłby zdecydować się na wdrożenie danej konfiguracji. Innym źródłem uzasadnienia dla wprowadzanych celowo błędów konfiguracyjnych były raporty organizacji zajmujących się cyberbezpieczeństwem, z których wynikało, że dany problem jest powszechny w infrastrukturach rzeczywistych organizacji.

2.3. Metodyka testów penetracyjnych

Testy przeprowadzone zostają z założeniem, że atakujący posiada dostęp do sieci lokalnej. Kontrolowane przez niego urządzenie posiada adres IP przydzielony przez

serwer DHCP oraz maskę sieciową. Otwiera to prostą drogę do zlokalizowania komputerów klienckich oraz kontrolerów domeny. Najprostsze sposoby to wykorzystanie wiedzy, że komputery w sieci *Active Directory* często wymieniają się udziałami sieciowymi przez co muszą mieć otwarty port *tcp/445 SMB*. Narzędzie *nmap* pozwala na zlokalizowanie urządzeń z otwartym portem usługi Samba.

```
$ nmap -Pn -p445 10.0.0.0/24 --open
Starting Nmap 7.93 ( https://nmap.org ) at 2022-11-27 05:01 EST
Nmap scan report for 10.0.0.11
Host is up (0.022s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Nmap scan report for 10.0.0.12
Host is up (0.022s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Nmap scan report for 10.0.0.100
Host is up (0.0011s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Nmap scan report for 10.0.0.150
Host is up (0.0016s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
```

Rysunek 3: Zrzut ekranowy przedstawia proces skanowania sieci w poszukiwaniu urządzeń z otwartym portem *TCP 445*. Użyto narzędzia *nmap*. Źródło: opracowanie własne.

Lepszą metodą jest jednak skorzystanie z narzędzia *crackmapexec*, które daje atakującemu znacznie więcej informacji, takich jak domena sieciowa, nazwy znalezionych hostów oraz czy *SMB signing* jest wyłączone.

```
$ crackmapexec smb 10.0.0.0/24
SMB         10.0.0.12       445    DCRO             [*] Windows Server 2016 Datacenter Evaluation 14393 x64 (name:DCRO) (domain:company.sec) (signing:True) (SMBv1:True)
SMB         10.0.0.11       445    PDC              [*] Windows Server 2016 Datacenter Evaluation 14393 x64 (name:PDC) (domain:company.sec) (signing:True) (SMBv1:True)
SMB         10.0.0.100      445    PC1              [*] Windows 10.0 Build 19041 x64 (name:PC1) (domain:company.sec) (signing:False) (SMBv1:False)
SMB         10.0.0.150      445    PC2              [*] Windows 10.0 Build 19041 x64 (name:PC2) (domain:company.sec) (signing:False) (SMBv1:False)
```

Rysunek 4: Zrzut ekranowy przedstawia wykorzystanie narzędzia *crackmapexec* do skanowania urządzeń w sieci. Źródło: opracowanie własne.

2.4. Metodyka prowadzenia dokumentacji

Każdy fragment tekstu zamieszczony w niniejszej pracy przeszedł następujące etapy:

1. **Autor fragmentu tekstu** nazywany dalej „autorem" opisuje teorię związaną z danym zagadnieniem lub wykonane w ramach części praktycznej czynności.
2. Napisany tekst zostaje poddany recenzji drugiego autora pracy, zwanego dalej **„współautorem"**. Współautor przekazuje autorowi swoje uwagi merytoryczne, estetyczne i językowe.
   (a) Autor wprowadza poprawki do tekstu na podstawie uwag współautora.
   (b) Cykl jest powtarzany aż do osiągnięcia obustronnej akceptacji wykonanej pracy.
3. Całość pracy zostaje przekazana do recenzji opiekuna projektu.
   (a) Opiekun projektu przekazuje uwagi merytoryczne, estetyczne i językowe autorom pracy.
   (b) Autorzy pracy wspólnie wprowadzają zasugerowane przez opiekuna projektu zmiany.
   (c) Cykl jest powtarzany aż do osiągnięcia odpowiednich rezultatów.
4. Zakończenie prac nad dokumentacją projektu inżynierskiego. Przekazanie pracy do dalszych etapów.

2.5. Ramy czasowe

2.5.1. Wrzesień 2022

Pierwszy miesiąc poświęcony został na planowaniu. Określono temat pracy i zdefiniowano cele. Zdefiniowano wstępne wymagania techniczne infrastruktury niezbędnej do realizacji tematu. Podjęto decyzję o narzędziach i platformie wykorzystywanej do stworzenia pracy.

2.5.2. Październik 2022

Rozpoczęto pracę nad częścią teoretyczną. Dokonano analizy działania protokołu Kerberos i usługi *Active Directory*. Zdefiniowano zagrożenia mechanizmów uwierzytelniania w domenie *Active Directory*. Sporządzono więc listę ataków, które warto przedstawić. Zdefiniowano wymagania konfiguracyjne dla wdrożonego środowiska laboratoryjnego. Drugą połowę miesiąca zajęło wdrożenie infrastruktury i dokonanie odpowiedniej konfiguracji. Decyzje o sposobie konfiguracji podejmowano na podstawie analizy materiałów źródłowych oraz metodą prób i błędów.

2.5.3. Listopad 2022

W listopadzie rozpoczęto testy penetracyjne wdrożonej infrastruktury zgodnie z wcześniej przygotowanymi założeniami. Przygotowano notatki, ana ich podstawie udokumentowano proces testów penetracyjnych w niniejszej pracy. W międzyczasie rozwijano sekcję teoretyczną o analizę działania usługi *Active Directory*.

2.5.4. Grudzień 2022

Dokonano poprawek na podstawie wytycznych dostarczonych przez opiekuna projektu. Przygotowano załączniki i materiały niezbędne do prezentacji projektu inżynierskiego. Przekazano projekt wraz z dokumentacją do zatwierdzenia.

2.6. Narzędzia

2.6.1. Oracle VM VirtualBox

Wybrany program do wirtualizacji serwerów i komputerów klienckich to Oracle VM VirtualBox. Jest to narzędzie w pełni darmowe do użytku niekomercyjnego. Zapewnia wszystkie najistotniejsze funkcjonalności. W przeciwieństwie do konkurencyjnego, darmowego rozwiązania jakim jest program VMware Workstation Player, VirtualBox pozwala na wykonywanie migawek z bieżącego stanu maszyny. Było szczególnie istotne dla naszego projektu, ponieważ istniało ryzyko uszkodzenia systemu podczas przeprowadzanych testów.

2.6.2. Microsoft Windows Server 2016

Inżynierowie firmy Microsoft są autorami rozwiązania *Active Directory* i przez długi czas tylko i wyłącznie systemy z rodziny Windows pozwalały na wdrożenie usługi. Obecnie istnieje możliwość wdrożenia *Active Directory* w systemie Linux na bazie usługi Samba4, jednakże przy istotnie ograniczonej funkcjonalności (na poziomie funkcjonalności charakterystycznych dla usługi *Active Directory* znanej z

systemu Windows Server 2008 R2). Między innymi z tego powodu jest to rozwiązanie stosunkowo rzadko spotykane w organizacjach. Dlatego też wdrożoną infrastrukturę oparto o systemy Microsoft Windows. Wybrano wersję Windows Server 2016. Nie jest to najnowsza wersja systemu Windows Server, ale cały czas posiada wsparcie techniczne i aktualizacje bezpieczeństwa. Jest natomiast znacznie częściej spotykana w organizacjach niż najnowsza wersja Windows Server 2022, ponieważ aktualizacja systemu serwerowego do najnowszej wersji jest procesem złożonym i często nie znajduje z perspektywy firm uzasadnienia finansowego. Użycie starszej wersji jaką jest Windows Server 2016 samo w sobie nie może być również rozpatrywane jako podatność, ze względu na ciągłość wsparcia technicznego i aktualizacji bezpieczeństwa dostarczanych przez Microsoft. Do celów edukacyjnych wersję *Trial* na okres 180 dni można pobrać ze strony: `https://info.microsoft.com/ww-landing-windows-server-2016.html`. Studenci mają także możliwość pobrania pełnej wersji systemu ze strony `https://azureforeducation.microsoft.com/devtools` w ramach programu Microsoft Azure Dev Tools for Teaching.

2.6.3. Microsoft Windows 10 Enterprise/Education

System Windows 10 w wersji Enterprise jest naturalnym uzupełnieniem systemu Windows Server 2016. Systemy z rodziny Windows są według raportu dostarczonego przez StackOverflow najczęściej wykorzystywane przez deweloperów w pracy profesjonalnej — stanowią aż 45,3% wszystkich urządzeń [16]. Jak natomiast wynika z raportu dostarczonego przez firmę AdDuplex system Windows 10 jest najczęściej używaną wersją z pośród klienckich dystrybucji systemu Windows (na podstawie danych z czerwca 2022 roku) [17]. Podobnie jak w przypadku dystrybucji serwerowej, wersję *Trial* można pobrać za darmo na okres 90 dni ze strony: `https://www.microsoft.com/en-us/evalcenter/download-windows-10-enterprise`. Studenci mają także możliwość pobrania pełnej wersji systemu w wersji Education wraz z kluczem produktu ze strony `https://azureforeducation.microsoft.com/devtools` w ramach programu Microsoft Azure Dev Tools for Teaching. Wersja Education bazuje na wersji Enterprise i nie istnieją między nimi różnice, które wpływałyby na rezultaty otrzymane w niniejszej pracy.

2.6.4. Kali Linux

Kali Linux system operacyjny oparty na dystrybucji Debian używany powszechnie do prowadzenia testów penetracyjnych. Zawiera wbudowanych wiele narzędzi przydatnych do przeprowadzania ataków, dzięki czemu nie trzeba instalować ich ręcznie. Tester bezpieczeństwa korzysta zazwyczaj z bardzo dużej ilości programów,

skryptów, list i innych narzędzi więc użycie systemu Kali Linux pozwala zaoszczędzić bardzo dużo czasu na konfiguracji systemu i oprogramowania. System Kali Linux można pobrać za darmo ze strony `https://www.kali.org/get-kali/`.

2.6.5. Impacket

Impacket jest zbiorem narzędzi pozwalających na niskopoziomowy dostęp do wielu protokołów używanych w środowisku *Active Directory*. Wszystkie narzędzia są napisane w języku Python oraz są otwarto-źródłowe. Opis niektórych z narzędzi można znaleźć na stronie `https://www.secureauth.com/labs/open-source-tools/impacket/`. Skrypty te są powszechnie wykorzystywane przez pentesterów, członków red teamów czy grup przestępczych podczas atakowania infrastruktury *Active Directory* [18].

2.6.6. Wireshark

Wireshark jest snifferem, czyli programem komputerowym, służącym do przechwytywania ruchu sieciowego. Wireshark jest wolnym i otwartoźródłowym oprogramowaniem. Można go pobrać za darmo ze strony: `https://www.wireshark.org`.

2.6.7. Microsoft Visio Professional 2021

Do projektowania schematów graficznych i wykresów użyto programu Microsoft Visio Proffesional 2021. Zapewnia on bogatą bibliotekę piktogramów i umożliwia eksport utworzonych wykresów jako grafik wektorowych. Studenci mają możliwość pobrania pełnej wersji programu ze strony `https://azureforeducation.microsoft.com/devtools` w ramach programu Microsoft Azure Dev Tools for Teaching co było główną przyczyną wybrania narzędzia.

2.6.8. Edytor tekstu LaTeX

Niniejsza praca została stworzona z użyciem oprogramowania LaTeX. Daje on niemal nieograniczone możliwości edycji i formatowania dokumentów tekstowych. Pozwala na zmaksymalizowanie poziomu estetyki i przejrzystości przygotowanego dokumentu. Istnieje wiele edytorów kodu źródłowego LaTeX. Wybrany został internetowy edytor `overleaf.com` ponieważ pozwala na współdzielenie dokumentu i synchronizację w czasie rzeczywistym. Dzięki temu każdy z autorów mógł pracować w tym samym czasie i obserwować zmiany wprowadzone przez drugą osobę na bieżąco.

2.7. Podsumowanie

Metodyka organizacji pracy, wdrażania infrastruktury oraz testowania bezpieczeństwa stanowi podstawę dla wszystkich działań podejmowanych w kolejnych krokach. Wybrane zostały też narzędzia, które pozwolą na realizację założeń projektu.

