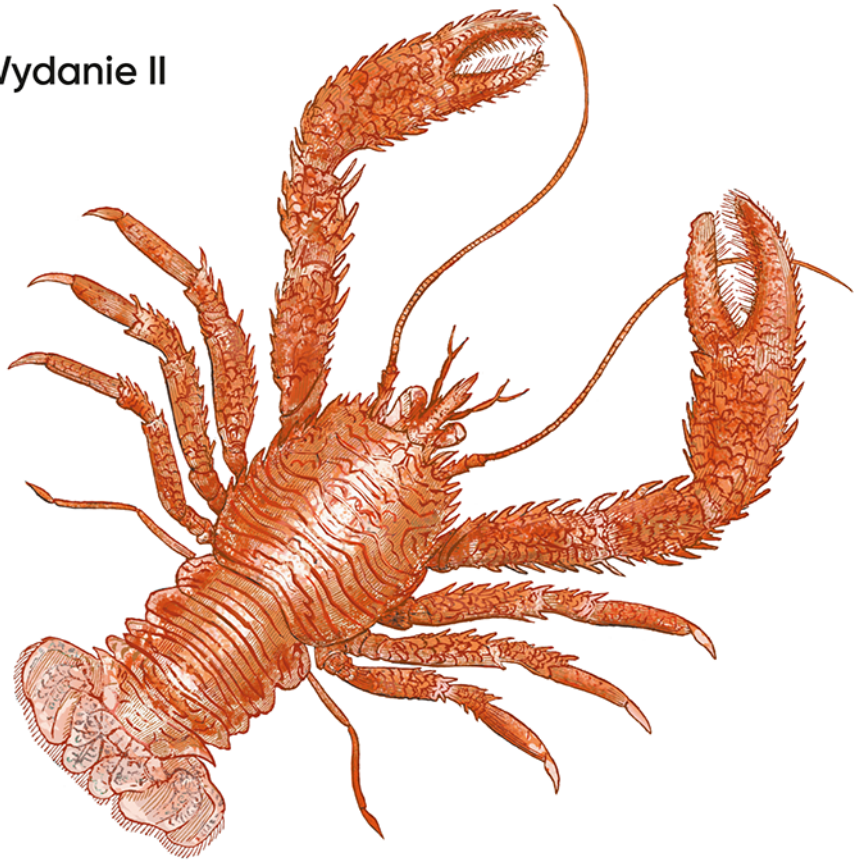
O'REILLY®

Helion

Sieci Zero Trust

Budowanie bezpiecznych systemów w niezaufanym środowisku

Wydanie II



Razi Rais, Christina Morillo,
Evan Gilman, Doug Barth

Tytuł oryginału: Zero Trust Networks: Building Secure Systems in Untrusted Network, 2nd Edition

Tłumaczenie: Andrzej Watrak

ISBN: 978-83-289-1481-0

Helion S.A.
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *https://helion.pl* (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
https://helion.pl/user/opinie/sizet2
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

Spis treści

ROZDZIAŁ 6.

Zaufanie do tożsamości

Istnieje pokusa, aby połączyć zaufanie do użytkownika z zaufaniem do urządzenia. W organizacjach, które dbają o bezpieczeństwo, stosuje się certyfikaty X.509 urządzeń, będące solidniejszymi poświadczeniami niż hasła. Można powiedzieć, że certyfikat urządzenia jednoznacznie identyfikuje użytkownika, ale czy na pewno? Jak uzyskać pewność, że przy klawiaturze siedzi uprawniony użytkownik? A może zostawił swoje urządzenie odblokowane i bez nadzoru?

Łączenie tożsamości użytkownika z tożsamością urządzenia może przysparzać problemów, jeżeli użytkownik korzysta z kilku urządzeń, co dzisiaj staje się normą. Poświadczenia, które trzeba kopiować między urządzeniami, są narażone na ujawnienie. Ponadto urządzenia, w zależności od możliwości, wymagają różnych poświadczeń. W sieciach z kioskami problem jest jeszcze poważniejszy.

W sieciach Zero Trust tożsamość i zaufanie do użytkowników są oddzielone od urządzeń. Użytkowników i urządzenia można identyfikować z zastosowaniem tej samej technologii, ale należy wyraźnie podkreślić, że z użyciem różnych poświadczeń.

W tym rozdziale wyjaśnimy, na czym polega identyfikacja użytkowników i przechowywanie ich tożsamości oraz kiedy i jak ich uwierzytelniać. Zaufanie użytkowników jest często większe, gdy zaangażowanych jest kilka osób. Dlatego pokażemy, jak zbudować zaufanie grupowe i kulturę bezpieczeństwa.

Urząd tożsamości

Każdy użytkownik ma tożsamość, która identyfikuje go w określony sposób w społeczności. W sieci tożsamość użytkownika oznacza sposób rozpoznawania go w systemie. Zważywszy, jak dużo ludzi jest na świecie, identyfikacja użytkownika jest zaskakująco trudnym problemem.

Przeanalizujmy dwa typy tożsamości:

- nieformalną,
- autorytatywną.

Tożsamość nieformalna to taka, którą grupy określają samodzielnie. Rozważmy sytuację, w której spotykamy jakiegoś człowieka. Na podstawie tego, jak wygląda i się zachowuje, określamy jego tożsamość. Jeżeli spotkamy go później, oceniając jego fizyczne cechy, będziemy mogli racjonalnie uznać, że to ten sam człowiek, którego widzieliśmy wcześniej. Możemy go nawet zidentyfikować na odległość, na przykład na podstawie głosu.

Tożsamość nieformalna jest wykorzystywana w systemach komputerowych. W społecznościach internetowych powszechnie stosuje się konta pseudonimowe, czyli niepowiązane z prawdziwymi imionami i nazwiskami. Społeczność nie zna prawdziwej tożsamości osoby, ale na podstawie jej częstych interakcji buduje jej tożsamość nieformalną.

Tożsamość nieformalna sprawdza się w małych grupach, w których zaufanie między osobami jest duże, a ryzyko małe. Ujawnia jednak swoje słabości w poważniejszych sytuacjach, ponieważ:

- jest fikcyjna i można ją sfabrykować,
- można przypisać sobie tożsamość innej osoby,
- można budować wiele tożsamości,
- kilka osób może współdzielić jedną tożsamość.

Jeżeli wymagana jest pewniejsza forma tożsamości, urząd musi utworzyć autorytatywne poświadczenia tożsamości dla poszczególnych osób. W praktyce zajmują się tym urzędy państwowe. Obywatele dostają dowody tożsamości (prawa jazdy, paszporty), które okazują innym osobom. W sytuacjach niewielkiego ryzyka tego rodzaju dokumenty są wystarczającymi dowodami, ale gdy ryzyko jest większe, lepszymi potwierdzeniami są poświadczenia zapisane w rządowej bazie danych.

Systemy komputerowe również potrzebują centralnego urzędu tożsamości użytkowników. Użytkownicy otrzymują poświadczenia (o różnej sile), które identyfikują ich w systemie. W zależności od ryzyka mogą być sprawdzane poświadczenia zapisane w centralnej bazie danych. W dalszej części rozdziału dowiesz się, jak funkcjonują tego rodzaju systemy.

Poświadczenia mogą zostać utracone lub skradzione, dlatego ważne jest, aby urząd stosował mechanizmy dające obywatelowi kontrolę nad swoją tożsamością. Aby mógł on ponownie uzyskać dowód tożsamości, musi okazać urzędowi inne informacje umożliwiające identyfikację, na przykład akt urodzenia lub odcisk palca. Podobnych mechanizmów potrzebują również systemy komputerowe w razie zgubienia lub kradzieży poświadczeń. Zazwyczaj wymagana jest inna forma weryfikacji, taka jak użycie kodu odzyskiwania lub alternatywne uwierzytelnienie. Wybór mechanizmu potwierdzania tożsamości ma wpływ na bezpieczeństwo, o czym napiszemy w dalszej części rozdziału.

Budowanie tożsamości w prywatnym systemie

Wiesz już, jak przechowywać i potwierdzać tożsamość, ale jak ją utworzyć na samym początku? Człowiek używający systemu komputerowego musi w jakiś sposób cyfrowo potwierdzać swoją tożsamość. Dlatego potrzebne jest jak najściślejsze powiązanie cyfrowej reprezentacji tożsamości z człowiekiem.

Utworzenie cyfrowej tożsamości i jej powiązanie z człowiekiem to bardzo delikatne operacje. Metoda uwierzytelniania człowieka w systemie komputerowym musi być bardzo skuteczna, aby na przykład haker nie mógł się podszyć pod pracownika. Podobny mechanizm jest potrzebny do odzyskiwania dostępu do konta, gdy użytkownik nie jest w stanie wprowadzić poprawnych poświadczeń.

Atakowanie systemów do odzyskiwania tożsamości

Użytkownik może swoje poświadczenia zgubić (np. kartę inteligentną) lub zapomnieć (hasło). Aby je odzyskać, na przykład zresetować hasło, musi się uwierzytelnić za pomocą alternatywnego, niekiedy niestandardowego systemu. Takie systemy są obiektami częstych, nierzadko skutecznych ataków. Na przykład w 2012 roku haker włamał się na konto serwisu Amazon popularnego dziennikarza, uzyskał cztery ostanie cyfry jego karty kredytowej i wykorzystał je do potwierdzania nieswojej tożsamości podczas kontaktu telefonicznego z działem pomocy technicznej Apple. Dlatego należy dokładnie sprawdzać proces resetowania hasła. Tajne informacje często są mniej tajne, niż można by sądzić.