3. Wdrożenie infrastruktury opartej na usłudze *Active Directory*

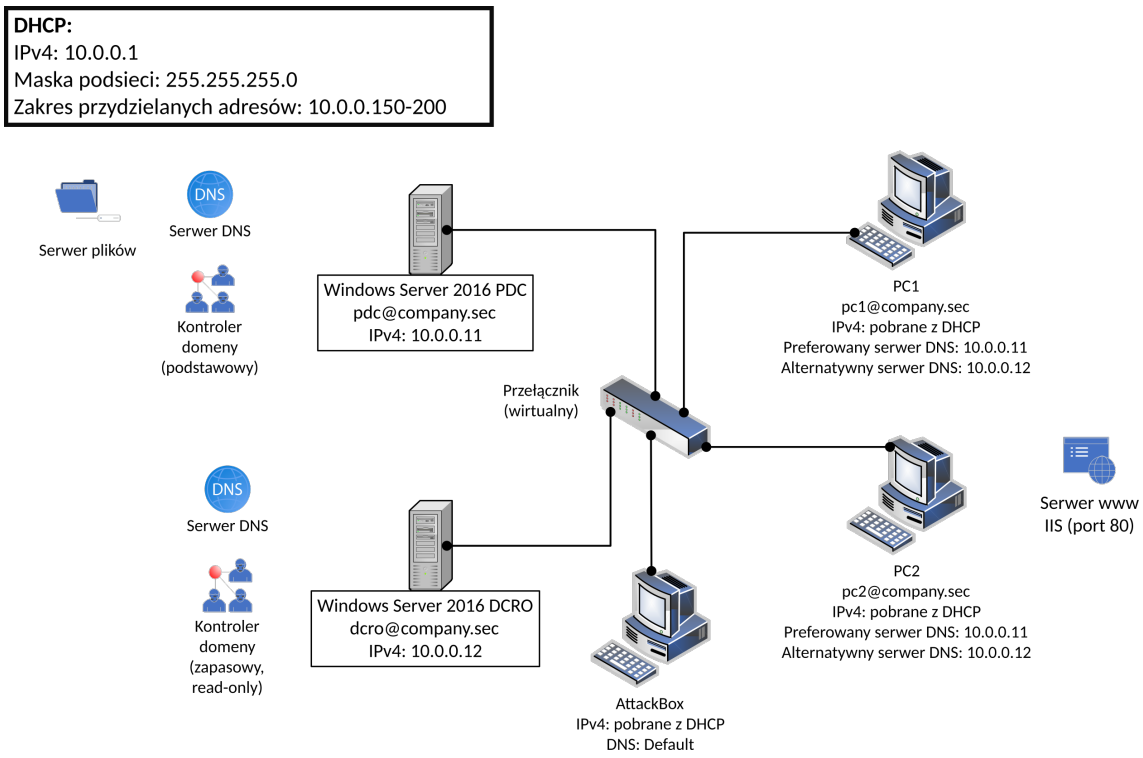
3.1. Projekt infrastruktury

Do wdrożenia infrastruktury wykorzystaliśmy hiperwizor typu II, oprogramowanie Oracle VM VirtualBox. Utworzyliśmy 5 wirtualnych maszyn. Połączyliśmy je przy pomocy karty sieci izolowanej *„VirtualBox Host-Only Ethernet Adapter"*. Wybraliśmy ten typ sieci, ponieważ umożliwia on uruchomienie serwera DHCP, dzięki czemu jesteśmy w stanie lepiej zasymulować sieć organizacji oraz pozwala na komunikację z systemem gospodarza. Karta typu *Host-Only* nie pozwala natomiast na dostęp do sieci Internet, która nie jest na większości maszyn potrzebna. Wirtualnym maszynom, które potrzebują mieć dostęp do sieci skonfigurowana została druga kartę sieciowa uruchomiona w trybie NAT.

Utworzone maszyny to kolejno:

- Windows Server 2016 „PDC" (*Primary Domain Controler*) - podstawowy kontroler domeny. Pełni również funkcję:
  - Preferowanego serwera DNS dla urządzeń w domenie company.sec.
  - Serwera plików dla działu IT - udostępniony został katalog sieciowy.
- Windows Server 2016 „DCRO" (*Domain Controler Read-Only*) - zapasowy kontroler domeny skonfigurowany w trybie tylko do odczytu. Pełni też funkcję zapasowego serwera DNS,
- Windows 10 „PC1" - komputer pracownika firmy,
- Windows 10 „PC2" - komputer pracownika firmy - programisty stron internetowych. Ze względu na jego potrzeby zawodowe na tym komputerze konieczne było uruchomienie serwera stron Internetowych IIS.
- Kali Linux „AttackBox" - opis został przedstawiony w sekcji „Narzędzia".

Poniżej przedstawiony został schemat wdrożonej infrastruktury:



Rysunek 5: Schemat wdrożonej infrastruktury. Źródło: opracowanie własne.

3.2. Konfiguracja danych adresowych

Przydzielone zostały następujące dane adresowe:

- Windows Server 2016 „PDC” - dane statyczne:
  - IPv4: 10.0.0.11,
  - Maska podsieci: 255.255.255.0,
  - Preferowany serwer DNS: 127.0.0.1,
  - Alternatywny serwer DNS: 10.0.0.12.
- Windows Server 2016 „RODC” - dane statyczne:
  - IPv4: 10.0.0.12
  - Maska podsieci: 255.255.255.0,
  - Preferowany serwer DNS: 10.0.0.11,
  - Alternatywny serwer DNS: 127.0.0.1.
- Windows 10 „PC1” - automatyczna konfiguracja (pobranie danych adresowych z serwera DHCP). Statycznie określono:

  - Preferowany serwer DNS: 10.0.0.11,
  - Alternatywny serwer DNS: 10.0.0.12.

- Windows 10 „PC2" - automatyczna konfiguracja (pobranie danych adresowych z serwera DHCP). Statycznie określono:
  - Preferowany serwer DNS: 10.0.0.11,
  - Alternatywny serwer DNS: 10.0.0.12.

- Kali Linux „AttackBox" - automatyczna konfiguracja (pobranie danych adresowych z serwera DHCP)

Serwer DHCP skonfigurowano w oprogramowaniu wirtualizacyjnym w następujący sposób:

- Adres serwera: 10.0.0.1,
- Maska podsieci: 255.255.255.0,
- Dolna granica adresów: 10.0.0.150,
- Górna granica adresów: 10.0.0.200.

Dokument hipertekstowy zawierający wszystkie zrzuty ekranowe wykonane w trakcie wdrażania infrastruktury stanowi załącznik do niniejszej pracy.

3.3. Wdrożenie usługi domenowej *Active Directory*

Na wirtualnej maszynie „PDC" z systemem Windows Server 2016 zainstalowane zostały role *„Active Directory Domain Services"* (domenowe usługi *Active Directory*). Kontroler domeny został dodany do nowego lasu. Wybrano nazwę domeny „company.sec". Zainstalowano też rolę *„File and Storage Services"* dzięki czemu system mógł pełnić funkcję serwera plików.

Na serwerze „DCRO" również zainstalowano rolę „*Active Directory Domain Services*", jednak tym razem zamiast utworzyć nowy las, system został dodany do istniejącej już domeny *„company.sec"*, która mogła zostać rozwiązana, gdyż preferowanym serwerem DNS była wirtualna maszyna „PDC", która jest kontrolerem domeny *„company.sec"*. Wybrano opcję *„Read-only domain controler"* przez co kontroler domeny „DCRO" został ustawiony w tryb tylko do odczytu, co oznacza, że nie będzie można z jego poziomu dokonywać zmian w domenie. Będzie on natomiast przechowywał całą bazę danych *Active Directory* i w razie awarii „PDC" przejmie

rolę serwera DNS, gdyż jest ustawiony jako alternatywny serwer DNS na każdej z wirtualnych maszyn w domenie.

3.4. Konfiguracja wdrożonego systemu informatycznego

Konfigurację środowiska rozpoczynamy od utworzenia dużej liczby użytkowników, tak aby zasymulować warunki panujące w firmie. Aby to zrealizować użyto skryptu napisanego w języku *PowerShell*. Został on załączony do pracy. Skrypt ten przyjmuje jako argument wejściowy plik *.csv*, pobiera z niego dane o użytkownikach i na ich podstawie tworzy konta domenowe. Wykorzystany plik *.csv* stanowi załącznik do niniejszej pracy.

Pracowników przydzielamy do grup, które stosowane są w systemie *Active Directory* do reglamentacji uprawnień. Odpowiednio utworzone zostają więc grupy o nazwach odpowiadających działowi firmy:

- *IT*,
- *Graphics*,
- *Administration*,
- *Security*.

Pracownicy działu IT firmy *„Company"* potrzebują także współdzielonego folderu do przechowywania danych. Utworzony zostaje więc folder *IT* i udostępniony jako zasób sieciowy. Zezwolono na dostęp do niego tylko pracownikom działu *IT* (grupa bezpieczeństwa *IT*). Dla testu utworzony został w udostępnionym folderze plik tekstowy *„Sensitive data.txt"* i wpisano do niego losowy ciąg znaków w celu weryfikowania, czy odpowiedni użytkownicy rzeczywiście mają do niego dostęp.

Komputery Windows 10 „PC1" i Windows 10 „PC2" zostały dodane do domeny „company.sec". Ponadto na komputerze Windows 10 „PC2" zainstalowany został serwer stron internetowych IIS, ponieważ urządzenie „PC2" należy do programisty stron Internetowych w firmie *„Company"* i lokalny serwer www jest mu niezbędny w codziennej pracy.

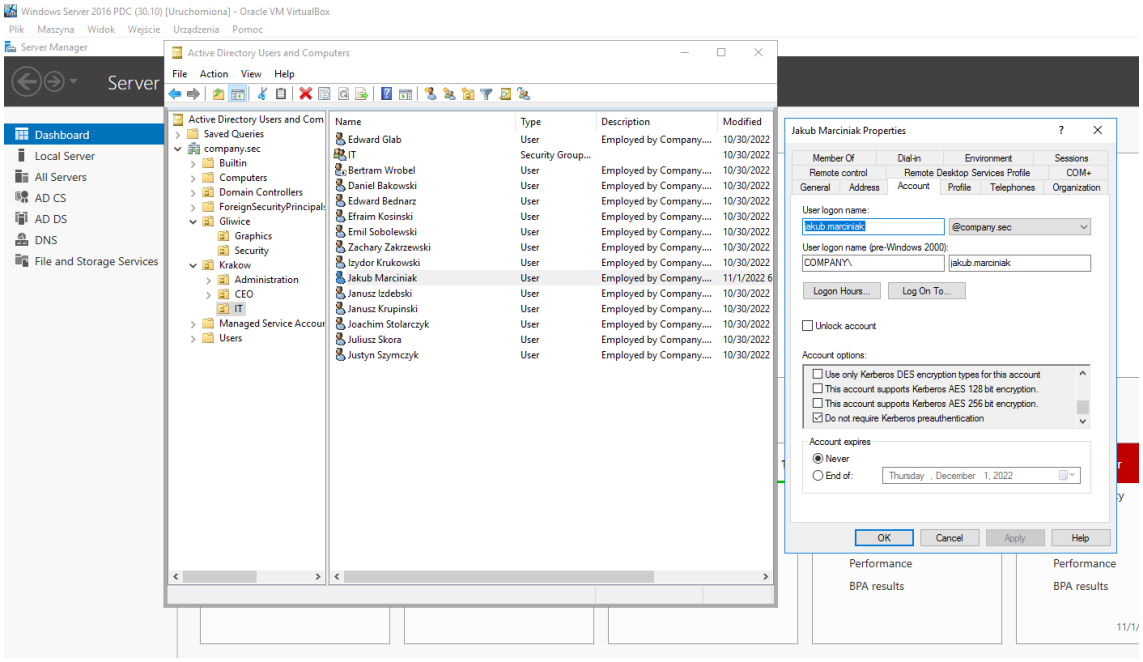
Na podstawie raportów **Microsoft Digital Defense Report** [26] [27] wytypowano również często występujące w organizacjach modyfikacje domyślnej konfiguracji powodujące podatności. Wdrożono więc następujące poprawki do domyślnej konfiguracji *Active Directory*, które pozwolą na przeprowadzenie ataków przedstawionych w kolejnych rozdziałach.

3.4.1. Scenariusz konfiguracji: włączenie konta lokalnego Administratora na komputerach.

Przyczyną, dla której pracownicy działów IT często włączają konto standardowego Administratora lokalnego jest konieczność utrzymania na stacji roboczej przynajmniej jednego konta użytkownika o uprawnieniach administracyjnych. Nie biorą jednak pod uwagę, że zamiast tego mogą oni pozostawić konto użytkownika z uprawnieniami Administracyjnymi uzyskiwanymi na bazie mechanizmu UAC (*User Access Control*). Aby uprościć zarządzanie infrastrukturą w dużych organizacjach zdarza się też, że Administrator ustawia to samo hasło do konta lokalnego Administratora na każdym komputerze w organizacji. Dodatkowy problem stanowi też współdzielenie tego hasła przez pracowników działu IT. Jednym z celów pracy będzie sprawdzenie, czy taka Metodyka zarządzania infrastrukturą w organizacji może nieść ze sobą niebezpieczeństwo.