Zważywszy na newralgiczność tej operacji, ważne jest, aby zasady zarządzania tożsamością były dobrze przemyślane i skuteczne. Najważniejsze jest bezpieczeństwo ludzi. Dobra informacja jest taka, że wiadomo, jak to robić dobrze.

Dowód tożsamości wydany przez urząd

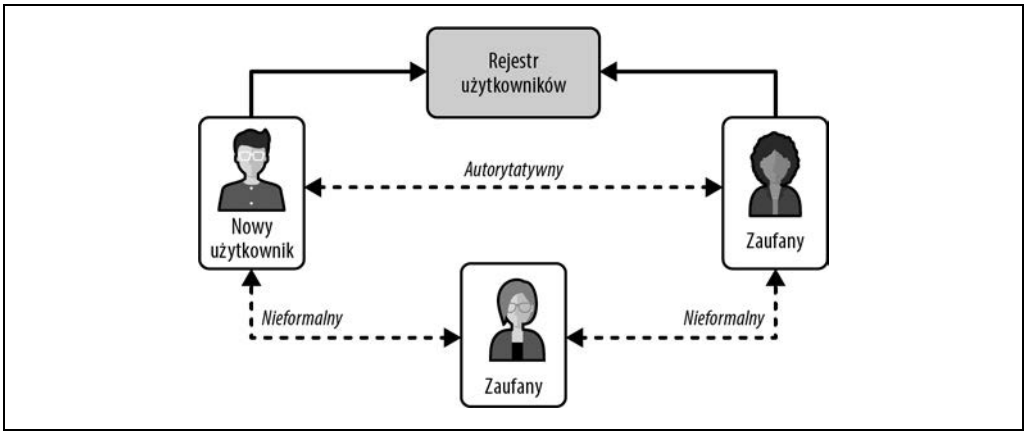
Z pewnością nie jest zaskoczeniem, że podstawową formą potwierdzenia tożsamości człowieka jest dowód tożsamości wydany przez państwowy urząd. W końcu uwierzytelnianie człowieka jest dokładnie tym, do czego je zaprojektowano!

W niektórych implementacjach, aby potencjalni oszuści mieli utrudnione zadanie, pożądane jest nawet stosowanie kilku form potwierdzania tożsamości. Oczywiście, aby zapobiec obchodzeniu kontroli, personel trzeba przeszkolić pod kątem potwierdzania dokumentów tożsamości.

Nic nie zastąpi człowieka

Cyfrowe systemy uwierzytelniania, pomimo wszelkich starań ich twórców, nie są tak skuteczne jak ich naturalne pierwowzory, w których zaangażowani są ludzie. Najlepiej jest, gdy nową cyfrową tożsamość człowieka tworzy inny człowiek. Zdecydowanie odradza się wysyłanie wiadomości e-mail lub stosowania innych form w ciemno. Często jednak zdarza się, że urządzenie przed pierwszym użyciem jest skonfigurowane tak, że ufa użytkownikowi. Podejście to ma ten mankament, że jest podatne na nadużycia.

Często stworzenie cyfrowej tożsamości jest poprzedzone długotrwałym procesem, w którym zaangażowany jest człowiek. Może to być na przykład seria rozmów kwalifikacyjnych lub zawarcie umowy biznesowej. Dana osoba miała wcześniej kontakt z innymi, zaufanymi osobami, które poznały niektóre jej cechy. Ta wiedza jest wykorzystywana do dalszego uwierzytelniania z udziałem człowieka. Ilustruje to rysunek 6.1.



Rysunek 6.1. Zaufany administrator podczas wprowadzania nowego użytkownika do bazy danych polega na zaufanym pracowniku i jego dowodzie tożsamości

Na przykład menedżer ds. rekrutacji może w celu uwierzytelnienia nowego pracownika skierować go do działu pomocy technicznej, który zna już tę osobę i może potwierdzić jej tożsamość. Jest to dobra metoda uwierzytelniania, ale jak zawsze w sieci Zero Trust, nie może być jedyną.

Oczekiwania i fakty

Zwykle przed stworzeniem cyfrowej tożsamości człowieka dostępnych jest wiele informacji o nim. Ważne jest, aby wykorzystać ich jak najwięcej i sprawdzić, czy fakty są zgodne z oczekiwaniami. Oczekiwania te są typowe dla sieci Zero Trust, gromadzi je i weryfikuje człowiek. Mogą obejmować język, którym się posługuje użytkownik, adres zamieszkania zapisany w dowodzie osobistym i wiele innych. W rzetelnej firmie w celu określenia rzeczywistych oczekiwań wykorzystuje się informacje zdobyte podczas sprawdzania przeszłości użytkownika. Ludzie codziennie używają takich metod do wzajemnego uwierzytelniania się, zarówno nieformalnego, jak i oficjalnego, więc metody te są dojrzałe i niezawodne.

Przechowywanie tożsamości

Ponieważ tożsamość w świecie realnym trzeba związać ze światem wirtualnym, należy ją przekształcić w bity. Bity te są bardzo poufne i zazwyczaj trzeba je przechowywać permanentnie. Dlatego pokażemy teraz, jak je bezpiecznie gromadzić.

Rejestry użytkowników

Aby system ufał użytkownikom, potrzebuje ich centralnego rejestru. Na zapisie w rejestrze opiera się całe przyszłe uwierzytelnienie. Przechowywanie wszystkich bardzo poufnych danych jest nie lada wyzwaniem, którego nie da się uniknąć.

Do podejmowania decyzji uwierzytelniających w sieci Zero Trust wykorzystuje się różnorodne dane o użytkownikach. Rejestry zawierają zarówno podstawowe informacje, takie jak imiona i nazwiska, numery telefonów czy funkcje w organizacji, jak też rozszerzone, na przykład lokalizacje i klucze publiczne certyfikatów X.509.

Ponieważ dane użytkowników są niezwykle newralgiczne, nie należy ich wszystkich przechowywać w jednej bazie. Informacji tych ogólnie nie uznaje się za tajne, ale takimi się stają, gdy są wykorzystywane przy podejmowaniu decyzji autoryzacyjnych. Ponadto posiadanie szerokiej wiedzy o wszystkich użytkownikach systemu może zagrażać ich prywatności. Na przykład informacje o lokalizacji użytkowników można wykorzystać do ich szpiegowania. Dane te mogą być również zagrożeniem dla bezpieczeństwa, ponieważ można ich użyć do przeprowadzenia ataku na inny system. Sposobem na dodatkowe potwierdzanie tożsamości użytkowników może być zastosowanie systemu wymagającego podania informacji opartych na faktach.

Zamiast przechowywać wszystkie dane użytkowników w jednej bazie, warto je podzielić na kilka osobnych baz, najlepiej udostępnianych wyłącznie za pośrednictwem ograniczonego interfejsu API. Dzięki temu źródłowe informacje nie będą ujawniane, a aplikacje mające do nich dostęp będą mogły formułować pewne założenia dotyczące użytkowników. Na przykład można za pomocą interfejsu API udostępniać następujące dane:

- czy dany użytkownik znajduje się obecnie w miejscu o podanych współrzędnych lub w jego pobliżu,
- jak często dany użytkownik zmienia lokalizację.

Utrzymywanie rejestru

Dokładność rejestru użytkowników ma kluczowe znaczenie dla bezpieczeństwa sieci Zero Trust. Przez cały czas istnienia systemu pojawiają się nowi użytkownicy, a inni odchodzą. Dlatego aby zapewnić dokładność systemu, trzeba ustalić ścisłe procedury wprowadzania i usuwania danych. W miarę możliwości należy zintegrować system identyfikacji oparty na protokole LDAP (ang. *Lightweight Directory Access Protocol*, prosty protokół udostępniania usług katalogowych) lub lokalnych kontach z innymi systemami organizacji. Weźmy na przykład system kadrowy, w którym rejestruje się pracowników rozpoczynających i kończących pracę w organizacji. Te dwa źródła danych powinny być spójne. Jeżeli nie będzie systemu, który je zintegruje lub sprawdzi ich zawartość, szybko pojawią się rozbieżności. Stworzenie zautomatyzowanych procesów łączenia systemów to praca, która szybko się zwróci.

Gdy istnieją dwa osobne systemy zarządzania tożsamością, pojawia się ważna kwestia: który z nich jest autorytatywny. Oczywiście, jeden z nich musi rejestrować tożsamości, ale w wyborze należy się kierować potrzebami organizacji. Nie ma większego znaczenia, który system zostanie wybrany, ważne jest tylko, aby był autorytatywny i by na nim opierały się wszystkie inne systemy wykorzystujące dane tożsamościowe.

Warto ograniczać gromadzone dane

System rejestrowania tożsamości nie musi zawierać wszystkich informacji o użytkownikach. Wcześniej wspomnieliśmy, że dobrym rozwiązaniem jest przemyślana segmentacja danych. System ewidencji powinien zawierać wyłącznie informacje niezbędne do identyfikowania osób. Mogą to być podstawowe dane, takie jak imiona, nazwiska i kilka innych, które użytkownicy mogliby wykorzystywać do odzyskiwania utraconej tożsamości. Dodatkowe informacje można gromadzić w systemach pochodnych, opartych na autorytatywnych identyfikatorach.

Kiedy weryfikować tożsamość?

Uwierzytelnianie jest obowiązkowe w sieci Zero Trust, ale dzięki zastosowaniu pewnego pomysłowego sposobu można znacznie zwiększyć bezpieczeństwo i jednocześnie zminimalizować niedogodności dla użytkowników. Podejście „To ma nie być proste, tylko bezpieczne" jest wprawdzie kuszące, a nawet logiczne, jednak przy projektowaniu sieci Zero Trust jednym z najważniejszych czynników jest wygoda użytkownika. Uciążliwe mechanizmy zabezpieczeń są często osłabiane i neutralizowane przez samych użytkowników. Złe doświadczenia będą ich zniechęcać do korzystania, prowokować do stosowania skrótów i obchodzenia wymaganych przepisów.

Uwierzytelnianie w celu uzyskania zaufania

Operacja uwierzytelnienia użytkownika polega na tym, że system weryfikuje, czy użytkownik jest rzeczywiście tym, za kogo się podaje. Jak się dowiesz w następnym podrozdziale, metody uwierzytelnienia różnią się wiarygodnością, a niektóre są najskuteczniejsze wtedy, gdy stosuje się je razem z innymi. Ponieważ żadna z nich nie jest absolutnie pewna, wynikowi operacji trzeba przypisywać pewien poziom zaufania.

Na przykład do zalogowania się do serwisu muzycznego wystarczy podać hasło, natomiast konto inwestycyjne zazwyczaj wymaga wpisania hasła i dodatkowego kodu. Wynika to z faktu, że inwestowanie jest newralgiczną operacją. System musi ufać, że użytkownik jest uprawniony. Serwis muzyczny nie jest tak newralgiczną usługą i nie wymaga wprowadzania dodatkowego kodu, ponieważ byłoby to uciążliwe.

Aby użytkownik mógł podnieść poziom swojego zaufania, musi przejść dodatkowe procedury uwierzytelnienia. Dotyczy to sytuacji, gdy zachodzi taka potrzeba. Użytkownik, którego ocena zaufania spadnie poniżej minimalnej dla konkretnego żądania, musi okazać dodatkowy dowód, który po zaakceptowaniu podniesie zaufanie do wymaganego poziomu.

Powyższa koncepcja nie jest niczym nowym i jest dzisiaj powszechnie stosowana. Doskonałym przykładem jest wymóg wprowadzenia hasła przed wykonaniem newralgicznej operacji. Należy jednak pamiętać, że poziom zaufania, jaki można osiągnąć dzięki samemu uwierzytelnieniu, trzeba ograniczyć, aby nie narażać się na konsekwencje słabego zabezpieczenia urządzenia i inne niepożądane skutki.

Zaufanie jako czynnik uwierzytelniający

Ponieważ uwierzytelnianie opiera się na zaufaniu, a głównym celem jest uwolnienie użytkowników od niepotrzebnych utrudnień, sensownym rozwiązaniem jest uzależnienie procesu uwierzytelnienia od oceny zaufania. Oznacza to, że nie należy prosić użytkownika o dodatkowe uwierzytelnienie, jeżeli jego ocena zaufania jest wystarczająco wysoka. I odwrotnie: jeżeli ocena jest zbyt niska, użytkownik musi się dodatkowo uwierzytelnić. Nie trzeba więc określać, jakie konkretnie operacje wymagają dodatkowego uwierzytelnienia. Zamiast tego należy uzależnić proces uwierzytelnienia i wymagania od oceny zaufania. System na podstawie informacji o newralgiczności operacji i zaufaniu poszczególnych metod powinien dobierać ich kombinacje tak, aby osiągnąć cel przy minimalnej inwazyjności.

Opisane podejście zasadniczo różni się od tradycyjnego procesu uwierzytelniania, w którym określa się najbardziej newralgiczne obszary i operacje wymagające najściślejszego uwierzytelnienia bez uwzględniania wcześniejszych metod i poziomu zaufania. Pod pewnymi względami tradycyjne podejście można porównać do ochrony obwodowej, kiedy to newralgiczne operacje muszą przejść określone testy, po których nie ma już żadnych zabezpieczeń. Natomiast decyzja na podstawie oceny zaufania pozwala uniknąć nadmiernych wymagań dotyczących uwierzytelniania. Uwierzytelnianie i autoryzację stosuje się elastycznie wtedy, gdy jest to konieczne.

Korzystanie z kilku kanałów

Skutecznym podejściem podczas uwierzytelniania i autoryzacji żądania jest docieranie do źródła wieloma kanałami. Dodatkowym czynnikiem może być kod jednorazowy, szczególnie gdy generujący go system znajduje się na osobnym urządzeniu. Podobną możliwość zapewniają powiadomienia push i aktywne połączenia z urządzeniami mobilnymi. Istnieje wiele zastosowań i form tego rozwiązania.

W zależności od przypadku wielokanałowość może być integralną cechą mechanizmu uwierzytelniania cyfrowego. Ewentualnie kanały można wykorzystywać wyłącznie do autoryzacji, gdy źródło żądania musi potwierdzać ryzykowne operacje. Oba podejścia są równie skuteczne, jednak przy podejmowaniu decyzji, kiedy i gdzie je stosować, należy (jak zawsze) mieć na uwadze wygodę użytkownika.

Bezpieczeństwo kanału

Kanały komunikacji różnią się poziomami uwierzytelnienia i zaufania. Podczas korzystania z wielu kanałów ważna jest wiedza o zaufaniu do każdego z nich. Od tego zależy, z których kanałów trzeba korzystać i kiedy. Na przykład bezpieczeństwo urządzeń z kodami rotacyjnymi jest takie samo jak bezpieczeństwo systemu użytego do dystrybucji tych kodów lub systemu do identyfikacji wymaganej w celu uzyskania urządzenia od administratora. Podobnie monit z korporacyjnego czatu jest równie wiarygodny jak poświadczenia wymagane do zalogowania się do niego. Należy pamiętać, że drugi kanał musi być inny niż użyty do pierwszego uwierzytelnienia lub autoryzacji.

Wielokanałowość jest skuteczna dlatego, że o wiele trudniej jest naruszyć bezpieczeństwo wielu kanałów niż jednego. Więcej na ten temat dowiesz się w następnym podrozdziale.

Buforowanie tożsamości i zaufania

Buforowanie sesji to dojrzała, dobrze udokumentowana technologia, więc nie poświęcimy jej wiele miejsca. Podkreślimy natomiast kilka zasad projektowania ważnych dla bezpiecznego działania w sieci Zero Trust. Kluczowe znaczenie ma częsta weryfikacja autoryzacji klienta. Jest to jeden z niewielu mechanizmów na płaszczyźnie sterowania, które mają wpływ na aplikacje na płaszczyźnie danych, gdy zmienia się zaufanie. Im częściej jest stosowany, tym lepiej. W niektórych wdrożeniach płaszczyzna sterowania autoryzuje wszystkie żądania. Jest to idealne rozwiązanie, ale nie zawsze możliwe.

Wiele aplikacji weryfikuje tokeny SSO (ang. *Single Sign-On*, jednokrotne logowanie) tylko na początku sesji, a następnie tworzy własne. Ten tryb eliminuje kontrolę sesji na płaszczyźnie sterowania i jest niepożądany. Autoryzując żądania za pomocą tokenów płaszczyzny sterowania, a nie tokenów aplikacji, można łatwo unieważniać żądania, gdy poziom zaufania waha się lub spada.

Jak weryfikować tożsamość?

Teraz gdy już wiesz, kiedy należy uwierzytelniać użytkownika, przyjrzyjmy się, jak to robić. Powszechnie stosuje się cztery metody identyfikacji:

Co użytkownik wie

: Wiedza, którą ma sam użytkownik (hasło, numer PIN).

Co użytkownik ma

: Fizyczne poświadczenia, które użytkownik może podać (sprzętowy token, brelok, karta inteligentna, klucz USB, identyfikator dostępu).

Czym użytkownik jest

: Cecha wrodzona/biometryczna, która jednoznacznie identyfikuje użytkownika (np. linie papilarne, tęczówka, głos, twarz).

Wzorce zachowań

: Analiza charakterystycznych wzorców zachowań (np. sposób trzymania urządzenia lub pisania) z wykorzystaniem uczenia maszynowego.

Użytkownika można identyfikować z użyciem jednej lub kilku powyższych metod. Wybór zależy od wymaganego poziomu zaufania. W przypadku operacji wysokiego ryzyka, które wymagają wielu czynników uwierzytelniania, należy wybrać metody należące do różnych, wymienionych wyżej grup, ponieważ ataki głównie koncentrują się na określonej jednej grupie. Na przykład każdy, kto ukradnie token (*co użytkownik ma*), może go wykorzystać. Gdyby były wykorzystywane dwa tokeny, z dużym prawdopodobieństwem byłyby trzymane razem i skradzione oba.

Wybór czynników, które będą stosowane razem, zależy od urządzenia użytkownika. Jeżeli jest to komputer stacjonarny, wtedy skuteczną, zalecaną kombinacją jest hasło (*co użytkownik wie*) i token sprzętowy (*co użytkownik ma*). Jednak w przypadku urządzenia mobilnego lepszy jest odcisk palca (*czym użytkownik jest*) i hasło (*co użytkownik wie*).

Fizyczne bezpieczeństwo jest niezbędne, aby ufać użytkownikom

W tym podrozdziale skupiliśmy się na technikach weryfikacji tożsamości użytkownika. Należy jednak pamiętać, że użytkownik może być fizycznie lub groźbą zmuszony do ujawnienia tych technik lub poświadczeń, jak również do udostępnienia zaufanego konta. Analiza zachowań i historycznych trendów ogranicza takie ataki, ale nie udaremnia ich całkowicie.

Co użytkownik wie: hasła

Hasła są obecnie najpowszechniejszym czynnikiem uwierzytelniania w systemach komputerowych. Jego przydatność często jednak się kwestionuje z powodu skłonności użytkowników do ustawiania złych haseł, ale ma ważną zaletę: właściwie zastosowany daje gwarancję, że użytkownik zachowuje się rozsądnie.