3.4.2. Scenariusz konfiguracji: Wyłączenie na wybranym koncie użytkownika wymogu preautentykacji

Firma Company rozwija się ponadprzeciętnie szybko i liczba kont użytkowników w *Active Directory* dla pracowników przekroczyła kilkaset tysięcy. W celu sprawnego zarządzania dostępami przy zachowaniu zasady „najmniejszego uprzywilejowania" Administratorzy utworzyli też bardzo dużą ilość grup bezpieczeństwa. W rezultacie każdy użytkownik należy do kilkuset grup. Okazało się, że bilet przyznawania biletów protokołu Kerberos jest tak duży, że usługa odpowiedzialna za uwierzytelnianie jest silnie przeciążona i nie w stanie realizować swoich zadań ze względu na ograniczone zasoby mocy obliczeniowej. Administrator zauważył, że problem znika, jeśli odznaczy opcję ***Do not require Kerberos preauthentication***. W rezultacie tego działania proces uwierzytelniania oparty na protokole Kerberos znów odbywa się sprawnie, a procesor serwera na którym zainstalowany jest kontroler domeny nie jest przeciążony. W analizowanym środowisku wprowadzona zostaje więc następująca konfiguracja: na wybranych kontach użytkowników opcja ***Do not require Kerberos preauthentication*** zostaje aktywowana, tak jak na zamieszczonym wcześniej zrzucie ekranowym. Należy jednak nadmienić, że dysponując symulowanym środowiskiem opartym o wirtualizację typu II wdrożoną na komputerze osobistym nie istnieje możliwość odpowiedniego zasymulowania warunków, które przedstawione zostały w scenariuszu i które doprowadzą do wywołania omawianego problemu wydajnościowego. Do wdrażanej konfiguracji nie zostaje więc dodana ani tak duża ilość użytkowników, ani tak duża ilość grup bezpieczeństwa. Aktywo-



Rysunek 6: Aktywowanie opcji ***Do not require Kerberos preauthentication*** dla jednego z użytkowników domenowych

wano jedynie wspomnianą opcję na koncie użytkowników domenowych *pawel.zapior* oraz *jakub.marciniak*, co często jest wykorzystywane jako rozwiązanie opisanego problemu. Na etapie testów penetracyjnych wdrożonej infrastruktury przeanalizujemy konsekwencje związane z aktywowaniem tej opcji.

3.4.3. Scenariusz konfiguracji: przypisanie usługi IIS do konta Administrator

Zgodnie z przedstawioną w części teoretycznej zasadą działania usług w *Active Directory*, utworzony został *Service Principal Name* dla usługi IIS z kontem *Administrator*. Akcja ta została wykonana z poziomu konta **Administratora Domeny**. Zostało użyte następujące polecenie w celu przypisania usługi do konta:

```
1 setspn -a http/PC2.company.sec company.sec\Administrator
```

4. Testy bezpieczeństwa wdrożonej infrastruktury

Rozdział ten stanowi dokumentacje przeprowadzonej analizy bezpieczeństwa wdrożonej infrastruktury. Zostały w nim zaprezentowane zagrożenia wynikające z błędnej konfiguracji środowiska opartego o usługę *Active Directory*. Kolejne sekcje przedstawiają skutecznie przeprowadzone ataki. Każdy atak został omówiony, a następnie zaprezentowany w praktyce w sekcjach nazwanych „Scenariuszami ataku".

4.1. *Password spraying*

Za angielską nazwą *password spraying*, która w dosłownym tłumaczeniu oznacza „rozpylanie haseł" kryje się założenie, że atakowane będzie nie konkretne konto, ale grupa kont. Teoretycznie nie muszą być one połączone wspólną domeną, ale praktyce jest to atak przeprowadzany przede wszystkim na organizacje, w których pracownicy posiadają konta w systemie informatycznym. *Password spraying* jest to rodzaj ataku *brute-force*. Podobnie jak w przypadku ataku *Password Guessing* będziemy przy pomocy dużej ilości prób próbowali zgadnąć poprawne powiązanie loginu i hasła przypisanego do konta w domenie. Zamiast zgadywać hasło, będziemy jednak próbowali zgadnąć login. Loginem będzie często adres e-mail bądź w przypadku domeny *Active Directory*, będzie on podobny do formatu adresu e-mail. Potrzebna jest więc lista wielu loginów oraz jedno, wybrane hasło. Listę loginów można przygotować na kilka sposobów. Po pierwsze: jeśli loginem jest adres e-mail, to są one często podawane do publicznej wiadomości. Adres e-mail jest to informacja, która z założenia nie jest chroniona tajemnicą. Dlatego też łatwo znaleźć w Internecie liczne adresy e-mail dla wybranej domeny. Jeśli znajdziemy kilka reprezentatywnych przykładów dla konkretnej firmy z pewnością zauważymy, że adresy poczty elektronicznej rzadko są losowe - zazwyczaj powstają na bazie imienia i nazwiska pracownika organizacji. Można więc na podstawie przejętych adresów e-mail wywnioskować według jakiego wzoru budowane są adresy e-mail w danej organizacji. Czasem jest to *imię.nazwisko@domena.xyz*, innym razem pierwsza litera imienia oraz nazwisko (*inazwisko@domena.xyz*) a czasem *nazwisko.imie@domena.xyz*. Na tej podstawie można samodzielnie wygenerować prawdopodobnie występujące w organizacji loginy do kont użytkowników i utworzyć z nich listę. Kolejnym krokiem

jest wybranie hasła. Atak opiera się na założeniu, że przynajmniej jedno konto z listy będzie stosowało słabe i przewidywalne hasło. Jest to więc atak, mający większe szanse na powodzenie niż atak *password guessing*, gdyż nie jest atakowane tylko jedno konto [19].

4.1.1. Scenariusz ataku

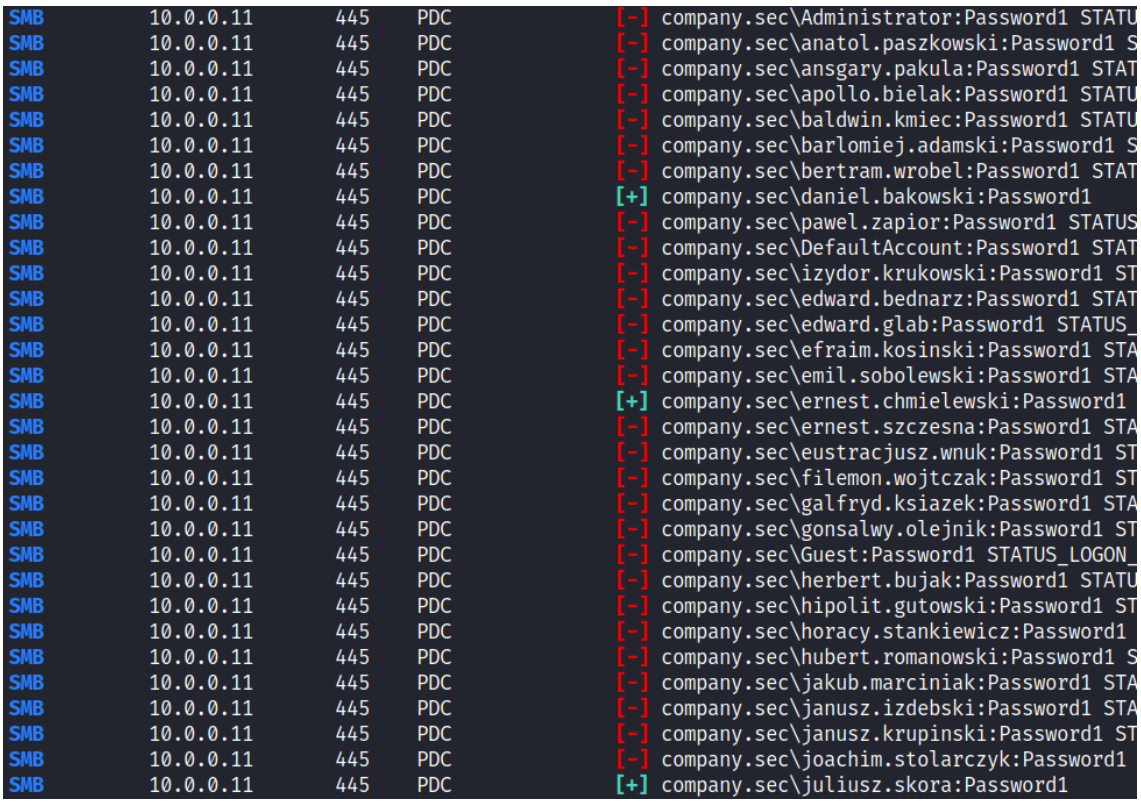
Atak ***Password Spraying*** zgodnie z klasyfikacją *MITRE ATT&CK* [19] zaliczany jest do ataków siłowych (ang. *Brute Force*). Zakłada on wykorzystanie jednego lub kilku haseł w celu uzyskania popranych danych uwierzytelniających konta domenowego w sposób opisany poprzednim podrozdziale. Atak ten jest powszechnie wykorzystywany przez wiele grup przestępczych, ponieważ minimalizuje on prawdopodobieństwo zablokowania pojedynczego konta domenowego, co mogłoby nastąpić przy wielokrotnych próbach logowania się na jedno konto wieloma hasłami. Wcześniej przejęty użytkownik posiadał bardzo proste hasło do swojego konta domenowego *'Password1'*. Warto sprawdzić, czy inni użytkownicy nie posiadają dokładnie takiego samego hasła. W tym celu użyte zostało narzędzie *crackmapexec*. Dokonano próby logowania na każdego użytkownika w domenie używając protokołu SMB na kontroler domeny „PDC" z hasłem *'Password1'*.

Pozytywny rezultat ataku pozwolił uzyskać dostęp potencjalnemu atakującemu do kont kolejnych dwóch użytkowników.

4.2. Atak LLMNR/NBT-NS Poisoning and SMB Relay

4.2.1. Systemy rozwiązywania nazw

Choć tłumaczenie nazw domenowych kojarzy się dziś dość jednoznacznie z systemem DNS, to nie jest on jedynym istniejącym rozwiązaniem zapewniającym rozwiązywanie nazw. Omówienie procesu rozwiązywania nazw nie jest przedmiotem zainteresowania pracy, zatem omówione zostaną jedynie podstawy niezbędne do zrozumienia przeprowadzonego ataku. Należy więc nadmienić, że system LLMNR (*Link Local Multicast Network Resolution*) umożliwia rozwiązywanie nazw w systemie zdecentralizowanym (bez udziału serwera). LLMNR działa w systemach od Windows Vista/Windows 2008 wzwyż i zastąpił w tej roli system *netBIOS*. Działa on tylko w obrębie sieci lokalnej. Domyślnie jest on włączony w systemach Windows. To czy jest aktywny na danym komputerze można sprawdzić uruchamiając kolejno: Panel sterowania ⟶ Sieć i Internet ⟶ Centrum sieci i udostępniania ⟶ Zaawansowane ustawienia udostępniania. Zaznaczona powinna zostać opcja „Włącz odnajdowanie sieci". Protokół działa wykorzystując lokalny adres multicastowy IPv6



Rysunek 7: Wykonanie ataku *Password spraying*. Znak zielonego plusa oznacza pomyśle zalogowanie się na konto użytkownika

FF02::1:3 oraz adres grupowy IPv4 224.0.0.252. Po rozpoczęciu wyszukiwania nazwy wysłane zostaje zapytanie na jeden z wyżej wymienionych adresów (domyślnie w systemie Windows na adres IPv6). Na tych adresach nasłuchują wszystkie inne komputery, które mają aktywowaną opcję odnajdowania sieci. I tutaj pojawia się niebezpieczeństwo, ponieważ zapytanie trafia nie tylko do właściwego komputera, ale może zostać przejęte przez atakującego, który znajduje się w sieci lokalnej.

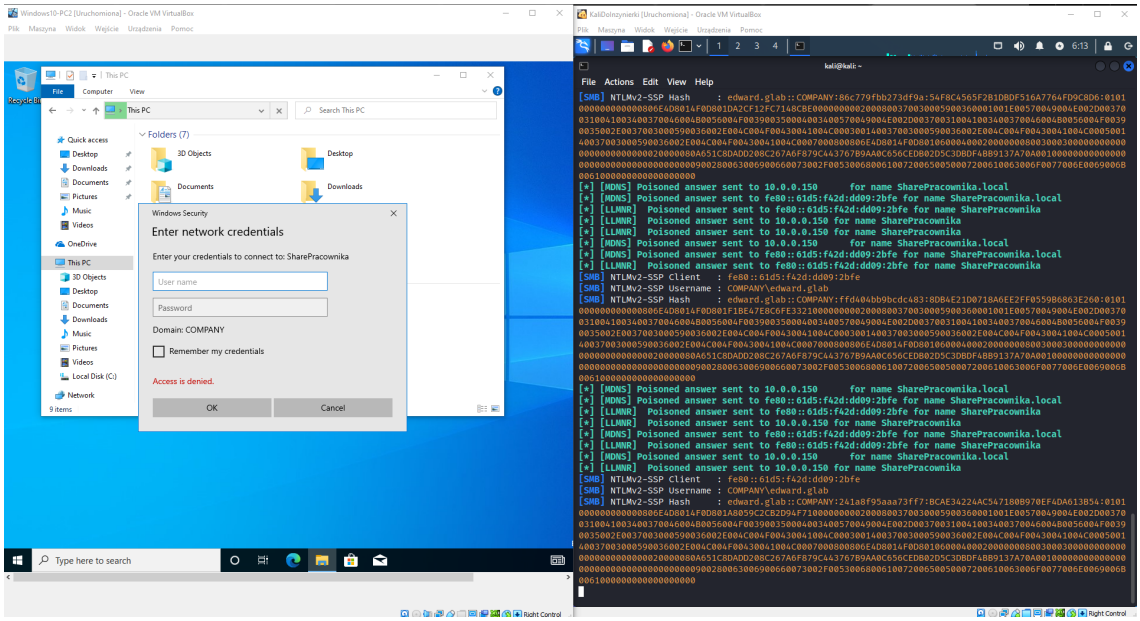
4.2.2. Atak LLMNR/NBT-NS Poisoning

Włączona obsługa LLMNR (*Link-Local Multicast Name Resolution*) oraz obsługa NBT-NS (*NetBIOS Name Service*) to podstawowe funkcjonalności działające w środowisku *Active Directory*. Wiąże się z nimi jednak pewne zagrożenie — jest to atak typu *Man in The Middle*, który może zostać przeprowadzony, gdy użytkownik poszukuje w sieci lokalnej zasobu i korzysta ze zdecentralizowanego systemu rozwiązywania nazw. Atak w swojej najprostszej postaci pozwoli uzyskać wynik funkcji skrótu *NetNTLM* użytkownika. W przypadku analizowanej infrastruktury będzie to wynik funkcji *NetNTLMv2*, ponieważ ta właśnie została zastosowana w procesie uwierzytelniania w systemach Windows Server 2016 oraz Windows 10. Wykorzystu-

jąc narzędzie z pakietu *Impacket* o nazwie *Responder* istnieje możliwość utworzenia „fałszywego” serwera SMB. Celem tego działania jest rozgłaszanie zasobu w sieci użytkownikowi, który poszukuje dowolnych zasobów sieciowych. Takie ataki nazywamy zatruwaniem (ang. „*poisoning*”), ponieważ do prawidłowo działającego systemu wprowadzany złośliwy zasób.