Dobre hasło ma następujące cechy:

Jest długie

Najnowsza norma instytutu NIST określa, że hasło powinno składać się z co najmniej 8 znaków. Jednak ci, co dbają o swoje bezpieczeństwo, nierzadko mają hasła o długości 20 znaków. Często zaleca się stosowanie frazy ułatwiającej zapamiętanie długiego hasła.

Trudno je odgadnąć

Użytkownicy zazwyczaj nie wykorzystują w pełni swoich możliwości wymyślania naprawdę złożonych haseł, więc dobrym rozwiązaniem jest tworzenie ich z użyciem generatora liczb losowych. Nie jest to jednak wygodny sposób, jeżeli hasła są trudne do zapamiętania.

Nie jest stosowane gdzie indziej

Hasło jest sprawdzane z użyciem danych zawartych w usłudze. Jeżeli stosuje się je w kilku miejscach, jego poufność jest taka jak poufność najsłabszego użytego magazynu.

Tworzenie długich, trudnych do odgadnięcia haseł dla poszczególnych usług i aplikacji może być dla korzystającego z nich użytkownika nie lada wyzwaniem. Z tego powodu może on korzystać z menedżera haseł, który pozwala tworzyć hasła znacznie trudniejsze do odgadnięcia bez narażania danych na zagrożenia.

Podczas tworzenia usługi weryfikującej hasła należy przestrzegać dobrych praktyk. Haseł nie można przechowywać ani rejestrować w oryginalnej formie, tylko w postaci kryptograficznego skrótu. Koszt ataku metodą brutalnej siły, mierzony ilością potrzebnego czasu i pamięci, zależy od siły algorytmu skracającego. Instytut NIST publikuje od czasu do czasu zalecenia dotyczące haseł (*https://oreil.ly/clc4T*). Zalecenia te zmieniają się, ponieważ komputery są coraz wydajniejsze. Dlatego przy wyborze algorytmu należy zapoznać się z dobrymi praktykami branżowymi.

Co użytkownik ma: jednorazowe hasło czasowe

Jednorazowe hasło czasowe to metoda uwierzytelniania użytkownika z wykorzystaniem nieustannie zmienianego kodu. Dokument RFC 6238 definiuje normę stosowaną w urządzeniach i aplikacjach (*https://oreil.ly/3wwSQ*). Do generowania kodu często stosuje się aplikacje mobilne. Jest to dobre rozwiązanie, ponieważ użytkownik zwykle ma przy sobie telefon komórkowy.

Niezależnie od tego, czy użytkownik korzysta z aplikacji czy urządzenia, musi podać jednorazowe hasło czasowe właściwe dla danej usługi. Na podstawie poufnego hasła i bieżącego czasu jest generowany kryptograficzny skrót, który po przycięciu musi być wprowadzony przez użytkownika. Jeżeli zegary na urządzeniu i serwerze są mniej więcej zsynchronizowane, zgodny kod potwierdza, że użytkownik ma właściwy klucz. Krytyczne znaczenie ma sposób przechowywania współdzielonego klucza zarówno na urządzeniu, jak i serwerze uwierzytelniającym. Utrata kontroli nad poufnym kluczem trwale dyskwalifikuje ten mechanizm uwierzytelniania. Dokument RFC zaleca szyfrowanie klucza za pomocą urządzenia, na przykład modułu TPM, i ograniczanie dostępu do zaszyfrowanych danych.

Klucz umieszczony na urządzeniu mobilnym jest narażony na większe niebezpieczeństwo niż zapisany na serwerze. W przypadku nawiązania połączenia ze szkodliwym punktem końcowym haker mógłby przechwycić klucz. Alternatywnym do jednorazowego hasła czasowego rozwiązaniem, które chroni przed tego rodzaju atakiem, jest wysyłanie szyfrowanym kanałem na telefon komórkowy użytkownika losowego kodu. Użytkownik przez wprowadzenie tego kodu na innym urządzeniu potwierdza, że korzysta z należącego do niego telefonu.

SMS nie jest bezpiecznym kanałem komunikacji

Losowo wygenerowany kod uwierzytelniający musi być wysłany do docelowego urządzenia w niezawodny, uniemożliwiający ujawnienie sposób. Niektóre systemy wysyłają takie kody w wiadomościach SMS, te jednak nie gwarantują należytej ochrony. Dlatego nie zaleca się używania wiadomości SMS do tego celu.

Co użytkownik ma: certyfikaty

Inną metodą uwierzytelniania użytkowników jest generowanie indywidualnych certyfikatów X.509. Certyfikat generuje się z użyciem wiarygodnego klucza prywatnego, a następnie podpisuje kluczem prywatnym organizacji. Certyfikatu nie sposób zmodyfikować bez unieważnienia podpisu organizacji, więc można go używać jako poświadczenia w każdej usłudze ufającej podpisowi.

Ponieważ certyfikat X.509 jest przeznaczony dla komputera, a nie dla człowieka, można w nim na potrzeby uwierzytelniania umieszczać różne szczegółowe informacje. Mogą to być na przykład dane użytkownika. Takim danym można ufać, ponieważ podpisuje je zaufana organizacja. Dzięki temu w mniej dojrzałych sieciach eliminuje się potrzebę tworzenia zaufanego rejestru użytkowników.

Identyfikacja użytkowników z zastosowaniem certyfikatów w dużej mierze zależy od tego, czy są bezpiecznie przechowywane. Aby zapobiec kradzieży klucza prywatnego, zdecydowanie zaleca się generowanie go i przechowywanie na osobnym urządzeniu. Więcej na ten temat dowiesz się w następnym podrozdziale.

Co użytkownik ma: token bezpieczeństwa

Token bezpieczeństwa to urządzenie sprzętowe, które służy głównie do uwierzytelniania użytkowników, ale nie tylko. Nie jest to magazyn do przechowywania poświadczeń utworzonych w innym systemie. Jest to sprzęt, który generuje klucz prywatny. Ta informacja nie wychodzi poza token. Urządzenie użytkownika wykorzystuje interfejs API do wykonywania operacji kryptograficznych w imieniu użytkownika i potwierdzania, że jest on w posiadaniu tokenu.

Wraz z rozwojem technologii zabezpieczeń organizacje coraz częściej korzystają ze sprzętowych mechanizmów weryfikowania tożsamości użytkowników. Urządzenia takie jak karty inteligentne czy klucze YubiKey można używać do niezawodnego potwierdzania tożsamości użytkownika. Przez powiązanie tożsamości ze sprzętem znacznie ogranicza się ryzyko przechwycenia lub wykradzenia poświadczeń użytkownika bez jego wiedzy. Tak może się stać jedynie w razie kradzieży.

Sprzęt jest obecnie najbezpieczniejszym miejscem przechowywania klucza prywatnego. Klucz jest fundamentem dla wielu różnych metod uwierzytelniania. Tradycyjnie wykorzystuje się go w połączeniu z certyfikatem X.509, ale od pewnego czasu popularność zyskuje standard U2F (ang. *Universal 2nd Factor*, uniwersalny drugi czynnik). Jest to odpowiednik pełnowymiarowej infrastruktury PKI, wykorzystywany w usługach internetowych, oferujący prosty protokół typu wyzwanie-odpowiedź. Każda metoda uwierzytelniania oparta na kryptografii asymetrycznej wykorzystuje token bezpieczeństwa.

Sprzętowy token skutecznie chroni poświadczenia przed kradzieżą, ale nie daje pewności, że nikt go nie ukradnie ani nie użyje w złej wierze. Dlatego ważna jest świadomość, że token jest świetnym narzędziem do budowania bezpiecznego systemu, ale nie zastępuje całkowicie użytkownika przy potwierdzaniu tożsamości. Aby mieć jak największą pewność, że dany użytkownik jest tym, za kogo się podaje, zdecydowanie zaleca się stosowanie klucza bezpieczeństwa z dodatkowymi czynnikami uwierzytelniającymi, na przykład hasłem lub czujnikiem biometrycznym.

Czym użytkownik jest: biometria

Potwierdzenie tożsamości użytkownika na podstawie jego cech fizycznych nazywa się **biometrią**. W miarę pojawiania się zaawansowanych czujników w urządzeniach, z których korzystamy na co dzień, biometria zyskuje coraz większą popularność. Stosowana rozważnie zapewnia większą wygodę i bezpieczeństwo. Biometria opiera się na analizie następujących cech:

- linii papilarnych palców,
- linii papilarnych dłoni,
- tęczówki,
- głosu,
- twarzy.

Przetwarzanie danych biometrycznych wydaje się idealną metodą uwierzytelniania. W końcu celem jest potwierdzenie, że użytkownik jest tym, za kogo się podaje. Czy istnieje coś lepszego niż pomiar jego fizycznych cech? Biometria jest użyteczną metodą w zapewnianiu bezpieczeństwa systemu, ale ma pewne wady, o których należy pamiętać.

Uwierzytelnianie użytkownika na podstawie danych biometrycznych polega na dokładnym analizowaniu jego cech fizycznych. Jeżeli hakerowi uda się oszukać skaner, uzyska dostęp do systemu. Odciski palców, będące powszechnymi czynnikami biometrycznymi, zostawiamy na wszystkim, czego dotykamy. Jak się okazuje, haker może zrobić zdjęcie niewidocznego śladu linii papilarnych, wykonać ich wydruk 3D i oszukać skaner. Ponadto danych biometrycznych nie można zmieniać, ponieważ są one cechą fizyczną człowieka. Mogą również utrudniać dostęp komuś, kto nie ma linii papilarnych (adermatoglifia) lub stracił palce w wypadku.

Wreszcie z biometrią wiąże się zaskakująco dużo problemów prawnych w porównaniu z innymi mechanizmami uwierzytelniania. Na przykład w Stanach Zjednoczonych sąd może nakazać obywatelowi uwierzytelnienie się za pomocą odcisku palca na urządzeniu, ale nie może nakazać ujawnienia hasła ze względu na zawarte w konstytucji prawo do nieobciążania się.

Wzorce zachowań

Uwierzytelnianie behawioralne to metoda weryfikacji tożsamości wykorzystująca uczenie maszynowe do rozpoznawania unikalnych zachowań człowieka, takich jak sposób pisania, trzymania urządzenia itp. Jest skuteczna, ponieważ szybko dostosowuje się do zmian w zachowaniu danej osoby, co utrudnia jej naśladowanie przez innych. Aby zapewnić dodatkowe zabezpieczenie, uwierzytelnianie behawioralne często stosuje się w połączeniu z innymi formami, takimi jak hasła lub numery PIN. Jak wiadomo, żadna metoda stosowana samodzielnie nie jest w stu procentach pewna.

Uwierzytelnianie behawioralne w porównaniu z innymi formami uwierzytelniania, na przykład hasłami lub numerami PIN, ma kilka wad:

- Jest bardziej inwazyjne, ponieważ wymaga od użytkowników podania dodatkowych danych o sobie, takich jak odciski palców czy skan tęczówki.
- Jest bardziej zawodne, ponieważ wzorce zachowań zmieniają się z upływem czasu, co utrudnia jednoznaczną weryfikację tożsamości.

Jednak pomimo tych mankamentów uwierzytelnianie behawioralne jest kolejną przydatną metodą ochrony poufnych informacji i tożsamości.

Uwierzytelnianie pozapasmowe

W uwierzytelnianiu pozapasmowym wykorzystuje się dodatkowy kanał komunikacyjny — oprócz użytego do uwierzytelnienia żądania. Na przykład użytkownik podczas pierwszego logowania do witryny internetowej może otrzymać telefon w celu potwierdzenia operacji. W ten sposób utrudnia się ataki na konto, ponieważ haker musiałby przejąć kontrolę nad pozapasmowym kanałem komunikacji.

Uwierzytelnianie pozapasmowe ma różne formy. Wybór zależy od wymaganego poziomu bezpieczeństwa interakcji:

- Wiadomość e-mail o niedawnych newralgicznych operacjach.
- Wymóg potwierdzenia przed realizacją żądania. Może to być zwykła odpowiedź „tak” albo jednorazowe hasło czasowe.
- Kontakt z osobą odpowiedzialną za potwierdzanie operacji.

Umiejętnie wykorzystywane uwierzytelnianie pozapasmowe jest użytecznym narzędziem zwiększającym bezpieczeństwo systemu. Podobnie jak w przypadku innych mechanizmów, przy wyborze formy i częstości uwierzytelniania odpowiedniej dla żądania jest potrzebna pewna wiedza.

Logowanie SSO

Wraz ze wzrostem liczby wykorzystywanych przez użytkowników usług pojawiła się potrzeba oddzielenia ich od procesu uwierzytelniania. Takie rozdzielenie jest korzystne zarówno dla użytkowników, jak i usług:

- Użytkownicy muszą uwierzytelniać się tylko za pomocą jednej usługi.
- Poświadczenia są przechowywane w oddzielnej usłudze, objętej bardziej rygorystycznymi standardami bezpieczeństwa.
- Przechowywanie poświadczeń w mniejszej liczbie miejsc oznacza mniejsze ryzyko i łatwiejszą rotację.

Logowanie SSO (ang. *Single Sign-On*, logowanie jednokrotne) to dojrzała koncepcja. Użytkownik uwierzytelnia się w centralnym systemie i uzyskuje token przeznaczony do dalszej komunikacji z zabezpieczonymi usługami. Usługa po otrzymaniu żądania od użytkownika kontaktuje się z wykorzystaniem zabezpieczonego kanału z systemem uwierzytelniającym i weryfikuje token.

Jest to przeciwieństwo zdecentralizowanego uwierzytelniania. W sieci Zero Trust opartej na zdecentralizowanym uwierzytelnieniu płaszczyzna sterowania przesyła poświadczenia i zasady dostępu do płaszczyzny danych. Płaszczyzna danych sama decyduje, kogo, gdzie i kiedy uwierzytelnić, zgodnie z zasadami narzuconymi przez płaszczyznę sterowania. Czasami takie podejście preferuje się zamiast dojrzalszego logowania jednokrotnego, ponieważ nie wymaga uruchamiania dodatkowej usługi. Nie jest jednak zalecane ze względu na swoją złożoność.

Centralny system powinien sprawdzać tokeny SSO tak często, jak to możliwe. Każde odwołanie do płaszczyzny sterowania w celu autoryzacji tokenu daje możliwość odebrania dostępu lub zmiany poziomu zaufania źródła żądania.

Popularną implementacją jest usługa realizująca własne logowanie z uwierzytelnianiem SSO w tle. Ma ona jednak tę wadę, że umożliwia płaszczyźnie sterowania jednorazową autoryzację żądania i daje aplikacji swobodę przy podejmowaniu kolejnych decyzji. Nie należy jednak lekkomyślnie stosować tego podejścia, ponieważ kluczowym atrybutem sieci Zero Trust jest możliwość modyfikowania i anulowania zaufania.

Opcje SSO

Logowanie SSO jest znane od dawna i obecnie dostępnych jest wiele przeznaczonych do tego celu dojrzałych protokołów i technologii. Najpopularniejsze z nich to:

SAML

Język SAML (ang. *Security Assertion Markup Language*, oparty na znacznikach język negocjowania zabezpieczenia) to oparty na języku XML standard bezpiecznej wymiany danych uwierzytelniających i autoryzacyjnych.

WS-Federation

WS-Federation to protokół negocjowania warunków generowania tokenu, wykorzystywany w aplikacjach (jednostkach podrzędnych) i przez dostawców tożsamości. Poświadczenia przesyła się w roszczeniach (ang. *claim*), przy czym — jak na ironię — są to asercje SAML.

Kerberos

Dojrzały, bardzo skalowalny protokół szeroko stosowany w środowiskach korporacyjnych do jednokrotnego logowania wielu użytkowników. Jego konfiguracja jest jednak dość skomplikowana.

OAuth

Popularny protokół autoryzacyjny, również wykorzystywany do logowania SSO. Prostszy w konfiguracji niż SAML i Kerberos, ale nie działa równie dobrze z urządzeniami mobilnymi.

OpenID Connect (OIDC)

Warstwa tożsamości oparta na protokole OAuth 2.0. Tożsamość użytkownika jest weryfikowana z wykorzystaniem podstawowych informacji o jego profilu i wyniku uwierzytelnienia przez serwer autoryzacyjny. Proces jest interakcyjny, podobny do REST (ang. *Representational State Transfer*, zmiana stanu poprzez reprezentacje).

CAS

CAS (ang. *Central Authentication Service*, centralna usługa uwierzytelniania) to otwarty standard logowania SSO, stosowany w korporacyjnych aplikacjach internetowych i mobilnych. Umożliwia uwierzytelnianie i autoryzację w usługach obsługujących standard REST.

Bardzo ważne jest, aby w sieci Zero Trust uwierzytelnianie odbywało się na płaszczyźnie sterowania. Dlatego projektując systemy uwierzytelniania, należy jak największą odpowiedzialność powierzać tej płaszczyźnie i weryfikować autoryzację z maksymalną, ale rozsądną częstotliwością.

Tożsamość obciążenia

Tożsamość obciążenia to unikalny identyfikator przypisany oprogramowaniu lub usłudze (aplikacji, skryptowi, zadaniu cron, kontenerowi), który umożliwia uwierzytelnianie i daje dostęp do innych usług i zasobów. Różni się od tożsamości użytkownika cyklem życia i zastosowaniem. W miarę jak coraz więcej organizacji wdraża metodykę DevSecOps, cykl życia tożsamości obciążenia musi być automatyzowany, a jego wykorzystanie stale monitorowane.

Tożsamość obciążeń oferuje wielu dostawców usług chmurowych, dzięki czemu korzystanie z niej jest bardzo proste. Oto przykłady kilku z nich:

- Amazon (AWS) (*https://oreil.ly/2as--*),
- Microsoft (Azure) (*https://oreil.ly/bizky*),
- Google (GKE) (*https://oreil.ly/-82-L*).

SPIFFE

SPIFFE (ang. *Secure Production Identity Framework For Everyone*, bezpieczna platforma tożsamości produkcyjnej dla każdego, *https://oreil.ly/Dik0d*) to zbiór otwartych standardów tworzenia tożsamości usług i obciążeń w heterogenicznym środowisku organizacji. Specyfikacja SPIFFE dzieli się na trzy główne części:

SPIFFE ID

Standard wzajemnego identyfikowania się usług, wykorzystywany w identyfikatorach URI (ang. *Uniform Resource Identifier*, jednolity identyfikator zasobów).

SPIFFE Verifiable Identity Document (SVID)

Standard kodowania identyfikatorów SPIFFE w dokumencie weryfikowalnym kryptograficznie.

Interfejs API obciążenia

Specyfikacja interfejsu API służącego do wydawania i pobierania identyfikatorów SVID.