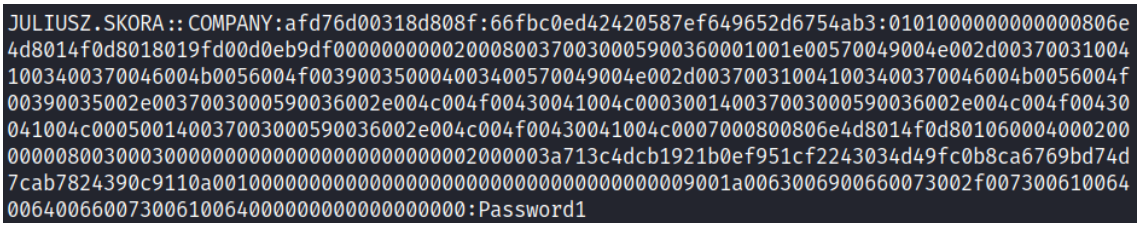
4.2.3. Scenariusz ataku *LLMNR/NBT-NS Poisoning*

Atakujący przebywa w firmie i podszywa się pod pracownika. Podczas przerwy na kawę powiedział, że na jego komputerze nie działa zasób sieciowy **\\Share**. Pan Edward Głąb (nazwa użytkownika: edward.glab) postanowił to sprawdzić. W momencie, gdy wyszukał na swoim komputerze udział o wspomnianej nazwie, rozgłosił się w sieci swoimi danymi uwierzytelniającymi. Ponieważ atakujący kontrolował wspomniany złośliwy zasób protokołu SMB, był on wstanie przechwycić dane uwierzytelniające pana Edwarda Głąba w formie loginu i wyniku funkcji skrótu z hasła (zwanego także „hashem”) *NetNTLMv2*.



Rysunek 8: Na komputerze Windows 10 „PC2” pracownik próbuje połączyć się z udziałem sieciowym. Tymczasem atakujący znajdujący się w sieci przechwytuje jego login i hasło w postaci skrótu NetNTLMv2. Źródło: opracowanie własne.

W międzyczasie do zasobu połączyli się inni użytkownicy. Atakujący zaczął proces łamania otrzymanych skrótów haseł. W tym celu wykorzystał narzędzie **hashcat**. Udało mu się uzyskać poświadczenia uwierzytelniające tylko jednego użytkownika *juliusz.skora:Password1*, ponieważ pan Juliusz używał wyjątkowo słabego hasła, które łatwo można było złamać.



Rysunek 9: Przechwycony skrót *NetNTLMv2* hasła udało się złamać i odczytać właściwe poświadczenia użytkownika juliusz.skora. Źródło: opracowanie własne.

4.2.4. *SMB Relay*

Wyżej opisany atak jest wyjątkowo groźny, ponieważ w dość prosty sposób pozwala na uzyskanie loginu i hasła jednego z pracowników. Należy jednak zauważyć, ze opiera się on na założeniu, że zaatakowany pracownik ma hasło na tyle słabe, że przechwycony skrót jest możliwy do złamania w pewnym akceptowalnym przez atakującego okresie czasu i przy użyciu ograniczonych zasobów mocy obliczeniowej. Jeżeli każdy pracownik w firmie posiadałby więc silne, np. 20 znakowe hasło złożone z losowych znaków to wyżej opisany atak nie przyniósłby rezultatu, ponieważ złamanie skrótu z takiego hasła zajęłoby czas i zasoby znacznie przekraczające możliwości współczesnych komputerów. Atakujący wykorzystuje więc narzędzie pakietu *Impacket* o nazwie **ntlmrelayx**. Pozwala ono na przekazanie istniejącej sesji protokołu SMB/HTTP na inne urządzenie (od 2008 roku nie można przekazać sesji na maszynę, na której ta sesja została utworzona [7]). Warunkiem koniecznym dla powodzenia takiego ataku jest to, że wartość ustawienia funkcjonalności *SMB signing* musi być równa *False*. Taki stan rzeczy domyślnie ma miejsce na urządzeniach klienckich z systemem operacyjnym z rodziny Windows, przez co one stają się głównym celem omawianego ataku. Urządzenia z zainstalowanym systemem Windows Server domyślnie mają ustawioną opcję *SMB signing* na wartość *True*, co czyli je odpornym na atak.

Co istotne powodzenie tego ataku daje znacznie więcej korzyści. Przekazując sesję zwykłego użytkownika na jakikolwiek komputer atakujący nie uzyska nic wartościowego. Jednakże przekazując sesję użytkownika, który posiada uprawnienia lokalnego administratora na komputerze, który otrzyma przekazaną sesję, wykonany zostanie zrzut lokalnych poświadczeń SAM.

Definicja 4.1. (SAM - *Security Account Manager*) *to baza danych, która na komputerach z systemem operacyjnym Windows przechowuje informacje o kontach użytkowników, informacje o przynależności do grup, domen, kontach komputera i inne kluczowe z punktu widzenia bezpieczeństwa informacje [8].*

4.2.5. Scenariusz ataku

Z poniższego zrzutu ekranowego można odczytać, że użytkownik „Administrator" wykonał połączenie z urządzenia o adresie IP 10.0.0.11 do komputerów o adresach IP 10.0.0.100 oraz 10.0.0.150. Poskutkowało to zrzutem lokalnych poświadczeń.

```
[*] Servers started, waiting for connections
[*] SMBD-Thread-5 (process_request_thread): Connection from COMPANY/ADMINISTRATOR@10.0.0.11 controlled, attacking target smb://10.0.0.100
[*] Authenticating against smb://10.0.0.100 as COMPANY/ADMINISTRATOR SUCCEED
[*] SMBD-Thread-5 (process_request_thread): Connection from COMPANY/ADMINISTRATOR@10.0.0.11 controlled, attacking target smb://10.0.0.150
[*] Service RemoteRegistry is in stopped state
[*] Starting service RemoteRegistry
[*] Authenticating against smb://10.0.0.150 as COMPANY/ADMINISTRATOR SUCCEED
[*] SMBD-Thread-5 (process_request_thread): Connection from COMPANY/ADMINISTRATOR@10.0.0.11 controlled, but there are no more targets left!
[*] Service RemoteRegistry is in stopped state
[*] Service RemoteRegistry is disabled, enabling it
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x236b3c25a0cbbde18bc7e3ae0986b238
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:acb40c73b7de2da04315ad09d3988fad:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] Target system bootKey: 0xb75c49c3f2e85986842896776479e97b
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] SMBD-Thread-8 (process_request_thread): Connection from COMPANY/ADMINISTRATOR@10.0.0.11 controlled, but there are no more targets left!
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:beb3493b92c3ab36698cae9472d4f49d:::
Administrator:500:aad3b435b51404eeaad3b435b51404ee:acb40c73b7de2da04315ad09d3988fad:::
anowak:1001:aad3b435b51404eeaad3b435b51404ee:c2d5208149e1a1bc823041734f20e754:::
[*] Done dumping SAM hashes for host: 10.0.0.100
[*] Stopping service RemoteRegistry
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:c6bbc3b7425de0d7b5baf275248269cc:::
jkowalski:1001:aad3b435b51404eeaad3b435b51404ee:c2d5208149e1a1bc823041734f20e754:::
[*] Done dumping SAM hashes for host: 10.0.0.150
[*] Stopping service RemoteRegistry
[*] Restoring the disabled state for service RemoteRegistry
```

Rysunek 10: Na zrzucie ekranowym użytkownik Administrator posiadający uprawnienia Administratora Domeny wyszukuje udziały sieciowe. Źródło: opracowanie własne.

```
$ cat 10.0.0.100_samhashes.sam
Administrator:500:aad3b435b51404eeaad3b435b51404ee:acb40c73b7de2da04315ad09d3988fad:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:beb3493b92c3ab36698cae9472d4f49d:::
anowak:1001:aad3b435b51404eeaad3b435b51404ee:c2d5208149e1a1bc823041734f20e754:::
$ cat 10.0.0.150_samhashes.sam
Administrator:500:aad3b435b51404eeaad3b435b51404ee:acb40c73b7de2da04315ad09d3988fad:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:c6bbc3b7425de0d7b5baf275248269cc:::
jkowalski:1001:aad3b435b51404eeaad3b435b51404ee:c2d5208149e1a1bc823041734f20e754:::
```

Rysunek 11: Zrzut ekranowy przedstawia sytuację gdy atakujący przechwycił poświadczenia użytkowników w postaci nazwy użytkownika i skrótu hasła. Źródło: opracowanie własne.

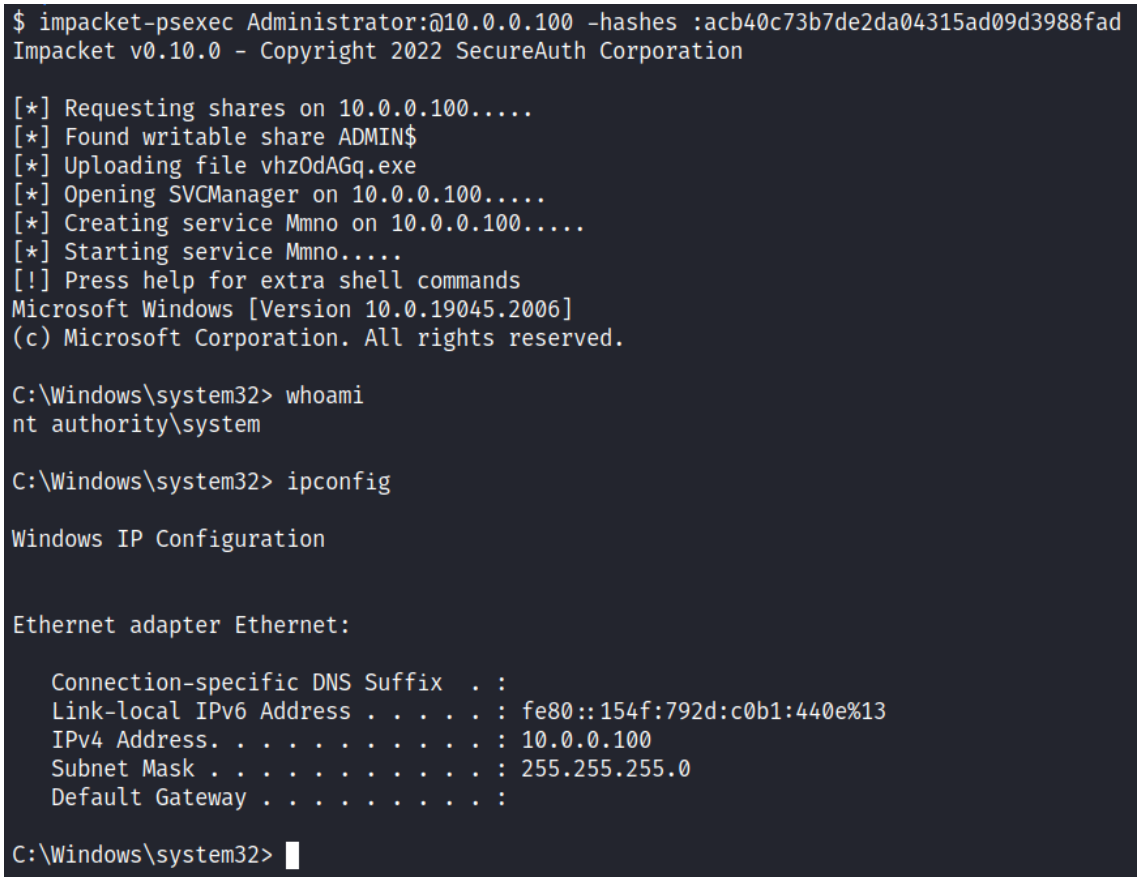
Zgodnie ze scenariuszem konfiguracyjnym przedstawionym w poprzednim rozdziale, wdrożone na potrzeby niniejszej pracy środowisko posiada włączone lokalne konta Administratorów o pełnych uprawnieniach na systemach klienckich.