Standard SPIFFE koncentruje się głównie na specyfikacji i platformie, a SPIRE (ang. *SPIFFE Runtime Environment*, środowisko wykonawcze SPIFFE, *https://oreil.ly/Z1_Vw*) definiuje produkcyjny interfejs API, atestację węzłów i obciążeń, bezpieczne wydawanie identyfikatorów SVID obciążeń oraz weryfikację ich zgodności z określonymi wymaganiami. Więcej informacji o architekturze, implementacjach i praktycznych zastosowaniach standardu SPIRE można znaleźć na stronie *https://oreil.ly/yRxco*.

Standardy SPIFFE i SPIRE są projektami fundacji Cloud Native Computing Foundation (*https://oreil.ly/bmO2f*, *https://oreil.ly/Awqk0*).

W kierunku lokalnego uwierzytelniania

Uwierzytelnianie lokalne rozszerzone na usługi zewnętrzne to kolejny mechanizm zyskujący coraz większą popularność. Użytkownik uwierzytelnia swoją obecność za pomocą zaufanego urządzenia, a następnie urządzenie potwierdza tożsamość z zastosowaniem zewnętrznej usługi. Otwarte standardy, takie jak FIDO Alliance UAF, wykorzystują asymetryczną kryptografię i lokalne systemy uwierzytelniania urządzeń (np. hasła i dane biometryczne) do przenoszenia zaufania z dużej liczby usług do stosunkowo niewielu punktów końcowych kontrolowanych przez użytkownika.

Standard UAF przypomina pod pewnymi względami menedżera haseł. Jednak zamiast haseł przechowuje klucze prywatne. Usługa uwierzytelniająca otrzymuje klucz publiczny użytkownika, za pomocą którego potwierdza, że użytkownik posiada klucz prywatny. Przeniesienie uwierzytelniania do inteligentnego urządzenia lokalnego niesie szereg korzyści:

- system wyzwań i odpowiedzi chroni przed atakami odtworzeniowymi,
- ochrona przed atakami typu „człowiek pośrodku", jeśli usługa uwierzytelniająca odmówi podpisania wyzwania (chyba że pochodzi ono z tej samej domeny, którą odwiedza użytkownik),
- brak możliwości ponownego wykorzystania poświadczeń, ponieważ są one w prosty sposób generowane dla poszczególnych usług.

Uwierzytelnianie i autoryzowanie grupy użytkowników

Niemal każdy system obsługuje pewien niewielki zbiór operacji i żądań, które należy ściśle chronić. Poziom ryzyka, jakie można tolerować, zależy od sytuacji, jednak w praktyce nie ma dolnej granicy.

Jednym z wyzwań jest ograniczone zaufanie, jakie można pokładać w człowieku. W rzeczywistym świecie, aby autoryzować szczególnie wrażliwe działanie, trzeba uzyskać zgodę kilku osób. W świecie cyfrowym osiąga się to na kilka sposobów, a co najważniejsze, można uzyskać kryptograficzną gwarancję.

Algorytm SSS

Algorytm SSS (ang. *Shamir's Secret Sharing*, udostępnianie poufnych informacji metodą Shamira) służy do rozpowszechniania pojedynczej poufnej informacji w grupie osób. Informację dzieli się na *n* części, które się następnie rozpowszechnia (rysunek 6.2). W zależności od konfiguracji algorytmu podczas dzielenia informacji na części, do ponownego jej złożenia jest potrzebnych *k* części. W przypadku dużych ilości danych algorytmu nie stosuje się bezpośrednio, tylko dzieli się i rozpowszechnia symetryczny klucz szyfrujący, ponieważ poufna informacja musi być mniejsza niż pewne dane wykorzystywane w algorytmie.

```
~ $ echo 'this is a secret' | ssss-split -n 5 -t 2
Generating shares using a (2,5) scheme with dynamic security level.
Enter the secret, at most 128 ASCII characters: Using a 128 bit security level.
1-4054162f42f328c2ecbff990e9e1996f
2-93285deac4d6406cde841b05b350f61f
3-22039b5646ca98093092ba897ac02cb0
4-35d0ca61c89c9130baf3de2f06322866
5-84fb0cdd4a80495554e57fa3cfa2f2c9
~ $ ssss-combine -t 2
Enter 2 shares separated by newlines:
Share [1/2]: 5-84fb0cdd4a80495554e57fa3cfa2f2c9
Share [2/2]: 4-35d0ca61c89c9130baf3de2f06322866
Resulting secret: this is a secret
```

Rysunek 6.2. Przykład użycia algorytmu SSS

Implementacja tego algorytmu w systemie Linux nosi nazwę *ssss* (*https://oreil.ly/Y4spd*). W innych systemach operacyjnych i językach programowania są dostępne odpowiednie aplikacje i biblioteki.

Projekt Red October

Projekt Red October to kolejna implementacja uwierzytelniania grupowego w celu udzielenia dostępu do danych (*https://oreil.ly/qVKbV*). Usługa internetowa szyfruje z wykorzystaniem kryptografii asymetrycznej dane w taki sposób, że do ich odszyfrowania jest potrzebna grupa osób. Na serwerze nie zapisuje się zaszyfrowanych danych, tylko klucze prywatny i publiczny zaszyfrowane z użyciem hasła użytkownika.

Przed wysłaniem danych do zaszyfrowania jest generowany losowy klucz, który jest następnie szyfrowany z użyciem kombinacji unikatowych kluczy użytkownika zgodnie z żądaną zasadą odblokowania. W najprostszym przypadku użytkownik szyfruje niektóre dane tak, aby do ich odszyfrowania trzeba było zaangażować dwie osoby z grupy. Klucz jest więc podwójnie szyfrowany z wykorzystaniem pary unikatowych kluczy użytkownika.

Podpisywanie głównej strefy DNS

Ciekawym przykładem procedury uwierzytelniania grupowego jest ceremonia podpisania głównej strefy DNS. Jej celem jest wygenerowanie głównych kluczy, na których opiera się zaufanie do usługi DNSSEC. W razie przechwycenia klucza głównego cała usługa DNSSEC utraciłaby wiarygodność. Dlatego w celu zminimalizowania tego zagrożenia organizowana jest ceremonia podpisania klucza głównego.

Pierwsza ceremonia odbyła się 16 czerwca 2010 roku, a kolejne mają miejsce co kwartał. W każdej bierze udział siedmiu uczestników, z których każdy odgrywa inną rolę. Zakładając, że wskaźnik nieuczciwości każdego uczestnika jest równy 5%, szansa przechwycenia klucza jest równa jednej na milion. W celu zorganizowania ceremonii tworzy się tajny dokument proceduralny. Klucz cyfrowy jest chroniony za pomocą modułu HSM, skanerów biometrycznych i odseparowanych systemów. Na koniec generuje się i podpisuje nową parę kluczy (prywatny i publiczny) i w ten sposób gwarantuje zaufanie do internetu na kolejny kwartał.

Na stronie firmy Cloudflare można znaleźć więcej informacji o ceremonii podpisania kluczy (*https://oreil.ly/SamQZ*), a na stronie organizacji IANA — materiały na temat odbytych ceremonii (*https://oreil.ly/MR2tm*).

Widzisz coś, powiedz coś

Zarówno użytkownicy, jak i urządzenia muszą dbać o bezpieczeństwo sieci Zero Trust. W organizacjach tradycyjnie tworzy się wyznaczone do tego celu zespoły. Przyjęło się, że to one ponoszą wyłączną odpowiedzialność za bezpieczeństwo systemu. Aby je zagwarantować, zespoły muszą sprawdzać wszystkie wprowadzane zmiany. Takie podejście przyczynia się jednak do antagonizmów między zespołem a resztą organizacji i w efekcie do pogorszenia bezpieczeństwa.

Lepszą strategią jest budowanie kultury współpracy. Należy zachęcać użytkowników do informowania, że coś, co robią albo widzą, wygląda dziwnie lub niebezpiecznie, nawet jeżeli wydaje się mało istotne. Takie dzielenie się wiedzą daje znacznie lepszy kontekst zagrożeń, przed którymi zespół ds. bezpieczeństwa broni organizację. Zgłoszenie phishingowej wiadomości e-mail bez wchodzenia z nią w interakcję umożliwia zespołowi sprawdzenie, czy haker usiłuje dostać się do sieci. Natychmiast należy informować o zagubieniu lub kradzieży urządzenia. Zespół ds. bezpieczeństwa może wysyłać użytkownikom powiadomienia o dowolnej porze o zaginięciach urządzeń.

Zespół ds. bezpieczeństwa, odpowiadając na informacje lub ostrzeżenia od użytkowników, musi mieć świadomość, że jego reakcja na incydent ma duży wpływ na organizację. Użytkownik zganiony za zagubienie urządzenia będzie w przyszłości mniej skłonny do informowania o takim fakcie. Również zgłoszenia fałszywych alarmów w środku nocy powinny być przyjmowane życzliwie, aby użytkownicy nie trwali w niepewności. Jeżeli to możliwe, lepiej zgłaszać za dużo niż za mało.

Sygnały zaufania

Historia aktywności użytkowników jest bogatym źródłem danych pozwalających określać wiarygodność ich bieżących działań. Można zbudować system, który bada aktywność, buduje model dopuszczalnego zachowania, porównuje bieżące zachowanie z modelem i wylicza oceny zaufania użytkowników.