4.3. Atak „*Pass the hash*"

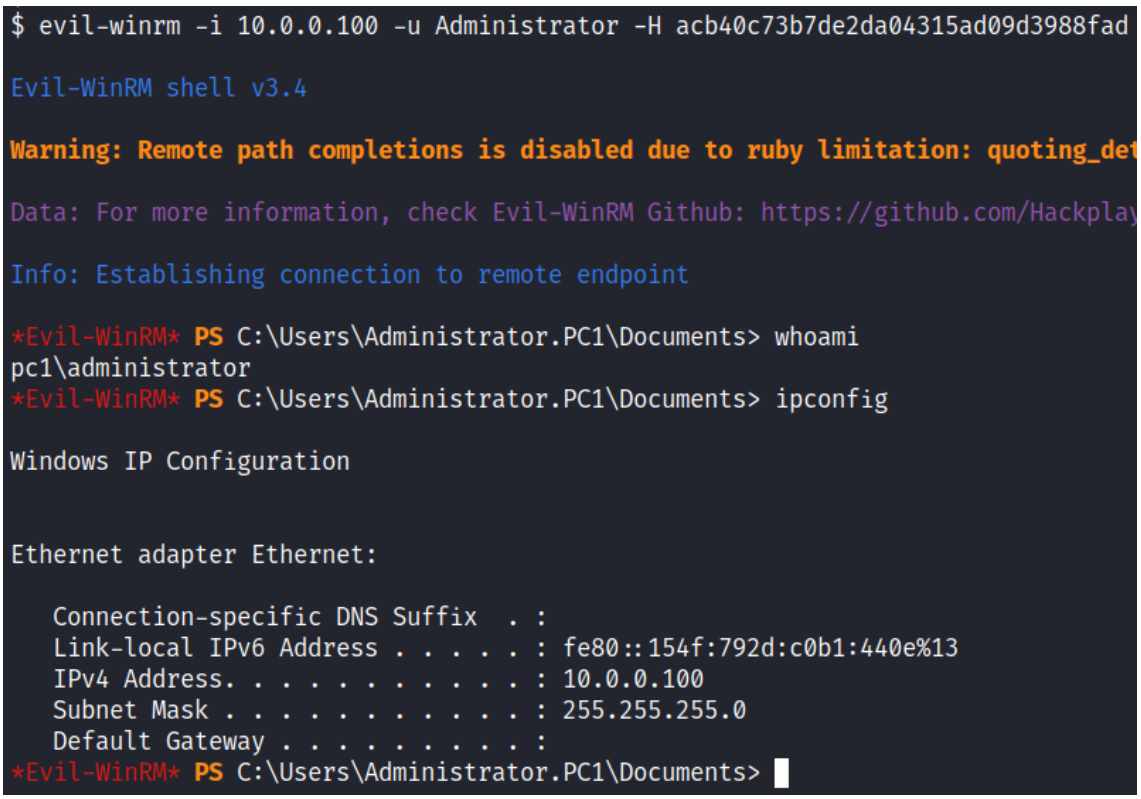
Atak „Pass the hash" (ang. „Przekaż skrót hasła") jest kontynuacją scenariusza przedstawionego w podrozdziale **LLMNR/NBT-NS Poisoning and SMB Relay**. Tym razem przyjęte zostaje założenie, że w rezultacie wymuszonej polityki haseł konto „Administrator" posiada bardzo silne hasło i mimo wielu tygodni łamania atakujący nie uzyskał oczekiwanego rezultatu. Atak „Pass the hash" zakłada, że istnieje możliwość zalogowania się do zasobów SMB, WinRM oraz RDP[1] bez znajomości hasła w postaci jawnego tekstu, a jedynie posiadając jego NTLM. Istotną uwagą, jaką należy poczynić jest to, że domyślnie taka metoda uwierzytelniania nie jest włączona dla protokołu RDP. Jest jednak aktywna dla protokołów SMB i WinRM. Na poniższych zrzutach ekranowych zostały pokazane sesje uzyskane za pomocą połączenia poprzez protokoły SMB oraz WinRM.

Definicja 4.2. (WinRM - *Windows Remote Management*) *to stworzona przez firmę Microsoft implementacja protokołu WS Management oparta na standardzie Simple Object Access Protocol pozwalająca na zdalne zarządzanie infrastrukturą IT. W przypadku omawianej infrastruktury będą to urządzenia Windows Server 2016 „PDC" oraz „DCRO" [9].*

[1] *Remote Desktop Protocol* - protokół umożliwiający zdalny dostęp do graficznego interfejsu systemu



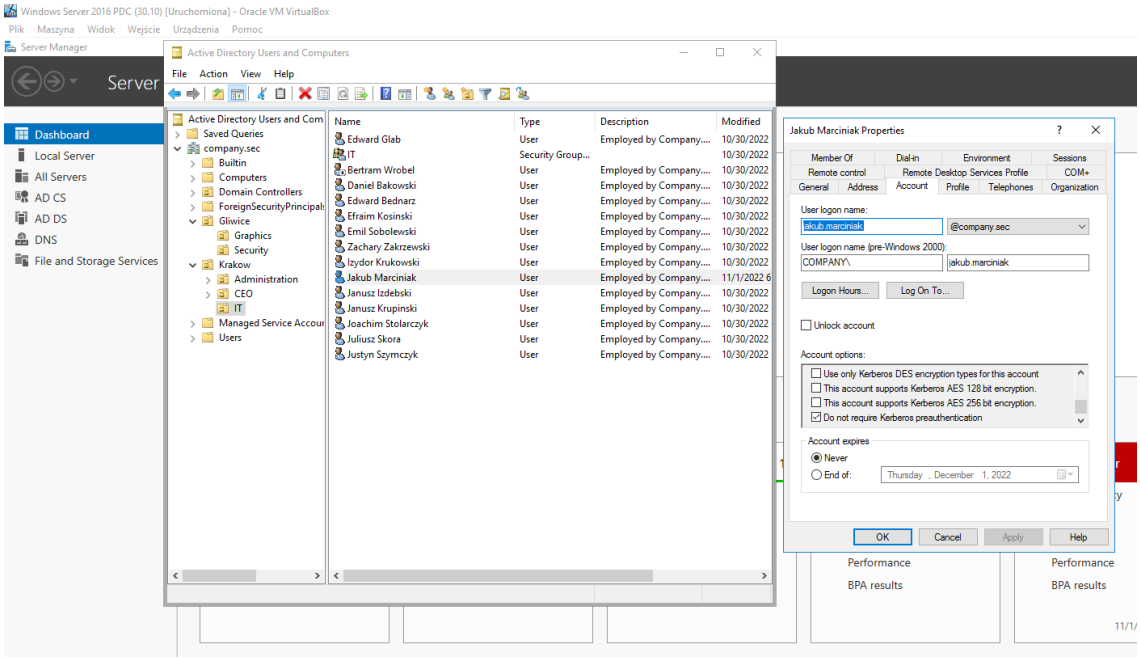
Rysunek 12: Zrzut ekranowy przedstawia autoryzację sesji protokołu SMB uzyskaną przy pomocy ataku *„Pass the hash”*. Użyto narzędzia *Impacket*. Źródło: opracowanie własne.



Rysunek 13: Zrzut ekranowy przedstawiający autoryzację sesji protokołu WinRM uzyskaną przy pomocy ataku *„Pass the hash”*. Użyto narzędzia EvilWinRM. Źródło: opracowanie własne.

4.4. AS-REP Roasting

Atak o nazwie *AS-REP Roasting* zakłada zastosowanie niebezpiecznej konfiguracji konta użytkownika polegającej na włączeniu opcji ***Do not require Kerberos preauthentication*** (Nie wymagaj preuwierzytelniania protokołu Kerberos). Poniżej przedstawiony został zrzut ekranowy, pokazujący gdzie można wprowadzić omawiane ustawienie.

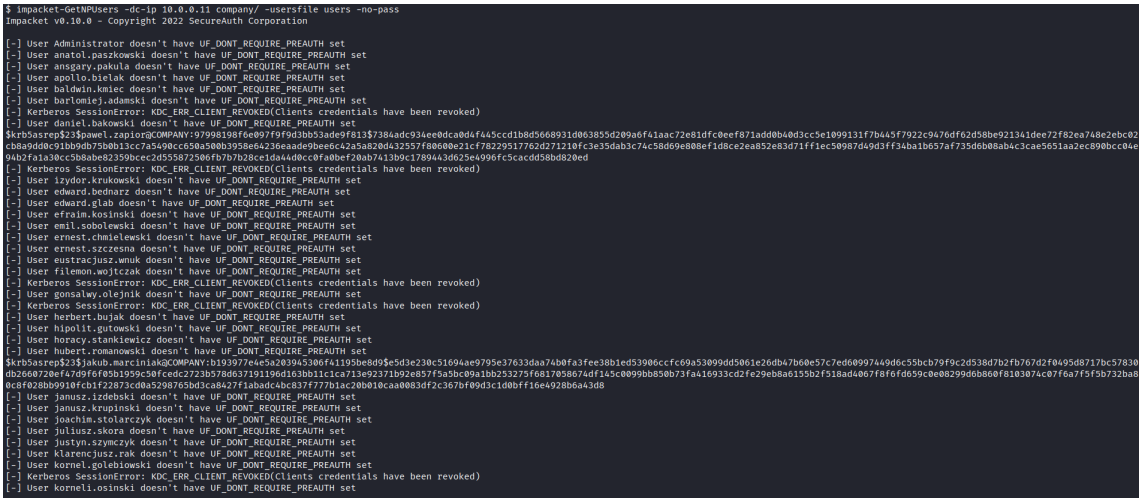


Rysunek 14: Zrzut ekranowy przedstawia aktywowanie opcji **Do not require Kerberos preauthentication** dla użytkownika jakub.marciniak. Źródło: opracowanie własne.

4.4.1. Scenariusz ataku

Wykorzystując fakt, że atakujący przejął już konto użytkownika domenowego w ramach pierwszego ataku przedstawionego w ramach testów bezpieczeństwa infrastruktury (Nazwa użytkownika: juliusz.skora, Hasło: Password1) ma on możliwość wykonania zrzutu całej struktury bazy LDAP co pozwoli mu odkryć wszystkich użytkowników w domenie. Następnie korzystając ze skryptu z narzędzia *Impacket* o nazwie *GetNPUsers* atakujący może sprawdzić czy, którykolwiek z użytkowników w domenie posiada aktywowaną opcję ***Do not require Kerberos preauthentication*** i uzyskać skrót jego hasła **Kerberos 5, etype 23, AS-REP**.

Jak widać na powyższym zrzucie ekranowym, konta dwóch użytkowników zostały skonfigurowane w ten sposób. Atak spowodował przejęcie skrótów ich haseł.



Rysunek 15: Przy pomocy pakietu *Impacket* atakujący sprawdza, czy istnieją użytkownicy domenowi, od których preuwierzytelnianie protokołu Kerberos nie jest wymagane. Źródło: opracowanie własne.

Łamiąc je atakujący jest wstanie przejąć ich konta i wykonywać wszystkie działania w domenie w ich imieniu. W tym przypadku użytkownik pawel.zapior posiada uprawnienia Administratora Domeny więc przejęcie jego konta daje atakującemu uprawnienia do zarządzania domeną.

4.5. Kerberoasting

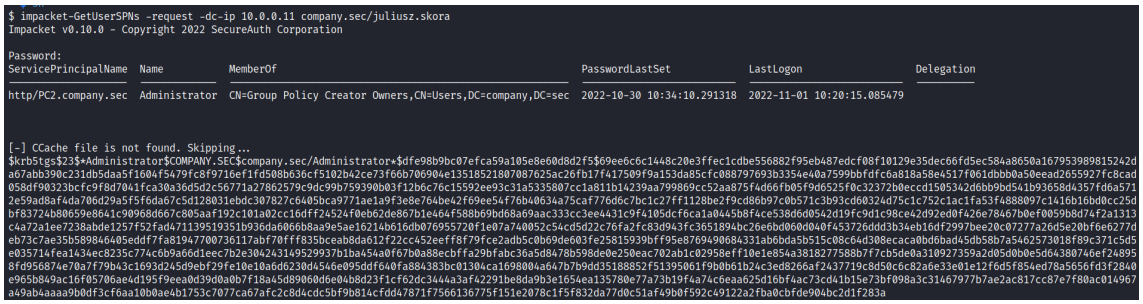
Na wstępie warto przypomnieć, że we wdrożonym środowisku przypisane zostało konto użytkownika Administrator do usługi stron Internetowych IIS. Skutkiem takiej konfiguracji każdy użytkownik domenowy jest wstanie zażądać biletu do tej usługi, a w odpowiedzi otrzyma zahashowane hasło konta przypisanego do SPN w formacie **$krb5tgs$23$*** (jeżeli poprosi o szyfrowanie RC4) albo **$krb5tgs$18$*** (jeżeli poprosi o szyfrowanie AES-256).

Do usługi działającej w ramach *Active Directory* można przypisać zarówno konto użytkownika jak i konto maszynowe. Celem ataku Kerberoasting jest przechwycenie biletu TGS dla usługi, która została przypisana do konta użytkownika. Usługi powiązane z kontem komputera są odporne na atak, ponieważ hasła kont maszynowych są niemożliwe do złamania ze względu na długość (120 znaków) i są automatycznie zmieniane co 30 dni w podstawowej konfiguracji. W związku z tym część biletów TGS jest szyfrowana za pomocą kluczy bazujących na hasłach użytkowników. W przypadku powodzenia ataku przechwycone dane uwierzytelniające mogą zostać złamane na lokalnej maszynie atakującego. Z tego powodu do Kerberoasting

atakujący potrzebuje tylko konta domenowego standardowego użytkownika, które może zażądać biletu TGS. Nie są wymagane inne specjalne uprawnienia [20].

4.5.1. Scenariusz ataku

Aby zażądać biletu TGS dla usługi wykorzystano skrypt pakietu Impacket GetUserSPNs podając w zapytaniu dane uwierzytelniające przejętego wcześniej użytkownika domenowego.



Rysunek 16: Uzyskanie zahashowanych danych uwierzytelniających poprzez atak *Kerberoasting*. Źródło: opracowanie własne.

4.6. DNS IPv6 Spoofing

Ostatnim przedstawionym atakiem będzie *DNS IPv6 Spoofing*. Atak ten polega na podszywaniu się pod serwer DNS i wykorzystaniu niebezpiecznej, podstawowej konfiguracji komputerów z systemami z rodziny Microsoft Windows. Zarówno systemy użytkowe jak i serwerowe priorytetowo wybierają komunikacje z serwerem DNS, który rozgłasza się na bazie protokołu IPv6. Z doświadczenia zawodowego Bartłomieja Adamskiego wynika, że zdecydowana większość sieci korporacyjnych ma skonfigurowany serwer DNS na bazie adresu IPv4. Na komputerach i serwerach obsługa IPv6 domyślnie nie jest jednak wyłączona. Stwarza to zagrożenie w postaci możliwości uruchomienia własnego serwera DNS działającego na bazie protokołu IPv6. Gdy owy serwer się pojawi, wszystkie zapytania o rozwiązanie nazw domenowych zaczną być kierowane do złośliwego serwera DNS.

4.6.1. Scenariusz ataku

Wykorzystując narzędzie mitm6 [23] zostaje utworzony złośliwy serwer DNS działając wykorzystujący do komunikacji protokół IPv6.

Na zamieszczonych zrzutach ekranowych przedstawiono widok z komputera klienckiego przed i po uruchomieniu serwera DNS IPv6.

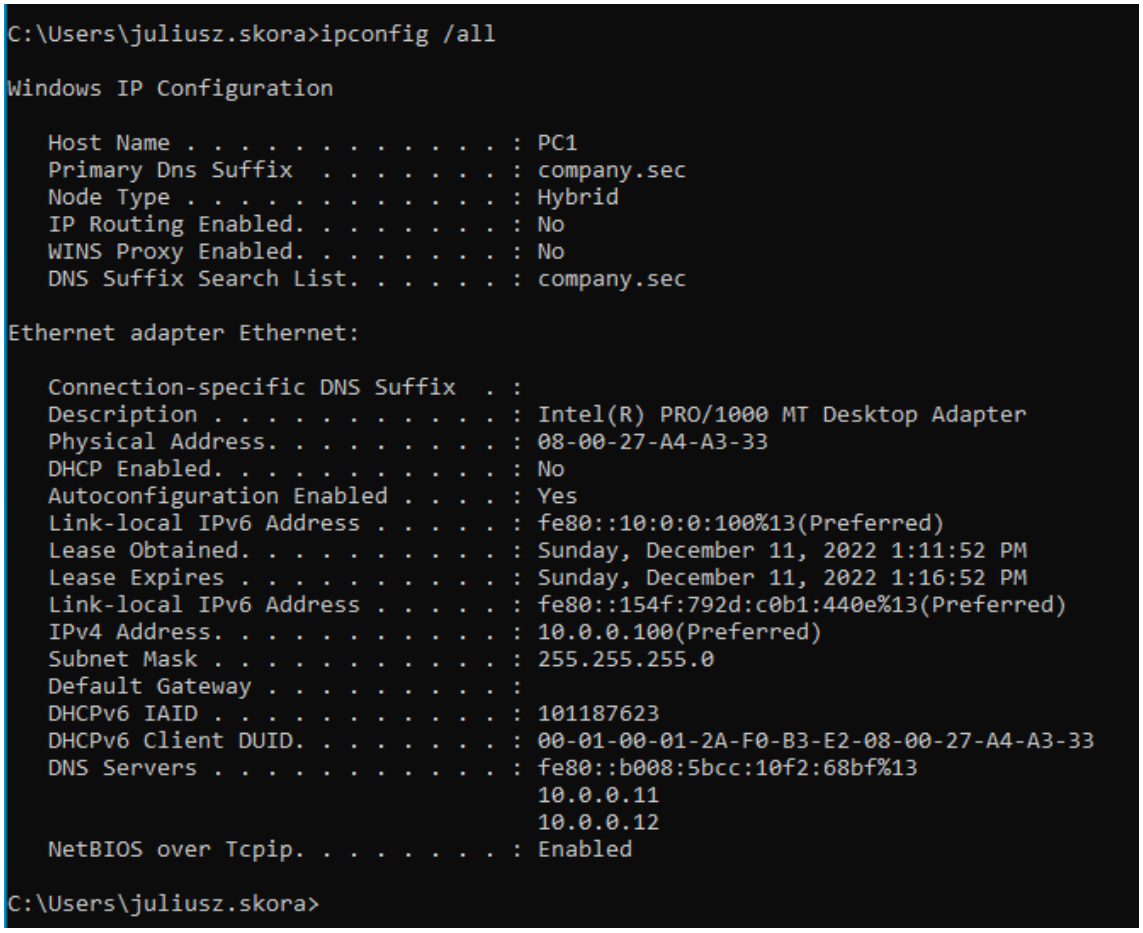
```
└─$ sudo mitm6 -i eth0
[sudo] password for kali:
Starting mitm6 using the following configuration:
Primary adapter: eth0 [08:00:27:92:f9:72]
IPv4 address: 10.0.0.201
IPv6 address: fe80::b008:5bcc:10f2:68bf
Warning: Not filtering on any domain, mitm6 will reply to all DNS queries.
Unless this is what you want, specify at least one domain with -d
IPv6 address fe80::10:0:0:100 is now assigned to mac=08:00:27:a4:a3:33 host=PC1.company.sec. ipv4=10.0.0.100
IPv6 address fe80::863:1 is now assigned to mac=0a:00:27:00:00:17 host=DESKTOP-QCSC7JU. ipv4=
Sent spoofed reply for safebrowsing.googleapis.com. to fe80::35d5:a535:fee6:4c61
Sent spoofed reply for wpad.company.sec. to fe80::154f:792d:c0b1:440e
Sent spoofed reply for wpad.company.sec. to fe80::154f:792d:c0b1:440e
Sent spoofed reply for api.msn.com. to fe80::154f:792d:c0b1:440e
```

Rysunek 17: Zrzut ekranowy przedstawia włączenie serwera DNS działającego na bazie protokołu IPv6. Źródło: opracowanie własne.

```
C:\Users\juliusz.skora>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : PC1
   Primary Dns Suffix  . . . . . . . : company.sec
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : company.sec

Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Desktop Adapter
   Physical Address. . . . . . . . . : 08-00-27-A4-A3-33
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::154f:792d:c0b1:440e%13(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.0.0.100(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
   DHCPv6 IAID . . . . . . . . . . . : 101187623
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2A-F0-B3-E2-08-00-27-A4-A3-33
   DNS Servers . . . . . . . . . . . : 10.0.0.11
                                       10.0.0.12
   NetBIOS over Tcpip. . . . . . . . : Enabled

C:\Users\juliusz.skora>
```

Rysunek 18: Zrzut ekranowy przedstawia widok z komputera klienckiego przed włączeniem serwera DNS atakującego. Źródło: opracowanie własne.

Atak ten jest niebezpieczny, ponieważ podczas rozwiązywania nazwy domenowej użytkownik przesyłający zapytanie zawsze deklaruje swoją tożsamość przesyłając swoje dane uwierzytelniające w postaci nazwy użytkownika i skrótu hasła. Pojawia się więc możliwość przechwytywania wrażliwych danych uwierzytelniających.



Rysunek 19: Zrzut ekranowy przedstawia widok z komputera klienckiego po włączeniu serwera DNS atakującego. Źródło: opracowanie własne.

4.7. Podatności typu „Zero-day”

Jest jeszcze jeden istotny aspekt bezpieczeństwa wszelkich systemów informatycznych. Chodzi o podatności typu *„Zero-day”* — termin ten oznacza luki, które zostały wykryte bardzo niedawno i nie istnieją jeszcze odpowiednie poprawki bezpieczeństwa, które mogłyby by błąd naprawić. *„Zero-day”* są o tyle niebezpieczne, że użytkownik nie ma możliwości ich naprawić do momentu pojawienia się aktualizacji. Jest to specyficzna grupa zagrożeń, ponieważ obrona przed podatnościami typu *„zero-day”* opiera się w większości przypadków na bieżącym aktualizowaniu oprogramowania zaraz gdy tylko pojawi się łatka bezpieczeństwa. W praktyce *„Zero-day”* nie są możliwe do pełnego wyeliminowania, ponieważ wraz z powstawaniem nowych funkcjonalności oprogramowania, zawsze popełnione zostaną jakieś błędy, gdyż człowiek z natury jest istotą omylną. Okazuje się też, że w praktyce nie są one wcale najczęściej wykorzystywane, ponieważ zwyczajnie nie jest to opłacalne dla grup przestępczych. Bazując na „*Data Breach Investigations Report*”, który dostarczany jest

co roku przez Verizon Business okazuje się, że najczęściej występującą podatnością wykrywaną w systemach teleinformatycznych badanych organizacji wciąż jest *EternalBlue* (CVE-2017-0144) upubliczniony w 2017 roku. To właśnie takie podatności są najbardziej niebezpieczne, ponieważ ich użycie nie wymaga wysokich umiejętności technicznych. Istnieje wiele dobrze udokumentowanych gotowych narzędzi pozwalających na ich łatwe wykorzystanie. Problem z przedstawieniem podatności typu *„Zero-day"* w pracy dyplomowej ma charakter praktyczny, gdyż od momentu publikacji danej podatności do jej załatania przez dostawcę oprogramowania, zazwyczaj mija znacznie mniej czasu niż od momentu rozpoczęcia tworzenia pracy dyplomowej do jej obronienia. Omawianie niebezpieczeństw związanych z konfiguracją ma zaś znacznie bardziej uniwersalny charakter i pozostaje aktualne przez dość długi czas.

4.8. Podsumowanie

Testy bezpieczeństwa systemów uwierzytelnienia *Active Directory* są procesem, w którym sprawdzane są różne aspekty bezpieczeństwa systemów uwierzytelniania, takie jak konta użytkowników, uprawnienia dostępu, polityki haseł czy pozostawione podstawowe niebezpieczne konfiguracje. Celem tych testów jest zidentyfikowanie słabych punktów w systemie uwierzytelniania i zaproponowanie sposobów poprawy bezpieczeństwa. Ważne jest, aby systemy uwierzytelniania były odpowiednio chronione, ponieważ są one kluczowe dla ochrony dostępu do danych i zasobów sieciowych.

Podsumowanie

Po wdrożeniu środowiska opartego o usługę *Active Directory* zgodnie z projektem przedstawionym w rozdziale „Wdrożenie infrastruktury opartej na usłudze *Active Directory*" przeprowadzona została analiza bezpieczeństwa tego środowiska. Przedstawione zostały ataki, które naruszyły w sposób istotny bezpieczeństwo domeny opartej na usłudze *Active Directory*. Bazując na doświadczeniu uzyskanym podczas testów penetracyjnych można określić jakie są przyczyny wykrytych podatności oraz jak zmodyfikować konfigurację, aby zwiększyć poziom bezpieczeństwa.

Podsumowanie wykrytych zagrożeń

W ramach przeprowadzonych testów penetracyjnych wdrożonej infrastruktury okazało się, że wykryte zostały następujące problemy z bezpieczeństwem:

- Część użytkowników stosuje słabe, łatwe do złamania hasła. To wskazuje brak wdrożenia odpowiedniej polityki haseł w dla użytkowników domenowych.
- Niektóre konta użytkowników mają wyłączony wymóg preuwierzytelniania. To pozwala na pomyślne przeprowadzenie ataku „AS-REP Roasting".
- Konto użytkownika „Administrator" zostało powiązane z usługą serwera stron Internetowych IIS. Konto to posiada znacząco wyższe uprawnienia niż jest to potrzebne do działania usługi. To zwiększa ryzyko przeprowadzenia skutecznego ataku **Kerberoasting** na konto Administracyjne.
- Istnieje możliwość podszycia się pod preferowany serwer DNS ponieważ obsługa protokołu IPv6 jest włączona, ale komunikacja w sieci nie jest na jego bazie realizowana.

Podsumowanie działań naprawczych

Dzięki analizie bezpieczeństwa wdrożonego środowiska udało się wykryć zagrożenia związane z uwierzytelnianiem w domenie *Active Directory*. Na tej podstawie można też wywnioskować jakie akcje powinny zostać podjęte przez administratora infrastruktury IT w celu mitygacji zagrożeń. Zebrane zostały w postaci niniejszej listy:

- Należy pozostawić nieaktywnymi konta lokalnego Administratora na komputerach podłączonych do domeny.
- Należy wdrożyć politykę haseł dla użytkowników. Warto tutaj przypomnieć warunki jakie spełnia mocne hasło z punktu widzenia współczesnych możliwości ich łamania:
  - Hasło zawiera co najmniej 14 znaków
  - Hasło zawiera:
    * wielkie litery,
    * małe litery,
    * cyfry,
    * symbole.
  - Hasło nie jest oparte na wyrazie. W szczególności hasło nie powinno zawierać nazwy organizacji i wyrazów w nią powiązanych, gdyż jest to wykorzystywane przez atakujących do ataku typu *Password Spraying*.
  - Hasło nie zostało skompromitowane w przeszłości i nie było używane w innych serwisach.