Wzorce zachowań użytkowników są zazwyczaj przewidywalne. Nikt nie jest w stanie uwierzytelniać się kilka razy na sekundę. Jest też mało prawdopodobne, aby ktoś uwierzytelniał się setki razy. Tego typu zachowania są niezwykle podejrzane. Można im przeciwdziałać przez stosowanie odpowiednich metod (np. CAPTCHA, czyli zapytania, na które jest w stanie odpowiedzieć tylko człowiek) lub blokowanie kont. Aby ograniczyć liczbę fałszywych alarmów, należy ustawiać wysokie wartości progowe. Uwzględnienie aktywności użytkownika w ogólnej ocenie zaufania pozwala wykrywać podejrzane, choć nie ewidentnie złe, zachowania.

Analizując wzorce zachowań, nie należy ograniczać się do prób uwierzytelnienia. Na podstawie wzorców korzystania z aplikacji również można wykrywać złe intencje. Większość użytkowników ma zazwyczaj dość ograniczone role w organizacji i potrzebuje dostępu wyłącznie do części danych. Aby zwiększyć bezpieczeństwo, powinno się odbierać użytkownikom prawa dostępu do danych, chyba że rzeczywiście potrzebują ich w swojej pracy. Jednak tego typu restrykcyjne działania ograniczają zdolność organizacji do szybkiego reagowania na wyjątkowe sytuacje. Administratorzy systemów to grupa użytkowników, którzy mają szeroki dostęp. Takie podejście osłabia mechanizm obrony. Zamiast wybierać pomiędzy dwiema skrajnościami, można oceniać aktywność użytkownika, a następnie na podstawie wyniku określać, czy można mu zaufać i udzielić dostępu do szczególnie newralgicznego zasobu. Twarde zasady w systemie muszą jednak obowiązywać, aby w mniej jasnych przypadkach można było weryfikować wiarygodność użytkowników na podstawie zarejestrowanych aktywności.

Kolejnym przydatnym narzędziem do oceny wiarygodności użytkownika jest lista zidentyfikowanych źródeł szkodliwego ruchu, taka jak publikowana przez organizację Spamhaus. Ruch pochodzący z takiego źródła może być sygnałem zagrożenia użytkownika i próby wykorzystania jego tożsamości.

Geolokalizacja jest kolejnym ważnym czynnikiem pozwalającym określić wiarygodność użytkownika. Aby sprawdzić, czy jego obecna lokalizacja jest nietypowa, należy porównać ją z innymi, w których przebywał wcześniej. Czy urządzenie użytkownika nagle pojawiło się w nowej lokalizacji, do której nie można dotrzeć w rozsądnym czasie? Jeśli użytkownik ma kilka urządzeń, czy jest rozbieżność w ich lokalizacjach? Geolokalizacja może być niedokładna lub myląca, dlatego nie powinno się do niej przywiązywać nadmiernej wagi. Czasami użytkownicy zostawiają urządzenia w domach, jak również bazy danych geolokalizacyjnych mogą być niedokładne.

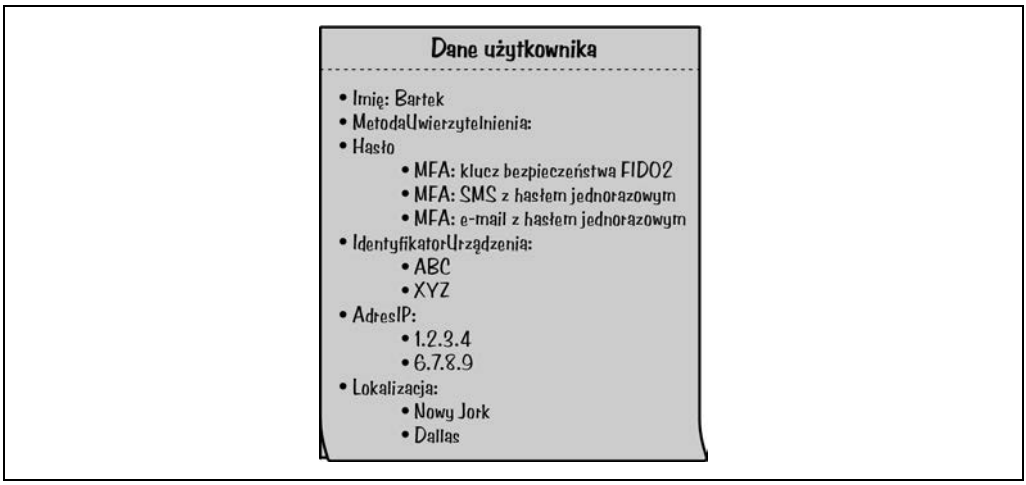
Przykładowy scenariusz

Przeanalizujmy scenariusz, w którym użytkownik wysyła żądanie przyznania dostępu do newralgicznego zasobu. Rysunki 6.3 – 6.6 przedstawiają kluczowe komponenty, a poniżej znajduje się analiza żądania.

Przypadek: użytkownik chce uzyskać wgląd w poufny raport finansowy

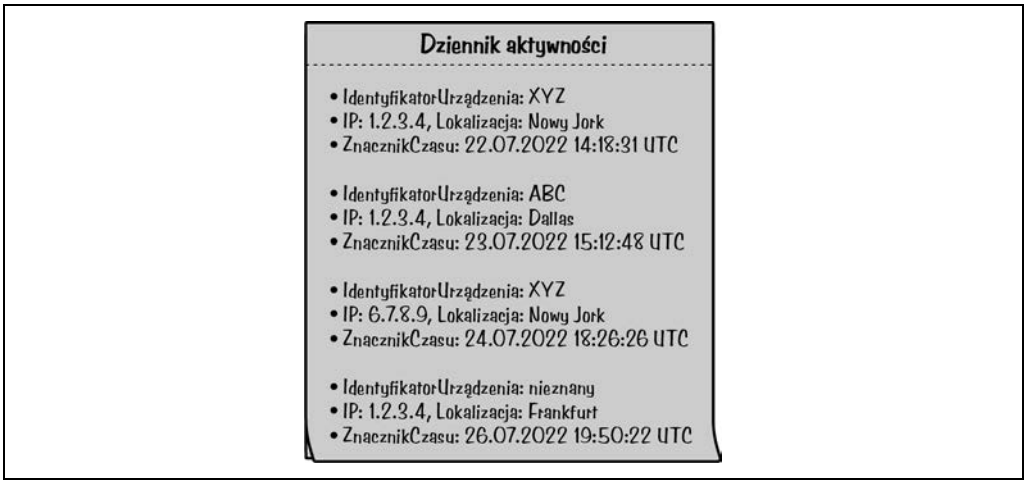
Oto co wiadomo o żądaniu użytkownika:

- Dotyczy poufnego raportu finansowego.
- Użytkownik korzysta ze służbowego laptopa o identyfikatorze *ABC*, spełniającego wszystkie zasady w organizacji.
- Użytkownik uwierzytelnił się metodą MFA za pomocą hasła i wiadomości SMS.
- Użytkownik wysłał żądanie w godzinach pracy.

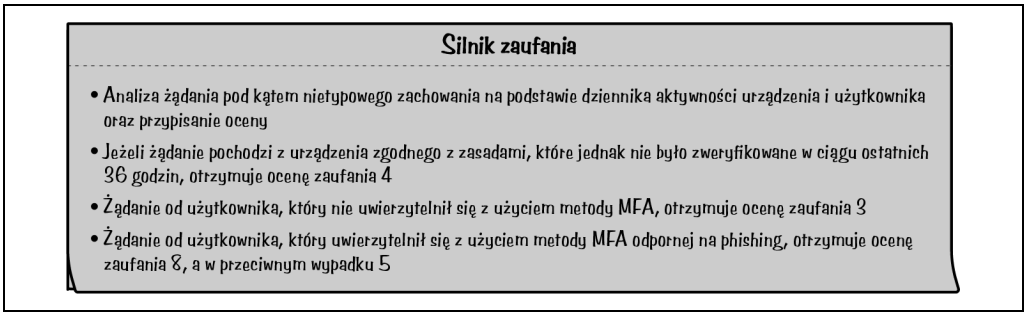


Rysunek 6.3. Tożsamość użytkownika obejmuje m.in. jego imię, hasło i metody uwierzytelnienia MFA, w tym klucze bezpieczeństwa FIDO2[1], wiadomości SMS i e-mail

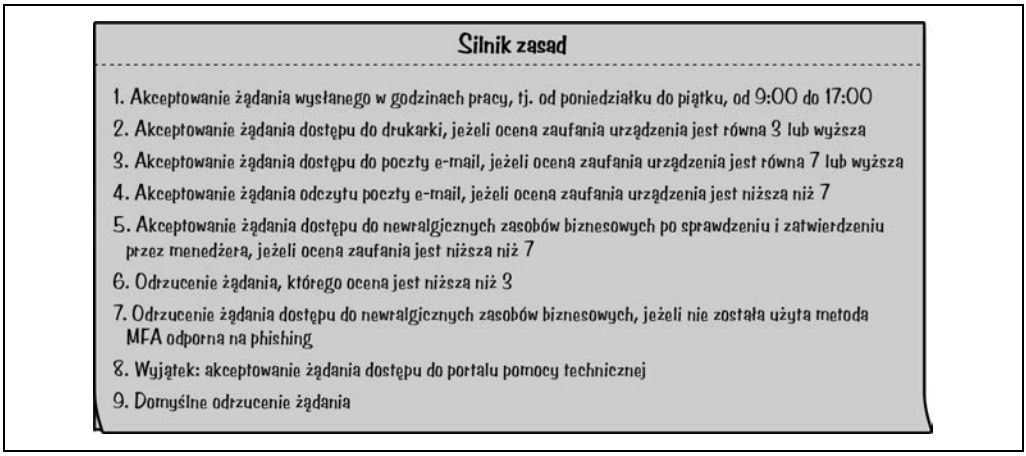
[1] FIDO (ang. *Fast Identity Online*, szybka identyfikacja online, *https://oreil.ly/cZJI4*) to otwarty standard bezpiecznego uwierzytelniania bez użycia hasła.