  Niestety, nie wszystkie z wyżej wymienionych punktów można wdrożyć w organizacji. Nie jest możliwe kontrolowanie pracowników pod kątem budowania haseł w oparciu o wyrazy słownikowe czy unikalności stosowanych haseł. Jest jednak możliwe wymuszenie w ramach polityki haseł ich długości, złożoności w znaczeniu ilości rodzajów użytych znaków oraz zdefiniowanie częstotliwości obowiązku zmiany hasła. Jeżeli organizacja korzysta z Azure *Active Directory* możliwe jest także zdefiniowanie listy haseł, których pracownicy nie będą mogli użyć.
- Warto zapewnić szkolenie pracowników w zakresie cyberbezpieczeństwa. Sumienni pracownicy będą stosować się do przekazanych instrukcji. Będą stosowali silniejsze hasła i będą bardziej nieufni w miejscu pracy wobec osób, których nie znają. Będą też bardziej czujni podczas przeglądania skrzynki mailowej.
- Nigdy nie należy wyłączać preuwierzytelniania protokołu Kerberos na kontach użytkowników.
- Należy instalować poprawki bezpieczeństwa.

- Konta powiązane z usługami *Active Directory* powinny mieć najmniejsze możliwe uprawnienia dostosowane do potrzeb danej usługi i nie powinny być używane przez żadnych pracowników.
- Należy ograniczyć użycie niebezpiecznych algorytmów takich jak RC4. Zamiast tego warto stosować AES-128, AES-256 oraz AES-512, które są powszechnie uznawane za bezpieczne.
- Wdrożenie komunikacji na bazie IPv6. Alternatywnie: wyłączenie obsługi IPv6 — to może jednak powodować niedostępność niektórych serwisów.

Wdrożenie poprawek bezpieczeństwa

Kiedy administrator w firmie *„company.sec"* otrzyma raport z testów bezpieczeństwa infrastruktury, powinien niezwłocznie upewnić się, czy istnieje możliwość wdrożenia poprawek zasugerowanych w raporcie, a następnie wprowadzić zmiany. Niestety, w praktyce nie zawsze wszystkie działania naprawcze mogą zostać szybko przeprowadzone. Często organizacje stosują niebezpieczne konfiguracje w celu utrzymania kompatybilności wstecznej ze starszymi urządzeniami. Wówczas wprowadzenie zmian może skutkować zaburzeniami działania całego systemu teleinformatycznego.

Podejmowaniem decyzji dotyczących wykrytych podatności zajmuje się dział cyberbezpieczeństwa zwany **zarządzanie ryzykiem**. Ryzyko można:

- **zaakceptować** (ang. *risk acceptance*) — firma może uznać, że ryzyko związane z daną podatnością jest małe, a koszt działań naprawczych jest niewspółmiernie duży w stosunku do zysku, jakim jest zwiększenie poziomu bezpieczeństwa. Działania naprawcze nie zostaną więc podjęte.
- **uniknąć** (ang. *risk avoidance*) — firma może zdecydować, że utrzymanie bezpiecznego środowiska powiązanego usługą *Active Directory* jest zbyt kosztowne i zrezygnować z dalszego stosowania tego rozwiązania.
- **przenieść** (ang. *risk transfer*) — firma może zdecydować, że sama nie jest w stanie poradzić sobie z zapewnieniem bezpieczeństwa usługi *Active Directory*. Zatrudni więc do tego celu zewnętrzną firmę, na której będzie spoczywać odpowiedzialność za zapewnienie bezpieczeństwa.
- **mitygować** (ang. *risk mitigation*) — firma może naprawić podatności zgodnie z sugestiami wynikającymi z raportu z testów bezpieczeństwa.

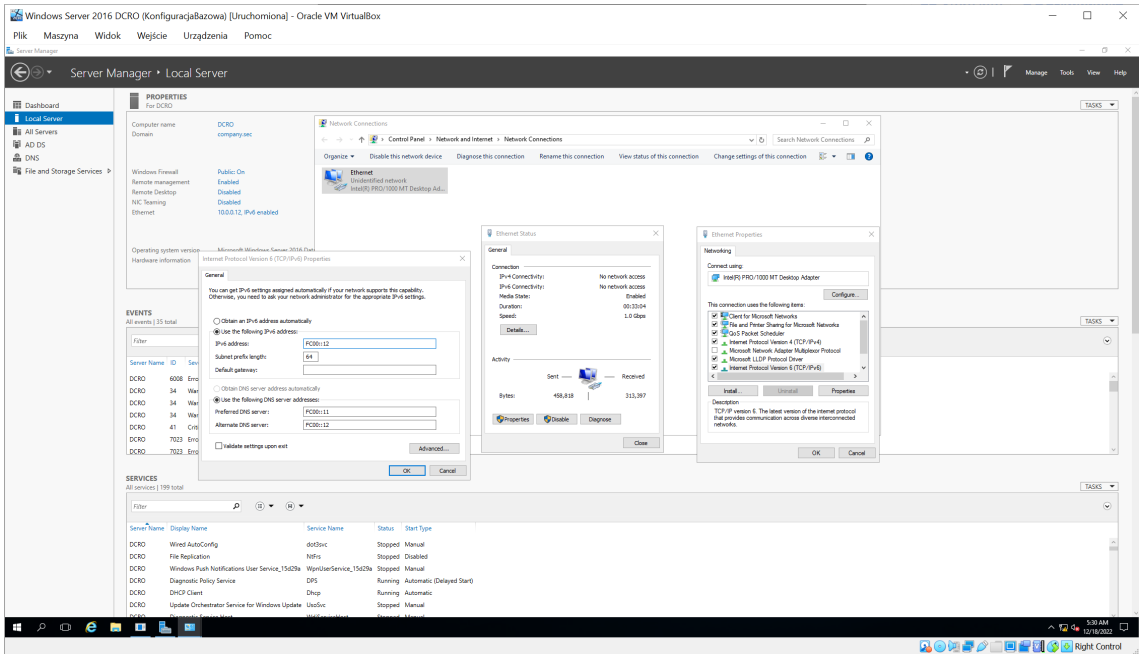
- **eksploatować** (ang. *risk exploitation*) — dotyczy to sytuacji, kiedy ryzyko okazuje się pozytywnym czynnikiem, na przykład, kiedy zapewnia produktowi większa popularność.

W przypadku infrastruktury organizacji *„company.sec”* wszystkie zalecenia mogą zostać wdrożone. Poniżej przedstawiona została lista podjętych działań naprawczych.

1. Wdrożono komunikację między komputerami na bazie protokołu IPv6. Urządzeniom przydzielono adresy unikalne lokalne:

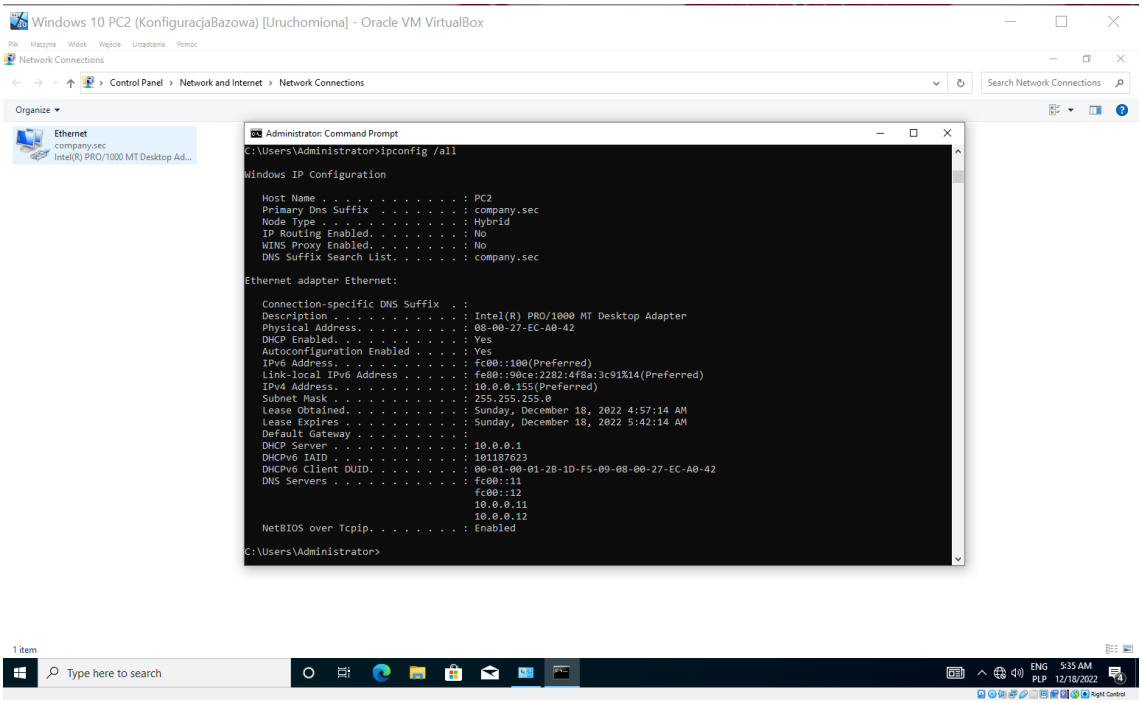
   - Windows Server 2016 „PDC” — *FC00::11*,
   - Windows Server 2016 „DCRO” — *FC00::12*,
   - Windows 10 „PC1” — *FC00::101*,
   - Windows 10 „PC2” — *FC00::102*.

   Jako preferowane serwery DNS na komputerach z systemem Windows 10 przypisano adresy FC00::11 oraz FC00::12.

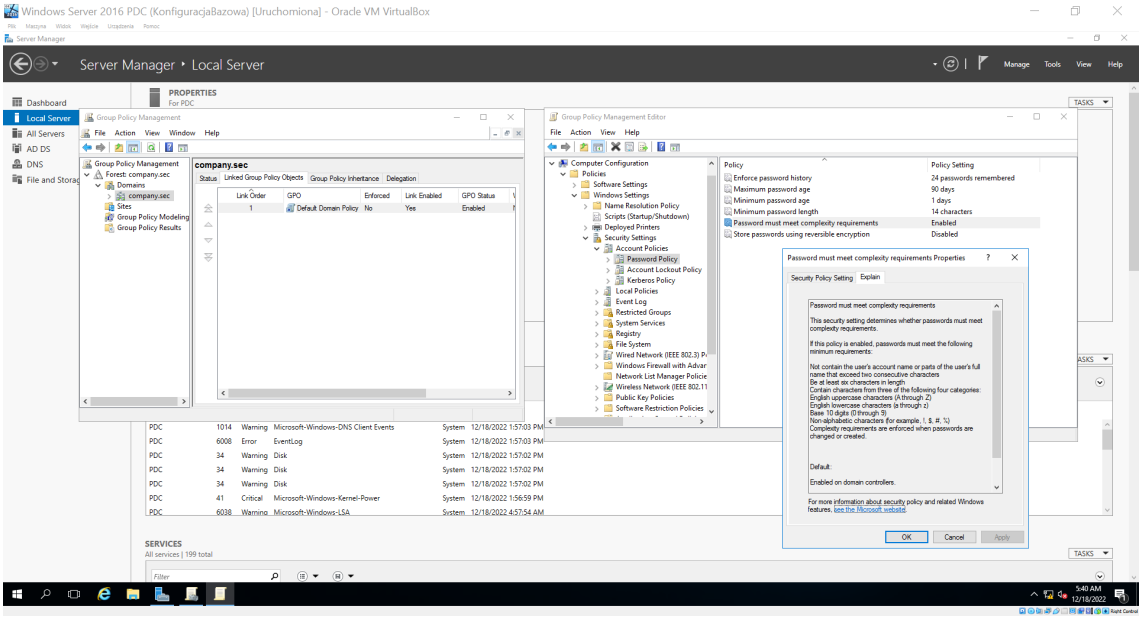


Rysunek 20: Przykładowa konfiguracja protokołu IPv6 na urządzeniu Windows Server 2016 „DCRO”

2. Zastosowano zasady grup wymuszające na użytkownikach stosowanie silnych haseł. Jako minimalną długość hasła przyjęto 14 znaków.



Rysunek 21: Weryfikacja kolejności preferowanych serwerów DNS na komputerze Windows 10 „PC1”. Użyto polecenia *ipconfig /all*. Źródło: opracowanie własne.



Rysunek 22: Konfiguracja polityki haseł dla użytkowników w domenowych. Źródło: opracowanie własne.

3. Dezaktywowano lokalne, wbudowane konta użytkownika *Administrator* na komputerach z systemem Windows 10.

4. Zweryfikowano, czy najnowsze poprawki bezpieczeństwa są zainstalowane.

5. Wyłączono możliwość stosowania szyfru *RC4* z poziomu kontrolerów domeny. W tym celu naniesiono następujące zmiany w rejestrze systemu na urządzeniach „PDC” oraz „DCRO” [24]:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
   SecurityProviders\SCHANNEL\Ciphers\RC4 128/128]
"Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
   SecurityProviders\SCHANNEL\Ciphers\RC4 40/128]
"Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
   SecurityProviders\SCHANNEL\Ciphers\RC4 56/128]
"Enabled"=dword:00000000
```

6. Utworzono konto standardowego użytkownika *http*. Hasło dla konta ma 60 znaków i jest wygenerowane metodą pseudolosową. Szacuje się, że złamanie takiego hasła zajęłoby około $100 \cdot 10^{93}$ lat [25]. Konto przypisano do usługi **http** zamiast konta *Administrator*.