Rysunek 6.4. Dziennik aktywności urządzeń, wykorzystywany do audytowania, wykrywania nietypowych zachowań i wyliczania oceny zaufania



Rysunek 6.5. Silnik zaufania wylicza na podstawie dynamicznych i statycznych reguł ocenę zaufania, którą przypisuje żądaniu udzielenia dostępu



Rysunek 6.6. Silnik zasad podejmujący ostateczną decyzję o zatwierdzeniu lub odrzuceniu żądania udzielenia dostępu do zasobu

Analiza żądania

1. Żądanie użytkownika (akcja: wyświetlenie raportu, identyfikator urządzenia: ABC, uwierzytelnienie: hasło/MFA (SMS), lokalizacja: Dallas, IP: 6.7.8.9, czas: 28.07.2022 11:00) trafia do komponentu wykonawczego.
2. Komponent wykonawczy przesyła żądanie do silnika zasad do akceptacji.
3. Silnik zasad odbiera żądanie i komunikuje się z silnikiem zaufania w celu uzyskania oceny.
4. Silnik zaufania ocenia żądanie:
   - W dzienniku aktywności urządzeń wykrywa podejrzane działanie. Adres IP wydaje się adresem anonimowego serwera proxy, a ponadto urządzenie jest nieznane. Przypisuje niską ocenę zaufania, równą 3.
   - Użytkownik uwierzytelnił się metodą MFA z wiadomością SMS, więc uzyskał ocenę 5.
   - Urządzenie jest zgodne z zasadami organizacji i zostało zweryfikowane w ciągu ostatnich 36 godzin.
   - Silnik zaufania wylicza średnią ocenę 4 i wysyła ją do silnika zasad.
5. Silnik zasad odbiera ocenę od silnika zaufania.
6. W celu autoryzacji żądania silnik zasad sprawdza jego zgodność z regułami:
   - Pierwsza reguła dopuszcza żądanie, ponieważ zostało wysłane w godzinach pracy.
   - Reguły 2 – 4 nie odnoszą się do bieżącego żądania, które dotyczy newralgicznego zasobu (raportu finansowego).
   - Piąta reguła odnosi się do bieżącego żądania, ponieważ jego ocena zaufania jest niższa niż 7, a ponadto dotyczy ono dostępu do raportu finansowego. Wymagana jest akceptacja menedżera.
   - Szósta reguła nie odnosi się do żądania, ponieważ jego ocena jest wyższa niż 3.
   - Siódma reguła odnosi się do żądania, ponieważ dotyczy ono dostępu do newralgicznego raportu finansowego. Ponadto użytkownik uwierzytelnił się metodą MFA z wiadomością SMS, a nie metodą odporną na phishing, na przykład z użyciem klucza bezpieczeństwa FIDO2. Wynikiem zastosowania tej reguły jest odrzucenie żądania.
   - Ósma reguła nie dotyczy żądania, ponieważ nie zostało wysłane do pomocy technicznej. Jest egzekwowana tylko wtedy, gdy żadna z wcześniejszych reguł nie zostanie zastosowana.
   - Dziewiąta, domyślna reguła nie dotyczy żądania, ponieważ jest egzekwowana tylko wtedy, gdy żadna z wcześniejszych reguł nie zostanie zastosowana.
   - Silnik zasad wysyła do komponentu wykonawczego odmowę wykonania operacji z sugestią, aby użytkownik zamiast wiadomości SMS zastosował metodę MFA odporną na phishing, na przykład klucz bezpieczeństwa FIDO2. Ponadto wymagana jest akceptacja menedżera, ponieważ ocena zaufania jest bardzo niska, a żądanie dotyczy newralgicznego zasobu.
7. Komponent wykonawczy odbiera wynik od silnika zasad i blokuje użytkownikowi dostęp do raportu. Dołącza też zrozumiałe dla użytkownika uzasadnienie decyzji oraz informację o ograniczonym dostępie po zastosowaniu metody MFA odpornej na phishing (np. klucza bezpieczeństwa FIDO2) i konieczności zaakceptowania żądania przez menedżera.

Podsumowanie

W tym rozdziale skupiliśmy się na budowaniu zaufania do użytkowników systemu. Pokazaliśmy, jak definiuje się tożsamość i jak ważne jest posiadanie uprawnień, do których można się odwoływać podczas sprawdzania tożsamości użytkownika w systemie. Aby użytkownik miał tożsamość, musi być wprowadzony do systemu, dlatego omówiliśmy idealne metody tworzenia tożsamości. Tożsamość trzeba gdzieś przechowywać, a system jest bardzo pożądanym celem dla hakerów. Opisaliśmy, jak bezpiecznie przechowywać dane, jak ważne jest ograniczenie ilości danych przechowywanych w jednym miejscu oraz jak przetwarzać tożsamość użytkowników, którzy dołączają do organizacji i ją opuszczają.

Po pokazaniu, jak definiować i przechowywać wiarygodną tożsamość, skupiliśmy się na uwierzytelnianiu użytkowników, którzy już ją mają. Uwierzytelnianie może być uciążliwe, dlatego opisaliśmy, kiedy należy to robić. Użytkowników nie można przytłaczać żądaniami uwierzytelnienia, ponieważ zwiększa się w ten sposób prawdopodobieństwo przypadkowego uwierzytelnienia w szkodliwej usłudze. Dlatego niezwykle istotne jest znalezienie właściwej równowagi.

Istnieje wiele sposobów uwierzytelniania użytkowników, dlatego zagłębiliśmy się tylko w podstawowe pojęcia. Omówiliśmy kilka mechanizmów, które są obecnie używane. Przyjrzeliśmy się także kilku innym, które pojawiają się na rynku w miarę ewoluowania zagrożeń.

Często zwiększenie zaufania użytkowników do systemu wymaga wdrożenia procedur, w których kilku użytkowników bierze udział w osiągnięciu określonego celu. Omówiliśmy grupowe systemy uwierzytelniania i autoryzacji, między innymi „reguły dwuosobowe", które można wykorzystać do zabezpieczenia newralgicznych danych. Przedyskutowaliśmy również budowanie kultury współpracy w organizacji przez zachęcanie użytkowników do zgłaszania wszelkich podejrzanych działań.

W sieci Zero Trust można na podstawie dzienników aktywności użytkowników tworzyć ich profile i wykorzystywać je do oceniania nowych działań. Wymieniliśmy kilka sygnałów, które można zastosować do budowania takich profili.

W następnym rozdziale zajmiemy się budowaniem zaufania do aplikacji.

Ta książka to lektura obowiązkowa dla początkujących i profesjonalistów!

Karan Dwivedi, kierownik do spraw inżynierii bezpieczeństwa w Google

Zapewnienie bezpieczeństwa zasobów systemu sieciowego jest dla firm, organizacji i instytucji zadaniem absolutnie kluczowym. W praktyce tradycyjne, scentralizowane zapory sieciowe często okazują się niewystarczające, a ich konfiguracja bywa kłopotliwa. Problemy z dostępem VPN i złożoność implementacji protokołu TLS w wielu aplikacjach dodatkowo komplikują sytuację. W obliczu tych wyzwań i konieczności spełniania norm bezpieczeństwa model Zero Trust to doskonałe rozwiązanie, które skutecznie podniesie poziom ochrony zasobów sieciowych.

Dzięki tej książce nauczysz się stosować w praktyce zasady Zero Trust: nic nie jest oczywiste, a każde żądanie dostępu ma być sprawdzone i autoryzowane. Poznasz najważniejsze koncepcje tego modelu, takie jak silnik zaufania, silnik zasad czy agregat kontekstowy. Dowiesz się, jak budować zaufanie między różnymi elementami sieci, bazując na istniejących technikach. Spojrzysz na model Zero Trust z punktu widzenia hakera, a następnie zagłębisz się w szczegóły architektur, standardów i struktur Zero Trust opracowanych przez organizacje NIST, CISA czy DoD. W ten sposób zrozumiesz model zerowego zaufania z perspektywy wiodących instytucji w branży. Przeanalizujesz też wpływ sztucznej inteligencji, komputerów kwantowych i technologii ochrony prywatności na model Zero Trust.

W książce:

- najważniejsze koncepcje modelu Zero Trust
- czym jest bezpieczeństwo systemu w modelu Zero Trust
- budowa sieci Zero Trust w środowisku produkcyjnym
- przykłady przejścia organizacji na model Zero Trust
- architektury, normy i struktury Zero Trust

Razi Rais od lat buduje odporne na ataki systemy w Microsofcie i aktywnie udziela się w radzie doradczej GIAC.

Christina Morillo realizuje złożone projekty w dziedzinie bezpieczeństwa. Bierze udział w inicjatywach związanych z cyberbezpieczeństwem.

Evan Gilman jest współtwórcą otwartego oprogramowania, prelegentem i autorem.

Doug Barth pracował jako inżynier do spraw infrastruktury i inżynier do spraw produktu. Budował systemy monitoringu i sieci kratowe.

Zero Trust to nie tylko strategia. To sposób myślenia.

Ann Johnson, wiceprezes do spraw bezpieczeństwa w Microsofcie

Helion
helion.pl
HELION S.A.
ul. Kościuszki 1c
44-100 Gliwice
tel.: 32 230 98 63
helion@helion.pl
KOD KORZYŚCI
Sięgnij po więcej!
ISBN 978-83-289-1481-0
9 788328 914810
Cena: 77,00 zł