```
setspn -a http/PC2.company.sec company.sec\http
```

7. Dla wszystkich kont użytkowników wymagane jest preuwierzytelnianie.

Wnioski

Usługa *Active Directory* w istotnym stopniu pomaga w zarządzaniu dużymi systemami teleinformatycznymi. Możliwość dodania komputerów, drukarek, przełączników i serwerów do domeny rozwiązuje też wiele problemów związanych z bezpieczeństwem. Jednym z najistotniejszych jest możliwość uniknięcia konieczności stosowania kont współdzielonych, ponieważ każdy pracownik posiada swoje uniwersalne konto. Scentralizowane zarządzanie dostępami także w istotny sposób poprawia bezpieczeństwo organizacji wdrażającej domenę *Active Directory*. Odrębne zarządzanie dostępami dużej ilości użytkowników do dużej ilości usług prędzej czy później zacznie być przytłaczające dla pracowników działu IT, którzy w rezultacie mogą popełniać błędy w konfigurowaniu reglamentacji dostępu. W końcu warto zwrócić uwagę na funkcjonalność scentralizowanego uwierzytelniania z perspektywy standardowego użytkownika, który jednym hasłem może zalogować się do wielu usług.

Dzięki temu nie musi pamiętać dużej ilości haseł, nie czuje też potrzeby zapisywania haseł na kartce (co jest szczególnie niebezpieczne). Jedno, wybrane przez użytkownika hasło może być silniejsze i trudniejsze do złamania przez atakującego. W ramach części teoretycznej przeanalizowane zostało działanie mechanizmów uwierzytelniania w domenie *Active Directory*. Przedstawione zostało działanie protokołu Kerberos oraz niektórych innych możliwości usługi *Active Directory* odnoszących się do uwierzytelniania użytkowników. Przeprowadzona analiza wykazała, że usługa *Active Directory* jest bezpieczna pod warunkiem zastosowania odpowiedniej konfiguracji. Etap testów penetracyjnych wykazał liczne podatności analizowanego środowiska. Co jednak najważniejsze, wszystkie były rezultatem nieodpowiedniej konfiguracji. Każda z przedstawionych podatności posiada też dobrze udokumentowaną metodę naprawy. Częstym wektorem ataku są zwykli pracownicy, którzy czasem nie rozumieją niebezpieczeństw związanych z procesem uwierzytelnianiem. Nie zdają sobie sprawy, że wystarczy nieodpowiedzialne zachowanie standardowego użytkownika z ograniczonym dostępem, aby spowodować poważne problemy dla organizacji. Nie jest to jednak nic dziwnego, gdyż jest to wiedza skomplikowana. Dlatego tak ważne są szkolenia z zakresu cyberbezpieczeństwa dla pracowników nietechnicznych. Z drugiej strony, przejęcie konta standardowego użytkownika nie prowadzi jeszcze do kompromitacji całej domeny. Tutaj dużą rolę odgrywa problem nieodpowiedzialnych zachowań administratorów, którzy próbują uprościć swoją pracę. Problemem są też często decyzje biznesowe, które stoją w sprzeczności do generującej ogromne koszty dbałości o cyberbezpieczeństwo. Długo można wymieniać przyczyny, dla których usługi związane z usługą *Active Directory* są częstymi wektorami ataku, ale powyższe przykłady wystarczą, aby dojść do wniosku, że zawsze na końcu jest człowiek, który podejmuje decyzje o tym jak duże zasoby (finansowe, intelektualne, czasowe) poświęcone zostaną na dbałość o cyberbezpieczeństwo.

„Nieważne jednak jak wiele wydaje się na rozwiązania technologiczne. System bezpieczeństwa jest tak silny jak jego najsłabsze ogniwo - człowiek"

Kevin Mitnick [15]

Uwagi

„Company.sec" jest przykładową nazwą organizacji i nie istnieje w rzeczywistości. Imiona i nazwiska pracowników organizacji Company zostały wygenerowane metodą pseudolosową z użyciem internetowego generatora `https://ohmyluck.com/`

`pl/random-name/`. Wszelkie podobieństwa są przypadkowe. Przedstawiona konfiguracja została przygotowana na podstawie arbitralnie podjętych przez autorów decyzji i nie jest odzwierciedleniem środowiska działającego w żadnej realnie funkcjonującej firmie lub organizacji.

Podziękowania

Dr inż. Adrianowi Kapczyńskiemu pragniemy przekazać podziękowania za przekazaną podczas studiów wiedzę i pomoc w realizacji projektu inżynierskiego.

Skrypt w języku PowerShell automatyzujący tworzenie nowych kont użytkowników na podstawie danych importowanych z pliku .csv

```
1  Import-Module ActiveDirectory
2
3   [System.Reflection.Assembly]::LoadWithPartialName("System.
      windows.forms") | Out-Null
4
5  $dialog = New-Object System.Windows.Forms.OpenFileDialog
6  $dialog.InitialDirectory = $StartDir
7  $dialog.Filter = "CSV (*.csv)| *.csv"
8  $dialog.ShowDialog() | Out-Null
9
10 $CSVFile = $dialog.FileName
11
12 if ([System.IO.File]::Exists($CSVFile)) {
13     Write-Host "File valid."
14     $CSV = Import-Csv -LiteralPath "$CSVFile"
15 } else {
16     Write-Host "File not valid."
17     Exit
18 }
19
20 foreach($user in $CSV) {
21
22     $SecurePassword = ConvertTo-SecureString "$($user.'First
      Name'[0])$($user.'Last Name')$($user.'Employee ID')!@#" -
      AsPlainText -Force
23     $Username = "$($user.'First Name Low').$($user.'Last Name
      Low')"
24     $Username = $Username.Replace(" ", "")
25
```

```
26      New-ADUser -Name "$($user.'First Name') $($user.'Last
    Name')" `
27                  -GivenName $user.'First Name' `
28                  -Surname $user.'Last Name' `
29                  -UserPrincipalName $Username `
30                  -SamAccountName $Username `
31                  -EmailAddress $user.'Email Address' `
32                  -Description $user.Description `
33                  -OfficePhone $user.'Office Phone' `
34                  -Path "$($user.'Organizational Unit')" `
35                  -City "Krakow" `
36                  -ChangePasswordAtLogon $false `
37                  -AccountPassword $SecurePassword `
38                  -Enabled $([System.Convert]::ToBoolean($user.
    Enabled))
39
40
41      Write-Host "Created user $Username "
42  }
43
44  Read-Host -Prompt "Complete"
```

Skrypt w języku PowerShell automatyzujący aktywowanie roli kontrolera domeny na serwerze Windows Server 2016 „PDC”

```
#
# Windows PowerShell script for AD DS Deployment
#

Import-Module ADDSDeployment
Install-ADDSForest `
-CreateDnsDelegation:$false `
-DatabasePath "C:\Windows\NTDS" `
-DomainMode "WinThreshold" `
-DomainName "company.sec" `
-DomainNetbiosName "FIRMA" `
-ForestMode "WinThreshold" `
-InstallDns:$true `
-LogPath "C:\Windows\NTDS" `
-NoRebootOnCompletion:$false `
-SysvolPath "C:\Windows\SYSVOL" `
-Force:$true
```

Skrypt w języku PowerShell automatyzujący aktywowanie roli kontrolera domeny na serwerze Windows Server 2016 „DCRO”

```
#
# Windows PowerShell script for AD DS Deployment
#

Import-Module ADDSDeployment
Install-ADDSDomainController `
-AllowPasswordReplicationAccountName @("COMPANY\Allowed RODC Password Replication Group") `
-NoGlobalCatalog:$false `
-Credential (Get-Credential) `
-CriticalReplicationOnly:$false `
-DatabasePath "C:\Windows\NTDS" `
-DenyPasswordReplicationAccountName @("BUILTIN\Administrators", "BUILTIN\Server Operators", "BUILTIN\Backup Operators", "BUILTIN\Account Operators", "COMPANY\Denied RODC Password Replication Group") `
-DomainName "company.sec" `
-InstallDns:$true `
-LogPath "C:\Windows\NTDS" `
-NoRebootOnCompletion:$false `
-ReadOnlyReplica:$true `
-SiteName "Default-First-Site-Name" `
-SysvolPath "C:\Windows\SYSVOL" `
-Force:$true
```

Literatura

[1] Dobrowolski Z., Koncepcja społeczeństwa informacyjnego Daniela Bella, http://www.bbc.uw.edu.pl/Content/20/08.pdf, Instytut Informacji Naukowych i Studiów Bibliologicznych UW.

[2] Słownik Języka Polskiego PWN - „uwierzytelnianie", https://sjp.pwn.pl/slowniki/uwierzytelnianie.html

[3] Microsoft, Co to jest: Uwierzytelnianie wieloskładnikowe?, https://support.microsoft.com/pl-pl/topic/co-to-jest-uwierzytelnianie-wielosk%C5%82adnikowe-e5e39437-121c-be60-d123-eda06bddf661

[4] Massachusetts Institute of Technology, Kerberos: The Network Authentication Protocol, https://web.mit.edu/kerberos/,

[5] Orange Polska, Jak to się zaczęło, czyli historia internetu, https://www.orange.pl/poradnik/twoj-internet/jak-to-sie-zaczelo-czyli-historia-internetu/,

[6] Polsko-Japońska Akademia Technik Komputerowych, System Kerberos, https://edu.pjwstk.edu.pl/wyklady/bsi/scb/main59.html,

[7] Microsoft Security Bulletin MS08-068 - Important, Vulnerability in SMB Could Allow Remote Code Execution (957097), http://web.archive.org/web/20170807221019/https://technet.microsoft.com/en-us/library/security/ms08-068.aspx,

[8] Microsoft Learn, Audit SAM, https://learn.microsoft.com/en-us/windows/security/threat-protection/auditing/audit-sam,

[9] Microsoft Learn, Windows Remote Management, https://learn.microsoft.com/en-us/windows/win32/winrm/portal,

[10] Microsoft Support, Tworzenie i używanie silnych haseł, https://support.microsoft.com/pl-pl/windows/tworzenie-i-uywanie-silnych-haseĆ-c5cebb49-8c53-4f5e-2bc4-fe357ca048eb,

[11] Protokół Kerberos, https://www.ibm.com/docs/pl/aix/7.2?topic=network-kerberos,

[12] Kerberos Pre-Authentication: Why It Should Not Be Disabled, https://social.technet.microsoft.com/wiki/contents/articles/23559.kerberos-pre-authentication-why-it-should-not-be-disabled.aspx,

[13] „Windows Internals, part 1: system architecture, processes, threads, memory management, and more", Pavel Yosifovich, Alex Ionescu, David A. Solomon,

[14] Data Breach Investigations Report, https://www.verizon.com/business/resources/reports/dbir/,

[15] The Art of Deception: Controlling the Human Element of Security, Kevin D. Mitnick, William L. Simon, Steve Wozniak.

[16] Developer Survey Results 2019, https://insights.stackoverflow.com/survey/2019#technology-_-developers-primary-operating-systems,

[17] AdDuplex Report for June 2022, https://reports.adduplex.com/#/r/2022-06,

[18] Repozytorium GitHub narzędzia Impacket, https://github.com/SecureAuthCorp/impacket,

[19] Opis ataku Password Spraying, Mitre Att&ck, https://attack.mitre.org/techniques/T1110/003/,

[20] Kerberoasting, Machine Account (AD Computer Object) Password Updates, https://adsecurity.org/?p=280,

[21] What is lightweight directory access protocol (LDAP) authentication?, RedHat, https://www.redhat.com/en/topics/security/what-is-ldap-authentication,

[22] Distinguished Names, Microsoft, https://learn.microsoft.com/en-us/previous-versions/windows/desktop/ldap/distinguished-names,

[23] Repozytorium zawierające kod źródłowy narzędzia mitm6, https://github.com/dirkjanm/mitm6

[24] Microsoft Learn, Zarządzanie protokołami SSL/TLS i pakietami szyfrów dla *Active Directory*,https://learn.microsoft.com/en-us/windows-server/identity/ad-fs/operations/manage-ssl-protocols-in-ad-fs

[25] Narzędzie przygotowane przez security.org pozwalające sprawdzić jak długo zajmie złamanie podanego hasła, https://www.security.org/how-secure-is-my-password/,

[26] Microsoft Digital Defense Report 2022, https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE5bUvv,

[27] Microsoft Digital Defense Report 2021, https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RWMFIi.

Lista załączników

- Plik *Dokument_hipertekstowy_wdrozenie_infrastruktury.zip* zawiera kod źródłowy dokumentu hipertekstowego, który pozwala na przeglądanie zrzutów ekranowych wykonanych podczas wdrażania i konfigurowania infrastruktury.
- Plik *kerberos_preauth.pcapng* zawierający przechwycony przy pomocy programu Wireshark ruch sieciowy zaobserwowany podczas uwierzytelniania się użytkownika domeny *Active Directory* w celu połączenia się z zasobem sieciowym.
- Plik *NewUsers.csv* zawiera dane użytkowników dodanych przy pomocy skryptu.